
Lakehouse Based Medallion Methodology in Construction Supervision

André João de Vilhena Costa

Dissertation

Master in Modelling, Data Analysis and Decision Support Systems

Supervised by:

PhD Bruno Miguel Delindro Veloso

2026

Acknowledgments

I would like to express my sincere gratitude to FEP, to my colleagues, and to the MADSAD Professors for their teaching, guidance, and availability throughout the programme.

I am especially grateful to Professor Bruno Veloso, supervisor of this thesis, and to António Silva, R&D at DevScope and adviser to this work.

I am especially grateful to Professor Bruno Veloso, supervisor of this thesis, for his guidance and feedback, deep reorganisation of this document and revisions that greatly improved its quality, and to António Silva, RD at DevScope and adviser to this work, for his support and valuable advice throughout the project.

I also wish to thank TPF, Arq. e Eng. SA and Applus+ for their support

To my family and friends, thank you for your understanding regarding the time and attention that this work took from us, and that would otherwise have been yours.

To all of you, my sincere thanks.

Abstract

Construction supervision in civil engineering projects generates large volumes of information, including inspection records, payment documents, photographs, health and safety records, quality reports, schedules, meteorological data, correspondence, and cost information. In practice, these data are often dispersed across different formats, systems, and repositories, making them difficult to organise, analyse, trace, and reuse. As construction projects become increasingly data-intensive, supervision activities require structured methods capable of transforming fragmented records into reliable evidence for monitoring, reporting, contractual interpretation, and decision support.

The problem addressed in this dissertation is the absence of an integrated and reusable approach for managing construction supervision data from acquisition to analysis. Much of this information is stored in unstructured or semi-structured formats, such as PDF and Word, Excel and Microsoft Project files, Outlook correspondence, images, and external data sources. This fragmentation limits the ability to consolidate records, compare information across sources, identify risks, support financial analysis, and produce consistent reports. This dissertation explores scalable methods that can be adapted to different projects and supervision contexts.

The dissertation proposes a methodology inspired by lakehouse architectures and medallion style data organisation. A lightweight local prototype is developed in which supervision data are progressively organised into Bronze, Silver, and Gold layers. Raw evidence is preserved in its original form, intermediate datasets are extracted, cleaned, standardised, and harmonised, and curated outputs are prepared for analysis, visualisation, reporting, and Artificial Intelligence assisted interpretation. The methodology includes document and inspection record consolidation, SQL based data structuring, meteorological analysis using data from the Instituto Português do Mar e da Atmosfera Application Programming Interface, contract price revision and financial schedule analysis. It also includes an image-based risk analysis prototype combining YOLOv8 computer vision and a Claude-based Large Language Model pipeline.

The results show the feasibility of combining data engineering, data analysis, computer vision, Large Language Models, and Application Programming Interface based services within a construction inspection workflow. The proposed approach transforms dispersed supervision records into structured, traceable, and reportable evidence, supporting monitoring, risk identification, financial analysis, contractual interpretation, and monthly reporting. The work provides a reusable foundation for improving the organisation and exploitation of supervision data in civil engineering projects.

Keywords: Construction Supervision, data engineering, Lakehouse, Medallion Architecture, Computer Vision, Large Language Models

Resumo

A fiscalização de obras em projetos de engenharia civil gera grandes volumes de informação, incluindo registos de inspeção, documentos de pagamento, fotografias, registos de segurança e saúde, relatórios de qualidade, cronogramas, dados meteorológicos, correspondência e informação de custos. Estes dados encontram-se frequentemente dispersos por diferentes formatos, sistemas e repositórios, o que dificulta a sua organização, análise, rastreabilidade e reutilização. À medida que o volume de dados em projetos de construção, cresce, exigem-se métodos estruturados capazes de transformar registos fragmentados em evidência fiável para monitorização, reporte, interpretação contratual e apoio à decisão.

Esta dissertação procura cobrir para uma abordagem integrada e reutilizável para a gestão de dados de fiscalização de obras, desde a aquisição até à análise. Grande parte desta informação encontra-se armazenada em formatos não estruturados ou semiestruturados, tais como ficheiros PDF e Word, Excel e Microsoft Project, correspondência Outlook, imagens e fontes de dados externas. Esta fragmentação limita a capacidade de consolidar registos, comparar informação entre diferentes fontes, identificar riscos, apoiar a análise financeira e produzir relatórios consistentes. Esta dissertação explora métodos escaláveis que podem ser adaptados a diferentes projetos e contextos de fiscalização.

A dissertação propõe uma metodologia inspirada em arquiteturas lakehouse e organização de dados em estilo medallion. É desenvolvido um protótipo, simples, no qual os dados de fiscalização são progressivamente organizados em camadas Bronze, Silver e Gold. Os dados originais são preservados em bruto, de onde os dados intermédios são extraídos, limpos, normalizados e harmonizados, que são então preparados para análise, visualização, reporte e interpretação assistida por Inteligência Artificial. A metodologia inclui a consolidação de documentos e registos de inspeção, a estruturação em bases de dados SQL, a análise meteorológica com recurso a dados da Application Programming Interface do Instituto Português do Mar e da Atmosfera, a revisão de preços contratual e análise do plano financeiro. Inclui ainda um protótipo de análise de risco baseado em imagem, que combina visão computacional com YOLOv8 e Large Language Model baseado em Claude.

Os resultados mostram a viabilidade de combinar engenharia de dados, análise de dados, visão computacional, Large Language Models e serviços baseados em Application Programming Interface num fluxo de dados na fiscalização de obras. A abordagem proposta transforma registos de fiscalização dispersos em evidência estruturada, rastreável e reportável, apoiando a monitorização, a identificação de riscos, a análise financeira, a interpretação contratual e o reporte mensal, melhorando a organização e a exploração de dados em fiscalização de projetos de engenharia civil.

Palavras-chave: Fiscalização de Obras, Engenharia de Dados, Lakehouse, Arquitetura Medallion, Visão Computacional, Large Language Models

Contents

Acknowledgements	iii
Abstract	v
Resumo	vii
1 Introduction	1
1.1 Context and Motivation	1
1.2 Problem Definition	2
1.3 Objectives and Research Questions	2
2 Theoretical Foundations and Methods	5
2.1 Data Analytics Process Models and CRISP-DM	5
2.2 Data Pipeline Architecture and Lakehouse/Medallion Concepts	6
2.3 Bronze Layer Source Synchronisation	7
2.4 Silver Layer Data Extraction and Structuring	8
2.5 Gold Layer Analytical Outputs and Reporting	9
2.6 Outlook and Email Correspondence Analysis	10
3 Literature Review	13
3.1 Digital Transformation and Data-Driven Supervision	13
3.2 CRISP-DM and Complementary Data Analytics Governance References	14
3.3 Modern Data Architectures and Local Pipeline Adaptation	15
3.4 Medallion Architecture, File Synchronisation and Data Governance	15
3.5 Document, Correspondence and Inspection Record Analytics	17
3.6 Document Extraction from PDF, Word and Excel sources	17
3.7 Contract Price Revision and Financial Schedule Analysis	18
3.8 Computer Vision and Image-Based Risk Analysis	20
3.9 Multimodal LLM Based Safety Interpretation and Fusion	21
3.10 Analytical Outputs, Decision Support and AI-Assisted Reporting	22
3.11 Summary	23
4 Proposed Methodology and Computational Implementation	25
4.1 CRISP-DM, Medallion Architecture and Governance Principles	25
4.2 Bronze Synchronisation	26
4.3 Outlook Analytics	27
4.3.1 Bronze Layer: Outlook Extraction and Structured Preservation	28
4.3.2 Silver Layer: Exploratory Data Analysis and Validation	29
4.3.3 Gold Layer: Interactive Dashboard and Reporting Outputs	29
4.3.4 Assessment of the Workflow	29

4.3.5	Subject Sensitivity and Network Analysis	30
4.4	Meteorological Data Analytics	30
4.4.1	Purpose and General Architecture	30
4.4.2	Study Location, API Access and Station Selection	31
4.4.3	Database Structure and Station Selection Model	32
4.4.4	Data Quality, Continuous Records and Period Aggregation	33
4.4.5	Classification Criteria for Interpretative Graphs	34
4.4.6	Analytical Interpretation and Methodological Limitations	35
4.5	Inspection Records Consolidation	36
4.5.1	Source Data and Folder Structure	37
4.5.2	Reading and Extracting Information from Word Reports	40
4.5.3	Extraction of Resources from Embedded Excel Objects	41
4.5.4	Monthly Consolidation into Excel Matrices	41
4.5.5	Daily Global Resource Summaries	42
4.5.6	Comparison with Contractor-Proposed Resource Loads	42
4.6	Contract Price Revision	44
4.6.1	Implementation context	44
4.6.2	Formula-type selection and calculation logic	44
4.6.3	Computational procedure adopted	45
4.6.4	Treatment and validation of indices	46
4.6.5	Price revision workflow	46
4.6.6	Treatment of monthly and cumulative payment certificates	46
4.6.7	Contribution to contract management	47
4.7	Financial Schedule Analysis	47
4.7.1	Planned and actual financial execution	47
4.7.2	Computational workflow	47
4.8	AI Image Risk Analyser	49
4.9	API Based LLM Assisted Monthly Reporting	51
4.10	Overall Integrated Data Engineering and Reporting Workflow	53
5	Results	57
5.1	Outlook Analysis	57
5.2	Meteorological Analysis	62
5.3	Daily Inspection Records	64
5.4	Price Revision and Financial Analysis	67
5.5	Risk Image Analysis	70
5.6	API Based LLM Assisted Monthly Reporting	73
5.7	Response to the Research Objectives	74
6	Conclusions	75
	Bibliography	77
	Appendices	i
A	Haversine formula of earth surface distance	i

B	Bronze Layer Source Synchronisation	ii
C	Outlook extraction code	vii
D	Meteorological Data System	xiv
E	Daily reports	xxii
F	Price revision	xxvii
G	Financery chronogram	xxxvi
H	AI image risk analiser	xlii
I	API-Based LLM-Assisted Monthly Reporting	lii
J	AI Image Risk Analyser Report	lv
K	Monthly Construction Supervision Report – 2026-05	lxiv

Glossary

ACID	Atomicity, Consistency, Isolation and Durability
AI	Artificial Intelligence
API	Application Programming Interface
BIM	Building Information Modelling
CIFE	Comissão de Índices e Fórmulas de Empreitadas
CSV	Comma-Separated Values
CV	Computer Vision
CRISP-DM	Cross-Industry Standard Process for Data Mining
DL	Deep Learning
ELT	Extract, Load, Transform
ETL	Extract, Transform, Load
HTML	HyperText Markup Language
IMPIC	Instituto dos Mercados Públicos, do Imobiliário e da Construção
IPMA	Instituto Português do Mar e da Atmosfera
JSON	JavaScript Object Notation
JSONL	JSON Lines
LLM	Large Language Model
YOLO	You Only Look Once
ML	Machine Learning
OCR	Optical Character Recognition
PDF	Portable Document Format
PNG	Portable Network Graphics
PPE	Personal Protective Equipment
RAG	Retrieval-Augmented Generation
RAG	Retrieval-Augmented Generation
RDF	Registos Diários da Fiscalização
SQL	Structured Query Language

List of Figures

2.1	Lakehouse architecture as an integration of data warehouse and data lake capabilities. (D. Lee, Wentling, Haines, and Babu (2025)).	6
4.1	Outlook extraction and analytical exploration workflow.	28
4.2	Daily inspection record, pages 1 and 2.	38
4.3	Daily inspection record, pages 3 and 4.	39
4.4	Image risk analyser workflow combining computer vision, LLM interpretation, fusion logic, and report generation.	50
4.5	Monthly master pipeline workflow with evidence synthesis, figure selection, data-backed commentary, and API-based LLM reporting.	53
4.6	Global process flow from domain analytics pipelines to API-based LLM monthly reporting.	54
5.1	General Outlook correspondence analysis outputs.	58
5.2	Scatter plot date vs hour by keyword	59
5.3	HTML Outlook email selector.	59
5.4	Ouputs of of emails after selection through folder and sender.	60
5.5	Filtered by folder and sender hours vs weekday	60
5.6	Outlook emails network analysis	61
5.7	Monthly meteorological report based on continuous hourly records.	63
5.8	Labour force extracted from the daily reports	64
5.9	Equipments on site extracted from daily reports	65
5.10	Variation of personnel and equipment from daily report	65
5.11	Difference between actual daily report quantities and bid-proposed quantities	66
5.12	Activities per week	66
5.13	Activities per month	67
5.14	Financial schedule evolution graphical representation	68
5.15	Price revision evolution	69
5.16	Examples of construction supervision images used in the AI image risk analysis workflow	71
C.1	Entity relationship diagram - Outlook extraction database	ix
D.1	Entity relationship diagram of the meteorological data system.	xv

List of Tables

3.1	Official formula-types for Portuguese price revision according to the nature of the works	19
4.1	Quality classes assigned to each aggregated period.	33
4.2	Meteorological indicators calculated for each daily period.	34
4.3	Operational temperature classes.	34
4.4	Precipitation classes used in the interpretative graphs.	35
4.5	Wind classes used in the interpretative graphs.	35
5.1	Batch-level results from the AI Image Risk Analyser	71
5.2	Image-level safety assessment results	72

Chapter 1

Introduction

The following sections frame the context of this dissertation, presenting the motivation for treating supervision records as data assets, then defining the fragmentation and reuse problems associated with current document centred workflows, and finally establishing the objectives and research questions that guide the proposed data analytics approach.

1.1 Context and Motivation

Infrastructure construction projects are often affected by deviations in schedule and cost, particularly in public works, where operational uncertainty, contractual requirements, and the need for timely approvals increase management difficulty (Antunes, 2012). In this context, supervision companies act as the owner’s representative on site and contribute directly to project governance by producing and validating the information used to support technical, financial, and contractual decisions.

Despite this central role, many supervision workflows remain strongly document-centred and manual. Information is frequently recorded in heterogeneous formats such as spreadsheets, handwritten notes, static reports, PDF documents, Word files, email messages, planning files, and unstructured photographic archives. As a result, supervision teams often spend significant effort locating, consolidating, and interpreting information that is dispersed across folders, systems, contracts, and stakeholders. This fragmentation reduces the ability to compare performance across projects, identify recurrent issues, detect risks early, and produce consistent evidence-based reports.

Digital transformation offers an opportunity to address these limitations by reorganising supervision information into more structured, traceable, and reusable data environments. Modern data analytics approaches emphasise structured data models, repeatable transformations, scalable processing pipelines, and reliable data preparation as foundations for analysis and decision support (Aggarwal, 2015; Han, Kamber, & Pei, 2011; Moreira, de Leon Ferreira, Carvalho, & Horváth, 2019). In parallel, visualisation and reporting mechanisms are essential for transforming complex datasets into information that can support monitoring, remote supervision, and management decisions (Ware, 2012).

The motivation of this work is therefore to explore how principles from contemporary data organisation can be adapted to construction supervision under realistic constraints. Rather than assuming the deployment of a full enterprise data platform, this dissertation investigates a lightweight, locally deployable approach capable of preserving raw supervi-

sion evidence, structuring heterogeneous records, and preparing curated outputs for analysis, dashboards, reporting, and future Artificial Intelligence assisted interpretation.

1.2 Problem Definition

The central problem addressed in this dissertation is the lack of a unified, traceable, and analytically prepared data environment for construction supervision. Although supervision activities produce extensive information during project execution, these records are often fragmented across different formats, repositories, and levels of structure. This limits their re-use and reduces their value for systematic monitoring, risk identification, contractual analysis, financial control, and monthly reporting.

This problem is particularly relevant in supervision contexts involving multiple contracts or geographically dispersed works. Without a consistent data organisation strategy, supervision remains dependent on manual consolidation and reactive document review. Delays or inaccuracies in supervision documentation, especially in measurement and approval processes, can contribute to weaker control over progress, payments, and contractual obligations (Antunes, 2012). The absence of integrated data structures also makes it difficult to connect inspection records with planning information, meteorological conditions, correspondence, photographic evidence, financial schedules, price revision data and many other aspects.

The dissertation does not treat digitalisation simply as the storage of documents in a central location, question that, in general, is already solved. The problem is instead approached as a data organisation and transformation challenge: how to move from dispersed operational records to structured, traceable, and decision-ready information. This requires methods for data acquisition, preservation of original evidence, extraction of relevant content, cleaning and standardisation, integration across sources, and preparation of outputs that can support both human interpretation and future computational analysis.

1.3 Objectives and Research Questions

The main objective of this dissertation is to define and demonstrate a structured data analytics methodology for managing construction supervision information from acquisition to interpretation. The work aims to adapt Lakehouse inspired and Medallion style principles to a realistic supervision context, using a lightweight local prototype capable of preserving raw evidence, preparing structured datasets, and supporting analytical outputs for reporting and decision support.

More specifically, the dissertation pursues the following objectives:

- To design a traceable data organisation model for heterogeneous construction supervision records, structured around Bronze, Silver, and Gold layers.
- To develop computational pipelines capable of extracting, cleaning, transforming, and integrating supervision data from multiple sources, including documents, correspondence, schedules, financial records, meteorological data, and photographic evidence.
- To demonstrate how curated supervision data can support analytical outputs, including dashboards, monthly reporting, financial analysis, risk identification, and AI assisted interpretation, while maintaining human oversight and professional accountability.

Based on these objectives, the dissertation is guided by the following research questions:

- How can heterogeneous construction supervision records be organised into a traceable and reusable data environment suitable for analytical processing?
- How can data extraction, transformation, and integration pipelines convert fragmented supervision records into structured datasets and decision-ready outputs?
- To what extent can dashboards, reporting tools, image analysis, and Large Language Model assisted interpretation improve the use of supervision data for monitoring, reporting, and decision support?

The contribution of this work lies in the development of a methodological and computational prototype for organising and analysing construction supervision data. Its purpose is not to reproduce the complete technological stack of enterprise Lakehouse platforms, nor to present Artificial Intelligence as a replacement for professional judgement. Instead, the dissertation demonstrates how contemporary data organisation and analytics principles can be adapted to support supervision activities in a practical, traceable, and scalable manner.

Chapter 2

Theoretical Foundations and Methods

This chapter presents the theoretical foundations and methodological principles that support the proposed data driven approach to construction supervision. The chapter has a dual purpose. First, it introduces the theoretical concepts required to understand the transformation of heterogeneous supervision information into structured, traceable, and decision ready data. Second, it explains how these concepts are methodologically adapted in this work through a lightweight, locally deployable prototype.

2.1 Data Analytics Process Models and CRISP-DM

Data analytics projects require computational tools, but also a structured process capable of connecting the problem definition, the available data, the analytical procedures and the final outputs.

The Cross-Industry Standard Process for Data Mining (CRISP-DM), is a well known model for data mining and applied analytics projects. The original methodology organises analytics into six phases (Chapman et al., 2000):

1. **Business understanding:** problem definition, objectives, constraints, success criteria and analysis expected value.
2. **Data understanding:** initial collection, inspection and exploration of the available data, including data sources, data quality issues and relevant variables.
3. **Data preparation:** selection, cleaning, transformation, integration and structuring of the data into a suitable form for analysis.
4. **Modelling:** application of analytical, statistical, machine learning or rule-based methods to extract patterns, produce classifications, calculate indicators or support interpretation.
5. **Evaluation:** assessment of whether the analytical results are coherent, reliable and useful in relation to the initial objectives.
6. **Deployment:** delivery of the results through reports, dashboards, databases, decision-support tools or other outputs that can be used by stakeholders.

These phases remain relevant because they describe a logical progression from a practical problem to data preparation, analysis, validation and results communication (Aggarwal, 2015; Chapman et al., 2000; Han et al., 2011). In this dissertation, CRISP-DM is used as a general analytical process reference, while the operational organisation of heterogeneous supervision data is provided by the Lakehouse inspired Medallion structure (Armbrust et al., 2021; Azure Databricks, 2026; Databricks, 2024; Reis & Housley, 2022).

2.2 Data Pipeline Architecture and Lakehouse/Medallion Concepts

Look for a centralised data architecture, to reduce fragmentation and improve analytical readiness, is relevant, since initial data is distributed, for example, across structured records, semi-structured documents, unstructured text, email correspondence, financial records, meteorological data, and images.

Lakehouse concept is here a relevant theoretical foundation, because combines the flexibility, scalability, and cost-efficiency of data lakes, with the governance, reliability, schema enforcement, and analytical performance of data warehouses. Armbrust et al. (2021) present the Lakehouse as a new generation of open data platforms designed to support both data warehousing and advanced analytics. Similarly, D. Lee et al. (2025) describe the Lakehouse as an architecture that combines the flexibility and scale of data lakes with data management, schema enforcement, and Atomicity, Consistency, Isolation and Durability (ACID)¹ transactions associated with data warehouses (Figure 2.1).

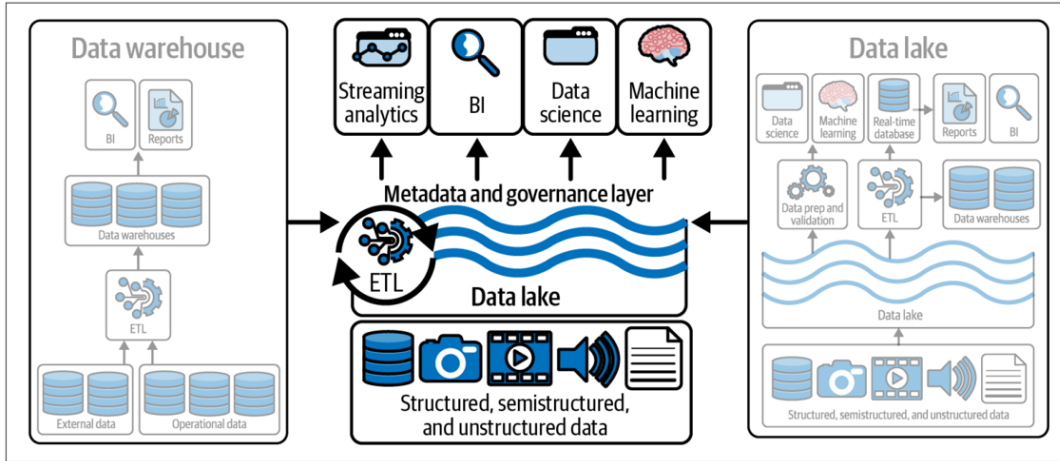


Figure 2.1: Lakehouse architecture as an integration of data warehouse and data lake capabilities. (D. Lee et al. (2025)).

The Medallion architecture organises data into progressive layers of refinement, referred as Bronze, Silver, and Gold. The Bronze layer stores raw records as close as possible to

¹ACID means atomicity, consistency, isolation, and durability. Atomicity ensures that a transaction is completed entirely or not applied at all; consistency ensures that the data remain valid after the transaction; isolation ensures that concurrent transactions do not interfere with one another; and durability ensures that committed changes remain stored even after system failures.

their original form. The Silver layer applies extraction, cleaning, standardisation, validation, and harmonisation. The Gold layer contains curated datasets prepared for dashboards, indicators, forecasting, Machine Learning (ML), Artificial Intelligence (AI) assisted reporting, and decision support. This layered logic is consistent with data preparation principles in data mining, where raw operational data must be transformed into reliable analytical representations before being used for modelling or decision support (Aggarwal, 2015; Han et al., 2011). It is also aligned with current Lakehouse practice, where Medallion architecture is used as a framework for incremental data quality (Databricks, 2025; D. Lee et al., 2025).

2.3 Bronze Layer Source Synchronisation

In a lakehouse or medallion based data architecture, data is organised into progressive layers according to its level of processing, quality and analytical readiness. The Bronze layer is generally associated with the preservation of raw or near raw data Azure Databricks (2026); Databricks (2024). The same principle is adapted here, where project documentation must first be stabilised in a controlled raw repository before performing any automated extraction.

The Bronze layer acts as an ingestion boundary between operational file storage and analytical processing. Operational repositories are working environments, where files may be modified, synchronised, renamed or reorganised. Processing documents directly from these locations could reduce reproducibility, since the source material may change during or between computational runs. Differently, a controlled Bronze repository provides a stable location from which extraction routines operate.

The synchronisation routine does not import all available projects from the repository and only updates projects that already exist in the Bronze repository. This preserves the analytical scope defined for the study and avoids uncontrolled expansion of the dataset. Project folders are associated using an initial folder-name prefix, treated as a stable project key. This is useful because full project folder names may vary between repositories, while the codified initial project identifier remain constant.

From a computational perspective, the method combines recursive file system traversal², metadata comparison and content verification. Each source file is compared with its corresponding target path in the Bronze repository. If the file does not yet exist in the Bronze layer, it is classified as a new file. If the file already exists, modification dates are compared, and the source file is only considered for replacement when it is more recent than the Bronze version. Where appropriate, file size and a SHA-256 hash are also used to avoid unnecessary overwriting when two files have identical content despite timestamp differences Python Software Foundation (2026a).

In a dry run mode, before applying any change, the script generates a report describing the files that would be copied, updated, skipped or ignored. This enables inspection of the expected operation before modifying the Bronze repository. In this way, the synchronisation process becomes auditable and reduces the risk of accidental overwriting, incorrect project matching or uncontrolled file ingestion.

Once Bronze layer have raw data, later analytical outputs may contain project names, personal names, organisation names, or financial information. Data governance can be

²Recursive file-system traversal implemented with `Path.rglob()` method from Python's `pathlib` module Python Software Foundation (2025)

strengthened through pseudonymisation, replacing direct identifiers with stable pseudo-identifiers. Although this does not render the data fully anonymous, it reduces unnecessary exposure of identifiable information while preserving analytical continuity across meetings, reports, and projects European Data Protection Board (2025); European Union (2016).

2.4 Silver Layer Data Extraction and Structuring

Here, raw data are transformed into cleaned, structured and more reliable datasets. It is where raw data is prepared for analytical use by applying extraction, cleaning, standardisation and validation procedures Azure Databricks (2026); Databricks (2024). This intermediate layer is consistent with the general role of data preprocessing in analytics Aggarwal (2015); Han et al. (2011).

Because many relevant records are originally stored in semi-structured or unstructured formats, this stage is particularly important. Daily reports, meeting minutes, inspection records, Word documents, Portable Document Format (PDF) files, spreadsheets, e-mail messages and photographic metadata are not immediately suitable for analysis. They often contain inconsistent naming conventions, repeated information, missing fields, different date formats, manually written descriptions and project-specific terminology. The Silver layer therefore acts as the transformation boundary between documentary evidence and analytical datasets.

It depends on the nature of data source, the methods applied. Tabular sources, such as spreadsheets or financial schedules, may require column standardisation, data type conversion, missing value treatment and validation of numerical fields. Textual sources, such as reports, correspondence or meeting records, may require parsing, keyword extraction, entity identification, date normalisation and classification of relevant topics. Based sources, such as meteorological data, require transformation from external response formats into project-specific tables and summaries. These operations follow the general logic of Extract, Transform, Load (ETL) processes³, in which data are acquired from source systems, prepared through transformation rules and made available for analytical use Kimball and Ross (2013); Moreira et al. (2019).

Traceability is achieved, When possible, extracted records should preserve a connection to the original file, source path, date, project, document type or processing run. This is important because analytical conclusions may need to be justified with reference to the underlying evidence. For this reason, Silver outputs should not be understood as independent replacements for the original documents, but as structured representations derived from them. Maintaining this relationship between source evidence and processed records improves reproducibility and supports later verification of the analytical process Kimball and Ross (2013); Moreira et al. (2019).

In this dissertation, have been extracted and organised information from different supervision domains. These include correspondence ingestion into a SQLite database, meteorolo-

³But proposed workflow combines elements of both ETL and also Extract, Load, Transform (ELT) approaches. At the architectural level, it follows an ELT inspired logic, since source evidence is first loaded and preserved in the Bronze layer before being transformed into Silver and Gold outputs. At the implementation level, several individual routines follow an ETL pattern, as Python scripts extract information from source files or Application Programming Interface (API)s, transform it through cleaning and structuring procedures, and then load the results into databases, tables, dashboards or report-ready outputs.

gical data transformation from the IPMA API, extraction of activities, labour and equipment from daily supervision records, processing of financial and price revision information, and preparation of structured inputs for image analysis and reporting workflows. Although the degree of automation varies between domains, the common methodological role of the Silver layer is to increase data consistency, comparability and analytical readiness.

2.5 Gold Layer Analytical Outputs and Reporting

The Gold layer corresponds to the stage where structured data are transformed into curated outputs prepared for analysis, visualisation, reporting and decision support. In a Medalion style architecture, Gold datasets are generally business ready products derived from the cleaned and structured data available in the Silver layer Azure Databricks (2026); Databricks (2024). In this dissertation, the same principle is adapted to construction supervision by producing outputs that support monitoring, contractual interpretation, financial analysis, risk identification and monthly reporting.

Unlike the Bronze layer, which prioritises preservation, and the Silver layer, which prioritises structuring, the Gold layer prioritises interpretation and use. Its outputs are designed to answer practical supervision questions, such as how work is progressing, which activities are being reported, how actual execution compares with planned values, whether weather conditions may have affected production, which correspondence topics are recurrent, or which visual risks are identifiable in photographic records. These outputs may take the form of dashboards, plots, summary tables, analytical indicators, report sections, HyperText Markup Language (HTML) pages or exportable documents.

From a data analytics perspective, this stage is associated with the transformation of prepared datasets into information products that can support monitoring and decision-making. Visualisation plays an important role in this process, since graphical representations can help users identify patterns, compare values, detect deviations and interpret complex datasets more effectively (??). Similarly, reporting outputs provide a structured way to communicate selected indicators, evidence and interpretations to project stakeholders.

In the construction supervision context, Gold outputs must remain closely connected to their source data. Analytical summaries are only useful if they can be traced back to the records from which they were derived. This is particularly relevant when outputs are used to support monthly reports, financial claims, contractual analysis or health and safety observations. Therefore, the Gold layer should not be understood as a final automated decision making layer, but as a curated analytical layer that supports professional judgement.

The Gold layer developed in this work includes several types of outputs. E-mail data are transformed into analytical dashboards and topic summaries. Meteorological records are converted into project-related weather reports based on the closest available stations. Daily inspection records are summarised through activity, labour and equipment indicators. Financial information is used to compare planned and executed values and to support price revision analysis. Photographic evidence is processed through computer vision and Large Language Model (LLM) assisted interpretation to generate risk-oriented comments. Finally, selected outputs are assembled into report-ready material for monthly construction supervision reporting.

Here, human validation becomes especially important. Although dashboards, automated summaries and LLM assisted text generation can improve consistency and reduce reporting

effort, they do not replace technical responsibility. Large Language Models may support summarisation, classification and synthesis, but their outputs require verification, especially when used in professional or high-consequence contexts Bommasani et al. (2021); National Institute of Standards and Technology (2023b). The Gold layer should therefore be treated as a decision-support and reporting-support layer, where computational outputs are reviewed, corrected and validated by professionals before being used in formal supervision documents.

2.6 Outlook and Email Correspondence Analysis

Email correspondence is a major source of supervision evidence because it records instructions, approvals, requests, clarifications, warnings, contractual exchanges and coordination decisions between stakeholders. From a data perspective, e-mail messages combine structured or semi-structured metadata, such as sender, recipients, date, subject, folder path, attachments and conversation information, with unstructured body text. The Outlook Object Model also represents e-mail messages, recipients and attachments through specific objects and collections, which supports their extraction into structured records (Microsoft, 2024a, 2024b, 2024c). For this reason, Outlook correspondence analysis requires a combination of metadata analysis, text mining, information retrieval and exploratory data analysis (Aggarwal & Zhai, 2012; Kobayashi, Mol, Berkers, Kismihók, & Den Hartog, 2018; Manning, Raghavan, & Schütze, 2008).

From a data analytics perspective, e-mail analysis can be divided into two complementary dimensions: metadata analysis and text-content analysis. Metadata analysis uses fields such as sender, recipient, timestamp, folder, attachment presence and message classification to describe communication patterns and document flow. These variables make it possible to analyse communication intensity, temporal concentration, dominant actors, folder-level distribution and the existence of supporting attachments. This type of descriptive analysis is consistent with exploratory data analysis, where summary statistics and visualisations are used to understand the structure, quality and behaviour of a dataset before final interpretation (Tukey, 1977).

The second dimension concerns the textual content of e-mail bodies and subjects. Since e-mail body text is usually unstructured, it must be converted into a form that supports search, filtering and interpretation. Information retrieval and text mining workflows commonly include text normalisation, tokenisation, treatment of frequent or irrelevant terms, indexing and term-based retrieval (Aggarwal & Zhai, 2012; Manning et al., 2008). In this dissertation, these principles support the extraction of Outlook messages into searchable body-text files and structured database fields, enabling later filtering by subject terms, body keywords and predefined thematic expressions.

Keyword based search is a transparent rule based method for retrieving textual records. It identifies documents or messages that contain specific words or expressions considered relevant to the analytical objective. In information retrieval, this corresponds to a term based retrieval logic, where the presence, frequency or matching of terms is used to support document selection and ranking (Manning et al., 2008). Although keyword search does not capture all semantic relations, synonyms or implicit meanings, it has the advantage of being auditable and easy to validate. In construction supervision, this transparency is important because retrieved correspondence may need to be inspected and justified as documentary evidence.

The use of externally defined keyword lists also supports reproducibility and methodolo-

gical control. By storing analytical themes outside the code, for example in a spreadsheet such as `keys.xlsx`, the vocabulary used for retrieval can be reviewed, updated and reused without changing the processing script. This is consistent with the principle that information retrieval systems depend not only on the search algorithm, but also on the definition of the document representation, indexing strategy and query terms (Manning et al., 2008). Keyword themes are therefore treated as configurable analytical rules rather than hidden model parameters.

Subject line analysis provides an additional way to explore e-mail correspondence. The subject field often contains a short description of the topic, request, location, contract or issue under discussion. Word frequency analysis over subject lines can help identify recurrent terms and dominant topics, but the results may depend on preprocessing decisions such as lower-casing, accent normalisation, stop-word removal or grouping of repeated terms. Text mining literature recognises that preprocessing decisions influence the representation of textual data and therefore affect later retrieval, ranking or classification results (Aggarwal & Zhai, 2012; Manning et al., 2008). So, subject sensitivity analysis can be used to compare whether term rankings and topic interpretations remain stable under alternative preprocessing assumptions.

Full-text search supports correspondence analysis by allowing larger collections of body-text records to be indexed and searched efficiently. SQLite provides full-text search extensions, such as FTS5, which are designed to index text and retrieve documents containing specified search terms more efficiently than simple sequential scanning (SQLite Consortium, 2025a). In the developed Outlook workflow, saved plain-text body files and full-text indexing support the movement from aggregate dashboard indicators to inspectable source messages. This connection is important because supervision conclusions may need to be traced back to specific correspondence records.

Exploratory Data Analysis is also relevant to the Outlook workflow. Before correspondence data can be used for reporting or interpretation, the extracted dataset must be checked for completeness, coherence and possible errors. Summary tables, temporal plots, sender and recipient rankings, attachment statistics and folder-level distributions help identify missing values, unusual activity patterns, dominant stakeholders and possible classification problems. This follows the general logic of exploratory data analysis, where visual and tabular summaries are used to understand data structure, detect anomalies and assess analytical readiness before producing final outputs (Tukey, 1977).

Communication network analysis adds a relational perspective to e-mail correspondence. In this approach, actors are represented as nodes and communication exchanges are represented as edges connecting senders and recipients. When repeated exchanges are aggregated, the number of messages can be used as an edge weight, allowing stronger communication links to be distinguished from occasional interactions. This representation follows the general logic of social network analysis and graph theory, where relations between actors are modelled as networks composed of nodes and edges (Newman, 2010; Wasserman & Faust, 1994).

Network indicators can support the interpretation of communication structure. Measures such as degree, weighted degree and centrality indicators can help identify actors with many connections, actors involved in frequent exchanges, or actors occupying structurally important positions in the communication network (Newman, 2010; Wasserman & Faust, 1994). In the implemented workflow, these calculations are supported by Python-based processing of sender/recipient relations and, where available, by `networkx`, a Python package for representing and analysing network structures (Hagberg, Schult, & Swart, 2008). These outputs should be interpreted as exploratory indicators of communication flow, not as automatic

assessments of responsibility, performance or contractual relevance.

These theoretical principles are implemented through a staged workflow. Outlook messages are first extracted and preserved in a structured local SQLite database, together with message metadata, recipients, attachments, folder paths and body-text references. The extracted data are then explored and validated through descriptive statistics, temporal analysis, sender and recipient summaries, attachment analysis and subject-word frequencies. Finally, curated outputs are produced through an interactive dashboard, keyword-based retrieval, subject sensitivity analysis, exportable evidence tables and communication network outputs. This organisation connects e-mail correspondence analysis with the broader Medallion style logic of the dissertation, where raw communication evidence is preserved, structured, explored and transformed into report-ready analytical outputs (Azure Databricks, 2026; Databricks, 2024).

Chapter 3

Literature Review

It is here presented the main research streams that support the proposed methodology, organised around themes, like digital transformation in construction supervision, modern data architectures, Python-based data processing, SQL based structuring, document and correspondence analytics, meteorological data analysis, financial and contractual analytics, computer vision, multimodal safety assessment, dashboard-based reporting, and Large Language Model-assisted monthly reporting.

3.1 Digital Transformation and Data-Driven Supervision

Construction supervision depends on the continuous production, validation, interpretation, and communication of technical, contractual, financial, environmental, and safety-related evidence (Antunes, 2012). The literature on digital transformation in construction highlights the increasing importance of data-driven workflows, automation, remote monitoring, and decision-support mechanisms (Oesterreich & Teuteberg, 2016; Olanipekun & Sutrisna, 2021; Radman, Jelodar, Lovreglio, Ghazizadeh, & Wilkinson, 2022). These developments are relevant because supervision information is often dispersed across several types of documents (Da Silva, 2025; de Vilhena Costa, Coelho, & Correia, 2024).

Building Information Modelling (BIM) occupies a central position in the digital transformation of the construction sector because it promotes structured information management, collaborative work processes, and the digital representation of built assets throughout their life cycle (EU BIM Task Group, 2017; International Organization for Standardization, 2019). In the European context, BIM has been promoted as a strategic instrument for modernising public procurement and improving the planning, design, construction, and operation of public assets (EU BIM Task Group, 2017). In Portugal, this transition has also gained legal and policy relevance. Portaria n.º 255/2023 introduced references to digital construction data modelling and BIM in the instructions for the preparation of public works projects, while Decreto-Lei n.º 10/2024 established the mandatory presentation of architectural projects according to the BIM methodology from 1 January 2030 (República Portuguesa, 2023, 2024). More recently, Resolução do Conselho de Ministros n.º 89/2026 approved the national strategy PortugalBIM, aiming to create the conditions for BIM adoption in the architecture, engineering, construction and operation sector through policies, standardisation, technologies, and capacity building (Presidência do Conselho de Ministros, 2026). These developments show that BIM is becoming an important regulatory and organisational reference for

construction digitalisation in Portugal. However, BIM does not eliminate the need for complementary data analytics, machine learning, and AI assisted workflows, but reinforce the need for it. Simultaneously, supervision still produces heterogeneous operational evidence that extends beyond model based project information.

Literature supports the view that construction supervision can benefit from methods that transform fragmented operational records into structured, traceable, and decision-ready information. This requires structured data preparation, integration, analysis, and visualisation methods capable of supporting dashboards, reporting, risk analysis, and future AI assisted monthly reporting (Aggarwal, 2015; Han et al., 2011; ?).

3.2 CRISP-DM and Complementary Data Analytics Governance References

CRISP-DM remains an important reference in applied data analytics because it provides an independent structure for connecting a practical problem to data understanding, preparation, analysis, evaluation and delivery of outputs (Aggarwal, 2015; Chapman et al., 2000; Han et al., 2011; Moreira et al., 2019). However, contemporary data analytics projects increasingly require multidisciplinary collaboration, iterative development, operational deployment and lifecycle management of data products, aspects that extend beyond the original scope of CRISP-DM as a data mining process model (Saltz, 2021)

The literature suggests that CRISP-DM can be complemented with data engineering, data architecture and data governance perspectives to better address the requirements of contemporary data analytics projects. Data engineering literature emphasises reliable data pipelines, maintainable architectures, metadata management, data quality controls and the separation of raw, processed and curated data (Kleppmann, 2017; Reis & Housley, 2022). Lakehouse and Medallion approaches provide an architectural model for organising data into progressive refinement layers, commonly described as Bronze, Silver and Gold (Armbrust et al., 2021; Azure Databricks, 2026; Databricks, 2024). This is relevant when the objective is not only to produce one analytical result, but to create a reusable and traceable data environment.

The literature on machine learning and AI lifecycle management also reinforces the need to complement CRISP-DM when analytical systems include model-based or AI-assisted components. Extensions such as CRISP-ML(Q) place stronger emphasis on quality assurance, monitoring and maintenance in machine learning workflows (Studer et al., 2021). Other proposals, such as context-aware extensions of CRISP-DM, also reflect the need to adapt the original process model to more complex data science settings involving model reuse, operationalisation and changing application contexts (Martínez-Plumed et al., 2017). These contributions do not make CRISP-DM obsolete; they show that it is better understood as a methodological baseline that can be combined with architecture, governance and lifecycle references.

Several other recent data quality and AI governance standards reflect concerns that were not central when CRISP-DM was created. ISO/IEC 5259 addresses data quality for analytics and machine learning, ISO/IEC 5338 addresses AI system lifecycle processes, ISO/IEC 23894 focuses on AI risk management, and ISO/IEC 42001 addresses organisational AI management systems (International Organization for Standardization, 2023a, 2023b, 2023c, 2024a, 2024b). The NIST AI Risk Management Framework and the European AI Act further

reinforce the relevance of risk awareness, transparency, human oversight and accountability in AI-supported systems (European Parliament and Council of the European Union, 2024; National Institute of Standards and Technology, 2023a).

These standards and frameworks are used here as literature references rather than as formal compliance requirements. They help justify why a contemporary data analytics methodology should combine a traditional process model with data architecture and governance principles. Therefore, the literature supports the methodological position adopted in this work: CRISP-DM provides the general analytical process logic, while Lakehouse-inspired Medallion organisation and governance references support traceability, data quality, auditability, lifecycle awareness and human validation.

3.3 Modern Data Architectures and Local Pipeline Adaptation

Recent literature on data Lakehouse architectures discusses the need to combine scalable data storage, structured querying, metadata management, governance, and advanced analytics. The Lakehouse concept has been proposed as a way to combine the flexibility of data lakes with the reliability and analytical capabilities of data warehouses (Armbrust et al., 2021; Harby & Zulkernine, 2025). Related work on data lakes also emphasises metadata management, governance, and the risk of uncontrolled data repositories or “data swamps” (Azzabi et al., 2024; Sawadogo & Darmont, 2021).

Delta Lake and related open table formats provide practical mechanisms for implementing Lakehouse architectures, including transactional reliability, schema enforcement, audit history, and batch or streaming workloads (Armbrust et al., 2020; D. Lee et al., 2025). The Medallion architecture, expressed through Bronze, Silver, and Gold layers, is particularly relevant because it provides a model for progressive data refinement.

Such principles support the design of a lightweight, locally deployable data pipeline. The implementation does not reproduce a full production-grade Lakehouse platform. Instead, it adapts Lakehouse and Medallion principles to a local folder-based structure, supported by an external disk, Python scripts, structured datasets, and reporting outputs. The literature is therefore used as a reference for best practices in data organisation, traceability, and analytical readiness, rather than as a claim of full technological equivalence.

3.4 Medallion Architecture, File Synchronisation and Data Governance

The medallion architecture is a layered data-management pattern commonly associated with lakehouse systems. It organises data into successive layers, usually designated as Bronze, Silver and Gold, where each layer represents an increasing degree of processing, quality and analytical value Azure Databricks (2026); Databricks (2024). The Bronze layer typically stores raw source data, the Silver layer contains cleaned and standardised datasets, and the Gold layer supports reporting, business intelligence and decision-making outputs.

Although the medallion architecture is often presented in the context of cloud data platforms, databases and large-scale analytical systems, its logic is also applicable to file-based

workflows. Construction supervision is a documentation-intensive activity, producing meeting minutes, technical records, correspondence, drawings, reports and financial documents. These files are often stored in operational repositories that are continuously updated by different actors. In this context, establishing a Bronze layer provides a controlled repository of raw evidence before any structured extraction, cleaning or analysis takes place.

The Bronze layer is particularly relevant because it separates operational document management from analytical processing. Operational folders may be affected by user edits, cloud synchronisation mechanisms, file locks, incomplete downloads or temporary system files. Cloud-synchronised environments may also expose placeholder or system-managed items that are not ordinary locally available files Microsoft Support (2026). Therefore, robust file-ingestion processes should be able to skip inaccessible or system-like files and record these cases without interrupting the whole workflow.

The computational implementation of such a process can rely on standard file-system operations. Python's `pathlib` library provides object-oriented filesystem paths with semantics appropriate for different operating systems, supporting clearer and more robust path handling than plain string manipulation Python Software Foundation (2026b). For file transfer, `shutil` provides high-level file operations, including copy functions such as `copy2`, which attempt to preserve file metadata where supported by the operating system Python Software Foundation (2026c). For content comparison, `hashlib` provides secure hash and message digest algorithms, including SHA-256, which can be used to produce reproducible file-content fingerprints Python Software Foundation (2026a).

The use of modification dates alone is not always sufficient for reliable file synchronisation. Files may have different timestamps despite identical content, especially when copied between locations or synchronised through cloud services. Conversely, a newer timestamp may indicate an updated document that should replace an older Bronze version. Combining timestamp comparison with file size and content hashing provides a practical compromise between computational efficiency and reliability. Hashing is not used here for cryptographic security certification, but as a deterministic method for detecting whether two files are content-equivalent.

Data governance is another important dimension of the proposed workflow. Construction supervision documentation may include personal data, project identifiers, company names, financial values and contractual information. Under the General Data Protection Regulation, pseudo-anonymisation refers to processing personal data in such a way that it can no longer be attributed to a specific data subject without the use of additional information, provided that this additional information is kept separately and protected by technical and organisational measures European Union (2016). The European Data Protection Board further clarifies that pseudo-anonymisation can reduce risks to data subjects, but does not by itself transform personal data into anonymous data European Data Protection Board (2025).

Privacy-risk management frameworks also emphasise the need to identify and manage privacy risks throughout data processing activities National Institute of Standards and Technology (2025). In the proposed workflow, this supports a distinction between the Bronze layer, where original documents are preserved as raw evidence, and later analytical outputs, where direct identifiers can be replaced by stable pseudo-identifiers. This allows the analytical process to preserve continuity across records while reducing direct exposure of identifiable information in intermediate datasets, dashboards and reports.

3.5 Document, Correspondence and Inspection Record Analytics

A significant part of supervision evidence exists in semi-structured or unstructured form. Daily inspection records may be produced as PDF or Word documents and may include embedded photographs or references to image folders. Outlook correspondence records instructions, approvals, warnings, clarifications, contractual exchanges, claims-related communication, and other supervision evidence. These records are rich in meaning but difficult to analyse systematically when they remain dispersed across documents, mailboxes, folders, and reporting periods.

The literature on information retrieval and text mining supports the extraction of relevant information from large collections of text and documents (Aggarwal & Zhai, 2012; Kobayashi et al., 2018; Manning et al., 2008). In this work, these foundations support the analysis of Outlook correspondence through metadata, message threads, subjects, body text, attachments, and recurring topics. Microsoft Graph or mailbox exports may be used as access routes depending on permissions and technical constraints (Microsoft, 2025, 2026).

Inspection record consolidation is grounded in data integration and data quality. Data integration concerns the combination of records from different sources into a coherent structure, while data quality concerns dimensions such as accuracy, completeness, consistency, timeliness, and fitness for use (Lenzerini, 2002; R. Y. Wang & Strong, 1996). These principles are important because inspection records may support contractual decisions, payment validation, risk identification, non-conformity tracking, and dispute resolution.

3.6 Document Extraction from PDF, Word and Excel sources

Classical construction supervision documentation¹, is frequently stored in hierarchical folder repositories and may include editable Excel spreadsheets, Word documents, digital PDFs and, less frequently, scanned PDFs. The extraction method should therefore be selected according to the document format, the level of embedded structure and the need for traceability. For editable spreadsheets, deterministic extraction is generally appropriate because the information is already organised in worksheets, rows, columns and cells. Python libraries such as pandas and openpyxl support the direct reading of Excel files and workbook structures into tabular datasets (openpyxl contributors, 2026; The pandas development team, 2026). For editable Word documents, python-docx provides programmatic access to .docx document content, supporting extraction or generation of paragraphs, tables and formatted text (python-docx contributors, 2026).

Digital PDFs require text and table extraction methods when the content is machine readable. Apache Tika supports text and metadata extraction from many file types, including

¹The term “classical” is used here to distinguish conventional document based construction supervision workflows from Building Information Modelling (BIM) centred information management environments. BIM and related information management standards, such as ISO 19650, aim to structure project information digitally across the asset life cycle (International Organization for Standardization, 2019). However, despite the transformative potential of BIM for construction digitalisation and information management, many operational construction processes still rely on document repositories, spreadsheets, Word files, PDFs and folder-based exchanges, especially at site and supervision level (Alsofiani, 2024; Koeleman, Ribeirinho, Rockhill, Sjödin, & Strube, 2019).

PDF and Office formats, while Camelot is designed for table extraction from PDFs (Apache Software Foundation, 2026; Camelot contributors, 2026). Scanned PDFs and image-only records require optical character recognition, for which Tesseract is a widely used open-source OCR engine (Tesseract OCR contributors, 2026). However, when scanned documents are rare, Optical Character Recognition (OCR) is better understood as a fallback mechanism rather than the central extraction strategy.

Recent research has also explored free or light OCR document understanding using multimodal models and LLM's. Donut (G. Kim et al., 2022), mPLUG-DocOwl (Ye et al., 2023) and UReader (Ye et al., 2023) are examples of approaches that combine visual layout understanding and semantic interpretation for document understanding tasks. These methods are relevant for complex forms or visually rich documents, but their use as a first extraction layer may introduce API costs, data-exposure concerns and risks of semantic over-interpretation.

In lakehouse oriented workflows, an ELT approach first loads raw or near-raw data into a controlled repository before transformation (Databricks, 2026b). Therefore, for auditable construction supervision data, deterministic extraction from editable or digital sources remains closer to an ELT logic than early LLM-mediated extraction, while LLMs are more safely positioned downstream for classification, synthesis or report drafting.

3.7 Contract Price Revision and Financial Schedule Analysis

Contract price revision is part of a broader international concern with price adjustment mechanisms in long-duration construction contracts. In construction projects, the time gap between tender submission and actual execution exposes both clients and contractors to variations in labour costs, material prices, fuel prices, energy costs and general inflation. International guidance on procurement and contract administration recognises that price adjustment clauses can reduce the risk created by market volatility by using reference dates, cost indices, coefficients and contractual formulae to update payment values over time (Asian Development Bank, 2023; Statistics Canada, 2022). This perspective is relevant because it frames price revision as a risk-sharing and contract-management mechanism, rather than as an arbitrary increase in the contract value.

The Portuguese framework gives this general principle a specific legal and technical form. Price revision in Portugal is regulated by Decree-Law No. 6/2004, which establishes the regime applicable to public works contracts, private works contracts and contracts for the acquisition of goods and services (Diário da República, 2004a). Within this regime, revision is calculated through formula-based mechanisms that relate the value of the works executed in a given period to the variation of official labour, material and equipment indices between the base month and the execution month. The procedure therefore depends on the contractual formula or formula-type, the reference month, the auto month and the certified value of the works.

The formula-type system is central to the Portuguese procedure. Formulae F01 to F14 were established by Despacho No. 1592/2004, considering Retificação No. 383/2004, while formulae F15 to F23 were established by Despacho No. 22637/2004 (Diário da República, 2004b, 2004c, 2004d). These formulae define the labour, material, equipment and fixed components associated with different types of works. The formula-type is therefore not merely

descriptive; it determines the weighted structure used to calculate the revision coefficient. The monthly indices required to apply these formulae are published by Instituto dos Mercados Públicos, do Imobiliário e da Construção (IMPIC), including indices for labour, materials and support equipment (IMPIC, 2026a, 2026b).

Table 3.1 summarises the official formula-types considered in this work. The table is included in the literature review because it identifies the regulatory classification that supports the methodological implementation developed later in Chapter 4.

Table 3.1: Official formula-types for Portuguese price revision according to the nature of the works

Formula type	Type of works
F01	Residential buildings
F02	Administrative buildings
F03	School buildings
F04	Healthcare buildings
F05	Light building rehabilitation
F06	Medium building rehabilitation
F07	Deep building rehabilitation
F08	Sports fields with changing rooms
F09	External works and landscaping
F10	Roads
F11	Tunnels
F12	Reinforced or prestressed concrete bridges
F13	Reinforced or prestressed concrete viaducts
F14	Reinforced or prestressed concrete grade-separated crossings
F15	Major road repairs
F16	Road maintenance
F17	Road paving
F18	Reinforced concrete structures
F19	Steel structures
F20	Electrical installations
F21	Water supply and wastewater networks
F22	Earth dams
F23	Irrigation and drainage networks

This framework is directly relevant to the present dissertation because it supports the treatment of price revision as a reproducible data process. The literature and regulatory sources indicate that price revision requires a clear link between contractual information, reference periods, official indices and certified work values. In the proposed workflow, these elements are organised as structured datasets, allowing the Portuguese price revision calculation to be reproduced, validated and incorporated into financial reporting. The objective is not to replace the legal procedure, but to operationalise it through controlled data preparation, formula selection, index validation and auditable outputs.

Financial schedule analysis is related to price revision because payment certificates are both the basis for revision calculations and a record of the financial evolution of the contract. International project-control literature, particularly Earned Value Management, relates

planned value, earned value and actual cost, allowing project performance to be interpreted through indicators such as schedule variance, cost variance, schedule performance index and cost performance index (Fleming & Koppelman, 2016; Proaño-Narváez et al., 2022). These concepts are useful because they show that financial execution can contribute to project monitoring when connected to planned progress and measured work.

Nevertheless, financial deviation must be interpreted with caution. A difference between the tender financial schedule and the actual certified values does not necessarily prove physical delay. It may result from an unrealistic initial financial schedule, different sequencing of work fronts, certification timing, measurement procedures or changes in the relative weight of activities. A rigorous assessment of delay would require integration with the physical work plan, measured quantities, labour and equipment allocation, activity dependencies and critical path analysis. Therefore, in this dissertation, financial schedule evolution and price revision outputs are treated as structured sources of evidence for project supervision, rather than as a complete delay-analysis or Earned Value Management system. ““

3.8 Computer Vision and Image-Based Risk Analysis

Photographic records are central to construction supervision because they document visible site conditions, work progress, equipment, materials, personal protective equipment, collective protection measures, and potential risks. However, photographic archives are difficult to analyse manually, because of their number and scale. Computer vision provides a technical foundation for extracting information from images through classification, object detection, and visual recognition.

Deep Learning (DL) and object detection are particularly relevant to this task. Foundational work on DL and You Only Look Once (YOLO) based object detection supports the use of neural models for visual recognition and real-time detection (LeCun, Bengio, & Hinton, 2015; Redmon, Divvala, Girshick, & Farhadi, 2016). Recent studies apply YOLOv8 and related object-detection models to construction safety, PPE detection, and worker-safety monitoring, showing their relevance for supporting automated recognition of PPE compliance and construction-site safety conditions (Barlybayev et al., 2024; D. Kim & Xiong, 2025; X. Liu et al., 2024). Official Ultralytics documentation also positions YOLOv8 as a real-time object-detection model family with a range of pretrained models and an accuracy-speed trade-off suitable for applied detection tasks (Ultralytics, 2026).

Recent developments in multimodal large language models also make it possible to analyse construction images through API based vision services. Unlike object detection models, which mainly return predefined labels, bounding boxes, and confidence scores, multimodal LLMs can generate contextual scene interpretations, observations, risk descriptions, uncertainty notes, and narrative summaries from visual input (Anthropic, 2026c, 2026d). This is relevant for construction supervision because photographic evidence often requires semantic interpretation: the same detected objects may represent different risk conditions depending on their spatial arrangement, activity context, proximity to hazards, temporary access conditions, or visible collective protection measures. LLM based image analysis can therefore complement structured object detection by producing descriptive and risk-oriented interpretations of site photographs.

In this work, YOLOv8 is used in the Site Image Risk Analyser as the structured computer vision branch. It detects safety-relevant objects such as persons, helmets, ladders, and other

visible elements, producing labels, bounding boxes, and confidence scores. These outputs support rule-based risk interpretation, including PPE related checks and hazard related flags. Alongside this branch, a multimodal LLM branch uses the Claude Vision API to interpret the same construction images and produce contextual risk assessments, plain-language observations, uncertainty notes, and narrative summaries (Anthropic, 2026d). The two branches are deliberately kept separate so that the structured detection output and the semantic interpretation output can be evaluated independently before being combined.

The Site Image Risk Analyser therefore follows a dual-pipeline architecture. Images are processed through a file-scanning and batch-processing workflow, after which the YOLOv8 computer vision branch and the Claude-based LLM branch run in parallel. The Computer Vision (CV) branch produces typed detection outputs, while the LLM branch produces typed interpretative outputs. A risk engine then applies deterministic rules to the CV detections, including Personal Protective Equipment (PPE) related and hazard-related checks, and a fusion engine combines the CV and LLM results. This fusion stage identifies shared findings, CV only findings, LLM only findings, and explicit conflicts between the two pipelines. It also assigns a combined risk level and indicates whether manual review is recommended.

This hybrid approach is relevant to the research objective because it does not rely exclusively on either object detection or generative interpretation. YOLOv8 provides structured, auditable evidence through detected classes, locations, and confidence scores. The LLM branch adds broader scene understanding and explanatory text, which may capture contextual risks that are not directly represented by a predefined object class. The fusion layer makes agreements and disagreements explicit, supporting traceability and human review. In this sense, the proposed workflow uses computer vision for measurable visual detection, LLM vision capabilities for semantic interpretation, and deterministic fusion for transparent construction-safety reporting.

3.9 Multimodal LLM Based Safety Interpretation and Fusion

Recent developments in Large Language Models and multimodal AI have expanded the possibility of interpreting images not only through object detection, but also through contextual scene reasoning. LLM based vision systems can describe scene conditions, identify uncertainty, and produce narrative assessments. However, such outputs may be less structured than computer vision detections and may require grounding, validation, and human oversight.

Retrieval-Augmented Generation (RAG) and LLM agent literature are relevant because they address the use of external evidence, structured retrieval, tool use, and multi-step reasoning in AI systems (Fan et al., 2024; Gao et al., 2024; Lewis et al., 2020; Plaat et al., 2025; L. Wang et al., 2024). Anthropic documentation supports the development of applications using Claude models through API based workflows (Anthropic, 2026a, 2026b). In this work, Claude is used as the LLM branch of the Site Image Risk Analyser, producing risk factors, observations, narrative summaries, and uncertainty notes from construction site images.

The dual-pipeline design responds to a methodological problem identified in AI assisted safety assessment: structured detections and semantic reasoning provide different types of evidence. The CV branch provides precise, auditable object detections, while the LLM branch

provides contextual interpretation. The fusion layer compares both outputs, identifies shared findings, CV only findings, LLM only findings, and explicit conflicts, and produces a combined risk level. This design supports human review by making agreement and disagreement visible instead of hiding uncertainty inside a single model output.

3.10 Analytical Outputs, Decision Support and AI-Assisted Reporting

In a data analytics workflow, the value of the system does not depend only on data acquisition and processing, but also on the capacity to transform processed data into structured, interpretable, traceable and reusable outputs. Data quality literature emphasises that analytical outputs should be assessed through dimensions such as accuracy, completeness, consistency, timeliness, validity and usability, since poor data quality can compromise downstream interpretation and decision-making (Batini, Cappiello, Francalanci, & Maurino, 2009; ISO/IEC, 2008; R. Y. Wang & Strong, 1996). In construction supervision, this requirement is particularly relevant because heterogeneous records, including daily reports, correspondence, images, meteorological data, financial schedules and contractual information, must be transformed into reliable evidence for monitoring, reporting and decision support.

Structured outputs are also important for reproducibility and auditability. In analytical pipelines, explicit schemas and validated data objects help define clear data contracts between components, reducing ambiguity and limiting silent errors during data exchange. This principle is consistent with data engineering practices in which intermediate and final artefacts should be standardised, documented and suitable for later verification (Kleppmann, 2017; Reis & Housley, 2022). In the proposed workflow, this is reflected in the production of tabular exports, Comma-Separated Values (CSV) files, JSON Lines (JSONL) logs, HTML reports and curated Gold-layer datasets. In the Site Image Risk Analyser, typed outputs define validated objects for computer vision results, LLM-based interpretations and fused risk classifications. Although tools such as Pydantic are mainly technical resources, their use supports schema-based validation and contributes to more controlled data exchange between pipeline stages (JSON Schema, 2026; Pydantic, 2026).

Auditability becomes especially important when analytical systems produce classifications, risk indicators or interpretative summaries. In construction safety analytics, computer vision has been increasingly explored for detecting unsafe behaviours, personal protective equipment, workers, machinery and hazardous site conditions (Fang, Love, Luo, & Ding, 2020; J. Lee et al., 2023; W. Liu, Meng, Xing, Zhou, & Wang, 2021; Nath, Behzadan, & Paal, 2020). However, image-based safety analysis is not only a detection problem. Visual ambiguity, occlusion, incomplete context and model uncertainty mean that automated outputs should be interpreted as decision-support evidence rather than final safety judgements. For this reason, deterministic rule-based logic can provide an additional control layer by producing repeatable classifications from the same inputs and by making conflicts between analytical branches visible for human review.

Dashboards and reports represent the communication layer of a data analytics system. Information visualisation literature argues that visual representations should reduce cognitive load, support pattern recognition and help users reason about complex data (??). In business intelligence and decision-support contexts, dashboards are commonly used to organise

indicators, monitor trends and present information in a form that supports managerial interpretation (Sharda, Delen, & Turban, 2018). In construction management, dashboard-based approaches have been applied to monitor progress, cost, quality, safety, resources and project performance, especially when data originate from multiple sources and require integrated interpretation (Ghosh, Mukherjee, & McCuen, 2024). In this work, dashboards, Structured Query Language (SQL) queries, HTML reports and exportable files act as analytical communication artefacts that translate curated datasets into information that can be inspected and used by supervision teams.

The final reporting layer extends this logic through LLM assisted monthly report generation. Large language models can support summarisation, classification, question answering and report drafting, but their outputs are vulnerable to hallucination, unsupported generalisation and loss of traceability when they are not grounded in reliable evidence (Ji et al., 2023; Rawte, Sheth, & Das, 2023). For this reason, recent work on retrieval-augmented and evidence-grounded generation emphasises the importance of connecting generated text to external sources, structured context or retrieved evidence (Lewis et al., 2020). In the proposed methodology, the LLM is therefore not used as an autonomous source of truth. Instead, it receives curated evidence packages produced by previous analytical stages, including indicators, figures, tables, comments and selected outputs from different supervision domains.

This positioning is also consistent with AI governance literature. The NIST AI Risk Management Framework emphasises that AI systems should be governed, measured, managed and monitored, particularly when their outputs may influence consequential decisions (National Institute of Standards and Technology, 2023a). In this dissertation, AI-assisted reporting is therefore understood as a data analytics output layer and not as autonomous decision-making. Its purpose is to organise and communicate evidence generated by the pipeline, helping to produce coherent monthly supervision reports from multiple data sources. The convergence of the LLM output towards the intended report is achieved through structured inputs, predefined prompts, configuration settings, reporting templates and human review. Final conclusions remain dependent on professional validation, ensuring that the analytical workflow supports decision-making without replacing technical responsibility.

3.11 Summary

The reviewed literature shows that the construction sector is undergoing a gradual digital transformation, supported by developments in Building Information Modelling, data engineering, automation, data analytics, machine learning, computer vision, multimodal artificial intelligence, dashboarding, and assisted reporting (Oesterreich & Teuteberg, 2016; Olanipekun & Sutrisna, 2021; Radman et al., 2022). However, the adoption of these methods remains uneven, particularly in activities that have traditionally been less computationally supported. Construction supervision is one such area, where operational practice still depends heavily on documents, spreadsheets, email correspondence, photographic records, payment information, schedules, and manually consolidated reports (Antunes, 2012; de Vilhena Costa et al., 2024).

A significant part of digitalisation in the construction sector has been centred on Building Information Modelling. BIM responds to an important need for structured project information, collaborative information management, interoperability, and digital representation of built assets across their life cycle (EU BIM Task Group, 2017; International Organization for

Standardization, 2019). In Portugal, this importance is reinforced by the emerging legal and policy framework, including Portaria n.º 255/2023, Decreto-Lei n.º 10/2024, and Resolução do Conselho de Ministros n.º 89/2026, which progressively frame BIM within public works project preparation, urban licensing, and the national PortugalBIM strategy (Presidência do Conselho de Ministros, 2026; República Portuguesa, 2023, 2024). Nevertheless, BIM alone does not eliminate the need for complementary analytical workflows. Even when part of the project information is modelled and structured, supervision activities continue to generate heterogeneous operational evidence, including correspondence, inspection records, photographs, measurements, financial schedules, weather data, and monthly reports. These records require extraction, cleaning, integration, querying, visualisation, risk identification, financial analysis, and AI assisted interpretation.

The literature also indicates that BIM implementation and broader digitalisation remain affected by organisational, technical, financial, and skills-related barriers, particularly in late-adopting contexts and in small or medium-sized organisations (Lourenço, Arantes, & Costa, 2025). Advanced data analytics, machine learning, and artificial intelligence solutions are often associated with specialised consultancy, dedicated technical teams, cloud platforms, and significant implementation costs. This creates a risk of technological dependence, where organisations remain reliant on external expertise to develop, maintain, and adapt analytical workflows. For supervision companies or project teams with limited internal informatics resources, this dependence can restrict experimentation and slow the adoption of data-driven supervision practices.

Within this context, the contribution of this dissertation is deliberately limited in scope, but practical in orientation. It does not aim to replace enterprise platforms, full BIM based ecosystems, or specialised consultancy services. Instead, it provides an accessible proof of concept showing how construction supervision data can be progressively organised and explored through locally deployable methods. The proposed workflow combines data processing, SQL structuring, document extraction, Outlook correspondence analysis, IPMA meteorological data, contract price revision, financial schedule analysis, computer vision, multimodal interpretation, deterministic data fusion, and structured reporting. The objective is to demonstrate how heterogeneous supervision evidence can be connected into a traceable workflow for monthly reporting and decision support.

Therefore, the main gap identified in the literature is not the absence of individual technologies, but the limited integration of these technologies into lightweight, cost-conscious, and supervision-oriented methodological frameworks. The value of this work lies in showing that data-driven supervision can be explored independently of the size of the company or the existence of extensive in-house informatics resources, provided that the workflow is structured, traceable, and adapted to the operational realities of supervision practice.

Chapter 4

Proposed Methodology and Computational Implementation

This chapter presents the methodological framework adopted in this thesis and outlines the procedures followed during the research process. It presents a set of analytical tools developed to exemplify the support proposed for the construction supervision data methodology. Rather than isolated scripts, these tools should be seen as modular components of an integrated analytical toolkit. Each module addresses a specific source of supervision evidence, including daily reports, e-mail communications, meteorological records, financial and planning data, and site images. They are examples of a broader approach, since additional tools could be developed for other supervision and project-management needs. The outputs produced by these modules are structured in reusable formats such as CSV, Excel workbooks, SQLite tables, charts, heatmaps, and HTML dashboards.

The last section introduces an API based LLM assisted reporting layer as a higher-level synthesis mechanism. Is not a fully autonomous AI agent. Instead, it follows a controlled workflow that collects, organises, and summarises the outputs produced by the analytical modules, builds a structured evidence package, and sends it to an LLM through an external API. Its purpose is not to replace the individual tools or professional judgement, but to support the preparation of an integrated and reviewable monthly supervision report.

4.1 CRISP-DM, Medallion Architecture and Governance Principles

The methodology adopted in this dissertation combines CRISP-DM as a general data analytics process reference with a Lakehouse inspired Medallion organisation as the operational structure of the implemented prototype. As discussed in Chapters 2 and 3, CRISP-DM provides the general analytical logic for moving from problem understanding to data preparation, analysis, evaluation and delivery of outputs (Chapman et al., 2000). In this work, that logic was adapted to a modular data analytics workflow applied to construction supervision records.

The complementary operational structure was provided by the Bronze, Silver and Gold organisation. The Bronze layer preserved raw or near-raw supervision evidence, the Silver layer supported extraction, cleaning, structuring and validation, and the Gold layer concentrated curated outputs prepared for analysis, visualisation, reporting and decision support.

This Medallion style organisation was particularly relevant because the source data were heterogeneous, including documents, spreadsheets, Outlook correspondence, API based meteorological data, financial records, contractual information and site images (Armbrust et al., 2021; Azure Databricks, 2026; Databricks, 2026c).

In practical terms, the CRISP-DM phases were not applied as a rigid linear sequence. Business understanding was reflected in the identification of supervision needs, such as monitoring, reporting, contractual interpretation, financial analysis and risk identification. Data understanding and data preparation were implemented through the inspection, extraction and structuring of the different data sources. The analytical phase was represented by the Python-based modules developed for Outlook analytics, meteorological analysis, daily inspection record consolidation, resource comparison, financial schedule analysis, price revision calculation, image risk analysis and dashboard generation. Evaluation was performed through the inspection of logs, intermediate files, visualisations, consistency checks and generated outputs. Deployment was represented by Gold-layer artefacts such as Excel workbooks, CSV files, SQLite tables, HTML dashboards, figures, evidence packages and LLM-assisted monthly reports.

The governance references mentioned in Chapter 3 were not implemented as formal compliance frameworks. Instead, they informed the methodological emphasis on traceability, auditability, data quality, risk awareness and human validation. In the implemented workflow, these principles are reflected in the preservation of original evidence, the production of structured outputs, the use of logs and evidence packages, the separation between deterministic calculations and LLM generated interpretation, and the requirement for professional review before formal use of generated reports (National Institute of Standards and Technology, 2023a).

Therefore, the proposed methodology should be understood as a hybrid data analytics methodology. CRISP-DM provides the general analytical reasoning, the Medallion structure provides the operational data organisation, and governance principles support the concern with reliability, traceability and responsible AI-assisted reporting. This combination is aligned with the objective of the dissertation: to demonstrate a lightweight, local and reusable approach for transforming fragmented construction supervision records into structured, auditable and decision-ready analytical outputs.

4.2 Bronze Synchronisation

The Bronze synchronisation stage updates the controlled repository of raw supervision evidence from the operational source repository. Its purpose is to ensure that the Bronze layer contains the most recent available files for the projects already included in the defined scope, without altering the original nature of the records or introducing uncontrolled project folders.

The routine was implemented as a Python script and follows a conservative ingestion logic. Project folders are matched between the operational source and the Bronze repository using the initial project code, while files are copied only when they are new or when a newer version with different content is detected. The process can be executed first in dry-run mode, allowing the proposed actions to be reviewed before any files are copied, and then in apply mode to perform the approved updates.

The workflow also produces synchronisation reports, including ignored source projects, duplicated project keys, skipped files, and copied files. These reports support transparency,

verification, and reproducibility of the ingestion process. The Bronze layer is therefore refreshed without deleting existing records, preserving its role as a controlled repository of raw evidence for downstream extraction and analytical processing.

4.3 Outlook Analytics

The Outlook analytics workflow was developed to transform project-related email communication into structured, portable and reusable analytical evidence. The workflow separates extraction from analysis, so that the analytical stage can be performed without reconnecting to Outlook. This is particularly relevant when the extracted dataset needs to be migrated to another computer or reused in later reporting tasks.

The process is organised into three analytical layers. The Bronze layer preserves the extracted Outlook data in a structured database. The Silver layer supports exploratory analysis, validation and data-quality assessment. The Gold layer produces curated analytical outputs through an interactive dashboard and exportable evidence tables. Figure 4.1 summarises the overall workflow, showing the transition from structured extraction to analytical exploration.

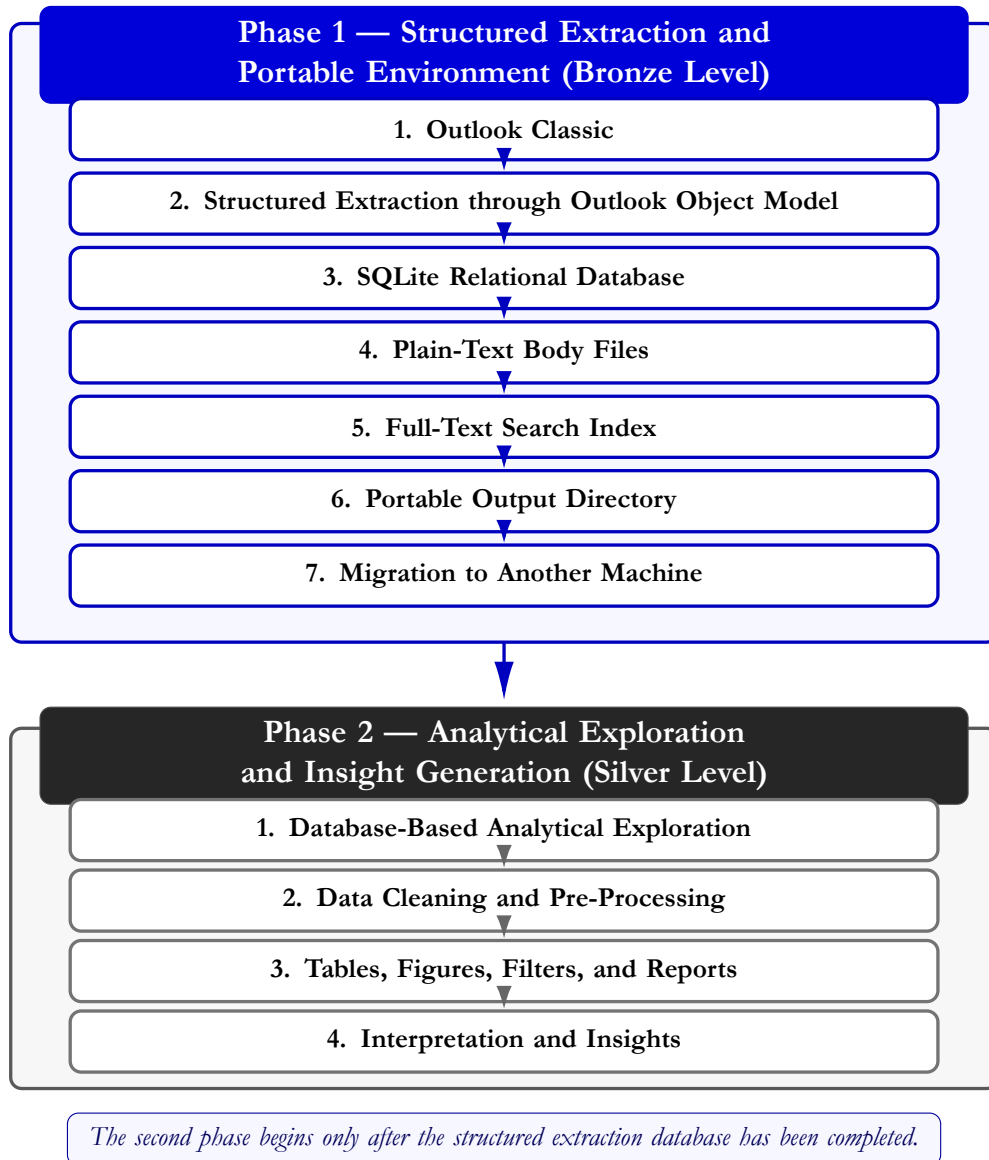


Figure 4.1: Outlook extraction and analytical exploration workflow.

4.3.1 Bronze Layer: Outlook Extraction and Structured Preservation

The Bronze layer extracts information from Outlook Classic using the Outlook Object Model through the Windows COM interface. The routine recursively scans Outlook stores, folders and subfolders, selecting only email items and ignoring other Outlook objects. For each email, it records message metadata, sender information, recipients, attachments, folder path, conversation identifiers, timestamps and references to the saved plain-text body.

The extracted data are stored in a local SQLite database, `outlook_scraped_master.db`, together with associated text files, optional CSV exports, logs and a manifest file. SQLite was adopted because it provides a portable relational structure suitable for local querying and migration between computers (Python Software Foundation, 2026d; SQLite Consortium, 2025b). The database separates emails, recipients, attachments and folders into related tables, reflecting the structure of the Outlook Object Model, in which each email may contain several recipients and attachments (Microsoft, 2024a, 2024b, 2024c).

At this stage, the objective is preservation rather than interpretation. The Bronze database keeps the extracted values as close as possible to the original mailbox information, while applying only technical structuring operations such as identifiers, relational tables, body-file references and error logging. This preserves traceability and allows later analytical procedures to be repeated without reconnecting to Outlook.

4.3.2 Silver Layer: Exploratory Data Analysis and Validation

The Silver layer reads the Bronze SQLite database and performs exploratory analysis and validation. Its purpose is to understand the structure, quality and analytical potential of the extracted dataset before producing final outputs. This stage includes summary tables, sender and recipient statistics, attachment analysis, temporal distributions, subject-word frequencies and validation of the sent/received classification.

This intermediate layer supports data understanding and quality control. It helps detect missing values, unusual folder structures, dominant senders, communication peaks and possible classification errors. The exploratory figures and tables are therefore not treated as final evidence, but as instruments for checking whether the extracted dataset is coherent and suitable for further analysis. This follows the logic of exploratory data analysis, where visual and tabular summaries are used to identify patterns and anomalies before formal interpretation (Tukey, 1977).

4.3.3 Gold Layer: Interactive Dashboard and Reporting Outputs

The Gold layer produces the final analytical outputs. It reads the structured database and generates an interactive HTML dashboard, exportable tables and supporting files for reporting. The dashboard allows the user to explore communication data by folder, sender, date interval, subject terms, body keywords and sent/received status.

The dashboard includes folder-level statistics, temporal charts, sender and receiver analysis, filtered email lists, subject-word sensitivity analysis and keyword-based search over saved body-text files. These functions connect aggregate indicators to individual email records, making it possible to move from general patterns to inspectable documentary evidence. Filtered results can also be exported to Excel-compatible files, supporting documentation and reuse in reporting (pandas development team, 2026a, 2026b).

The keyword based search follows a transparent rule based retrieval approach. Keyword themes are defined externally in keys.xlsx, allowing the analytical vocabulary to be updated without modifying the code. This supports thematic exploration while keeping the process auditable. Although this method does not capture all semantic relations, it is suitable for exploratory identification of relevant communication records (?).

4.3.4 Assessment of the Workflow

The Outlook analytics workflow provides a local Medallion-style analytical block for communication data. The Bronze layer preserves the extracted mailbox evidence, the Silver layer supports exploratory validation, and the Gold layer produces curated outputs for interpretation, reporting and export. This staged organisation is consistent with the principle of progressively transforming raw or near-raw data into analysis-ready information (Databricks, 2026c).

The main strength of the workflow is its separation between evidence preservation and interpretation. The original extracted database remains unchanged, while analytical scripts generate derived outputs that can be reviewed, exported and improved. This improves reproducibility, transparency and reuse, in line with general data-management principles such as the FAIR principles (Wilkinson et al., 2016).

The workflow also has limitations. Keyword and word-frequency methods depend on explicit terms and may not capture synonyms, implicit references or contextual meaning. For this reason, the dashboard should be interpreted as an exploratory decision-support tool rather than as an automatic semantic classification system. Nevertheless, it provides a structured and auditable way to analyse large volumes of project-related email communication.

4.3.5 Subject Sensitivity and Network Analysis

Two additional analytical components were implemented in the Gold layer to extend the Outlook dashboard beyond descriptive statistics: subject sensitivity analysis and communication network analysis. These components were developed in Python and integrated with the existing SQLite-based Outlook workflow.

The subject sensitivity analysis was implemented inside the HTML dashboard generated by the main Gold script. The Python code prepares the e-mail records with pandas, converts the analytical tables to JSON with the standard json library, and embeds them in the HTML output. The interactive calculations are then performed in JavaScript inside the dashboard, allowing the user to filter by folder and actor, compare different subject-word preprocessing scenarios, and inspect rankings, robustness scores and stability graphics. The method is supported by information retrieval principles related to term occurrence, ranking and relevance assessment (P).

The communication network analysis was implemented as an independent Gold-layer Python script. It reads the Bronze SQLite database using the standard sqlite3 library and processes the email and recipient tables with pandas. Sender–recipient relations are created by merging the e-mail and recipient tables, then aggregated into weighted network edges and actor-level node metrics. The script exports CSV files and an Excel workbook using pandas and openpyxl. When available, networkx is used to calculate centrality indicators and export GraphML files for later inspection in external graph-analysis tools (Hagberg et al., 2008).

These two modules add complementary analytical perspectives to the Outlook workflow. Subject sensitivity improves the robustness of textual interpretation, while network analysis provides a relational view of communication flows within each folder or project context. Both modules remain traceable to the Bronze database and produce exportable outputs, supporting transparency, reproducibility and later human validation.

4.4 Meteorological Data Analytics

4.4.1 Purpose and General Architecture

The availability of continuous and reliable meteorological data is relevant for environmental monitoring, fieldwork planning, risk assessment and construction supervision contexts where weather conditions may influence operational or analytical outcomes. In this work, meteorological observations were obtained from the automatic weather station network operated by

the Portuguese Institute for Sea and Atmosphere (Instituto Português do Mar e da Atmosfera (IPMA)). IPMA provides open access to meteorological data through its public API, including station metadata and recent hourly observations from automatic meteorological stations (IPMA, 2026c, 2026d).

An Application Programming Interface (API) is a structured communication mechanism that allows one software system to request and receive data from another system through predefined rules and endpoints (Fielding, 2000; Fielding, Nottingham, & Reschke, 2022). In this work, the IPMA API was used as the communication layer between the local Python script and the IPMA meteorological data service. The script sends HTTP requests to predefined IPMA endpoints and receives meteorological data in JSON format, allowing station metadata and hourly observations to be collected automatically without manual download (Ecma International, 2017; IPMA, 2026c, 2026d).

The implemented workflow follows a Medallion-style data architecture, in which data evolve from raw ingestion to progressively more structured, curated and analytical outputs (Databricks, 2026a). In the present work, this concept was adapted to meteorological data processing through three layers. The Bronze layer preserves the raw hourly observations retrieved from IPMA. The Silver layer structures, cleans, aggregates, classifies and visualises the data. The Gold analytical layer corresponds to the interpretation of the resulting reports, supporting the identification of meteorological patterns, station variability and reporting evidence.

The methodology identifies the three nearest IPMA meteorological stations to a predefined study location. The closest station is retained as S1 and used as the primary operational meteorological reference. The second and third nearest stations, S2 and S3, are retained as comparison stations. This allows the analysis to remain anchored in a single reference station while also providing transparency regarding local spatial variability.

The collection process was configured for automatic execution because the workflow depends on recent hourly observations available from the IPMA API. To reduce the risk of gaps in the local archive, Windows Task Scheduler was used to launch the meteorological collection routine twice per day: once when the computer starts and once at 10:00. The scheduled task does not execute the Python code directly; instead, it launches a batch file located in the Bronze meteorological directory. This batch file changes to the correct working directory, records execution information in a log file, waits briefly for the internet connection to become available, and then runs the Python script responsible for collecting and processing the data.

This separation between scheduling and execution improves operational control. Windows Task Scheduler defines when the process is triggered, the batch file defines how the Python routine is launched, and the Python script performs the actual data collection, database update and report generation. The operational script produces two complementary families of outputs: continuous hourly reports, which preserve the temporal detail of the raw observations, and aggregated morning and afternoon period reports, which provide reduced indicators for structured interpretation and reporting.

4.4.2 Study Location, API Access and Station Selection

In the first step of the procedure, the geographical location of the study area (WGS84 latitude and longitude coordinates, from reference system EPSG:4326, stored in a config.json file),

are used to calculate the spatial relationship between the project location and each available IPMA meteorological station.

The system uses two main IPMA API resources. The first is the station metadata file, `stations.json`, which contains information about the available automatic meteorological stations. This file is used to obtain each station's identifier, name and geographic coordinates. The second resource is the hourly observation file, `observations.json`, which contains recent meteorological observations from the automatic station network.

The station-selection procedure is objective and reproducible. After downloading the IPMA station catalogue, the script calculates the distance between the study location and each available station using the Haversine formula in Appendix A (Sinnott, 1984). The three stations with the shortest calculated distances are selected and stored. S1 is used as the operational reference, while S2 and S3 are kept as comparison stations.

This strategy avoids hard-coding a station identifier and makes the script adaptable to different study locations. It also improves transparency because the reports show not only the primary station but also nearby alternatives that may help inspect local spatial variability. Nevertheless, proximity alone does not guarantee perfect microclimatic representativeness. Factors such as altitude, exposure, distance from the coastline, slope, land use and local topography should also be considered during interpretation.

The variables collected include air temperature, relative humidity, accumulated precipitation, wind speed, wind direction, atmospheric pressure and radiation, whenever these parameters are available for the selected station and observation time. These variables correspond to standard meteorological parameters commonly used in surface weather observation.

4.4.3 Database Structure and Station Selection Model

The meteorological workflow uses a local SQLite database to preserve the collected data and the station-selection metadata. SQLite was selected because it is lightweight, portable and sufficient for the volume of data generated by daily meteorological observations. The database structure was designed to maintain raw observations, aggregated period records and station information in separate but related tables. This separation improves traceability because the raw evidence, processed indicators and spatial station metadata can be inspected independently.

The table `study_location` stores the project location used as the spatial reference for station selection. The table `ipma_station` stores the metadata of the IPMA meteorological stations retrieved from the station endpoint, including station identifier, station name, latitude, longitude and calculated distance to the project location. The table `selected_stations` stores the three nearest stations ranked by distance from the project location. In this table, station S1 corresponds to the nearest station and is used as the primary operational reference, while S2 and S3 are retained as comparison stations.

The raw hourly meteorological observations are stored in the table `weather_raw_hourly`. This table uses the combination of observation timestamp and station identifier as its primary key. This design prevents duplicate records while allowing observations from more than one station to be stored in the same raw-data table. As a result, the transition from a single-station workflow to a three-station workflow does not require deleting or restructuring the previously acquired raw observations.

This database model keeps the previously acquired data compatible with the updated

implementation. Existing records from the original nearest station workflow remain valid, while new executions progressively add observations and period indicators for the second and third nearest stations.

The entity relationship diagram in Figure D.1 summarises this database model and shows how the main tables are connected. The diagram highlights the separation between project location metadata, IPMA station metadata, ranked station selection, raw hourly observations and aggregated daily period records. It also shows how the same raw and aggregated tables can support the three-station model, since observations and period indicators are linked to the corresponding station_id. This structure allows S1 to be used as the primary meteorological reference while S2 and S3 remain available for comparison without changing the integrity of the historical archive.

4.4.4 Data Quality, Continuous Records and Period Aggregation

The raw data may contain missing or invalid values. In the implemented procedure, values coded as -99.0 are treated as missing data and converted to null values before processing. This conversion is necessary because retaining missing-data codes as numerical values would distort statistical calculations such as means, minima, maxima and accumulated totals.

For each daily period, the number of valid hourly observations is counted. Each aggregated record is then assigned a quality class, as shown in Table 4.1.

Table 4.1: Quality classes assigned to each aggregated period.

Quality class	Meaning
Complete	All expected hourly observations are available
Partial	One or more observations are available, but not all
No data	No valid observation is available for the period

This classification allows the reliability of each aggregated record to be evaluated. It also makes it possible to identify periods affected by missing observations, API unavailability, station transmission problems or interruptions in the local collection routine.

The meteorological workflow produces two complementary analytical datasets. The first dataset preserves the continuous hourly observations retrieved from the IPMA API. These records are stored in the raw hourly table and are used to generate all data reports, where temperature, precipitation and wind are represented as continuous time series throughout the day. This view is important because it preserves the full temporal behaviour of the meteorological variables, allowing sudden changes, daily cycles, peaks, low values and short-duration events to be inspected without being hidden by aggregation.

The second dataset aggregates the hourly observations into two daily working periods: morning and afternoon. The morning period is defined using the observations at 08:00, 09:00 and 10:00, while the afternoon period is defined using the observations at 14:00, 15:00 and 16:00. These period-based records are used to generate the stats data reports, which provide a more compact representation of the meteorological conditions during selected working windows.

For each period and station, the meteorological indicators listed in Table 4.2 are calculated.

Table 4.2: Meteorological indicators calculated for each daily period.

Indicator	Aggregation method
Mean temperature	Mean of valid hourly air temperature values
Minimum temperature	Lowest valid hourly air temperature value
Maximum temperature	Highest valid hourly temperature value
Mean relative humidity	Mean of valid hourly humidity values
Accumulated precipitation	Sum of valid hourly precipitation values
Mean wind speed	Mean of valid hourly wind speed values
Maximum wind speed	Highest valid hourly wind speed value
Mean atmospheric pressure	Mean of valid hourly pressure values
Mean radiation	Mean of valid hourly radiation values

The inclusion of minimum and maximum temperature improves the thermal characterisation of each period. These values represent the lowest and highest hourly temperature observations within the selected period. They should therefore be interpreted as period-based hourly extremes, not necessarily as absolute instantaneous extremes for the whole day.

The continuous hourly reports and the aggregated period reports serve different but complementary purposes. The continuous records provide greater temporal detail and are more suitable for identifying intra-day variability, abrupt changes and peak conditions. The morning and afternoon aggregations reduce dependence on a single hourly value and provide concise indicators for comparison between working periods. Together, these two levels of analysis support both detailed meteorological inspection and structured reporting.

4.4.5 Classification Criteria for Interpretative Graphs

The interpretative graphs are based on simple and transparent classes. For classification summaries, the primary station, S1, is used as the operational reference. The comparison stations S2 and S3 are represented in line plots but are not included in the pie charts or rain-count charts.

Temperature classes are defined operationally using the mean temperature of each period, as shown in Table 4.3.

Table 4.3: Operational temperature classes.

Class	Criterion
Low temperature	$< 10^{\circ}\text{C}$
Moderate temperature	$10\text{--}25^{\circ}\text{C}$
High temperature	$> 25^{\circ}\text{C}$
No data	No valid temperature value

These thresholds are operational and can be adapted to the climatic context of the study area. Unlike precipitation and wind, there is no single universal low/moderate/high air temperature classification that is valid for all regions, seasons and applications. Therefore, the thresholds should be interpreted as analytical classes designed to support comparison between periods.

Precipitation is classified using the mean hourly precipitation intensity within each period. Since each period contains up to three hourly observations, the total precipitation of the period is divided by the number of valid observations to estimate an average intensity in mm/h. The proposed classes are shown in Table 4.4. The thresholds are consistent with the IPMA classification of precipitation intensity, which defines weak precipitation as less than 0.5 mm/h, moderate precipitation as 0.5 to 4 mm/h and strong precipitation as more than 4 mm/h (IPMA, 2026a).

Table 4.4: Precipitation classes used in the interpretative graphs.

Class	Criterion
No precipitation	0 mm
Low / weak precipitation	> 0 and < 0.5 mm/h
Moderate precipitation	0.5–4 mm/h
High / strong precipitation	> 4 mm/h
No data	No valid precipitation value

Wind is classified using the mean wind speed of each period. The simplified wind classes are shown in Table 4.5. These thresholds are based on the IPMA wind classification, in which weak wind is below 15 km/h, moderate wind is between 16 and 35 km/h, and strong wind is between 36 and 55 km/h (IPMA, 2026b).

Table 4.5: Wind classes used in the interpretative graphs.

Class	Criterion
Low / weak wind	< 15 km/h
Moderate wind	15–35 km/h
High / strong wind	> 35 km/h
No data	No valid wind value

For the purposes of this study, the higher IPMA wind classes were grouped into the broader “high / strong wind” category, because the objective of the graphs is to provide a simple low–moderate–high interpretation.

4.4.6 Analytical Interpretation and Methodological Limitations

The Gold analytical layer corresponds to the interpretation of the structured datasets and dashboards produced in the Silver layer. This layer does not correspond to a separate raw data store, but rather to the extraction of analytical insight from processed information. The dashboards make it possible to identify dominant meteorological conditions, such as weeks or months characterised by high temperature, frequent precipitation or stronger wind regimes. The comparison between S1, S2 and S3 in the temporal line plots also allows local spatial consistency or divergence to be inspected. This is useful because nearby stations may show similar trends but different magnitudes due to local exposure, altitude or terrain.

The count of periods with and without precipitation is particularly useful for operational interpretation. In many fieldwork and environmental contexts, the occurrence of rain may

be more relevant than the total precipitation amount alone. Similarly, the distribution of wind and temperature classes provides a compact overview of the dominant meteorological conditions during each reporting interval.

The main limitation of this methodology is that the selected meteorological stations may not fully represent the microclimatic conditions of the exact study location. Even when the nearest station is used, local differences in altitude, exposure, urbanisation, vegetation or proximity to water bodies may influence meteorological conditions. The use of three nearby stations improves transparency but does not eliminate this limitation. S1 is retained as the operational reference, while S2 and S3 provide spatial comparison. Differences between the three stations should therefore be interpreted as contextual evidence of local meteorological variability, not as direct measurements at the project location.

Another limitation is related to the temporal availability of public hourly observations. Since the system depends on recent hourly data, regular execution is necessary to avoid gaps. The use of an automatic execution at user logon and a second scheduled execution at 10:00 reduces this risk but does not eliminate it if the computer remains off for long periods.

The inset map depends on geographic coordinates and online or cached map tiles. If the basemap is temporarily unavailable, the meteorological data collection and report generation can still be interpreted from the numerical and graphical outputs, but the geographic context may be incomplete in that specific execution.

The temperature, precipitation and wind classes used in the interpretative graphs are intended to support descriptive analysis. The precipitation and wind thresholds are based on IPMA classifications, while the temperature thresholds are operational and should be adapted if the study requires a different thermal interpretation. Finally, the Silver reports are updated while the corresponding reporting interval is still open. Therefore, weekly, monthly, quarterly and annual dashboards should be interpreted as provisional until the end of their respective period.

4.5 Inspection Records Consolidation

The procedure developed in this work implements a digital strategy for transforming daily construction inspection reports into structured, auditable and analysable datasets. The starting point is a set of daily inspection records in .docx format, stored in a hierarchical folder structure organised by month. These reports contain qualitative information on site activities, inspection actions and notes, as well as quantitative information on labour and equipment resources. The developed Python script scans the relevant folders, reads each report, extracts the required information, consolidates it by month, and generates Excel outputs containing matrices, logs and graphical summaries.

This approach is aligned with the current direction of construction project-control research, which increasingly emphasises the conversion of site records into structured digital information. Traditional construction monitoring is commonly described as labour-intensive, time-consuming and dependent on manual documentation, which may reduce accuracy and delay decision-making (Nakanishi, Kaneta, & Nishino, 2022). Automated progress-monitoring literature similarly highlights the need to transform dispersed site information into systematic, usable project-control data (Pal, Lin, Hsieh, & Golparvar-Fard, 2023). Therefore, the developed procedure follows a data-driven construction-monitoring logic, even though its primary data source is not a sensor, a camera or a BIM model, but the official daily inspection

documentation already produced by the supervision team.

The strategy does not attempt to replace inspection judgement. Instead, it strengthens it by converting the daily records produced by the supervision team into a structured database. Daily reports remain a central source of construction information because they record activities, resources, constraints, decisions and relevant site events. Recent research also shows a growing interest in automating the generation and interpretation of daily construction reports, including approaches based on computer vision and language models (Xiao, Chen, & Golparvar-Fard, 2024). In this context, the present strategy can be understood as a practical, document-based implementation of digital construction monitoring.

4.5.1 Source Data and Folder Structure

The process starts from the project daily inspection report repository. The script is configured to read all .docx reports located under a specified root directory.

The original data "daily reports" are configured as presented in the following images.

Logo

REGISTO DIÁRIO DE FISCALIZAÇÃO

PROJECT OWNER
PROJECT DESIGNATION

Ref. #:

Revisão: 00

Data:

1. INFORMAÇÕES SOBRE AS CONDIÇÕES ATMOSFÉRICAS

Manhã

☐

☒

☐

☐

☐

Céu limpo

Pouco nublado

Muito nublado

Chuva fraca

Chuva intensa

Tarde

☐

☐

☐

☐

☐

Céu limpo

Pouco nublado

Muito nublado

Chuva fraca

Chuva intensa

2. ATIVIDADES REALIZADAS OU EM CURSO

2.1. FRENTE DE TRABALHO Nº1

Sem trabalhos nesta frente de obra;

2.2. FRENTE DE TRABALHO Nº2

Corte e Montagem das Armaduras da Caixa de Visita;

3. AÇÕES DESENVOLVIDAS PELA EQUIPA DE FISCALIZAÇÃO E COORDENAÇÃO DE SEGURANÇA EM OBRA

Elaboração do Relatório diário, com a indicação das atividades desenvolvidas, carga de meios humanos e dos equipamentos;

4. NOTAS

Nada a registar;

MOD.

Pág. 1/4

Logo

REGISTO DIÁRIO DE FISCALIZAÇÃO

PROJECT OWNER
PROJECT DESIGNATION

Ref. #:

Revisão: 00

Data:

5. REGISTO FOTOGRÁFICO



Frente de trabalho:

Frente de trabalhos nº1

Descrição:

Sem trabalhos nesta frente de obra;



Frente de trabalho:

Frente de trabalhos nº2

Descrição:

Corte e Montagem das Armaduras da Caixa de Visita;

MOD.

Pág. 2/4

Page 1

Page 2

Figure 4.2: Daily inspection record, pages 1 and 2.

Logo

REGISTO DIÁRIO DE FISCALIZAÇÃO

PROJECT OWNER
PROJECT DESIGNATION

Ref. #:

Revisão: 00

Data:



Fronte de trabalho:

Fronte de trabalho n°1

Descrição:

Corte e Montagem das Armaduras da Caixa de Visita;



Fronte de trabalho:

Fronte de trabalho n°2

Descrição:

Corte e Montagem das Armaduras da Caixa de Visita;

Logo

REGISTO DIÁRIO DE FISCALIZAÇÃO

PROJECT OWNER
PROJECT DESIGNATION

Ref. #:

Revisão: 00

Data:

6. RECURSOS

MÃO DE OBRA		EQUIPAMENTO	
CATEGORIA	Quant.	CATEGORIA	Quant.
Fente de trabalho n°1			
Diretor de Obra	1.0	Contentor Escritório	3.0
TQAS		Separação de Resíduos	1.0
Administrativo	1.0	Giratória de rastos	1.0
Representante EE		Fossa séptica estanque	1.0
Diretor Técnico EE		Contentor WC	1.0
Apoio DTE			
Topógrafo			
Projetista			
Encarregado			
Pedreiro			
Motorista Pesados			
Manobrador Mini-giratória			
Frente de traalho n°2			
Encarregado	1.0	Separação de Resíduos	1.0
Técnico de Grua (Somirav)		Grua Automontante	1.0
Armador de Ferro		Gerador	1.0
Carpinteiro		Carrinha transporte	1.0
Manobrador Mini-giratória		WC Químico	1.0
Soldador Eletromecânico		Contentor Pessoal	1.0
Servente		Máquina de Cortar/Dobrar Ferro	1.0
Eletricista		Balde de Betonagem	1.0
Pedreiro	1.0	Máquina de Soldar	
Motorista Pesados		Vibrador de Betão	
Pintor - Regrout		Camião de Autobomba	
Somirav		Camião de Betonagem	
Totais Diários	4.00	Totais Diários	15.00

Figure 4.3: Daily inspection record, pages 3 and 4.

The reports may be organised into monthly subfolders, such as September, October or November 2025. The script uses recursive search, which means that it is not necessary to process each month manually. This is relevant for projects with long execution periods, where manual aggregation of daily reports would be repetitive, time-consuming and more exposed to human error.

The code defines separate paths for input and output. The input folder corresponds to the official document repository, while the output folder corresponds to the working directory where the generated Excel files and graphical summaries are stored.

This separation preserves the original records and ensures that the analytical products are generated in a controlled location. The main configuration variables used for this purpose are `INPUT_DIR`, `OUTPUT_DIR` and `PESQUISAR_SUBPASTAS`. The use of a recursive search is consistent with a scalable document-processing strategy, since the same code can process one month or several months without changing the internal extraction logic.

4.5.2 Reading and Extracting Information from Word Reports

Each .docx report is opened and processed individually. The script uses the python-docx library to access the Word document content. This library is designed to create, read and manipulate Microsoft Word .docx files, making it appropriate for extracting the textual content of the reports (python-openxml, 2026). In this workflow, it is used mainly to read the report paragraphs and extract the relevant textual sections.

Before analysis, the text is normalised. This includes removing non-breaking spaces, standardising white spaces, removing accents for comparison purposes, and converting section titles to a common uppercase representation. This step is necessary because daily reports may contain small variations in headings, accentuation or spacing. The script therefore defines aliases for the main report sections, allowing it to recognise equivalent forms such as “AÇÕES” and “ACOES”.

The main extracted sections are:

- activities performed or in progress;
- actions developed by the inspection and safety coordination team;
- notes;
- resources.

The first three are treated as textual sections. The script reads the document body, identifies section headings and assigns each line to the corresponding section until another relevant section begins. For activities, the script also detects likely work-front or location headings, such as *Frente de trabalho nº1* or *Frente de trabalho nº2*. When such a heading is detected, the activity is prefixed with that location. This avoids ambiguity when similar activities occur in different work fronts.

This contextualisation is relevant for construction monitoring because activity descriptions without location may be insufficient for later analysis. The code implements this logic through functions for heading detection, line cleaning, item splitting and activity parsing.

4.5.3 Extraction of Resources from Embedded Excel Objects

The resources section required a specific extraction strategy. In the report model used in this project, labour and equipment are not reliably available as simple paragraphs in the Word body. Instead, they are stored in an Excel object embedded inside the .docx file. The script therefore opens the .docx file as a compressed package and searches for embedded .xlsx files inside the internal word/embeddings/ folder.

This is technically justified because .docx files are Office Open XML packages containing multiple internal files and folders. The Python zipfile module provides tools for reading and extracting ZIP archives, which makes it appropriate for accessing embedded objects within a Word document (Python Software Foundation, 2026e). Once the embedded Excel file is found, it is extracted temporarily and read using openpyxl. This library is designed to read and write Excel files, including .xlsx files, and is therefore suitable for extracting the resource table stored inside the reports (openpyxl, 2026).

The observed structure of the embedded resource sheet is:

- columns B/C: labour category and quantity;
- columns D/E: equipment category and quantity.

The code documents this assumption directly in the resource extraction function. It reads each row, ignores headers such as “MÃO DE OBRA”, “EQUIPAMENTO”, “CATEGORIA” and “QUANT.”, and accumulates all valid quantities by category. This produces two dictionaries per report, one for personnel and one for equipment.

This design is important because the same type of labour or equipment may appear more than once in the embedded table, for example under different work fronts. Instead of treating these repetitions as duplicates, the script sums them to obtain the total daily quantity.

4.5.4 Monthly Consolidation into Excel Matrices

After all reports are processed, the script groups them by year and month. This is necessary because the output matrices use the days of the month as columns. Creating one workbook per month avoids mixing different time periods in the same matrix and makes each file easier to interpret.

For activities, inspection actions and notes, the output is a presence matrix:

- rows represent activities, actions or notes;
- columns represent the days of the month;
- the cell value is “X” when the item occurred on that date.

For personnel and equipment, the output is a quantity matrix:

- rows represent personnel categories or equipment types;
- columns represent the days of the month;
- the cell value is the quantity recorded on that date.

The Excel files are created with openpyxl, which also formats headers, column widths, borders and frozen panes. Each monthly workbook contains the following sheets:

- *Activities*;
- *Inspection Actions*;
- *Notes*;
- *Personnel*;
- *Equipment*;
- *Resource Summary*;
- *Log*.

This structure transforms semi-structured daily reports into monthly datasets that can be inspected, filtered, archived and used in later analytical stages.

4.5.5 Daily Global Resource Summaries

In addition to category-level matrices, the script calculates daily global resource totals. For each report date, it sums all labour quantities and all equipment quantities:

$$\text{Total personnel}_d = \sum_{i=1}^n \text{Personnel category}_{i,d} \quad (4.1)$$

$$\text{Total equipment}_d = \sum_{j=1}^m \text{Equipment type}_{j,d} \quad (4.2)$$

$$\text{Total resources}_d = \text{Total personnel}_d + \text{Total equipment}_d \quad (4.3)$$

This produces a compact daily resource series that is easier to analyse visually than the detailed category matrices. The values are written to the *Resource Summary* sheet, which provides a daily table and graphical summaries of resource mobilisation. The global resource file aggregates all months into a single workbook, enabling the analysis of resource trends across the full project period.

This is consistent with resource-oriented construction-control practice. Monitoring equipment and labour resources supports project management and decision-making, particularly because worker and equipment tracking are key components of construction progress and productivity monitoring (Nakanishi et al., 2022; Sherafat et al., 2020).

4.5.6 Comparison with Contractor-Proposed Resource Loads

The most recent version of the strategy extends the analysis by comparing the actual resources recorded in the Registos Diários da Fiscalização (RDF) reports with the contractor resources proposed at bidding stage. Two external Excel files are used:

- Carga de Equipamento.xlsx;
- Carga de Pessoal.xlsx.

The purpose of this comparison is to move from simple resource recording to resource-control analysis. The RDF reports provide the actual daily mobilisation and presence on site, while the contractor proposal files provide the planned or proposed resource load. By plotting both series together, the script supports the identification of deviations between proposed and actual mobilisation.

This type of comparison is consistent with project-controls logic, where planned resource usage is compared with actual deployment to support progress evaluation and early detection of deviations. Resource-loaded planning and subsequent comparison with site records are common principles in construction control because they allow the supervision team to assess whether the resources mobilised are coherent with the expected production effort (Nakanishi et al., 2022; Sherafat et al., 2020).

In the updated workflow, the *Resource Summary* sheet is expanded to include:

- date;
- actual RDF personnel;
- actual RDF equipment;
- actual RDF total resources;
- proposed personnel;
- proposed equipment;
- proposed total resources.

The plots then show both actual and proposed values:

- actual personnel versus proposed personnel;
- actual equipment versus proposed equipment;
- actual total resources versus proposed total resources.

Keyword-Based Selector Mapping and Heatmap Analysis

To extend the analytical value of the Daily Inspection Report workflow, a Gold layer procedure was developed to identify construction related themes in the textual outputs produced at the Silver layer. The method is based on a keyword table, stored in *keys.xlsx*, where each selector, such as excavation, formwork, concreting or compaction, is associated with a set of related keywords. This approach follows the broader objective of transforming semi-structured construction documentation into structured datasets suitable for monitoring, analysis and decision support (Nakanishi et al., 2022; Pal et al., 2023).

The second input consists of the monthly Silver Excel files, that contain the consolidated daily records extracted from the original .docx reports, including the textual sheets *Atividades*, *Ações Fiscalização*, and *Notas*. The Gold script reads these sheets, normalises the text by converting it to lowercase, removing accents and punctuation, and then searches for the keywords associated with each selector.

When a keyword is found in a textual item marked as occurring on a given day, the corresponding selector is counted for that date. By repeating this process across all monthly

files, the script converts descriptive daily-report content into a structured temporal matrix of selector occurrences. This matrix can then be analysed at daily, weekly and monthly levels.

The main outputs are the *Daily Selector Map*, the *Weekly Heatmap*, the *Monthly Heatmap*, the *Match Details* sheet and a *README* sheet. The daily map shows selector occurrences by date, while the weekly and monthly heatmaps aggregate the same information at broader temporal scales. The *Match Details* sheet preserves traceability by recording the date, selector, matched keyword, source file, source sheet and original text that generated each match.

This approach provides a practical way to transform qualitative site descriptions into structured analytical information. It allows the supervision team to identify when specific activities or themes were reported, how frequently they occurred and how their intensity evolved over time. Although keyword-based methods may be affected by ambiguous terminology or project-specific wording, the preservation of match details allows the results to be checked and the keyword table to be refined.

The procedure was implemented in Python using pandas for tabular processing and aggregation, and openpyxl for reading and writing Excel workbooks. Conditional formatting was applied to the output sheets to support heatmap-style visual interpretation. Overall, the method offers a transparent and low-cost bridge between traditional daily reporting and data-driven construction monitoring, especially in projects where formal daily documentation is already systematically produced (Kavaliauskas, Fernandez, McGuinness, & Jurelionis, 2022; Vassena, Perfetti, Comai, Mastrolembro Ventura, & Ciribini, 2023).

4.6 Contract Price Revision

4.6.1 Implementation context

The contract price revision module operationalises the Portuguese price revision procedure already introduced in the theoretical and literature review chapters. This section focuses on how the procedure was implemented computationally. The calculation follows the principles established by Decree-Law No. 6/2004 and by the official formula-type structure defined for Portuguese public works, using IMPIC indices for labour, materials and support equipment (Diário da República, 2004a, 2004b, 2004c, 2004d; IMPIC, 2026b).

The purpose of this module is to transform the price revision process into a controlled and auditable workflow. The monthly payment certificate is not adjusted by a single inflation rate or by a single labour index. Instead, the certified value of the works is revised using a coefficient calculated from the applicable formula-type, the base-month indices and the indices corresponding to the monthly payment. This ensures that the calculation reflects the weighted structure of the relevant labour, material, equipment and fixed components.

4.6.2 Formula-type selection and calculation logic

The implemented procedure uses the selected *Formula_Tipo* from the contract file as the starting point for the calculation. This field identifies the applicable formula-type according to the nature of the works, following the official F01–F23 classification summarised in Table 3.1. The selected formula-type determines the set of index components and weights used to calculate the revision coefficient.

In general terms, the price revision coefficient is calculated as:

$$K_t = \sum_{i=1}^n p_i \frac{I_{i,t}}{I_{i,0}} + p_f \quad (4.4)$$

where K_t is the price revision coefficient for the a specific month payment, p_i is the weight of each revisable component, $I_{i,t}$ is the index of component i in that month, $I_{i,0}$ is the corresponding index in the base month, and p_f is the fixed or non-reviewable component, when applicable.

The revision amount is then obtained by applying the coefficient variation to the value of works executed in the period:

$$R_t = V_t (K_t - 1) \quad (4.5)$$

where R_t is the revision amount and V_t is the value of the works executed in the period, excluding VAT. The revised value of the payment certificate is therefore:

$$V_{t,rev} = V_t + R_t \quad (4.6)$$

The global monthly payment value is known after the coefficient has been calculated from the applicable formula components.

4.6.3 Computational procedure adopted

The computational procedure is based on three main input files: the contract and financial schedule file, the monthly payment certificate file, and the normalised indices file. The contract file contains the contract identification, base month, selected type of construction works (what have implications in the formula) and, when applicable, a contract-specific formula library. The monthly payment file contains the payment certificate number, the execution month and the value to be revised. The indices file stores the monthly values of the labour, material and equipment indices required for the calculation.

The code read Excel files, process tabular data and generate auditable outputs. The use of pandas supports the reading and processing of structured data such as contract tables, payment certificate records and normalised indices (pandas, 2026). The use of openpyxl supports the creation of Excel workbooks containing the summary, detailed calculation, formula source, indices used and historical records (openpyxl, 2026).

A central feature of the procedure is the formula selection hierarchy. The code first attempts to use a valid contract-specific formula from the sheet BiblNLPeca_Formulas. If the contract file does not contain valid component rows, the procedure searches the local official formula library biblioteca_formulas_oficiais.xlsx. If this file is missing or incomplete, it is created or updated from the official F01–F23 formula structures embedded in the script. If no valid formula can be obtained, the calculation stops instead of silently falling back to an unrelated or outdated formula.

The implemented hierarchy is therefore:

1. use a valid contract-specific formula from Biblioteca_Formulas, if available;
2. otherwise, use the validated local official formula library biblioteca_formulas_oficiais.xlsx;
3. if the official library is missing or incomplete, create or update it from the embedded official F01–F23 formula structures;

4. if no valid formula can be obtained, stop the calculation and report the problem.

The code structure assures application of the labour index, together with material and equipment, in a specific formula-type, to the full value of a specific month payment.

4.6.4 Treatment and validation of indices

The module first reads the existing local indices file and then attempts to update it using the available IMPIC sources. The local file works as a historical cache, while the update routine allows the dataset to incorporate newly published indices when they become available. This is relevant because official monthly indices may be published after the execution month and may not always be available at the time of preparing the payment certificate.

The code also normalises official Comissão de Índices e Fórmulas de Empreitadas (CIFE) style tables into a controlled structure containing fields such as month, index code, description, category and index value. This normalisation is necessary because the official files are not originally organised as a database table and may include labour formulae, material indices and equipment indices in separate sections (IMPIC, 2026a). The resulting structure allows the calculation routine to retrieve the required base-month and auto-month indices consistently.

A validation step is included before calculating the final result. If any required index is missing for either the base month or the auto month, the code does not continue silently. Instead, it creates an auxiliary Excel file listing the missing indices that must be completed or awaited. This improves the reliability of the process and supports later verification of the calculation.

4.6.5 Price revision workflow

In the implemented workflow, the process starts with the preparation of the contract file and the auto file. The contract file provides the base month, formula type and contractual information, while the auto file provides the execution month and the certified value of the works.

The code then selects the applicable formula, validates the required indices, calculates the coefficient K_t , applies it to the monthly value of the auto and generates the revised value. The final stage consists of producing the output workbook and updating the historical file of autos.

4.6.6 Treatment of monthly and cumulative payment certificates

A relevant implementation issue is the distinction between monthly and cumulative payment certificates. If the auto value corresponds only to the works executed in the period, the revision is calculated directly over that amount. However, if the auto value is cumulative, the code subtracts the previously certified accumulated value in order to isolate only the new value executed in the current period.

This control avoids double counting. Without it, the same production could be revised more than once in successive autos. For this reason, the procedure maintains a historical file of autos. This file stores, for each payment certificate, the period value, accumulated value, revision amount and accumulated revision. It therefore preserves the sequence of calculations and supports the traceability of the cumulative revision process.

4.6.7 Contribution to contract management

The implemented module improves the traceability and reproducibility of the price revision process. Each result workbook contains the summary of the calculation, the detailed coefficient calculation, the formula used, the indices used, the diagnostic information on the formula source and the updated history of autos. This makes it possible to verify the origin of each value and to reproduce the calculation if necessary.

Within the wider methodology, the price revision module connects the legal framework, the applicable formula-type, the official indices, the contractual information and the certified value of the works. It also reduces the risk of manual error, particularly when several autos are calculated over time, when cumulative payment certificates are used, or when different formula-types are tested during validation. The use of structured Excel inputs and outputs, supported by Python routines, contributes to a more auditable financial and contractual analysis process.

4.7 Financial Schedule Analysis

4.7.1 Planned and actual financial execution

The financial schedule is a time-based representation of the expected monetary production of a construction contract. It distributes the planned value of the works across the execution period and provides a baseline for monitoring the financial evolution of the contract. In project control, the baseline is essential because it allows actual execution to be compared with the approved plan (AACE International, 2021).

In the methodology adopted in this work, the planned financial schedule is obtained from the contract file. The relevant sheet contains the planned monthly value for each month of the contract. These values represent the expected financial production according to the approved schedule. The cumulative planned curve is obtained by summing the planned monthly values over time.

The actual financial schedule is obtained from the payment certificate history. Each auto represents certified production for a given period. The code uses the column `Valor_Periodo_Sem_IVA` from the `Historico_Autos` sheet as the actual monthly value. This value corresponds to the certified amount for that period, excluding VAT. The cumulative actual curve is then calculated as the cumulative sum of the actual monthly values.

This distinction is important because the actual curve must not be built from the planned schedule. The planned schedule represents what was expected, while the actual schedule represents what was certified. Therefore, the comparison between both curves provides a direct view of the financial development of the contract.

4.7.2 Computational workflow

The developed code follows a clear sequence. First, it reads the planned financial schedule from the contract file. Second, it searches the results folder for the most recent price revision result file. The most recent file is identified using the period and the auto number contained in the filename. Third, the code reads the `Historico_Autos` sheet from that file and extracts the actual monthly values from the field `Valor_Periodo_Sem_IVA`.

The data processing is performed with pandas, which is used to read Excel sheets, clean column names, merge planned and actual data, and calculate cumulative values (pandas, 2026). The output workbook is written with support from Excel writing libraries such as openpyxl, allowing the processed data, source notes and comparison tables to be stored in a structured workbook (openpyxl, 2026).

Progress tracking and comparison with the baseline schedule

In the context of construction project management, the Portuguese concept of *balizamento* may be understood as the periodic tracking of actual progress against the baseline initially presented in the bid. This comparison should not be limited to financial execution. Although the financial schedule provides an important view of the amounts planned and certified over time, it does not necessarily represent the real physical progress of the works. A delay in financial execution may indicate a delay in production, but it may also result from an unrealistic or poorly distributed financial schedule submitted at the tender stage. For this reason, the analysis of planned versus actual execution should be interpreted together with the physical sequence of activities, the critical path, the availability and productivity of labour, and the allocation of construction equipment.

The baseline schedule defines the planned sequence of operations and provides a reference for monitoring progress, deviations and corrective actions. Critical path analysis is particularly relevant because delays in critical activities directly affect the project completion date, whereas delays in non-critical activities may be absorbed by available float (AACE International, 2006; Project Management Institute, 2006). Therefore, a financial deviation alone is not sufficient to conclude that the project is delayed. A lower certified value may occur in activities with high financial weight but limited impact on the critical path, while apparently minor physical delays in critical operations may compromise the overall completion date.

A complete progress-tracking system should therefore integrate several dimensions: financial execution, physical quantities, labour resources, equipment resources, work fronts and the logical sequence of activities. This is consistent with project management guidance that emphasises integrated control of scope, schedule, cost and resources (International Organization for Standardization, 2020; Project Management Institute, 2019). Earned value approaches also support this integrated interpretation by relating planned value, earned value and actual cost, allowing performance to be assessed beyond simple expenditure or invoicing (Project Management Institute, 2004, 2019). In construction, however, such integration requires detailed and reliable information on measured quantities, work packages, activity relationships, productivity rates and resource mobilisation.

For this reason, although progress tracking is one of the most important instruments for project control, its full development would exceed the scope of the present work. A rigorous *balizamento* model would require an autonomous methodological framework capable of linking the financial schedule, the physical work plan, the critical path, labour and equipment allocation, and the actual sequence of operations. The financial schedule evolution developed in this work should therefore be interpreted as one component of a broader project-control system, useful for detecting deviations and supporting periodic analysis, but not sufficient on its own to determine the real delay or acceleration of the project.

4.8 AI Image Risk Analyser

The Image Risk Analyser is a research prototype developed to support automated construction site safety assessment from photographic records. The tool addresses the problem that construction site images often contain relevant evidence about workers, personal protective equipment, access conditions, temporary works, equipment, collective protection measures, and visible hazards, but these images are difficult to analyse manually at scale. The objective of the analyser is therefore not to replace human supervision, but to support the preliminary identification, structuring, and reporting of image-based safety risks.

The system adopts a dual-pipeline architecture. Each construction site image is processed through two independent analytical branches running in parallel. The first branch is a Computer Vision pipeline based on YOLOv8, which performs object detection and identifies safety-relevant elements such as persons, helmets, ladders, and other visible objects. This branch produces structured detections, including object labels, bounding boxes, and confidence scores. The second branch is a Large Language Model pipeline based on Claude Vision, accessed through the Anthropic API, which performs contextual scene interpretation and produces qualitative observations, risk factors, uncertainty notes, and narrative summaries.

The use of two independent pipelines is methodologically relevant because Computer Vision and Large Language Models provide different types of evidence. The Computer Vision branch is more structured and spatially explicit, since it identifies visible objects and their locations in the image. However, it has limited capacity to interpret the broader construction context. The Large Language Model branch can provide richer semantic interpretation of the scene, but its outputs are more narrative and may require stronger validation. Keeping the two branches separate allows their outputs to be compared, evaluated, and improved independently.

After both analyses are completed, the results are processed by a rule-based risk engine and a fusion engine. The risk engine interprets Computer Vision detections according to predefined safety rules, including personal protective equipment checks and hazard-related conditions. The fusion engine then combines the Computer Vision and Large Language Model outputs, identifying shared findings, CV-only findings, LLM-only findings, and explicit conflicts between the two branches. This design is important because disagreements between the two systems are not hidden; instead, they are made visible and can be flagged for manual review.

Each image produces three structured outputs: a CVResult, an LLMResult, and a FusionResult. The CVResult includes detected objects, inferred scene flags, and a summary of the detections. The LLMResult includes risk factors, observations, a narrative summary, and uncertainty notes. The FusionResult combines both branches and records shared findings, separate findings from each pipeline, conflicts, a combined risk level, and whether manual review is recommended. These outputs are defined through typed Pydantic schemas, making the data contract between pipeline components explicit and reducing the risk of silent inconsistencies during processing.

The architecture is organised around modular components. A file scanner recursively identifies images in the input directories, a batch processor coordinates the analysis of each image, and a parallel runner executes the Computer Vision and Large Language Model branches concurrently. The backend router instantiates the selected analytical backends from configuration, while the risk engine and fusion engine produce the final safety assessment. The

system also includes an event logger that writes a structured JSONL audit trail, supporting traceability and later review.

The outputs of the analyser are designed for reporting and integration with the broader supervision data pipeline. For each batch of images, the system generates an HTML report for visual inspection and a CSV export containing one row per image with the structured analytical fields. These outputs can support safety review, monthly reporting, dashboard integration, and the future development of AI-assisted supervision reports. In this sense, the Image Risk Analyser acts as a specialised analytical module within the broader methodology, transforming unstructured photographic evidence into structured, auditable, and reviewable risk information.

The system is explicitly experimental and should be interpreted as a decision-support prototype. Its combined risk verdicts are not intended to constitute autonomous safety decisions. Instead, the analyser supports prioritisation and evidence organisation by identifying possible risks, highlighting agreements and conflicts between analytical pipelines, and indicating cases where human review is required. This human-in-the-loop position is essential because construction safety interpretation depends on context, professional judgement, and responsibility that cannot be delegated entirely to automated systems.

As shown in Figure 4.4, the image risk analyser follows a dual-analysis strategy in which computer vision results and LLM-based contextual interpretation are fused before report generation.

AI Image Risk Analyser – Process Workflow

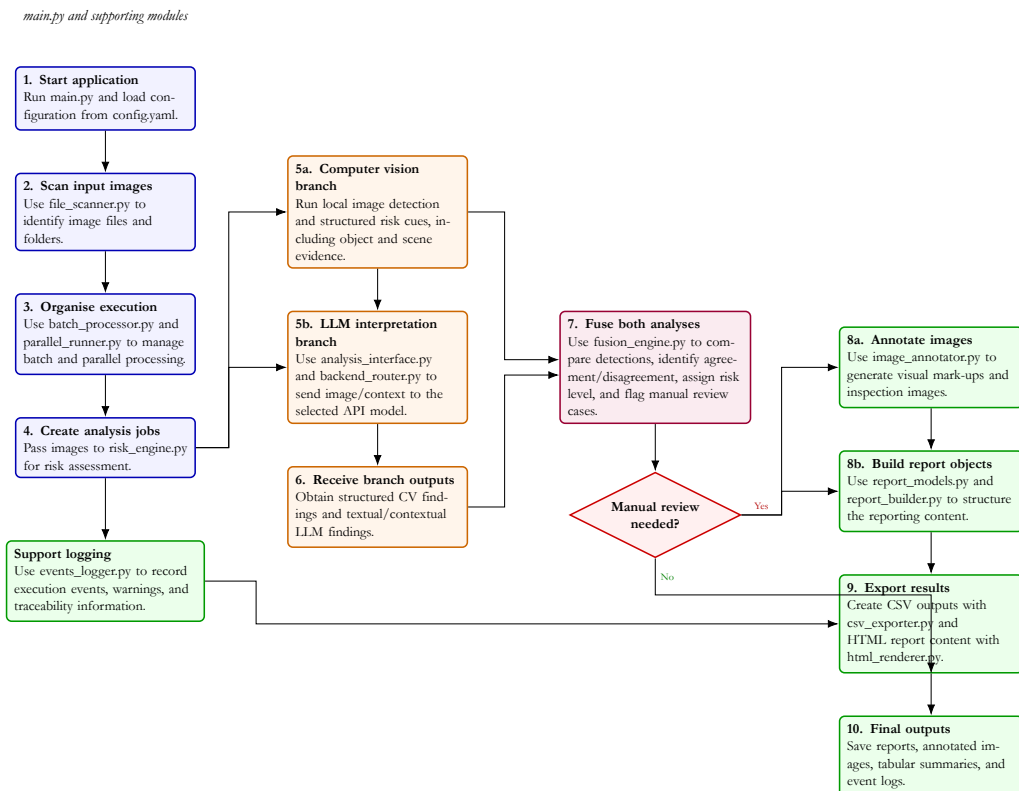


Figure 4.4: Image risk analyser workflow combining computer vision, LLM interpretation, fusion logic, and report generation.

4.9 API Based LLM Assisted Monthly Reporting

The data pipeline developed in this work provides the foundation for the integration of LLM-assisted reporting into construction supervision workflows. Rather than treating the LLM as an autonomous decision-maker, the implemented approach positions it as an assistive synthesis layer operating over structured and traceable project evidence. Its role is to support the organisation, interpretation, and summarisation of information produced by the different components of the methodology, including inspection records, Outlook correspondence, meteorological indicators, financial schedule outputs, contract price revision calculations, dashboard indicators, and image-based risk analysis.

Within the implemented prototype, the LLM-assisted component is designed to support the preparation of monthly supervision reports. The monthly master pipeline scans configured output folders, collects available files, selects recent outputs from each analytical domain, and builds a structured evidence package. This package contains file metadata, selected summaries of readable files, references to generated outputs, and reporting instructions. It is then submitted to Claude through the Anthropic API, which generates a draft monthly report in Markdown format. The report is also converted into HTML for browser-based review.

A later extension of the pipeline incorporates figure-based reporting. In this version, selected visual outputs from Outlook analytics, meteorological dashboards, daily report analyses, financial schedule evolution, and image-risk analysis are copied into the monthly report output folder and inserted into the final report. Where possible, the comments associated with these figures are generated from the structured data supporting the plots, rather than from the image alone. This is particularly relevant for charts such as Outlook activity by hour, day, month, heatmaps, recipient rankings, and meteorological weekly or monthly dashboards. When supporting data are not available, visual interpretation by the LLM is used as a fallback.

This design preserves a clear separation between calculation and interpretation. Quantitative processing, extraction, aggregation, visualisation, and domain-specific analysis remain the responsibility of the Python pipelines created in the Bronze, Silver, and Gold layers. The LLM is used only after these processes have produced auditable outputs. Its role is therefore not to calculate the indicators directly, but to transform collected evidence and selected figures into a coherent technical narrative.

The approach follows an evidence-oriented logic. The LLM is not expected to generate conclusions from general knowledge alone, but to work from the available project data, previously produced analytical outputs, and selected visual artefacts. This is particularly important in construction supervision, where monthly reports may support contractual, financial, safety-related, and managerial decisions. For this reason, the pipeline saves the evidence package as a JSON file before the LLM call. This file provides an auditable record of the information made available to the model and supports later human review of the generated report.

The implemented strategy also builds on the dual-pipeline logic developed for image risk analysis. In that module, structured computer vision detections are combined with contextual scene interpretation produced by an LLM. The resulting fusion process identifies agreements, disagreements, risk levels, and cases that require manual review. In the broader monthly reporting workflow, the same principle is adopted at a higher level: automated interpretation is used to organise evidence, comment on relevant figures, highlight issues, and support prioritisation, while professional validation remains necessary.

From a methodological perspective, this component should be understood as a controlled

reporting assistant rather than a fully autonomous supervision agent. It follows a defined workflow: collect configured outputs, select recent files, summarise readable sources, select relevant figures, build an evidence package, generate a draft synthesis, insert figure commentary, and save the result for human validation. The current implementation does not autonomously decide which tools to run, does not iteratively call external tools, and does not replace the judgement of engineers, supervisors, or decision-makers.

Although the prototype demonstrates the feasibility of LLM-assisted integration across several supervision data sources, full autonomous supervision remains outside the scope of this work. Construction supervision operates in a regulated, contractual, and safety-critical environment where accountability, professional judgement, and legal responsibility remain essential. Consequently, the contribution of this component lies in demonstrating how LLMs can support evidence collection, figure interpretation, report drafting, risk awareness, and decision support while maintaining human-in-the-loop control.

The proposed direction creates a basis for future development of more specialised analytical assistants. These may include progress monitoring assistants, safety review assistants, cost control assistants, correspondence analysis assistants, or risk assessment assistants, each operating over specific curated datasets. However, in the present work, the focus remains on the development of an integrated LLM-assisted evidence synthesis layer capable of bringing together heterogeneous supervision outputs into a coherent and reviewable monthly reporting workflow.

Monthly Master Pipeline – Process Workflow

monthly_master_pipeline.py

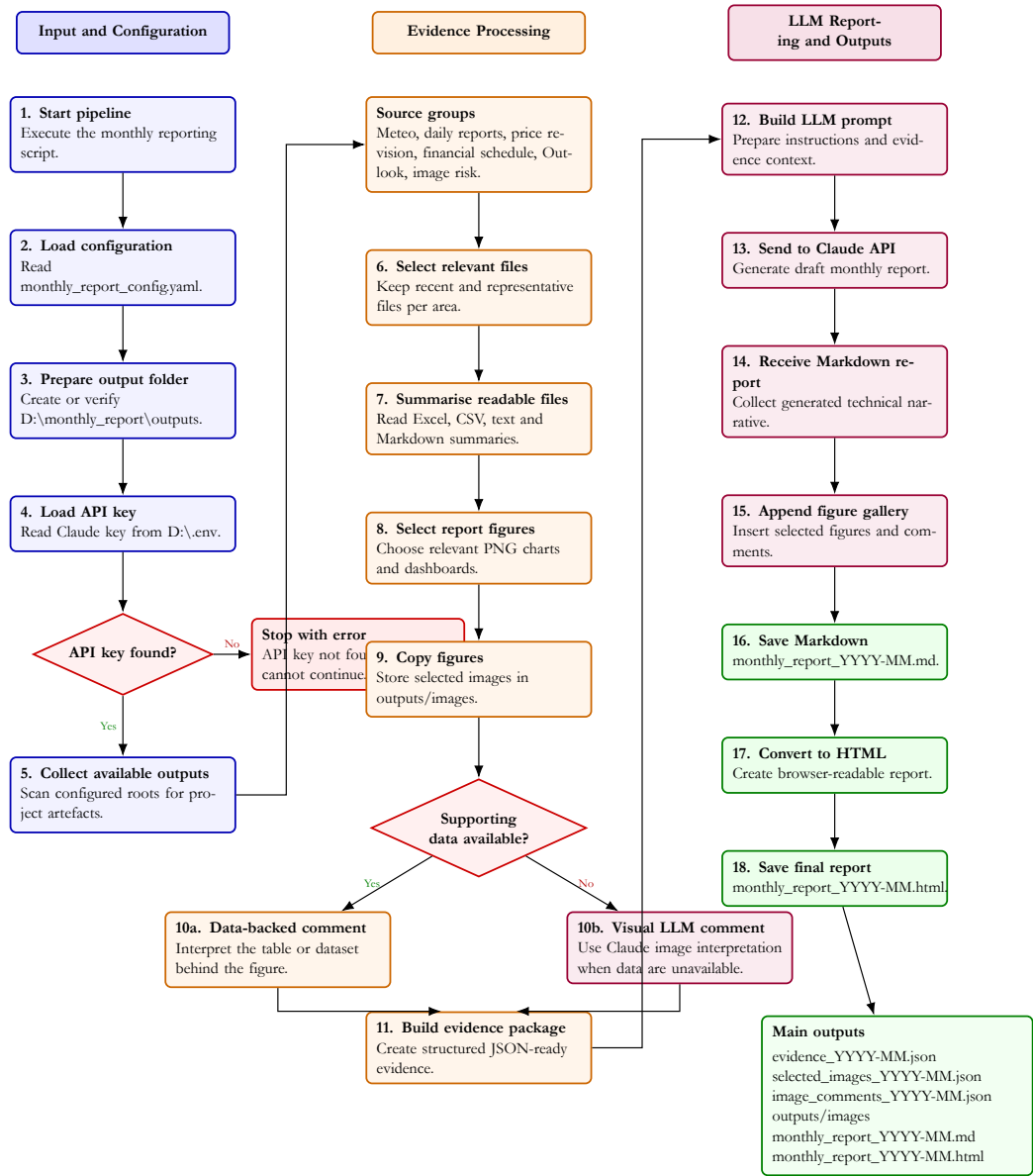


Figure 4.5: Monthly master pipeline workflow with evidence synthesis, figure selection, data-backed commentary, and API-based LLM reporting.

4.10 Overall Integrated Data Engineering and Reporting Workflow

The overall workflow developed in this work can be understood as an integrated data engineering and reporting process for construction supervision. Several domain-specific pipelines were implemented to process heterogeneous sources, including Outlook correspondence, meteorological data, daily inspection reports, price revision files, financial schedule information, and image-based risk analysis. Each pipeline produces structured outputs such as Excel

files, CSV tables, dashboards, charts, reports, logs, or evidence records.

These outputs are then brought together by the monthly master pipeline, which acts as an integration layer above the individual analytical modules. Instead of reprocessing all raw data, this layer scans the configured output folders, selects recent and relevant artefacts, summarises readable files, and builds a structured evidence package. This package is then submitted to an LLM through an external API, allowing the model to generate a monthly supervision report from the available project evidence.

The resulting architecture demonstrates a practical transition from fragmented construction supervision records towards a more structured and semi-automated reporting environment. The deterministic Python scripts remain responsible for extraction, transformation, calculation, classification, and analytical output generation, while the LLM is used as a controlled synthesis layer for narrative reporting. This preserves human-in-the-loop validation and reinforces the role of the system as a decision-support and reporting assistant, rather than an autonomous supervision agent.

Global Process Flow – Analytics Pipelines to LLM Monthly Reporting

Main phases only

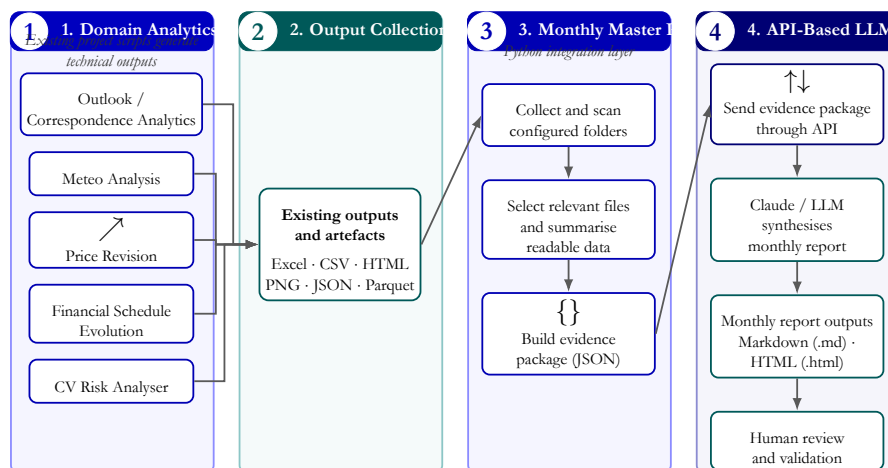


Figure 4.6: Global process flow from domain analytics pipelines to API-based LLM monthly reporting.

As shown in Figure 4.4, the image risk analyser combines computer vision, LLM-based contextual interpretation, fusion logic, and report generation.

The monthly master pipeline represented the main integration step. It collected outputs from the different analytical domains, selected recent files, summarised readable data, built a structured evidence package, and sent that package to Claude through an API. The resulting monthly report was saved in Markdown and HTML. In its extended version, the pipeline also selected relevant visual outputs, copied them into the reporting folder, inserted them into the final report, and generated comments for these figures. Where possible, figure commentary was based on the structured data supporting the plots, reducing the need to send every image directly to the LLM. This demonstrated a practical form of API-based LLM-assisted evidence synthesis, figure interpretation, and monthly reporting.

A significant conclusion is that the LLM layer should be positioned as an assistive reporting and synthesis component, not as an autonomous decision-maker. The implemented

pipeline does not ask the LLM to replace calculations, extract raw project data directly, or make binding decisions. Instead, deterministic Python scripts generate the evidence, figures, charts, and analytical outputs, while the LLM organises this material into a coherent technical narrative. This separation between calculation and interpretation is essential in construction supervision, where financial, contractual, safety, and managerial decisions require accountability and professional judgement.

The evidence package is one of the most important outcomes of the monthly reporting workflow. By saving the information made available to the LLM as a JavaScript Object Notation (JSON) file before the API call, the system creates an auditable record of the basis used to generate the report. This improves transparency and supports human review. The later inclusion of selected figures and figure comments further strengthens the report by connecting the narrative with visual evidence. However, the current implementation does not yet provide full statement-level traceability. The generated report does not automatically cite each conclusion to a specific file, row, chart value, or table cell. This remains an important limitation and a priority for future development.

From a data analysis perspective, the work produced several useful descriptive and diagnostic outputs, including resource evolution charts, planned-versus-actual financial comparisons, weather classifications, correspondence indicators, communication-network tables, theme evolution, weekly anomalies, image risk summaries, and commented report figures. These outputs support supervision analysis and reporting. Nevertheless, the work is not primarily a predictive data science study. It does not train new forecasting models, perform causal inference, or statistically validate predictive algorithms. Its data science elements are exploratory and applied, while the main emphasis remains on data engineering and analytics integration.

Several limitations remain. The architecture depends on local paths, manually prepared templates, file naming conventions, installed Python libraries, external APIs, and the availability of source files. The monthly master pipeline currently collects existing outputs but does not yet orchestrate the automatic execution of all upstream scripts. Although selected Portable Network Graphics (PNG) charts and dashboard figures can now be inserted into the report and commented, the quality of those comments depends on the availability of supporting data or on the capacity of the LLM to interpret the visual content. Some outputs, such as complex HTML dashboards, JSON files, and Parquet logs, are still not deeply interpreted. The freshness and quality of the monthly report depend on the freshness and quality of the upstream outputs.

Another limitation is the absence of a complete validation framework. The accuracy of Outlook extraction, RDF parsing, embedded Excel extraction, meteorological classification, resource comparison, price revision calculation, financial schedule alignment, image risk classification, figure commentary, and LLM report generation was not systematically measured against manually verified reference datasets. Future work should include validation datasets, expected outputs, error metrics, review procedures, and acceptance criteria.

Future development should focus on strengthening the architecture in four main directions. First, the storage layer should be improved to move closer to a true lakehouse, including stronger metadata management, shared schemas, versioning, and possibly transactional table formats. Second, the Medallion methodology should be standardised across all pipelines, with clearer Bronze, Silver, and Gold responsibilities. Third, orchestration should be improved so that upstream scripts can be executed automatically in the correct order, with logs, depend-

ency checks, failure handling, and freshness validation. Fourth, the LLM-assisted reporting layer should be extended with source-level citations, stronger chart-to-data traceability, Word or PDF export, and stronger mechanisms for human validation.

Chapter 5

Results

This chapter presents the results obtained from the implemented prototype. Following the logic of CRISP-DM and the Medallion structure, the results evolve from data understanding and preparation towards analytical outputs, visualisation, interpretation, and reporting. Here, the focus is not on the implementation process, but on the evidence produced by each module and its usefulness for construction supervision.

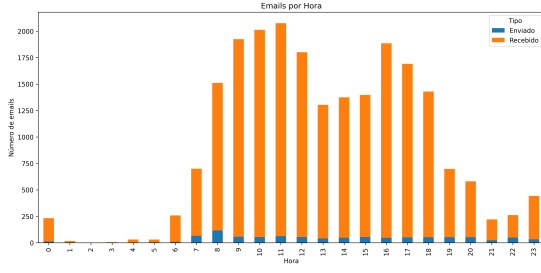
5.1 Outlook Analysis

The Outlook analysis begins with the extraction of structured information from Outlook Classic. The implemented scrape process collects e-mail metadata, sender and recipient information, dates, folders, attachments, and message-body references, storing them in a local SQLite database. This database transforms dispersed correspondence into a queryable source of evidence, enabling subsequent filtering, aggregation, visualisation, and network analysis.

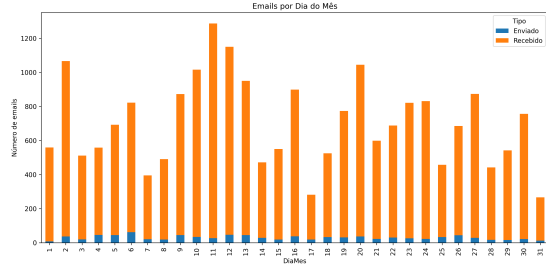
General Outlook correspondence Analysis The general analysis of Outlook correspondence provides a broad view of project communication behaviour. By structuring e-mail metadata, it becomes possible to identify the relative weight of different senders and recipients, observe periods of high communication intensity, and detect moments of reduced interaction. These indicators support the identification of communication peaks, inactive periods, and possible imbalances between project stakeholders.

This type of analysis is useful because many coordination signals remain outside formal reports or individual task records. Concentrated communication may indicate urgent issues, contractual pressure, technical clarification needs, or periods of increased coordination. Conversely, low communication activity may suggest reduced project interaction, lack of follow-up, or phases where relevant information is not being formally exchanged.

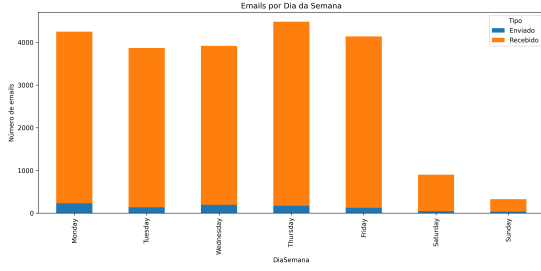
The correspondence analysis therefore acts as a communication monitoring layer. It does not evaluate the technical content of each message in isolation, but provides indicators about communication rhythm, actor participation, and stakeholder activity. This helps reveal patterns that may not be visible through the independent work of individual collaborators and supports a more systematic understanding of project coordination (Figure 5.1).



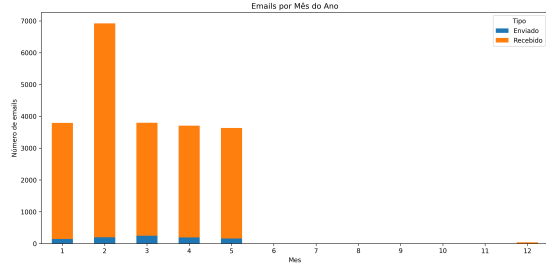
(a) E-mail distribution by hour



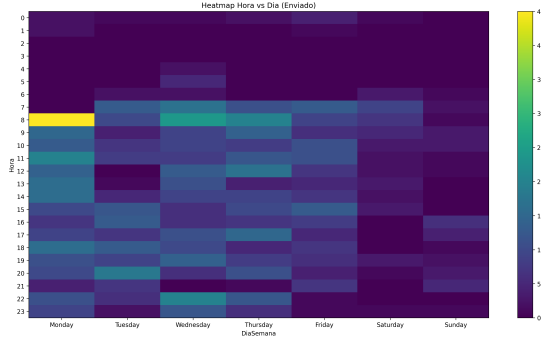
(b) E-mail distribution by day of month



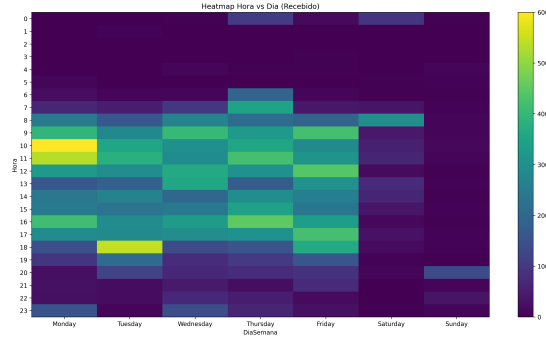
(c) E-mail distribution by weekday



(d) E-mail distribution by month



(e) Sent e-mail heatmap



(f) Received e-mail heatmap

Figure 5.1: General Outlook correspondence analysis outputs.

The Outlook analysis provide also a perspective on the most active stakeholders, both as senders and recipients, as well as on e-mail domains, attachment file types, and recurrent words in message content, in the format of horizontal bar plots. These outputs support a broader understanding of project communication behaviour and help identify the actors, organisations, and document types most present in the correspondence flow.

For privacy protection, the code also supports pseudonymisation, replacing names and e-mail addresses with stable anonymous identifiers while storing the correspondence in a separate hidden dictionary.

It is also possible to locate e-mails related to a specific subject over time using keyword-based filtering, as illustrated in Figure ???. In this example, the keyword “compactação” was used to retrieve the corresponding messages. The resulting scatter plot represents the selected e-mails by date and hour, making it possible to identify the periods in which this topic was more present in communications between project stakeholders.

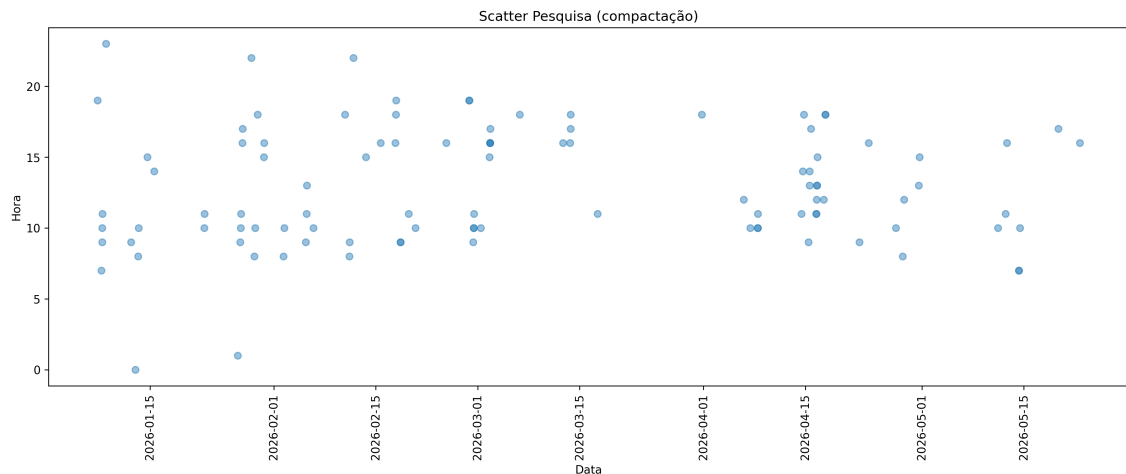
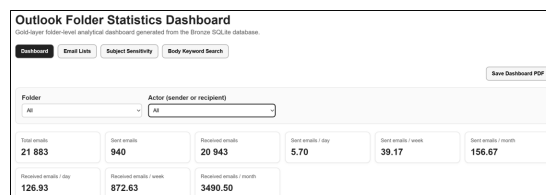


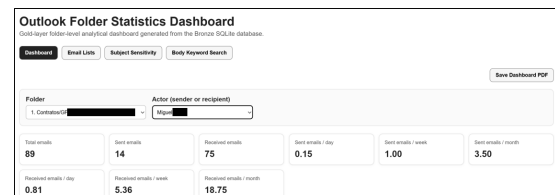
Figure 5.2: Scatter plot date vs hour by keyword

Filtered Outlook correspondence analysis The possibility of results obtained from the Outlook correspondence pipeline is vast, including structured e-mail records include folder statistics, temporal communication patterns, sender and recipient analysis, heatmaps, attachment indicators, and experiments in sensitivity analysis and exploration in network outputs.

Also is interesting to filter information, and analyse it in smaller compartments. The possibility to filter emails of a certain project or sender, is here possible through an HTML interface (Figure 5.3) that produce selected filtered data accordingly to folder and sender (Figure 5.4 and Figure 5.14).



(a) unselected header



(b) Selection folder and sender

Figure 5.3: HTML Outlook email selector.



Figure 5.4: Ouputs of of emails after selection through folder and sender.

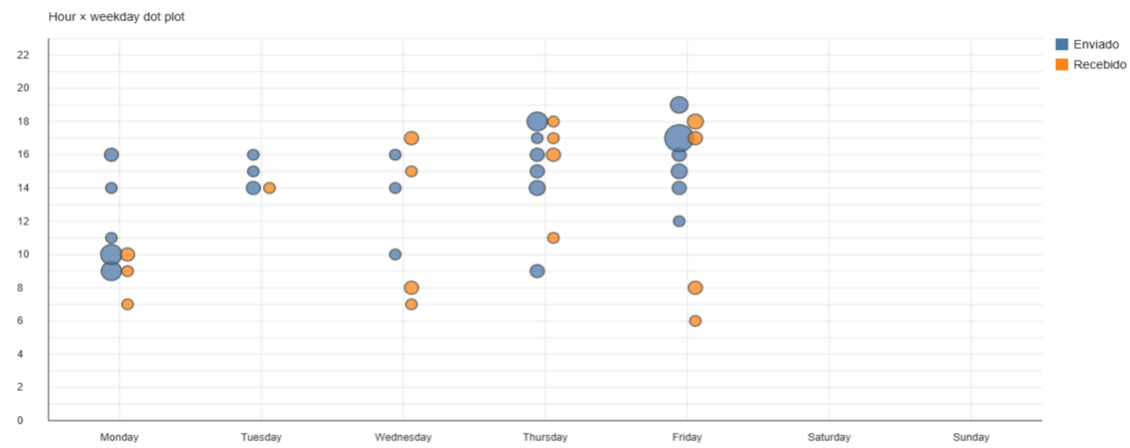


Figure 5.5: Filtered by folder and sender hours vs weekday

The network dashboard provides a visual overview of the main communication flows within the selected folder, highlighting the most active actors, the strongest sender–recipient links, and the relative intensity of email exchanges (Figure 5.6).

Outlook Communication Network Dashboard

Gold-layer network analysis based on sender-recipient relations extracted from Outlook.

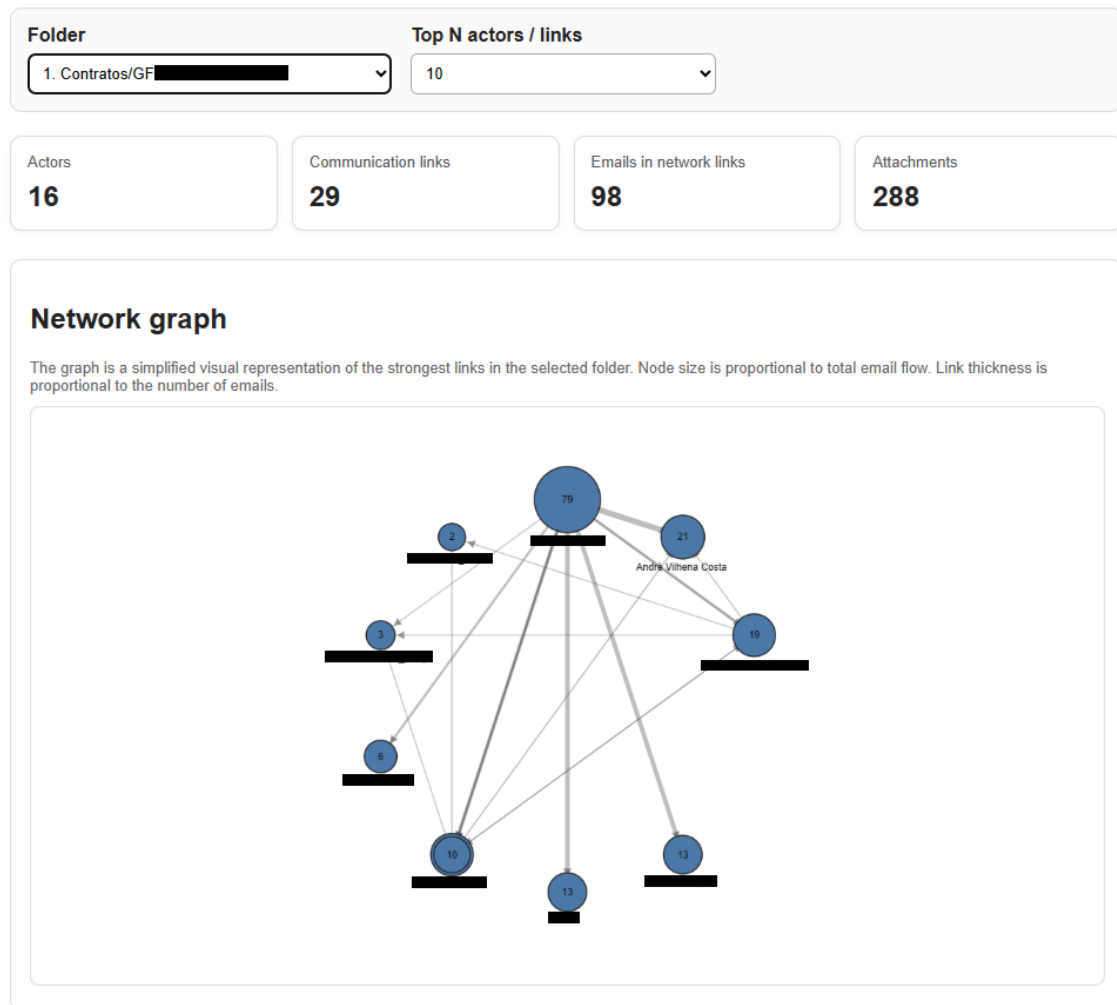


Figure 5.6: Outlook emails network analysis

Result discussion. The Outlook analysis showed that:

1. Outlook allows basic filtering but it does not provide a complete analytical exploration of e-mails, metadata and communication patterns.
2. The implemented pipeline transformed e-mails into a structured dataset, enabling temporal analysis, keyword-based searches, sender and recipient analysis, attachment analysis and network visualisation.
3. E-mail frequency can be used as an indirect indicator of communication intensity, workload concentration or periods of increased coordination between stakeholders.
4. Keyword searches allow specific subjects to be located in time, supporting the identification of when certain topics became relevant in project communications.

5. Network visualisation supports the analysis of interactions between stakeholders, helping identify central actors, stronger communication links and possible coordination patterns.
6. Future development should include a more systematic sensitivity analysis of e-mail content by identifying keywords, subject terms, and filtering rules to select e-mails containing specific information for analysis.

5.2 Meteorological Analysis

The meteorological outputs generated from public IPMA data, include daily classifications, weekly, monthly, quarterly, and yearly visualisations, and indicators supporting the interpretation of weather conditions during the supervision period.

The Silver layer generates graphical reports for four temporal scales: weekly, monthly (Figure 5.7), quarterly and annual. Each report is saved using a standardised file name containing the report type and the start and end dates of the period. While a reporting period is still active, the corresponding file is updated. Once the period ends, that report becomes the final version for the interval, and the next execution of the script begins producing the report for the following period.

Produced reports includes line graphs for temperature, precipitation and wind; comparison lines for the three nearest meteorological stations; pie charts based on temperature, precipitation and wind classification classes; a count of periods or hourly records with and without precipitation; a summary block containing project location, station metadata, project–station distances and primary station indicators; and an inset geographic map showing the project location and the three selected meteorological stations. The report includes the required attribution to OpenStreetMap contributors (OpenStreetMap contributors, 2026).

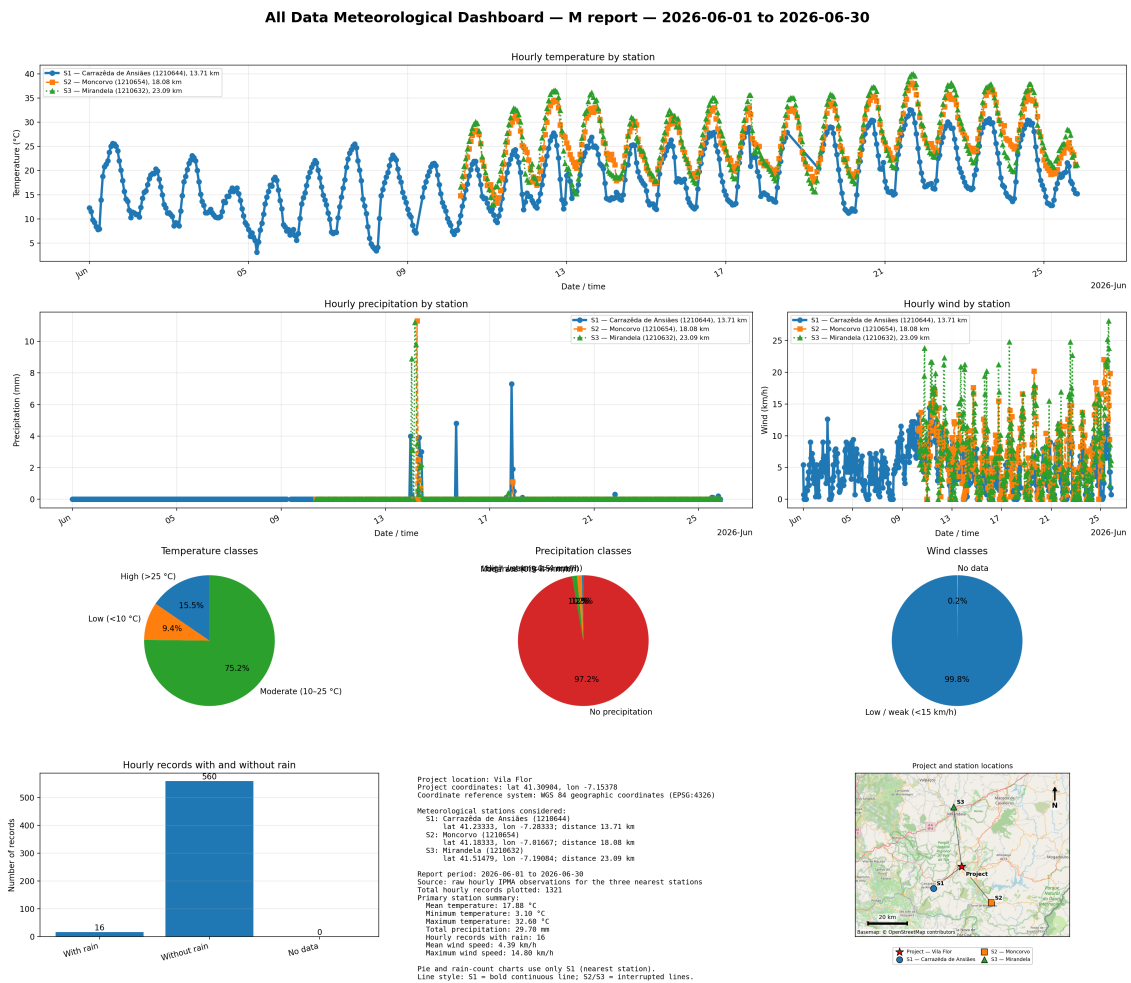


Figure 5.7: Monthly meteorological report based on continuous hourly records.

Result discussion. The meteorological analysis showed that:

1. Climate change is expected to increase the frequency and severity of extreme weather events, with a greater impact on construction activities. This may lead to more frequent legal disputes between clients and contractors regarding risk allocation and responsibility for additional costs, thereby justifying the development of meteorological analysis for this purpose.
2. Meteorological observations have been integrated into the supervision data environment as a structured data source, complementing internal project records with contextual evidence.
3. The IPMA API enabled automatic collection of recent hourly observations, reducing dependence on manual downloads and supporting the creation of a local meteorological archive.
4. The Bronze, Silver and Gold organisation made it possible to preserve raw observations, transform them into structured indicators, and generate curated graphical outputs for reporting.

5. Automatic selection of the three nearest meteorological stations improved the transparency of the analysis.
6. The generated weekly, monthly, quarterly, and annual reports support construction supervision, enabling continuously available reports at different time scales.
7. Simultaneously with this work, a similarly structured meteorological dataset was used to analyse correlations between production deficits and corresponding financial deviations, together with on-site resources deficit (labour and equipment), analysing a specific claim worth several hundred thousand euros.
8. The creation of a local meteorological archive supports the statistical assessment of specific periods, to evaluate whether certain weather conditions were exceptional when compared with the available historical observations.

5.3 Daily Inspection Records

This section presents the results extracted from daily supervision records, including consolidated activity matrices, inspection notes, personnel and equipment records, resource evolution charts, and comparisons between registered and planned resources.

Images represent some of the datasets, relative to several months, what represent many tens or hundreds of individual reports, that can be extracted in seconds (examples of Figure 5.8) and (Figure 5.9). From that data, easily can be compared the contractually previewed resources and those present on site (Figure 5.10).

	Mês ##																															
Categoria de pessoal	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
Administrativo	1								1							1								1	1							
Carpinteiro									2	2	2	2	2			2	2	2	2	2			2	2	2	2	2				2	2
Diretor de Obra	1	1	1	1	1				1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	
Diretor Técnico EE																											1	1				
Eletricidade									1																							
Eletricista																	1															
Encarregado	1	1	1	1	1				1	2	1	1	2			1	1	1	1	1			2	1	1	1	1			2	2	
Encarregado Eletricidade									1																							
Manobrador Mini-giratória											1																					
Motorista Pesados												1																				
Pedreiro	1	1	1	1					1	1	1	1	1			1	1	1		1			3	3	3	3	3			3	4	
Somirav																2																
Topógrafo					1																											
TQAS			1						1							1																

Figure 5.8: Labour force extracted from the daily reports

	Mês ##																														
Tipo de equipamento	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Balde de Betonagem	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Carrinha transporte	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Cilindro Ligeiro												1																			
Contentor Escritório	3	3	3	3	3				3	3	3	3	3			3	3	3	3	3			3	3	3	3	3			3	3
Contentor Pessoal	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Contentor WC	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Fossa séptica estanque	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Gerador	1	1	1	1	1				1	1	1	1	1			2	2	2	2	2			2	2	2	2	2			2	2
Giratória de rastros	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Grua Automontante	1	1	1	1	1				1	1	1	1	1			2	2	2	2	2			2	2	2	2	2			2	2
Máquina de Cortar/Dobrar Ferro	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Separação de Resíduos	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
Separação de Resíduos	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1
WC Químico	1	1	1	1	1				1	1	1	1	1			1	1	1	1	1			1	1	1	1	1			1	1

Figure 5.9: Equipments on site extracted from daily reports

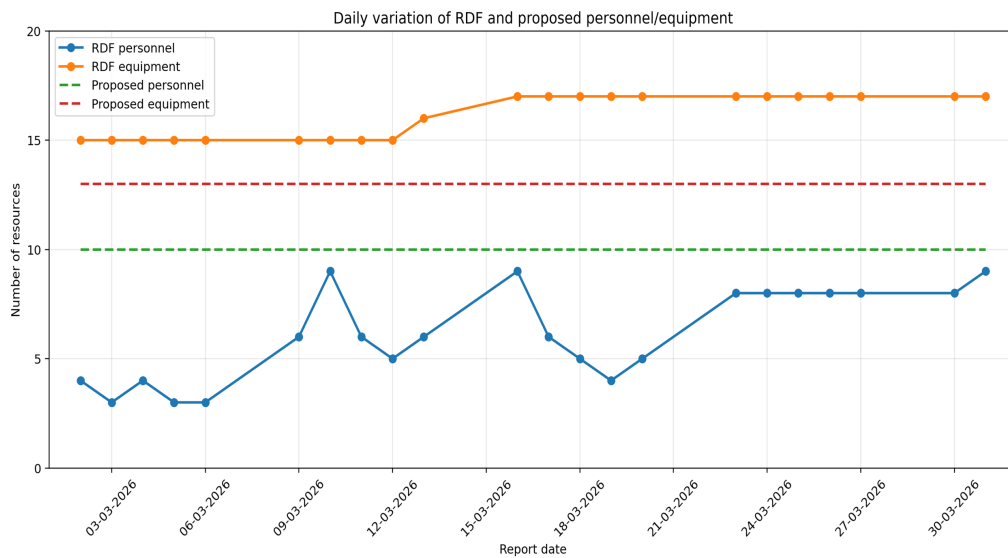


Figure 5.10: Variation of personnel and equipment from daily report

The supervision team can visually detect whether the contractor is consistently mobilising fewer resources than proposed, whether mobilisation peaks correspond to the planned workload, and whether equipment and personnel trends are coherent with the construction phase. That can be highlighted by plotting the differences between the effective load and the planned one (Figure 5.11).

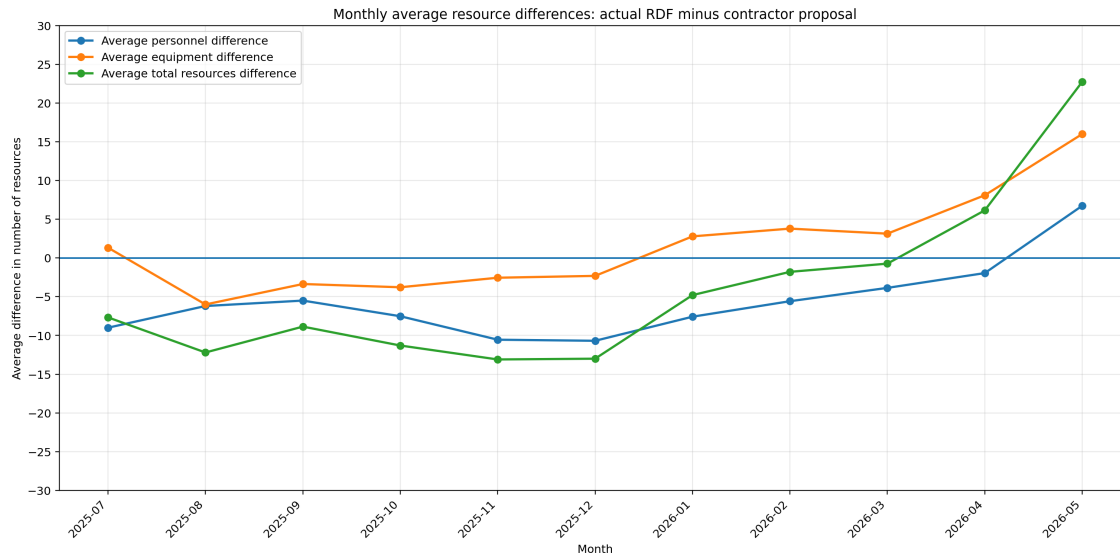


Figure 5.11: Difference between actual daily report quantities and bid-proposed quantities

Figures 5.12 and 5.13 illustrates an example of the keyword-based selector mapping output produced from daily inspection records. The figure represents the temporal organisation of selector occurrences using a heatmap logic, where rows correspond to selectors and columns correspond to time periods. The intensity of colour reflects the number of occurrences detected for each selector in the corresponding temporal interval.

selector	2025-W29	2025-W30	2025-W31	2025-W32	2025-W33	2025-W34	2025-W35	2025-W36	2025-W37	2025-W38	2025-W39	2025-W40	2025-W41	2025-W42	2025-W43	2025-W44	2025-W45	2025-W46	2025-W47	2025-W48	2025-W49	2025-W50	2025-W51	2026-W02	2026-W03	2026-W04	2026-W05	2026-W07	2026-W08	2026-W09	2026-W10	2026-W11	2026-W12	2026-W13	2026-W14	2026-W15	2026-W16	2026-W17	2026-W18	2026-W19	2026-W20	2026-W21	2026-W22
Desmatção	0	0	0	0	0	0	0	0	1	2	0	0	1	0	1	0	2	2	0	0	0	1	2	0	0	0	0	0	1	2	0	0	0	1	0	0	2	0	0	0	1	1	
Escavação	3	2	1	1	2	1	0	0	0	3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	3	1	0	0	4	1	0	0	0	1	6	4	5	5	3	
Entivação	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	1	2	0	0	0	0	
Instalação de conduta	0	0	0	0	1	0	0	0	0	2	2	0	0	0	2	2	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0	1	0	0	0	0	1	0	0	0	0	
montagem de armadura	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	4	2	0	1	0	0	0	0	0	0	0	1	0	5	4	3	3	0	3	2	0	2	
cofragem	0	0	0	0	0	0	0	0	0	1	4	1	0	3	3	1	6	3	0	3	3	1	0	0	3	5	0	1	0	1	1	0	2	3	4	1	5	1	2	2	1	4	6
betonagem	0	0	0	0	0	0	0	0	2	2	0	0	0	0	0	2	3	0	3	0	2	0	0	2	0	0	1	0	2	0	0	1	1	1	1	0	1	4	1	1	1	2	0
descofragem	0	0	0	0	0	0	0	0	0	0	1	0	3	0	0	0	2	1	1	0	0	0	1	0	1	1	4	0	2	1	0	0	0	0	1	1	2	0	0	0	1	0	
conduta	0	0	0	0	1	0	0	0	2	2	0	0	0	2	2	0	0	0	1	0	0	0	1	0	0	0	0	2	0	0	0	1	1	0	0	0	1	0	0	0	0	0	
aterro	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	3	0	0	0	0		
ABGE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Betuminosos	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	1	0	0	0	0	
Compactação	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	1	0	0	0	0		
Ensaio	0	0	0	0	0	0	2	1	1	5	5	5	5	6	2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	0	0	0	1	0	0	0	0	

Figure 5.12: Activities per week

selector	2025-07	2025-08	2025-09	2025-10	2025-11	2025-12	2026-01	2026-02	2026-03	2026-04	2026-05
Desmatção	0	0	3	2	4	3	0	3	1	2	2
Escavação	6	4	3	0	1	0	0	4	5	11	18
Entivação	0	0	0	1	3	0	0	0	0	6	0
Instalação de conduta	0	1	4	4	0	0	0	2	2	1	0
montagem de armadura	0	0	0	0	3	6	1	0	12	10	4
cofragem	0	0	5	8	12	4	8	2	6	13	13
betonagem	0	0	4	5	5	2	4	2	4	6	4
descofragem	0	0	1	3	4	1	6	3	0	5	1
conduta	0	1	4	4	1	1	0	2	2	1	0
aterro	0	0	0	0	0	0	0	0	3	3	0
ABGE	0	0	0	0	0	0	0	0	0	0	0
Betuminosos	0	0	0	0	0	0	0	0	2	1	0
Compactação	0	0	0	0	0	0	0	0	3	1	0
Ensaio	0	2	14	17	1	0	0	0	3	1	0

Figure 5.13: Activities per month

Result discussion. The daily inspection records analysis showed that:

1. Daily inspection reports have been transformed from dispersed document-based records into structured datasets containing activities, labour, equipment and production-related information.
2. The automatic extraction process reduces the time required to consolidate information from multiple reports, making it possible to analyse several months of records, hundreds of individual reports, in a short and systematic way.
3. The generated summaries support the comparison between resources actually registered on site and resources planned or contractually expected, helping identify possible mobilisation gaps.
4. The graphical outputs make resource evolution easier to interpret, allowing the supervision team to observe trends, peaks, reductions and inconsistencies in personnel and equipment over time.
5. The keyword-based heatmaps support the temporal tracking of activities or relevant subjects, helping identify when specific works, events or issues were more frequently reported.
6. These outputs improve the reuse of daily inspection records for monitoring and reporting, providing a historical perspective of the project.
7. Future development should improve validation rules and classification methods, so that extracted activities, resources and quantities can be more consistently compared across different reporting formats and periods.

5.4 Price Revision and Financial Analysis

This work presents a very brief and simple analysis, relative to contract financial aspects, like planned versus actual financial schedule evolution and price revision calculations.

Schedule evolution is presented through a graph comparing planned and actual financial execution. The graph includes four main elements: planned monthly values, actual monthly values, planned cumulative values and actual cumulative values. Monthly values are represented as bars because they correspond to discrete periods. Cumulative values are represented as lines because they express the progressive financial evolution of the contract. This type of representation is consistent with standard plotting practice, where bar charts are used for category-based values and line plots for ordered series (Matplotlib, 2026a, 2026b).

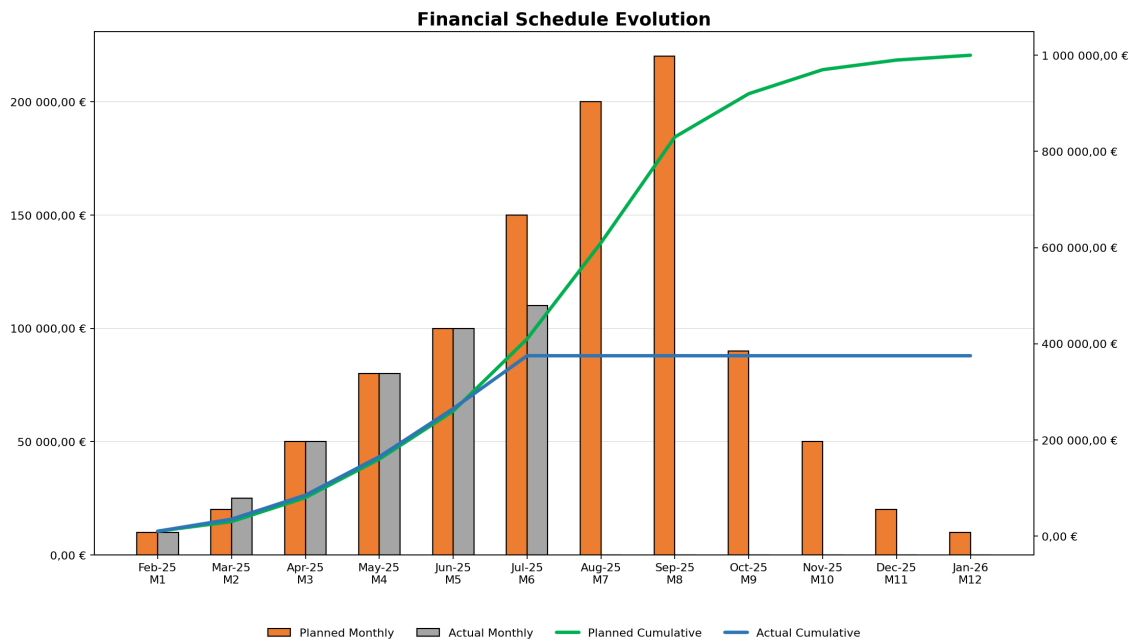


Figure 5.14: Financial schedule evolution graphical representation

The planned monthly bars are shown for all months in the approved financial schedule. The actual monthly bars are shown only in months where a monthly payment exists. The planned cumulative line represents the expected financial progression, while the actual cumulative line represents the certified progression. This structure allows deviations between planned and actual execution to be identified visually.

If the actual cumulative line remains below the planned cumulative line, the contract is financially behind the approved baseline. If the actual monthly bars are consistently lower than the planned monthly bars, production in those periods is below the expected level. Conversely, if the actual cumulative line is above the planned cumulative line, the contract may be progressing faster than planned. Differences between planned and executed, can also be explained (it happens often), when the approved schedule reflects a wrong planification or when no longer represents the actual execution strategy.

The financial schedule evolution is directly related to the price revision process. The same monthly payments that define actual monthly execution are also used to determine the values subject to price revision. The certified monthly value identifies the production to be revised, while the corresponding period identifies the relevant month for the application of cost indices.

Therefore, the financial schedule and price revision workflows are complementary. The financial schedule provides the planned-versus-actual monitoring structure, while the price

revision calculation updates the certified value according to official indices. Together, they support contract management by linking planning, certification, cost variation and financial control.

The developed code contributes to this integration by preserving the history of monthly payment and by using the same historical records for both revision calculation and financial monitoring. This reduces inconsistencies between separate spreadsheets and improves the auditability of the contract management process.

The code workflow of price revision produces a visual output that supports the interpretation of the price revision process over time. Figure 5.15 compares the planned monthly financial schedule with the actual monthly certified values and adds two revision-specific indicators: the monthly price revision and the accumulated price revision. This representation is useful because it links the financial rhythm of the contract with the monetary effect of the revision calculation. The planned and actual monthly bars remain associated with the left vertical axis, while the revision values are represented on the right vertical axis, avoiding the distortion that would occur if all values were plotted on the same scale.

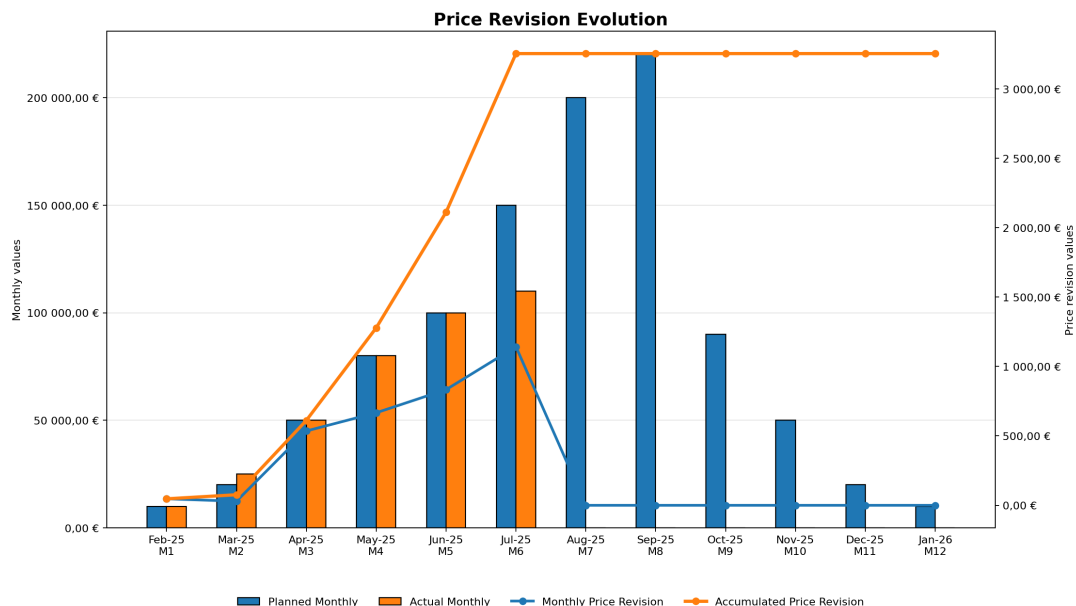


Figure 5.15: Price revision evolution

The figure therefore complements the tabular outputs of the calculation. While the workbook sheets provide the detailed coefficient calculation, formula components and indices used, the chart allows a faster visual reading of how the revision evolves as the autos are processed. In the example shown, the accumulated price revision increases only in the months with certified production and then stabilises when no additional autos are recorded, illustrating the importance of maintaining a historical sequence of payment certificates.

Result discussion. The price revision and financial analysis showed that:

1. Financial schedule data can be transformed into structured analytical outputs that compare planned and actual contract execution over time.

2. The combined representation of monthly and cumulative values makes it easier to identify deviations between the approved financial schedule and the certified execution.
3. The analysis supports contract monitoring by showing whether financial execution is delayed, aligned with, or ahead of the planned baseline.
4. The integration between financial schedule records and price revision calculations improves consistency, because the same certified monthly values can be used for both execution monitoring and revision calculation.
5. The price revision outputs improve the traceability of the calculation process by preserving monthly payment history, applied indices, coefficients and accumulated revision values.
6. Price revision can be time-consuming due to repeated calculations and the continuous search for updated indices, and it may also involve additional costs when supported by specialised software. The solution proposed in this work, mitigated such complexity.
7. The graphical representation of price revision evolution helps interpret the financial impact of cost variation over time, complementing the detailed workbook outputs.
8. These outputs support decision-making and reporting, but their reliability depends on the correctness of the original financial schedule, payment certificates, official indices and contractual assumptions used in the calculation.

5.5 Risk Image Analysis

The results of the image risk analyser, include computer vision detections, LLM based contextual interpretations, fused risk classifications, annotated images, CSV summaries, and HTML reporting outputs.

Seven photographs have been submitted to the image risk analyser pipeline (Figure 5.16), with classifications of risk accordingly to Table 5.1.

The rectangles shown around human figures correspond to bounding boxes generated by the YOLOv8 object-detection model. For each detected person, the model returns a class label, a confidence score, and the coordinates of the bounding box. These coordinates are then used to draw a rectangular overlay on the image, allowing the detected human profiles to be visually inspected and compared with the associated risk interpretation.



Figure 5.16: Examples of construction supervision images used in the AI image risk analysis workflow

Table 5.1: Batch-level results from the AI Image Risk Analyser

Indicator	Result
Total photographs analysed	7
Low-risk images	0
Medium-risk images	1
High-risk images	2
Critical-risk images	4
Images requiring manual review	6
Images with CV–LLM conflicts	0

Table 5.2 summarises the image-level results obtained from the AI Image Risk Analyser, indicating the main visible activity, the assigned risk level, and the dominant safety theme identified for each photograph.

Table 5.2: Image-level safety assessment results

Image	Main visible activity	Risk level	Main identified risk theme
1	Roadside excavation and material	High	Machinery proximity, scattered materials, PPE concerns
2	Deep trench excavation	Critical	Unsupported trench walls and worker exposure
3	Trench excavation with machinery	Critical	Cave-in risk and fall protection deficiencies
4	Worker inside trench with utilities	Critical	Utility strike risk and inadequate trench support
5	Material dumping on roadway	High	Traffic control and worker proximity to vehicles
6	Compaction work inside trench	Critical	Trench cave-in and inadequate protective systems
7	Soil testing device	Medium	Equipment exposure and minor site-safety concerns

A PDF version of the AI Image Risk Analyser report is presented in Appendix J. This report was generated from the HTML output produced by the Python pipeline and compares the results of the YOLOv8 based computer vision branch with the conclusions obtained from the LLM API based image analysis branch.

Result discussion. The risk image analysis showed that:

1. Photographic evidence can be incorporated into the supervision data workflow as an additional source of risk related information.
2. The YOLOv8 based computer vision branch supported the detection of visible human presence in site photographs, providing objective bounding-box outputs that can be inspected visually.
3. The LLM based visual interpretation branch added contextual analysis by describing visible activities, identifying potential safety concerns and assigning risk levels.
4. The fusion of computer vision outputs and LLM-assisted interpretation produced structured summaries, annotated images, CSV files and HTML/PDF reports, making photographic evidence easier to review and reuse.
5. The results demonstrated the potential of AI assisted image analysis for supporting safety observation and risk screening in construction supervision.
6. The method should not be interpreted as a formal safety assessment, because the analysis depends on image quality, visible context, model limitations and the absence of information outside the photograph.
7. The YOLOv8 based computer vision component enables the detection of visible safety risks in site imagery, such as missing personal protective equipment (e.g., helmets), and

provides confidence scores as indicators of detection certainty. However, its interpretation is limited to explicitly observable features. In contrast, the LLM based visual interpretation branch offers richer contextual reasoning about activities and potential safety concerns, but without formal reliability metrics and with a risk of incorrect or inferred (hallucinated) outputs. Overall, YOLOv8 provides structured and measurable detections, while the LLM adds semantic context, making the two approaches complementary.

8. Human validation remains essential, especially for critical classifications, ambiguous images, missing context or cases where safety, contractual or legal consequences may arise.

5.6 API Based LLM Assisted Monthly Reporting

The monthly reporting pipeline produces an evidence package that brings together selected figures, data-backed observations, visual comments, Markdown outputs, an HTML report, and the final construction supervision monthly report.

The *Construction Supervision Monthly Report* represents the final reporting output of the proposed workflow, consolidating structured data from inspection records, financial analysis, meteorological information, correspondence, price revision, and risk-related observations into an integrated monthly document. Because the report is generated with support from an LLM through an API, its interpretation and final validation remain dependent on professional review. This report is presented in Appendix K.

Result discussion. The API based LLM assisted monthly reporting showed that:

1. Structured outputs from different supervision modules can be consolidated into a common evidence package for monthly reporting.
2. The workflow demonstrates how data from correspondence, daily inspection records, meteorology, financial analysis, price revision and image risk analysis can be brought together in a single reporting process.
3. The use of an LLM through an API supported the transformation of fragmented analytical outputs into a more coherent narrative report.
4. The generated Markdown, HTML and report outputs improved the readability and communication of the supervision evidence collected during the month.
5. The evidence package is important because it helps separate source information from generated interpretation, improving traceability and reviewability.
6. The monthly report should be interpreted as an assisted reporting output, ready to human edition and not as an autonomous professional conclusion.
7. Human validation remains necessary to confirm technical accuracy, contractual relevance, risk interpretation and the adequacy of the final report before formal use.

5.7 Response to the Research Objectives

The results presented in this chapter indicate that can be considered the research objectives were achieved. The first objective, related to the design of a traceable data organisation model for heterogeneous construction supervision records, was addressed through the implementation of a Bronze, Silver and Gold structure. This organisation made it possible to preserve original or near-original evidence, process intermediate structured datasets, and generate curated outputs for analysis, visualisation and reporting.

The second objective, related to the development of computational pipelines for extraction, cleaning, transformation and integration, was also addressed. The implemented modules processed different types of supervision data, including Outlook correspondence, daily inspection records, meteorological observations, financial schedules, price revision records and photographic evidence. These pipelines transformed fragmented operational records into structured datasets, graphical outputs, tables, dashboards, reports and evidence packages.

The third objective, related to the analytical use of curated supervision data, was demonstrated through several outputs. The Outlook analysis supported communication monitoring and keyword-based exploration. The meteorological module produced temporal reports based on IPMA data. The daily inspection record pipeline generated resource summaries, activity heatmaps and comparisons between planned and actual resources. The financial and price revision modules supported contract monitoring. The image risk analyser demonstrated the use of computer vision and LLM-assisted interpretation for safety-related photographic evidence. Finally, the monthly reporting pipeline showed how curated outputs can be consolidated into an integrated report, while maintaining the need for professional review and validation.

The research questions were therefore answered through the implemented prototype. Heterogeneous construction supervision records can be organised into a reusable analytical environment by combining a Medallion style data structure with local databases, structured files and curated reporting outputs. Fragmented supervision records can be converted into decision-ready outputs through modular extraction, transformation and integration pipelines. Dashboards, graphical reports, image analysis and LLM assisted reporting can improve the use of supervision data by making information easier to explore, compare and communicate.

However, the results should be interpreted within the scope of a prototype. The system demonstrates methodological feasibility and practical usefulness, but it does not replace professional judgement, formal supervision procedures or contractual validation. The outputs require human interpretation, especially when they are used for technical, financial, contractual or safety-related decisions. Therefore, the dissertation answers the research questions by demonstrating that the proposed data analytics methodology can support construction supervision, while confirming that human oversight and professional accountability remain essential.

Chapter 6

Conclusions

This dissertation addressed the challenge of managing heterogeneous information generated during construction supervision from a data analytics perspective. Supervision records are commonly dispersed across inspection reports, correspondence, schedules, meteorological records, financial data, photographic evidence and contractual documentation. Although these records constitute an important evidence base for monitoring, payment validation, risk identification, contractual interpretation and reporting, their fragmentation reduces traceability and limits their systematic reuse.

The main conclusion of this work is that a lightweight, local and modular data analytics methodology can significantly improve the organisation and practical use of construction supervision information. The prototype demonstrated that fragmented records can be progressively transformed into structured, traceable and reusable analytical outputs when they are organised according to a clear data pipeline logic. The use of Bronze, Silver and Gold layers provided a practical structure for separating raw evidence, processed datasets and curated outputs, creating a more transparent connection between source records and final reports, dashboards or analytical interpretations.

The dissertation also showed that this type of methodology does not require a full enterprise data platform to produce value. The implemented system cannot be classified as a production ready lakehouse, but the Lakehouse inspired and Medallion style organisation proved useful as a guiding structure in this context. The main contribution is therefore methodological and demonstrative: the work shows how data engineering principles can be adapted to a supervision environment where information is heterogeneous, operationally fragmented and often stored in formats not originally designed for analytical reuse.

The implemented modules confirmed that different types of supervision evidence can be transformed into decision-support artefacts. Outlook correspondence was converted into structured communication indicators; meteorological data were collected and summarised through API-based workflows; daily inspection records were transformed into resource and activity outputs; financial schedules and price revision records were organised for contract monitoring; and photographic evidence was explored through computer vision and LLM assisted interpretation. These results show that the proposed methodology can support monitoring, reporting, financial analysis, risk identification and evidence consolidation.

A second important conclusion is that AI-assisted tools are useful but need to be grounded in structured evidence and controlled by human validation. The image risk analyser and the API based monthly reporting workflow demonstrated the potential of computer vision

and LLM assisted synthesis, but they also confirmed the need for professional oversight. In construction supervision, outputs related to safety, payments, delays, contractual interpretation or reporting cannot be treated as automatic conclusions. They must remain reviewable, traceable and subject to technical and professional accountability.

The work has several limitations. The prototype remains dependent on local paths, folder conventions, manually configured scripts, available APIs and the quality of the original records. The Bronze, Silver and Gold structure was not implemented with the same maturity across all data domains, and some workflows are more developed than others. The dissertation focused mainly on data engineering, descriptive analytics, evidence consolidation and assisted reporting. It did not develop predictive machine learning, causal modelling, anomaly detection or automated decision-making. The image analysis and LLM assisted reporting components should therefore be understood as exploratory decision-support tools, not as formal safety assessments or autonomous professional conclusions.

Future work should focus on strengthening the architecture, improving automation and expanding analytical depth. Two major development directions could be pursued. The first is a holistic model, designed from the point of data acquisition onwards. Instead of relying on the extraction of heterogeneous information from daily reports, data could be captured directly through dedicated digital applications (e.g., for daily inspection records), simultaneously generating structured databases for subsequent analysis. Such a model should also be designed to support multiple construction materials (e.g., earthworks, concrete and hot-mix asphalt) and different types of construction contracts, including buildings, roads, drainage and sanitation infrastructure, tunnels and other civil engineering projects. The second direction is an organisational model, tailored to the specific resources, business processes, governance requirements and existing data landscape of a particular company.

The prototype could also evolve towards a more formal lakehouse, data warehouse or cloud based environment, with stronger metadata management, orchestration, schema control, access control, versioning, logging and automated data quality checks. However, any full-scale implementation should be carefully planned and aligned with the organisation's data governance policies. In particular, decisions regarding cloud storage, data retention and the periodic use of AI services should be balanced against operational costs, ensuring the investment is justified by improvements in productivity, decision-making and quality management.

Additional supervision domains can be integrated, including non-conformities, inspection and test records, approvals, design changes, claims, environmental records, health and safety documentation, and quality control records. Further developments could explore predictive models for delay forecasting, financial deviation detection, activity classification, risk anticipation and anomaly detection. In the case of LLM assisted reporting, future versions should improve retrieval mechanisms, link generated statements to specific source files or structured records, and clearly distinguish factual evidence from model-generated interpretation.

In conclusion, this dissertation demonstrates that construction supervision can benefit from a structured data analytics foundation inspired by Lakehouse and Medallion principles. The proposed prototype does not represent a final or complete system, but it shows that fragmented operational evidence can be progressively organised, processed, analysed and converted into useful reporting and decision-support outputs. Its main value lies in demonstrating the feasibility of combining data engineering, domain-specific analytics, computer vision, API based LLM synthesis and human validation within a practical construction supervision context.

Bibliography

- AACE International. (2006). Recommended practice no. 49r-06: Identifying the critical path [Computer software manual]. Retrieved from <https://web.aacei.org/>
- AACE International. (2021). Six elements of project controls: The basics [Computer software manual]. Retrieved from <https://source.aacei.org/2021/08/31/part-1-six-elements-of-project-controls-the-basics/>
- Aggarwal, C. C. (2015). *Data mining: The textbook*. Springer International Publishing. doi: 10.1007/978-3-319-14142-8
- Aggarwal, C. C., & Zhai, C. (Eds.). (2012). *Mining text data*. Springer.
- Alsofiani, M. A. (2024). Digitalization in Infrastructure Construction Projects: A PRISMA-Based Review of Benefits and Obstacles. *arXiv preprint arXiv:2405.16875*. Retrieved from <https://arxiv.org/abs/2405.16875>
- Anthropic. (2026a). *Build with claude: Claude api documentation*. Online documentation. Retrieved 2026-06-04, from <https://platform.claude.com/docs/en/home>
- Anthropic. (2026b). *Claude models overview*. Online documentation. Retrieved 2026-06-04, from <https://platform.claude.com/docs/en/about-claude/models/overview> (Accessed online)
- Anthropic. (2026c). *Models overview*. Claude API Documentation. (Accessed: 2026-06-10)
- Anthropic. (2026d). *Vision*. Claude API Documentation. (Accessed: 2026-06-10)
- Antunes, P. A. L. D. (2012). *Desvios de prazos e de custos na execução de empreitadas de obras públicas* (Tech. Rep.). Universidade Lusófona de Humanidades e Tecnologias Departamento de Engenharia Civil. Retrieved from <https://recil.ulusofona.pt/server/api/core/bitstreams/1af74162-2f40-4e1b-8ecc-55be92573400/content>
- Apache Software Foundation. (2026). *Apache Tika*. <https://tika.apache.org/>. (Accessed: 19 June 2026)
- Armbrust, M., Das, T., Sun, L., Yavuz, B., Zhu, S., Murthy, M., ... Zaharia, M. (2020). Delta lake: High-performance ACID table storage over cloud object stores. *Proceedings of the VLDB Endowment*, 13(12), 3411–3424.
- Armbrust, M., et al. (2021). Lakehouse: A new generation of open platforms that unify data warehousing and advanced analytics. In *Cidr conference*.
- Asian Development Bank. (2023). Guidance note on price adjustment in procurement [Computer software manual]. Retrieved from <https://www.adb.org/>
- Azure Databricks. (2026). *What is the medallion lakehouse architecture?* Retrieved from <https://learn.microsoft.com/en-us/azure/databricks/lakehouse/medallion> (Accessed 2026-06-14)
- Azzabi, S., et al. (2024). Data lakes: A survey of concepts and architectures. *Computers*, 13(7), 183.
- Barlybayev, A., et al. (2024). Personal protective equipment detection using YOLOv8. *Cogent*

Engineering.

- Batini, C., Cappiello, C., Francalanci, C., & Maurino, A. (2009). Methodologies for data quality assessment and improvement. *ACM Computing Surveys*, 41(3), 1–52.
- Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., ... others (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*. Retrieved from <https://arxiv.org/abs/2108.07258>
- Camelot contributors. (2026). *Camelot: PDF Table Extraction for Humans*. <https://camelot-py.readthedocs.io/>. (Accessed: 19 June 2026)
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). Crisp-dm 1.0: Step-by-step data mining guide [Computer software manual]. <https://public.dhe.ibm.com/software/analytics/spss/documentation/modeler/14.2/es/CRISP-DM.pdf>. (Accessed: 2026-05-24)
- Da Silva, S. (2025). Digital integration in construction: A case study on common data environment deployment in a major infrastructure project. *Infrastructures*, 10(10), 266. doi: 10.3390/infrastructures10100266
- Databricks. (2024). *What is medallion architecture?* Retrieved from <https://www.databricks.com/blog/what-is-medallion-architecture> (Accessed 2026-06-14)
- Databricks. (2025). *What is the medallion lakehouse architecture?* Online documentation. (Accessed online)
- Databricks. (2026a). *Medallion architecture*. Retrieved from <https://www.databricks.com/glossary/medallion-architecture> (Accessed 27 May 2026)
- Databricks. (2026b). *What is Extract, Load, Transform (ELT)?* <https://www.databricks.com/blog/what-is-elt>. (Accessed: 19 June 2026)
- Databricks. (2026c). *What is the medallion lakehouse architecture?* <https://docs.databricks.com/aws/en/lakehouse/medallion>. (Accessed: 2026-05-24)
- de Vilhena Costa, A., Coelho, I., & Correia, A. G. (2024). Data analysis in geotechnics – an approach to the construction of an urban tunnel. In *Geotechnical engineering challenges to meet current and emerging needs of society at ecmge24* (pp. 3487–3490). CRC Press. doi: 10.1201/9781003431749-689
- Diário da República. (2004a). Decreto-lei n.º 6/2004, de 6 de janeiro: Regime de revisão de preços [Computer software manual]. Retrieved from <https://diariodarepublica.pt/dr/detalhe/decreto-lei/6-2004-240800>
- Diário da República. (2004b). Despacho n.º 1592/2004, de 23 de janeiro: Fórmulas tipo de revisão de preços f01–f14 [Computer software manual]. Retrieved from <https://diariodarepublica.pt/dr/detalhe/despacho/1592-2004-2861331>
- Diário da República. (2004c). Despacho n.º 22637/2004, de 5 de novembro: Fórmulas tipo de revisão de preços f15–f23 [Computer software manual]. Retrieved from <https://diariodarepublica.pt/dr/detalhe/despacho/22637-2004-1869377>
- Diário da República. (2004d). Retificação n.º 383/2004, de 25 de fevereiro [Computer software manual]. Retrieved from <https://diariodarepublica.pt/>
- Ecma International. (2017). *The json data interchange syntax*. Standard ECMA-404, 2nd Edition. Retrieved from <https://www.ecma-international.org/publications-and-standards/standards/ecma-404/>
- EU BIM Task Group. (2017). *Handbook for the introduction of building information modelling by the european public sector*. European Union BIM Task Group. (Available at: <https://eubim.eu/handbook-selection/>)

- European Data Protection Board. (2025). *Guidelines 01/2025 on pseudonymisation*. Retrieved from https://www.edpb.europa.eu/system/files/2025-01/edpb_guidelines_202501_pseudonymisation_en.pdf (Accessed 2026-06-14)
- European Parliament and Council of the European Union. (2024). *Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence*. Official Journal of the European Union.
- European Union. (2016). *Regulation (eu) 2016/679, article 4: Definitions*. Retrieved from <https://gdpr-info.eu/art-4-gdpr/> (Accessed 2026-06-14)
- Fan, W., Ding, Y., Ning, L., Wang, S., Li, H., Yin, D., ... Li, Q. (2024). A survey on rag meeting llms: Towards retrieval-augmented large language models. *arXiv preprint arXiv:2405.06211*. doi: 10.48550/arXiv.2405.06211
- Fang, W., Love, P. E. D., Luo, H., & Ding, L. (2020). Computer vision for behaviour-based safety in construction: A review and future directions. *Advanced Engineering Informatics*, 43, 100980.
- Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures* (Doctoral dissertation, University of California, Irvine). Retrieved from <https://ics.uci.edu/~fielding/pubs/dissertation/top.htm>
- Fielding, R. T., Nottingham, M., & Reschke, J. (2022). *Http semantics*. RFC 9110. Retrieved from <https://www.rfc-editor.org/rfc/rfc9110>
- Fleming, Q. W., & Koppelman, J. M. (2016). *Earned value project management* (4th ed.). Project Management Institute.
- Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... Wang, H. (2024). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*. doi: 10.48550/arXiv.2312.10997
- Ghosh, S., Mukherjee, N., & McCuen, T. (2024). Use of visual dashboards in construction projects. In T. Leathem, W. Collins, & A. Perrenoud (Eds.), *Proceedings of 60th annual associated schools of construction international conference* (Vol. 5, p. 486-494). EasyChair. Retrieved from /publications/paper/PffL doi: 10.29007/wxks
- Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In *Proceedings of the 7th python in science conference* (pp. 11–15).
- Han, J., Kamber, M., & Pei, J. (2011). *Data mining: Concepts and techniques* (3rd ed.). Morgan Kaufmann.
- Harby, A. A., & Zulkernine, F. (2025). Data lakehouse: A survey and experimental study. *Information Systems*, 127, 102460.
- IMPIC. (2026a). Publicação em diário da república: Índices de revisão de preços em empreitadas [Computer software manual]. Retrieved from <https://www.impic.pt/impic/pt-pt/indices-de-revisao-de-precos-em-empreitadas/publicacao-em-diario-da-republica>
- IMPIC. (2026b). Índices de custos de mão-de-obra, de materiais e de equipamentos de apoio [Computer software manual]. Retrieved from <https://www.impic.pt/impic/pt-pt/indices-de-revisao-de-precos-em-empreitadas/indices-de-custos-de-mao-de-obra-de-materiais-e-de-equipamentos-de-apoio>
- International Organization for Standardization. (2019). *Iso 19650: Building information modelling*. ISO. (Information management using building information modelling)
- International Organization for Standardization. (2020). *Iso 21502:2020: Project, programme and portfolio management – guidance on project management* [Computer software manual]. Retrieved from <https://www.iso.org/standard/74947.html>

- International Organization for Standardization. (2023a). *ISO/IEC 23894:2023 Information technology – Artificial intelligence – Guidance on risk management*. International Organization for Standardization.
- International Organization for Standardization. (2023b). *ISO/IEC 42001:2023 Information technology – Artificial intelligence – Management system*. International Organization for Standardization.
- International Organization for Standardization. (2023c). *ISO/IEC 5338:2023 Information technology – Artificial intelligence – AI system life cycle processes*. International Organization for Standardization.
- International Organization for Standardization. (2024a). *ISO/IEC 5259-3:2024 Artificial intelligence – Data quality for analytics and machine learning – Part 3: Data quality management requirements and guidelines*. International Organization for Standardization.
- International Organization for Standardization. (2024b). *ISO/IEC 5259-4:2024 Artificial intelligence – Data quality for analytics and machine learning – Part 4: Data quality process framework*. International Organization for Standardization.
- IPMA. (2026a). *Classificação da intensidade da precipitação líquida*. Retrieved from https://www.ipma.pt/pt/educativa/faq/meteorologia/previsao/faqdetail.html?f=%2Fpt%2Feducativa%2Ffaq%2Fmeteorologia%2Fprevisao%2Ffaq_0033.html (Accessed 27 May 2026)
- IPMA. (2026b). *Classificação do vento*. Retrieved from https://www.ipma.pt/pt/educativa/faq/meteorologia/previsao/faq_0035.html (Accessed 27 May 2026)
- IPMA. (2026c). *Open data: observation, meteorology, stations*. Retrieved from <https://api.ipma.pt/open-data/observation/meteorology/stations/> (Accessed 27 May 2026)
- IPMA. (2026d). *Public open-data api*. Retrieved from <https://api.ipma.pt/> (Accessed 27 May 2026)
- ISO/IEC. (2008). *Iso/iec 25012: Software engineering – software product quality requirements and evaluation – data quality model*. (International Standard)
- Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., ... Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12), 1–38.
- JSON Schema. (2026). *Json schema documentation*. <https://json-schema.org/>. (Accessed 2026)
- Kavaliauskas, P., Fernandez, J. B., McGuinness, K., & Jurelionis, A. (2022). Automation of construction progress monitoring by integrating 3d point cloud data with an ifc-based bim model. *Buildings*, 12(10), 1754. Retrieved from <https://www.mdpi.com/2075-5309/12/10/1754> doi: 10.3390/buildings12101754
- Kim, D., & Xiong, S. (2025). Enhancing worker safety: Real-time automated detection of personal protective equipment to prevent falls from heights at construction sites using improved yolov8 and edge devices. *Journal of Construction Engineering and Management*, 151(1), 04024187. doi: 10.1061/JCEMD4.COENG-14985
- Kim, G., Hong, T., Yim, M., Nam, J., Park, J., Yim, J., ... Park, S. (2022). OCR-Free Document Understanding Transformer. In *European conference on computer vision*. Springer. doi: 10.1007/978-3-031-19815-1_29
- Kimball, R., & Ross, M. (2013). *The data warehouse toolkit* (3rd ed.). Wiley.
- Kleppmann, M. (2017). *Designing data-intensive applications*. O'Reilly Media.
- Kobayashi, V. B., Mol, S. T., Berkers, H. A., Kismihók, G., & Den Hartog, D. N. (2018). Text mining in organizational research. *Organizational Research Methods*, 21(3), 733–765.
- Koeleman, J., Ribeiro, M. J., Rockhill, D., Sjödin, E., & Strube, G. (2019). *Decoding digital*

- transformation in construction*. <https://www.mckinsey.com/capabilities/operations/our-insights/decoding-digital-transformation-in-construction>. (Accessed: 19 June 2026)
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521, 436–444.
- Lee, D., Wentling, T., Haines, S., & Babu, P. (2025). *Delta lake: The definitive guide: Modern data lakehouse architectures with data lakes*. Sebastopol, CA: O'Reilly Media. (First edition)
- Lee, J., et al. (2023). Construction site safety management: A computer vision and deep learning approach. *Applied Sciences*.
- Lenzerini, M. (2002). Data integration: A theoretical perspective. In *Proceedings of the twenty-first acm sigmod-sigact-sigart symposium on principles of database systems* (pp. 233–246). ACM.
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in neural information processing systems*.
- Liu, W., Meng, X., Xing, Z., Zhou, S., & Wang, X. (2021). Applications of computer vision in monitoring the unsafe behavior of construction workers: Current status and challenges. *Buildings*, 11(9), 409.
- Liu, X., Du, R., Tan, L., Xu, J., Chen, C., Jiang, H., & Aldwais, S. (2024). CIB-SE-YOLOv8: Optimized YOLOv8 for real-time safety equipment detection on construction sites. *arXiv preprint arXiv:2410.20699*.
- Lourenço, M. P., Arantes, A., & Costa, A. A. (2025). Barriers to building information modeling (bim) implementation in late-adopting eu countries: The case of portugal. *Buildings*, 15(10), 1651. doi: 10.3390/buildings15101651
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge: Cambridge University Press. Retrieved from <https://nlp.stanford.edu/IR-book/>
- Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Flach, P., Hernández-Orallo, J., Kull, M., ... Ramírez-Quintana, M. J. (2017). Casp-dm: Context aware standard process for data mining. *arXiv preprint arXiv:1709.09003*. Retrieved from <https://arxiv.org/abs/1709.09003>
- Matplotlib. (2026a). matplotlib.pyplot.bar documentation [Computer software manual]. Retrieved from https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.bar.html
- Matplotlib. (2026b). matplotlib.pyplot.plot documentation [Computer software manual]. Retrieved from https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.plot.html
- Microsoft. (2024a). *Mailitem.attachments property (outlook)*. <https://learn.microsoft.com/en-us/office/vba/api/outlook.mailitem.attachments>. (Accessed: 2026-05-24)
- Microsoft. (2024b). *Mailitem object (outlook)*. <https://learn.microsoft.com/en-us/office/vba/api/outlook.mailitem>. (Accessed: 2026-05-24)
- Microsoft. (2024c). *Mailitem.recipients property (outlook)*. <https://learn.microsoft.com/en-us/office/vba/api/outlook.mailitem.recipients>. (Accessed: 2026-05-24)
- Microsoft. (2025). *List attachments: Microsoft graph rest api*. Microsoft Learn. (Accessed online)
- Microsoft. (2026). *Microsoft graph mail api documentation*. Online documentation.
- Microsoft Support. (2026). *Save disk space with onedrive files on-demand for windows*. Retrieved from <https://support.microsoft.com/en-us/office/save-disk-space-with-onedrive-files-on-demand-for-windows-0e6860d3-d9f3-4971-b321-7092438fb38e> (Accessed 2026-06-14)
- Moreira, J., de Leon Ferreira, Carvalho, A. C. P., & Horváth, T. (2019). *A general introduction to data analytics*. John Wiley & Sons, Inc.

- Nakanishi, Y., Kaneta, T., & Nishino, S. (2022). A review of monitoring construction equipment in support of construction project management. *Frontiers in Built Environment*, 7, 632593. Retrieved from <https://www.frontiersin.org/journals/built-environment/articles/10.3389/fbuil.2021.632593/full> doi: 10.3389/fbuil.2021.632593
- Nath, N. D., Behzadan, A. H., & Paal, S. G. (2020). Deep learning for site safety: Real-time detection of personal protective equipment. *Automation in Construction*, 112, 103085.
- National Institute of Standards and Technology. (2023a). *Artificial intelligence risk management framework 1.0*. National Institute of Standards and Technology. doi: 10.6028/NIST.AI.100-1
- National Institute of Standards and Technology. (2023b). *Artificial intelligence risk management framework (ai rmf 1.0)*. Retrieved from <https://www.nist.gov/itl/ai-risk-management-framework>
- National Institute of Standards and Technology. (2025). *Nist privacy framework*. Retrieved from <https://www.nist.gov/privacy-framework> (Accessed 2026-06-14)
- Newman, M. (2010). *Networks: An introduction*. Oxford: Oxford University Press.
- Oesterreich, T. D., & Teuteberg, F. (2016). Understanding the implications of digitisation and automation in the context of industry 4.0: A triangulation approach and elements of a research agenda for the construction industry. *Computers in Industry*, 83, 121–139. doi: 10.1016/j.compind.2016.09.006
- Olanipekun, A. O., & Sutrisna, M. (2021). Facilitating digital transformation in construction: A systematic review of the current state of the art. *Frontiers in Built Environment*, 7, 660758. doi: 10.3389/fbuil.2021.660758
- openpyxl. (2026). openpyxl documentation [Computer software manual]. Retrieved from <https://openpyxl.readthedocs.io/>
- openpyxl contributors. (2026). *openpyxl: A Python library to read/write Excel files*. <https://openpyxl.readthedocs.io/>. (Accessed: 19 June 2026)
- OpenStreetMap contributors. (2026). *OpenStreetMap*. <https://www.openstreetmap.org/>. (Map data available under the Open Database License. Accessed: 2026-06-11)
- Pal, A., Lin, J. J., Hsieh, S.-H., & Golparvar-Fard, M. (2023). Automated vision-based construction progress monitoring in built environment through digital twin. *Developments in the Built Environment*, 16, 100247. Retrieved from <https://www.sciencedirect.com/science/article/pii/S2666165923001291> doi: 10.1016/j.dibe.2023.100247
- pandas. (2026). pandas.read_excel documentation [Computer software manual]. Retrieved from https://pandas.pydata.org/docs/reference/api/pandas.read_excel.html (Accessed documentation for the Python library used to read Excel files into DataFrames)
- pandas development team. (2026a). *pandas.dataframe.to_excel*. https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_excel.html. (Accessed: 2026-05-24)
- pandas development team. (2026b). *pandas.excelwriter*. <https://pandas.pydata.org/docs/reference/api/pandas.ExcelWriter.html>. (Accessed: 2026-05-24)
- Plaat, A., et al. (2025). Agentic large language models: A survey. *arXiv preprint arXiv:2503.23037*.
- Presidência do Conselho de Ministros. (2026). *Resolução do conselho de ministros n.º 89/2026, de 21 de maio*. Diário da República, Série I, n.º 98/2026. (Aprova a Estratégia Nacional para a Implementação da Metodologia BIM, designada PortugalBIM)
- Proaño-Narváez, M., et al. (2022). Earned value management in construction projects: A systematic review. *Buildings*, 12(3). (Review of earned value management applications

- in construction project control)
- Project Management Institute. (2004). How to make earned value work on your project [Computer software manual]. Retrieved from <https://www.pmi.org/learning/library/make-earned-value-work-project-6001>
- Project Management Institute. (2006). Fundamentals of scheduling and resource leveling [Computer software manual]. Retrieved from <https://www.pmi.org/learning/library/scheduling-resource-leveling-project-progression-8006>
- Project Management Institute. (2019). The standard for earned value management [Computer software manual]. Retrieved from <https://www.pmi.org/standards/earned-value-management>
- Pydantic. (2026). *Pydantic documentation*. Online documentation. (Accessed online)
- python-docx contributors. (2026). *python-docx documentation*. <https://python-docx.readthedocs.io/>. (Accessed: 19 June 2026)
- python-openxml. (2026). python-docx documentation [Computer software manual]. Retrieved from <https://python-docx.readthedocs.io/> (Accessed documentation for the Python library used to read Word documents)
- Python Software Foundation. (2025). *shutil — high-level file operations*. Retrieved from <https://docs.python.org/3/library/shutil.html> (Accessed: 2026-07-02)
- Python Software Foundation. (2026a). *hashlib — secure hashes and message digests*. Retrieved from <https://docs.python.org/3/library/hashlib.html> (Accessed 2026-06-14)
- Python Software Foundation. (2026b). *pathlib — object-oriented filesystem paths*. Retrieved from <https://docs.python.org/3/library/pathlib.html> (Accessed 2026-06-14)
- Python Software Foundation. (2026c). *shutil — high-level file operations*. Retrieved from <https://docs.python.org/3/library/shutil.html> (Accessed 2026-06-14)
- Python Software Foundation. (2026d). *sqlite3 — db-api 2.0 interface for sqlite databases*. <https://docs.python.org/3/library/sqlite3.html>. (Accessed: 2026-05-24)
- Python Software Foundation. (2026e). *zipfile — work with zip archives* [Computer software manual]. Retrieved from <https://docs.python.org/3/library/zipfile.html> (Python documentation)
- Radman, K., Jelodar, M. B., Lovreglio, R., Ghazizadeh, E., & Wilkinson, S. (2022). Digital technologies and data-driven delay management process for construction projects. *Frontiers in Built Environment*, 8, 1029586. doi: 10.3389/fbuil.2022.1029586
- Rawte, V., Sheth, A., & Das, A. (2023). A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779–788).
- Reis, J., & Housley, M. (2022). *Fundamentals of data engineering*. O'Reilly Media.
- República Portuguesa. (2023). *Portaria n.º 255/2023, de 7 de agosto*. Diário da República, Série I, n.º 152/2023. (Aprova o conteúdo obrigatório do projeto de execução, bem como os procedimentos e normas a adotar na elaboração e faseamento de projetos de obras públicas)
- República Portuguesa. (2024). *Decreto-lei n.º 10/2024, de 8 de janeiro*. Diário da República, Série I. (Procede à reforma e simplificação dos licenciamentos no âmbito do urbanismo, ordenamento do território e indústria)
- Sawadogo, P., & Darmont, J. (2021). On data lake architectures and metadata management.

- Journal of Intelligent Information Systems*.
- Sharda, R., Delen, D., & Turban, E. (2018). *Business intelligence, analytics, and data science: A managerial perspective* (4th ed.). Pearson.
- Sherafat, B., Ahn, C. R., Akhavian, R., Behzadan, A. H., Golparvar-Fard, M., Kim, H., ... Azar, E. (2020). Automated methods for activity recognition of construction workers and equipment: State-of-the-art review. *Journal of Construction Engineering and Management*, 146(6), 03120002. doi: 10.1061/(ASCE)CO.1943-7862.0001843
- Sinnott, R. W. (1984). Virtues of the haversine. *Sky and Telescope*, 68(2), 159.
- SQLite Consortium. (2025a). *Sqlite fts5 extension*. <https://sqlite.org/fts5.html>. (Accessed: 2026-05-24)
- SQLite Consortium. (2025b). *Sqlite home page*. <https://sqlite.org/index.html>. (Accessed: 2026-05-24)
- Statistics Canada. (2022). Building construction price indexes: Technical guide [Computer software manual]. Retrieved from <https://www.statcan.gc.ca/> (Technical reference on construction price indices and cost variation measurement)
- Studer, S., Bui, T. B., Drescher, C., Hanuschkin, A., Winkler, L., Peters, S., & Müller, K.-R. (2021). Towards CRISP-ML(Q): A machine learning process model with quality assurance methodology. *Machine Learning and Knowledge Extraction*, 3(2), 392–413.
- Tesseract OCR contributors. (2026). *Tesseract OCR*. <https://github.com/tesseract-ocr/tesseract>. (Accessed: 19 June 2026)
- The pandas development team. (2026). *pandas.read_excel*. https://pandas.pydata.org/docs/reference/api/pandas.read_excel.html. (Accessed: 19 June 2026)
- Tukey, J. W. (1977). *Exploratory data analysis*. Reading, Massachusetts: Addison-Wesley.
- Ultralytics. (2026). *Explore ultralytics yolov8*. Ultralytics Documentation. (Accessed: 2026-06-10)
- Vassena, G. P. M., Perfetti, L., Comai, S., Mastrolembo Ventura, S., & Ciribini, A. L. C. (2023). Construction progress monitoring through the integration of 4d bim and surveying methods. *Buildings*, 13(10), 2488. Retrieved from <https://www.mdpi.com/2075-5309/13/10/2488> doi: 10.3390/buildings13102488
- Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., ... Wen, J.-R. (2024). A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6), 186345. doi: 10.1007/s11704-024-40231-1
- Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4), 5–33.
- Ware, C. (2012). *Information visualization: Perception for design* (3rd ed.). Waltham: Morgan Kaufmann.
- Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications*. Cambridge: Cambridge University Press.
- Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., ... Mons, B. (2016). The fair guiding principles for scientific data management and stewardship. *Scientific Data*, 3, 160018. Retrieved from <https://www.nature.com/articles/sdata201618> doi: 10.1038/sdata.2016.18
- Xiao, B., Chen, Z., & Golparvar-Fard, M. (2024). Automated daily report generation from construction videos using chatgpt and computer vision. *Automation in Construction*, 168, 105825. Retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0926580524006101> doi: 10.1016/j.autcon.2024.105825

- Ye, J., Hu, A., Xu, H., Ye, Q., Yan, M., Dan, Y., ... Huang, F. (2023). mPLUG-DocOwl: Modularized Multimodal Large Language Model for Document Understanding. *arXiv preprint arXiv:2307.02499*. Retrieved from <https://arxiv.org/abs/2307.02499>
- Ye, J., Hu, A., Xu, H., Ye, Q., Yan, M., Xu, G., ... Huang, F. (2023). UReader: Universal OCR-free Visually-situated Language Understanding with Multimodal Large Language Model. *arXiv preprint arXiv:2310.05126*. Retrieved from <https://arxiv.org/abs/2310.05126>

Appendix A

Haversine formula of earth surface distance

The nearest meteorological station is selected by calculating the great-circle distance between the project location and each available IPMA meteorological station. Since both the project location and the stations are represented by geographic coordinates, expressed as latitude and longitude in the WGS 84 coordinate reference system (EPSG:4326), the distance was estimated using the Haversine formula (Sinnott, 1984). This formula accounts for the curvature of the Earth and is therefore more appropriate than a simple planar distance calculation for comparing geographic points. The distance d , in kilometres, between two points is given by:

$$d = 2R \arcsin \left(\sqrt{\sin^2 \left(\frac{\Delta\varphi}{2} \right) + \cos(\varphi_1) \cos(\varphi_2) \sin^2 \left(\frac{\Delta\lambda}{2} \right)} \right) \quad (\text{A.1})$$

where R is the mean radius of the Earth, assumed as 6371.0088 km, φ_1 and φ_2 are the latitudes of the project location and the meteorological station, λ_1 and λ_2 are their longitudes, $\Delta\varphi = \varphi_2 - \varphi_1$, and $\Delta\lambda = \lambda_2 - \lambda_1$. All angular values are expressed in radians.

This way, it is possible to select the station, or the stations, with the smallest calculated distance, used as the reference meteorological station(s) for the project.

Appendix B

Bronze Layer Source Synchronisation

B.1 Overview of the Bronze Synchronisation Pipeline

The Bronze synchronisation workflow is a preliminary data ingestion and repository refresh process. Serves to update the controlled Bronze repository with the most recent available raw documentation from the operational source repository, before any structured extraction, transformation or analytical processing is performed.

The synchronisation process is intentionally conservative. It updates only projects that already exist in the Bronze repository. Projects that exist only in the operational source folder are ignored and recorded in a report. This avoids the uncontrolled creation of new project folders and helps preserve the defined analytical scope of the repository.

B.2 Script

D:\BRONZE\Bronze Synchronisation\update_bronze.py

B.3 Purpose

The script does not transform the content of the files. It does not parse meeting minutes, reports or financial documents. Only control file synchronisation, ensuring that the Bronze repository contains the most recent available raw documents for the projects already included in the repository.

This separation is important because operational folders may be frequently modified, reorganised or synchronised by users or cloud storage systems.

B.4 Main Inputs

- Operational source repository: C:\Users\geond\ApplusGlobal
- Controlled Bronze repository: D:\BRONZE
- Existing project folders in the Bronze repository following the defined project-folder naming convention.

B.5 Main Outputs

- Synchronisation report: sync_report_YYYYMMDD_HHMMSS_dry_run.csv
- Synchronisation report generated after applying the update: sync_report_YYYYMMDD_HHMMSS_apply.csv
- Ignored source projects report: ignored_source_projects_YYYYMMDD_HHMMSS.csv
- Duplicate target project keys report: duplicate_target_project_keys_YYYYMMDD_HHMMSS.csv

The reports are stored in:

D:\BRONZE\Bronze Synchronisation\sync_reports

B.6 Project Matching Logic

Projects are matched between the operational source repository and the Bronze repository using the first twelve characters of the project folder name. This prefix is treated as a stable project key.

This rule was adopted because full project folder names may vary between repositories due to descriptive additions, naming differences or synchronisation conventions, while the initial project identifier tends to remain more stable.

The matching logic follows three principles:

1. Search for project folders in the operational source repository.
2. Search for project folders already existing in the Bronze repository.
3. Match source and target project folders using the first twelve characters of the folder name.

If a project exists in the operational source but no corresponding key exists in the Bronze repository, the project is ignored. This prevents the uncontrolled creation of new projects in the Bronze layer.

B.7 File Update Logic

For each matched project, the script recursively scans all source files and builds the equivalent target path inside the Bronze repository. The update decision is made file by file.

The script applies the following rules:

- if the file exists in the operational source but not in the Bronze repository, it is classified as `copy_new`;
- if the file exists in both locations and the source file is newer than the Bronze version, it is classified as a possible update;
- if the source file is newer and the content is different, it is classified as `copy_updated`;
- if the source file is newer but the content is identical, it is skipped;
- if the Bronze file is equal or newer, the source file is skipped;
- if the file is temporary, hidden, system-like or inaccessible, it is skipped and recorded as an error or skipped item.

The script does not delete files from the Bronze repository. This is consistent with the role of the Bronze layer as a raw evidence repository, where the objective is to preserve source material and maintain traceability rather than perform aggressive cleaning or deletion.

B.8 Dry-Run and Apply Modes

The script is designed to run first in dry-run mode. In this mode, no files are copied. The script only generates a report describing what would happen if the update were applied.

The dry-run command is:

```
python "D:\BRONZE\Bronze Synchronisation\update_bronze.py"
```

After reviewing the generated report, the update can be applied with:

```
python "D:\BRONZE\Bronze Synchronisation\update_bronze.py" --apply
```

This two-step procedure reduces the risk of accidental overwriting and supports auditability. It allows the user to inspect the number of new files, updated files, skipped files, ignored projects and possible permission errors before modifying the Bronze repository.

B.9 Main Processing Logic

1. Define the operational source folder and the controlled Bronze repository.
2. Define the script folder and the synchronisation reports folder.
3. Search the operational source repository for project folders following the defined naming convention.
4. Search the Bronze repository for existing project folders following the same naming convention.
5. Build project keys using the first twelve characters of the folder names.
6. Match source projects to existing Bronze projects using the project key.
7. Ignore source projects that do not have a corresponding Bronze project.
8. Detect duplicate project keys in the Bronze repository and report them.
9. Recursively scan files inside each matched source project.

10. Ignore temporary, hidden, system-like and inaccessible files.
11. For each source file, calculate the equivalent target path in the Bronze repository.
12. Check whether the file already exists in the Bronze repository.
13. Compare modification dates between source and target files.
14. When necessary, compare file size and SHA-256 hash to avoid unnecessary overwriting.
15. Classify each file as `copy_new`, `copy_updated`, `skip` or `skip_error`.
16. In dry-run mode, only report the intended actions.
17. In apply mode, copy the new or updated files to the Bronze repository.
18. Write synchronisation reports documenting the action and reason for each analysed file.

B.10 Report Fields

The main synchronisation report includes the following fields:

- `mode`;
- `project_key`;
- `source_project`;
- `target_project`;
- `relative_path`;
- `source_file`;
- `target_file`;
- `target_exists`;
- `source_modified`;
- `target_modified`;
- `source_size_bytes`;
- `target_size_bytes`;
- `action`;
- `reason`;
- `copied`.

The most relevant action values are:

- `copy_new` – the file exists in the source but not in the Bronze repository;
- `copy_updated` – the source file is newer and different from the Bronze version;
- `skip` – the file does not require copying;
- `skip_error` – the file could not be processed due to access, permission or synchronisation issues.

B.11 Libraries Used

- `argparse` – command-line interface and execution mode selection.
- `csv` – creation of synchronisation reports.
- `hashlib` – SHA-256 content hashing for file comparison Python Software Foundation (2026a).
- `shutil` – high-level file-copy operations, including metadata-preserving copying where supported by the operating system Python Software Foundation (2026c).
- `dataclasses` – structured representation of matched projects.
- `datetime` – timestamp creation and modification-date reporting.
- `pathlib.Path` – object-oriented and platform-aware file-system path handling Python Software Foundation (2026b).

B.12 Strategies Used

- **Bronze-first control:** only projects already existing in the Bronze repository are updated.
- **Stable key matching:** project folders are matched using the initial characters of the folder name.
- **Recursive scanning:** all files within matched source project folders are inspected.
- **Date-based update detection:** source files are copied only when they are missing or newer than the target version.
- **Hash-based verification:** file content is compared to avoid unnecessary overwriting when files are identical despite timestamp differences.

- **Metadata-preserving copy:** copied files preserve metadata as far as supported by the operating system.
- **Dry-run safety:** the first execution produces a report without copying files.
- **Apply mode:** the same logic is used to perform the actual update after the dry-run report has been reviewed.
- **Permission resilience:** inaccessible files are skipped and reported instead of stopping the whole process.
- **Audit reporting:** every file-level decision is recorded with an action and reason.

B.13 Data Governance and Pseudo-Anonymisation Relevance

The Bronze synchronisation stage preserves the original project documentation in its raw form. These source documents may include project identifiers, personal names, organisation names, financial information and contractual data. At the Bronze level, this preservation is necessary because the raw files act as evidence and as the source for subsequent extraction routines.

However, later analytical outputs do not always require direct exposure of these identifiers. For this reason, subsequent stages may apply pseudo-anonymisation, replacing project, person and organisation identifiers with stable pseudo-identifiers. Pseudo-anonymisation is recognised in data-protection frameworks as a technique where data cannot be attributed to a specific individual without additional information, provided that this additional information is kept separately and protected by appropriate measures European Data Protection Board (2025); European Union (2016).

This approach does not make the data fully anonymous, but it reduces unnecessary exposure of identifiable information in intermediate datasets, dashboards and reports. It also allows the same entity to be followed consistently across multiple outputs, supporting analytical continuity while contributing to privacy-risk management National Institute of Standards and Technology (2025).

B.14 Architectural Characterisation

The Bronze synchronisation process is a source-refresh and ingestion-control stage. It does not perform analytical transformation, text extraction or dashboard creation. Its main function is to ensure that the Bronze repository is up to date, controlled and traceable before downstream processing begins.

In the wider medallion workflow, this stage can be characterised as follows:

- **Operational source:** working project folders maintained outside the controlled analytical repository.
- **Bronze synchronisation:** controlled update of existing Bronze project folders with missing or newer raw files.
- **Bronze repository:** stable raw-document layer used by downstream extraction routines.
- **Silver extraction:** structured extraction from refreshed raw documents into databases and tabular outputs.
- **Gold analysis:** analytical reporting, dashboards and management indicators based on curated datasets.

The strongest architectural characteristic of this process is the separation between operational file storage and analytical processing. The synchronisation script creates a controlled raw-document basis for later extraction and analysis, while preserving traceability through detailed file-level reports.

B.15 Recommended Interpretation

The script should be interpreted as a Bronze-layer preparation tool. It improves the reliability of downstream processing by ensuring that the controlled raw repository contains the most recent available files for the projects already included in the analytical scope.

The dry-run report should be reviewed before applying the update, especially when large numbers of new files are detected or when the operational source is managed by a cloud-synchronisation system. After the update is applied, the generated reports should be kept as evidence of the synchronisation operation.

This process supports the overall methodological objective of the work: to build a reproducible, traceable and privacy-aware computational workflow for construction supervision documentation, based

on a medallion-inspired separation between raw data, structured data and analytical outputs.

Appendix C

Outlook extraction code

C.1 Overview of the outlookskrape Pipeline

The codebase is organised as a three-stage Outlook e-mail extraction and analysis workflow. The overall architectural logic is:

1. **Phase 1 – Bronze extraction:** extract Outlook Classic data into a portable SQLite database and companion body-text files.
2. **Phase 2 – direct exploratory analysis:** scan Outlook directly and generate exploratory Excel tables and figures.
3. **Phase 3+4+5 – Gold analytics:** consume the Bronze SQLite database and derived CSV files to produce folder statistics, investigative analytics, reports, and HTML dashboards.

The most important design distinction is that Phase 1 and Phase 2 depend on Outlook Classic through COM automation, whereas the Gold layer is designed to work without reconnecting to Outlook. After the Bronze extraction exists, the later analytical scripts read the SQLite database, body-text files, or CSV outputs.

C.2 Phase 1: Bronze Extraction to SQLite

C.2.1 Script

D:\BRONZE\outlookskrape\Outlook_skrape_sql\01_extract_outlook_to_sqlite.py

C.2.2 Purpose

This script extracts structured information from Outlook Classic and persists it into a portable project directory. Its main outputs are:

- a SQLite database containing e-mail metadata, folders, recipients, attachments, extraction runs, and error logs;
- plain-text body files for e-mail content;
- optional HTML body files;
- CSV exports for external inspection;
- a full-text-search table for later content search.

C.2.3 Main Processing Logic

1. Configure paths, output folders, database path, and execution flags.
2. Create the required output directories.
3. Initialise Outlook COM with `pythoncom.CoInitialize()`.
4. Connect to SQLite and enable database pragmas such as foreign keys and WAL mode.
5. Create or update the database schema.
6. Register a new extraction run.
7. Connect to Outlook Classic through `win32com.client.Dispatch("Outlook.Application").GetNamespace("MAPI")`.
8. Recursively traverse all Outlook root folders and subfolders.
9. Ignore non-mail Outlook items using the Outlook MailItem class identifier.

10. Extract e-mail metadata, sender information, recipients, attachments, dates, folder paths, sizes, and flags.
11. Save message body content to text files and optionally save HTML bodies.
12. Compute body hashes using SHA-256.
13. Insert or update records in SQLite using stable Outlook identifiers such as EntryID.
14. Log errors with traceback information.
15. Finish the extraction run, export CSV snapshots, and create a manifest.

C.2.4 Database-Oriented Strategy

The Bronze database acts as the durable analytical base. The schema separates core entities into relational tables:

- `extraction_runs` – run metadata and execution status;
- `folders` – Outlook folder paths and folder metadata;
- `emails` – one row per e-mail, including subject, sender, dates, folder, size, flags, and body-file references;
- `recipients` – recipient records linked to e-mails;
- `attachments` – attachment metadata linked to e-mails;
- `errors` – error records with stage, subject, entry identifier, and traceback;
- `full-text-search` table – optional searchable message-body content.

This normalised structure makes the output portable and allows later scripts to analyse e-mails without Outlook.

C.2.5 Resources Used

Libraries

- `os` – operating-system path and environment utilities.
- `re` – regular expressions for cleaning text and filenames.
- `csv` – CSV snapshot exports.
- `json` – manifest or structured metadata output.
- `time` – elapsed-time measurement.
- `hashlib` – SHA-256 hashing of message body text.
- `sqlite3` – local relational database storage.
- `traceback` – detailed exception logging.
- `pathlib.Path` – path construction and directory management.
- `datetime` – timestamp handling and extraction-run metadata.
- `pythoncom` – COM initialisation and cleanup.
- `win32com.client` – Outlook Classic automation through MAPI.

Strategies

- **Bronze-layer persistence:** raw and semi-structured Outlook data are preserved before higher-level analysis.
- **Recursive traversal:** all root folders and subfolders are scanned.
- **Defensive extraction:** helper functions such as safe string, integer, float, date, and property access reduce failures caused by missing Outlook fields.
- **Idempotent database update:** e-mails are inserted or updated using unique Outlook identifiers.
- **Separation of metadata and body text:** metadata stays in SQLite, while large bodies are stored as external text files.
- **Attachment classification:** inline attachments are detected using content ID, hidden flags, MIME type, and attachment method.
- **Full-text search readiness:** message bodies can be loaded into an FTS table for later search.
- **Auditability:** extraction runs, counters, errors, logs, CSV snapshots, and manifests support traceability.

In Figure C.1 the entity relation diagram of the sql database is presented, including all tables, attributes, and keys.

Entity-Relationship Diagram (ERD) — Outlook Extraction Database

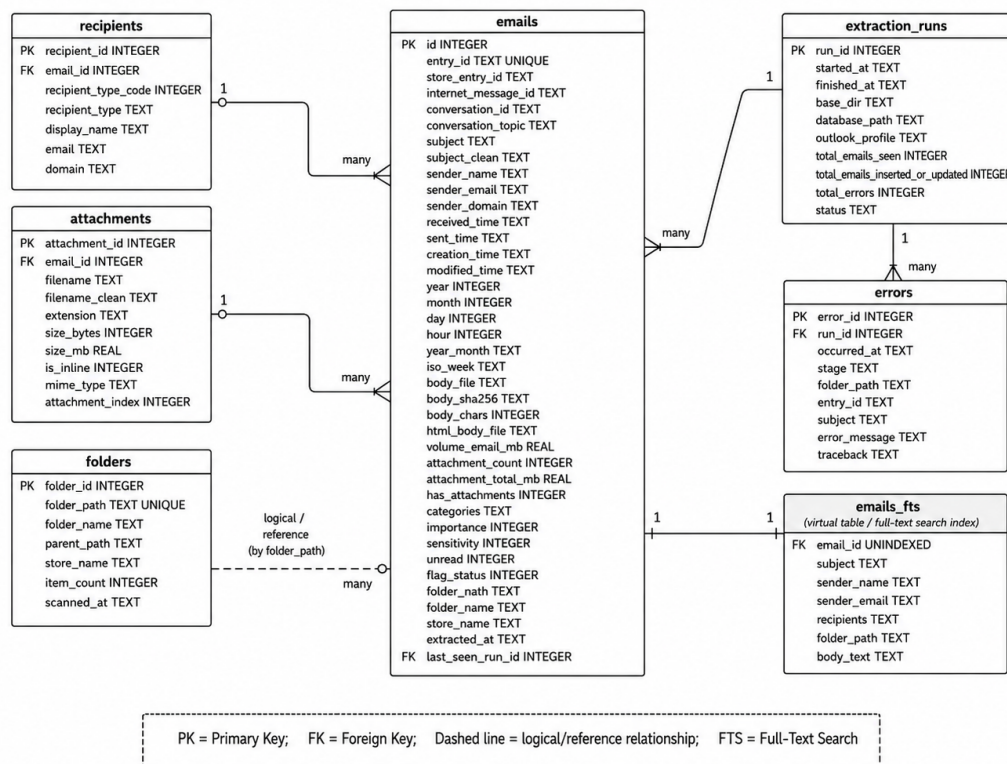


Figure C.1: Entity relationship diagram - Outlook extraction database

C.3 Phase 2: Direct Outlook Exploratory Analysis

C.3.1 Script

D:\BRONZE\outlookscrape\outlook.py

C.3.2 Purpose

This script performs a direct exploratory analysis of Outlook data. Unlike the Gold scripts, it connects directly to Outlook, scans the mailbox, creates a pandas dataframe, classifies messages as sent or received, generates charts, and exports an Excel workbook.

It is useful for fast exploratory analysis, but it is less portable than the Bronze/Gold architecture because it depends on live Outlook access.

C.3.3 Main Processing Logic

1. Initialise COM and connect to Outlook through the MAPI namespace.
2. Ask the user to enter one or more own sender names separated by semicolons.
3. Recursively scan Outlook folders.
4. Ignore non-mail items.
5. Collect raw fields such as received time, sender name, recipients, subject, attachment count, and folder path.
6. Build a pandas dataframe.
7. Convert date fields and derive temporal attributes such as date, hour, year-month, week, week-day, day of month, and month.
8. Classify each e-mail dynamically as Enviado or Recebido.
9. Filter unwanted sender patterns, while also preserving an unfiltered copy for subfolder analysis.
10. Create summary tables by sender, recipient, folder, subfolder, type, and interaction.

11. Generate visualisations such as scatter plots, stacked bars, top senders, top recipients, KPI trends, heatmaps, and weekly views.
12. Optionally run keyword search over sender, recipient, and subject fields.
13. Export results to emails_base.xlsx.
14. Save chart images to the figuras folder.

C.3.4 Resources Used

Libraries

- pythoncom – COM initialisation.
- win32com.client – Outlook automation.
- pandas – dataframe construction, cleaning, aggregation, and Excel export.
- matplotlib.pyplot – chart generation.
- seaborn – heatmap visualisation.
- numpy – numerical support.
- datetime – date and time handling.
- os – figure-folder creation and file output paths.

Strategies

- **Interactive classification:** the user provides sender identifiers that determine whether e-mails are treated as sent or received.
- **Recursive scan:** all available Outlook folders are traversed.
- **Temporal feature engineering:** raw received timestamps are transformed into analysis-friendly variables.
- **Exploratory aggregation:** pandas groupby, value_counts, and pivot tables are used to summarise activity.
- **Visual analytics:** charts provide quick inspection of time patterns, peaks, interactions, and communication volume.
- **Excel reporting:** multiple analytical tables are exported to separate workbook sheets.
- **Keyword search:** optional text filtering supports focused investigation of selected subjects, senders, or recipients.

C.4 Phase 3+4+5: Gold Analytics Pipeline

C.4.1 Scripts

D:\GOLD\outskrape\03_folder_statistics_dashboard_gold.py

D:\GOLD\outskrape\04_outlook_investigative_gold.py

D:\GOLD\outskrape\05_outlook_investigative_dashboard.py

C.4.2 Purpose

The Gold layer transforms Bronze data into curated analytical outputs. It is designed to run without Outlook. Scripts 03 and 04 read the Bronze SQLite database directly. Script 05 reads CSV outputs generated by Script 04 and builds a final interactive dashboard.

C.5 Script 03: Folder Statistics and Dashboard

C.5.1 Purpose

Script 03 produces folder-level statistical summaries and an interactive HTML dashboard from the Bronze SQLite database.

C.5.2 Main Processing Logic

1. Read e-mail data from the Bronze SQLite database.
2. Aggregate recipients by e-mail.
3. Aggregate attachments by e-mail.
4. Merge e-mails, recipients, and attachment summaries.
5. Classify messages as sent or received using configured own-sender identifiers.
6. Simplify folder paths for dashboard presentation.

7. Build folder-level statistical summaries.
8. Build folder-sender summaries.
9. Build daily, weekly, monthly, and hourly time-series tables.
10. Export CSV outputs.
11. Export an Excel workbook.
12. Generate an interactive HTML folder dashboard.

C.5.3 Main Outputs

- folder_statistical_summary.csv
- folder_sender_summary.csv
- folder_daily_timeseries.csv
- folder_weekly_timeseries.csv
- folder_monthly_timeseries.csv
- folder_hourly_timeseries.csv
- email_list_with_attachments.csv
- folder_statistical_summary.xlsx
- folder_dashboard.html

C.6 Script 04: Investigative Gold Analysis

C.6.1 Purpose

Script 04 performs deeper investigative analytics using the Bronze database and body-text files. It adds content-based analysis, theme tracking, communication-network tables, and anomaly detection.

C.6.2 Main Processing Logic

1. Read e-mails from the Bronze SQLite database.
2. Read recipients from the Bronze SQLite database.
3. Resolve and read saved body-text files.
4. Normalise text and tokenise message subjects and bodies.
5. Remove Portuguese, English, e-mail-noise, and low-information stopwords.
6. Build folder-level document representations.
7. Compute TF-IDF-like significant terms by folder.
8. Optionally load keyword themes from keys.xlsx.
9. Match keyword themes against subject and body text.
10. Build monthly and weekly theme-evolution tables.
11. Build communication network edges and nodes from sender-recipient links.
12. Detect weekly anomalies by folder using z-scores.
13. Export CSV and Excel outputs.
14. Write a summary HTML report.

C.6.3 Main Outputs

- tfidf_terms_by_folder.csv
- theme_email_matches.csv
- theme_evolution_monthly.csv
- theme_evolution_weekly.csv
- communication_network_edges.csv
- communication_network_nodes.csv
- weekly_email_counts_by_folder.csv
- weekly_anomalies_by_folder.csv
- outlook_investigative_analysis.xlsx
- outlook_investigative_analysis_report.html

C.6.4 Investigative Strategies

- **TF-IDF-like scoring:** each folder is treated as a document, and terms are scored by within-folder frequency and across-folder rarity.

- **Keyword-theme matching:** optional keyword definitions in keys.xlsx are used to classify e-mails into investigation themes.
- **Theme evolution:** matched themes are aggregated by month and week.
- **Communication network modelling:** sender-recipient relationships are converted into edge and node tables.
- **Anomaly detection:** weekly folder activity is scored using z-scores, with high absolute values marked as anomalies.
- **Report generation:** key outputs are summarised into an HTML report.

C.7 Script 05: Investigative Dashboard

C.7.1 Purpose

Script 05 builds an independent HTML dashboard from the CSV outputs generated by Script 04. It does not connect to Outlook and does not read the Bronze SQLite database directly.

C.7.2 Main Processing Logic

1. Read CSV outputs from the investigative-analysis folder.
2. Handle missing files safely by returning empty dataframes.
3. Convert pandas dataframes into JSON-like records.
4. Embed the data into an HTML document.
5. Build dashboard pages for:
 - significant terms by folder;
 - theme evolution;
 - communication network;
 - weekly anomalies.
6. Write the final HTML dashboard.

C.7.3 Main Output

- outlook_investigative_dashboard.html

C.8 Gold-Layer Resources Used

C.8.1 Libraries

- pandas – relational-style joins, aggregation, time-series tables, CSV and Excel export.
- sqlite3 – reading the Bronze SQLite database.
- pathlib.Path – path and directory handling.
- json – conversion of analytical tables into JavaScript-compatible structures.
- math – logarithmic IDF and numerical calculations.
- re – text normalisation and tokenisation.
- collections.Counter – term counting and frequency analysis.
- collections.defaultdict – grouped document, keyword, and aggregation structures.
- openpyxl via pandas.ExcelWriter – Excel workbook generation.

C.8.2 Strategies

- **Read-only Bronze consumption:** Gold scripts read the Bronze database but do not modify it.
- **No Outlook dependency:** after Phase 1, analytics can run on another machine without Outlook Classic.
- **Curated analytical outputs:** Gold outputs are structured as CSV, Excel, HTML report, and dashboard artefacts.
- **Folder-centric analysis:** folders are treated as meaningful analytical units.
- **Temporal aggregation:** activity is summarised by hour, day, week, and month.
- **Sender-recipient modelling:** recipient tables are aggregated and transformed into communication-network structures.
- **Client-side dashboards:** HTML dashboards embed data and JavaScript so they can be opened locally.

- **Graceful degradation:** missing optional keyword files or CSV files are handled without breaking the full pipeline.

C.9 Architectural Characterisation

C.9.1 Phase 1

Phase 1 is the canonical extraction phase. It is the most robust and reusable component because it creates a persistent, structured Bronze dataset. Its central role is to decouple Outlook access from later analysis.

C.9.2 Phase 2

Phase 2 is an exploratory, direct-to-Outlook analysis script. It is useful for rapid inspection and visual analytics, but it is less modular because extraction, transformation, visualisation, and reporting are combined in a single script.

C.9.3 Phase 3+4+5

Phases 3, 4, and 5 form the Gold analytics layer. This layer is more analytical and presentation-oriented. It uses the Bronze database and body-text files as inputs, produces structured summary tables, applies text and network analysis, detects anomalies, and publishes the results through Excel and HTML dashboards.

C.10 Recommended Interpretation

The complete workflow can be interpreted as a lightweight local data-platform architecture:

- **Bronze:** raw Outlook extraction, durable storage, and auditability.
- **Exploratory:** direct Outlook inspection and quick visual analysis.
- **Gold:** curated analytics, investigation-specific features, dashboards, and reporting.

The strongest architectural pattern is the separation between the Outlook-dependent extraction layer and the Outlook-independent analytics layer. This makes the later analysis more reproducible, portable, and easier to rerun.

Appendix D

Meteorological Data System

D.1 Overview of the Meteorological Data Pipeline

The script `weather_ipma_daily_reviewed.py` implements a local meteorological data pipeline based on the public open-data services of the Portuguese Institute for Sea and Atmosphere (IPMA). Its role is to collect recent hourly observations, preserve them in a local SQLite database, aggregate the observations into daily working periods, and generate graphical dashboards and CSV outputs for analytical review.

The current implementation uses a ranked three-station model. The system identifies the three nearest IPMA meteorological stations to the configured study location. The nearest station is stored as S1 and used as the primary operational meteorological reference. The second and third nearest stations are stored as S2 and S3 and are used as comparison stations in temporal line plots.

The pipeline can be interpreted as a Bronze-to-Silver workflow with a Gold interpretative layer:

- **Bronze:** raw hourly IPMA observations are collected and stored in SQLite for the selected stations;
- **Reviewed daily layer:** hourly observations are aggregated into configured daily periods, usually morning and afternoon;
- **Silver:** period-level and raw-hourly meteorological outputs are exported and visualised through weekly, monthly, quarterly and yearly dashboard reports;
- **Gold interpretation:** the dashboards support interpretation of weather patterns, station comparison, rainfall occurrence, wind regimes and reporting context.

D.2 Script

The operational Python script is located in:

D:\BRONZE\meteo\weather_ipma_daily_reviewed.py

The script is designed to be executed manually, through a batch file, or through Windows Task Scheduler.

D.3 Purpose

The script automates daily meteorological data collection and review. It identifies the three nearest IPMA meteorological stations to a configured study location, retrieves observations from the IPMA public API, stores raw hourly data, aggregates the previous day into configured periods, and produces analytical reports.

The main outputs are:

- a SQLite database containing the study location, IPMA station metadata, selected stations S1, S2 and S3, raw hourly observations, and daily period aggregates;
- monthly CSV exports of aggregated daily periods;
- classified CSV files for report periods;
- individual graphical plots for temperature, precipitation, wind, classification distributions and rain occurrence;

- consolidated dashboard images for weekly, monthly, quarterly and yearly reporting;
- dashboard maps showing the project location and the three selected meteorological stations.

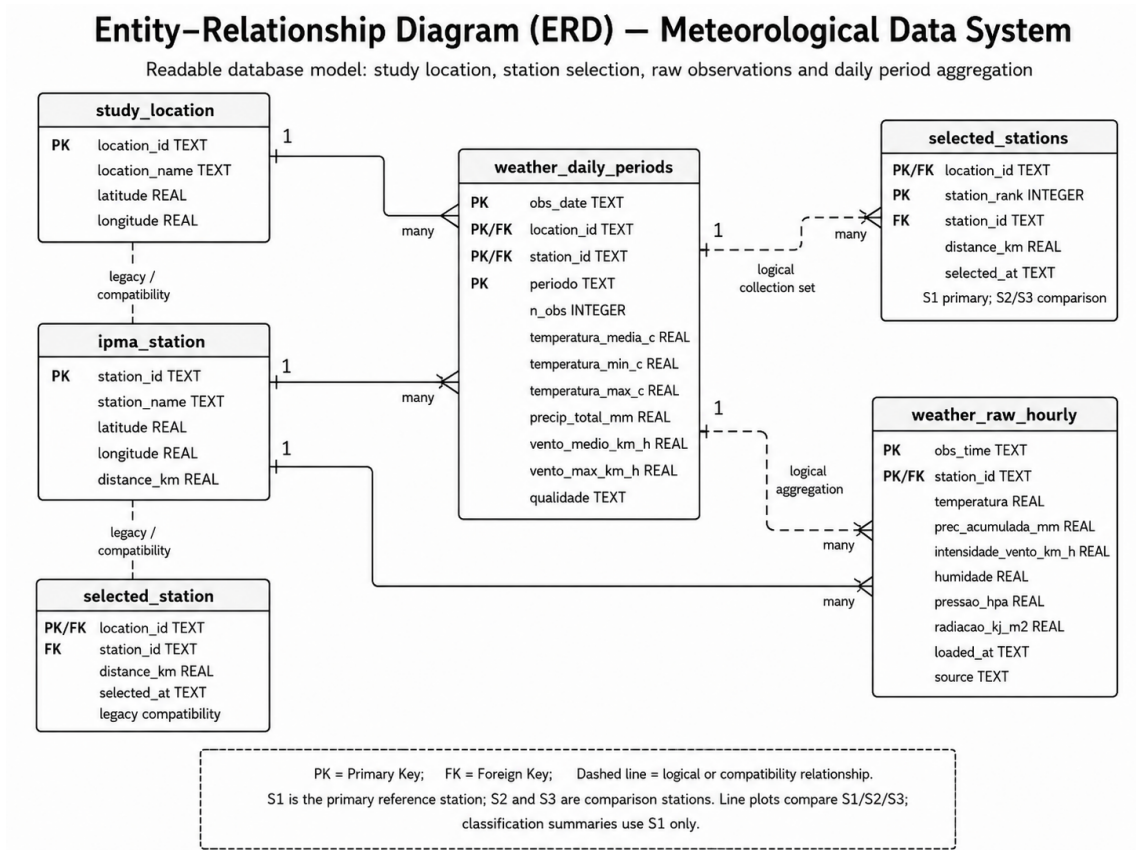


Figure D.1: Entity relationship diagram of the meteorological data system.

D.4 Main Workflow

D.4.1 Configuration Loading

The script starts by reading a config.json file. The configuration must include:

- study_location – identifier, name, latitude and longitude of the location being monitored;
- schedule – execution hour, minute and timezone;
- periods – the hour groups used to define daily periods, such as morning and afternoon;
- paths – database path, exports folder and Silver dashboard folders.

If the Silver output directory is not explicitly defined, the script uses:

D:\SILVER\meteo

The current reporting structure stores dashboard outputs in:

D:\SILVER\meteo\output plots\stats data
D:\SILVER\meteo\output plots\all data

The stats data reports are based on aggregated morning and afternoon records. The all data reports are based on raw hourly observations.

D.4.2 Database Initialisation

The script creates and maintains a local SQLite database. The database contains the following main tables:

- `study_location` – stores the configured analysis location;
- `ipma_station` – stores IPMA station metadata and calculated distance to the study location;
- `selected_station` – preserves compatibility with the earlier single-station workflow by storing the primary selected station;
- `selected_stations` – stores the three nearest stations ranked by distance as S1, S2 and S3;
- `weather_raw_hourly` – stores raw hourly observations from the IPMA API;
- `weather_daily_periods` – stores aggregated daily period-level indicators.

The database strategy is designed to preserve raw data and derived daily summaries separately. The raw table uses the combination of observation timestamp and station identifier as a primary key. This allows observations from several stations to coexist in the same table while preventing duplicate records for the same station and time.

The transition from the earlier single-station model to the current three-station model does not require deleting the previous database. Existing records remain valid, while new executions progressively add observations and aggregates for S2 and S3.

D.5 Station Selection Strategy

The script downloads the IPMA station catalogue from the public stations endpoint:

<https://api.ipma.pt/open-data/observation/meteorology/stations/stations.json>

For each available station, it calculates the distance to the configured study location using the Haversine formula. The three nearest stations are selected and stored in the database.

The selected stations are ranked as follows:

- **S1** – nearest station and primary operational reference;
- **S2** – second nearest station and comparison series;
- **S3** – third nearest station and comparison series.

This approach avoids hard-coding a station identifier and makes the script adaptable to different study locations. It also improves transparency because the reports show both the primary reference station and nearby comparison stations.

D.6 Observation Collection

The script retrieves recent hourly observations from the IPMA observations endpoint:

<https://api.ipma.pt/open-data/observation/meteorology/stations/observations.json>

Observations are stored for S1, S2 and S3 whenever data are available from the IPMA endpoint. The raw hourly table includes:

- observation timestamp;
- station identifier;
- temperature;
- humidity;
- accumulated precipitation;
- wind speed in kilometres per hour;
- wind speed in metres per second;
- wind direction identifier;
- pressure;
- radiation;
- load timestamp;
- data source.

Missing or invalid IPMA values are cleaned before insertion. In particular, the script treats -99.0 as a missing value.

D.7 Daily Period Aggregation

After storing hourly observations, the script aggregates the previous day according to the configured periods. The aggregation is performed separately for each selected station. Therefore, the same date and period may generate separate records for S1, S2 and S3.

For each station and period, the script calculates:

- number of hourly observations;
- mean temperature;
- minimum temperature;
- maximum temperature;
- mean humidity;
- total precipitation;
- mean wind speed;
- maximum wind speed;
- mean pressure;
- mean radiation;
- quality status.

The quality status is classified as:

- Complete – the expected number of observations exists;
- Partial – some observations exist, but fewer than expected;
- No data – no valid observations exist for the period.

This period-based aggregation is useful when the analysis needs to compare working windows, such as morning versus afternoon, rather than full-day averages only.

D.8 Export Strategy

The script can export monthly CSV files containing the period-level data. The export file uses the pattern:

`weather_periods_YEAR_MONTH.csv`

For report production, the script also exports classified CSV files using report-specific prefixes. These files preserve the analytical basis used to generate each dashboard.

D.9 Dashboard Reports

The script generates graphical dashboards for four temporal scopes:

- W – weekly report;
- M – monthly report;
- Q – quarterly report;
- Y – yearly report.

The dashboards are divided into two families:

- stats data – reports based on aggregated morning and afternoon records;
- all data – reports based on raw hourly observations.

Each report is saved using a standardised naming pattern containing the report type and the date range. Examples include:

`meteo-stats-data-M-2026-06-01-2026-06-30.png`
`meteo-all-data-M-2026-06-01-2026-06-30.png`

In addition to the consolidated dashboard, the script generates individual plots:

- temperature line plot;
- precipitation line plot;
- wind line plot;
- temperature class pie chart;
- precipitation class pie chart;

- wind class pie chart;
- rain/no-rain count plot.

The line plots compare the three selected stations. S1 is plotted as the primary reference series, while S2 and S3 are plotted as comparison series. Classification pie charts and rain/no-rain count plots are calculated only from S1. This avoids multiplying the number of observations for the same project period and preserves a single operational reference for categorical summaries.

D.10 Geographic Map Inset

Each consolidated dashboard includes a geographic inset map. The map uses the project coordinates and the coordinates of S1, S2 and S3 to define the relevant regional extent automatically.

The map includes:

- the project location;
- the three selected meteorological stations;
- station labels S1, S2 and S3;
- a basemap with roads, localities and geographic context;
- a legend;
- a north arrow;
- a scale bar;
- basemap attribution.

The map is included as a contextual element in the dashboard. Its purpose is to make the spatial relationship between the project location and the meteorological stations transparent. The map does not alter the stored data or the aggregation logic.

D.11 Interpretive Classification

Before visualisation, the script adds interpretive classes to the data. For classification summaries, only S1 is used. S2 and S3 are used for temporal comparison in the line plots.

D.11.1 Temperature Classes

- Low (<10 °C)
- Moderate (10--25 °C)
- High (>25 °C)
- No data

D.11.2 Precipitation Classes

- No precipitation
- Low / weak (<0.5 mm/h)
- Moderate (0.5--4 mm/h)
- High / strong (>4 mm/h)
- No data

D.11.3 Wind Classes

- Low / weak (<15 km/h)
- Moderate (15--35 km/h)
- High / strong (>35 km/h)
- No data

The script also creates a rain status variable:

- With rain
- Without rain
- No data

D.12 Command-Line Interface

The script supports several command-line modes:

- `--init` – initialise the database and select the three nearest IPMA stations;
- `--run-once` – run one complete daily collection, aggregation, export and report cycle;

- `--daemon` – keep the process running and execute the daily job at the configured time;
- `--export-month YEAR MONTH` – export a specific monthly CSV;
- `--build-silver-current` – rebuild W/M/Q/Y reports for yesterday without collecting new IPMA data;
- `--build-silver-period PERIOD REFERENCE_DATE` – build a specific W, M, Q or Y report for the period containing the given reference date.

D.13 Resources Used

D.13.1 Libraries

- `argparse` – command-line interface and execution modes;
- `json` – configuration loading from `config.json`;
- `math` – Haversine distance calculation;
- `os` – path checks, directory creation and file output;
- `sqlite3` – local database storage;
- `sys` – error reporting to standard error;
- `time` – sleeping between daemon executions;
- `datetime`, `date` and `timedelta` – date arithmetic and reporting periods;
- `zoneinfo.ZoneInfo` – timezone-aware scheduling, especially for Europe/Lisbon;
- `matplotlib` – non-interactive chart generation using the Agg backend;
- `matplotlib.dates` – date-aware chart axes;
- `matplotlib.pyplot` – plotting line charts, pie charts, bar charts and dashboard figures;
- `pandas` – dataframe processing, SQL reads, aggregation, classification and CSV export;
- `requests` – HTTP requests to IPMA public API endpoints;
- `contextily` – retrieval of basemap tiles for the geographic inset map;
- `pyproj` – coordinate transformation for map plotting.

D.13.2 External Resources

- IPMA station metadata endpoint:
`https://api.ipma.pt/open-data/observation/meteorology/stations/stations.json`
- IPMA hourly observations endpoint:
`https://api.ipma.pt/open-data/observation/meteorology/stations/observations.json`
- OpenStreetMap basemap tiles used for the dashboard location inset, when available.
- Local configuration file:
`config.json`
- Local SQLite database path defined in:
`cfg["paths"]["database"]`
- Default Silver output folder:
`D:\SILVER\meteo`

D.14 Strategies

D.14.1 Data Acquisition Strategy

- Use the IPMA public API as the authoritative weather source.
- Select the three nearest stations automatically from geographic coordinates.
- Store S1 as the primary reference station and S2/S3 as comparison stations.
- Insert or replace hourly observations using a composite primary key of observation time and station identifier.

D.14.2 Data Quality Strategy

- Convert invalid numeric values to null values.
- Treat the IPMA sentinel value -99.0 as missing data.
- Track the number of observations per configured period and station.
- Mark each period as complete, partial or without data.

D.14.3 Aggregation Strategy

- Aggregate hourly data into meaningful daily work periods.
- Calculate mean, minimum, maximum and total indicators depending on the variable.
- Preserve the station identifier in the aggregated records.
- Keep raw hourly data and aggregated daily-period data in separate tables.
- Export period summaries by month for external use.

D.14.4 Reporting Strategy

- Generate reports for multiple time windows: weekly, monthly, quarterly and yearly.
- Produce both aggregated-period dashboards and raw-hourly dashboards.
- Use consistent file naming based on report period and date range.
- Save both consolidated dashboard images and individual component plots.
- Export classified data alongside graphics to preserve the analytical base of each report.
- Use S1-only classification charts and three-station comparison line plots.
- Include a geographic inset map showing the project location and S1/S2/S3.

D.14.5 Automation Strategy

- Support manual initialisation through `--init`.
- Support one-off execution through `--run-once`.
- Support unattended daily execution through `--daemon`.
- Use timezone-aware scheduling for reliable daily execution.
- Allow reports to be rebuilt without re-collecting data.

D.15 Architectural Characterisation

The script combines Bronze and Silver responsibilities in a compact meteorological data pipeline.

D.15.1 Bronze-Like Responsibilities

- Retrieve data from the IPMA API.
- Store raw hourly observations in SQLite.
- Preserve station metadata and selected-station information.
- Store the ranked three-station selection model.
- Maintain a durable local database for repeatable analysis.

D.15.2 Reviewed Daily Responsibilities

- Convert raw hourly observations into configured daily periods.
- Calculate meteorological indicators for each period and station.
- Assign quality labels based on observation completeness.

D.15.3 Silver Responsibilities

- Produce clean monthly CSV exports.
- Classify S1 records into interpretive weather categories.
- Generate visual reports for different reporting horizons.
- Compare S1, S2 and S3 in line plots.
- Produce dashboard images suitable for review and documentation.
- Include geographic context through a map inset.

D.16 Recommended Interpretation

The script is best understood as a daily IPMA weather ingestion and reporting job. It automates the full path from external public API data to local analytical outputs. Its strongest design features are:

- automatic three-station selection;
- distinction between S1 as primary reference and S2/S3 as comparison stations;
- durable SQLite storage;
- preservation of raw hourly data;
- explicit period-based aggregation;
- weather-quality and interpretive classification;
- separate raw-hourly and aggregated-period reports;
- repeatable dashboard generation;
- geographic map inset showing project and station locations;
- command-line modes for initialisation, execution, daemon scheduling, export and report re-building.

Appendix E

Daily reports

E.1 Overview of the Daily Reports Pipeline

The daily reports workflow is organised as a two-phase pipeline for processing RDF (*Registos Diários de Fiscalização*) documents and transforming them into resource-analysis outputs.

1. **Phase 1 – SILVER consolidation:** the script `consolidar_registos_diarios.py` reads daily RDF Word documents, extracts textual sections and embedded resource tables, and creates monthly and global Excel consolidation files.
2. **Phase 2 – GOLD resource analysis:** the script `gold_relav_diario_resource_analysis.py` reads the SILVER resource summaries and creates higher-level analytical outputs, including plots, monthly averages, differences between actual and proposed resources, a Gold Excel workbook, and a processing summary.

The main analytical theme is the comparison between resources actually registered in the RDF documents and the resources proposed by the contractor.

E.2 Phase 1: SILVER Consolidation of Daily RDF Records

E.2.1 Script

D:\BRONZE\Relat_diario\consolidar_registos_diarios.py

E.2.2 Purpose

This script consolidates daily inspection reports written as .docx files. It scans the input folder, including monthly subfolders, extracts relevant textual sections and resource data, and creates structured Excel outputs.

The main extracted sections are:

- ATIVIDADES REALIZADAS OU EM CURSO;
- AÇÕES DESENVOLVIDAS PELA EQUIPA DE FISCALIZAÇÃO E COORDENAÇÃO DE SEGURANÇA EM OBRA;
- NOTAS;
- RECURSOS.

The resulting Excel workbooks contain daily matrices and resource summaries that allow the user to inspect the evolution of activities, notes, personnel, and equipment during each month.

E.2.3 Main Inputs

- RDF Word documents in .docx format.
- Monthly subfolders containing RDF files.
- Contractor proposal workbook for equipment: *Carga de Equipamento.xlsx*
- Contractor proposal workbook for personnel: *Carga de Pessoal.xlsx*

E.2.4 Main Outputs

- Monthly Excel consolidation workbooks: consolidado_RDF_YYYY_MM.xlsx
- Global resource workbook: consolidado_RDF_recursos_global.xlsx
- Processing summary: resumo_processamento_RDF.txt

E.2.5 Main Processing Logic

1. Parse command-line arguments for input folder, output folder, and recursive folder scanning.
2. Search all RDF .docx files, ignoring temporary Word files and previously consolidated files.
3. For each RDF file, identify the report date from the filename or, as a fallback, from the internal XML text of the Word document.
4. Extract the main body paragraphs from the Word document.
5. Detect relevant section headings using normalised aliases.
6. Extract activities, inspection actions, and notes.
7. Detect local or work-front headings and prefix activities with that context.
8. Extract embedded Excel files from the Word package under word/embeddings/.
9. Read personnel and equipment quantities from the embedded resource table.
10. Read proposed resource quantities from contractor proposal workbooks.
11. Group all processed reports by year and month.
12. Build monthly matrices for activities, inspection actions, notes, personnel, and equipment.
13. Build daily totals for personnel, equipment, and total resources.
14. Compare RDF actual resources with proposed resources.
15. Write monthly Excel workbooks with formatted sheets and charts.
16. Write a global resource workbook covering all processed RDF dates.
17. Write a TXT execution summary with created files and warnings.

E.2.6 Excel Sheets Created

Each monthly workbook contains:

- Atividades;
- Ações Fiscalização;
- Notas;
- Pessoal;
- Equipamento;
- Resumo Recursos;
- Log.

E.2.7 Resource Summary Logic

The Resumo Recursos sheet compares:

- total personnel registered in RDF;
- total equipment registered in RDF;
- total RDF resources;
- proposed personnel;
- proposed equipment;
- total proposed resources.

The comparison is performed day by day. This enables detection of differences between actual daily mobilisation and the contractor proposal.

E.2.8 Libraries Used

- argparse – command-line interface.
- calendar – month-day calculation.
- math – chart-axis scaling and numerical utilities.
- re – regular expressions for text cleaning, date parsing, and splitting.
- shutil – cleanup of temporary folders.
- tempfile – temporary extraction of embedded Excel files.
- unicodedata – accent removal and text normalisation.

- `collections.defaultdict` – accumulation of quantities and grouped records.
- `dataclasses` – structured representation of each daily report.
- `datetime` – date parsing and monthly grouping.
- `pathlib.Path` – robust filesystem path handling.
- `typing` – type hints for dictionaries, lists, tuples, and optional values.
- `zipfile.ZipFile` – direct inspection of the .docx package.
- `python-docx` – reading Word document paragraphs.
- `openpyxl` – Excel creation, workbook reading, formatting, charts, and image embedding.
- `matplotlib` – generation of chart images for Excel insertion.

E.2.9 Strategies Used

- **Recursive folder scanning:** the script searches monthly subfolders for RDF files.
- **Robust date extraction:** dates are first parsed from filenames and then from internal DOCX XML as a fallback.
- **Text normalisation:** accents, spacing, punctuation, and case are normalised to improve heading detection.
- **Section aliasing:** multiple spelling variants are accepted for the same RDF section.
- **Contextual activity extraction:** probable local or work-front headings are detected and attached to activity descriptions.
- **Embedded Excel extraction:** resource tables embedded inside Word documents are extracted from the DOCX ZIP structure.
- **Flexible proposal parsing:** contractor proposal workbooks are read using two supported layouts: dates as columns or dates as rows.
- **Monthly matrix construction:** activities and notes are represented as daily presence matrices, while personnel and equipment are represented as quantity matrices.
- **Actual-vs-proposed comparison:** RDF resources are compared with proposed resources at daily level.
- **Audit logging:** warnings are preserved in the workbook log and TXT processing summary.

E.3 Phase 2: GOLD Resource Analysis

E.3.1 Script

D:\GOLD\Relat_diario\gold_relato_diario_resource_analysis.py

E.3.2 Purpose

This script reads the SILVER resource outputs and creates the GOLD analytical layer for RDF resource evolution. It focuses on summarising and visualising the difference between actual resources recorded in RDF and resources proposed by the contractor.

E.3.3 Main Inputs

- Preferred input: D:\SILVER\Relat_diario\outputs\consolidado_RDF_recursos_global.xlsx
- Fallback inputs: D:\SILVER\Relat_diario\outputs\consolidado_RDF_*.xlsx
- Required sheet: Resumo Recursos

E.3.4 Main Outputs

- GOLD Excel workbook: gold_RDF_resource_evolution.xlsx
- Daily resource evolution plot: gold_RDF_resource_evolution.png
- Monthly average resource plot: gold_RDF_monthly_average_resources.png
- Monthly average differences plot: gold_RDF_monthly_average_differences.png
- GOLD processing summary: gold_RDF_processing_summary.txt

E.3.5 Main Processing Logic

1. Ensure that the GOLD output folder exists.
2. Locate SILVER resource files, preferring the global resource workbook.
3. If the global file is not available, read monthly SILVER workbooks.

4. Read the Resumo Recursos sheet from each source workbook.
5. Normalise Portuguese and English column variants to canonical names.
6. Validate the expected resource-summary columns.
7. Convert dates and numeric columns.
8. Aggregate duplicate dates by summing daily values.
9. Build monthly average summaries.
10. Calculate monthly average differences between actual RDF values and proposed values.
11. Generate a daily evolution plot comparing actual and proposed personnel and equipment.
12. Generate a monthly average plot.
13. Generate a monthly difference plot.
14. Create the GOLD Excel workbook with summary, daily series, monthly averages, and embedded charts.
15. Write a TXT summary listing source files and generated outputs.

E.3.6 Expected Resource Columns

The script expects the following canonical columns:

- Data;
- Pessoal total RDF;
- Equipamento total RDF;
- Recursos totais RDF;
- Pessoal proposto;
- Equipamento proposto;
- Recursos propostos.

The script also accepts selected English variants and renames them to the canonical column names.

E.3.7 Monthly Summary Indicators

For each month, the script calculates:

- first RDF date in the month;
- last RDF date in the month;
- number of RDF days;
- average actual personnel;
- average actual equipment;
- average actual total resources;
- average proposed personnel;
- average proposed equipment;
- average proposed total resources;
- average personnel difference;
- average equipment difference;
- average total resource difference.

The difference is calculated as:

$\text{difference} = \text{actual RDF average} - \text{proposed average}$

Therefore, positive values indicate that actual resource mobilisation was above the proposal, while negative values indicate that actual mobilisation was below the proposal.

E.3.8 Libraries Used

- math – axis scaling and rounded plotting limits.
- pathlib.Path – input and output path management.
- typing – type hints.
- pandas – Excel reading, dataframe cleaning, date parsing, aggregation, monthly summaries, and numeric conversion.
- matplotlib – non-interactive chart generation.
- matplotlib.dates – date-axis formatting.
- openpyxl – creation of the GOLD Excel workbook, formatting, and image embedding.

E.3.9 Strategies Used

- **SILVER-first reading:** the Gold layer does not parse RDF Word files directly; it consumes curated SILVER outputs.
- **Global-first strategy:** the script prefers the global SILVER resource workbook and falls back to monthly workbooks when needed.
- **Column normalisation:** Portuguese and English column variants are mapped to a canonical schema.
- **Schema validation:** expected columns are checked before analysis.
- **Duplicate-date aggregation:** repeated dates from multiple SILVER files are summed into one daily record.
- **Monthly averaging:** daily resource series are summarised by month.
- **Difference analysis:** actual RDF values are compared against proposed resource values.
- **Chart-based reporting:** the Gold outputs include daily evolution, monthly averages, and monthly difference plots.
- **Excel packaging:** analytical tables and plots are combined into a single GOLD workbook.
- **Traceability:** the 'TXT' summary records dates found, source files used, and Gold files created.

E.4 Architectural Characterisation

E.4.1 Phase 1 Characterisation

Phase 1 is a document-consolidation and SILVER preparation stage. It transforms semi-structured daily Word documents into structured monthly Excel workbooks. It also extracts embedded resource tables and creates a global resource workbook that acts as the analytical bridge to the Gold layer. Its main role is to convert unstructured and semi-structured field documentation into structured, auditable, date-based matrices.

E.4.2 Phase 2 Characterisation

Phase 2 is a Gold analytical layer focused on resource evolution. It does not reprocess Word files. Instead, it reads the curated SILVER resource summaries, standardises the data, aggregates daily and monthly indicators, generates plots, and packages the results into a final analytical workbook. Its main role is to provide management-level interpretation of actual RDF resources compared with contractor-proposed resources.

E.5 Recommended Interpretation

The two scripts form a practical reporting architecture:

- **Bronze source:** RDF Word files and contractor proposal workbooks.
- **SILVER consolidation:** structured monthly RDF workbooks and global resource summaries.
- **GOLD analysis:** resource evolution analytics, monthly averages, differences, plots, Excel reporting, and processing summary.

The strongest architectural pattern is the separation between document parsing and analytical reporting. Phase 1 handles the complexity of Word and embedded Excel extraction, while Phase 2 works only with curated SILVER tables and focuses on resource analysis.

Appendix F

Price revision

F.1 Overview of the Price Revision Pipeline

The script `calcular_revisao_precos.py` implements a SILVER-layer price revision workflow for construction payment certificates. It calculates the price revision of a current *auto de empreitada* using contractual information, the measured or declared value of the auto, applicable IMPIC price indices, and a price-revision formula.

The script is designed to support both formal calculation and auditability. It reads contractual and auto templates, obtains or updates price indices, validates formula components, checks for missing indices, calculates the revision coefficient, writes a detailed Excel result workbook, updates the historical register of autos, and generates a visual representation of the price revision evolution.

F.2 Script

D:\SILVER\Revisao\calcular_revisao_precos.py

F.3 Purpose

This script serves to calculate the price revision associated with a construction monthly payment and to support its interpretation over time. The calculation is based on:

- the contract and financial schedule workbook;
- the current auto workbook;
- a normalised price-index workbook or CSV file;
- IMPIC index files downloaded or already stored locally;
- a contractual or official price-revision formula;
- the historical record of previous autos.

The script calculates both:

- the revision value for the current period;
- the accumulated revision value after the current auto.

It also avoids duplicating values when the auto is cumulative by subtracting the previously registered accumulated value.

F.4 Default Working Folder

The default working folder is:

D:\SILVER\Revisao

The typical files used by the script are:

- 01_contrato_cronograma_template.xlsx;
- 02_auto_template.xlsx;
- 03_indices_normalizados.xlsx;
- 04_historico_autos.xlsx;

- downloads_impic/;
- resultados_revisao/.

F.5 Main Inputs

F.5.1 Contract and Schedule Workbook

The contract workbook contains the contract-level metadata and the financial schedule. The script expects at least:

- a Contrato sheet in key/value format;
- a Cronograma_Financeiro sheet;
- optionally, a Biblioteca_Formulas sheet;
- optionally, a Formula_Revisao sheet for fallback use.

Important contract fields include:

- contract identifier;
- price revision formula type;
- base date for the indices;
- formula components and weights.

F.5.2 Auto Workbook

The auto workbook contains the current payment certificate information. The script expects:

- an Auto sheet in key/value format;
- a Trabalhos_Medidos sheet.

The auto may provide either a period value or an accumulated value. The script can decide automatically or follow the command-line option selected by the user.

F.5.3 Normalised Indices File

The normalised indices file may be provided as:

- 03_indices_normalizados.xlsx;
- or 03_indices_normalizados.csv.

The expected canonical fields are:

- Mes;
- Codigo_Indice;
- Descricao;
- Categoria;
- Valor_Indice;
- Base;
- Fonte_URL;
- Observacoes.

F.5.4 IMPIC Source Files

The script uses IMPIC price-index publications as the official source of index data. It can download files from the IMPIC web page or use files already present in the local downloads_impic folder.

F.5.5 Historical Autos Workbook

The file 04_historico_autos.xlsx stores previous calculations. This is essential when autos are cumulative, because the script uses the previous accumulated value to calculate only the new period value.

F.6 Main Processing Logic

1. Parse command-line arguments and build a configuration object.
2. Resolve all paths relative to the base working directory.
3. Read the contract workbook.
4. Read the auto workbook.
5. Read the historical auto workbook, if it exists.
6. Determine the contract identifier, auto number, base-index month, and auto-index month.

7. Select the applicable price-revision formula.
8. Load or obtain price indices from local files and, if enabled, from IMPIC.
9. Normalise index tables into a canonical structure.
10. Check whether all required base and auto indices exist.
11. If indices are missing, generate an auxiliary workbook with the missing entries to fill.
12. Determine the value of the current auto period.
13. Calculate the coefficient for each formula component.
14. Calculate the total coefficient K.
15. Calculate the revision value for the current auto.
16. Calculate the accumulated revision value.
17. Update the historical autos dataset.
18. Generate the price revision evolution plot.
19. Write the detailed result workbook, including the plot data and embedded chart.
20. Update the historical autos workbook.

F.7 Formula Selection Strategy

The script applies a controlled hierarchy to select the price-revision formula.

1. First, it tries to use the contractual formula library inside the contract workbook, usually in a sheet such as `Biblioteca_Formulas`.
2. If the contractual library is unavailable or invalid, it tries to use the local official formula library:

`biblioteca_formulas_oficiais.xlsx`
3. If the selected formula is an official formula from F01 to F23, the script can create or update the local official formula library using built-in formula components.
4. If the selected formula is not an official F01–F23 formula, the script can fall back to the `Formula_Revisao` sheet.

This hierarchy is deliberately conservative. If an official formula is declared but cannot be validated, the script stops instead of silently using the fallback formula.

F.8 Official Formula Library

The script contains an embedded library of official formula components for formulas F01 to F23. Each formula is represented as a list of:

- index code;
- component description;
- weight;
- revisable or non-revisable flag;
- legal source;
- observations.

The official formula sources are associated with Portuguese legal acts, namely:

- Despacho n.º 1592/2004 for formulas F01 to F14;
- Despacho n.º 22637/2004 for formulas F15 to F23.

The local official formula workbook also records legal source information and usage instructions, supporting traceability and audit review.

F.9 Formula Validation

Before calculation, formula components are cleaned and validated. The script checks:

- whether required columns exist;
- whether index codes are valid;
- whether weights are numeric;
- whether the sum of weights is approximately 1.00;
- whether at least one component is revisable.

Accepted index codes include examples such as:

- F01, F02, ..., F23;

- M01, M02, etc.;
- EQUIP;
- GLOBAL;
- FIXO.

F.10 Index Acquisition and Normalisation

F.10.1 Local Indices

The script first tries to read an existing local indices file. It accepts both Excel and CSV formats and normalises column names to the canonical schema.

F.10.2 IMPIC Download

When download mode is enabled, the script tries to access the IMPIC publication page, find Excel links, download new files, and combine them with existing local files.

F.10.3 CIFE Parsing

The script contains specific parsing logic for CIFE-style IMPIC files:

- **Quadro I:** labour indices, including formulas such as F01 to F23;
- **Quadro II:** material indices such as M01 to M58;
- **Quadro III:** equipment support indices, normalised as EQUIP.

A generic parser is also used as a fallback when a file is not recognised as CIFE.

F.10.4 Normalisation Report

During automatic IMPIC normalisation, the script creates a report workbook:

relatorio_normalizacao_impic.xlsx

This report records which IMPIC files were processed, ignored, or failed.

F.11 Missing Indices Strategy

The script identifies every index needed for:

- the base month;
- the auto month;
- every revisable component of the formula.

If required indices are missing, the script generates:

03_indices_em_falta_para_preencher.xlsx

This auxiliary workbook indicates the missing month/index pairs and allows the user to fill or validate the required values manually.

This strategy prevents calculation with incomplete index data.

F.12 Auto Value Strategy

The script supports multiple ways to determine the value to revise.

F.12.1 Declared Period Value

If the auto contains a period value, the script can use it directly.

F.12.2 Declared Accumulated Value

If the auto contains an accumulated value, the script subtracts the accumulated value of the previous auto stored in the historical workbook. This produces the new value for the current period.

F.12.3 Measured Works

If the command-line option selects measured works, the script sums the Valor_Periodo_Sem_IVA column from the Trabalhos_Medidos sheet.

F.13 Price Revision Calculation

For each formula component, the script calculates:

```
ratio = index_auto / index_base  
component_coefficient = weight * ratio
```

For non-revisable components or the fixed component FIXO, the ratio is treated as 1.0.
The total coefficient is:

```
K = sum(component_coefficient)
```

The revision value for the current auto is:

```
revision_value = auto_period_value * (K - 1)
```

The revised auto value is:

```
revised_auto_value = auto_period_value + revision_value
```

The accumulated revision after the current auto is calculated by adding the current revision value to the previous accumulated revision stored in history.

F.14 Price Revision Evolution Plot

In addition to the tabular Excel outputs, the script also generates a graphical representation of the price revision process over time. This visual output is named *Price Revision Evolution* and is created automatically after each successful calculation.

The plot combines the planned and actual monthly financial execution with the monetary effect of the price revision calculation. It includes:

- planned monthly values from the contract financial schedule;
- actual monthly values from the historical sequence of autos;
- monthly price revision values;
- accumulated price revision values.

The planned and actual monthly values are represented as bars and are read on the left vertical axis. The monthly price revision and accumulated price revision are represented as line series and are read on the right vertical axis. This distinction is necessary because the monthly execution values and the revision values have different orders of magnitude. If both were represented on the same axis, the price revision lines would be visually compressed and difficult to interpret.

The generated image follows the naming pattern of the corresponding result workbook:

```
resultado_revisao_precos_<ID_Contrato>_<Numero_Auto>_<Mes_Auto>_  
price_revision_evolution.png
```

For example:

```
resultado_revisao_precos_C001_Auto_006_2025-06_  
price_revision_evolution.png
```

The image is saved in the same result folder used for the Excel calculation outputs:

```
D:\SILVER\Revisao\resultados_revisao
```

The result workbook is also extended with additional sheets related to the chart. The sheet `Price_Revision_Evolution_Data` stores the tabular data used to build the plot, while the sheet `Price_Revision_Evolution` embeds the generated image inside the Excel workbook. This allows the visual output to remain connected to the calculation file and supports later review, reporting and audit.

The plot is useful because it provides a compact interpretation of how the price revision evolves as autos are processed. The planned monthly bars provide the contractual financial baseline, while the actual monthly bars show the certified values already registered in the auto history. The monthly price revision line shows the revision effect of each auto, and the accumulated price revision line shows the total revision value reached after each period. In months without certified autos, the actual monthly value and monthly revision remain zero, while the accumulated revision remains stable. This behaviour helps distinguish between the absence of new certified production and changes in the cumulative financial effect of price revision.

This graphical output therefore complements the detailed calculation sheets. The workbook continues to preserve the formula, indices and component-level calculation, while the plot provides a faster visual reading of the relationship between planned financial execution, actual certification and the accumulated impact of the Portuguese price revision procedure.

F.15 Main Outputs

The result workbook is created automatically in:

`resultados_revisao/`

unless an explicit output path is provided.

The result file name follows the pattern:

`resultado_revisao_precos_<ID_Contrato>_<Numero_Auto>_<Mes_Auto>.xlsx`

The workbook includes:

- `Resumo` – main calculation indicators;
- `Detalhe_Calculo` – component-level calculation;
- `Historico_Autos` – updated auto history;
- `Formula_Usada` – formula actually used;
- `Diagnostico_Formula` – formula origin and diagnostic notes;
- `Indices_Usados` – index table used in the calculation;
- `Trabalhos_Medidos` – measured works from the auto workbook;
- `Cronograma` – financial schedule from the contract workbook;
- `Price_Revision_Evolution_Data` – data used to generate the price revision evolution chart;
- `Price_Revision_Evolution` – embedded graphical representation of planned monthly values, actual monthly autos, monthly revision and accumulated revision.

The historical workbook is also updated:

`04_historico_autos.xlsx`

The plot image is also saved as a separate .png file in the result folder, using the same base name as the result workbook.

F.16 Command-Line Interface

The script supports the following main options:

- `--base-dir` – working folder, defaulting to `D:\SILVER\Revisao`;
- `--contrato` – contract and financial schedule workbook;
- `--auto` – current auto workbook;
- `--indices` – normalised indices file;
- `--historico` – historical autos workbook;
- `--out` – optional explicit result workbook path;
- `--downloads` – local folder for downloaded IMPIC files;

- `--valor-base-origem` – selects whether the base value comes from the auto or measured works;
- `--tipo-valor-auto` – defines whether the auto value is treated as period, accumulated, or automatically inferred;
- `--sem-download` – disables IMPIC downloading and uses only local index files.

F.17 Resources Used

F.17.1 Libraries

- `argparse` – command-line interface and execution options.
- `re` – parsing formula codes, index codes, months, filenames, and text patterns.
- `dataclasses.dataclass` – structured configuration object.
- `pathlib.Path` – path resolution and file management.
- `typing` – type hints for dictionaries, optional values, and tuples.
- `pandas` – Excel and CSV reading, table cleaning, index normalisation, grouping, calculation, and workbook export.
- `matplotlib` – generation of the price revision evolution bar-and-line chart.
- `requests` – optional download of IMPIC index files and legal-source pages.
- `beautifulsoup4` – parsing links from the IMPIC publication web page.
- `xlrd` – reading legacy .xls IMPIC files.
- `openpyxl` – Excel writing through `pandas.ExcelWriter`.
- `openpyxl.drawing.image.Image` – insertion of the generated plot into the Excel result workbook.

F.17.2 External Resources

- IMPIC price-revision publication page:
<https://www.impic.pt/impic/pt-pt/indices-de-revisao-de-precos-em-empregadas/publicacao-em-diario-da-rep>
- Official formula source for formulas F01–F14:
Despacho n.º 1592/2004
- Official formula source for formulas F15–F23:
Despacho n.º 22637/2004
- Local official formula library:
`biblioteca_formulas_oficiais.xlsx`
- Local IMPIC downloads folder:
`downloads_impic`
- Result folder containing Excel outputs and price revision evolution plots:
`resultados_revisao`

F.18 Strategies

F.18.1 Robust Input Strategy

- Accepts configurable paths for contract, auto, indices, history, downloads, and output.
- Reads key/value sheets where column A stores the field and column B stores the value.
- Normalises field names and column names to reduce errors caused by accents, spacing, underscores, or language variation.
- Accepts Excel and CSV normalised index files.

F.18.2 Formula Strategy

- Prioritises the formula defined in the contract workbook.
- Uses a local official formula library when the contractual library is unavailable.
- Can create or update the official formula library for formulas F01 to F23.
- Stops execution when an official formula is declared but cannot be validated.
- Allows fallback to `Formula_Revisao` only when no valid official formula type is defined.

F.18.3 Index Strategy

- Reads local indices first.
- Optionally downloads or updates IMPIC index files.
- Normalises CIFE and generic IMPIC Excel formats.
- Merges local and downloaded indices, preserving the most recent values for duplicate month/index pairs.
- Creates an explicit missing-index workbook when calculation inputs are incomplete.

F.18.4 Calculation Strategy

- Calculates component-level index ratios.
- Applies component weights to obtain the total coefficient K.
- Treats fixed or non-revisable components as having ratio 1.0.
- Calculates both current-period and accumulated revision values.
- Supports both period autos and cumulative autos.
- Prevents negative period values when accumulated values conflict with the historical record.

F.18.5 Visualisation Strategy

- Aligns planned monthly values from the contract schedule with actual monthly autos from the historical record.
- Represents monthly planned and actual values as bars on the left axis.
- Represents monthly and accumulated price revision values as lines on the right axis.
- Saves the chart as a separate image file in the result folder.
- Embeds the chart into the Excel result workbook for reporting and audit.

F.18.6 Audit and Traceability Strategy

- Records the formula origin used in the calculation.
- Writes formula diagnostics to the result workbook.
- Stores the index table used in the result workbook.
- Maintains a historical register of autos.
- Writes the updated history into both the result workbook and the standalone historical workbook.
- Creates a normalisation report for IMPIC files.
- Creates a missing-index workbook instead of calculating with incomplete data.
- Preserves the chart data and visual output together with the calculation workbook.

F.19 Architectural Characterisation

This script is best understood as a SILVER analytical calculation module. It does not merely perform an arithmetic operation; it manages a controlled calculation workflow with input validation, official-data normalisation, formula selection, history management, visualisation, and auditable outputs.

F.19.1 SILVER Responsibilities

- Consolidate contract, auto, formula, index, and history data into one calculation context.
- Normalise semi-structured and official index sources.
- Validate formula and index availability before calculation.
- Produce clean calculation outputs suitable for review.
- Generate visual outputs to support interpretation and reporting.

F.19.2 Calculation Responsibilities

- Determine the correct auto period value.
- Apply the price-revision formula component by component.
- Calculate the total coefficient K.
- Calculate current and accumulated revision values.
- Update the auto history to support the next calculation.

F.19.3 Visualisation Responsibilities

- Compare planned monthly values and actual monthly autos.
- Display the monetary effect of monthly and accumulated price revision.
- Use a secondary axis for revision values to preserve readability.
- Store the plot as both an image file and an embedded workbook element.

F.19.4 Audit Responsibilities

- Preserve the formula used.
- Preserve the indices used.
- Preserve the contract schedule and measured works.
- Preserve the historical auto state.
- Record diagnostics and missing-data situations.
- Preserve the data used to generate the price revision evolution plot.

F.20 Recommended Interpretation

The script implements a robust and auditable workflow for construction price revision. Its strongest design features are:

- official and contractual formula hierarchy;
- automatic IMPIC index acquisition and normalisation;
- conservative validation before calculation;
- explicit missing-index handling;
- support for both period and accumulated autos;
- historical tracking to prevent duplicated accumulated revisions;
- generation of a price revision evolution plot;
- detailed Excel outputs for technical review and audit.

The price revision evolution plot should be interpreted as a visual complement to the calculation workbook. It does not replace the detailed coefficient calculation, formula validation or index verification. Instead, it provides a compact view of how the planned financial baseline, actual certified production, monthly revision and accumulated revision evolve across the contract timeline.

Appendix G

Financery chronogram

G.1 Overview of the Financial Schedule Evolution Pipeline

The script `evolucao_cronograma_financeiro.py` implements a SILVER-layer financial schedule evolution report. Its purpose is to compare the planned financial schedule of the contract against the actual execution represented by the autos already processed in the price-revision workflow.

The output is a planned-versus-actual financial evolution dataset, an Excel workbook, and a chart image. The chart combines monthly bars and cumulative lines, allowing the user to compare:

- planned monthly values;
- actual monthly values from autos;
- planned cumulative value;
- actual cumulative value.

The main analytical objective is to show whether the financial execution of the contract is ahead of, aligned with, or behind the planned schedule.

G.2 Script

D:\SILVER\cronograma_financeiro\evolucao_cronograma_financeiro.py

G.3 Purpose

The script generates a financial schedule evolution chart and supporting Excel workbook. It ensures that:

- all months from the planned schedule are represented;
- planned monthly bars are shown for all months;
- actual monthly bars are shown only in months with autos;
- months without autos receive an actual monthly value of zero;
- planned and actual cumulative lines are shown over the full planned timeline.

This is important because the actual execution must be aligned against the full planned contract schedule, not only against months where payment certificates already exist.

G.4 Main Inputs

G.4.1 Planned Financial Schedule

The planned schedule is read from:

D:\SILVER\Revisao\01_contrato_cronograma_template.xlsx

using the sheet:

Cronograma_Financeiro

The script searches for a valid month or period column. It prioritises date columns such as:

- Data_Inicio;
- Data Inicio;
- Inicio;
- Início;
- Data_Fim;
- Data Fim;
- Fim;
- Mes;
- Mês;
- Data;
- Mes_Referencia;
- Mês_Referencia;
- Período;
- Período.

For the planned monthly value, the script accepts columns such as:

- Valor_Previsto_Sem_IVA;
- Valor_Mensal_Previsto;
- Mensal_Previsto;
- Valor_Periodo_Previsto;
- Previsto_Mensal;
- Planned Monthly;
- Valor Previsto;
- Valor_Mensal.

The cumulative planned value may be read from an existing column. If it is absent or fully empty/zero, the script recalculates it as the cumulative sum of the planned monthly values.

G.4.2 Actual Financial Execution

The actual execution is read from the most recent price-revision result workbook found in:

D:\SILVER\Revisao\resultados_revisao

The selected file must match the pattern:

resultado_revisao_precos_<Contrato>_Auto_<Numero>_<YYYY-MM>.xlsx

The script reads the sheet:

Historico_Autos

and uses the following information:

- period column: Período_Auto, Período_Auto, or Mes_Indice_Aplicavel;
- actual monthly value column: Valor_Periodo_Sem_IVA.

If several autos exist for the same month, the values are grouped and summed by period.

G.5 Main Outputs

The script writes the following outputs to:

D:\SILVER\cronograma_financeiro

G.5.1 Excel Output

05_financial_schedule_planned_vs_actual.xlsx

This workbook contains:

- Evolution_Data – merged planned and actual schedule data;
- Planned_Schedule – planned schedule after normalisation;
- Actual_Autos – actual auto values extracted from the latest price-revision result;
- Notes – metadata describing the source files and alignment rules.

G.5.2 Image Output

05_financial_schedule_planned_vs_actual.png

The image shows:

- orange bars for planned monthly values;
- grey bars for actual monthly values;
- green line for planned cumulative value;
- blue line for actual cumulative value.

G.6 Main Processing Logic

1. Parse command-line arguments.
2. Resolve the contract workbook, price-revision results folder, and output folder.
3. Read the planned financial schedule from the contract workbook.
4. Normalise column names by stripping whitespace.
5. Detect the period/date column.
6. Convert period values into the canonical YYYY-MM format.
7. Detect the planned monthly value column.
8. Detect or recalculate the planned cumulative value.
9. Sort planned periods and assign an order number.
10. Create labels such as M1, M2, M3.
11. Search the price-revision results folder for matching result workbooks.
12. Select the latest result file based on period and auto number.
13. Read the Historico_Autos sheet.
14. Extract actual monthly values from Valor_Periodo_Sem_IVA.
15. Group actual values by period.
16. Calculate the actual cumulative value.
17. Merge actual data into the full planned schedule.
18. Fill missing actual monthly values with zero.
19. Recalculate the actual cumulative value over the full planned timeline.
20. Write the Excel workbook.
21. Generate and save the planned-versus-actual chart.

G.7 Period and Label Strategy

The script uses a canonical monthly period format: YYYY-MM

This canonical period is then converted into a shorter display label, such as: Sep-25

Each planned month also receives a milestone-style label: M1, M2, M3, ...

The final chart x-axis combines both labels: Sep-25 and M1

This makes the chart readable while preserving both calendar and contract-sequence information.

G.8 Latest Actual File Selection

The script searches the results folder for files matching the price-revision output pattern. It extracts:

- contract identifier;
- auto number;
- auto period.

The latest file is selected by sorting on:

1. auto period;
2. auto number;
3. filename.

This strategy makes the analysis use the most recent available Historico_Autos table.

G.9 Alignment Strategy

The core alignment strategy is based on the planned schedule. The planned schedule defines the full timeline, and actual data are left-joined into it.

This means:

- planned months are never dropped;
- actual months outside the planned schedule are not added to the chart;
- months without autos have Actual Monthly = 0;
- the actual cumulative line remains flat during months without autos.

This approach produces a stable contract baseline and makes delay or under-execution visually clear.

G.10 Chart Strategy

The chart uses two y-axes:

- the first axis represents monthly values with bars;
- the second axis represents cumulative values with lines.

The chart displays:

- planned monthly bars;
- actual monthly bars;
- planned cumulative line;
- actual cumulative line.

The y-axis is formatted using Portuguese-style Euro formatting, where the decimal separator is a comma and the thousands separator is a space.

G.11 Command-Line Interface

The script supports the following command-line options:

- `--contract-path` or `--contrato` – path to the contract workbook;
- `--results-dir` or `--resultados-dir` – folder containing price-revision result workbooks;
- `--output-dir` – output folder for the Excel workbook and chart image;
- `--out-xlsx` – name of the Excel output file;
- `--out-png` – name of the PNG chart output file.

G.12 Resources Used

G.12.1 Libraries

- `argparse` – command-line configuration.
- `re` – pattern matching for result workbook filenames and period extraction.
- `pathlib.Path` – filesystem path handling.
- `typing.Optional` – type hinting for optional return values.
- `pandas` – Excel reading, dataframe normalisation, period conversion, merging, grouping, cumulative sums, and Excel writing.
- `matplotlib.pyplot` – chart creation.
- `matplotlib.ticker.FuncFormatter` – custom Euro formatting on chart axes.
- `openpyxl` – Excel writing engine used through `pandas.ExcelWriter`.

G.12.2 Data Resources

- Planned contract schedule: `D:\SILVER\Revisao\01_contrato_cronograma_template.xlsx`
- Planned schedule sheet: `Cronograma_Financeiro`
- Price-revision result folder: `D:\SILVER\Revisao\resultados_revisao`
- Actual execution sheet:
- Actual execution sheet: `Historico_Autos`
- Actual monthly value field: `Valor_Periodo_Sem_IVA`
- Output folder: `D:\SILVER\cronograma_financeiro`

G.13 Strategies

G.13.1 Robust Column Detection

The script accepts multiple possible names for the same logical column. This makes the script resilient to small variations in Portuguese, English, accentuation, or naming style.

G.13.2 Canonical Period Normalisation

All date-like values are converted to the canonical YYYY-MM format. This ensures that planned and actual values can be merged reliably by month.

G.13.3 Planned-Schedule Baseline

The planned schedule is used as the master timeline. This preserves the full contract plan even when there are months without autos.

G.13.4 Latest-History Strategy

The script automatically selects the latest price-revision result workbook and reads its Historico_Autos sheet. This avoids manual selection of the most recent execution state.

G.13.5 Cumulative Recalculation

The script recalculates the actual cumulative series after merging actual values into the planned timeline. This ensures that months without autos keep the cumulative actual line flat.

G.13.6 Transparent Output Notes

The Excel workbook includes a Notes sheet that records:

- the planned data source;
- the actual data folder;
- the actual file used;
- the rule used for actual values;
- the alignment rule.

This improves traceability and allows the chart to be audited later.

G.14 Architectural Characterisation

This script is a SILVER reporting module that depends on two upstream sources:

- the contract financial schedule from the price-revision input workbook;
- the auto history generated by the price-revision calculation workflow.

It does not perform price revision itself. Instead, it transforms already available planned and actual financial data into a timeline-aligned analytical report.

G.14.1 SILVER Responsibilities

- normalise the planned financial schedule;
- extract actual execution values from the latest price-revision history;
- align planned and actual data by month;
- calculate cumulative planned and actual financial execution;
- generate a planned-versus-actual Excel dataset;
- generate a planned-versus-actual chart image.

G.14.2 Reporting Responsibilities

- preserve all planned months;
- show missing autos as zero monthly actual execution;
- maintain a flat actual cumulative line during periods with no autos;
- format values as Euro amounts;
- provide a clear chart comparing monthly and cumulative planned-versus-actual execution.

G.15 Recommended Interpretation

The script is best interpreted as the financial schedule monitoring companion to the price-revision module. The price-revision module calculates and stores actual auto history, while this script converts that history into a visual planned-versus-actual schedule evolution.

Its strongest design features are:

- full preservation of the planned contract timeline;
- automatic selection of the latest price-revision history;
- robust monthly period normalisation;

- planned-versus-actual alignment;
- simultaneous monthly and cumulative visualisation;
- traceable Excel output with explicit notes.

Appendix H

AI image risk analiser

H.1 Overview of the Image Risk Analyser

The Image Risk Analyser is a modular image-safety analysis pipeline. Its main entry point is `main.py`, which coordinates configuration loading, backend creation, image discovery, batch processing, report generation, and CSV export.

The uploaded project files show that the system is organised into the following layers:

- **Configuration layer:** runtime settings are loaded from `config.yaml`.
- **Backend layer:** computer-vision and language-model backends are selected through a backend router.
- **Analysis interface layer:** abstract classes define the expected lifecycle and analysis methods for CV and LLM backends.
- **Risk layer:** deterministic rule-based risk factors are derived from CV detections.
- **Fusion layer:** CV and LLM outputs are combined into a final fused assessment.
- **Batch-processing layer:** images are scanned, processed, annotated, logged, and summarised.
- **Reporting layer:** structured report models are built and rendered to HTML.
- **Export layer:** image-analysis events are flattened and exported to CSV.

The system combines computer-vision evidence, language-model interpretation, and deterministic safety logic to produce image-level risk assessments.

H.2 Main Script

D:\SILVER\risk analyser\main.py

H.3 Supporting Modules Analysed

The following linked modules were analysed:

```
core/risk_engine.py
core/analysis_interface.py
core/backend_router.py
core/fusion_engine.py
```

```
pipeline/events_logger.py
pipeline/file_scanner.py
pipeline/parallel_runner.py
pipeline/batch_processor.py
```

```
reporting/report_models.py
reporting/report_builder.py
reporting/html_renderer.py
reporting/csv_exporter.py
```

utils/image_annotator.py
config.yaml

H.4 Purpose

The purpose of the pipeline is to analyse images for potential safety risks. It does this by:

- detecting relevant objects using a CV backend;
- interpreting the image using an LLM or multimodal backend;
- applying deterministic safety rules to CV detections;
- fusing CV and LLM outputs;
- identifying shared findings, branch-specific findings, and conflicts;
- assigning a combined risk level;
- deciding whether manual review is required;
- producing annotated images, structured logs, HTML reports, and CSV exports.

H.5 Configuration

The pipeline is configuration-driven. The configuration defines:

- the CV backend name and version;
- the LLM backend name and version;
- confidence threshold;
- PPE and hazard rules;
- input image folders;
- recursive scanning;
- batch identifier.

This design allows the system to be reconfigured without changing the Python source code.

H.6 Backend Abstraction

H.6.1 Analysis Interfaces

The module `analysis_interface.py` defines abstract interfaces for analysis backends. It defines a shared base class, `AnalysisBackend`, with backend metadata and a `load()` lifecycle method. It then defines separate abstract classes for CV and LLM analysers.

The CV interface requires an `analyze()` method that returns a structured `CVResult`. The LLM interface requires an `analyze()` method that returns a structured `LLMResult`.

H.6.2 Backend Router

The module `backend_router.py` acts as a factory for creating backend instances. It separates backend creation into two families:

- CV analysers;
- LLM analysers.

The router supports dummy CV and LLM backends, a YOLO CV backend, and a Claude LLM backend. This is a backend-abstraction strategy. It allows the pipeline to switch between prototype backends and production-like backends through configuration.

H.7 Risk Engine

The module `risk_engine.py` implements deterministic, rule-based safety interpretation of CV detections. It is intentionally separate from the LLM branch and from the fusion engine.

H.7.1 Supported Risk Logic

The risk engine currently supports:

- missing PPE rules;
- person-object association rules;
- presence-based hazard rules;
- configurable confidence thresholds;
- configurable overlap thresholds;

- configurable severity mapping.

H.7.2 Configured Labels

The engine works with configurable labels, including:

- person;
- helmet;
- vest;
- gloves;
- ladder;
- hole;
- machine.

The default PPE logic checks whether a detected person has associated PPE objects. The default hazard logic generates risks when ladders, holes, or machines are detected.

H.7.3 Risk Factors

The engine can generate risk factors such as:

- missing_helmet;
- missing_vest;
- missing_gloves;
- working_at_height;
- fall_hazard;
- machine_hazard.

Each risk factor has:

- name;
- severity;
- evidence;
- source.

The engine filters detections by confidence threshold and associates PPE with people using bounding-box overlap.

H.8 Parallel Runner

The module `parallel_runner.py` defines the execution container for one image. Despite the name, the current version executes the CV and LLM branches sequentially, while being structured so it can later be upgraded to true parallel execution with threads, async tasks, or workers.

The runner performs the following sequence:

1. run the CV analyser;
2. enrich the CV result with deterministic risk flags if a risk engine exists;
3. run the LLM analyser;
4. fuse both outputs using `FusionEngine`;
5. return a `ParallelAnalysisResult`.

The result contains:

- `cv_result`;
- `llm_result`;
- `fusion_result`.

H.9 Fusion Engine

The module `fusion_engine.py` combines CV and LLM analysis results into a final fused assessment. Its deterministic logic identifies shared findings, CV-only findings, LLM-only findings, simple conflicts, final risk level, and manual-review requirement.

H.9.1 Finding Normalisation

The fusion engine normalises CV findings by combining:

- detected object labels;
- CV scene flags.

It normalises LLM findings from LLM risk-factor names.

H.9.2 Conflict Detection

The current conflict logic checks selected disagreement patterns, such as:

- CV detects a helmet, but the LLM suggests that the helmet is missing, unclear, or not visible;
- CV detects a ladder, but the LLM does not mention ladder-related contextual risk.

These conflicts are used to trigger manual review.

H.9.3 Risk Fusion

The fusion engine takes the highest LLM risk severity as a baseline and can promote the final risk level when both branches share high-signal findings, such as:

- ladder_detected;
- hole_detected;
- machine_detected;
- working_at_height;
- fall_hazard;
- missing_helmet.

Manual review is required when conflicts exist or when the combined risk level is high or critical.

H.10 File Scanning

The module `file_scanner.py` scans local filesystems for supported image files. It supports recursive scanning, single-file inputs, multiple root paths, extension filtering, and deduplication.

The default supported extensions are:

- .jpg;
- .jpeg;
- .png;
- .bmp;
- .tif;
- .tiff;
- .webp.

The design can later be extended to support PDF image extraction, cloud storage, metadata filters, and modified-date filters.

H.11 Batch Processor

The module `batch_processor.py` orchestrates end-to-end processing of a batch of images. Its responsibilities include iterating through image paths, loading image data when a loader is available, running the dual analysis pipeline, generating annotated previews, creating events, logging events, and collecting results.

For each image, the processor:

1. loads the image if an image loader is configured;
2. runs the `ParallelRunner`;
3. generates an annotated image if an annotator is configured;
4. creates an `ImageAnalysisEvent`;
5. logs the event if an event logger exists;
6. adds the event to the output list;
7. records failures without stopping the whole batch.

The returned batch summary includes:

- batch identifier;
- start and finish timestamps;
- total images;
- number of succeeded items;
- number of failed items;
- output events;

- failed item details.

This makes the batch pipeline fault-tolerant: one failed image does not stop the rest of the batch.

H.12 Image Annotation

The module `image_annotator.py` uses OpenCV to draw bounding boxes and labels on analysed images. It reads the image, loops through detections, draws rectangles, writes labels with confidence scores, and saves the annotated image to:

`reports/images`

The method returns the path of the annotated image.

H.13 Event Logging

The module `events_logger.py` logs structured image-analysis events. It writes:

- full JSONL event records for operational and debug logging;
- flattened Parquet rows for analytics, reporting, or future data-lake integration.

The design goals are traceability and analytics-readiness.

H.13.1 JSONL Logging

Each full event is serialised to JSON and written as one line in:

`logs/events_log.jsonl`

H.13.2 Parquet Analytics

Each event is flattened into an analytics row containing:

- event identifier;
- timestamp;
- image path;
- event type;
- CV backend and latency;
- number of CV detections;
- LLM backend and latency;
- number of observations and risk factors;
- fused risk level;
- manual-review flag;
- shared, CV-only, and LLM-only finding counts;
- conflict count;
- final conclusion;
- recommended action.

The flattened rows are stored in:

`logs/events_log.parquet`

H.14 Report Models

The module `report_models.py` defines the structured report schema using Pydantic. It includes models for:

- detections;
- risk factors;
- conflicts;
- image-level report sections;
- batch summary statistics;
- full batch report.

The report schema supports risk levels of:

- low;
- medium;
- high;
- critical.

The image-level report section stores both raw and fused information, including CV detections, scene flags, LLM observations, LLM risk factors, uncertainty notes, shared findings, branch-specific findings, and conflicts.

H.15 Report Builder

The module `report_builder.py` converts raw image-analysis events into report-friendly Pydantic objects. It is explicitly format-agnostic: it does not contain HTML, Markdown, or PDF rendering logic. It builds:

- a batch-level summary;
- one image-level section per analysed image;
- report detections;
- report risk factors;
- report conflicts.

The batch summary counts:

- total images;
- low-risk images;
- medium-risk images;
- high-risk images;
- critical-risk images;
- images requiring manual review;
- images with conflicts.

H.16 HTML Renderer

The module `html_renderer.py` converts a structured `BatchReport` into a styled HTML document. It is presentation-only: it does not compute report logic, aggregate events, or change analytical results.

The HTML report includes:

- header with batch metadata;
- batch summary cards;
- risk distribution section;
- image-level findings;
- annotated image previews when available;
- CV findings;
- LLM observations and risk factors;
- fusion conclusions;
- conflicts;
- recommended actions.

The renderer also escapes text for safe HTML output and defines the visual styling directly in the generated HTML.

H.17 CSV Exporter

The module `csv_exporter.py` exports one row per analysed image. The CSV includes image-level, CV, LLM, and fusion fields.

The exported columns include:

- image path;
- timestamp;
- CV backend;
- CV latency;
- CV detections;

- CV confidence scores;
- CV scene flags;
- CV summary;
- LLM backend;
- LLM latency;
- LLM observations;
- LLM risk factors;
- LLM risk severities;
- LLM risk evidence;
- LLM narrative summary;
- LLM recommended action;
- LLM uncertainty notes;
- fused risk level;
- shared findings;
- CV-only findings;
- LLM-only findings;
- conflicts;
- final conclusion;
- recommended action;
- manual-review flag.

CSV files are written to:

reports/csv

and named using the batch identifier and a UTC timestamp.

H.18 Main Processing Flow

The complete process can be summarised as follows:

1. Load config.yaml.
2. Create the configured CV backend through BackendRouter.
3. Create the configured LLM backend through BackendRouter.
4. Load both backend resources.
5. Create the RiskEngine.
6. Create the ParallelRunner.
7. Create the EventLogger.
8. Create the ImageAnnotator.
9. Create the BatchProcessor.
10. Create the FileScanner.
11. Scan configured input paths for images.
12. If no images are found, exit cleanly.
13. For each image, run CV analysis.
14. Enrich CV output with deterministic risk factors.
15. Run LLM analysis.
16. Fuse CV and LLM outputs.
17. Generate annotated image preview.
18. Create a structured ImageAnalysisEvent.
19. Log the event to JSONL and Parquet.
20. Collect batch outputs and failures.
21. Build a structured batch report.
22. Render the HTML report.
23. Save and open reports/latest_report.html.
24. Export flattened events to CSV.

H.19 Resources Used

H.19.1 External Libraries

- `yaml` – reading pipeline configuration.
- `webbrowser` – opening the generated HTML report.
- `pathlib.Path` – path handling.
- `datetime` – timestamps for events, batches, reports, and exports.
- `uuid` – event identifiers.
- `json` – JSONL serialisation and flattened JSON fields.
- `pandas` – Parquet analytics logging.
- `csv` – CSV report export.
- `html.escape` – safe HTML rendering.
- `pydantic` – structured report models.
- `cv2` / `OpenCV` – image annotation with bounding boxes and labels.
- `dataclasses` – lightweight result and summary containers.
- `abc` – abstract backend interfaces.
- `typing` – type hints and structured interfaces.

H.19.2 Internal Modules

- `core.analysis_interface` – abstract backend lifecycle and analyse methods.
- `core.backend_router` – CV and LLM backend factory.
- `core.risk_engine` – deterministic CV-derived safety rules.
- `core.fusion_engine` – CV/LLM fusion and final risk assessment.
- `pipeline.file_scanner` – local image discovery.
- `pipeline.parallel_runner` – single-image dual-branch execution.
- `pipeline.batch_processor` – batch orchestration.
- `pipeline.events_logger` – JSONL and Parquet event logging.
- `reporting.report_models` – Pydantic report schema.
- `reporting.report_builder` – conversion from events to report objects.
- `reporting.html_renderer` – HTML report rendering.
- `reporting.csv_exporter` – tabular CSV export.
- `utils.image_annotator` – annotated image generation.

H.20 Strategies

H.20.1 Configuration-Driven Execution

The pipeline is controlled by `config.yaml`. This enables backend selection, input-folder selection, rule configuration, threshold configuration, and batch identification without changing source code.

H.20.2 Backend Abstraction

CV and LLM implementations are hidden behind abstract interfaces and created through a router. This allows dummy, YOLO, Claude, or future backends to be plugged into the same pipeline.

H.20.3 Rule-Based CV Risk Enrichment

The CV result is enriched with deterministic risks before fusion. This provides transparent and auditable safety logic for PPE and presence-based hazards.

H.20.4 Dual-Branch Analysis

Each image is analysed through two branches:

- CV branch for structured visual evidence;
- LLM branch for semantic observations and contextual interpretation.

H.20.5 Deterministic Fusion

The fusion engine uses rule-based logic to compare CV and LLM findings. It identifies shared findings, branch-only findings, conflicts, combined risk level, and manual-review requirements.

H.20.6 Manual Review Strategy

Manual review is triggered when:

- CV and LLM branches disagree;
- the fused risk level is high;
- the fused risk level is critical.

H.20.7 Fault-Tolerant Batch Processing

Image-level failures are caught and recorded in the batch summary. A failure in one image does not stop the whole batch.

H.20.8 Traceability Strategy

Each successful image produces a structured event. Events are reused for:

- operational JSONL logging;
- Parquet analytics logging;
- HTML report generation;
- CSV export.

H.20.9 Human-in-the-Loop Reporting

The HTML report is automatically opened after generation. This supports direct inspection by a human reviewer or safety analyst.

H.20.10 Multi-Format Output

The same event stream supports several output formats:

- annotated image previews;
- JSONL logs;
- Parquet analytics rows;
- HTML report;
- CSV table.

H.21 Architectural Characterisation

The system is best characterised as a modular local image-risk analysis platform.

H.21.1 Strengths

- clear separation between orchestration, analysis, fusion, batch execution, logging, reporting, and export;
- configurable backend selection;
- auditable deterministic risk rules;
- combined CV and LLM reasoning;
- structured event model;
- support for both human-readable and machine-readable outputs;
- extensibility for new backends, storage systems, report formats, and scanning sources.

H.21.2 Current Limitations

- ParallelRunner is currently sequential despite its name;
- file scanning is local-filesystem based;
- conflict detection is rule-based and limited to predefined patterns;
- event logging rewrites the full Parquet file on append, which is acceptable for prototypes but may need optimisation for large datasets;
- detailed backend behaviour depends on the model implementation files, such as YOLO and Claude-specific modules.

H.22 Recommended Interpretation

The Image Risk Analyser should be interpreted as a SILVER-layer analytical pipeline for construction or site-safety image review. It is designed to convert raw images into structured, reviewable safety findings.

Its strongest design pattern is the use of a common event stream. Once an image is analysed, the same event is used for logging, analytics, HTML reporting, and CSV export. This creates consistency across outputs and improves traceability.

The system is also well-positioned for future extension. The backend interfaces and router allow new CV or LLM implementations to be added, while the report and export layers can consume the same event model without requiring changes to the analysis layer.

Appendix I

API-Based LLM-Assisted Monthly Reporting

I.1 Overview of the Monthly Master Pipeline

The monthly master pipeline implements a GOLD-level reporting workflow. Its purpose is to collect relevant outputs already produced by the different analytical modules, summarise those outputs into a structured evidence package, select relevant visual artefacts, obtain or generate comments for those artefacts, send the evidence to Claude through the Anthropic API, and save the resulting monthly report in both Markdown and HTML formats.

The script does not replace the existing analytical pipelines. Instead, it acts as a reporting layer above them. The existing workflows continue to produce their own outputs, such as Excel workbooks, CSV tables, text summaries, HTML dashboards, PNG charts, JSON files, and Parquet logs. The monthly master pipeline gathers those outputs, selects recent and relevant files, creates compact summaries, inserts selected figures into the report, and asks the LLM to produce a management-oriented monthly narrative.

I.2 Purpose

The script automates the production of a monthly construction supervision report from existing project outputs. Most important tasks are:

- loads the monthly report configuration;
- loads the Claude API key from a local .env file;
- scans configured output folders from previous pipelines;
- collects recent files from each technical area;
- selects the most relevant files per area;
- summarises Excel, CSV, and text files;
- selects specific PNG figures for inclusion in the report;
- selects the most recent meteorological weekly, monthly, quarterly, and yearly dashboards;
- copies selected images into the monthly report output folder;
- generates comments for figures using supporting data when available;
- uses image-based Claude interpretation as a fallback when supporting data are not available;
- builds a structured evidence package;
- saves the evidence package as JSON;
- builds a prompt for the LLM;
- sends the evidence package to Claude through the Anthropic API;
- receives a Markdown monthly report;
- appends a figure gallery with image commentary;
- saves the Markdown report;
- converts the Markdown report to HTML;
- saves the final HTML report with embedded figures.

I.3 Figure Selection and Commentary

The improved version of the pipeline includes a figure-selection stage. This stage copies selected PNG images into:

D:\monthly_report\outputs\images

The selected figures include Outlook activity plots, Outlook heatmaps, recipient-ranking charts, meteorological dashboards, daily report activity figures, resource analysis figures, financial schedule evolution charts, and image-risk analyser outputs.

For some figures, especially those generated from structured data, the preferred strategy is to comment on the underlying data rather than relying only on visual interpretation of the PNG file. This applies particularly to:

- Outlook activity by hour;
- Outlook activity by weekday;
- Outlook activity by day of month;
- Outlook activity by month;
- sent and received heatmaps;
- top recipient rankings;
- meteorological weekly, monthly, quarterly, and yearly dashboards.

This data-backed commentary strategy reduces the volume of image data sent to the LLM, decreases the probability of API rate-limit errors, and improves consistency between the figure and the generated interpretation.

I.4 Main Outputs

The script produces the following main outputs:

- evidence package: outputs/evidence_YYYY-MM.json
- selected image list: outputs/selected_images_YYYY-MM.json
- image comments: outputs/image_comments_YYYY-MM.json
- copied report figures: outputs/images
- Markdown monthly report: outputs/monthly_report_YYYY-MM.md
- HTML monthly report with embedded figures: outputs/monthly_report_YYYY-MM.html

I.5 Updated Strategies

I.5.1 Figure-Inclusion Strategy

The monthly report is no longer limited to text and file metadata. Selected figures are copied into the report output folder and inserted into the Markdown and HTML reports. This allows the report to combine written synthesis with visual evidence.

I.5.2 Data-Backed Commentary Strategy

Whenever possible, figure commentary is generated from the structured data used to create the plots. This is preferable to interpreting the PNG alone, because it avoids unnecessary image-token consumption and makes the interpretation more directly linked to tabular evidence.

I.5.3 Image-Based Fallback Strategy

When supporting data are not available, the script can fall back to image-based LLM commentary. This is useful for figures that cannot be easily reconstructed from tabular data, such as annotated safety images or selected visual dashboards.

I.5.4 Rate-Limit Reduction Strategy

The use of data-backed comments for charts generated from structured data reduces the number and size of image requests sent to the API. This helps avoid rate-limit errors and makes the pipeline more stable.

I.6 Current Limitations

The improved pipeline still has limitations:

- it does not yet run all upstream scripts automatically;
- data-backed commentary depends on the availability of the underlying data files;
- some figures may still require image-based interpretation;
- report comments are not yet linked to exact source rows or chart values;
- Word or PDF export is not yet implemented;
- human review remains necessary before using the report for formal project decisions.

Appendix J

AI Image Risk Analyser Report

This appendix presents an example output generated by the AI Image Risk Analyser. The report corresponds to a batch analysis of construction-site photographs using a dual-pipeline approach that combines computer vision evidence, large language model image interpretation, and fused risk conclusions. The first page summarises the batch-level results, including the number of analysed images, the distribution of risk levels, and the number of cases requiring manual review. The following pages present one image-level report per photograph, including the annotated image, computer vision findings, LLM-based observations, identified risk factors, recommended actions, and the final fused risk classification.

The report is included in full to illustrate the practical output of the prototype and to show how image-based evidence can be transformed into a structured safety-assessment document. The results should be interpreted as an assistive analytical output and not as a substitute for formal site inspection, professional safety assessment, or contractual decision-making.

SafeSite-Vision Batch Report

Dual-pipeline image safety analysis report combining CV evidence, LLM interpretation, and fused conclusions.

Batch ID test_batch_001	Generated at (UTC) 2026-06-13T14:34:31.924198	Total Images 7
-----------------------------------	---	--------------------------

Batch Summary

Low Risk 0	Medium Risk 1	High Risk 2
Critical Risk 4	Manual Review Required 6	Images With Conflicts 0

Risk Distribution

- Low 0 - Medium 1 - High 2 - Critical 4
--



CV Findings

Backend	yo1o yolov8n
Detections	0 None
Scene Flags	None
CV Summary	YOLO detected no objects above the configured confidence threshold.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	- Active excavation and material handling operation underway on roadside construction site - Multiple heavy equipment units operating in close proximity to personnel and materials - Loose materials and debris scattered throughout the immediate work area +2 more in the interactive HTML report.
Risk Factors	- unsecured_excavated_material High Large excavated logs and materials are stacked and positioned near the work area without visible securing straps,... - trip_hazard_materials High Scattered construction materials, logs, and debris are present on the ground surface creating potential trip hazar... - heavy_machinery_proximity High Excavator and loader are operating near workers and materials without visible exclusion zones or spotters managing... - inadequate_personal_protective_equipment Medium Workers visible in the scene do not appear to be wearing visible high-visibility vests or hard hats consistently +2 additional risk factor(s) in the interactive HTML report.
Narrative	This construction site demonstrates multiple significant safety concerns including unsecured excavated materials, inadequate worker protective equipment, and heavy machinery operating without visible safety controls or exclusion zones. The scattered debris, lack of perimeter barriers, and proximity of personnel to operating equipment create a high-risk environment for incidents.
Action	Immediately establish and enforce an exclusion zone around heavy machinery operations, ensure all personnel wear required PPE including hard hats and high-visibility vests, and secure all excavated materials with proper containment systems.



CV Findings

Backend	yo1o yo1ov8n
Detections	1 person 0.88
Scene Flags	missing_helmet worker_present
CV Summary	YOLO detected: 1 person. Derived CV risks: missing_helmet.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	- Active excavation trench approximately 2+ meters deep with unprotected walls - Orange excavator actively working in trench with worker present nearby - Dump truck and additional construction vehicles positioned around the work area +4 more in the interactive HTML report.
Risk Factors	- unsupported_trench_walls Critical Deep excavation trench with no visible shoring, bracing, or support systems. Walls appear to be unprotected soil t... - inadequate_worker_protection High Worker visible in high-visibility vest at ground level near the trench with no apparent hard hat visible and stand... - equipment_proximity_hazard High Active excavator operating in trench with worker present nearby. Limited visibility from operator's position creat... - trench_depth_hazard High Trench appears to be approximately 2+ meters deep with no ladder, ramp, or safe means of egress visible +2 additional risk factor(s) in the interactive HTML report.
Narrative	This construction site presents critical safety violations including unsupported trench walls without required shoring systems, inadequate worker protection measures, and dangerous proximity of personnel to active excavation equipment. The combination of deep excavation, unprotected walls, and worker proximity to mobile equipment creates an extremely hazardous environment with high risk of serious injury or fatality.
Action	Immediately cease all work operations and implement proper trench shoring/support systems, establish exclusion zones around equipment, ensure all personnel wear required PPE including hard hats, and verify safe egress points before resuming operations
Uncertainty	Cannot definitively confirm if worker is wearing hard



CV Findings

Backend	yo1o yolov8n
Detections	1 person 0.87
Scene Flags	missing_helmet worker_present
CV Summary	YOLO detected: 1 person. Derived CV risks: missing_helmet.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	• Deep trench excavation in progress with vertical, unsupported walls presenting significant str... • Workers are positioned at and near trench edges without visible protective systems • Heavy equipment (excavator) is actively operating in close proximity to the work area • +2 more in the interactive HTML report.
Risk Factors	• unsupported_trench_walls Critical Deep excavation with vertical walls showing no shoring, bracing, or support systems. Soil appears loose and unstab... • inadequate_fall_protection High Worker on right side of trench appears to be standing near edge without visible fall protection equipment or harne... • trench_cave_in_hazard Critical Excavated trench shows signs of soil instability with loose material and no protective barriers or sloped sides to... • excavator_proximity_hazard High Active excavator equipment positioned directly above/near trench with workers present below in immediate danger zo... • +2 additional risk factor(s) in the interactive HTML report.
Narrative	This construction site presents critical safety violations related to trench work. The unsupported excavation with vertical walls, absence of shoring or bracing systems, and workers positioned in immediate proximity to collapse hazards create extreme risk of trench cave-in and serious injury. Excavator operations near active workers compound the danger significantly.
Action	Immediately cease all work in the trench, evacuate all personnel, and implement proper trench support systems (shoring, shielding, or sloped walls per OSHA standards) before any resumption of work
Uncertainty	• Cannot clearly determine if proper rescue equipment and trained personnel are on standby • Unclear if soil engineering assessment has been



CV Findings

Backend	yo1o yo1ov8n
Detections	1 person 0.87
Scene Flags	missing_helmet worker_present
CV Summary	YOLO detected: 1 person. Derived CV risks: missing_helmet.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	- Worker is inside an excavated trench with multiple utilities present - Hard hat and high-visibility vest are being worn - Loose soil walls show no apparent shoring or protective systems +4 more in the interactive HTML report.
Risk Factors	- inadequate_ground_support High Worker is positioned in an excavated trench with loose soil walls above and around them. The exposed earth shows n... - trench_depth_hazard High The trench appears to be at least 4-5 feet deep based on the worker's position. No visible ladder, ramp, or safe m... - utility_strike_risk Critical Multiple pipes and conduits are visible in and around the trench (black pipe, red hose, blue conduit). No visible... - confined_space_proximity Medium The trench appears narrow with the worker in close proximity to walls and existing utility lines, limiting movemen... +2 additional risk factor(s) in the interactive HTML report.
Narrative	This trench work presents critical safety violations including inadequate ground support systems, unprotected utility work, and insufficient fall/egress protection. The presence of multiple utilities combined with an unsupported trench creates a high-risk environment for both cave-in and utility strike incidents. Immediate work stoppage and implementation of proper safety measures is required.
Action	Stop all work immediately and implement proper trench shoring/bracing system, verify utility locations with marking, establish safe means of egress, and ensure continuous on-site supervision before resuming operations.
Uncertainty	- Cannot determine exact trench depth from camera angle - Unclear if utilities have been formally marked or



CV Findings

Backend	yo1o yo1ov8n
Detections	2 person 0.78 person 0.72
Scene Flags	missing_helmet worker_present
CV Summary	YOLO detected: 2 person. Derived CV risks: missing_helmet, missing_helmet.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	- Construction crew is actively dumping material from a truck directly onto a roadway - Multiple workers in high-visibility clothing are positioned near the active dumping operation - Traffic control appears minimal with only one visible traffic cone for a large work zone +3 more in the interactive HTML report.
Risk Factors	- inadequate_traffic_control High Single traffic cone visible for a major dumping operation on what appears to be an active roadway with limited war... - unsecured_load_discharge High Truck actively dumping material onto roadway with minimal containment; material spreading across driving surface - worker_proximity_to_vehicle High Multiple workers in orange vests standing very close to active dumping truck without visible barriers or safe dist... - inadequate_road_barriers High Pink/red barriers present but appear insufficient in number and placement to protect work zone from vehicle traffic +3 additional risk factor(s) in the interactive HTML report.
Narrative	This construction site presents multiple critical safety deficiencies. The active dumping operation on the roadway has inadequate traffic control, insufficient worker spacing from equipment, and uncontrolled material discharge creating hazards. The combination of active vehicle operations, worker proximity, and minimal protective barriers creates a high-risk environment.
Action	Immediately halt the dumping operation, establish a properly marked exclusion zone with adequate barriers (minimum 50m), position a traffic control person with signage on both approaches, and require all workers to maintain safe distance from the operating truck and excavator.
Uncertainty	Unable to determine if traffic is actively diverted or if this is a closed roadway



CV Findings

Backend	yo1o yo1ov8n
Detections	1 person 0.78
Scene Flags	missing_helmet worker_present
CV Summary	YOLO detected: 1 person. Derived CV risks: missing_helmet.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	- Single worker operating compacting equipment in a deep trench - Trench appears to be utility-related based on configuration and location - Worker is wearing standard construction PPE but insufficient for this operation +2 more in the interactive HTML report.
Risk Factors	- inadequate_trench_shoring Critical Deep trench visible with no apparent shoring, bracing, or protective systems in place. Trench walls appear unsuppo... - confined_space_hazard High Worker operating compactor in narrow trench that could become a confined space with limited egress and ventilation... - cave_in_risk Critical Trench depth appears to exceed safe depth without protective systems. Loose soil composition visible on trench wal... - inadequate_personal_protective_equipment High Worker wearing hard hat and high-visibility vest, but no respiratory protection, face shield, or lower body protec... +2 additional risk factor(s) in the interactive HTML report.
Narrative	This site presents critical safety violations including an unshored trench of significant depth operated by an inadequately protected worker. Multiple hazards including cave-in risk, confined space exposure, and inadequate PPE create an extremely dangerous work environment that violates OSHA standards for excavation safety.
Action	Immediately stop work, evacuate worker from trench, and implement proper trench protection systems (shoring, sloping, or protective devices) before resuming operations.
Uncertainty	- Exact trench depth cannot be precisely determined but appears to exceed 4-5 feet - Cannot confirm if underground utilities have been properly marked or located



CV Findings

Backend	yo1o yolov8n
Detections	0 None
Scene Flags	None
CV Summary	YOLO detected no objects above the configured confidence threshold.

LLM Findings

Backend	claude claude-haiku-4-5-20251001
Observations	- This is a soil analysis instrument (Leitura do Solo - 'Soil Reading') displaying compaction, h... - The device shows soil readings with acceptable humidity levels (9.1%) and compaction data (78.... - Equipment is mounted on what appears to be temporary construction site infrastructure +2 more in the interactive HTML report.
Risk Factors	- exposed_device_mounting Medium Electronic measurement device (soil moisture/compaction meter) is mounted on what appears to be an exterior wall o... - cable_management_hazard Low Visible cable or connection point at bottom of device (reference to 'Cat: 0 0 0 0 0 0') appears to have exposed...
Narrative	A soil analysis monitoring device is installed on a construction site showing real-time soil compaction and moisture data. While the device itself appears functional and readings are within acceptable parameters, the mounting configuration exposes electronic components to potential environmental damage. Overall construction site safety status regarding this specific equipment is manageable with minor weatherproofing improveme...
Action	Improve device weatherproofing by installing a protective enclosure or awning to shield the electronics from direct rain, dust, and UV exposure to prevent equipment failure and data loss.
Uncertainty	- Cannot fully assess the structural integrity of the mounting system from this angle - Unclear if proper grounding and electrical safety measures are in place for the device +1 more in the interactive HTML report.

Fusion Result

Combined Risk	Medium
---------------	--------

Appendix K

Monthly Construction Supervision Report – 2026-05

This appendix presents the monthly construction supervision report generated for May 2026 through the API-based LLM-assisted evidence synthesis workflow. The report consolidates outputs produced by the different analytical modules developed in this work, including meteorological analysis, daily supervision records, financial schedule evolution, price revision outputs, Outlook correspondence indicators, and image-based risk analysis.

The purpose of including the report as an appendix is to document the practical result of the monthly master pipeline. The report should be interpreted as a reviewable draft generated from structured evidence and selected analytical outputs, not as an autonomous decision-making document. Its contents remain subject to technical validation by the supervision team and other responsible project stakeholders.

The following pages reproduce the generated monthly report in its original PDF format.

Construction Supervision Monthly Report

Report Period: 2026-05
Generated: 14 June 2026 21:16

CONSTRUCTION SUPERVISION MONTHLY REPORT

May 2026

1. EXECUTIVE SUMMARY

This report covers the construction supervision period for May 2026. The project demonstrates continued progression through structural and finishing phases, with significant activity concentration evident from April onwards. Key findings indicate:

- Actual resource deployment now exceeds contractor proposals as of May 2026
- Substantial cumulative price revisions (approximately €3,000,000) have been recorded since project commencement
- Significant divergence exists between planned and actual financial expenditure trajectories
- Meteorological conditions remained predominantly moderate throughout the reporting period
- Site safety and housekeeping concerns require supervisory attention

2. WEATHER AND SITE CONDITIONS

Meteorological Overview

Data from three monitoring stations (S1–Carrasquilla, S2–Minscovo, S3–Miravalles) spanning 13.71–23.09 km from the project site indicates:

May 2026 Conditions:

Temperature conditions remained predominantly moderate (73.6%), with a gradual warming trend evident from April through mid-June. High temperatures (>25 °C) occurred primarily in the final fortnight of the reporting period.

Data Quality:

The meteorological dataset comprises robust hourly records with minimal gaps, providing adequate coverage for construction supervision purposes. Station spatial separation should be considered when applying findings to specific site locations.

3. DAILY REPORTS AND MAIN ACTIVITIES

Activity Distribution

Daily reporting data reveals significant activity concentration in latter months, with project progression evident through structural and finishing phases:

- **Excavation activities** show marked escalation in May 2026, with 18 reports recorded, indicating intensive earthworks during this period
- **Sustained activities** including "Escavação" (Excavation) and "Cofragem" (Formwork) demonstrate consistent reporting throughout the monitoring period
- **Early-stage activities** (July–August 2025) showed minimal reporting, with only demolition and testing activities recorded

Reporting Consistency

Significant variability exists in reporting consistency across work categories. Some categories demonstrate sustained activity levels, whilst others show sporadic reporting patterns requiring supervisory review.

4. PERSONNEL AND EQUIPMENT RESOURCES

Resource Deployment Analysis

Personnel:

Actual personnel deployment remains significantly below contractor proposals throughout the monitoring period. Proposed personnel peaked around mid-2025; however, actual personnel remained substantially lower, suggesting either phased mobilisation or underutilisation of planned workforce.

Equipment Utilisation:

A critical concern emerges: actual equipment deployment substantially exceeds proposed levels from mid-2025 onwards. By May 2026, all three resource metrics (personnel, equipment, and total resources) have crossed into positive variance territory, indicating actual deployment now exceeds contractor proposals.

Key Observation:

The upward trend in resource variances from mid-2025 onwards suggests a significant shift in project resourcing strategy, with equipment-heavy deployment compensating for lower-than-proposed personnel levels.

5. FINANCIAL SCHEDULE EVOLUTION

Planned versus Actual Expenditure

Significant divergence exists between planned and actual expenditure trajectories:

- **Planned cumulative curve:** Demonstrates steady acceleration through September 2025, reaching approximately €1,000,000
- **Actual cumulative line:** Plateaus substantially earlier, stabilising around €350,000–€375,000 from August onwards

Critical Anomaly

A pronounced gap emerges between planned and actual expenditure from August 2025 onwards, indicating substantial underspend relative to the financial schedule. This divergence requires investigation to determine whether it reflects project delays, revised payment schedules, or other financial adjustments.

6. PRICE REVISION STATUS

Cumulative Price Revision

The accumulated price revision demonstrates a steep upward trajectory from February through July 2025, thereafter stabilising at approximately €3,000,000. This substantial cumulative increase suggests either inadequate initial budgeting or substantial scope changes during the project's early phases.

Peak Volatility:

July 2025 recorded the highest revision volatility, with significant cost adjustments during this period.

Current Status:

Price revision has stabilised since July 2025, indicating that major cost adjustments have been resolved and incorporated into the current budget baseline.

7. OUTLOOK AND CORRESPONDENCE INDICATORS

Email Communication Patterns

Analysis of Outlook correspondence reveals distinct communication patterns:

Weekday Activity:

Weekday activity (Monday through Friday) remains consistently elevated, with Thursday recording the highest volume. Weekend activity drops dramatically, with Saturday and Sunday showing approximately 80–90% lower engagement than weekday averages.

Temporal Patterns:

- Monday mornings (08:00 hour) show exceptional outgoing email activity, reaching approximately 45 messages, suggesting routine weekly briefings or project kickoffs
- Peak received email activity concentrates during mid-morning hours (10:00–11:00) and midday (12:00–13:00)
- Monday and Wednesday show the highest received email volumes

Daily Volatility:

Received emails range from approximately 270 to 1,300 messages daily, with notable peaks occurring around days 11–12 of the month, suggesting concentrated incoming communication periods.

Data Quality Concern

Critical Issue: Email activity is recorded only for months 1–5, with months 6–12 showing negligible or zero values. This pattern is inconsistent with typical construction project communication and suggests either incomplete data collection, a system migration, or a reporting error. **Source output should be checked.**

8. IMAGE RISK ANALYSIS

Site Safety and Housekeeping

Image analysis reveals several supervision points requiring immediate attention:

Excavation and Debris Management:

Loose excavated material and debris are scattered across the working surface, creating trip hazards and potential foreign object damage (FOD) risks.

Barrier Positioning:

Pink warning barriers appear inadequately positioned relative to excavation works and mobile plant operating in proximity.

Plant and Traffic Management:

Heavy machinery (excavators and mobile plant) are operating in areas requiring enhanced traffic management and coordination protocols.

Recommendations:

Enhanced site housekeeping protocols and barrier repositioning are required to mitigate identified hazards.

9. MAIN RISKS, ALERTS, AND ANOMALIES

Critical Issues

- **Financial Divergence:**** Substantial gap between planned and actual expenditure from August 2025 onwards requires investigation and corrective action planning
- **Resource Variance:**** Equipment deployment significantly exceeds proposals whilst personnel remain below proposed levels—this imbalance warrants review of project resourcing strategy
- **Price Revision Magnitude:**** Cumulative price revision of approximately €3,000,000 represents a significant cost impact requiring root-cause analysis
- **Data Integrity:**** Outlook correspondence data shows anomalous pattern (months 6–12 negligible) requiring verification
- **Site Safety:**** Loose debris, inadequate barrier positioning, and plant proximity hazards identified in image analysis

Alerts

- Personnel deployment remains substantially below contractor proposals
- Equipment utilisation significantly exceeds proposals
- Site housekeeping and barrier management require immediate supervisory intervention

10. RECOMMENDED ACTIONS FOR NEXT MONTH

- **Financial Schedule Review:**** Conduct detailed investigation into the planned versus actual expenditure divergence; determine causes and implement corrective measures to align actual spending with financial schedule
- **Resource Rebalancing:**** Review the equipment-heavy, personnel-light deployment strategy; assess whether current resource allocation optimises project delivery and cost-effectiveness
- **Data Verification:**** Check Outlook correspondence data source output to clarify the anomalous pattern in months 6–12; confirm data integrity and completeness
- **Site Safety Intervention:**** Implement immediate corrective actions addressing debris management, barrier repositioning, and plant traffic protocols identified in image risk analysis
- **Price Revision Analysis:**** Conduct root-cause analysis of the €3,000,000 cumulative price revision; document scope changes and cost drivers for future reference
- **Communication Workload Review:**** Assess whether concentrated mid-morning and midday email peaks (10:00–13:00) indicate workload management issues requiring process optimisation

11. OUTPUTS AND EVIDENCE USED

Data Sources

Meteorological Data:

- Path: D:\SILVER\meteo
- Files: 135 total (120 images, 15 CSV files)
- Latest update: 14 June 2026
- Coverage: Hourly records from three monitoring stations; annual, quarterly, monthly, and weekly dashboards analysed

Daily Reports:

- SILVER path: D:\SILVER\Relat_diario\outputs (13 files, 12 XLSX)
- GOLD path: D:\GOLD\Relat_diario (15 files, 6 images, 4 XLSX)
- Latest update: 29 May 2026
- Coverage: Activity distribution by month and week; resource variance analysis

Price Revision:

- Path: D:\SILVER\Revisao
- Files: 140 total (20 XLSX, 4 images)
- Latest update: 13 June 2026
- Coverage: Cumulative price revision evolution and historical records

Financial Schedule:

- Path: D:\SILVER\cronograma_financeiro
- Files: 16 total (4 XLSX, 5 images)
- Latest update: 13 June 2026
- Coverage: Planned versus actual expenditure comparison

Outlook Correspondence:

- Path: D:\GOLD\outskrape
- Files: 78 total (27 images, 4 XLSX, 29 CSV)
- Latest update: 12 June 2026
- Coverage: Activity by hour, weekday, day of month, and month-year; sent and received email heatmaps

Image Risk Analysis:

- Path: D:\SILVER\risk_analyser\reports

Report Prepared: May 2026

Data Current As Of: 14 June 2026

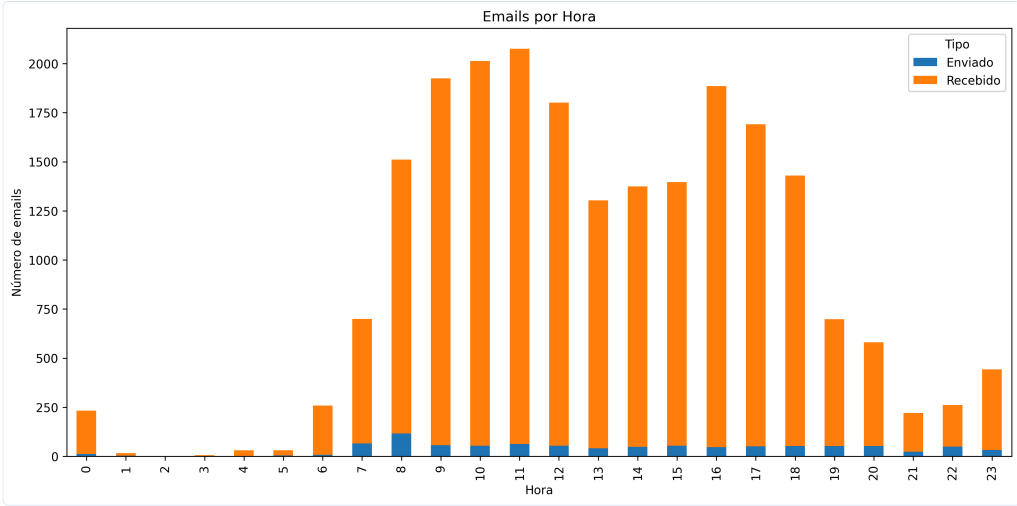
Outputs and Evidence Inventory

Area	Exists	Files	Images	XLSX	CSV	Newest file
meteo	True	135	120	0	15	D:\SILVER\meteo\output_plots\all_data\meteo-all-data-Y-2026-01-01-2026-12-31.png
daily_reports_silver	True	13	0	12	0	D:\SILVER\Relat_diario\outputs\resumo_processamento_RDF.txt
daily_reports_gold	True	15	6	4	0	D:\GOLD\Relat_diario\gold_RDF_keyword_selector_maps_EDITADO.xlsx
price_revision	True	140	4	20	0	D:\SILVER\Revisao\04_historico_autos.xlsx
financial_schedule	True	16	5	4	0	D:\SILVER\cronograma_financeiro\05_financial_schedule_planned_vs_actual.png
outlook	True	78	27	4	29	D:\GOLD\outskrape\gephi\Domain_entity_communication_network_2.gephi
image_risk	True	16	7	0	7	D:\SILVER\risk_analyser\reports\PDFs\AI_Image_Safety_Report_13-06-2026_15-34.pdf

12. Figure Gallery and LLM Commentary

Outlook

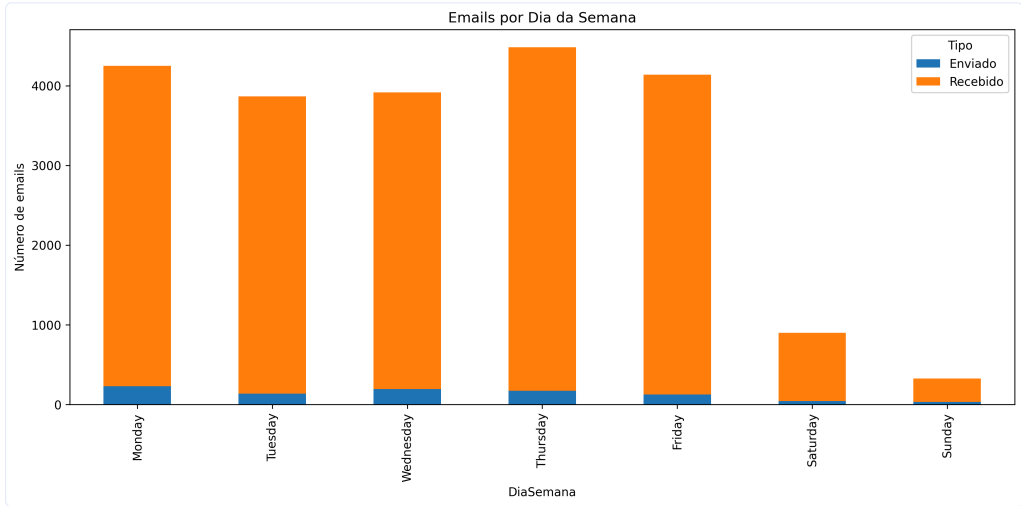
Outlook activity by hour



LLM commentary:

Automatic commentary could not be generated for this figure. Source file: D:\GOLD\outskrape\folder_analysis\previous_silver_figures\hora.png

Outlook activity by weekday



LLM commentary:

Technical Commentary: Outlook Activity by Weekday

Key Observations:

The data reveals a pronounced weekday–weekend disparity in email traffic. Weekday activity (Monday through Friday) remains consistently elevated, with Thursday recording the highest volume. Conversely, weekend activity drops dramatically, with Saturday and Sunday showing minimal engagement—approximately 80–90% lower than weekday averages.

Notable Patterns:

- Sent emails (blue segments) remain relatively modest throughout the week, suggesting incoming correspondence dominates the workload.
- Mid-week (Tuesday–Wednesday) shows a slight dip compared to Monday and Thursday, which may reflect project scheduling or meeting cycles.
- The sharp weekend decline is expected but warrants verification that no critical communications are being missed during off-hours.

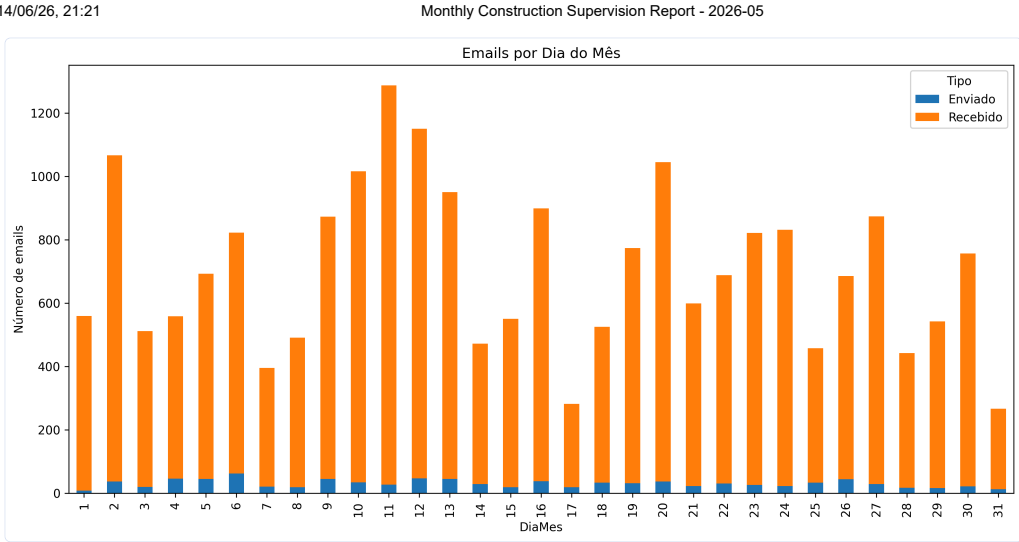
Recommendations for Supervision:

- Verify data completeness for weekend entries; the low volumes may indicate incomplete logging rather than genuine inactivity.
- Review Thursday's peak to identify whether this reflects genuine demand or data anomalies.
- Assess whether weekend email protocols are adequate for urgent site matters.
- Cross-reference with project schedules to confirm patterns align with construction activities.

Data Quality Note: The chart title is in Portuguese whilst axis labels are mixed; clarify source documentation for consistency.

Source file: D:\GOLD\outskrape\folder_analysis\previous_silver_figures\diasemana.png

Outlook activity by day of month



LLM commentary:

Technical Commentary: Outlook Activity by Day of Month

Data Overview

The chart displays email traffic (sent and received) across a calendar month, with received emails (orange) substantially exceeding sent emails (blue) throughout the period.

Key Observations

The data exhibits significant daily volatility, with received emails ranging from approximately 270 to 1,300 messages. Notable peaks occur around days 11–12, suggesting concentrated incoming communication periods, possibly linked to project milestones or reporting cycles.

Anomalies and Risks

Days 7, 14, and 17 show marked troughs in activity, which may indicate weekends or public holidays requiring verification. Day 17 is particularly concerning, with minimal traffic that warrants investigation. The sent-to-received ratio remains consistently low, suggesting potential communication bottlenecks or delays in response times.

Recommendations for Supervision

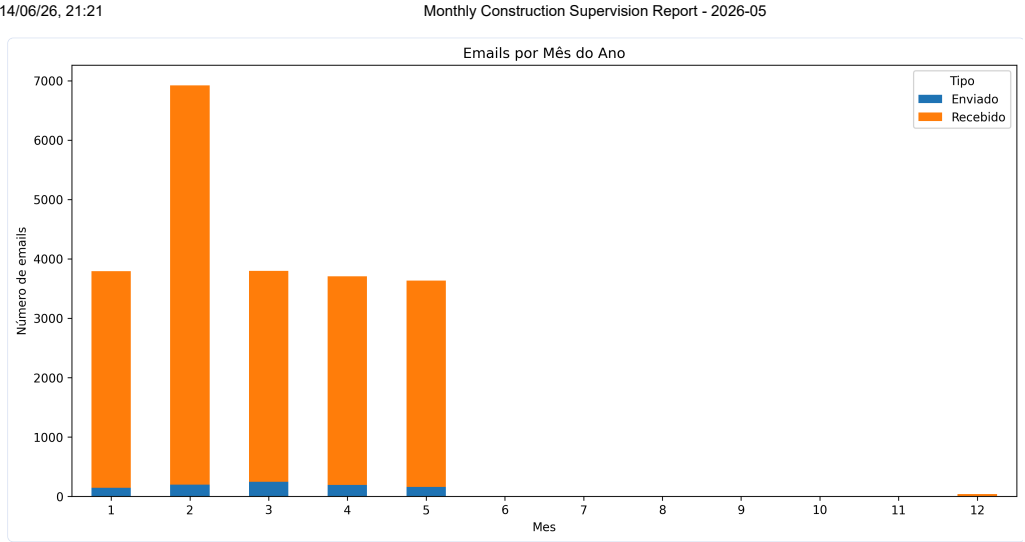
- Verify calendar alignment to distinguish weekends from anomalous low-activity days
- Investigate whether peak periods (days 11–12) correlate with project deadlines or site events
- Review communication protocols to ensure adequate outbound correspondence relative to incoming volume
- Cross-reference with site activity logs to validate email patterns reflect actual construction progress

Data Quality Note

Confirm data extraction methodology to ensure completeness and accuracy of the underlying Outlook records.

Source file: D:\GOLD\outskrape\folder_analysis\previous_silver_figures\d1ames.png

Outlook activity by month-year



LLM commentary:

Technical Commentary: Outlook Activity by Month

Data Quality Concern

This chart presents a significant data-quality anomaly requiring immediate investigation. Email activity is recorded only for months 1–5, with months 6–12 showing negligible or zero values. This pattern is inconsistent with typical construction project communication patterns and suggests either incomplete data collection, a system migration, or a reporting error rather than genuine operational trends.

Key Observations

Month 2 exhibits a pronounced spike in received emails (approximately 6,900), roughly double the activity in other recorded months. Sent emails remain consistently low across all months, indicating this account primarily receives rather than originates correspondence.

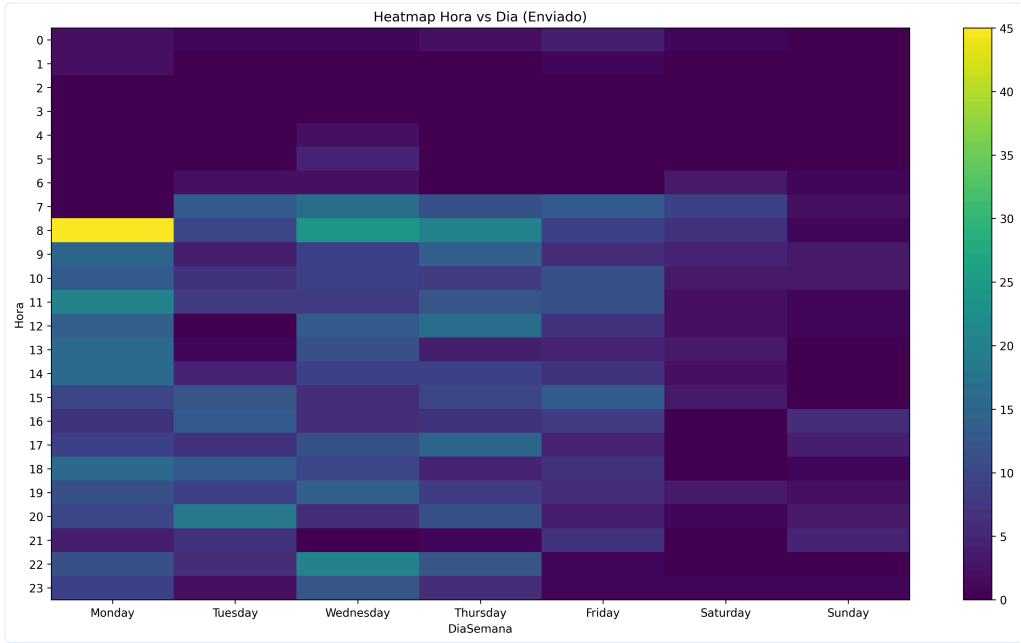
Recommendations for Supervision

- Verify data completeness for months 6–12 before drawing conclusions about project communication trends.
- Investigate the Month 2 spike to determine if it reflects genuine project activity or data duplication.
- Confirm whether the low sent-email ratio aligns with the account holder's actual role and responsibilities.
- Establish baseline expectations for monthly email volumes to monitor future project communication effectively.

The current dataset is insufficient for meaningful trend analysis without clarification of these anomalies.

Source file: D:\GOLD\outskrape\folder_analysis\previous_silver_figures\mesano.png

Outlook sent-email heatmap



LLM commentary:

Technical Commentary: Outlook Sent-Email Heatmap

Key Observations:

The heatmap reveals distinct email communication patterns across the working week. Monday mornings (08:00 hour) show exceptional activity, with the highest concentration of outgoing emails reaching approximately 45 messages. This anomaly warrants investigation—it may indicate routine weekly briefings, project kickoffs, or system-generated correspondence requiring verification.

Patterns Identified:

Mid-week activity (Tuesday–Thursday, 08:00–14:00 hours) demonstrates consistent engagement during standard business hours, with moderate intensity across these periods. Weekend activity is negligible, confirming appropriate work-life boundaries. Early morning hours (00:00–07:00) and late evening (22:00–23:00) show minimal traffic, suggesting limited after-hours communication.

Supervision Recommendations:

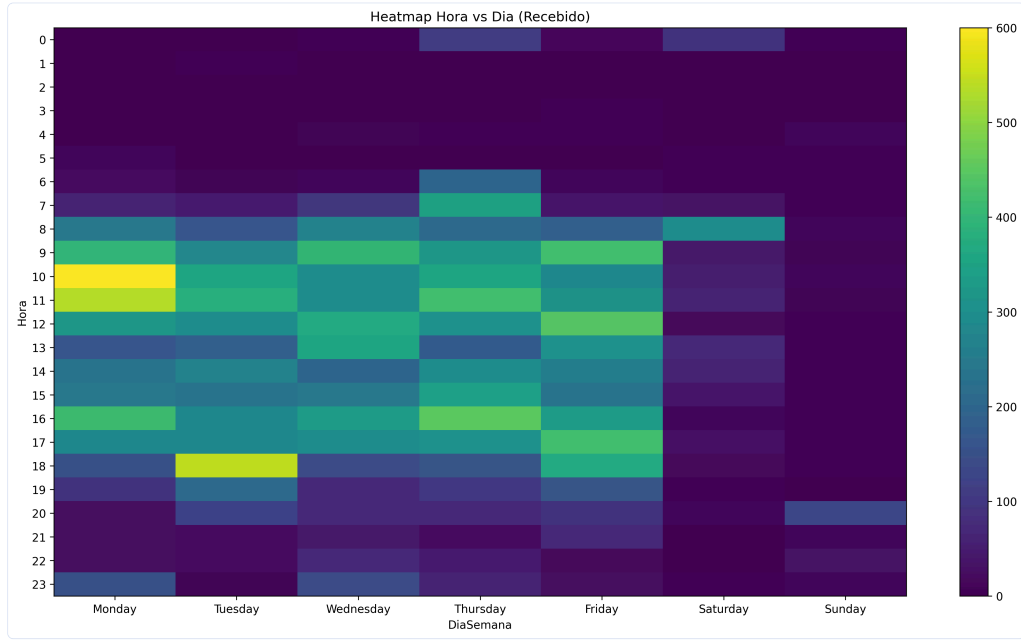
- Verify the Monday 08:00 spike to confirm it reflects legitimate project communications rather than data anomalies or automated processes.
- Monitor whether email volume correlates with construction milestones or site activities.
- Assess whether communication patterns support adequate site coordination and stakeholder engagement.
- Cross-reference with project schedules to validate timing appropriateness.

Data Quality Note:

Ensure the source data (outskrape folder analysis) has been validated for completeness and accuracy before drawing operational conclusions.

Source file: D:\GOLD\outskrape\folder_analysis\previous_silver_figures\heatmap_Enviado.png

Outlook received-email heatmap



LLM commentary:

Technical Commentary: Outlook Received-Email Heatmap

Communication Pattern Analysis

The heatmap reveals distinct email receipt patterns across the working week. Peak activity concentrates during mid-morning hours (10:00–11:00) and midday (12:00–13:00), with Monday and Wednesday showing the highest volumes. This suggests concentrated communication periods that may warrant workload management review.

Key Observations

Weekend activity (Saturday–Sunday) remains minimal, indicating appropriate work–life boundary maintenance. However, early morning hours (0:00–6:00) display sporadic activity, potentially indicating automated systems or time-zone related communications requiring verification.

Risk Considerations

The sharp decline in email traffic after 17:00 on weekdays, coupled with minimal Friday afternoon activity, may reflect genuine workflow patterns or potential data gaps. Supervisors should verify whether this represents actual communication reduction or incomplete data capture.

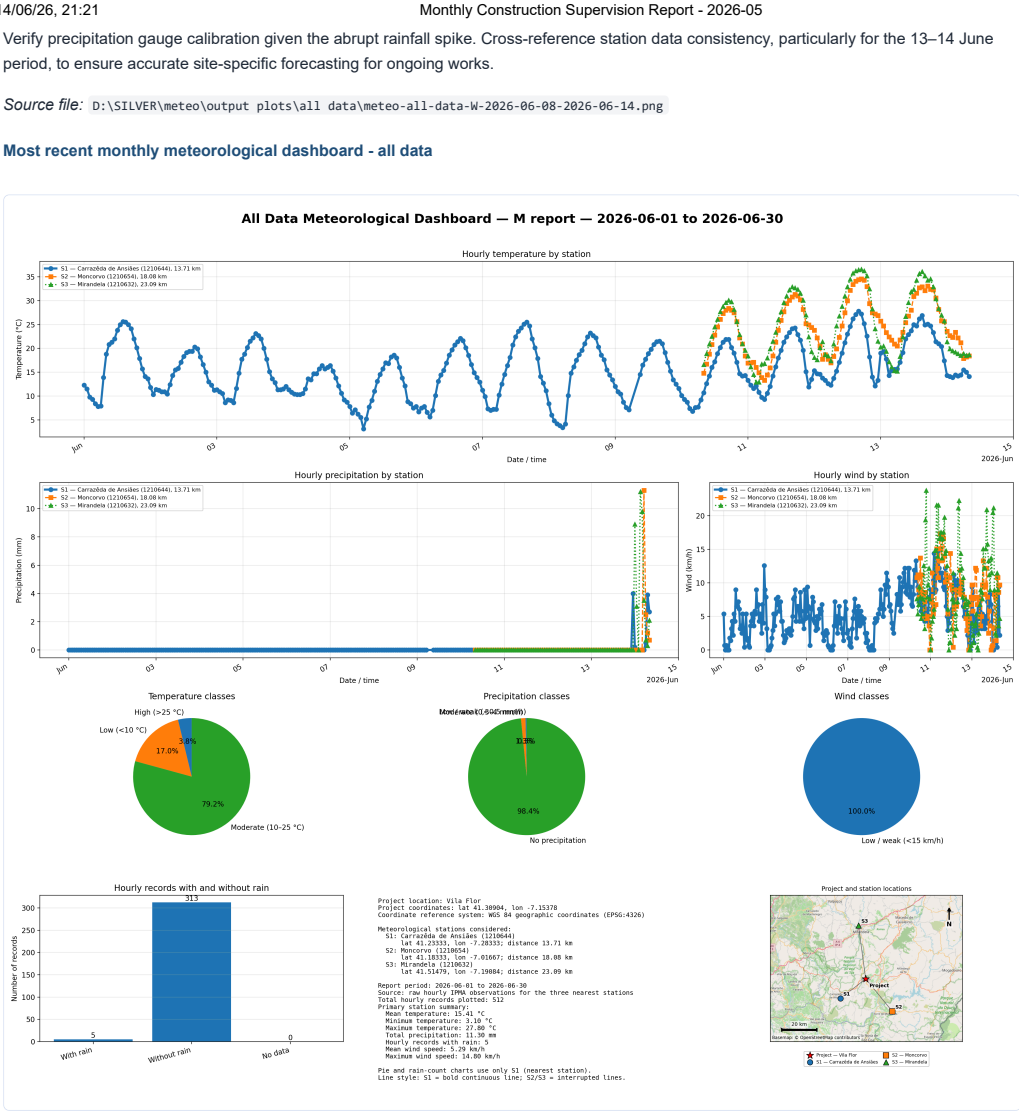
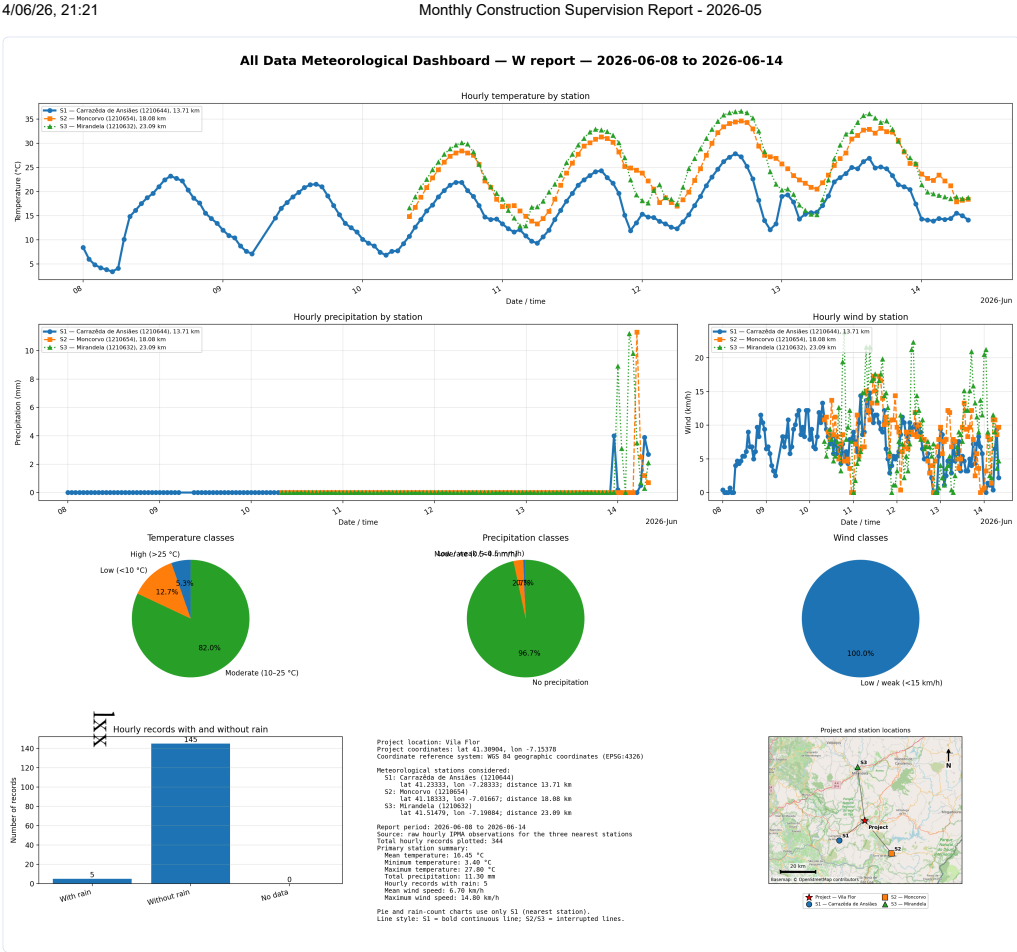
Recommendations

- Validate data completeness for evening hours
- Investigate the purpose of early-morning email clusters
- Consider whether peak-hour concentrations impact site coordination efficiency
- Cross-reference with project schedules to ensure communication aligns with critical activities

Source file: D:\GOLD\outskrape\folder_analysis\previous_silver_figures\heatmap_Recebido.png

Meteo

Most recent weekly meteorological dashboard - all data



LLM commentary:

Meteorological Dashboard Commentary — Week 8–14 June 2026

Data Quality & Coverage

The dashboard presents comprehensive data from three monitoring stations across the reporting period. Total hourly records number 344, with only 5 records affected by rainfall, indicating robust data capture. However, the significant spatial separation between stations (13.71–23.09 km) should be considered when applying findings to specific site locations.

Key Observations

Temperature patterns show consistent diurnal cycles with moderate conditions (82% of hours within 10–25 °C range). A notable weather transition occurred around 13 June, with increased variability in all parameters thereafter.

Precipitation was negligible throughout most of the week (96.7% no rainfall), with concentrated rainfall events concentrated on 13–14 June. This sudden onset warrants attention for drainage and ground stability assessments.

Wind conditions remained predominantly low (<15 km/h), classified as weak throughout. This stability is favourable for construction activities, though the absence of stronger wind events may indicate incomplete anemometer coverage or localised sheltering effects.

Recommendations

LLM commentary:

Technical Commentary: June 2026 Meteorological Dashboard

Data Coverage & Quality

The dashboard presents complete hourly records across three monitoring stations (S1–Carrasquilla, S2–Minscovo, S3–Miravalles) for the period 1–30 June 2026. Data integrity appears sound with 512 total hourly records and minimal gaps. Station distribution spans 13.71–23.09 km from the project site.

Key Observations

Temperature conditions remained predominantly moderate (79.2%), with 17.0% high-temperature hours (>25 °C) concentrated mid-to-late month. Diurnal cycles show consistent patterns with peaks around 25–30 °C.

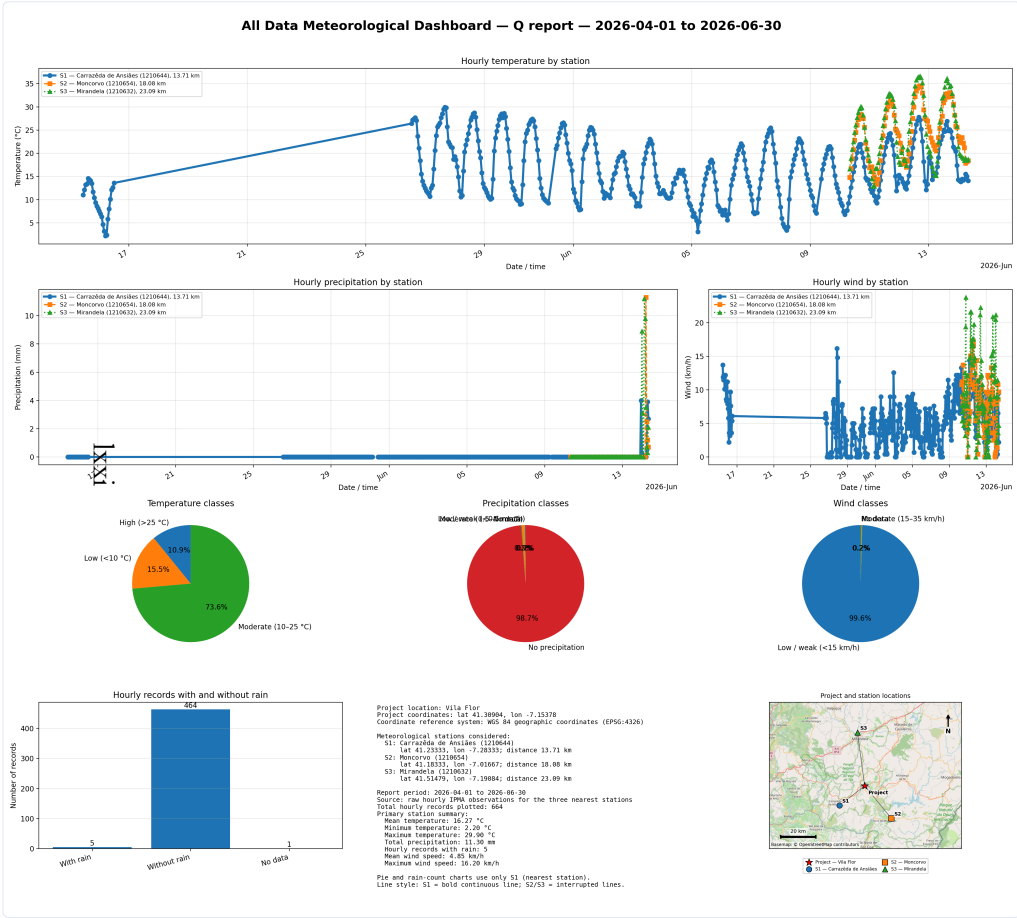
Wind conditions remained consistently low (<15 km/h) across 100% of observations, favouring dust control and safe crane operations.

Supervision Recommendations

Monitor the three-station network for continued consistency. The late-month precipitation spike should prompt review of drainage effectiveness and any associated ground settlement. Consider whether wind conditions remain representative for ongoing operations.

Source file: D:\SILVER\meteo\output\plots\all data\meteo-all-data-M-2026-06-01-2026-06-30.png

Most recent quarterly meteorological dashboard - all data



LLM commentary:

Technical Commentary: Q2 2026 Meteorological Dashboard

Data Overview

This quarterly report covers April–June 2026 across three monitoring stations. The dataset comprises 664 hourly records with minimal data gaps, providing adequate coverage for construction supervision purposes.

Temperature: Conditions remained predominantly moderate (73.6%), with a gradual warming trend from April through mid-June. High temperatures (>25 °C) occurred primarily in the final fortnight, reaching approximately 35 °C. This pattern is typical for the season and poses no exceptional risk to standard construction activities.

Precipitation: Notably dry period with 98.7% of hours recording no rainfall. Concentrated precipitation events occurred only in mid-June, with isolated peaks reaching approximately 10 mm/hour. This favourable condition minimised weather-related delays.

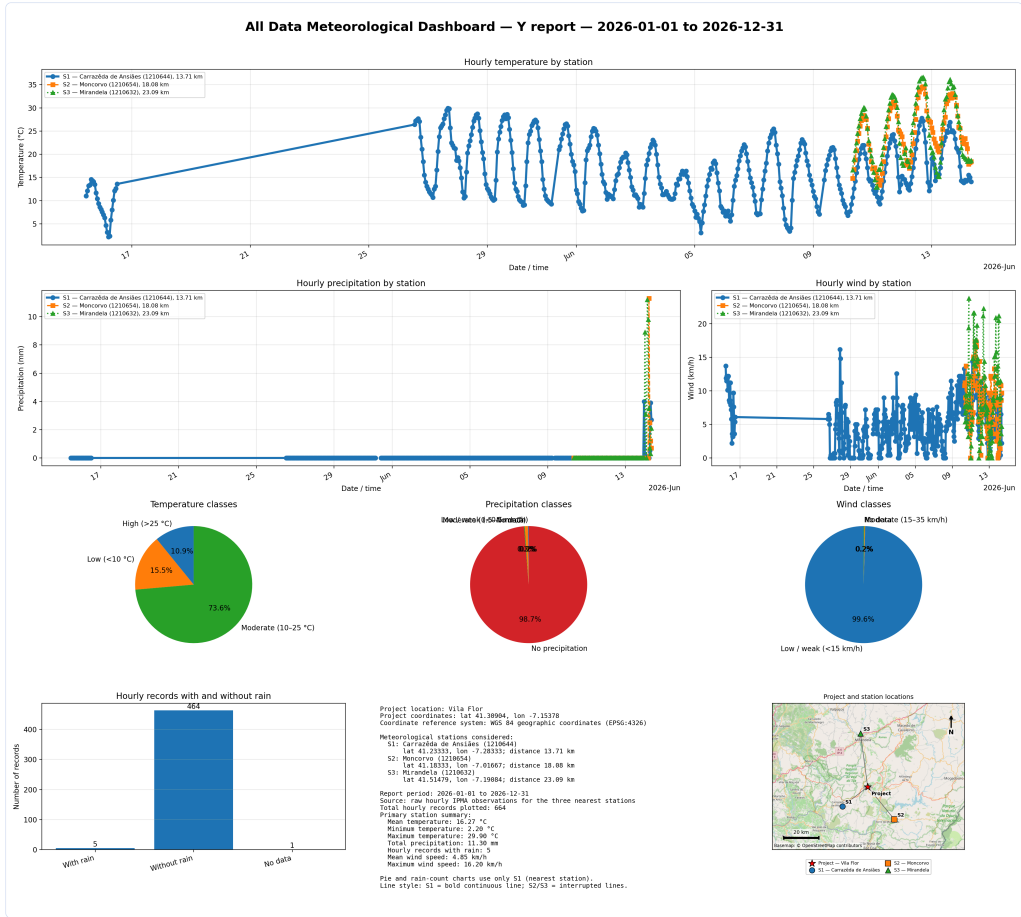
Wind: Predominantly low-to-weak conditions (99.6%), with occasional moderate gusts (15–35 km/h) in late June. Wind speeds remained within acceptable limits for most construction operations.

Recommendations

Monitor the late-June wind and precipitation spike for any scheduling impacts. Data quality appears sound; however, verify the five hourly records flagged as missing to ensure completeness for future analysis.

Source file: D:\SILVER\meteo\output\plots\all data\meteo-all-data-Q-2026-04-01-2026-06-30.png

Most recent yearly meteorological dashboard - all data



LLM commentary:

Technical Commentary: Meteorological Dashboard 2026

Data Coverage & Quality

The dashboard presents a complete annual dataset (2026-01-01 to 2026-12-31) from three meteorological stations. Total hourly records plotted: 664, with minimal data gaps noted. This provides adequate temporal resolution for construction supervision purposes.

Temperature Observations

Moderate conditions dominate (73.6%), with temperatures ranging approximately 5–30 °C. A clear seasonal trend is evident, with warmer peaks in mid-year. Low temperatures (<10 °C) represent 15.9% of records, primarily in early months—relevant for concrete curing and material handling protocols.

Precipitation & Wind Patterns

Critically, 98.7% of records show no precipitation, with concentrated rainfall occurring in late May/early June. This concentrated wet period warrants attention for scheduling earthworks and surface preparation. Wind conditions remain predominantly low (<15 km/h) at 99.6%, with isolated moderate gusts in June.

Supervisory Recommendations

Verify the three-station dataset consistency, particularly the June precipitation spike. Confirm data quality for the five recorded rain events. Plan weather-sensitive activities around the identified dry period. Monitor the June wind anomaly for potential impact on crane operations or exposed works.

Source file: D:\SILVER\meteo\output plots\all data\meteo-all-data-Y-2026-01-01-2026-12-31.png

Daily Reports

Daily report activities per month

selector	2025-07	2025-08	2025-09	2025-10	2025-11	2025-12	2026-01	2026-02	2026-03	2026-04	2026-05
Desmatação	0	0	3	2	4	3	0	3	1	2	2
Escavação	6	4	3	0	1	0	0	4	5	11	18
Entivação	0	0	0	1	3	0	0	0	0	6	0
Instalação de conduta	0	0	2	3	0	0	0	2	1	1	0
montagem de armadura	0	0	0	0	3	6	1	0	12	10	4
cofragem	0	0	5	8	12	4	8	2	6	13	13
betonagem	0	0	4	5	5	2	4	2	4	6	4
descofragem	0	0	1	3	4	1	6	3	0	5	1
aterro	0	0	0	0	0	0	0	0	3	3	0
ABGE	0	3	0	0	0	0	0	0	1	0	0
Betuminosos	0	0	0	0	0	0	0	0	2	1	0
Compactação	0	0	0	0	0	0	0	0	3	1	0
Ensaio	0	2	14	17	1	0	0	0	3	1	0

LLM commentary:

Technical Commentary: Daily Report Activities per Month

The data reveals significant activity concentration in the latter months, particularly from April onwards, indicating project progression through structural and finishing phases. Excavation shows a marked escalation in May (18 reports), suggesting intensive earthworks during this period.

Key observations:

- ****Early-stage activities**** (July–August 2025) show minimal reporting, with only demolition and testing activities recorded.
- ****Peak activity**** occurs April–May 2026, with formwork, excavation, and testing dominating the schedule.
- ****Data quality concern:**** Several activities (aterro, ABGE, bituminosos, compactação) show sporadic or minimal entries, warranting verification of reporting compliance or actual work execution.
- ****Anomaly:**** Ensaio (testing) records 14–17 entries in September–October 2025, then drops to zero—clarify whether testing was completed or reporting lapsed.

Recommendations:

Verify that low-activity months reflect genuine work schedules rather than incomplete reporting. Cross-reference excavation and formwork peaks with labour and equipment deployment records to confirm resource adequacy. Investigate the sudden cessation of testing activities to ensure quality assurance protocols remain active.

Source file: D:\GOLD\Relat_diario\activities per month.png

Daily report activities per week

selector	2025-W29	2025-W30	2025-W31	2025-W32	2025-W33	2025-W34	2025-W35	2025-W36	2025-W37	2025-W38	2025-W39	2025-W40	2025-W41	2025-W42	2025-W43	2025-W44	2025-W45	2025-W46	2025-W47	2025-W48	2025-W49	2025-W50	2025-W51	2025-W52	2026-W02	2026-W03	2026-W04	2026-W05	2026-W06	2026-W07	2026-W08	2026-W09	2026-W10	2026-W11	2026-W12	2026-W13	2026-W14	2026-W15	2026-W16	2026-W17	2026-W18	2026-W19	2026-W20	2026-W21	2026-W22	
Desmatamento	0	0	0	0	0	0	0	0	1	2	0	0	1	0	1	0	2	2	0	0	0	0	1	2	0	0	0	0	0	0	1	2	0	0	0	1	0	0	0	0	0	1	1	1		
Escavação	3	2	1	1	2	1	0	0	0	0	3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	3	1	0	0	4	1	0	0	0	1	6	4	5	5	3		
Entivação	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	4	1	0	0	0	3	1	2	0	0	0		
Instalação de conduta	0	0	0	0	0	0	0	0	0	0	2	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	1	0	0	0	0	1	0	0	0	0	0		
montagem de armadura	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	4	2	0	1	0	0	0	0	0	0	0	1	0	0	5	4	3	3	3	0	3	2	0	0	2	
cofragem	0	0	0	0	0	0	0	0	0	1	4	1	0	0	3	3	1	6	3	3	3	1	1	0	0	3	5	0	1	0	1	1	0	1	0	2	3	4	1	5	1	2	2	1	4	6
betonagem	0	0	0	0	0	0	0	2	2	0	0	0	0	0	2	3	0	3	0	2	0	0	2	0	3	0	1	0	2	0	0	1	1	1	1	0	1	1	1	0	1	1	2	0	0	
descofragem	0	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0	2	1	1	0	0	1	0	1	0	1	1	4	0	0	2	1	0	0	0	0	1	0	1	1	2	0	0	1	0	
aterro	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	0	3	0	0	0	0	0		
ABGE	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0		
Betuminosos	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	1	0	0	0	0		
Compactação	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	0	0	0	1	0	0	0	0		
Ensaio	0	0	0	0	0	0	2	1	1	5	5	5	5	6	2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	0	0	0	0	0	1	0	0	0	0		

LLM commentary:

Technical Commentary: Daily Report Activities per Week

This heatmap presents activity frequency across multiple construction work categories over a 52-week period. Several observations warrant supervisory attention:

Activity Distribution:

The data reveals significant variability in reporting consistency. Categories such as "Escavação" and "Cofragem" demonstrate sustained activity levels throughout the monitoring period, whilst others—notably "ABGE," "Betuminosos," and "Compactação"—show minimal or sporadic entries, suggesting either low activity frequency or potential under-reporting.

Data Quality Concerns:

Multiple rows contain predominantly zero values, raising questions about whether these activities genuinely did not occur or whether reporting protocols were not followed. The "Aterro" category, for instance, shows activity only in recent weeks, which may indicate late-stage works or incomplete historical documentation.

Risk Indicators:

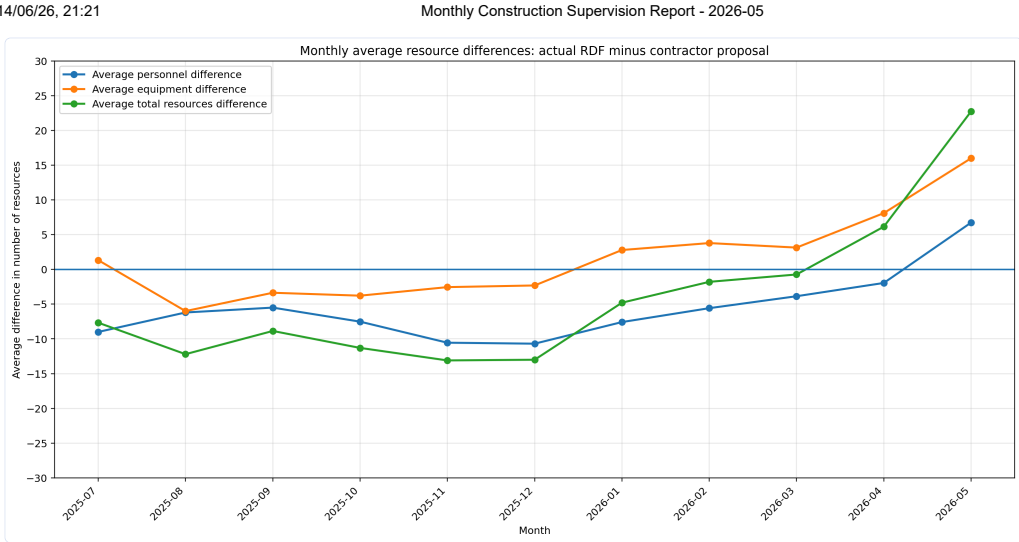
The concentration of higher values (shown in darker shading) in "Cofragem" and "Escavação" aligns with typical construction sequencing. However, the inconsistent reporting pattern across weeks suggests potential gaps in supervision documentation or resource allocation issues.

Recommendation:

Verify reporting procedures for low-activity categories and establish baseline expectations for daily report submission to ensure comprehensive project oversight.

Source file: D:\GOLD\Relat_diario\activities per week.png

RDF monthly average resource differences



LLM commentary:

Technical Commentary: RDF Monthly Average Resource Differences

Key Observations

The chart reveals a significant upward trend in resource variances from mid-2025 onwards, with all three metrics (personnel, equipment, and total resources) crossing into positive territory by May 2026. This indicates actual deployment now exceeds contractor proposals.

Critical Patterns

Early Period (Jul 2025–Dec 2025): Negative variances dominated, suggesting under-resourcing relative to proposals. The trough occurred around November–December 2025, with total resources approximately 12–13 units below proposal.

Recent Escalation (Jan–May 2026): A sharp reversal shows accelerating positive differences, reaching approximately +22 units for total resources by May 2026. Equipment variance (+16 units) now substantially exceeds personnel variance (+7 units).

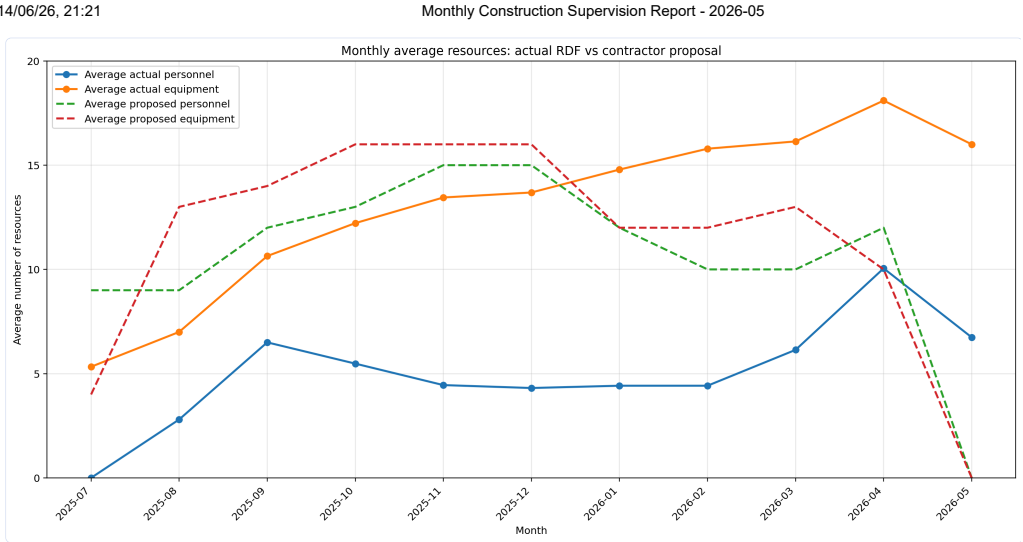
Supervision Concerns

- **Resource Creep:**** The steep May 2026 spike warrants investigation—confirm whether this reflects genuine project acceleration or reporting anomalies.
- **Equipment Dominance:**** Equipment differences now drive total variance; verify procurement justification and cost implications.
- **Data Quality:**** The sharp inflection point suggests potential changes in reporting methodology or project scope. Cross-reference with site records.

Recommendation: Request detailed breakdown of May 2026 variances and clarify whether increased resources align with schedule recovery or scope changes.

Source file: D:\GOLD\Relat_diario\gold_RDF_monthly_average_differences.png

RDF monthly average resources



LLM commentary:

Technical Commentary: RDF Monthly Average Resources

Key Observations

Resource Deployment Gap

Actual personnel deployment remains significantly below contractor proposals throughout the monitoring period. Whilst proposed personnel peaked around mid-2025, actual personnel remained substantially lower, suggesting either phased mobilisation or underutilisation of planned workforce.

Equipment Utilisation Concern

Actual equipment deployment substantially exceeds proposed levels from mid-2025 onwards, peaking in 2026-04. This divergence warrants investigation—equipment may be deployed inefficiently, or the contractor's proposal underestimated requirements.

Recent Mobilisation Activity

A notable surge in actual personnel occurred in 2026-03/04, approaching proposed levels for the first time. However, this coincides with a sharp decline in both proposed and actual resources by 2026-05, suggesting project phase completion or demobilisation.

Supervision Recommendations

- **Verify resource justification**** for the equipment overage in mid-2025 through 2026-04
- **Confirm final demobilisation schedule**** given the abrupt resource reduction in 2026-05
- **Review data quality**** to ensure actual figures reflect genuine site conditions rather than reporting inconsistencies
- **Assess schedule alignment**** between resource deployment and planned construction phases

Source file: D:\GOLD\Relat_diario\gold_RDF_monthly_average_resources.png

Price Revision

Price revision evolution



LLM commentary:

Technical Commentary: Price Revision Evolution

The chart reveals significant cost management concerns requiring immediate attention. The accumulated price revision (orange line) demonstrates a steep upward trajectory from February through July 2025, stabilising thereafter at approximately €3,000,000. This substantial cumulative increase suggests either inadequate initial budgeting or substantial scope changes during the project's early phases.

Key observations:

- Peak volatility:** July and August 2025 exhibit the highest monthly price revisions, indicating critical decision points or unforeseen conditions.
- Divergence pattern:** From July onwards, planned and actual monthly values converge, whilst monthly revisions decline sharply, suggesting improved cost control post-August.
- Data consistency:** The blue line (monthly price revision) shows erratic fluctuations, particularly between June and October, warranting investigation into revision drivers.
- Stabilisation phase:** November 2025 onwards demonstrates minimal monthly revisions, indicating project stabilisation.

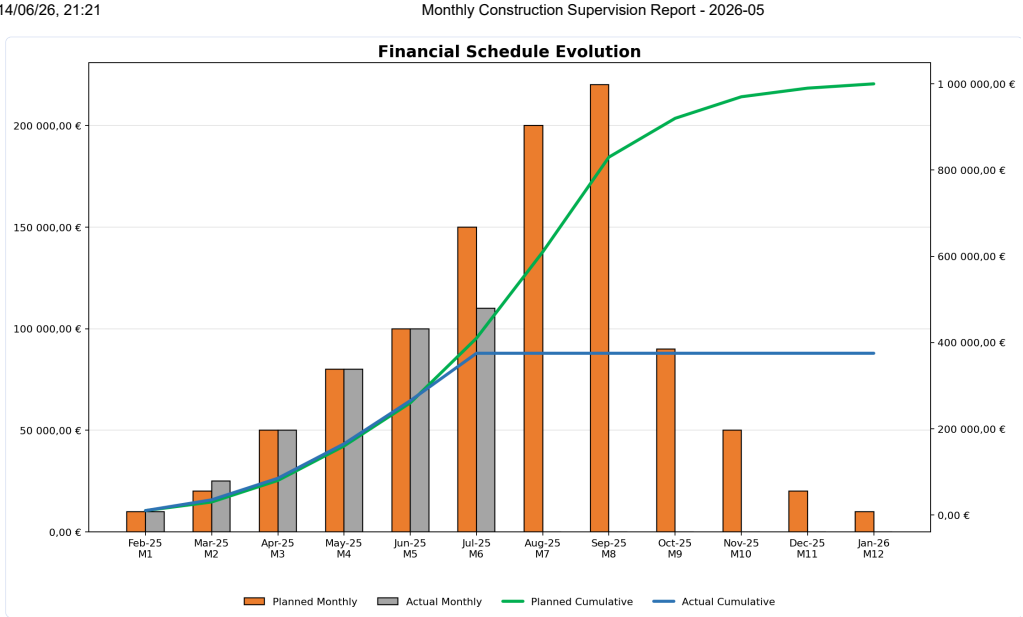
Recommendations:

Review the contractual basis and change order documentation for July–August activities. Verify whether revisions reflect genuine scope changes or estimation errors. Establish preventative measures to avoid similar patterns in future phases. Monitor the remaining project period closely, as the accumulated revision represents a material budget variance.

Source file: D:\SILVER\Revisao\resultados_revisao\resultado_revisao_precos_C001_Auto_006_2025-06_price_revision_evolution.png

Financial Schedule

Financial schedule planned versus actual



LLM commentary:

Technical Commentary: Financial Schedule Evolution

Key Observations:

The financial schedule reveals significant divergence between planned and actual expenditure trajectories. The planned cumulative curve (green) demonstrates steady acceleration through September 2025, reaching approximately €1,000,000, whilst the actual cumulative line (blue) plateaus substantially earlier, stabilising around €350,000–€375,000 from August onwards.

Critical Anomalies:

A pronounced gap emerges from July 2025 onwards, indicating substantial underspend against the original financial programme. Monthly actual disbursements (grey bars) consistently fall below planned allocations (orange bars), particularly during the critical mid-project phases (July–September).

Risk Assessment:

This pattern suggests either delayed payment processing, incomplete work certification, or cash-flow constraints. The flattening of actual cumulative spend from month eight warrants immediate investigation to determine whether this reflects genuine project delays, administrative bottlenecks, or data-recording issues.

Recommendations:

Verify payment documentation and work completion certificates for periods M6–M9. Confirm whether planned figures remain realistic or require revision. Establish corrective actions to align actual expenditure with programme targets, ensuring adequate resource mobilisation for remaining project phases.

Source file: D:\SILVER\cronograma_financeiro\05_financial_schedule_planned_vs_actual.png

Image Risk

Image risk analyser output - image 1.jpeg



LLM commentary:

Technical Commentary: Image Risk Analysis

Site Safety and Housekeeping Concerns

The image reveals several supervision points requiring immediate attention. Loose excavated material and debris are scattered across the working surface, creating trip hazards and potential FOD (foreign object damage) risks. The pink warning barrier appears inadequately positioned relative to the excavation works and mobile plant operating in proximity.

Plant and Traffic Management

Heavy machinery (excavator and support vehicles) are operating in close proximity to the work area. Sight lines appear compromised by the utility pole and scattered materials. No visible traffic management signage or personnel exclusion zones are evident, which is concerning given the active vehicle movements.

Utility and Structural Risks

The overhead power lines and utility infrastructure require careful monitoring during excavation activities. The depth and proximity of works to the pole foundation should be verified against utility strike protocols.

Recommendations

Enforce immediate site tidying to remove tripping hazards. Reposition warning barriers and establish clearer exclusion zones. Confirm utility clearances and implement enhanced traffic management procedures. Document compliance with site induction requirements for all personnel present.

Source file: D:\SILVER\risk analyser\reports\images\image 1.jpeg

Image risk analyser output - image 2.jpg



LLM commentary:

Technical Commentary: Image Risk Analysis – Excavation Works

Key Observations:

The image documents an active excavation operation with several supervision points requiring attention. The trench excavation appears to be progressing for utility installation, with appropriate shoring visible on the left side. A mini-excavator is positioned centrally within the trench, and drainage pipework is staged nearby.

Risk Factors Identified:

- Trench Stability:** The right side of the excavation shows minimal or no visible shoring support, presenting a potential collapse hazard. This requires immediate assessment against ground conditions and depth.
- Personnel Proximity:** A site operative is visible in close proximity to active machinery operations, necessitating verification of exclusion zones and communication protocols.
- Site Housekeeping:** Equipment and materials are distributed across the working area; ensure adequate segregation between plant movements and personnel zones.
- Weather Conditions:** Autumn conditions evident; monitor for water ingress and ground saturation affecting stability.

Recommendations:

Verify compliance with excavation safety standards, confirm appropriate shoring design, and reinforce traffic management procedures. Document ground conditions and any deviations from the excavation method statement.

Source file: D:\SILVER\risk analyser\reports\images\image 2.jpg

Image risk analyser output - image 3.jpg



LLM commentary:

Technical Commentary: Image Risk Analysis – Excavation Works

Key Observations:

The excavation trench exhibits several supervision concerns requiring immediate attention. The trench depth appears substantial, yet temporary support systems are minimal—only timber props are visible, which may be inadequate depending on soil classification and depth. The exposed soil profile shows mixed strata with potential instability, particularly on the left side where slumping is evident.

Risk Areas:

- Inadequate shoring/propping relative to apparent trench depth
- Loose spoil material stockpiled adjacent to the trench edge, increasing collapse risk
- Personnel proximity to unsupported excavation (highlighted figure on right)
- No visible trench inspection documentation or competent person sign-off

Recommendations:

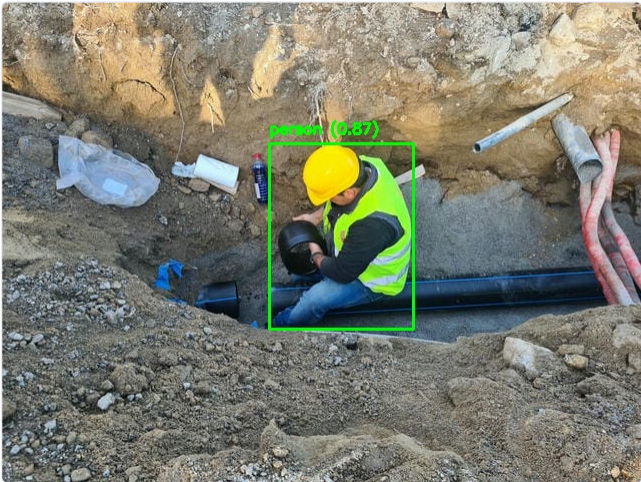
Verify that a qualified engineer has assessed soil conditions and approved the temporary works design. Confirm compliance with relevant excavation safety standards. Establish exclusion zones around the trench perimeter and ensure all personnel wear appropriate PPE. Document soil classification and implement enhanced support measures if depth exceeds safe working parameters. Conduct daily inspections before work commences.

Data Quality Note:

Image resolution is adequate for hazard identification; however, obtain supplementary documentation (method statements, soil reports) to validate temporary works adequacy.

Source file: D:\SILVER\risk analyser\reports\images\image 3.jpg

Image risk analyser output - image 4.jpeg



LLM commentary:

Technical Commentary: Image Risk Analysis – Image 4.jpeg

Observation:

The image risk analyser has flagged a personnel detection within an excavation work area. The automated detection correctly identifies a worker wearing appropriate personal protective equipment (hard hat and high-visibility vest) engaged in underground utility work.

Supervision Points:

The detection algorithm demonstrates effective capability in identifying workers in confined spaces. However, several site conditions warrant supervisor attention:

- ****Housekeeping:**** Scattered debris and materials are visible around the excavation perimeter, presenting trip hazards and potential FOD (foreign object debris) risks.
- ****Trench Safety:**** Confirm that appropriate shoring, support systems, or slope angles comply with relevant excavation standards and site safety procedures.
- ****PPE Compliance:**** Whilst visible PPE is noted, verify that all personnel in the trench are equipped with required respiratory protection if atmospheric monitoring indicates necessity.
- ****Documentation:**** Cross-reference this image timestamp with daily site logs to ensure all underground work was authorised and properly supervised.

Recommendation:

Review site induction records and conduct a toolbox talk regarding excavation housekeeping standards. Ensure the image timestamp aligns with planned underground activities.

Source file: D:\SILVER\risk analyser\reports\images\image 4.jpeg

Image risk analyser output - image 5.jpg



LLM commentary:

Technical Commentary: Image Risk Analysis – Image 5.jpg

Key Observations:

This image documents active material delivery and site operations with several supervision points requiring attention:

Safety Concerns:

- Tipper truck positioned actively discharging aggregate material onto the roadway
- Minimal traffic control visible; only one traffic cone deployed for a significant obstruction
- Pink barrier blocks present but appear inadequately positioned relative to the discharge zone
- Pedestrian and vehicular access risk is elevated given the proximity of active machinery to the public road

Site Management:

- Material stockpiling occurring directly on the carriageway, which may contravene traffic management protocols
- Excavator visible in background suggests concurrent earthworks operations
- Weather conditions appear overcast; surface conditions may affect vehicle stability

Data Quality:

The image timestamp annotation (visible in green) confirms documentation date/time capture, supporting audit trail integrity.

Recommendations:

- Review traffic management plan compliance; consider expanding exclusion zone
- Verify permits for roadway obstruction and material handling
- Assess coordination between concurrent operations (tipper discharge and excavation)
- Confirm operatives are briefed on pedestrian safety protocols

This image warrants follow-up inspection to ensure compliance with site safety standards.

Source file: D:\SILVER\risk_analyser\reports\images\image 5.jpg

Image risk analyser output - image 6.jpg



LLM commentary:

Technical Commentary: Image Risk Analyser Output – Image 6.jpg

Observation:

The image documents compaction works being undertaken on a trench or foundation base using a vibratory plate compactor. The operator is appropriately attired in high-visibility clothing and hard hat.

Key Supervision Points:

Safety Compliance: Verify that the operator is wearing appropriate personal protective equipment (PPE), including hearing protection, given the noise levels associated with vibratory equipment operation.

Equipment & Method: Confirm that the compactor is suitable for the soil conditions visible in the image. The granular material on the left and the existing concrete surface on the right suggest mixed ground conditions requiring careful assessment.

Compaction Quality: Ensure adequate passes are being made to achieve specified density requirements. Document the number of passes and any testing protocols (e.g., plate bearing tests) being employed.

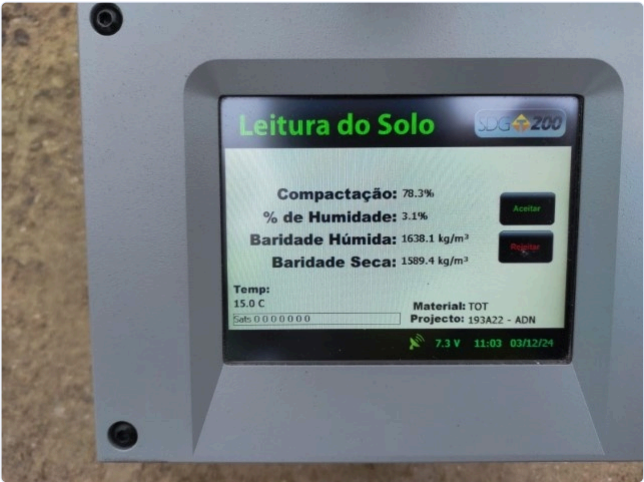
Data Quality: The green bounding box overlay suggests automated risk detection. Verify that the image classification accurately reflects site conditions and that no critical details are obscured.

Anomalies: The proximity to existing concrete structures warrants attention to prevent damage and ensure proper interface preparation.

Recommend site inspection to validate compaction standards and confirm adherence to specification requirements.

Source file: D:\SILVER\risk_analyser\reports\images\image 6.jpg

Image risk analyser output - image 7.jpg



LLM commentary:

Technical Commentary: Soil Testing Analysis

This image documents soil compaction testing conducted on site, displaying critical geotechnical parameters via a portable density gauge. The reading shows compaction at 78.3%, which falls below typical specification thresholds (usually 90–95% for structural fills). This represents a significant quality concern requiring immediate investigation.

The moisture content of 3.1% appears notably low relative to the wet density measurement of 1638.1 kg/m³, suggesting potential data inconsistency or measurement error. The dry density of 1589.4 kg/m³ should be verified against design requirements and the project's compaction curve.

Key observations for supervisory review:

- Compaction deficiency warrants remedial action (re-compaction or material replacement)
- Moisture readings require validation; consider repeat testing
- Timestamp (03/12/24, 11:03) and project reference (193A22–ADN) enable traceability
- Temperature at 15.0°C is acceptable for testing conditions

Recommended actions: Obtain specification compliance limits, cross-reference with laboratory test results, and determine whether this location requires remediation before proceeding with subsequent construction phases.

Source file: D:\SILVER\risk analyser\reports\images\image 7.jpg

Report Generation Date: 14 June 2026 21:16

Report Period: 2026-05

Report Status: DRAFT – subject to professional validation