Redefining an IT operation in the road freight business

Ana Adelaide Martins Moreira

Dissertação de Mestrado

Orientador na FEUP: Prof. Ricardo Soares

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Mestrado em Engenharia e Gestão Industrial

2024-06-27

Aos meus avós, Adélio e Adelaide

Redefinição de processos de uma operação de IT no setor transitário

Resumo

Tendo em conta a complexidade das operações de transitários, e a intensa competitividade do sector, é essencial que as empresas invistam em sistemas de informação sofisticados, promovendo a sua manutenção e melhoria contínua. É neste contexto que se insere o presente projeto de dissertação, que se centra na melhoria do desempenho da equipa de sistemas de informação, mais concretamente do grupo responsável pela integração dos sistemas da Rangel Transitários (RT), uma empresa do grupo Rangel que opera no sector transitário. Este núcleo trabalha no sentido de desenvolver e assegurar a sustentabilidade destes sistemas, garantindo a ausência de eventos disruptivos, e melhorando-os continuamente, de forma a ampliar e maximizar a atividade da RT.

Uma vez que o foco deste projeto era promover a melhoria contínua, a primeira fase de integração na organização e na equipa foi crucial para adquirir uma compreensão de todo o âmbito de atividades da equipa.

Durante este período, foram utilizadas técnicas de modelação de processos para organizar e sistematizar os processos que orientam as operações da equipa, inicialmente representados num mapa de processos.

Utilizando a modelação de processos multinível, foram desenvolvidas matrizes de responsabilidade que mapearam as fases de cada processo, bem como os respetivos intervenientes.

A modelação de segundo nível foi também apresentada através de diagramas *swimlane* para todas as fases do processo. Esses diagramas serviram de base para todas as análises realizadas e possibilitaram a identificação de oportunidades de melhoria. Além disso, foi desenvolvida uma modelação de terceiro nível para tarefas complexas que requerem um nível de detalhe maior e um fluxo de trabalho específico.

Foram desenhadas soluções para materializar as áreas de melhoria previamente identificadas, o que levou à redefinição dos processos da *squad road* e, consequentemente, à reformulação do fluxo de trabalho. Foram calculadas algumas métricas para avaliar o impacto destas soluções e analisar a sua viabilidade.

O projeto demonstrou a importância da análise constante de processos em projetos de desenvolvimento ágil de software, pois promove a implementação de práticas de engenharia reversa para a compreensão do fluxo de atividades e, assim, a identificação de oportunidades de melhoria. Assim, esta análise e posterior reformulação é fundamental para maximizar o desempenho da equipa, melhorando simultaneamente a qualidade dos incrementos produzidos e o nível de serviço prestado aos *stakeholders*.

Abstract

Considering the complexity of freight forwarders' operations, and the intense competitiveness of the sector, companies must invest in sophisticated information systems, while working towards their maintenance and continuous improvement. This is the context of this dissertation project, which focuses on improving the performance of the IT team, more specifically the group responsible for integrating the systems of Rangel Transitários (RT), a company in the Rangel group that operates in the road freight business. This squad works towards developing and sustaining these systems while guaranteeing the absence of disruptive events, continuously improving them, to further enlarge and optimize RT’s activities.

Since the focus of this project was to promote continuous improvement, the first phase of integration in the organization and the team was crucial to acquire an understanding of the team's whole scope of activities.

During this period, process modeling techniques were used to organize and systematize the processes that orient the team's operations, initially represented in a process map.

Using multi-level process modeling, responsibility matrices were developed to map the phases of each process, along with the actors involved.

The second-level modeling was also presented through swimlane diagrams for all the process phases. These diagrams served as the basis for all the analyses conducted and enabled the identification of opportunities for improvement. In addition, third-level modeling was developed for complex tasks that required a greater level of detail and a specific workflow.

Solutions were devised to materialize the areas for improvement previously identified, which led to the redefinition of the road freight squad's processes and, consequently, the reformulation of its workflow. Some metrics were calculated to assess the impact of these solutions and analyze their viability.

The project demonstrated the importance of constant process analysis in agile software development projects, as it promotes the implementation of reverse engineering practices to capture the flow of activities, and, thus, the identification of opportunities for improvement. Thus, this analysis and subsequent reformulation is fundamental to maximize the team's performance, while simultaneously improving the quality of the increments produced and the level of service provided to stakeholders.

Acknowledgements

First and foremost, I would like to thank my coordinator, Ricardo Soares, for the attentive supervision and support throughout my dissertation.

Secondly, I would like to acknowledge Rangel for the opportunity to develop this project. I want to thank Marco Brito, my supervisor in Rangel, for all the assistance, patience, and knowledge, as well as Miguel Cordeiro, who also played a big part in establishing the scope of this project. I would also like to acknowledge the rest of the transport team, mainly João Neto and Inês Fonseca, who were always available to help me and to answer my questions.

I want to express my deep gratitude to my family, who always supported me and encouraged me to work hard to achieve my goals. To my mother, father, brother, and my godmother Maria das Dores, one of the strongest, most resilient people in my life.

To my friends, Rita and Francisca, who accompanied me in university since day one, even in my semester abroad.

Finally, to my second family, the friends who grew up with me, Bruna, Daniel, Gonçalo, Escarleth, Renata, Sara, and Bernardo, it has been an honor.

Table of Contents

Abbreviations

EAI - Enterprise application integration

IT – Information technology

LSP - Logistic Service Platform

MOM - Message-oriented Middleware

RIAM - Rangel Internacional Aérea e Marítima SA

RT - Rangel Transitários SA

TMS - Transport Management System

List of Figures

List of Tables

1 Introduction

The present dissertation was conducted in Rangel Invest, which is part of the Rangel Logistics Solutions group. This project is integrated within the Dissertation course of the Master in Engineering and Industrial Management at the Faculty of Engineering of the University of Porto.

1.1 Project framework and motivation

The project was undertaken in the IT department of Rangel Invest, more specifically in the team dedicated to Rangel's freight transport systems. Particularly, the project was completed in the group dedicated to road freight operations, which will be referred to as Squad Road.

To maintain its competitive advantage and to provide a high-quality service to its stakeholders, Rangel Transitários (RT), which is the group's company that operates within the road freight business, utilizes a wide range of integrated IT systems. Consequently, the performance of these systems is directly connected to RT's operations. Thus, the team dedicated to developing and sustaining these systems must guarantee the absence of disruptive events, their continuous improvement, and, eventually, the integration with other systems, namely the ones used by their partners, to further enlarge and optimize RT's activities.

The developers' workload is organized into two major categories:

- their main activities, which include the collaborations in projects and the development of improvements to RT's systems and respective integrations;
- the provision of support to the system's users.

Since the team's work is multidisciplinary and thus, highly intertwined with RT, Squad Road collaborates with a group of employees known as functional team, which is led by a Product Owner.

Thus, given the wide range and complexity of the squad's activities, the main goal of this dissertation is reformulating the squad road's workflow, while integrating the IT department's operations as an IT business partner. As a result, the goal of the process reengineering is to improve the squad's performance, in terms of productivity and service level, in both core and support activities.

1.2 Reengineering of IT operations at Rangel

1.2.1 Rangel

Rangel Logistic Solutions is an international group of Portuguese origin that encompasses a wide range of activities, namely logistics and supply chain management. With a broad range of services including customs clearance, warehousing, shipping, and e-commerce fulfillment, Rangel is currently an industry leader, in both national and international markets.

As a freight forwarder, Rangel provides multimodal transportation via land, air, sea, and railroad. Furthermore, the company oversees customs procedures, while relieving buyers and sellers of the burden of managing the complexities, nuances, and regulatory compliance associated with conducting business worldwide. The main responsibility of freight forwarders is to act as intermediaries between their clients and transportation companies and, thus, to purchase transportation capacity from airlines, shipping firms, carriers, and rail operators for their clients on behalf of their customers.

Due to the high competitiveness of this sector, operational effectiveness is crucial for the long-term viability of the group. Similarly to other corporations in the industry, Rangel must automate and streamline its operations to maintain its competitive advantage and to provide a high service level to its stakeholders. Therefore, the group must invest in developing and maintaining its IT systems.

1.2.2 IT transport team

The team in which this dissertation was developed is part of Rangel’s IT department. This team collaborates and supports transportation companies within the Rangel group, namely Rangel Transitários SA (RT), Rangel Internacional Aérea e Marítima SA (RIAM), Rangel Distribuição e Logística (RDL) and Transporte Expresso in Angola, Mozambique, Tanzania and Cape Verde. Thus, the team focuses on the information systems used by these companies and works not only on developing new projects and features, but also on improving existing interfaces. Lastly, the team also provides support to the system’s users.

Figure 1 illustrates the structure of the transport team, which is divided into 2 squads. Squad Road supports the activities of RT, which is a road freight enterprise. In turn, the Squad Air & Sea collaborates with RIAM, while supporting aerial and maritime transportation, as well as RDL and the previously mentioned companies related to express transportation.

In terms of hierarchical structure, the team is overseen by a delivery and project manager, who coordinates the workflow according to the goals and deadlines that were previously established alongside the Corporate Information Technology Director, who oversees the IT department.

Each squad consists of one or two programmers, who work on the system code and solve technical problems associated with systems integration. They also provide technical support to the system’s users. The squad is also comprised of an IT business partner, who manages the workflow within each sprint, i.e., the squad’s tasks that should be completed within a defined period, usually 2 weeks, as well as the associated documentation. They are also responsible for collaboration and communication with the business functional team, namely with the Product Owner, who is the person responsible for the squad’s projects. Nonetheless, they are responsible for ensuring the quality of the team's deliverables through the performance of tests.

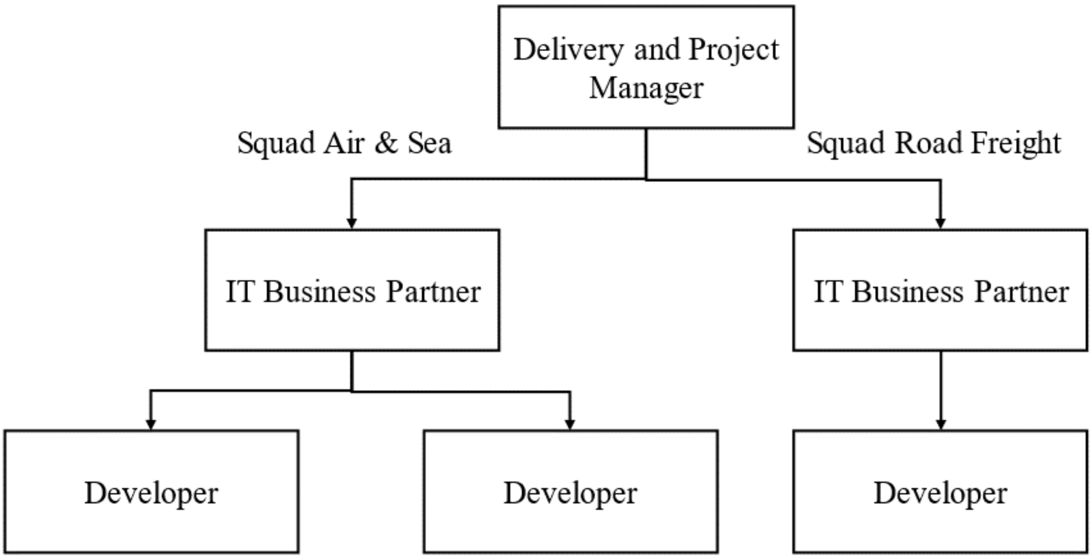

Figure 1 – Structure of the transport team

Even though the team is divided into squads, they are managed as a single team, since they share the same technological stack, i.e., both squads utilize a similar set of technological tools, such as programming languages, frameworks, and databases, among others. In other words, despite working with different companies and thus, with different systems, there is a parallelism in the work performed by both squads, particularly in terms of system integration. In turn, this hierarchical organization also creates an environment of collaboration, while promoting the exchange of knowledge between the developers.

In theory, the management and organization of each squad's work follow similar practices. However, it is different due to the divergent organizational structure. On the one hand, in the Air & Sea squad, there is a person who is fully dedicated to the role of IT Business Partner, thus acting as Scrum Master. On the other hand, the project and delivery manager accumulated the role of Squad Road's IT Business Partner, while also performing as a programmer.

1.2.3 The project

Since there was not a specific person allocated to perform the duties of an IT business partner in Squad Road, the developers were facing a severe work overload, which eventually compromised their workflow.

As stated above, the squad's operations are based on collaboration with the road freight's functional team to guarantee the optimal performance of the company's systems. These operations are currently managed with Scrum, an agile methodology that organizes the team's work by allocating tasks to constant periods, also known as sprints, during which those items must be completed.

Nonetheless, besides providing support to the system's users, which depending on the severity of the problems, can occupy a big portion of their time, the developers also perform a significant number of tasks that are not managed and properly prioritized with the Scrum methodology.

This fragmentation of tasks results in a low sprint completion rate, i.e., most of the tasks allocated for each sprint, which *a priori* should be the programmers' priorities, are not completed in the designated period. In fact, most tasks are carried over several sprints, which is not only detrimental in terms of delivering value and providing a high service level but is also a violation of the Scrum principles. In turn, this disorganization also has an impact on the provision of support.

Finally, due to the team's work overload and lack of coordination, programmers are unable to properly document their projects, particularly the new features or changes they have made to the system's integration.

1.3 Project goals

The primary objective of the project that originated this dissertation was to integrate the organization and the team in which the project took place. This included integrating the team's workflow and projects, as well as understanding the team's technological stack.

As the project aims to redefine the operations of the road freight squad, another objective is to model its baseline situation, while identifying opportunities for improvement. At a later stage, the focus shifted to developing strategies to address these opportunities, while redefining the squad's processes, i.e., the TO BE modeling.

Nonetheless, one of the goals was to quantify the squad's success rate with its core processes, to complement the indicators that were already implemented, which mainly addressed the squad's support processes. Finally, since the project had a highly qualitative component, the last goal is related to the materialization of the proposed solutions and the subsequent viability analysis.

1.4 Approach adopted for the project

The work that was performed in the context of this project was organized in a chronogram, which is displayed in Figure 2.

The first weeks were dedicated to onboarding, to better grasp the operations of all the interconnected companies, as well as integrating into the team. During the following weeks, in-depth research was conducted on the freight forwarding industry, namely on the relevance of IT systems in this sector, as well as the agile methodologies used to organize and manage the team's workflow. This phase culminated in the preparation and delivery of a preliminary version of the theoretical framework supporting this dissertation.

During the period in which this phase took place, the team, namely the road freight squad, was carefully observed and accompanied, to document its activities. This was also crucial to better understand the systems used by RT and their integration. Lastly, this phase culminated in the modeling of the squad's processes, while preparing the baseline situation that originated the present dissertation.

In turn, the AS IS modeling enabled the identification of opportunities for improvement and subsequently the definition of goals that would guide the design of solutions, which were developed over the following weeks. These solutions were encapsulated in the TO BE modeling of the newly proposed processes.

Lastly, a revision of the work performed during this project was conducted to prepare the final version of this dissertation.

Activities	Start	End
Integration in the company	1	1
Integration in the IT transport team	1	2
Theoreticalframeworkand methollodgy		
Industry research	2	3
Metholody research	2	6
Theoretical framework (preliminary version)	2	7
Assessment of the initial situation		
Accompanying the team	2	5
AS IS modeling	5	7
Baseline situation (preliminary version)	2	7
Project definition		
Identification of opportunities for improvement	6	9
Definition of goals	9	10
Development and implementation of solutions	11	17
TO BE modeling	18	19
Revising the project		
Analysis and review of the work performed	18	20
Delivery of the final version	21	21
Dissertation presentation	22	24

Figure 2 – Gantt chart representing the project's schedule

1.5 Dissertation structure

The present dissertation is organized into five chapters, the current one being a brief introduction and contextualization of the project.

Chapter 2 covers the theoretical framework and the methodologies adopted throughout this project.

Chapter 3 corresponds to the presentation of the project's baseline situation, in which the AS IS modeling of the road freight processes is presented.

In turn, Chapter 4 presents the TO BE modeling of the processes, which is meant to reflect the solutions that were designed to materialize the previously identified areas of improvement. Moreover, given the qualitative nature of the project, this chapter includes metrics that were calculated with the purpose of analyzing the viability of the proposed solutions.

Lastly, Chapter 5 encompasses the main conclusions of this project, as well as opportunities to further develop this project within the IT transport team.

2 Theoretical framework

The purpose of this chapter is to present a theoretical framework of the concepts and methodologies described throughout the dissertation and to contextualize the work that was performed.

Initially, the importance of using Information Technology (IT) in the road freight industry is highlighted. An overview of middleware technology and its relevance to logistics operators is also provided. This section is followed by the presentation of the concept of agile project management, as well as relevant methodologies, namely Scrum and Kanban, and their importance for managing software development.

Finally, a description of business process modeling in the context of IT and its importance to agile software development is presented. Lastly, there is a brief description of the Kaizen methodology, as well as the difference between lagging and leading metrics.

2.1 Utilization of information technology in the road freight industry

The main purpose of the theoretical context presented in this subchapter is to contextualize the sector in which this project was developed. Thus, this section does not present specific methodologies that were used throughout the dissertation but rather contextualizes the work performed while emphasizing the importance of IT in this industry.

2.1.1 Freight forwarding industry

Logistics can be defined as the process of planning, implementing, and controlling the efficient flow and storage of goods and information. Currently, logistics operations are becoming increasingly more demanding, which explains the growing tendency for companies to outsource their activities. (Baluch, 2005)

There are numerous factors that companies must take into consideration when planning their supply chain activities. Firstly, the externalization of some operations, such as warehousing, inventory management, and distribution, allows companies to focus on their core activities (Baluch, n.d.). Thus, by outsourcing, the company can focus on processes directly related to the provision of products and services to the customer, i.e. processes that are part of the company's value chain (Faria, 2022c). Nevertheless, externalizing these logistics activities provides the company with a broader reach and enables integration in foreign markets, without being constrained by the intrinsic details of those markets (Baluch, 2005).

In the supply chain, freight forwarders represent intermediaries between the company (that wants to transport its products) and the shipping and transportation company, which will collect and deliver the products. (Munro, 2023)

Therefore, establishing partnerships with freight forwarders benefits companies by avoiding constraints that could slow down the shipping process, particularly at a legal and financial level. Freight forwarders also manage all the documentation related to transportation, as well as the documentation related to the customs process, which can be quite bureaucratic and complex (Munro, 2023).

The usage of freight forwarders also has cost advantages. Since freight forwarders have a network of reliable carriers, they can negotiate more competitive prices for their customers. At the same time, most freight forwarders offer the possibility of consolidating loads, i.e. grouping several customers' loads in a single shipment, to reduce shipping costs (Munro, 2023).

Finally, freight forwarders manage the inventory during the shipping process and monitor the goods to their destination (Munro, 2023).

2.1.2 The significance of information technology

Given the complexity of freight forwarders' operations, it is essential that they are flexible and can adapt to their customers' needs. Today, increased competitiveness is intrinsically linked to the use of information technology and its sophistication. To guarantee the successful implementation of these technologies, there must be prior standardization of all processes, as well as an integrated database for the entire supply chain. Then, it will be possible to simplify daily operations and streamline the entire process. (Baluch, 2005)

Nonetheless, the usage of IT solutions simplifies the creation of documents (as well as their management), facilitates the exchange of information and communication for shippers and carriers, and enhances the consolidation of information in a single system. (AltexSoft, 2022)

Typically, these systems include order and inventory management, load planning, and shipment tracking. They also include invoicing and document management systems. Nevertheless, they enable the creation of reports and analysis and simplify customs and regulatory compliance. Finally, this type of software facilitates the integration of a customer portal, where customers can track their cargo and check the flow of the process. This visibility and transparency towards the customer constitute competitive advantages that emphasize the flexibility and efficiency of the freight forwarder. (AltexSoft, 2022)

2.2 The usage of middleware technology in system integration

Similarly to the previous subchapter, this section does not present the methodologies that were used throughout this dissertation. Thus, the purpose of including the theory behind systems integration is to provide a technical background of the transport team’s operations. These concepts will then be addressed whilst presenting the team’s operations, namely their technological stack.

2.2.1 System integration

According to Rajabalinejad et al. (2020), integration can be defined as the activity that enables the constitution of a unified system that combines the elements of several systems (including those owned by different organizations) and their interfaces, thereby promoting their interaction and interconnection. Systems can be integrated internally and externally. Internal integration occurs when the elements built for each system, including the interfaces, are integrated to form a functional subsystem or system. External integration occurs when the system is implemented in an operational environment.

Failures related to integration generally occur in the early stages of implementation, but several other problems only arise in later stages. (Rajabalinejad et al., 2020)

As systems integration is a complex and often iterative process, involving corrections and the constant implementation of improvements and new features, the software is delivered to users at different moments, which are carried out in several environments. Firstly, there’s the development environment, where developers write the code, and the integration environment, where the new code is combined and validated with the existing code. Next, there is a testing environment, in which functional and non-functional tests are conducted on the new code, to check that the pre-established requirements are met. Finally, the software is deployed into a productive environment and released to its users. (Mathenge, 2020)

The iterative nature of software development and, in this case, system integration, encompasses different types of maintenance actions. There are adaptative actions, that are performed when the software needs to fulfill new requirements, such as the integration of a new application. There are also corrective actions, which aim to correct bugs or any other elements in the code that are not functioning properly. Finally, there are evolutive improvements that are executed to implement new features or to optimize existing ones. These actions are usually prompted by requests or observations from the software users. (Chapin et al., 2001)

2.2.2 Middleware technology

The employment of viable software solutions by road freight operators capable of fulfilling the needs of their various operations constitutes a major challenge. The usage of multiple IT systems means that human operators play a central role in handling and processing data. This translates into heterogeneous planning and fragmentation problems, due to the existence of different data formats and the need to manage communications between systems.

To simplify and automate the work of the operators, systems are integrated into a central system called a TMS (Transport Management system). (Carlan et al., 2020)

The implementation of TMS leads to increased productivity in operations since workers no longer need to manually enter and synchronize information on the various platforms. This way they can concentrate on the company's core activities. The platform also allows for a more sophisticated analysis of information, leading to more informed decision-making. It also translates into growth opportunities for the organization, as it facilitates communication with external and international organizations. (Dias Neto, 2023)

Usually, the TMS is developed by external organizations that specialize in developing this type of software, so its implementation leads to problems related to integration with existing systems. Nonetheless, companies usually also develop customized functionalities to suit their activities. (Harfoushi & Obiedat, 2018)

It is therefore necessary to use middleware software that enables the efficient integration of the organization's applications, i.e. that enables the transmission of information between two or more applications. This process of combining various software applications, systems, and databases into a single system is known as EAI (Enterprise application integration).

It should be noted that although the concepts of EAI and middleware are related, they are distinct concepts. While EAI is a specific use case of this type of software, middleware encompasses a broader spectrum of tools that facilitate the exchange of information between applications. (Harfoushi & Obiedat, 2018)

One of the most widely used middleware solutions is known as MOM (Message-oriented Middleware), which acts as an intermediary between heterogeneous systems and platforms, allowing them to exchange information via messages. MOM works as a message mediator, managing information flows between platforms through modules that shape communication so that it is compatible with the respective platform. (Yongguo et al., 2019)

The use of MOM has several advantages, including the fact that it simplifies the work of developers by providing a standard way of handling the flow of information between the various applications. It also establishes asynchronous communication between applications, thus enabling the exchange of messages without the need for direct and continuous communication between them. Finally, it provides reliability, flexibility, expandability, and even greater fault tolerance, guaranteeing efficient integration. (Yongguo et al., 2019)

2.3 Agile Project Management

This subchapter contemplates the concept of agile project management, a methodology that was used throughout this dissertation, namely whilst addressing the team's core activities. This project management technique will also be presented in the context of software development while providing a framework for the team's operations. Nonetheless, two agile methodologies are presented, namely Scrum which was already established and integrated into the team, and Kanban, a methodology that was later implemented.

2.3.1 Definition

Project management is a process that encompasses the planning and implementation of the activities that need to be carried out during a project to achieve the requirements. It also includes monitoring progress and performance (Watt et al., n.d.). In turn, a project is a temporary, unique endeavor with a specific objective. (Faria, 2022b)

Typically, a traditional project management approach is more oriented towards stable projects with well-defined requirements. Thus, it focuses on linear and predictable activities, which allows the standardization of the planning, scheduling, and control phases, while providing structure to project management. It applies to larger teams, whose employees have a similar set of skills, in companies with a well-defined hierarchical structure. Usually, the final customer is not involved in the various phases of the project. (Agbejule & Lehtineva, 2022).

On the other hand, agile project management is based on an iterative and adaptive project life cycle. As such, it focuses on ensuring flexibility and adaptability, to respond quickly and effectively to the persistent changes that the project contemplates. (Pinton & Silveira Torres Jr., 2023)

This approach is based on cooperative work, without a traditional hierarchy of command, so it applies to smaller teams with multidisciplinary skills, i.e. with a holistic and complete vision. It also applies to teams where there is strong interaction and collaboration with the project's stakeholders, to maximize the value created for them (Agbejule & Lehtineva, 2022). These teams are also self-managed and autonomous, in which the tasks are delegated, assigned, and defined internally. (Sassa et al., 2023)

2.3.2 Overview of the Scrum methodology

Definition and framework of the agile methodology

According to Schwaber & Sutherland (2020), Scrum is a flexible and adaptable project development methodology based on lean thinking, i.e. reducing waste and redundancies. It is also based on empiricism, which states that knowledge comes from experience and decision-making. Therefore, it is an agile project management methodology that can be applied to complex unpredictable projects. (Breyter, 2022)

Scrum is based on the principles of transparency, inspection, and adaptation. Transparency means that the team has visibility of all the work that needs to be performed and of all the processes inherent to it. Inspection, on the other hand, ensures that the previously defined objectives are met by detecting any complications that may arise. Finally, teams must possess the ability to adapt and overcome any constraints as fast as possible to minimize further deviations. (Schwaber & Sutherland, 2020).

Due to the dynamism of this methodology and the fact that it focuses on solving short-term problems, there must be an atmosphere of cooperation and mutual help within the team, which in turn is reflected in increased productivity. Scrum is also particularly important in managing project uncertainty while focusing on maintaining a constant and efficient communication channel with the client. (Sassa et al., 2023)

Processes and phases

Due to the complexity of projects, the Scrum methodology is developed in standardized processes, which are organized according to the respective stage of the project's life cycle. This structure is achieved through the existence of Scrum events, in which the Scrum artifacts are verified and eventually adjusted. In turn, scrum artifacts are representations of the work done or to be done, which quantify the progress of the work. (Sassa et al., 2023)

Figure 3 summarizes the Scrum framework, including the Scrum Team, the artifacts, and the events, i.e. the meetings that take place during the project lifecycle.

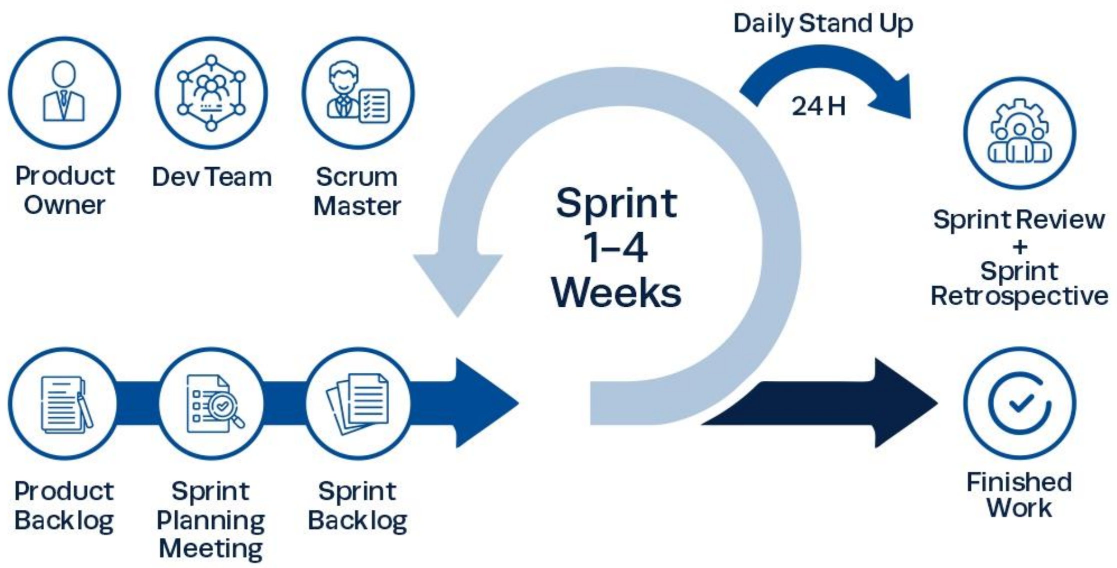

Figure 3 – Scrum Framework: in “https://projectmanagement.ie/blog/understanding-scrum/, consulted in 2024-04-22”

Thus, at an early stage of the project, the product owner creates the project vision, which specifies the main problem or need that the project seeks to fulfill. Next, the Scrum Master is identified, whose main function is to ensure the effectiveness of the team by guaranteeing the implementation of Scrum practices and removing any constraints. This person is also responsible for producing documentation. Thus, the Scrum team is formed, which is also made up of developers. (*A Guide to the Scrum Body of Knowledge (SBOK Guide)*, 2016)

Once the team has been created, the product owner, with the help of the Scrum Master, develops the epics, i.e. the general requirements of the project. Thus, they compile the product backlog, which translates clearly and objectively the stakeholders' requirements into a prioritized list of work items. The product goal is then established, and a timetable is drawn up with the planned deliveries. (Schwaber & Sutherland, 2020)

In the second phase of planning and estimating the work, the team collaborates on creating user stories, which originate from breaking down the epics into objective and less complex project requirements, followed by their approval and confirmation. Subsequently, tasks are created, which in turn correspond to decomposing the user stories into less complex and more manageable items. The work is organized into short intervals of time called sprints, which are preceded by a sprint planning ceremony in which the team prepares the sprint backlog, a document that combines the tasks for that sprint. This forecast is mainly done by the Developers since they are the people responsible for completing the tasks to achieve the product goal. (*A Guide to the Scrum Body of Knowledge (SBOK Guide)*, 2016)

To deliver a more realistic estimate, it is possible to calculate story points, which are an abstract unit of measurement used to approximate the amount of work required to complete a given task, including items on the product backlog. Their allocation is determined based on the complexity, volume, and unpredictability of a task, thus providing a more accurate, flexible and objective estimate of effort in comparison to time-based estimates. Furthermore, this metric results in better planning, prioritization, and predictability in agile software development projects. (*A Guide to the Scrum Body of Knowledge (SBOK Guide)*, 2016)

This is proceeded by an implementation phase, in which developers produce increments, which are sets of features. There are multiple increments within the same Sprint, so it is necessary to ensure that they are properly integrated. To check the project's progress, the team organizes daily meetings, also known as Daily Scrum, in which they discuss the completion of tasks and aspects that may hinder the production of increments. Thus, the scrum master should formally record these impediments in an Impediment Log, to identify them and subsequently deal with them. (*A Guide to the Scrum Body of Knowledge (SBOK Guide)*, 2016)

At the end of each Sprint, there is a sprint review in which the whole team and the stakeholders analyze the obtained results and, if necessary, adjust the Product Backlog. During the final event, known as Sprint Retrospective, the team reflects on the work that was completed and identifies opportunities for improvement. At the end of the project, the work performed is delivered. (Sassa et al., 2023)

Figure 4 summarizes the previously described processes and phases of the project lifecycle using the Scrum methodology.

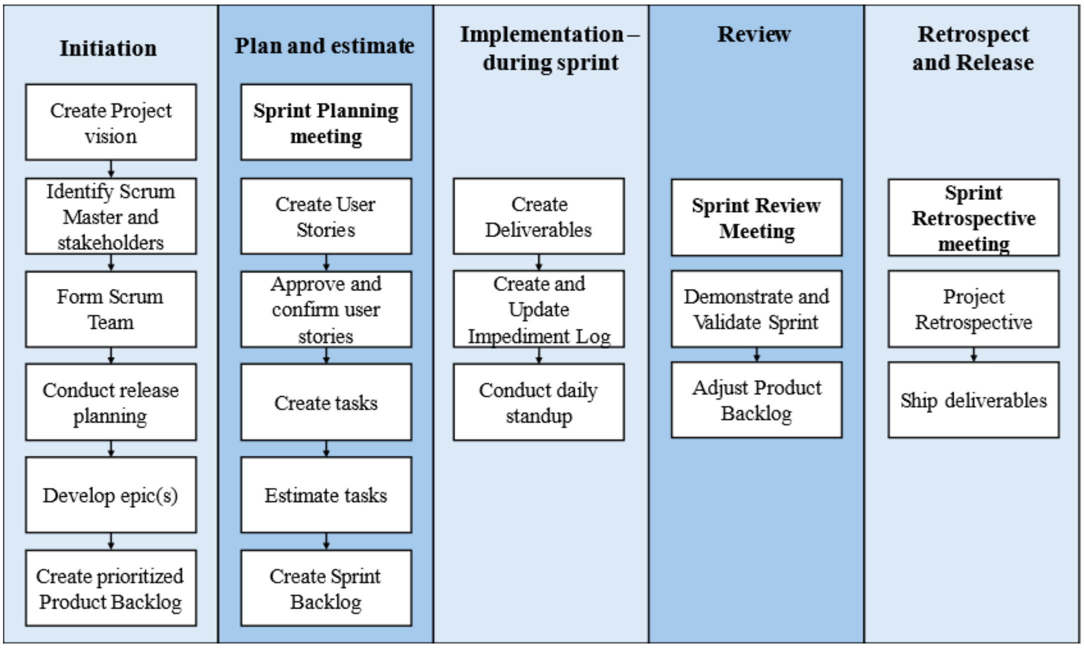

Figure 4 – Scrum processes and phases adapted from SCRUMstudy, A Guide to the Scrum Body of Knowledge (SBOK Guide), 2016

Scrum in Agile software development

The Scrum methodology is particularly relevant for managing the work of software development teams. According to Verwijs & Russo (2023), the effectiveness of Scrum Teams depends on multiple factors, such as the ability to establish strong collaboration with the stakeholders and consequently the ability to quickly respond and adapt to their requirements. This can be achieved through the definition of clear and practicable Sprint goals and the performance of high-quality Sprint Reviews. Nonetheless, the team must establish an environment that fosters and encourages continuous improvement. The last factor is related to the team's autonomy, which allows them to be flexible and responsive to the changing customer requirements.

Since software quality is a difficult concept to measure, Scrum users are driven by different notions of quality. External quality is expressed through compliance with established requirements imposed and the absence of defects. Internal quality is related to the sustainability of the software design, i.e. its continuity and accessibility. (Alami & Krancher, 2022)

Scrum ensures the fulfillment of both internal and external quality, through the promotion of values such as collaboration and exchange of knowledge, as well as transparency and responsibility. This supportive and positive environment fosters a better understanding of the requirements, which leads to fewer errors. Scrum also has advantages on a technical level, namely the emphasis on iterative work and the execution of formal inspections that ensure processes' optimal flow. (Alami & Krancher, 2022)

However, it is important to emphasize that the usage of Scrum as an agile software development tool does not replace practices related to quality assurance, process control, and optimal software development. Scrum complements these practices while promoting a positive work environment, which in turn fosters not only the fulfillment of requirements but also continuous improvement. (Alami & Krancher, 2022)

2.3.3 Overview of the Kanban methodology

Definition and concepts

Kanban is an agile methodology based on continuous improvement, flexibility, and optimization of the workflow. Its core principles include the gradual improvement of the current workflow and the maintenance of the current roles and responsibilities while promoting transparency and visibility of the team's current responsibilities and tasks. (Petricioli & Fertalj, 2022)

Similarly to Scrum, Kanban also utilizes a board, with different columns, that correspond to different stages of work, in which the team places its current tasks, also known as Kanban cards. However, to improve the team's efficiency and to avoid bottlenecks, each column can only hold a limited number of kanban cards, which is defined by the work-in-progress limit (Kirovska & Koceski, 2015). Nonetheless, the team can adopt a metric known as lead time, which quantifies the time from the creation of a Kanban card until its resolution, while going through the previously defined workflow stages. (Petricioli & Fertalj, 2022)

Comparison to Scrum

Unlike Scrum, the Kanban methodology employs a continuous workflow to complete tasks, rather than defined periods, namely sprints, which result in a more iterative workflow. (Salleh & Nohuddin, 2019) Nonetheless, there are no defined roles for the team, nor events for planning and reviewing tasks. In conclusion, Kanban is a more flexible strategy, while Scrum is more structured and prescriptive. (Brezočnik & Majer, 2016)

However, as previously mentioned, both are agile and adaptable methodologies that embrace change and emphasize working software over comprehensive documentation. (Kniberg, 2009)

2.4 Business Process modeling in the context of IT

Definitions

A process is a sequence of repetitive and thus recurring activities that are performed according to a standardized procedure. (Faria, 2022b). Additionally, a business process model is a representation of the internal workflow of activities intrinsic to the process, which is key to analyzing and integrating the company's operations. Thus, the business process must be modeled correctly and effectively, while providing a comprehensive representation of the process. (Aguilar-Savén, 2004)

Contextualization

Following the latter chapters, the purpose of this section is to provide a theoretical framework for the methodology that was chosen to model the team's processes, also known as multilevel modeling. The importance of modeling the team's business processes is also highlighted.

Nonetheless, as the simultaneous use of process modeling and agile methodologies can be perceived as counter-intuitive, arguments in the literature are exposed to demonstrate how process modeling can be used as an instrument to support agile software development projects.

Lastly, it presents the concept of Kaizen, a methodology that is currently used in the team to promote incremental improvement in their operations. Nonetheless, to measure the team's outcomes and the impact of the implemented improvements, there is a contextualization of performance indicators, namely leading and lagging indicators.

2.4.1 Business Process to Information technology and Multilevel modeling

To incorporate the complexity and specifics of Information Technology, BP2IT or Business Process to Information Technology aims to integrate an enterprise's business processes with IT. Therefore, this structured approach is based on the standardization of a company's IT-related activities, to enhance overall collaboration and effectiveness, since this methodology provides all the requirements to develop the systems, thus reducing the time it takes to develop them. It begins with a comprehensive review of the processes, followed by the creation of models to support their analysis. Finally, the specific IT requirements are defined, which in turn enables the assessment of the processes' performance. (Pinheiro, 2004)

In the first phase, after a global analysis of the company's activities, it is necessary to construct a cohesive and complete process map, in which the company's processes and their integration will be represented. These processes can be divided into three categories, namely core, support, and management. The core processes are related to the provision and delivery of products or services to the clients and therefore are part of the company's value chain. The support processes are not directly related to the creation of value for the company, but they support the company's main activities. Lastly, the management processes are responsible for major business decisions and represent the actions that originate important decisions. (Faria, 2022c)

As previously mentioned, the second step involves the formulation of models that reflect, clearly and simply, the entirety of the process's workflow, including its phases, participants, and outputs. A widely adopted methodology to model the processes is known as multilevel modeling. To represent the whole process, without being too complex or too simplistic, this approach divides the modeling into three different levels, each representing an increasing level of detail. (Faria, 2022a)

The first level is usually modeled with responsibility matrixes, which represent all the process's phases and its outputs, as well as the actors that participate in each phase. The actors that intervene in each phase can be classified according to their level of involvement. For example, according to the RACI approach, in each phase, actors can be responsible, accountable, consulted, or simply informed. However, this level does not provide the specific tasks that each actor performs. (Faria, 2022a)

On the other hand, the second level provides a higher level of detail, through the representation of the tasks that each actor performs during a certain phase of the process. This representation is usually done via swimlane diagrams, one per each phase. In this type of diagram, there are multiple lanes, namely one per actor, in which their tasks are placed. (Faria, 2022a)

Lastly, the goal of the third-level models is to provide specific details and information that are not clarified in the second level. Thus, this level presents instructions targeted to a given task, within a phase of the process, which can be represented in several ways, namely through flowcharts or swimlane diagrams. (Faria, 2022a)

The third step of the BP2IT approach is based on the analysis of the company's current workflow, to identify opportunities for improvement. This step is known as AS-IS analysis. Subsequently, there will be a redesign of the processes, also known as TO-BE modeling, to materialize the opportunities that were previously identified into an optimized version of the workflow. Nevertheless, these new models should be constructed using the same structured approach that was used in the AS-IS models. (Faria, 2022a)

Following the modeling and analysis of the processes, it is crucial to integrate the new models with the company's information systems by specifying the changes that will be implemented and, if necessary, the requirements needed to deploy them. The integration will be optimal if the IT systems can interact correctly with the sequence of activities corresponding to the company's business processes. In turn, it will lead to the optimization of the workflow and the worker's productivity, as well as the reduction of the costs associated with the processes. (Pinheiro, 2004)

2.4.2 Significance in Agile Software Development Environments

Process modeling can be used as an instrument to support agile software development projects. At first, it might seem counter-intuitive, since agile methodologies prioritize working software, agility, and adaptation over specification and documentation, the phases in which conceptual models are typically used. However, process modeling is particularly relevant for the first phases of a project's lifecycle and can have multiple uses. (González Moyano et al., 2022)

Firstly, this methodology is highly relevant in the definition of requirements, since it provides a structured approach to the modeling of the business processes, followed by the identification of the specifications, which can be functional or non-functional (Azevedo Junior & Campos, 2008). On the one hand, the functional requirements express the business needs, which will then define the behavior of the system. On the other hand, it is important to assess the non-functional requirements, which entail quality characteristics that are independent and not specified by the functional specifications.

Nonetheless, González Moyano et al (2022) specifies other uses of process modeling in an agile software development environment, namely its crucial role in engaging the stakeholders. Besides fostering their collaboration in the definition of requirements, it provides visibility of the project's scope and supports their understanding of the software development processes. Moreover, it supports the phases corresponding to the testing and validation of the system, in which stakeholders are typically involved.

In addition, process modeling simplifies software development, which is usually very complex, by supplying a model that should be followed during the implementation of the system. This model, along with the creation of templates that reflect common processes, avoids cases of replication. It also enables the understanding of the existing relations between the systems' functionalities and how they relate to the business process activities. (Imran & Soomro, 2022)

The usage of models also supports the performance of software maintenance activities, such as the correction of bugs, the development of new features, or even the provision of assistance to the stakeholders. (Kirovska & Koceski, 2015)

Finally, process modeling promotes the utilization of reverse engineering practices to determine how the system and its relationships were implemented, which in turn enables the identification of improvement opportunities. (Imran & Soomro, 2022)

In conclusion, although process modeling and agile software development might seem incompatible approaches, their integration provides several advantages, namely the reduction of the project's time, the improvement of its quality, and the increase of stakeholder collaboration. (Breyter, 2022)

2.4.3 The usage of Kaizen and performance indicators

Kaizen definition

Kaizen, according to its etymology, means gradual and continuous progress, as well as improvement and increased value. Thus, Kaizen focuses on maintaining the current working conditions, in the sense of complying with pre-established technology and management standards while, on the other hand, gradually improving the current conditions. (Karkoszka & Honorowicz, 2009)

Thus, this methodology is an incremental approach, based on small improvements, that concentrate mainly on the process and people, rather than the results. By doing so, organizations can carefully analyze and identify opportunities for improvement, followed by their implementation. They also check how the actions are progressing and whether they are contributing to the organization's continuous improvement. Finally, effective action is taken following what was defined in the planning phase, developing new standards. (Dalton, 2019)

Kaizen improves not only work processes and standards but the organization as a whole. This methodology is based on suggestions and questioning of current procedures, with a focus on workers' autonomy and critical thinking. It involves self-discipline, practical learning, and knowledge sharing. Nevertheless, it involves the adoption of basic standards for carrying out the work and the definition of targets to quantify the improvements. (Al-Baik & Miller, 2016)

This can be accomplished by the implementation of Kaizen events into the employees' routines, which are short daily or weekly sessions, in which the team can reflect and discuss ideas for implementing continuous improvement solutions in previously identified areas. (Dalton, 2019)

According to Al-Baik & Miller (2016), software development teams must hold a periodical analysis of their current work, mainly to identify opportunities for improvement. However, since the Kaizen methodology cannot be directly established in software development teams, due to its lack of clear definition and concepts, the team should adopt a critical stance and seek other methodologies to organize and continuously improve their activities.

Performance indicators

As previously mentioned, a team's current performance and the effectiveness of the implemented improvements can be measured by a range of performance indicators. On the one hand, the team can use indicators designed to predict and influence future outcomes, also known as leading indicators, which can be essential to guide a team's daily activities. On the other hand, lagging indicators are based on past outcomes, i.e., they provide insights into the organization's previous performance, which in turn is essential for the definition of long-term strategy. (Watts, 2019)

In conclusion, to maximize its performance, an organization should consider both leading and lagging indicators. Improving the leading indicators, i.e. continuously improving the team's short-term results, will ultimately lead to better outcomes, which will translate into improvements in the lagging indicators. (Watts, 2019)

3 Contextualization of the baseline situation AS-IS

The purpose of this chapter is to describe and contextualize the problem that originated this dissertation.

After a brief and general description of the problem, the AS IS modeling of the road freight squad is presented. The purpose of the models is to reflect the squad's activities and operations, which in turn enables the identification of inefficiencies and opportunities for improvement.

3.1 Problem Description

Currently, Rangel is constituted by a group of intertwined companies that operate in several markets across the world. To maintain its competitive advantage and to further expand its operations, the group must invest in developing and maintaining its IT systems. Therefore, given their complexity, they are prone to continuous improvement, which is reflected in the work performed by the IT department.

The theme that originated the present dissertation is related to the workflow of one subunit of the IT department. In particular, the work was developed in the context of the team dedicated to Rangel's freight transport systems, namely the squad dedicated to road freight operations. Currently, this squad is composed of two people, one of which is not entirely dedicated to the squad, since they accumulated several other roles and responsibilities. Thus, this dissertation's goal is to implement a more efficient and structured workflow.

The goal is to streamline the squad's operations across their areas of intervention, by identifying and addressing inefficiencies in their processes. These can be primarily found in the squad's core activities, which include the participation in projects that seek to improve the quality and resilience of the existing systems, either through the development of new features or through the integration of new partners into the systems. Nonetheless, there are also inefficiencies in the processes related to the provision of support to the system's users.

In summary, the work undertaken within this dissertation entailed modeling the squad's current processes, analyzing the flow of operations and, consequently, identifying opportunities for improvement. In particular, the purpose of process reengineering was to increase the squad's productivity and ultimately to increase the level of service provided to the stakeholders.

3.2 AS-IS Process Modeling

This subchapter contemplates the AS-IS modeling of the squad road's operations while representing its current operations and workflow. The process modeling was performed iteratively. The first drafts were produced during an initial phase, which consisted of observing and accompanying the squad road's daily activities, namely during the daily scrum meetings. After this phase, follow-up meetings were also held, to address specific aspects that weren't clear or well-defined. After the models were complete, they were revised to ensure that they were an accurate representation of the squad's operations.

The entirety of the AS-IS modeling that was conducted can be found in Appendix A. In this subchapter, there will be a presentation of the diagrams in which opportunities for improvement were identified. Considering the complexity of the multi-level modeling, the purpose of Table 1 is to facilitate the navigation through the processes and their phases.

Table 1 – Index containing the diagrams that represent AS IS modeling

Process	**Modeling**			
	First level	**Second level**		**Third level**
Project w/partner	Figure 6 and Figure 3 in Appendix A	Initiation	Figure 7 in Appendix A	Development of increments: Figure 11 and Figure 10 in Appendix A Update the Scrum Board: Figure 12 in Appendix A
		Sorting	Figure 14 and Figure 5 in Appendix A	
		Scrum – plan and estimate	Figure 9 and Figure 8 in Appendix A	
		Scrum – implementation with stakeholders	Figure 10 and Figure 9 in Appendix A	
		Scrum - review	Figure 12 and Figure 11 in Appendix A	
		Delivery	Figure 13 in Appendix A	
Incremental improvement	Figure 7 and Figure 2 in Appendix A	Identification and Placement of new action	Figure 14 in Appendix A	Analysis of request: Figure 15 in Appendix A Development of increments: Figure 11 and Figure 10 in Appendix A Update the Scrum Board: Figure 12 in Appendix A
		Sorting	Figure 14 and Figure 5 in Appendix A	
		Integration in the Product Backlog	Figure 16 in Appendix A	
Corrective action		Scrum – plan and estimate	Figure 8 in Appendix A	
		Scrum – implementation with operations	Figure 13 and Figure 17 in Appendix A	
		Scrum - review	Figure 12 and Figure 11 in Appendix A	
Business support and incident handling	Figure 8 and Figure 1 in Appendix A	Placement of new request	Figure 4 in Appendix A	
		Sorting	Figure 14 and Figure 5 in Appendix A	
		Problem resolution	Figure 6 in Appendix A	

In general, the modeling was not supported by existing documentation. The only exception was the process “Business support and incident handling”, for which there was a document with the work instructions to sort and solve a request in the helpdesk platform. However, this document is intended for the IT department, regardless of the team. Thus, the purpose of the models is to accurately represent the workflow specific to the transport team, which differs from the previously mentioned documents.

Per the methodology chosen for process analysis throughout this dissertation, firstly, the squad’s process map will be presented.

Secondly, there will be an analysis of both core and support processes, which were modeled using a multi-level approach. Each process will be presented with 2 levels of detail, the first level being a responsibility matrix, that shows the process’s phases and its actors. The second level will be represented by a swimlane diagram, that shows the workflow and the tasks of each phase, as well as its participants. Some tasks will also contemplate a third level of modeling, depending on the relevance of representing a higher level of detail.

It is important to note that the diagrams and maps are abstract models that seek to represent the squad’s current situation as realistically and faithfully as possible. However, the activities that are not directly related to the squad or the transport team were simplified, considering that they will not be the subjected to improvements. The sole purpose of its inclusion is to contextualize the team's operations and related activities.

3.2.1 Squad Road Process Map

Overview of the squad’s operations

As previously mentioned, the team works on the information systems used by road freight operators. In particular, the squad road deals mainly with systems integration, namely the integration of the applications used by Rangel’s companies and the Transport Management system, also known as LSP (Logistic Service Platform). It´s important to mention that this platform was not specifically developed by Rangel’s IT department. Instead, it was built by a specialized software development company and subsequently adapted to the group’s operations.

This platform is used by operators in their daily activities, such as the management of loads, storage and transportation, and registration of suppliers and customers. Therefore, since the LSP is a crucial element of the road freight business, and considering that it wasn’t developed in-house, it is essential to guarantee its efficient integration with the other applications used by the group.

To adapt the LSP to the changing needs of the road freight business and to optimize specific operations, the squad is continuously improving the platform, namely through the development of new features. Thus, the team creates applications and modules that are developed according to the processes that reflect the user’s daily operations. The process modeling is performed by another team, also known as the functional team of Rangel’s Road freight business, which is led by a Product Owner. Nonetheless, the squad supports the system’s users whenever they face an error due to a bug in the code or when the operators have doubts about how the system works.

The system integration is achieved with middleware technology, namely MOM, to guarantee communication between the LSP platform with internal and external partners. Notwithstanding, the optimal system integration also entails that the other applications are functioning properly. Currently, the squad maintains and supports, whenever necessary, applications related to external partners, namely suppliers.

In addition, the team is focused on the development of another application, which is a central piece of the express transport business. Moreover, this application is crucial for the management of shipments and collections in the national territory, while covering the entire shipping process, with features such as real-time tracking of loads and access to proof that the load has been delivered. Currently, this application has been developed to integrate new agents, i.e. freight forwarding partners located outside the national territory, to expand Rangel's business volume. Nonetheless, the squad also strives to optimize the application's functional capabilities and to improve its resilience and effectiveness.

Process map

As previously mentioned, the squad partakes in various activities, across several systems, which can be modeled into two different types of processes. These are presented in a structured process map, specified in Figure 5.

Firstly, the core processes create value for the road freight business and contribute to the expansion of their activities. These processes occur whenever there's a need to develop a project with Rangel's stakeholders, for example, when it is beneficial to integrate the partner's system into the group's systems, to optimize the information flow. On the other hand, the squad also works towards improving the road freight systems, to increase the productivity and efficiency of the operations, for example, through the development of a new feature.

Secondly, the squad operations also include support processes, which aim to assist the information system's users in their daily operations. These processes include the performance of corrective actions, whenever there's a bug in the system's code. Moreover, the squad also assists the business via a more immediate and direct channel, not only through the clarification of the user's doubts but also through the resolution of incidents.

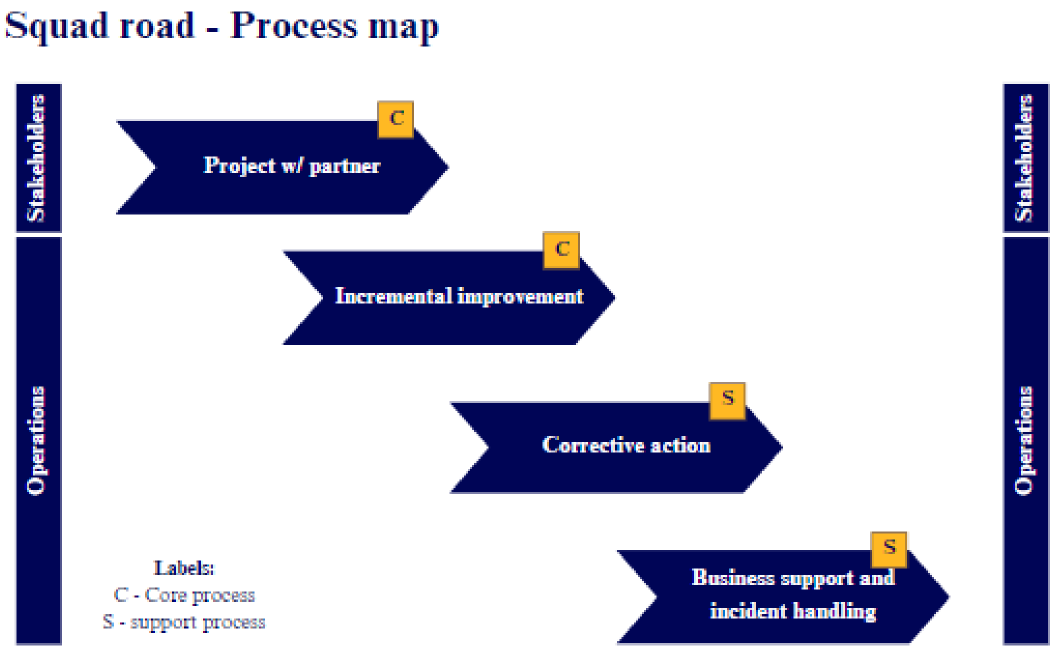

Figure 5 – Squad Road process map

3.2.2 First-level Modeling

Core processes

As previously mentioned, the squad's core activities are centered around the creation of value for Rangel, which can be achieved through the development of unique projects in collaboration with the firm's partners or through the improvement of the existing systems.

A priori, a project cannot be modeled in a similar way that a process is, due to its unique nature and specificities. However, since the squad's core activities are managed in a structured and uniform manner, the development of projects was modeled as a process. As depicted in Figure 6, the development is conducted in 6 different phases.

The first phase, also known as "Initiation", entails the follow-up of the idea that originated the project. The second phase, designated "Sorting", is common to all the processes that describe the team's operations and will be further presented in the subchapter regarding the support processes. The third and fifth phases are related to the application of the scrum methodology in the planning and review of the project's tasks. These phases are common to the processes of "Corrective action" and "Incremental improvement". The fourth phase corresponds to the implementation phase, i.e., the performance and completion of tasks and user stories, and the creation of increments.

It should be noted that the phases that involve the usage of the scrum methodology occur in a non-linear, cyclic manner, given that a specific project will not be completed in a single sprint. A project (or even an improvement action) will probably be developed in several sprints, which means it will undergo the third, fourth, and fifth phases cyclically until it is ready for the final phase, which corresponds to its delivery.

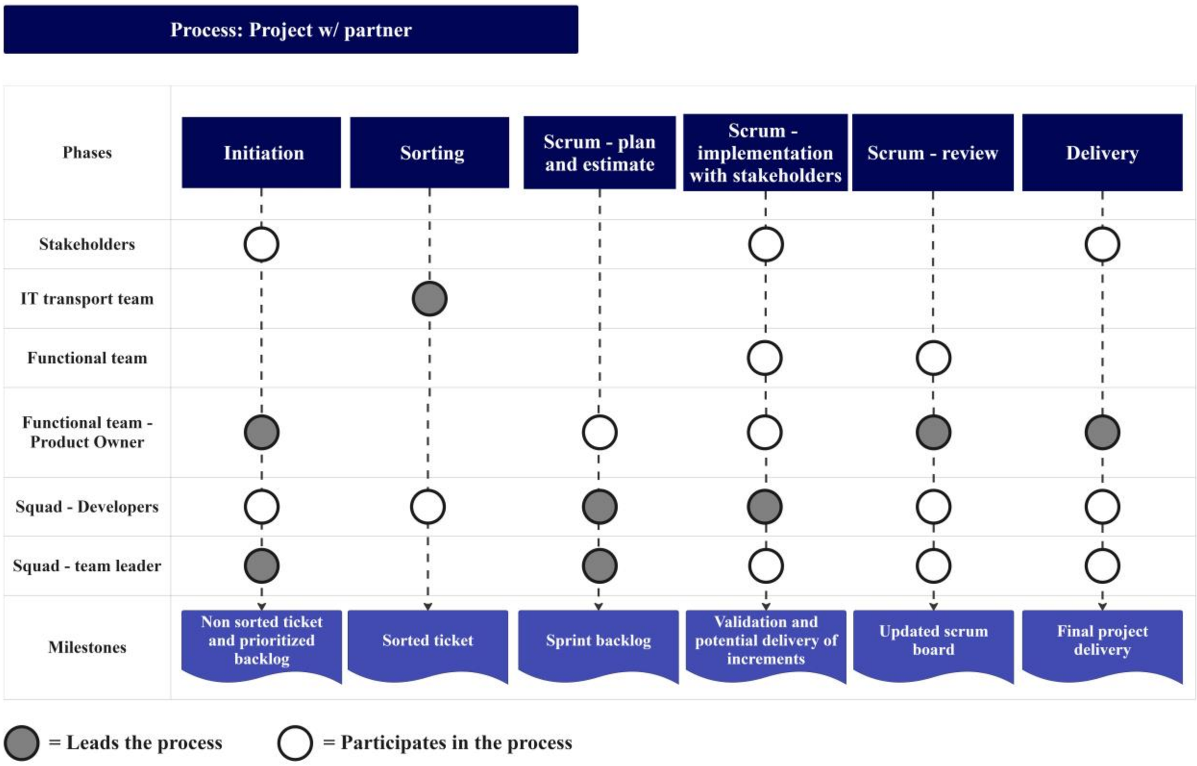

Figure 6 – AS IS: Responsibility matrix representing the core process "Project with Partner"

The squad's other core process involves the development of incremental improvements, i.e., the enhancement of the system's current features. Its phases are specified in the responsibility matrix defined in Figure 7.

Even though the development of corrective actions is not included in the squad's core activities, since it does not create value for RT, its modeling is similar to the previously mentioned process, since it implies the same flow of activities. The difference between both processes lies in the origin of the action, whether it was the need to fix an existing bug or an opportunity to improve the existing conditions. This distinction is explicit in the first phase, designated by "Identification and placement of an action".

The second phase corresponds to the “Sorting” of the Helpdesk request. The third phase includes the integration of the action in the product backlog, which also includes the squad’s projects. The fourth and sixth phases are the “Scrum – plan and estimate” and the “Scrum – review”. Finally, the fifth phase, “Scrum – implementation with operations” is similar to the project’s implementation phase but is adapted to better describe the specificities of working with in-house operations, rather than an external partner.

Similarly to the process “Project with partner”, the phases that include the scrum methodology are cyclic since an action can take more than one sprint to be implemented.

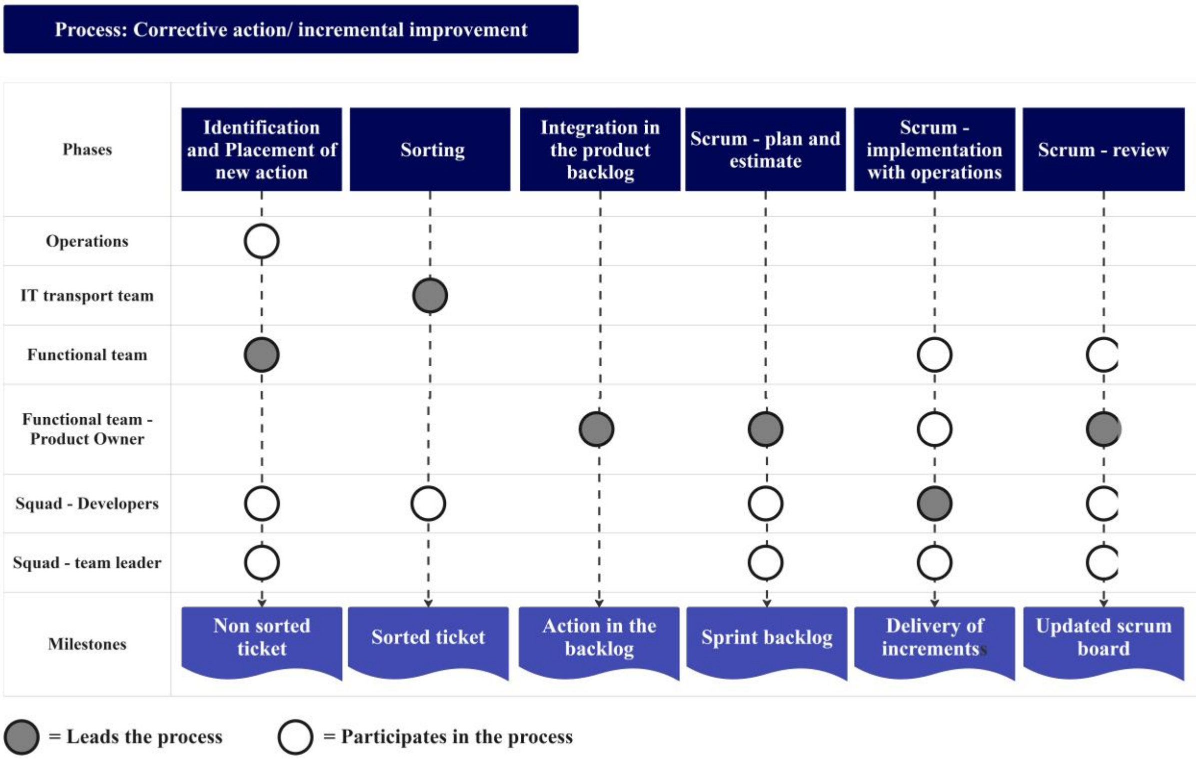

Figure 7 – AS IS: Responsibility matrix representing both the core process “Incremental improvement” and the support process “Corrective action”

It is worth mentioning that there are no performance indicators for the team's core processes and for the support process “Corrective actions”, i.e. there are no metrics in place to quantify compliance with project deadlines or the stakeholders’ service level.

Support processes

As mentioned above, the IT team, and more specifically the people dedicated to the road freight business, also provide support to the system's users. This support can include the correction of bugs in the systems codes, which is detailed in the process of “corrective action”, described in the responsibility matrix of Figure 7. It also includes the direct support provided to Rangel operators, in their daily activities, whose process is depicted in Figure 8.

It is important to mention that the team’s operations regarding the support process “Business support and incident handling” are currently managed with Kaizen. At the end of each week, there is a Kaizen event in which the team analyzes metrics related to their performance and productivity. The metrics that are currently analyzed are lagging indicators since they reflect the team’s historical performance. Some are calculated monthly, allowing a comparison with previous months. Others are calculated at the end of each week, to reflect the team’s weekly performance and to verify its compliance with the thresholds defined for IT teams.

In addition to the indicators that are calculated and analyzed specifically by the transport team, at the end of the week the members of the IT department receive a report with some metrics related to their entries and requests on the Helpdesk platform. These indicators are divided by team and are not analyzed during the weekly meetings, since their main purpose is to check that each employee has recorded their working hours and tasks.

However, the Kaizen methodology is not properly formalized, which can be observed, for example, in the absence of PDCA cycle (Plan, Do, Check, Act), i.e., although there is an analysis of metrics, the team does not establish a formalized plan to improve these metrics. Thus, since the metrics that are calculated weekly are not analyzed to implement a structured plan to improve them, they are not leading indicators.

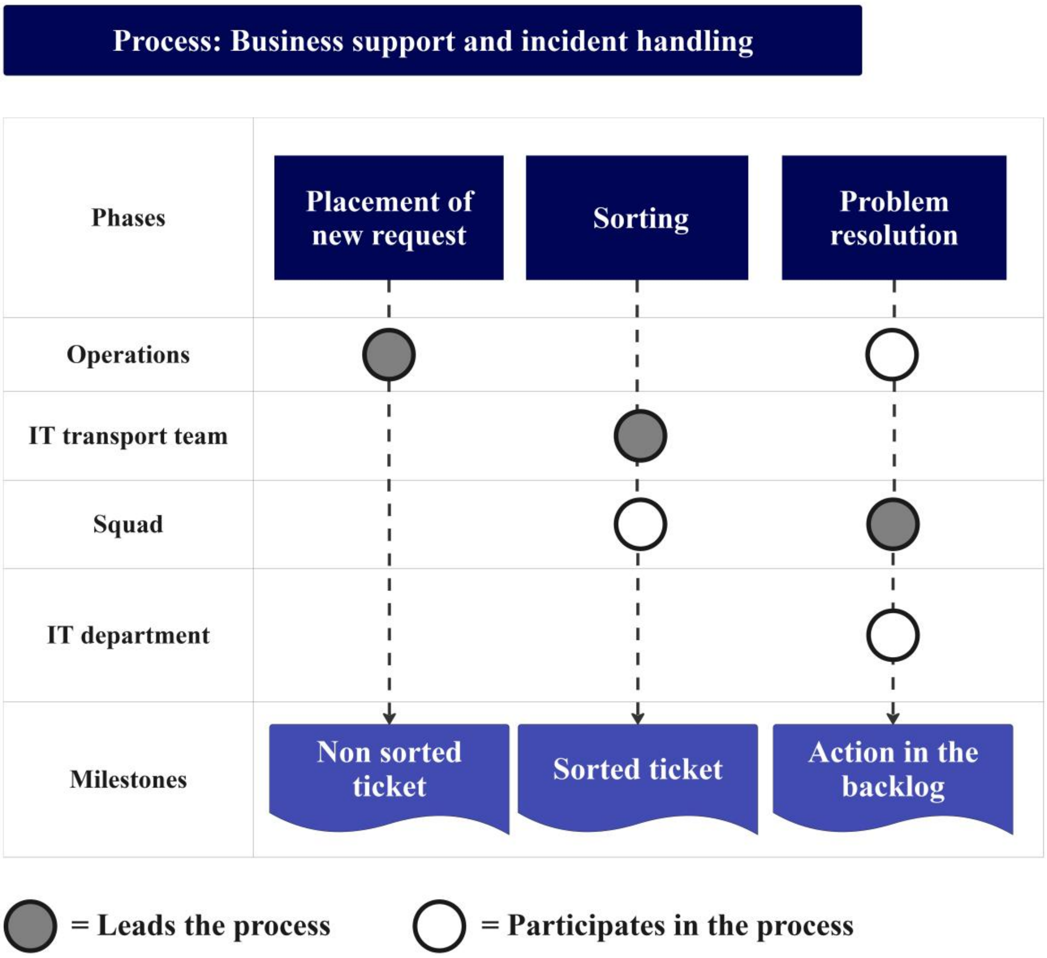

Figure 8 – AS IS: Responsibility matrix representing the support process “Business support and incident handling”

3.2.3 Second and third-level modeling

Core Process “Project with a partner”

Considering that the “initiation” phase is highly dependent on the project’s stakeholders and their inputs, its progression can be slightly different than the modeled flow of activities. Moreover, the phase initiates with the definition of requirements by the stakeholders, which are received and analyzed by the squad. In turn, the squad transmits these specifications, along with their input and recommendations, to the product owner of the functional team, which will adapt them to the business's functional needs.

Nonetheless, the product owner will produce a rough version of the product backlog for this project, which will then be analyzed by the squad’s team leader, who will create the project’s epics, as well as a plan for their release. Simultaneously, the squad developer will produce technical documentation, namely diagrams, to describe and orient the work that the squad needs to complete, namely in terms of the system's integration.

The squad should also produce a different type of document, which translates the technical diagrams into functional features and elements. This document is then analyzed by the functional team and by the higher management. Currently, due to the team's extensive workload, these types of documents are often neglected. Usually, they either aren't produced or remain incomplete.

The product owner will then validate this new version of the backlog and the team leader will describe the project in a Helpdesk ticket. In conclusion, this phase corresponds to the translation of the stakeholders' requisites into concrete requirements for the project and it occurs iteratively.

The third phase is represented in Figure 9. Since the squad collaborates with the functional team in both projects with external stakeholders and internal actions, the planning and estimation phase contemplates all these actions. Thus, there's a single scrum board that encompasses the product backlog and more specifically the sprint backlog, which includes all the scrum team's tasks, problems, and user stories. This scrum board was created and is currently maintained using JIRA software.

This phase contemplates the sprint planning meeting, which usually occurs in the weekly meeting. The participants of this meeting include the entirety of the scrum team, namely the squad and the functional team, which includes the product owner. Before this meeting, the product owner meets with higher management to determine which items to prioritize in the upcoming sprint. The sprints in which the work is developed usually have a duration of two weeks and thus, the weekly meeting does not necessarily coincide with the beginning of a sprint.

In this phase, if there are new epics introduced in the backlog, then the team will collaborate to split these epics into user stories and eventually into tasks, to simplify their completion. The team will then analyze the current prioritized backlog and the developers, i.e., the squad road, will provide their input as to what tasks can be completed in the sprint time frame.

However, there is an opportunity to improve the estimate that is done regarding the tasks that can be included in each sprint. The aim is to allocate tasks more realistically so that at the end of each sprint the work completed fulfills expectations, which constitutes one of the Scrum team's biggest problems.

Finally, the tasks for the sprint will be defined, which will be followed by the update of the Scrum Board. Nonetheless, at the end of the meeting, the product owner sends an email to higher management, which contains the tasks that are planned for that sprint.

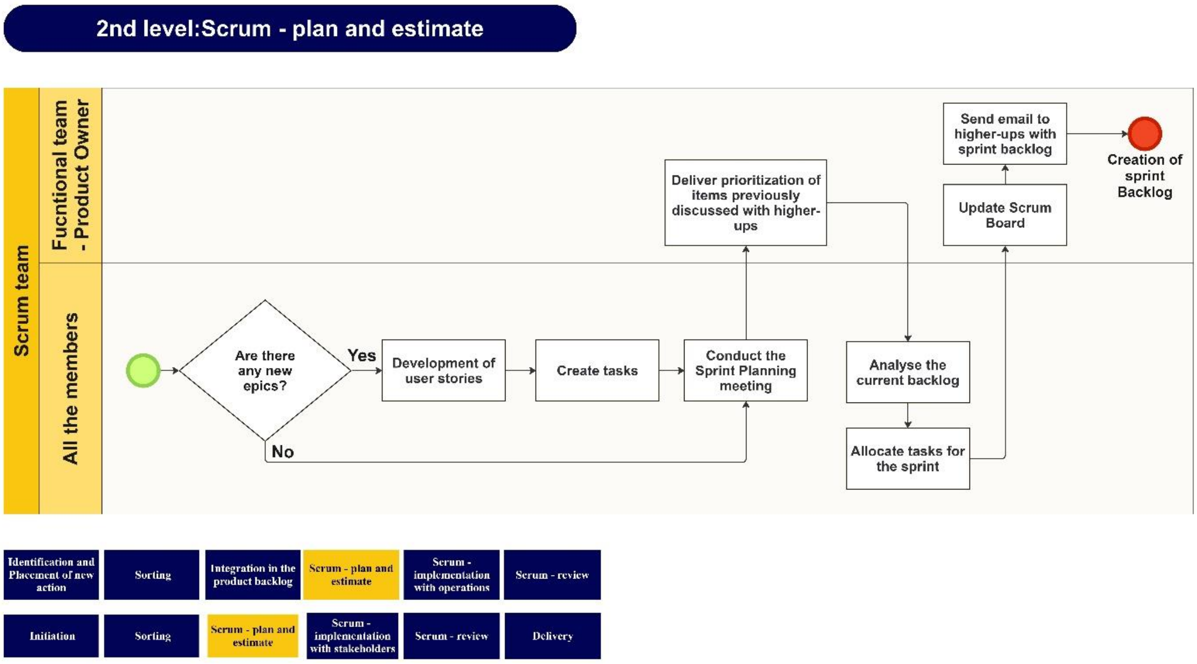

Figure 9 - AS IS: Swimlane diagram representing the phase “Scrum – plan and estimate”, common to the core process “Project with partner”, “Incremental improvement” and support process “Corrective action”

The fourth phase, modeled according to the swimlane diagram of Figure 10, corresponds to the implementation phase, and its main goal is the development of deliverables. This moment is one of the phases that can undergo several improvements, namely through the introduction of a person solely dedicated to the IT business partner’s functions, which will increase the efficiency of both the squad’s workflow and information flow.

According to the current flow of this phase, the creation of deliverables is followed by the performance of several tests in different environments. The first tests are performed by the squad to guarantee that the features created are ready to be subject to further tests in a functional environment. The functional team will then create several scenarios to evaluate if the deliverables fulfill the needed functional capabilities. If the tests are successful, the squad will test the deliverables in the stakeholders’ environment, who will analyze the results and provide feedback. In case the tests are successful, and according to the stakeholders’ requirements, i.e., if they intend to receive the deliverables progressively or not, the squad will deliver the increments.

As previously mentioned, due to the unique characteristics of each project, this phase can be adjusted to accommodate them. A project might need a higher volume of tests, the squad might collaborate in the functional tests and vice versa, and the tests might be performed in conjunction with the stakeholders, among others. Thus, the main goal of this phase is to provide a structured approach that can be adapted to each project.

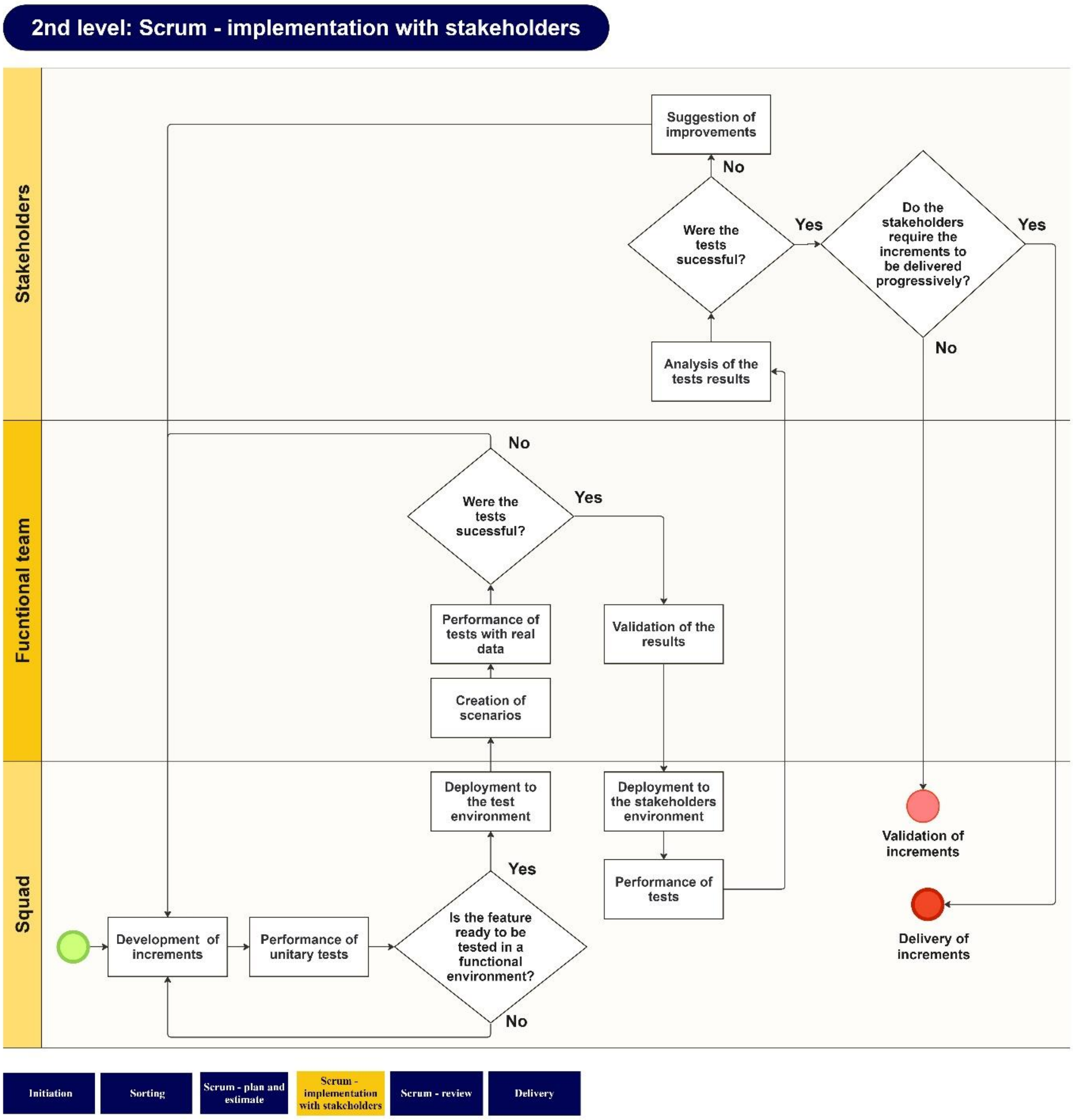

Figure 10 – AS IS: Swimlane diagram representing the phase “Scrum – implementation with stakeholders”, from the core process “Project with Partner”

Given the complexity and the importance of the tasks that the development of increments entails, it is relevant to understand its workflow, which is presented in Figure 11. This task is crucial for the success of the whole process.

Besides the creation of deliverables, this moment includes the performance of daily scrum ceremonies, in which the squad members exchange feedback about the work they have been completing and eventually assist each other with potential doubts. Currently, this ceremony occurs simultaneously with the daily scrum meetings for the squad Air & Sea since both squads are overseen by the same person. However, the members of the squad communicate among themselves throughout the day, whenever they need to, and thus these meetings are not necessarily structured and formal.

As previously mentioned, the code that the squad develops is usually modeled in technical diagrams, which allow a better understanding of the complex systems integrations. Thus, if the squad produces an increment that would be relevant to include in the technical documentation, it will be included or dully changed, if it was already featured in a diagram.

As mentioned above, the production of functional documentation is currently deeply neglected, which negatively impacts the squad, since its work is not properly reflected to both the functional team and higher-up management.

Currently, the team does not produce an impediment log, which constitutes an opportunity for improvement. During the daily scrum, the IT business partner can register obstacles that the squad might face while creating increments. The registration of these impediments will provide a better analysis and understanding of these problems, thus facilitating their resolution.

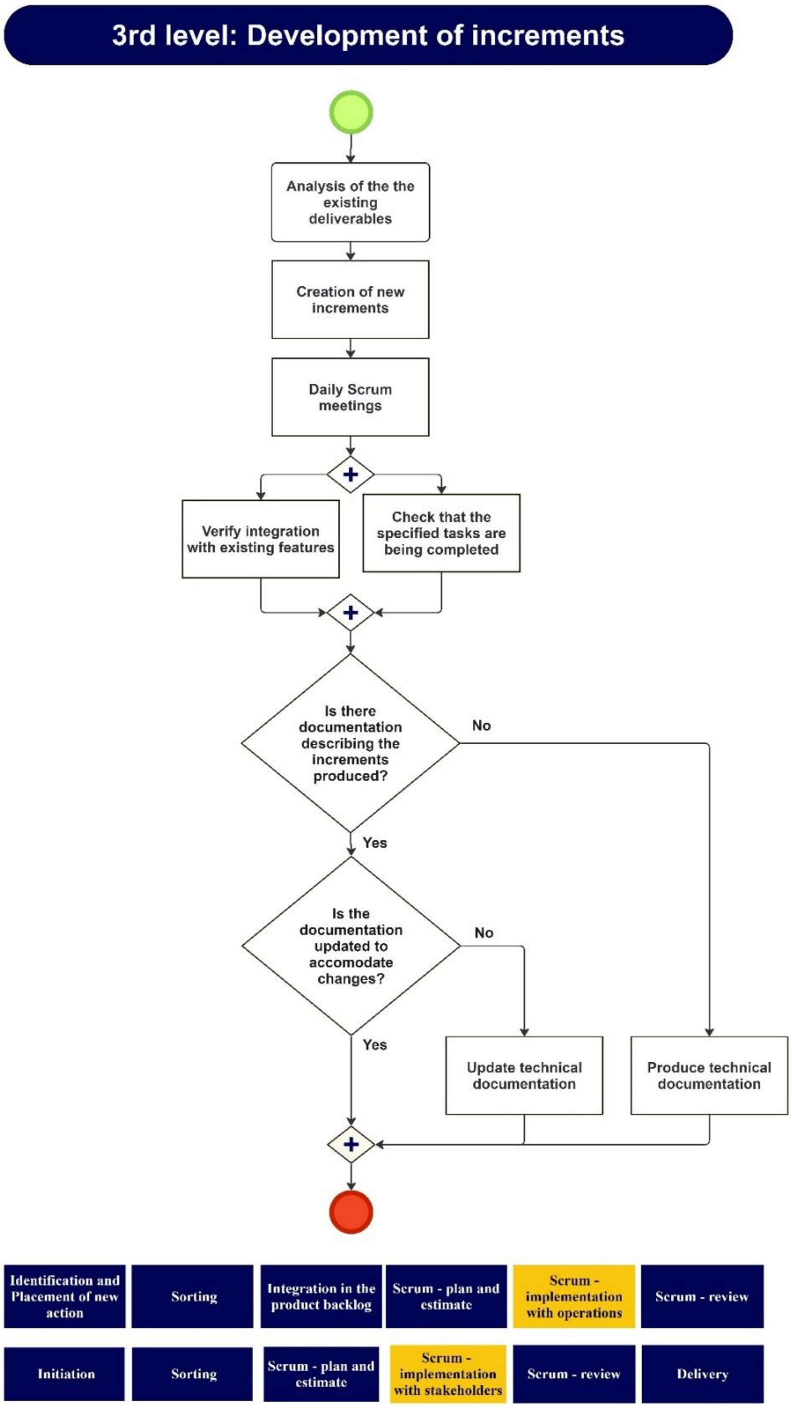

Figure 11 – AS IS: Flowchart diagram representing the task “Development of increments”, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

The fifth phase, also known as “Scrum - review” is displayed in Figure 12 and occurs during the squad’s weekly meetings with the functional team, in which the team reflects on the work that was performed during the sprint. If this meeting coincides with the end of the sprint, the scrum team will identify opportunities for improvement in the squad’s current operations, including the projects with the stakeholders. After this discussion, the Product Owner will validate the work that was performed during the sprint and update the Scrum Board. Otherwise, the team discusses the work that has been developed during the previous week, and the product owner will update the scrum board (which in this case coincides with the Sprint Backlog), according to the progress of the sprint’s tasks.

Similarly to the “Scrum – plan and estimate phase”, at the end of the sprint, the Product Owner compiles an email that summarizes the work conducted during that period.

Since this summary only presents the final status of the items in the sprint backlog, it does not accurately reflect the work performed by the squad. Most of the items are rather complex and are not usually finalized in one sprint, since their completion depends on multiple factors outside of the squad’s control. Thus, higher management is not always aware of these obstacles.

As mentioned above, there’s an opportunity to increase the efficiency of this phase’s workflow, through the analysis of the impediment log. This Scrum artifact will also serve as evidence of the squad’s work during the sprint, namely in terms of reflecting the impediments and obstacles that affect the creation of deliverables and thus, the completion of tasks.

Nonetheless, the squad’s team leader also needs to close the ticket (or tickets, if he chose to divide the project into several stages with unique milestones) that he previously opened to describe the project, so that the hours that the project took to complete are registered. Even though the tickets are not necessarily closed during the weekly meeting, this task will be included in this phase for simplification purposes, since it is completed in the final moments of the project lifecycle.

The final phase, corresponding to the delivery of the project to the stakeholders, simply involves its deployment to the partner’s environment. The delivery will be performed according to the partner’s specifications, i.e., if they had chosen to receive the deliverables progressively or in their entirety at the end of the project. There is also a meeting with the partners to finalize the project.

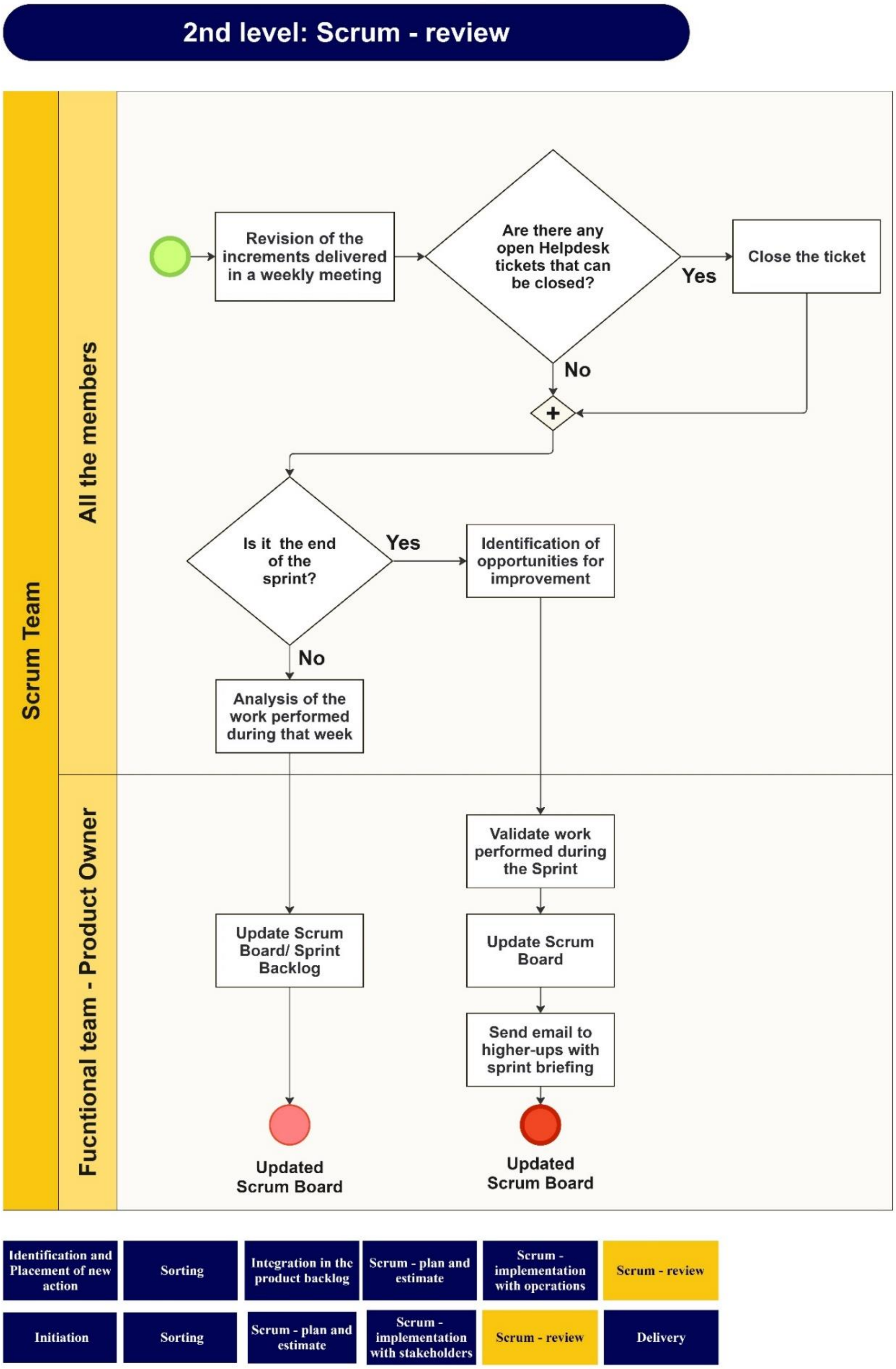

Figure 12 – AS IS: Swimlane diagram representing the phase “Scrum – review, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action

Core Process “Incremental improvement” and support process “Corrective Action”

Typically, this process begins when operations identify a feature in the systems that they use daily that might be prone to alterations. Thus, they establish contact with the functional team, which analyses their request.

This analysis corresponds to a complex task whose workflow has been detailed using third-level modeling. Since this task initializes the whole process, it is during this moment that the request for operations is either defined as a corrective action or as an incremental improvement, after the analysis of its current functioning.

If the request is based on a feature that causes recurring problems, then it will originate a corrective action, for which it is necessary to understand how the feature should be functioning. If the request is based on an opportunity to improve the existing conditions of the system and thus increase the operations efficiency and productivity, then the workflow will be more complex. The functional team will need to discuss with operations how could the feature be improved and will then analyze this information, as well as survey the requirements to implement these upgrades. In turn, the squad will provide a feasibility analysis on a technical level, as well as an estimate of the development duration.

Afterwards, the functional team will compile this information, which will be encapsulated in a helpdesk request. Thus, the second phase will entail the sorting of the request.

Currently, the squad also initializes corrective actions, whenever they detect that a feature or an element is not functioning properly. These problems can usually be identified whenever the team detects a pattern in the Helpdesk requests, i.e., whenever there are a lot of requests related to a specific feature that might not be functioning properly. Thus, to answer these requests and mostly to avoid them, the team performs a root cause analysis, followed by the resolution of the problem.

Nowadays, the resolution of these issues does not follow a formal process or workflow and is not included in the product backlog. Thus, besides not being registered, these changes require time from the team, which *a priori* would only be dedicated to backlog items.

The third phase entails the inclusion of the action in the product backlog, which is performed by the product owner. It also includes the analysis of the action's priority, in comparison to the other items in the backlog, followed by its prioritization. The action's priority level, to the detriment of the other items' priority, will determine if it is immediately addressed in the following sprint. Currently, some of these actions are not included in the backlog and are simply communicated to the team via a Helpdesk ticket. Since these actions are not included in the sprint workload, they will lead to inefficiencies and will jeopardize the completion of the items that are included in the sprint backlog. Besides not being properly prioritized, these actions take time and effort from the developers, which should be dedicated to completing the tasks that were specifically allocated to the sprint.

After the Scrum planning phase, in which both improvement actions and projects are addressed, there's the implementation phase, which is displayed in Figure 13. In this case, after the development of increments, they will be tested by the functional team, with possible participation from the squad, according to the action's requirements. Following the successful test results, the squad will release the changes or the new features to the productive environment. At this moment, they become visible and available for operations.

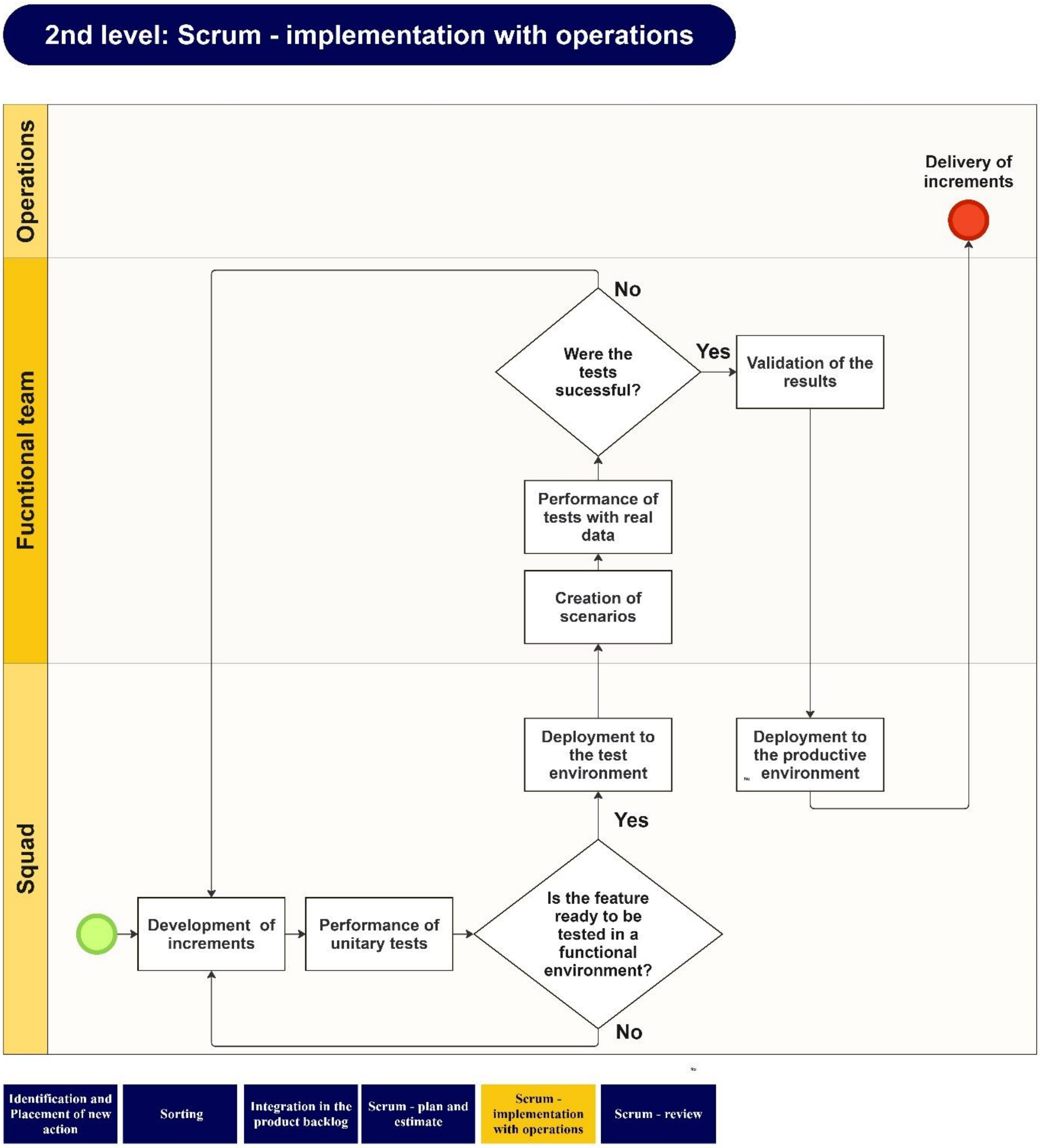

Figure 13 – AS IS: Swimlane diagram representing the phase “Scrum – implementation with operations”, common to the core process “Incremental improvement” and the support process “Corrective action”

The process ends with the Scrum review phase, which was already described above. However, if there are any open tickets regarding actions that were already concluded, then the product owner needs to close them.

Process Business support and incident handling

Through this process, the transport team provides support to operations, whenever they place a request on the Helpdesk platform. In the first phase, also known as “Placement of new request”, an operator places a new request, which includes a description of the problem and the associated application. Based on the selected application, the ticket will be automatically placed in the queue of the responsible IT team. Moreover, this ticket will be placed in the queue of unsorted requests.

The next phase, represented in Figure 14, can become very complex, which ultimately might delay the resolution of the ticket. Currently, this phase is performed by all members of the team, regardless of their business sector. Thus, during a previously accorded period (like a morning or an afternoon) one of the team's members is tasked with sorting the tickets that are placed in the unsorted queue, in parallel with their regular tasks.

First, the technician critically analyses the ticket's content. Based on this analysis, there might be a need to forward the ticket to another team or department. Firstly, if the ticket was indeed meant for the IT department, but for another team, the technician will forward the ticket to the respective team, and it will be eliminated from the transport team queue.

If the ticket was meant for another department other than the functional units that collaborate with the transport team, then the technician will contact the person who placed the request, so they can forward the ticket to the right department. In this case, the ticket will be rejected, and its root cause will be defined as "Out of IT's portfolio".

If the ticket was meant for the business functional team, the ticket's trajectory will depend on the person who is sorting it. If the technician does not work with the functional team for which the ticket is meant (for example, if the person performing the triage works within the squad Air & Sea, but the ticket is meant for the road freight functional team) but is able to identify that the ticket is not meant for the transport team, then the ticket will be immediately dispatched. Otherwise, the ticket will be processed normally by the transport team.

After this moment, the technician will perform the actual sorting. Firstly, they need to define the ticket's type, for example, if it is an information request, if it is a project, among others. This step is important because all the team's activities are registered in the Helpdesk platform, so that the technicians can register the amount of time spent on each task.

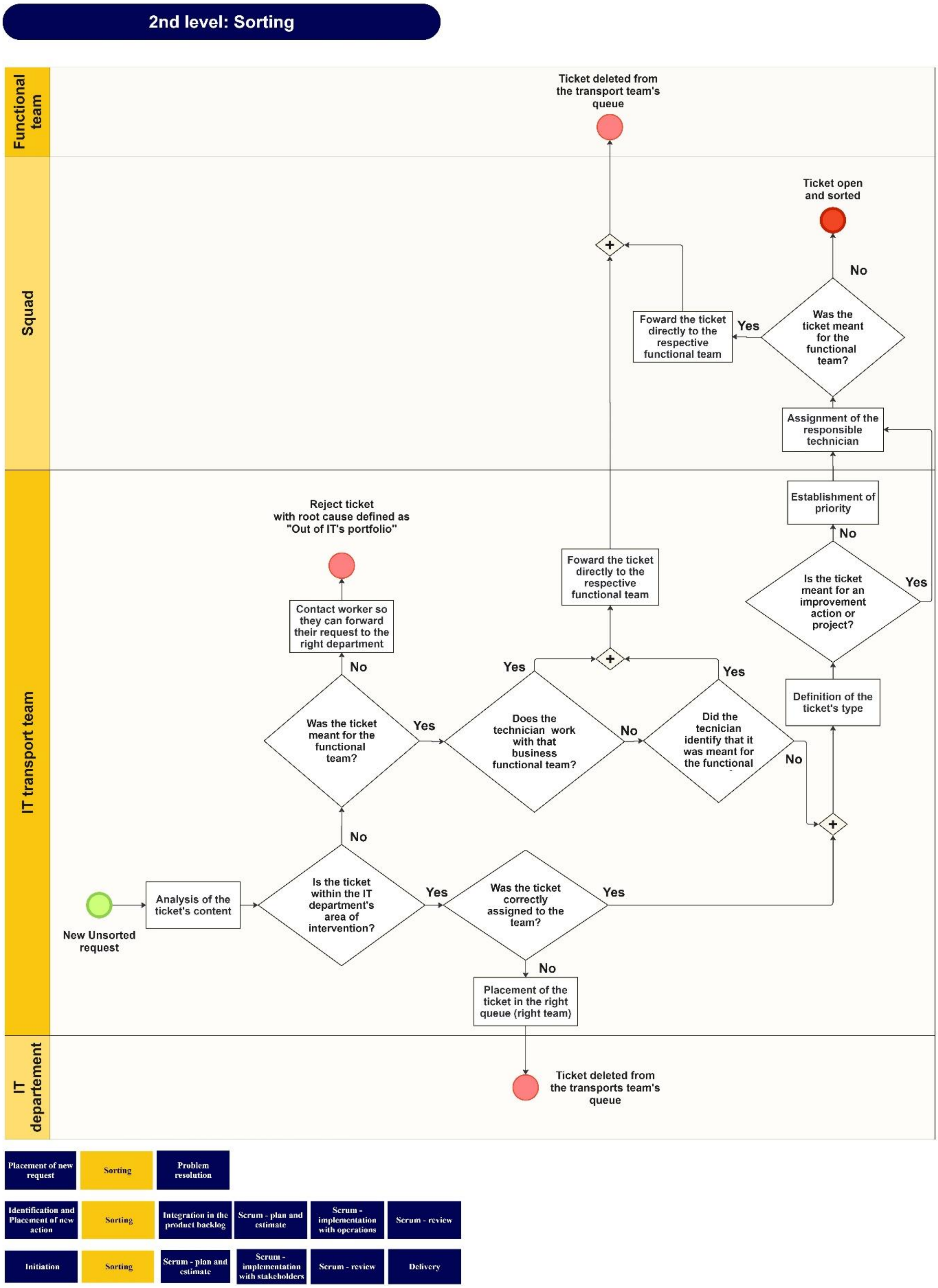

Figure 14 – AS IS: Swimlane diagram representing the phase "Sorting", common to all the processes

Even though all tickets are sorted, the tickets that are meant to register projects or improvement actions don't go through the task of assigning priority. At this moment, the technician attributes a level to the ticket, according to the level at which the problem affects the requester's operations. This scale is expressed in a matrix, which is presented in Figure 15, that evaluates the problem's impact on the operations and the existence of alternatives.

According to each priority level, a SLA is attributed to a ticket, which is the maximum amount of time in which the ticket should be resolved.

Thus, if the problem originated from a request that completely blocks the operations and there are no alternatives to perform them, then the ticket will have a priority level of 1 and a SLA of 4 hours. If there are alternatives, then the ticket will have level 2 priority and a SLA of 1 business day. On the other hand, if the problem has a medium impact, i.e., affects the operations but doesn't prevent their normal functioning, regardless of the existence of alternatives, then the priority will be a level 3 and the SLA will be 2 business days.

Lastly, the least serious type of ticket, which entails a problem that has no impact on operations, then the priority level will be 4 and the SLA will be 5 business days.

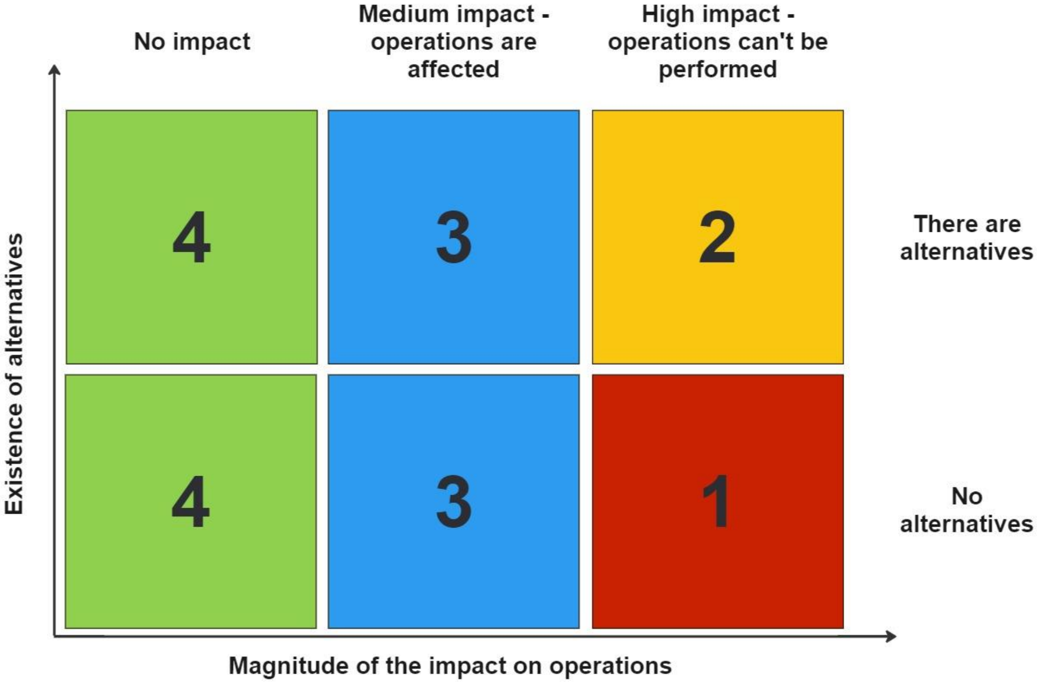

Figure 15 – Matrix representing the ticket´s priority level, according to its impact and the existence of alternatives

After its prioritization, the ticket will be assigned to the technician who will be responsible for solving the problem. In this moment, the technician might still find tickets that should be solved by the business functional team. In that case, they will forward the ticket. Otherwise, the ticket will be placed in their queue and its status will change to "open".

Finally, during the last phase, the ticket will be solved. If the technician needs further information to solve it, they will contact the requester. Until they receive a satisfactory reply, the ticket will be placed on hold, which means that the SLA will be paused, so it doesn't expire while the technician waits for a reply.

Nonetheless, if the resolution depends on another IT team, then the ticket will be placed in their queue. Finally, when the problem is solved, the technician will close the ticket, while specifying or completing the ticket's elements: type of request, root cause, user classification, sub-category, describe the problem resolution, and the amount of time spent for the resolution.

Moreover, to assess the quality of the support that's provided to the Helpdesk users, after they place 5 requests that are closed, they receive a survey to evaluate their experience. Finally, the people who manage the IT department will later analyze the users' feedback, to assess the team's performance.

4 Development and implementation of solutions – TO BE

This chapter describes the solutions that were developed to address the previously identified inefficiencies while materializing the respective opportunities for improvement. The chapter also presents indicators that were calculated throughout this project, to analyze the viability of the proposed solutions, as well as some also suggestions for metrics that could be calculated in the future.

The TO BE modeling can be found in Appendix B. Throughout this chapter, there will be a presentation of the diagrams that include the most meaningful changes to the team's processes and phases. Thus, Table 2 presents the TO BE models and their respective placements.

4.1 Core Processes and Support Process "Corrective Action"

Even though this subchapter is meant for both core processes "Project with partner" and "Incremental improvement", and thus for the support process "Corrective action", no suggestions for improvement will be made for the phases that are exclusive to the "Project with Partner" process. Throughout the course of this dissertation, there was no active participation in these phases, so the suggestions will only be included in Chapter 5 as possible future work.

4.1.1 TO-BE Process Modeling

The models that feature the larger amount of alterations are related to the processes "Incremental improvements" and "corrective actions". *A priori*, these processes are the most affected by the introduction of an IT business partner. Nevertheless, they were the processes that demonstrated the greatest feasibility in terms of addressing the problems and inefficiencies mentioned above.

Regarding the first phase "Identification and placement of new action", the most relevant change is the acknowledgment of actions that were identified by the squad, which were previously not visible to the functional team. This modification enables the integration of these tasks into the team workload and its prioritization.

Nonetheless, to accommodate this alteration, the third level model "Analysis of the request" was also subject to alterations. Thus, before the feasibility analysis is conducted by the functional team, the squad road provides inputs regarding the actions they propose.

Table 2 - Index containing the diagrams that represent TO BE modeling

Process	**Modeling**			
	First level	**Second level**		**Third level**
Project w/partner	Figure 3 in Appendix B	Initiation	Figure 16 in Appendix B	Development of increments: Figure 18 and Figure 13 in Appendix B Update the Scrum Board: Figure 11 in Appendix B
		Sorting	Figure 20 and Figure 5 in Appendix B	
		Scrum – plan and estimate	Figure 16 and Figure 10 in Appendix B	
		Implementation with stakeholders	Figure 10 and Figure 17 in Appendix B	
		Scrum - review	Figure 19 and Figure 15 in Appendix B	
		Delivery	Figure 18 in Appendix A	
Incremental improvement	Figure 2 in Appendix B	Identification and Placement of new action	Figure 7 in Appendix B	Analysis of request: Figure 8 in Appendix B Development of increments: Figure 18 and Figure 13 in Appendix B Update the Scrum Board: Figure 11 in Appendix B Update the Kanban Board: Figure 14 in Appendix B
		Sorting	Figure 20 and Figure 5 in Appendix B	
		Integration in the team's boards	Figure 9 in Appendix B	
Corrective action		Scrum – plan and estimate	Figure 10 in Appendix B	
		Implementation with operations	Figure 17 and Figure 12 in Appendix B	
		Scrum - review	Figure 19 and Figure 15 in Appendix B	
Business support and incident handling	Figure 1 in Appendix B	Placement of new request	Figure 4 in Appendix B	
		Sorting	Figure 20 and Figure 5 in Appendix B	
		Problem resolution	Figure 6 in Appendix B	

The second phase, which was previously labeled as “Integration in the Scrum board”, was rebranded as “Integration in the team's boards”, which enables the inclusion and prioritization of items that do not necessarily need to be included in the Scrum Backlog but can be included in a Kanban board. This change will enable better management of the team's core activities and visibility for the functional team of all the tasks the developers are working on.

Since there will be a new board to manage the team's work items, a new flowchart was created, titled "Update Kanban board", which is a third-level model. This diagram is a simplified version of the "Update Scrum Board" diagram, as the tasks and workflow are similar. However, the Kanban board is intended for less complex tasks that do not go through so many phases. Moreover, one of the Kanban principles is the definition of a "Work in progress limit", which will be established in this board, to guarantee that the developers are not working on an excessive number of tasks.

Although it may seem counterintuitive to divide the programmers' tasks into two different boards, this change is justified. In Jira, the software used to manage both boards, it is possible to create dashboards to visualize the tasks with custom filters. For example, the IT business partner can create a dashboard that includes all the tasks assigned to a given developer, regardless of where the task is placed. This new board ensures all work items related to incremental improvements and correction actions are included in Jira. These requests would previously only be placed in the Helpdesk platform, which should not happen since the purpose of this platform (except for the requests related to the support process "Business support and incident handling") is to simply record the time spent completing the request.

To demonstrate the impediments affecting the developers directly into the scrum board, namely the obstacles related to dependence on others, the scrum board was modified, and, consequently, the phase "Update scrum board" was also altered. Therefore, a new status has been included, which includes a new column "Dependence on others", in which the tasks outside the team's scope of control will be placed. The importance of this new status is related to the need to document the true status of the tasks, which in turn will enable a better understanding of the approach that should be taken to remove the obstacles.

As for the fourth phase "Scrum – plan and estimate", which is represented in Figure 16, some of the modifications are related to the inclusion of an IT Business partner, who will lead the analysis of the impediment log, which in turn helps the team in allocating the tasks for the sprint. Nonetheless, the IT Business Partner also takes over the task of sending upper management an email with the sprint plan, which includes the impediments still affecting the completion of tasks.

As mentioned earlier, the team does not adopt a rigid Scrum methodology, but rather a hybrid methodology, adapted to the team's activities. However, if the team adheres to a more rigorous Scrum methodology and starts breaking down work items into smaller, more manageable items, it is believed there will be a greater chance of completing the items.

In addition, to include a more realistic number of tasks in each sprint, this phase has been modified so that the team can discuss the time and effort required for each task. These steps are important because, although product owners understand the business requirements, they don't always understand the details of the implementation. So, a good estimate can give the product owner new insight into the level of effort required per work item. As a result, the team will be able to estimate the number of story points and the number of hours needed per task, which results in a more realistic attribution of tasks to the sprint.

Although story points are a more accurate and appropriate metric for quantifying the progression of tasks in software development, the team should also estimate the time required per task, as it is a more intuitive metric. Nonetheless, as the Scrum Team is still not used to this metric, including the time required per task facilitates the adaptation to this new procedure. Consequently, it provides more visibility for the functional team and upper management.

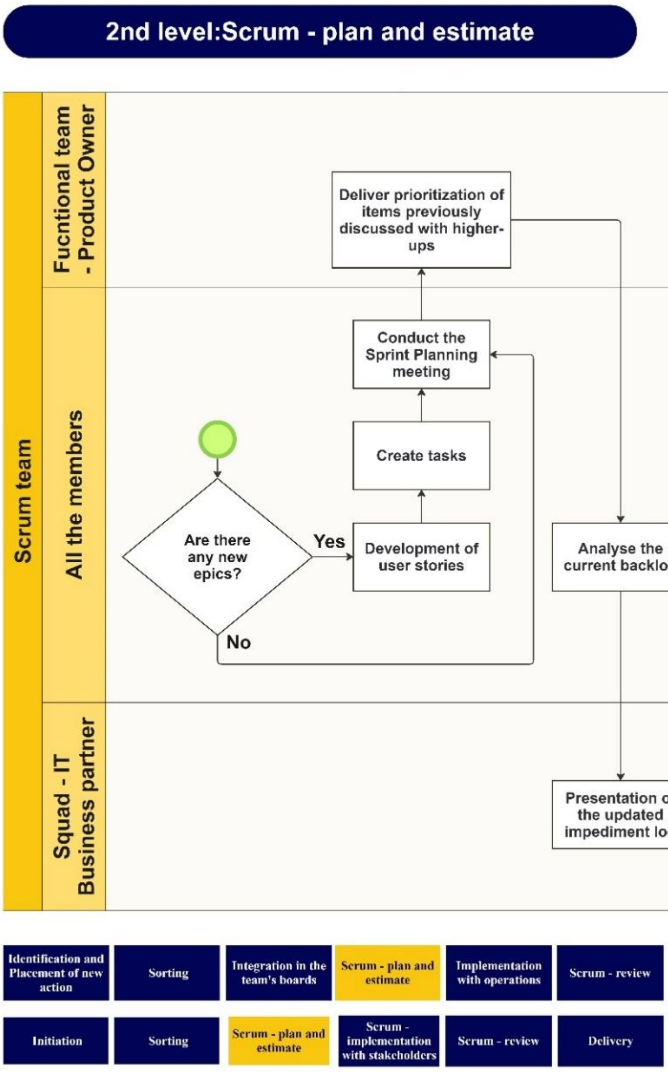

Figure 16 – TO BE: Swimlane diagram representing the phase “Scrum – plan and estimate”, common to the core process “Project with partner”, “Incremental improvement” and support process “Corrective action”

The fifth phase, previously known as “Scrum – Implementation with operations”, is now entitled “Implementation with operations”, and is displayed in Figure 17. This phase now includes the completion of all the team’s tasks, the items included in the scrum board, and the items on the Kanban board, which will be updated by the IT Business Partner. Another alteration is the possibility of updating the Scrum Board more, which can be done by any member of the Scrum team. Nonetheless, the IT Business Partner will test the new features and corrections after their deployment by the developers. These tasks will be completed in cooperation with the functional team, which will validate the results and intervene if necessary.

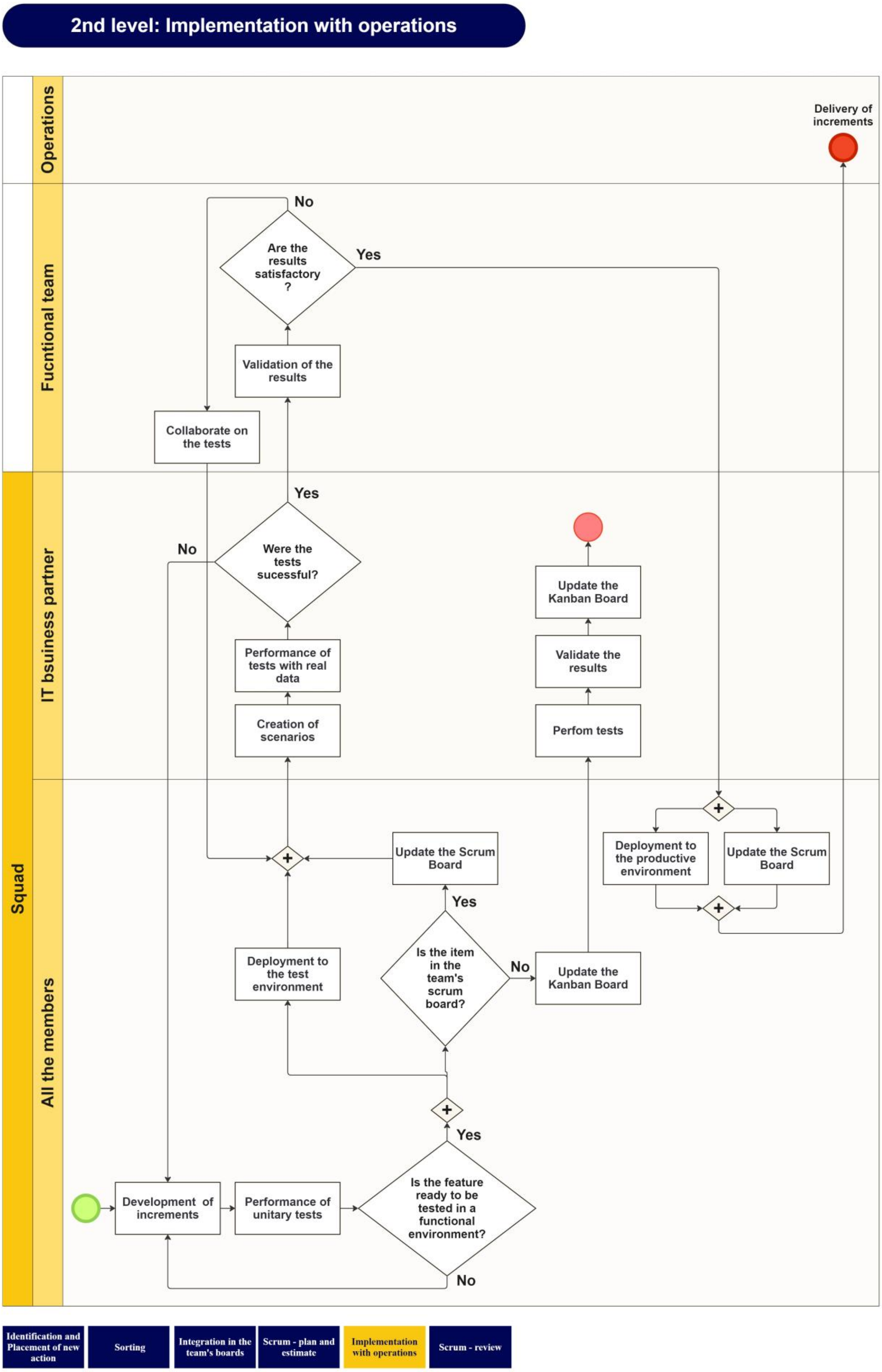

Figure 17 – TO BE: Swimlane diagram representing the phase “Scrum – implementation with operations”, common to the core process “Incremental improvement” and the support process “Corrective action”

In the context of the proposed alterations to this phase, there will also be modifications in the diagram “Development of increments”, which constitutes a third-level model, and is represented in Figure 18.

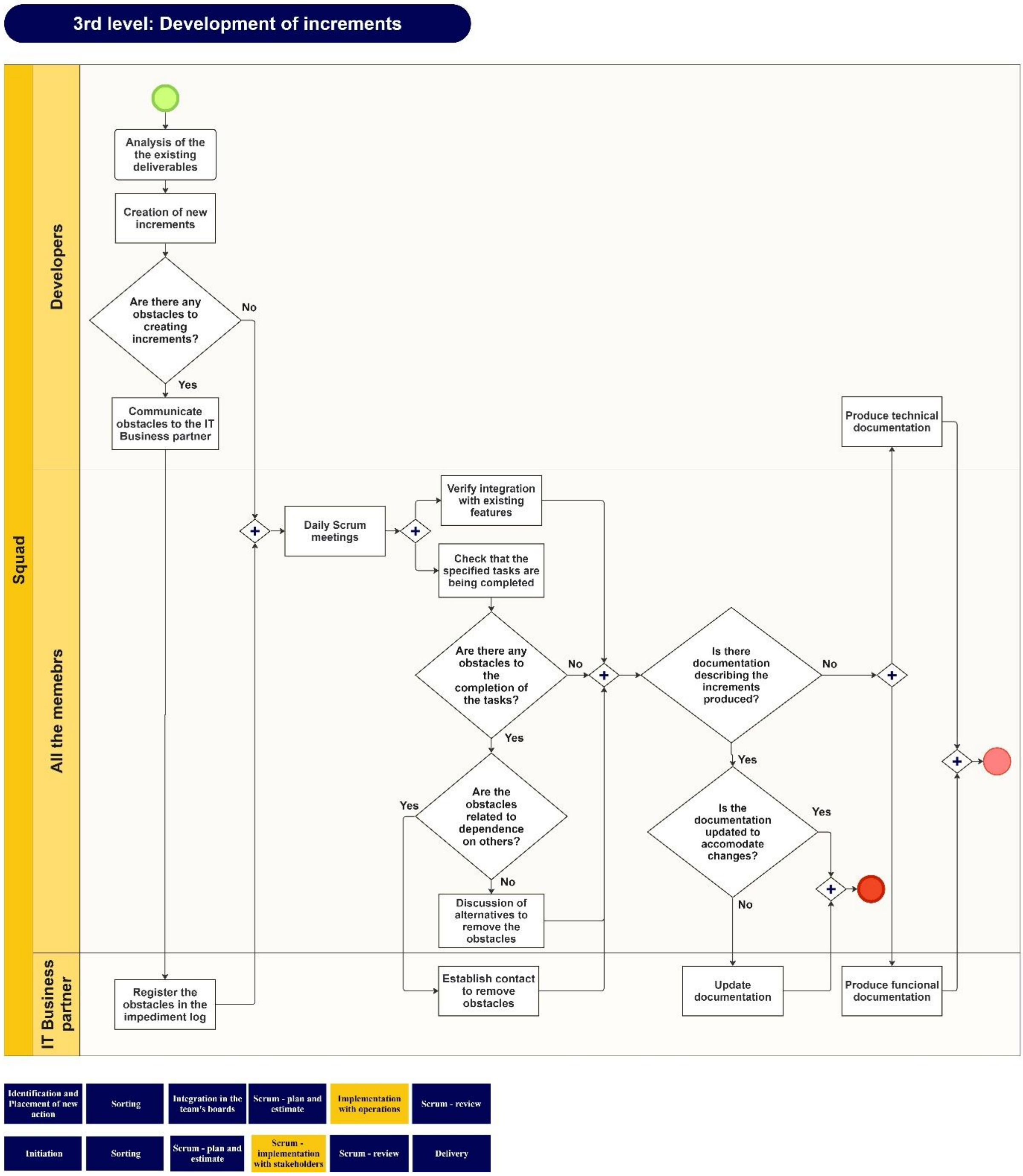

Figure 18 – TO BE: Swimlane diagram representing the task “Development of increments”, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

Once again, the main change in this model is the introduction of an IT Business Partner, who will support the developers in the creation of increments. Namely, the IT Business Partner will register obstacles to the completion of tasks and will adopt a crucial part in the Daily Scrum meetings. They will collaborate in verifying the progress of the tasks and ensuring their integration with the existing features. Also, they will analyze the current obstacles and act accordingly, to ensure that the issues affecting the Scrum Team are properly discussed and resolved. For impediments outside of the team’s scope of control, the IT business partner will establish contact with third parties. For other obstacles, the whole squad will try to understand the root cause of the problem and discuss alternatives or ways to overcome it.

Nonetheless, the IT business partner will intervene in documenting the work performed by the developers, to ensure this task is not neglected. On the one hand, this information must be available to both the functional team and higher-ups, so they can visualize and acknowledge the squad's work, namely the new features or changes that are being implemented in the systems. On the other hand, these documents are crucial for the squad itself, for future reference.

The last phase of this process, "Scrum – review" is displayed in Figure 19. The primary goal of the proposed changes is to enable a better understanding of the work performed during the sprint, which in turn will lead to better analysis of opportunities for improvement. Thus, during the Scrum team's weekly meeting, the IT business partner will present the updated impediment log, as well as other tasks outside of the Scrum board that the squad is currently working on.

If the meeting is held at the end of the sprint, then the team shall discuss alternatives to attempt to remove any obstacles that currently affect the team, followed by the validation of the work performed by the Product owner.

If the meeting does not coincide with the sprint review meeting and any urgent tasks should be completed, these should be included in the Sprint Backlog, and thus in the Scrum board. *A priori*, the team should not add work items during the duration of the sprint. However, since this inclusion is only intended for urgent items, it constitutes a better alternative to simply placing a request in the helpdesk platform. This will ensure that the task is properly prioritized, avoiding problems of incompatibility between the squad and the functional team.

At the end of the meeting, the IT business partner will send an email regarding the final status of the work items, while mentioning the obstacles that had an impact on the team's work, as well as other improvements or corrective actions that were performed.

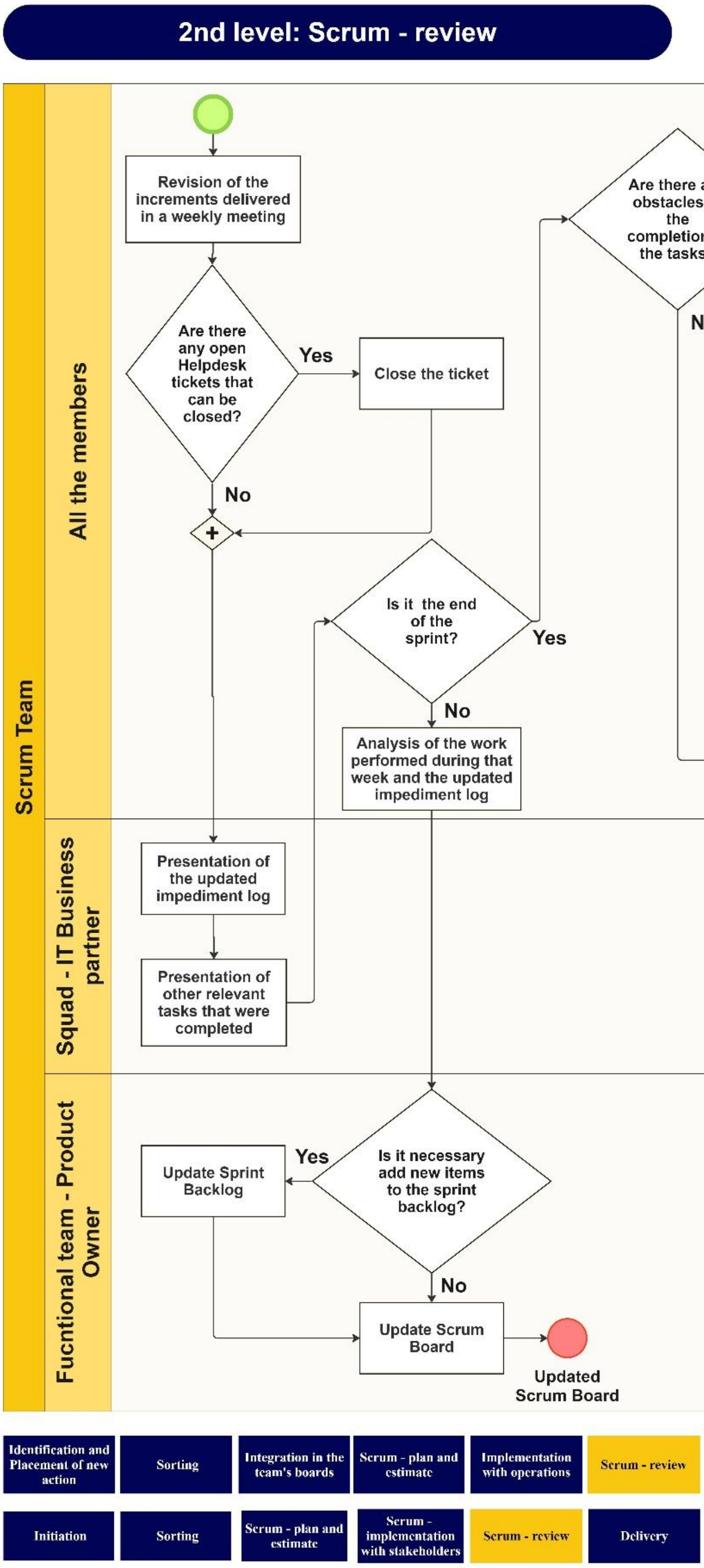

Figure 19 – TO BE: Swimlane diagram representing the phase “Scrum – review, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

In conclusion, the main purpose of the proposed alterations is to diminish the developers’ workload, by allowing them to focus a larger amount of time on the completion of increments. This is achieved by reinforcing the competences and responsibilities of the IT Business Partner, which will adopt an active role in terms of communication with the various stakeholders, such as the product owner, operations, other IT technicians, and partners, among others. Although the IT business partner takes on most of the responsibility for meetings with the functional team, the developers should still be present. However, if they cannot attend the meeting, there will still be someone to safeguard their work and showcase their concerns.

4.1.2 Performance indicators

As mentioned above, the only indicators that the squad used to measure its performance were related to the support process “Business support and incident handling”. Thus, within the scope of this dissertation, indicators related to these activities were calculated during the period from 03/04/2024 to 31/05/2024.

To enhance the analysis, the impediments that impacted the team in its activities were also incorporated. Based on the most frequent obstacles, four scenarios were created, not including scenario 0, which is the default, based on the recorded data. The scenarios aim to analyze the team's results if the different types of impediments are eliminated, as well as other factors. The description of each scenario is shown in Table 3.

Table 3 – Scenarios created based on the squad’s impediments and descriptions

SCENARIO	DESCRIPTION
0	Baseline scenario (without the removal of obstacles)
1	Scenario excluding obstacles labeled as “Lack of coordination within the Scrum team”
2	Scenario excluding obstacles labeled as “Dependent on others”
3	Scenario excluding obstacles labeled as “Dependent on others”, while simultaneously creating a new column in the Scrum Board for impediments labeled as “Dependent on others”
4	Scenario excluding obstacles labeled as “Lack of coordination within the Scrum team”, while simultaneously creating a new column in the Scrum Board for impediments labeled as “Dependent on others”
5	Scenario excluding obstacles labeled as “Dependent on others” and “Lack of coordination within the Scrum team”, while simultaneously creating a new column in the Scrum Board for impediments labeled as “Dependent on others”

Thus, the purpose of scenario 1 is to analyze the impact of the impediments based on the lack of coordination between the functional team and the squad. These obstacles occur when the objectives of the business team and the developers are not carefully outlined, for example, when the programmers produce increments that are not included in the scrum board, which subtracts time from the prioritized items.

On the other hand, scenario 2 attempts to analyze the impact of obstacles related to dependency on third parties, such as other IT teams, partners, and other departments, among others.

Scenario 3 is similar to scenario 2 in terms of excluding the same type of impediments, but it presumes the creation of a separate column on the scrum board for external dependencies, in which tasks that are blocked due to factors outside the team's scope of control will be placed. Thus, this scenario attempts to materialize an alteration that was previously suggested for the third level model “Update the Scrum Board”. Thus, the obstacles placed in this column will be excluded from the calculation of metrics at the end of the sprint. Scenario 4 is similar to scenario 3 but presumes the elimination of the same type of obstacles as scenario 1. Lastly, scenario 5 is the combination of scenarios 3 and 4.

The lagging indicators that were calculated for each sprint are displayed in Table 4.

Table 4 – Lagging Metrics calculated to quantify the work performed in terms of core activities and their formulas

METRICS	FORMULA	AFFECTED BY
COMPLETION RATE	Number of items completed / Total number of items	Creation of the new column
EXECUTION RATE	Sum of the measurement of the progress of each item / Total number of items	Creation of the new column
ITEMS FROM LAST SPRINT	Number of items that were also in the last sprint / Total number of items	Creation of the new column
CONSTRAINTS	Average of the constraints associated to the items in the sprint Constraints per task = Number of days impacted by constraints/ Total number of days per sprint	Removal of types of obstacles

It is important to note that these metrics will present the same values for different scenarios, as described in the column "Affected by" in Table 4. The purpose of this analysis is to find the best scenario, considering all indicators.

After calculating these metrics, which are showcased in Table 5 and in Table 6, it is possible to withdraw some conclusions. First, solely calculating the completion rate does not accurately reflect the work that was performed during the sprint, since it does not account for the progress that was achieved in some of the items, that might not have been completed due to obstacles exterior to the squad. Secondly, the exclusion of items that cannot be completed by the team due to third-party dependencies leads to an increase in the completion rate and the execution rate, which demonstrates that a significant proportion of the team's incomplete items are in fact beyond their control. Furthermore, excluding these items leads to a significant reduction in the percentage of items in a sprint that had already been introduced in a previous sprint, which enables the introduction of new items.

Finally, the 'Constraints' metric reveals the percentage of time affected by impediments, which amounted to significant figures in the first 3 sprints. Thus, by excluding some of the impediments, it is possible to conclude that they have a high impact on the squad's workflow.

However, in some sprints, even after excluding obstacles, the percentage of restrictions affecting the team's work was still not 0 percent, as it is unrealistic to assume that the workflow will be entirely obstacle-free.

Table 5 – Metrics calculated for each scenario to quantify the work performed in terms of core activities

		COMPLETION RATE [%]		EXECUTION RATE [%]		ITEMS FROM LAST SPRINT [%]	
	Scenarios	**0,1,2**	**3,4,5**	**0,1,2**	**3,4,5**	**0,1,2**	**3,4,5**
SPRINT	**1**	33%	38%	42%	48%	44%	38%
	2	18%	25%	25%	34%	55%	50%
	3	25%	44%	26%	46%	75%	67%
	4	38%	50%	44%	57%	62%	50%

Table 6 – Metric “Constraints” calculated for each scenario to quantify the work performed in terms of core activities

CONSTRAINTS [%]					
	Scenarios	**0**	**1,4**	**2,3**	**5**
SPRINT	**1**	46%	19%	38%	8%
	2	52%	48%	40%	35%
	3	56%	37%	27%	0%
	4	24%	24%	0%	0%

To monitor the daily progress of tasks and any open impediments, leading indicators can also be incorporated. A "daily task progress" metric can be implemented, which quantifies the progress of tasks, providing a detailed overview of the squad's workload.

On the other hand, a metric “Blocked tasks” can also be implemented to track the number of blocked tasks in real-time, which could facilitate the rapid identification of problems that are impeding the team's progress and enable their resolution.

These indicators will reflect the work already undertaken by the IT business partner in terms of recording and updating the impediment log and can therefore be analyzed in daily scrum meetings so that the squad has an updated and constant perception of its core activities.

By analyzing these indicators frequently, it will be possible to take actions that will be reflected in the leading indicators calculated at the end of each sprint.

Nonetheless, it is relevant to note that the transport team has a pre-established ceiling for the time dedicated to support activities, which should be a staggering maximum of 20 percent. Therefore, at least 80 percent of their time must be dedicated to their core activities. However, as shown in Table 7, this limit was not respected in sprint 3, and in sprint 2, the time dedicated to core activities was less than 90%. Thus, it is important to understand the progression of the support activities, to minimize their time.

Table 7 – Percentage of time dedicated to core activities, per sprint

		TIME DEDICATED TO CORE ACTIVITIES [%]
SPRINT	**1**	**92%**
	2	**87%**
	3	**65%**
	4	**92%**

4.2 Support process “Business support and incident handling”

4.2.1 TO-BE Process Modeling

Regarding the support process “Business support and incident handling”, the only phase subject to the prescription of improvements within the scope of this project is the “Sorting” phase. Nonetheless, this phase is common to all the squad’s processes and is therefore the phase whose improvement will have the greatest impact on the team's activities. The TO BE diagram for this phase is presented in Figure 20.

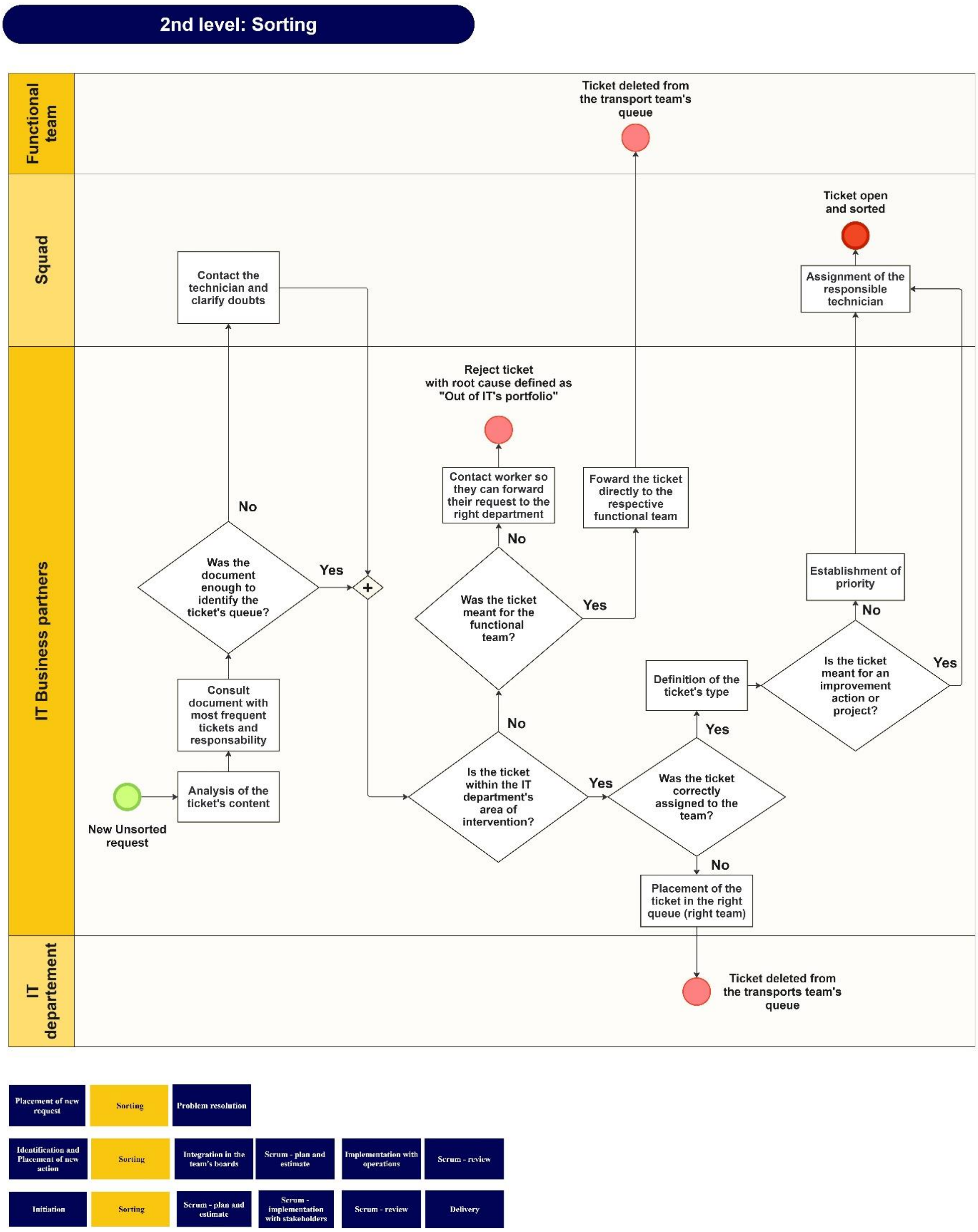

Figure 20 – TO BE: Swimlane diagram representing the phase "Sorting", common to all the processes

After analyzing the indicators relating to the time allocated for this phase, the main conclusion was that the technician who presented the best values, i.e., the person who was able to sort the requests faster, was the IT Business partner of Squad Air & Sea. Therefore, one of the suggested changes includes assigning triage exclusively to both IT business partners, not only to reduce the programmers' workload but also to reduce the average time from the moment the request is placed in the transport team queue until it is sorted.

As mentioned earlier, one of the main inefficiencies in this phase occurs when a technician does not triage the request properly (for example, when a request meant for the functional team is assigned to a technician in the transport team). The proposed solution therefore includes compiling a document with the most frequent items, which are often repetitive and can later be consulted during triage. If the document does not contain the necessary information, then the IT business partner can contact the respective technicians to clarify their doubts and to place the request in the right queue. Nevertheless, by concentrating this phase on two people, namely one from each squad, they will be able to specialize in this activity and to easily sort each ticket.

4.2.2 Performance indicators

As mentioned in Chapter 3, to assess the quality of the support the team provides to the system's users, at the end of each week the team analyzes metrics related to this process.

One of these metrics is the calculation of the percentage of time devoted to support and core activities. However, in this dissertation, these values will only be presented for the period corresponding to sprints, since the objective is to analyze factors that may hinder the core activities.

To formalize the Kaizen event in which this analysis takes place, namely, to implement the PDCA cycle, a strategy was devised to analyze the requests and the time spent resolving them in a more structured manner, which would enable the drawing of conclusions and the definition of actions. (Agarwal et al., 2017)

The strategy consisted of organizing the requests into groups according to their content (title, description, and the related application). At the moment, this development was only conducted for the road freight squad, as it was a response to the possibility of reducing the time dedicated to this process, thereby relieving the programmers' time for core activities. In addition, it implies an understanding of the specific activities of each squad. (Gates et al., 2021)

Firstly, with the data extracted from the Helpdesk platform, 3 data sets were created (Guan et al., 2022):

- A training dataset, which contained data from the first 6 months of 2023, to train the algorithm that would be implemented;
- A validation dataset, with data from the last 6 months of 2023, to evaluate the performance of the algorithm, which enabled the identification of potential issued and the adjustment of parameters to optimize its performance;
- A dataset designated test data, i.e., data from 2024, to analyze the algorithm final performance with untested data, to objectively evaluate its capacity to generalize and identify significant clusters in current circumstances.

Afterwards, Phyton script was created to process and analyze the data, as the data fed into the algorithm must be structured (Ayyadevara, 2018). Thus, the following techniques were applied to the unstructured text extracted from the Helpdesk platform (Milward, 2020).:

- Tokenization: to break the text into individual words or tokens;
- Stopword removal: to removing common words such as "the", "and", that should not be included in the analysis;
- Lemmatization, which reduces words to their base form. It is important to notice that Stemming was also considered, which removes suffixes and prefixes, without guaranteeing that the resulting "stem" will be a valid dictionary word. Thus, lemmatization was the technique that ensured that the text is processed with greater precision and accuracy, producing better results.

Nonetheless, the text was also processed considering specific features regarding the context of the text. For example, to consider abbreviations that are commonly used by the requesters, which should be processed in the same manner as the original word.

Finally, using clustering, an unsupervised machine learning technique, it was possible to categorize the requests, by grouping similar text samples together (Kapil et al., 2016). Based on the training and validation datasets, it was possible to improve the accuracy and coherence of the clusters, by using pre-identified keywords (Chen, 2022). Thus, it was possible to incorporate specific domain expertise and understanding into the clustering process, rather than relying solely on the algorithm to discover the clusters.

To determine the optimal number of clusters, the elbow method was used. It involves plotting the variation within each cluster against the number of clusters and identifying the "elbow" point where the curve flattens out, for which adding more clusters does not significantly improve the fit. Afterwards, the quality of the clustering was quantified using metrics such as inertia, silhouette score, Davies-Bouldin Index and Calinski-Harabasz Index (Zuccarelli, 2021).

After the clustering, it became possible to calculate some metrics, such as the number of requests per cluster, the total and average time spent resolving the requests in that cluster, as well as the percentage corresponding to the resolution time, considering the total time dedicated to the provision of support.

On the one hand, these metrics are calculated weekly and then analyzed at the Kaizen event to identify frequent problems and adjust strategies. Thus, these indicators will be leading, as their purpose is to predict or influence future results and enable corrective action to be taken promptly. An example of these metrics is showcased in Table 8.

Table 8 – Leading indicators calculated for the support process “Business support and incident handling” for the last week of May 2024 (27/05-31/05)

CLUSTER	TOTAL TIME SPENT PER CLUSTER (HOURS)	AVERAGE TIME SPENT PER CLUSTER (HOURS)	TIME SPENT PER CLUSTER CONSIDERING TOTAL SUPPORT TIME [%]	NUMBER OF TICKETS PER CLUSTER
2	0.75	0.375	24.32%	2
4	0.5	0.25	16.22%	2
6	1	0.5	32.43%	2
7	0.833333	0.416667	27.03%	2

The aim is to present the keywords that serve as labels for each cluster, as well as the titles and numbers of the requests, which will not be displayed due to privacy concerns.

A comparative analysis of the last few weeks will also be presented, to check the evolution of these metrics throughout time. This evolution is shown in figure 19 and table 1 in Appendix B.

On the other hand, these metrics are also calculated every month, thus constituting lagging indicators, to ascertain the impact of the actions that were taken during these weeks and to measure historical performance. Table 2 of Appendix B presents these metrics in a similar manner to Table 8, while presenting metrics from February to May 2024.

Similarly to the leading indicators, a comparison to the previous months is provided in appendix B, through Figures 20 and 21 and Table 3.

5 Conclusions and Project's prospects

The project results show that process modeling was fundamental to identify opportunities for improvement and to define the objectives and actions that would be developed as part of the project. The responsibility matrices enabled clear identification and definition of the actors involved in each process and phase. Further the flowcharts and swimlane diagrams permitted a detailed visualization and interpretation of all the tasks associated with the various phases.

As for the TO-BE modeling that was designed, it was not possible to redefine the workflow associated with the specific phases of the "Project with Partner" process, since there was no direct contact with external stakeholders during the project.

It was possible to identify some areas for improvement and therefore some actions to address them, which could be refined and enhanced when integrating this process.

Some of the suggestions include, in the "Initiation" phase, the facilitation of contact with partners through the IT business partner. During meetings or discussions with partners, this member of the squad could convey the business requirements more completely and concisely, which are often not transmitted in the most effective manner, as the functional team does not take part in some of these meetings, resulting in the sole exchange of technical specifications.

Nevertheless, this point of contact can address one of the factors that have the greatest impact on the team's activities, which is the dependence on third parties to fulfill tasks, particularly partners.

Finally, it could be possible to implement metrics to measure stakeholder satisfaction, in terms of the quality of deliverables, and compliance with the defined timetable, among others.

As for the "Incremental improvement" and "Corrective action" processes, although the Kanban board was only implemented to cover gaps in the Scrum board and promote the formalization of all actions regarding these processes, in the long term it could completely replace the Scrum board. This transition does not solely depend on the IT team, as it is a decision that must be made by the functional team and respective superiors.

Thus, it would open the possibility of incorporating critical chain project management into the management of the team's activities, which would allow for greater control of uncertainties and impediments to the completion of tasks.

In addition, it would address the problem of items not being resolved in a single sprint, and also the occasional necessity to add items to the board mid-sprint.

In terms of indicators for the core process "Incremental improvement" and the support process "Corrective action", the main conclusion drawn from their analysis is that the squad's work is heavily impacted by factors outside its control. The scenario that presented the best performance metrics was scenario 5, which excludes obstacles labeled as "Dependent on others" and "Lack of coordination within the Scrum team", while simultaneously creating a new column in the Scrum Board for impediments labeled as "Dependent on others"

However, this scenario is not realistic, as although the IT business partner can work to streamline projects and developments in collaboration with other stakeholders, it is not entirely possible to eliminate these obstacles. Therefore, the best scenario, taking into consideration feasibility, would be scenario 4, which excludes obstacles labeled as “Lack of coordination within the Scrum team”, while simultaneously creating a new column in the Scrum Board for impediments labeled as “Dependent on others”. The squad must then collaborate in proximity with the functional team to clearly and objectively define the work items for each sprint and their prioritization. This visibility and transparency are therefore fundamental to maximizing the production of increments, according to prioritization and business needs.

This transition is evident in the last sprint under analysis (sprint 4), in which the only obstacles impacting the team were related to third-party dependencies.

Finally, the leading indicators suggested in chapter 4 are to be implemented and analyzed in the daily scrum meetings.

Concerning the process of “Business support and incident handling”, there are several alternatives to simultaneously reduce the programmers' workload and improve the level of service provided. One option is to implement a search engine on the helpdesk platform, where a set of frequently asked questions and answers can be found, which will quickly clarify the user's doubts while avoiding the creation of a request. This development is quite ambitious and, although it is already being considered by other teams in the IT department, it is not likely to be deployed soon.

This work can also be done by the team's IT business partners, who can study historical data and document these frequently asked questions. Afterward, they can formulate answers or common procedures to address these issues. This work will thus optimize the “Problem Resolution” phase, while simultaneously reducing the technicians’ workload and providing a faster and standardized response to users.

Finally, it is possible to extend the analysis based on the request’s content, which was mentioned in Chapter 4 to the entire transport team. This will be possible by collaborating with the IT business partner of the Air & Sea squad, who will provide squad-specific knowledge, fundamental for text processing, and subsequent grouping.

References

A guide to the Scrum Body of knowledge (SBOK Guide). (2016th ed.). (2016). SCRUMstudy™, a brand of VMEdu, Inc.

Agarwal, S., Aggarwal, V., Akula, A. R., Dasgupta, G. B., & Sridhara, G. (2017). Automatic problem extraction and analysis from unstructured text in IT tickets. *IBM Journal of Research and Development*, *61*(1). https://doi.org/10.1147/JRD.2016.2629318

Agbejule, A., & Lehtineva, L. (2022). The relationship between traditional project management, agile project management and teamwork quality on project success. *International Journal of Organizational Analysis*, *30*(7), 124–136. https://doi.org/10.1108/IJOA-02-2022-3149

Alami, A., & Krancher, O. (2022). How Scrum adds value to achieving software quality? *Empirical Software Engineering*, *27*(7). https://doi.org/10.1007/s10664-022-10208-4

Al-Baik, O., & Miller, J. (2016). Kaizen cookbook: The success recipe for continuous learning and improvements. *Proceedings of the Annual Hawaii International Conference on System Sciences*, *2016-March*. https://doi.org/10.1109/HICSS.2016.666

AltexSoft. (2022, October 27). *Freight Forwarder: Key Workflows and Technology*. AltexSoft. https://www.altexsoft.com/blog/freight-forwarder/

Ayyadevara, V. K. (2018). Pro Machine Learning Algorithms: A Hands-On Approach to Implementing Algorithms in Python and R. In *Pro Machine Learning Algorithms: A Hands-On Approach to Implementing Algorithms in Python and R*. https://doi.org/10.1007/978-1-4842-3564-5

Azevedo Junior, D. P. de, & Campos, R. de. (2008). Definição de requisitos de software baseada numa arquitetura de modelagem de negócios. *Production*, *18*(1). https://doi.org/10.1590/s0103-65132008000100003

Baluch, I. (2005). The Changing Role of The Freight Forwarder. In *Transport Logistics: Past, Present and Predictions*. Winning Books.

Breyter, M. (2022). Agile Product and Project Management. In *Agile Product and Project Management*. https://doi.org/10.1007/978-1-4842-8200-7

Brezočnik, L., & Majer, Č. (2016). Comparison of agile methods: Scrum, Kanban, and Scrumban. *Proceedings of the 19th International Multiconference Information Society*, *C*(July).

Carlan, V., Huybrechts, T., Hellinckx, P., & Vanelslander, T. (2020). A universal middleware streaming framework and data analytics: Analysing their economic feasibility in road transport planning. *Research in Transportation Business and Management*, *34*. https://doi.org/10.1016/j.rtbm.2019.100424

Chapin, N., Hale, J. E., Khan, K. M., Ramil, J. F., & Tan, W. G. (2001). Types of software evolution and software maintenance. *Journal of Software Maintenance and Evolution*, *13*(1). https://doi.org/10.1002/smr.220

Chen, Z. L. (2022). Research and Application of Clustering Algorithm for Text Big Data. *Computational Intelligence and Neuroscience*, *2022*. https://doi.org/10.1155/2022/7042778

Dalton, J. (2019). Gemba Kaizen. In *Great Big Agile*. https://doi.org/10.1007/978-1-4842-4206-3_32

Dias Neto, J. C. (2023). *Enterprise Application Integration (EAI) on a Freight Forwarder*.

Faria, J. A. (2022a). *Business Process Models*.

Faria, J. A. (2022b). Intro to Process Thinking. In *Faculdade de Engenharia da Universidade do Porto*.

Faria, J. A. (2022c). Process Maps. In *Faculdade de Engenharia da Universidade do Porto*.

Gates, A., Gates, M., Sim, S., Elliott, S. A., Pillay, J., & Hartling, L. (2021). Creating efficiencies in the extraction of data from randomized trials: a prospective evaluation of a machine learning and text mining tool. *BMC Medical Research Methodology*, *21*(1). https://doi.org/10.1186/s12874-021-01354-2

González Moyano, C., Pufahl, L., Weber, I., & Mendling, J. (2022). Uses of business process modeling in agile software development projects. In *Information and Software Technology* (Vol. 152). https://doi.org/10.1016/j.infsof.2022.107028

Guan, R., Zhang, H., Liang, Y., Giunchiglia, F., Huang, L., & Feng, X. (2022). Deep Feature-Based Text Clustering and its Explanation. *IEEE Transactions on Knowledge and Data Engineering*, *34*(8). https://doi.org/10.1109/TKDE.2020.3028943

Harfoushi, O., & Obiedat, R. (2018). An Insight to Middleware Technologies used for Cloud Computing Frameworks: From the Perspective of Successful Enterprise Application Integration. *Modern Applied Science*, *12*(9). https://doi.org/10.5539/mas.v12n9p234

Imran, R., & Soomro, T. R. (2022). Mapping of Agile Processes into Project Management Knowledge Areas and Processes. *2022 International Conference on Business Analytics for Technology and Security, ICBATS 2022*. https://doi.org/10.1109/ICBATS54253.2022.9759013

Kapil, S., Chawla, M., & Ansari, M. D. (2016). On K-means data clustering algorithm with genetic algorithm. *2016 4th International Conference on Parallel, Distributed and Grid Computing, PDGC 2016*. https://doi.org/10.1109/PDGC.2016.7913145

Karkoszka, T., & Honorowicz, J. (2009). *Kaizen philosophy a manner of continuous improvement of processes and products Industrial management and organisation*.

Kirovska, N., & Koceski, S. (2015). Usage of Kanban methodology at software development teams. *Journal of Applied Economics and Business, 3*.

Kniberg, H. (2009). Kanban vs Scrum. *Crisp AB. Viitattu, 1*.

Mathenge, J. (2020, October 9). *Deploying vs Releasing Software: What's The Difference?* DevOps Blog. https://octopus.com/devops/continuous-delivery/deployments-vs-releases/

Milward, D. (2020). What Is Text Mining Text Analytics and Natural Language Processing? *Linguamatics*.

Munro, O. (2023). *What is Freight Forwarding? Definition, Benefits & Risks*. https://www.unleashedsoftware.com/blog/freight-forwarding

Petricioli, L., & Fertalj, K. (2022). Agile Software Development Methods and Hybridization Possibilities Beyond Scrumban. *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology, MIPRO 2022 - Proceedings*. https://doi.org/10.23919/MIPRO55190.2022.9803402

Pinheiro, A. J. F. (2004). *BP2IT : dos processos de negócio às tecnologias de informação* [Faculdade de Engenharia da Universidade do Porto]. http://hdl.handle.net/10216/11970

Pinton, M., & Silveira Torres Jr., A. (2023). THE AGILE TRANSITION PROCESS OF ORGANIZATIONS IN SOFTWARE DEVELOPMENT PROJECT MANAGEMENT. *REPAE - Revista de Ensino e Pesquisa Em Administração e Engenharia*, *9*(3). https://doi.org/10.51923/repae.v9i3.302

Rajabalinejad, M., van Dongen, L., & Ramtahalsing, M. (2020). Systems integration theory and fundamentals. *Safety and Reliability*, *39*(1), 83–113. https://doi.org/10.1080/09617353.2020.1712918

Salleh, N. M., & Nohuddin, P. N. E. (2019). Comparative study between lean six sigma and lean-agile for quality software requirement. *International Journal of Advanced Computer Science and Applications*, *10*(12). https://doi.org/10.14569/ijacsa.2019.0101230

Sassa, A. C., de Almeida, I. A., Pereira, T. N. F., & de Oliveira, M. S. (2023). Scrum: A Systematic Literature Review. *International Journal of Advanced Computer Science and Applications*, *14*(4). https://doi.org/10.14569/IJACSA.2023.0140420

Schwaber, K., & Sutherland, J. (2020). *Objetivo do Guia do Scrum*.

Shamshiri, A., Ryu, K. R., & Park, J. Y. (2024). Text mining and natural language processing in construction. In *Automation in Construction* (Vol. 158). https://doi.org/10.1016/j.autcon.2023.105200

Watt, A., BC Open Textbook Project, & Open Textbook Library. (n.d.). *Project management*.

Watts, S. (2019). Leading vs Lagging Indicators: What's The Difference? *BMC Blogs*.

Yongguo, J., Qiang, L., Changshuai, Q., Jian, S., & Qianqian, L. (2019). Message-oriented Middleware: A Review. *Proceedings - 5th International Conference on Big Data Computing and Communications, BIGCOM 2019*, 88–97. https://doi.org/10.1109/BIGCOM.2019.00023

Zuccarelli, E. (2021, January 31). *Performance Metrics in Machine Learning — Part 3: Clustering*. Towards Data Science. https://towardsdatascience.com/performance-metrics-in-machine-learning-part-3-clustering-d69550662dc6

Appendix A: AS IS modeling

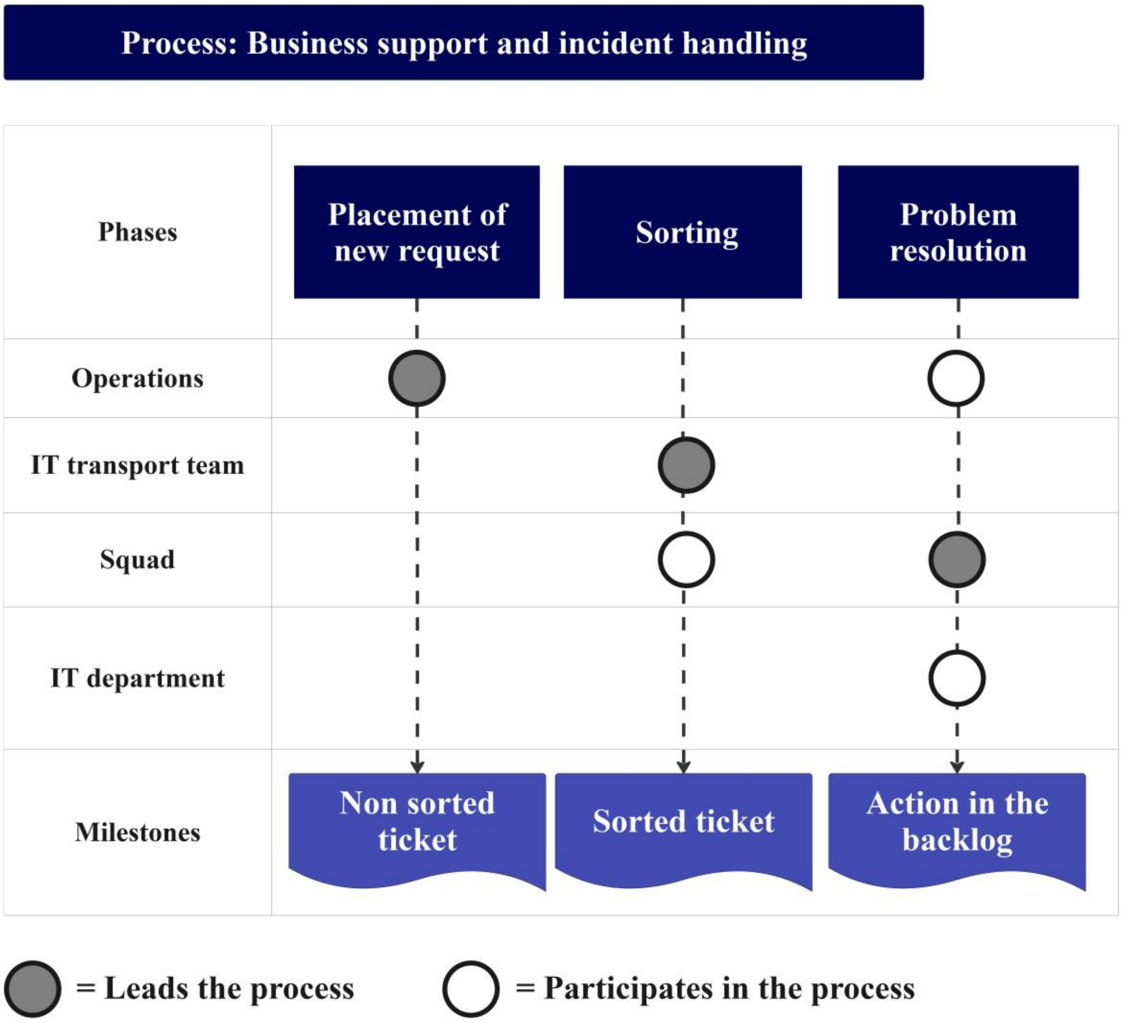

Figure A.1 - AS IS: Responsibility matrix representing the support process “Business support and incident handling”

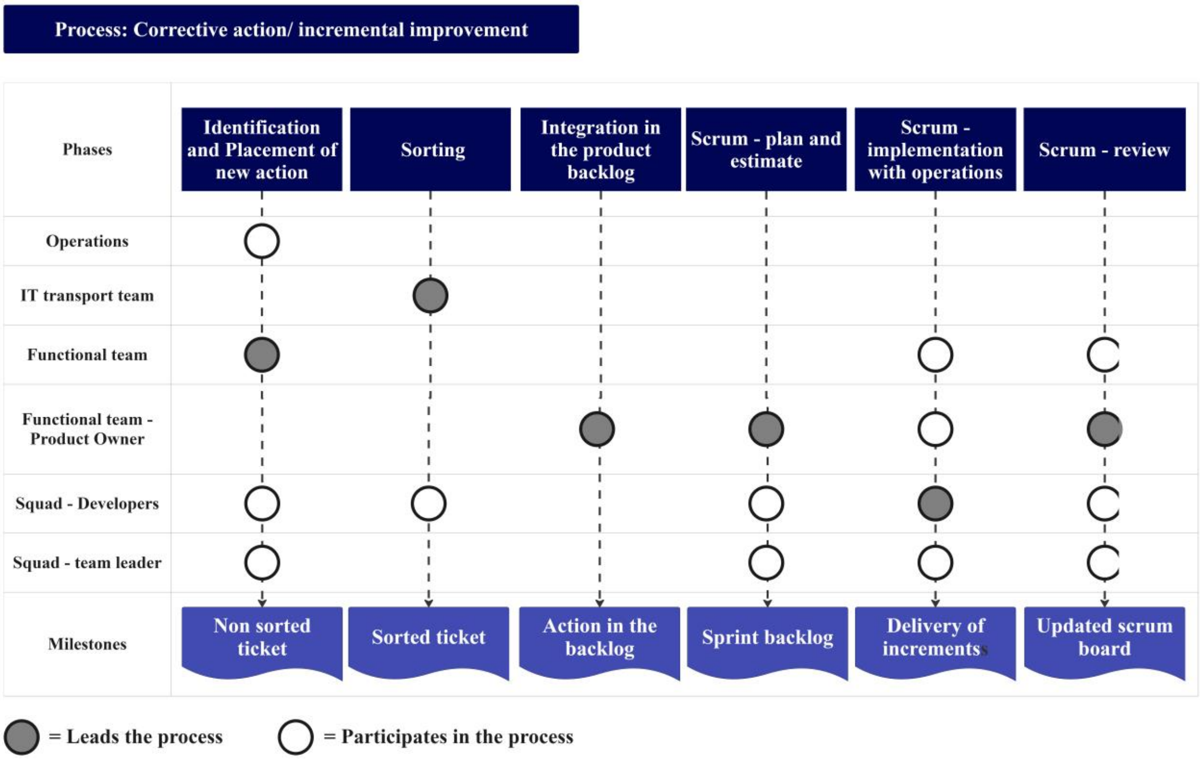

Figure A.2 - AS IS: Responsibility matrix representing both core process “Incremental improvement” and support process “Corrective action”

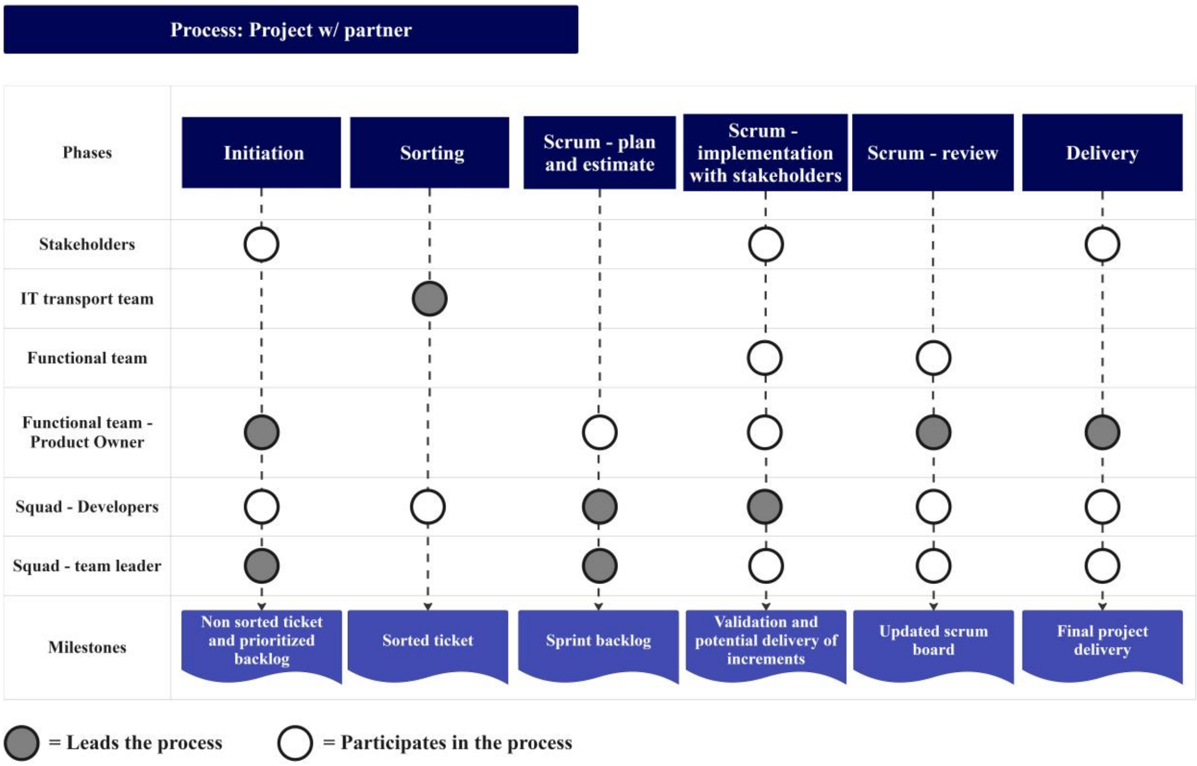

Figure A.3 - AS IS: Responsibility matrix representing the core process “Project with Partner”

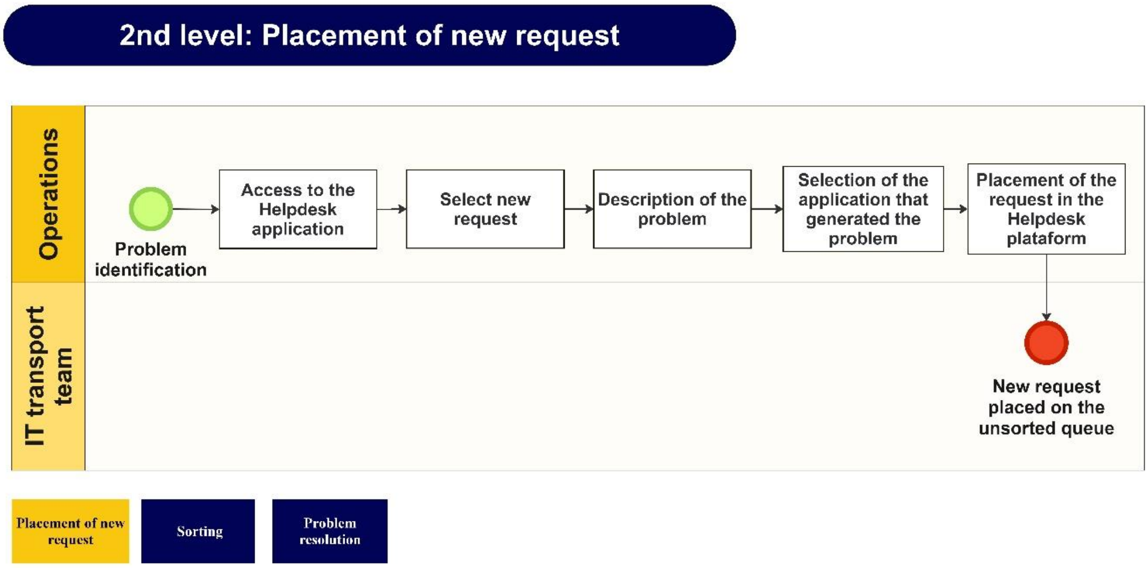

Figure A.4 - AS IS: Swimlane diagram representing the phase “Placement of new request”, from the process “Business support and incident handling”

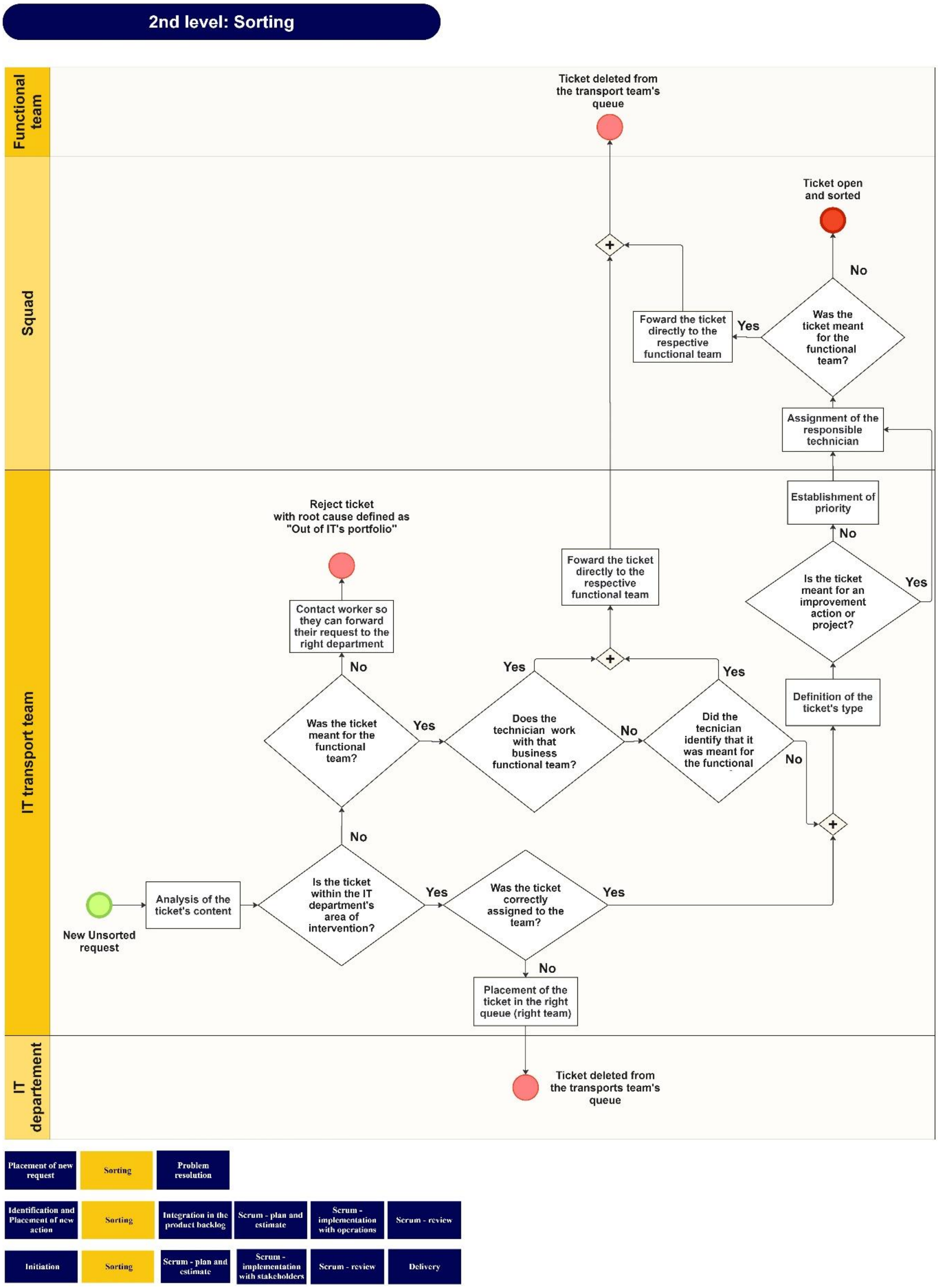

Figure A.5 - AS IS: Swimlane diagram representing the phase “Sorting”, common all the processes

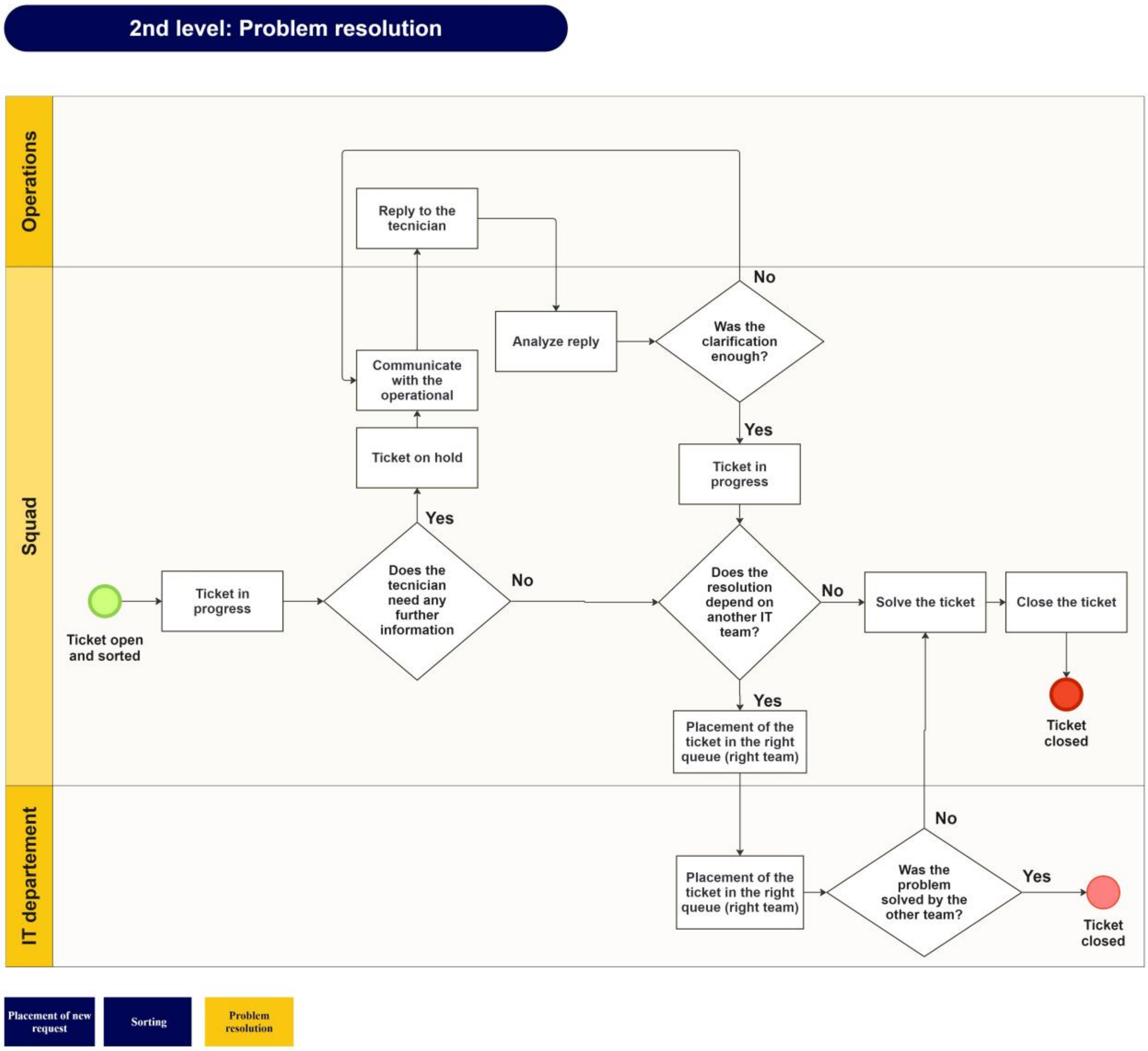

Figure A.6 - AS IS: Swimlane diagram representing the phase “Problem Resolution”, from the process “Business support and incident handling”

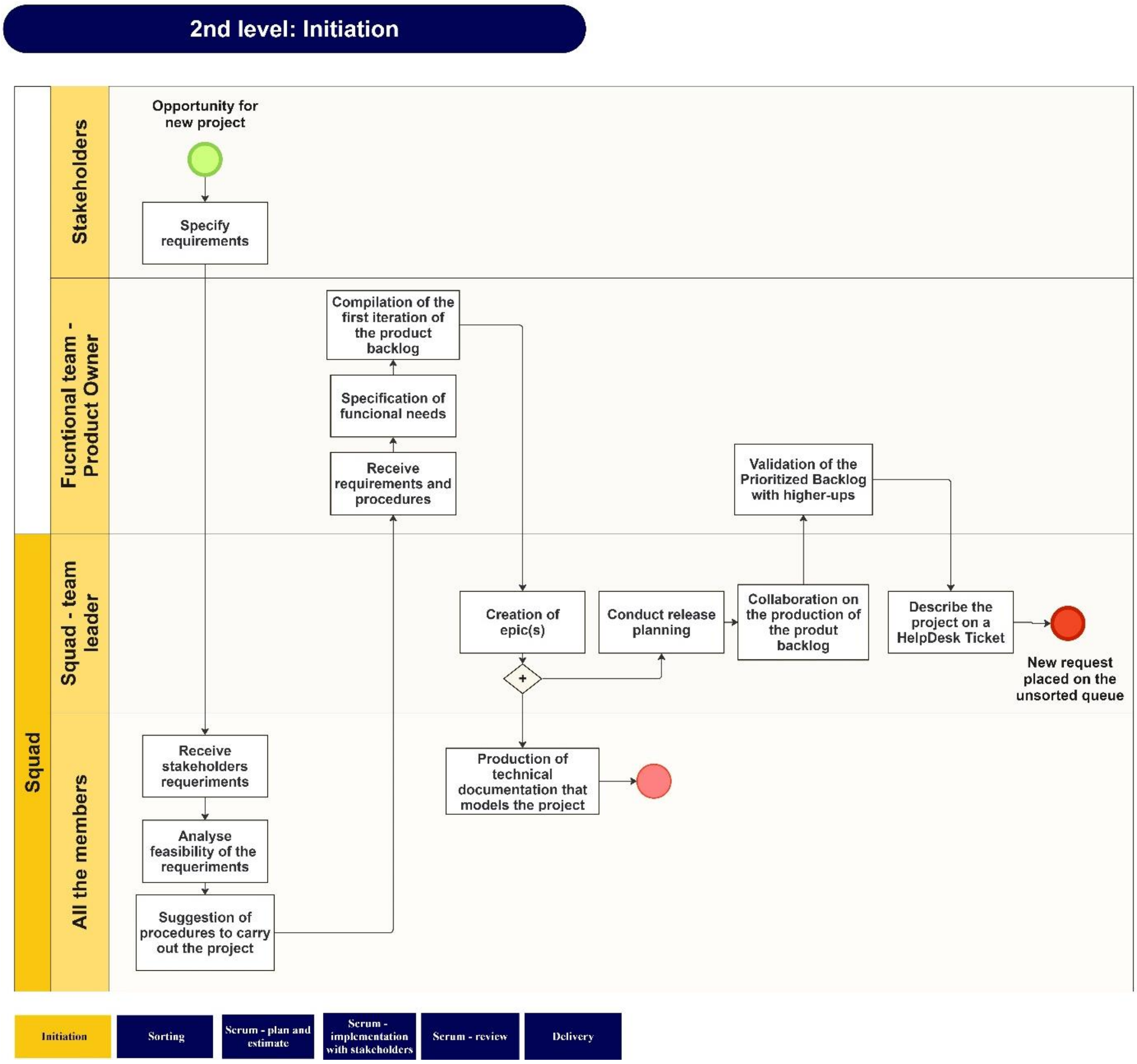

Figure A.7 - AS IS: Swimlane diagram representing the phase “Initiation”, from the process “Project with Partner”

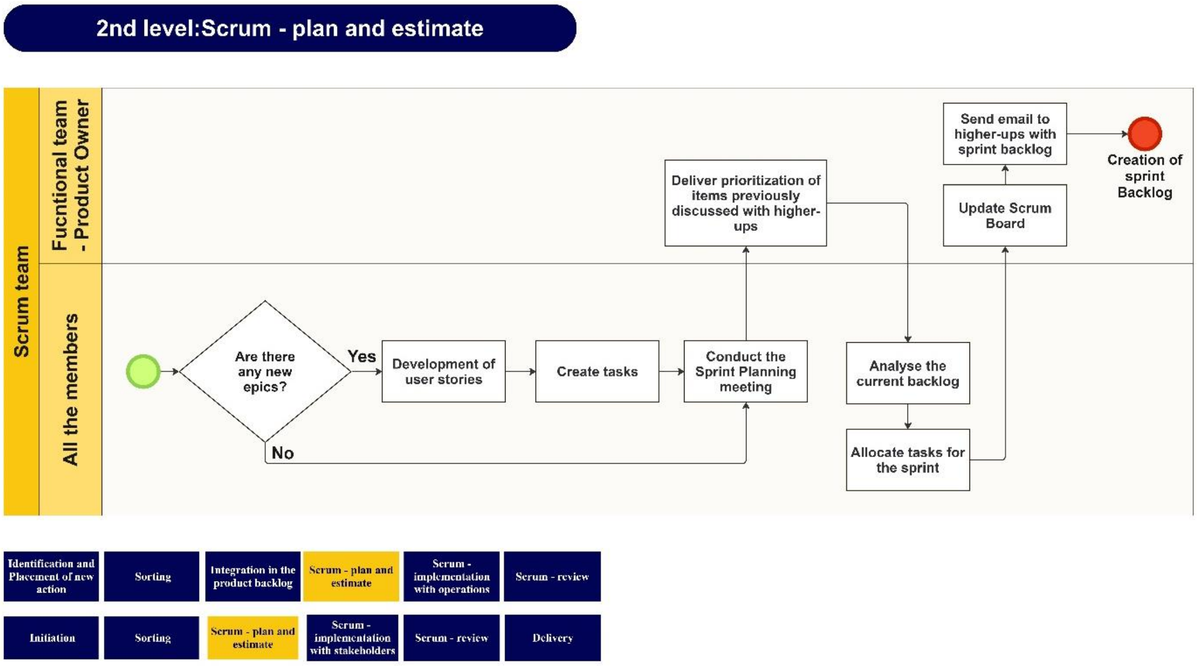

Figure A.8 - AS IS: Swimlane diagram representing the phase “Scrum – plan and estimate”, common to the core process “Project with partner”, “Incremental improvement” and support process “Corrective action”

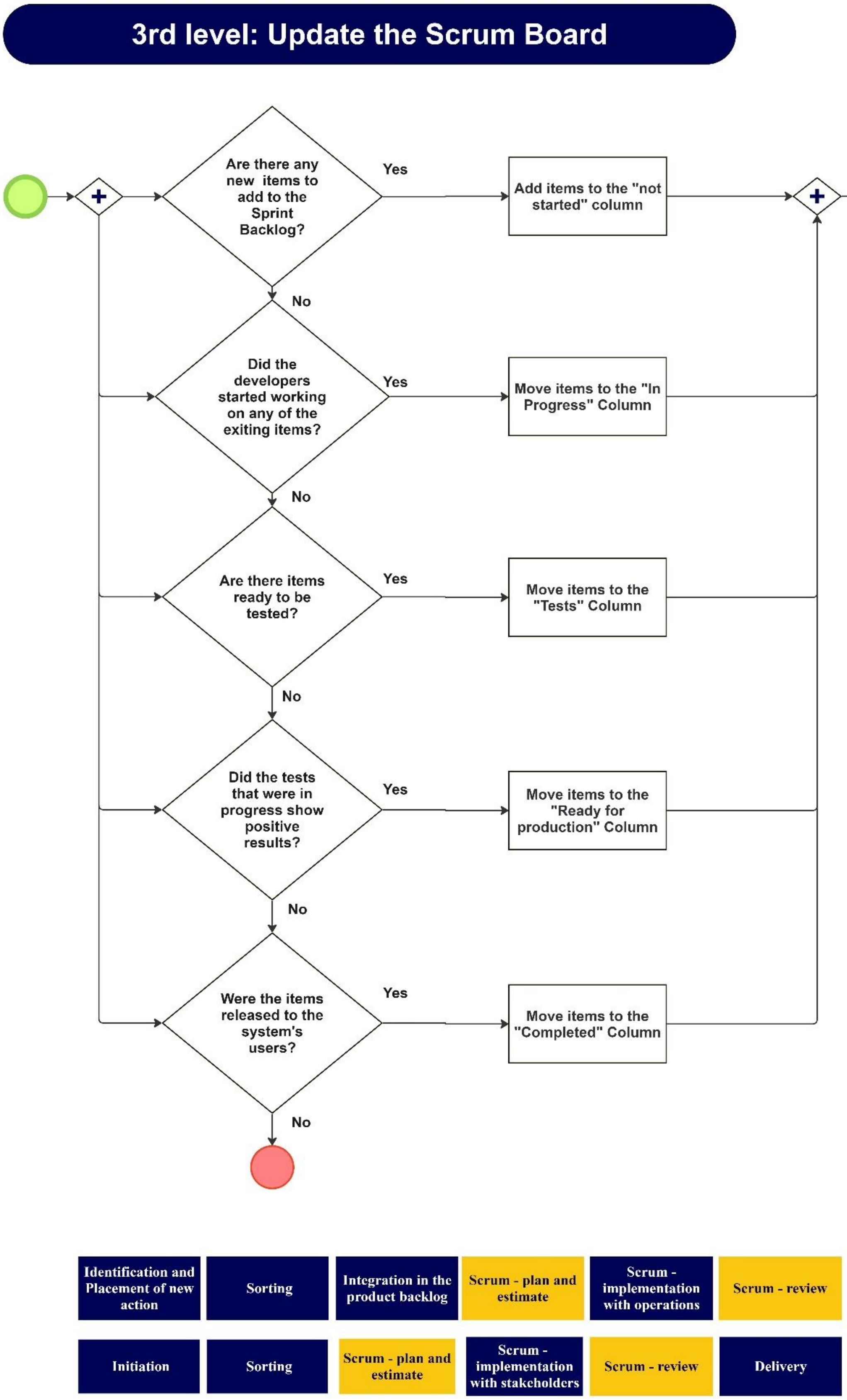

Figure A. 9 - AS IS: Flowchart diagram representing the task “Update Scrum Board”

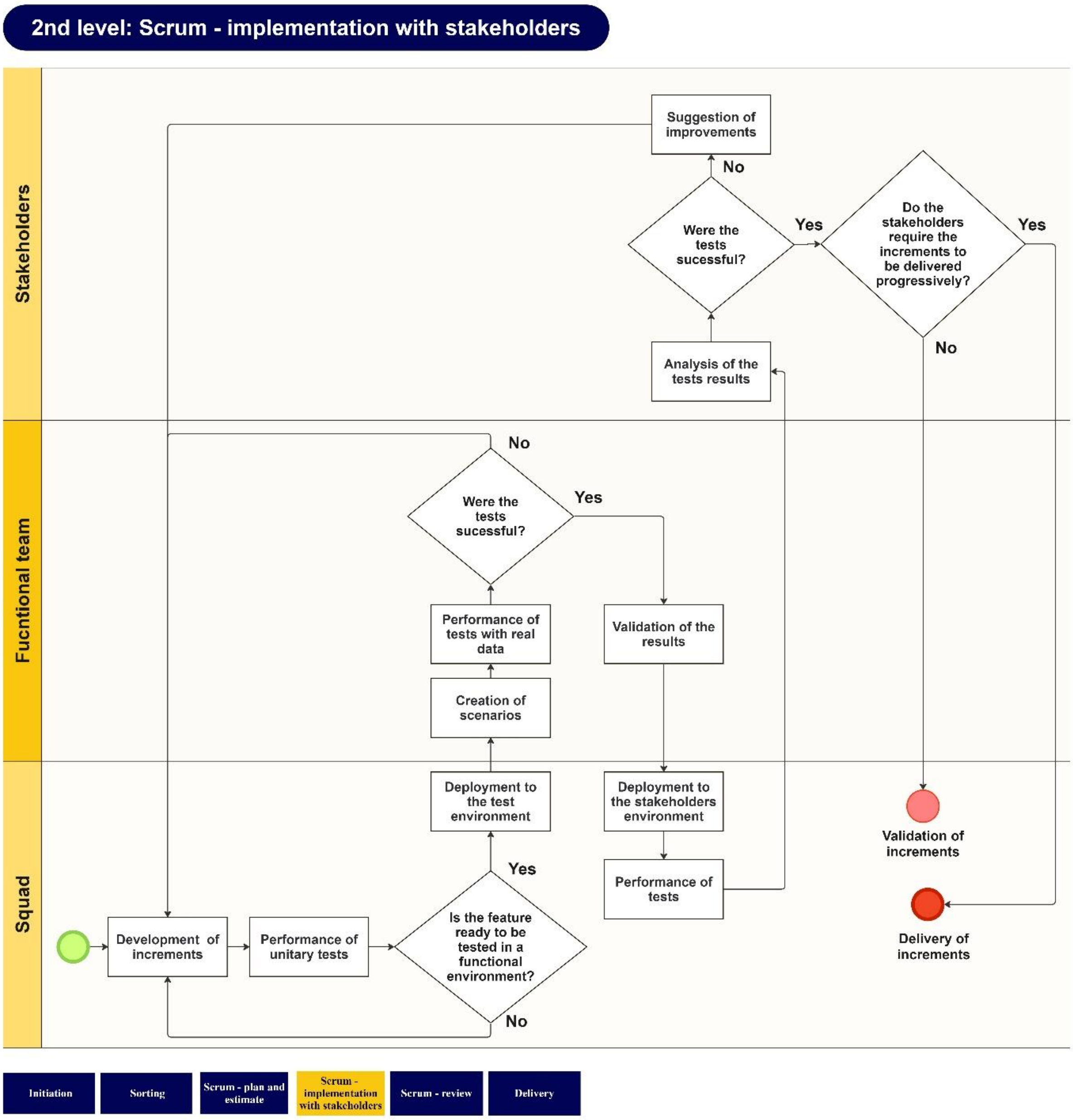

Figure A.10 - AS IS: Swimlane diagram representing the phase “Scrum – implementation with stakeholders”, from the process “Project with Partner”

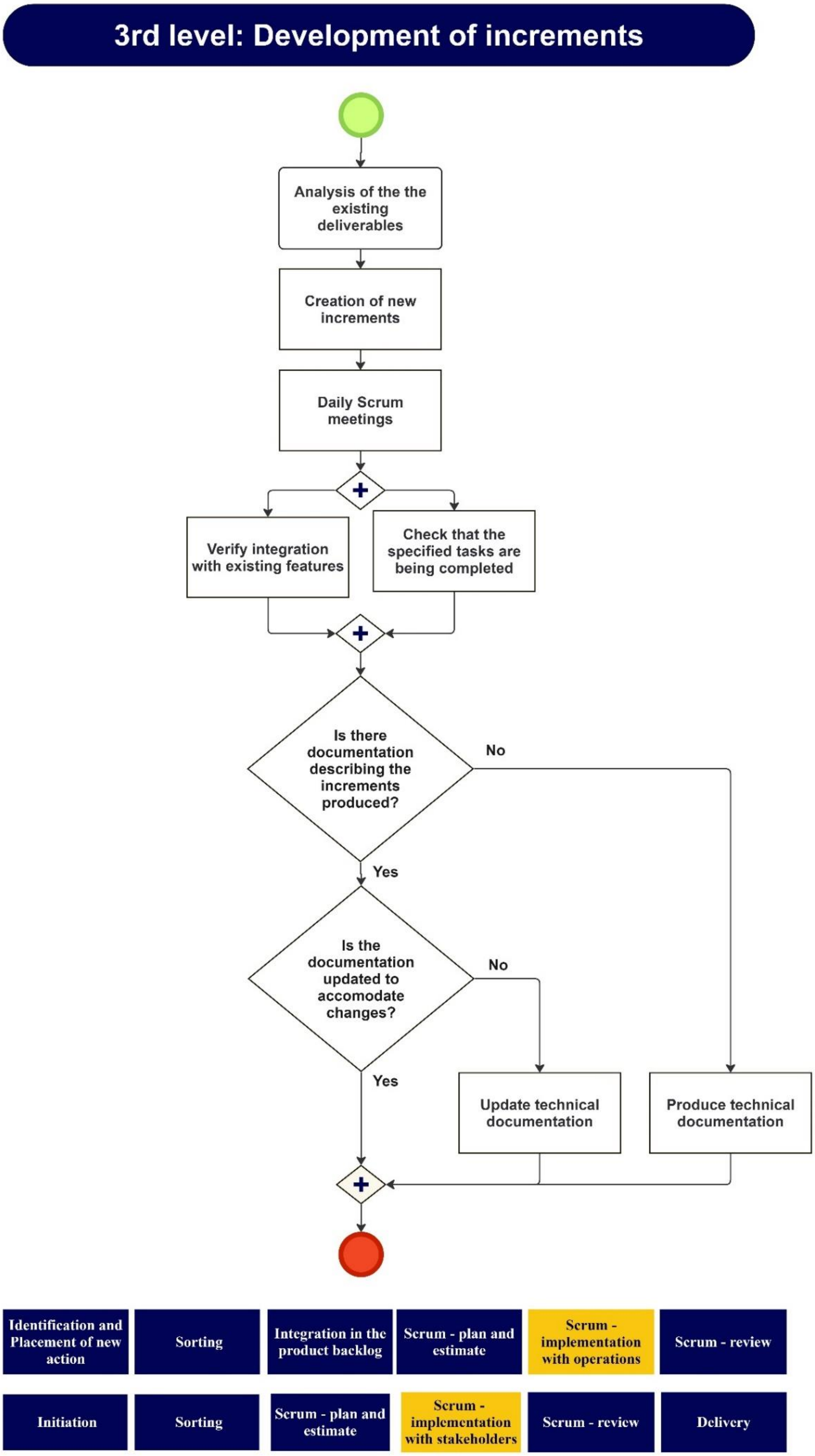

Figure A.11 - AS IS: Flowchart diagram representing the task “Development of increments”

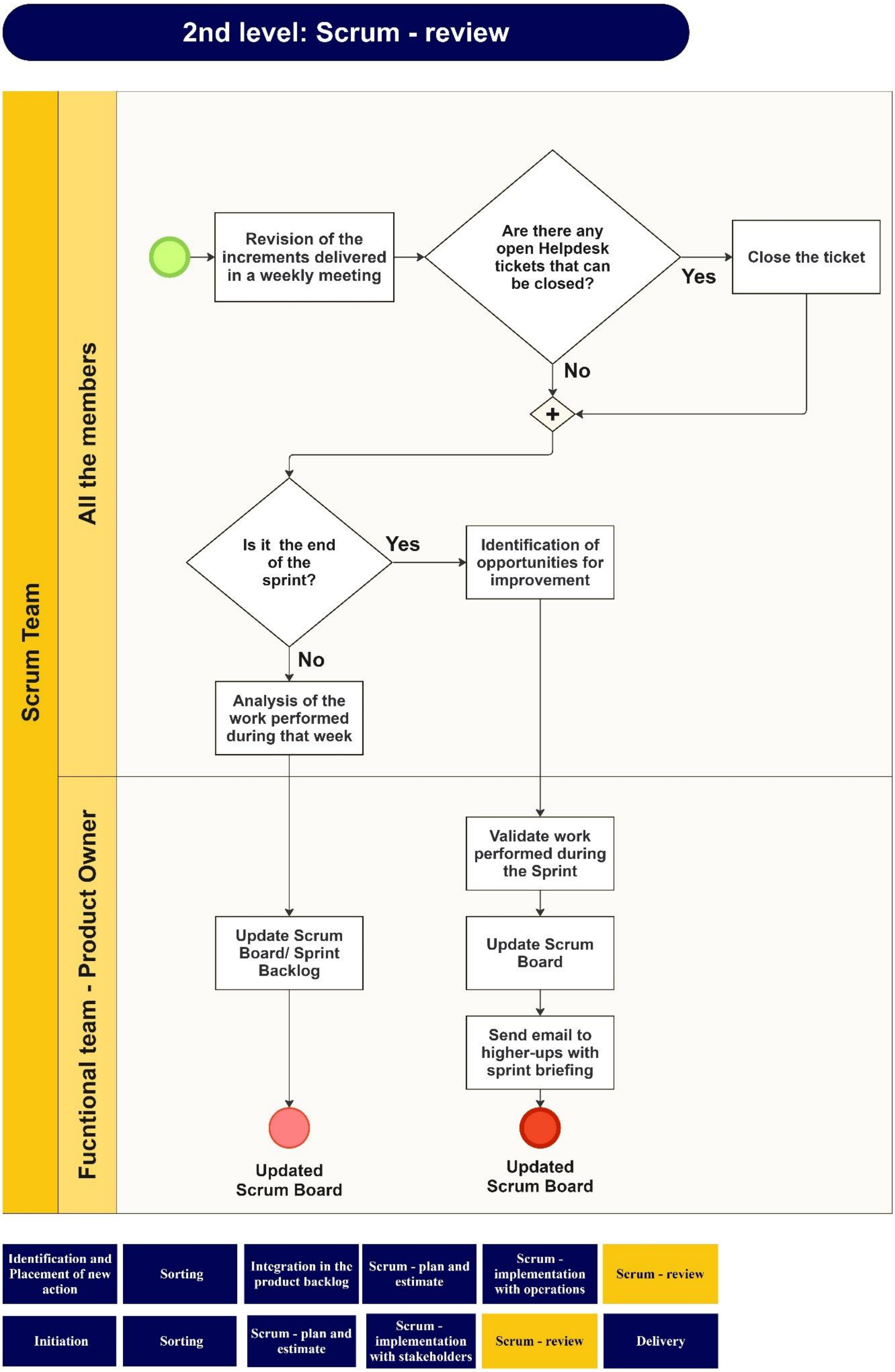

Figure A. 12 - AS IS: Swimlane diagram representing the phase “Scrum – review”, common to the core process “Project with partner”, “Incremental improvement” and support process “Corrective action”

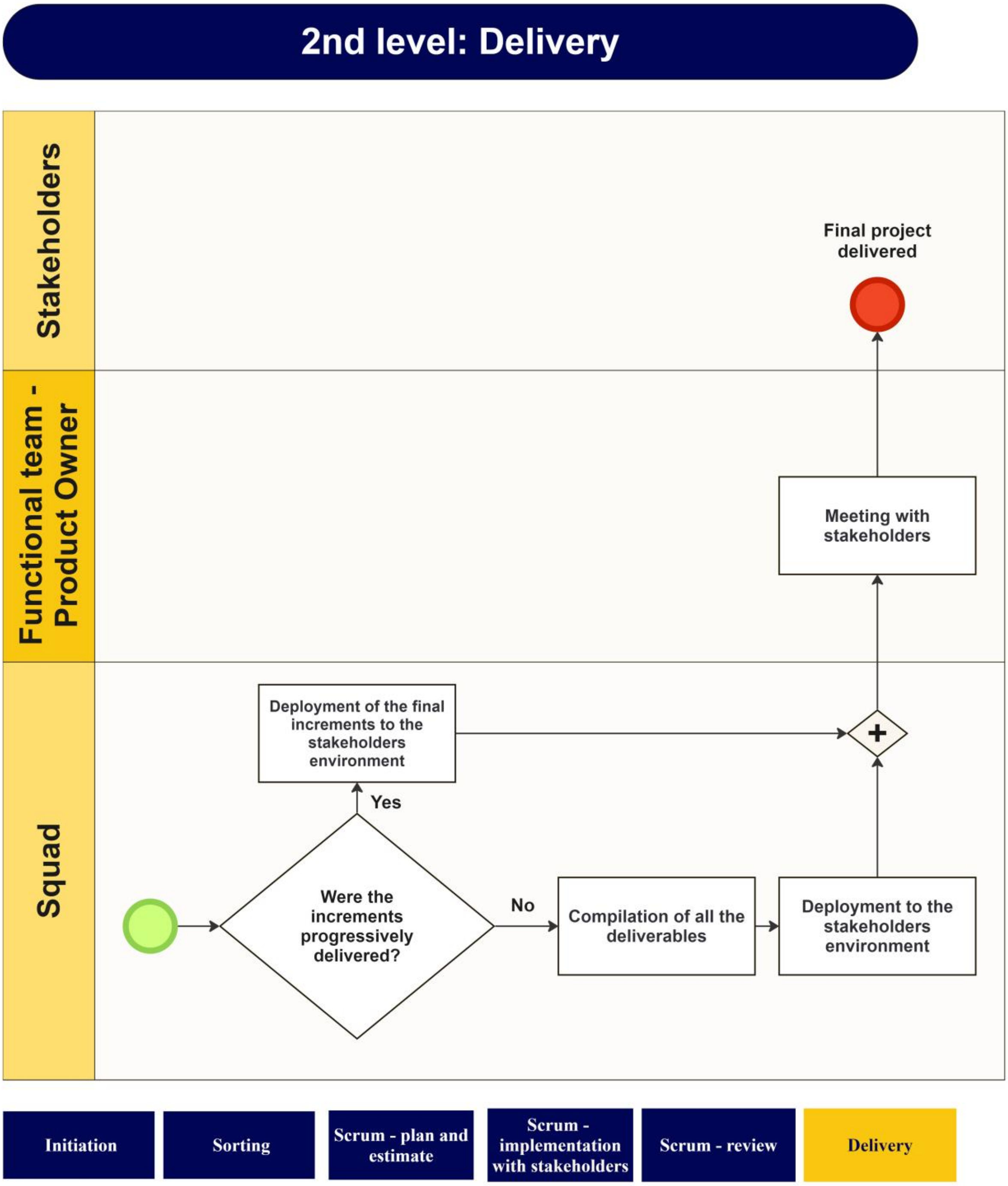

Figure A.13 - AS IS: Swimlane diagram representing the phase "Delivery", from the process "Project with Partner"

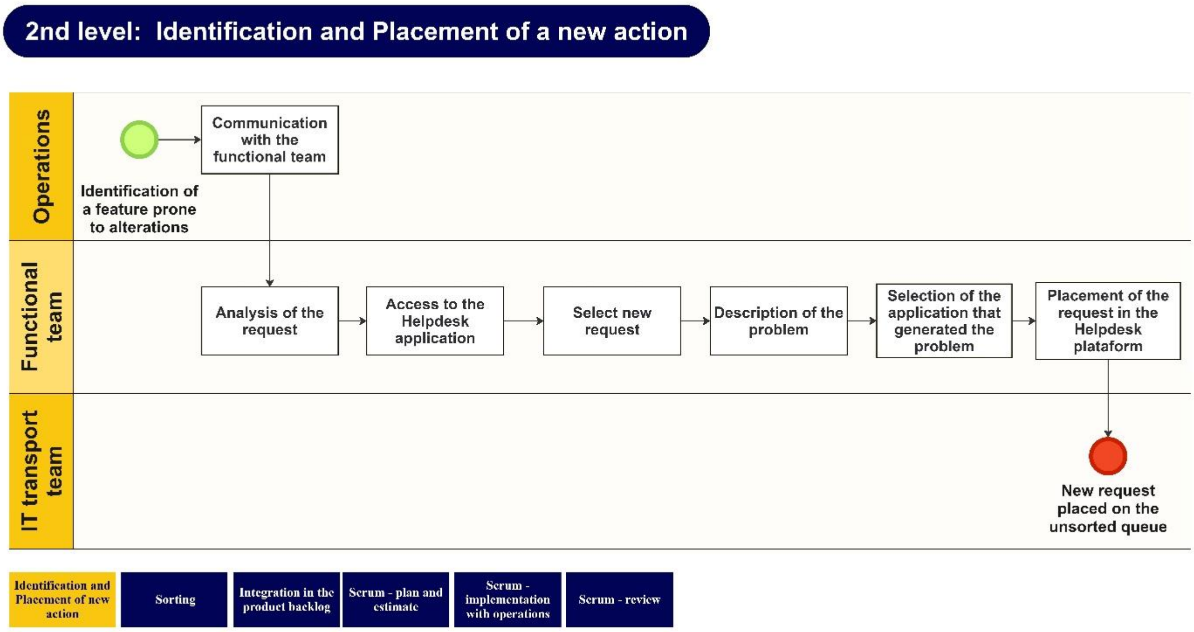

Figure A.14 - AS IS: Swimlane diagram representing the phase "Identification and Placement of a new action", common to the core process "Incremental improvement" and support process "Corrective action"

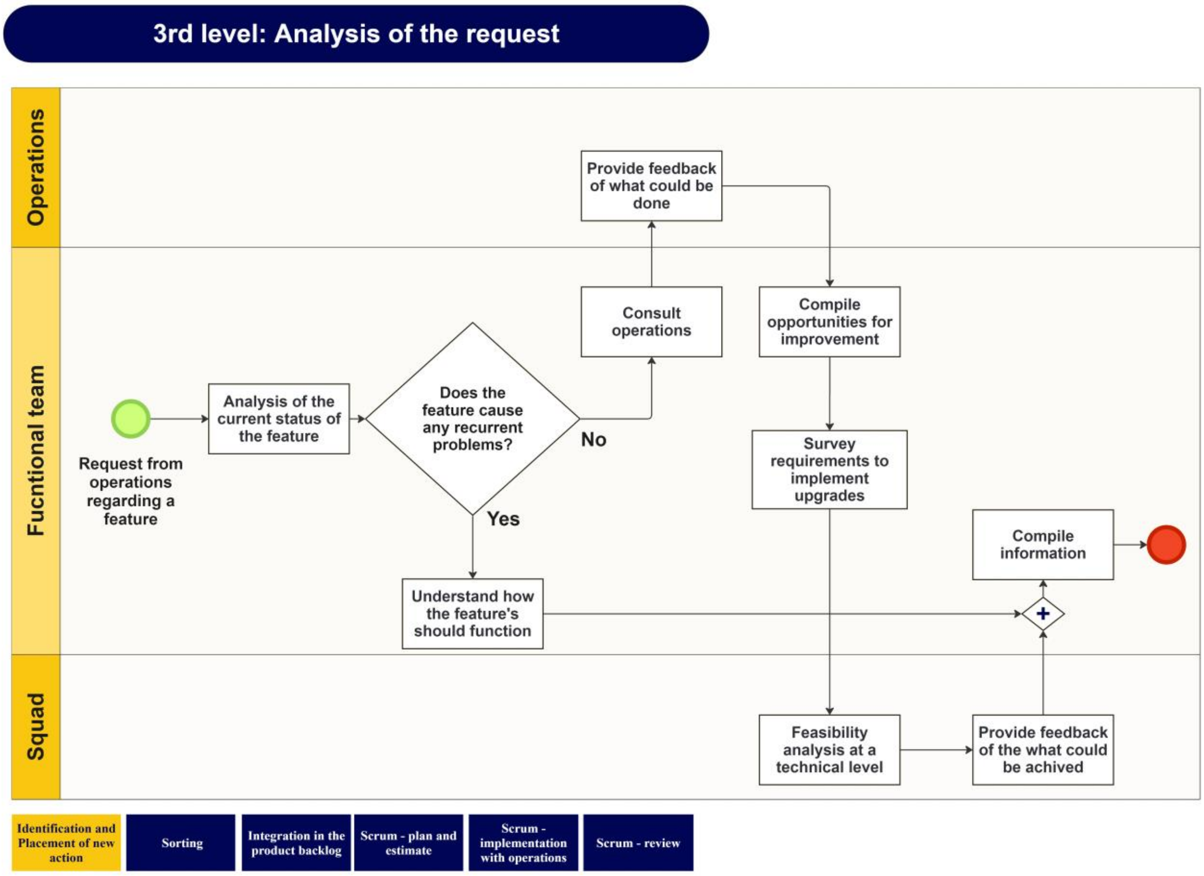

Figure A.15 - AS IS: Swimlane diagram representing the task "Analysis of request", common to the core process "Incremental improvement" and support process "Corrective action"

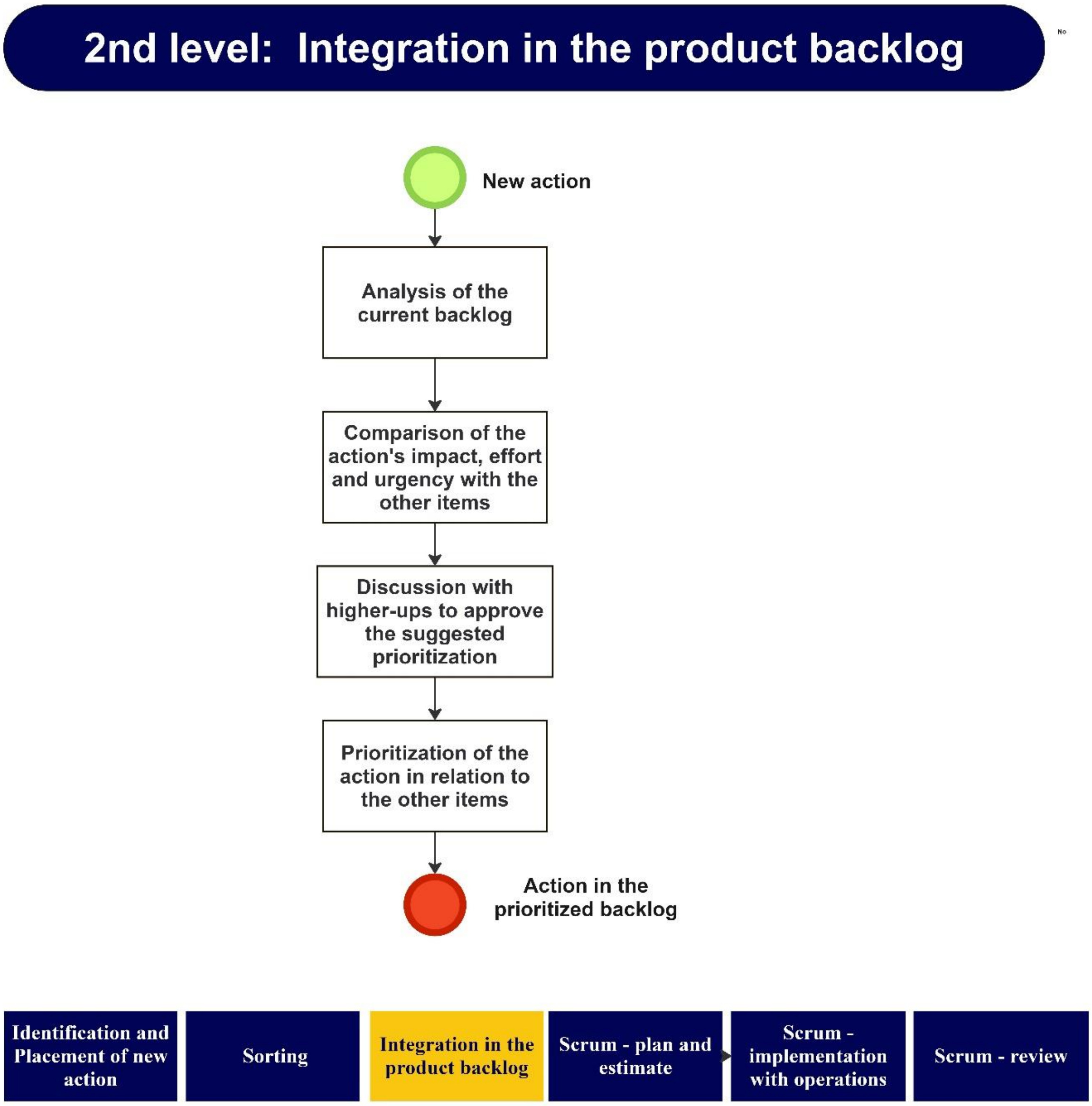

Figure A.16 - AS IS: Swimlane diagram representing the phase "Integration in the product Backlog", common to the core process "Incremental improvement" and support process "Corrective action"

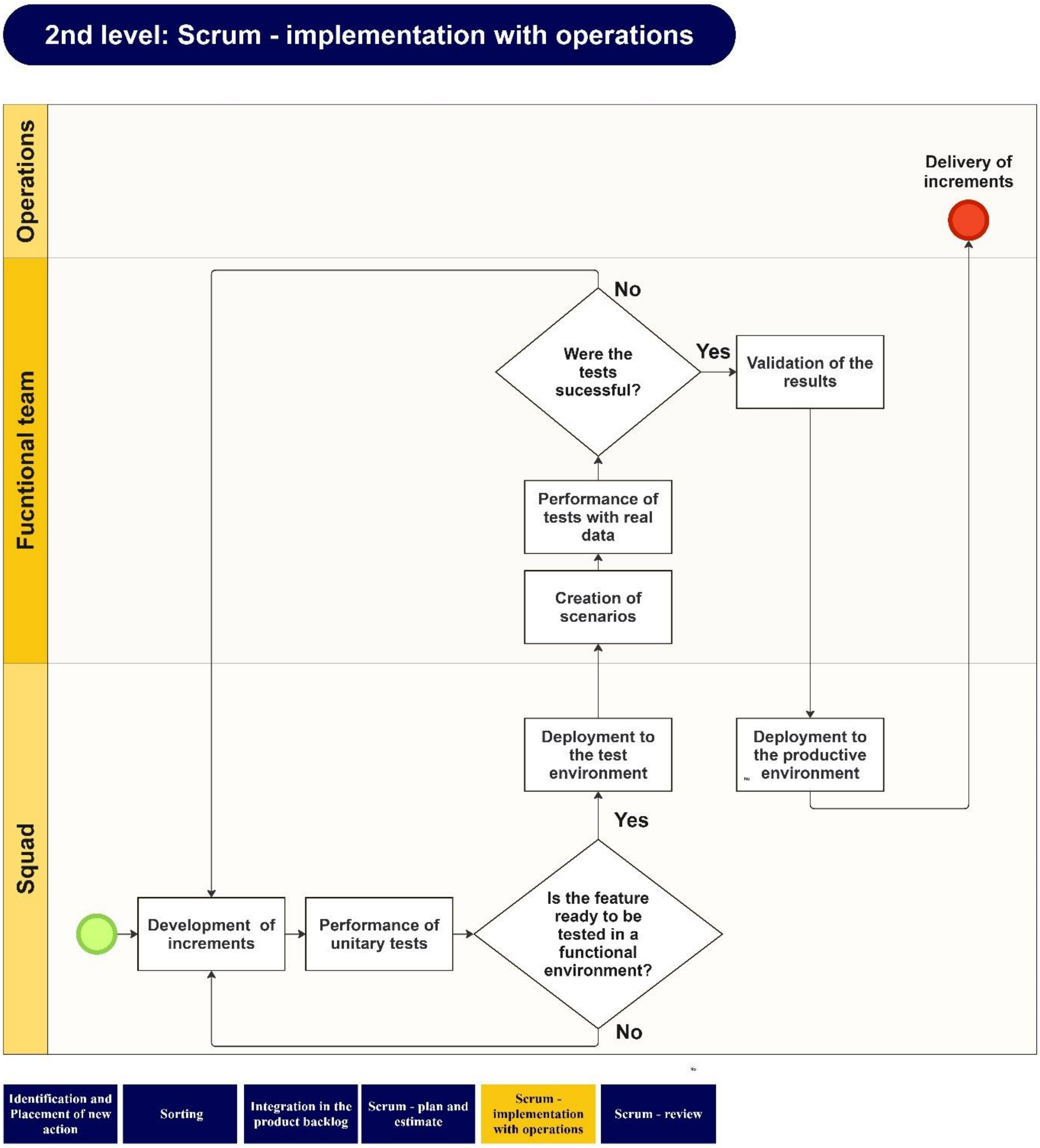

Figure A.17 - AS IS: Swimlane diagram representing the phase "Scrum – implementation with operations", common to the core process "Incremental improvement" and support process "Corrective action"

Appendix B: TO BE modeling and solutions

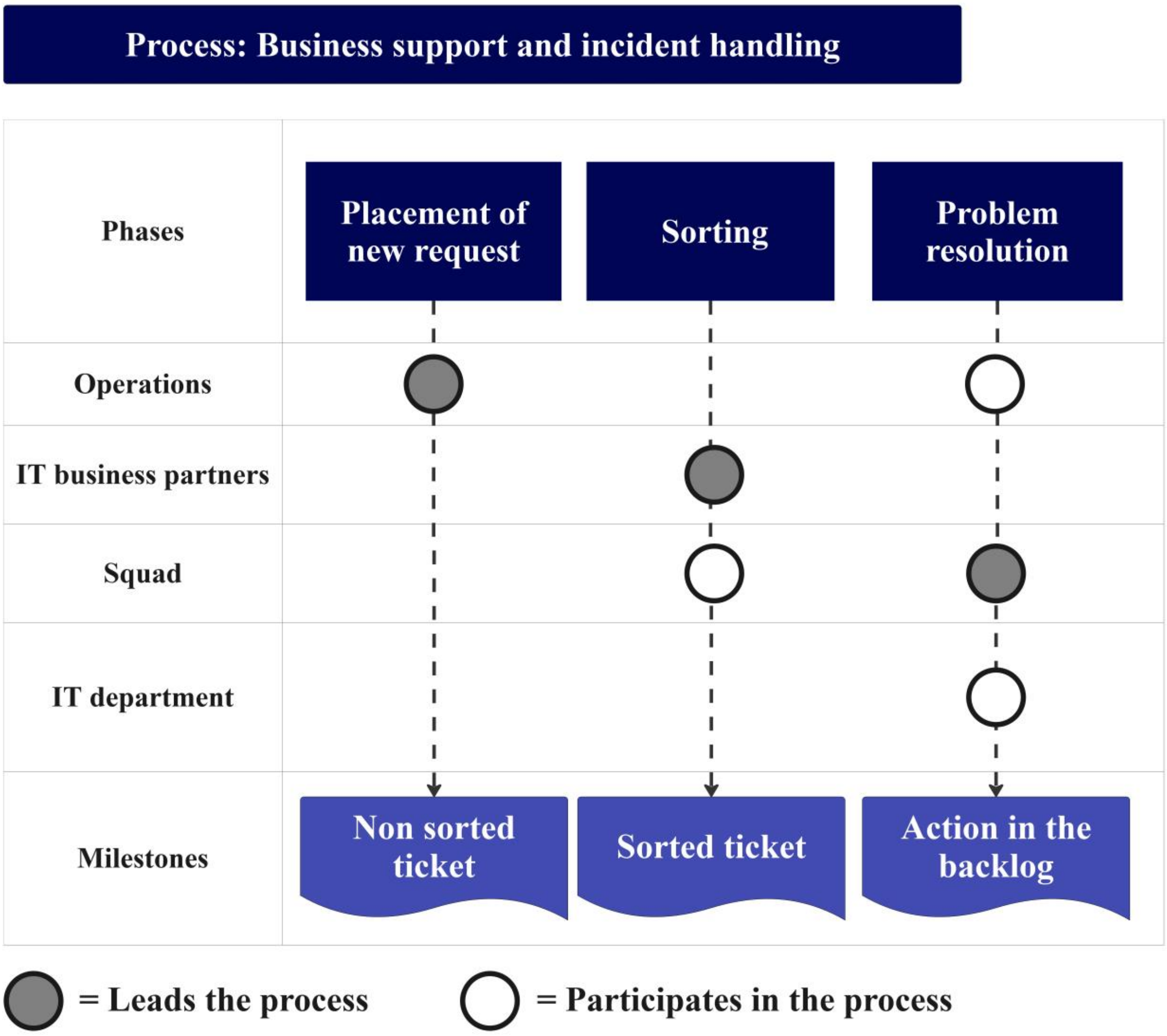

Figure B.1 – TO BE: Responsibility matrix representing the support process “Business support and incident handling”

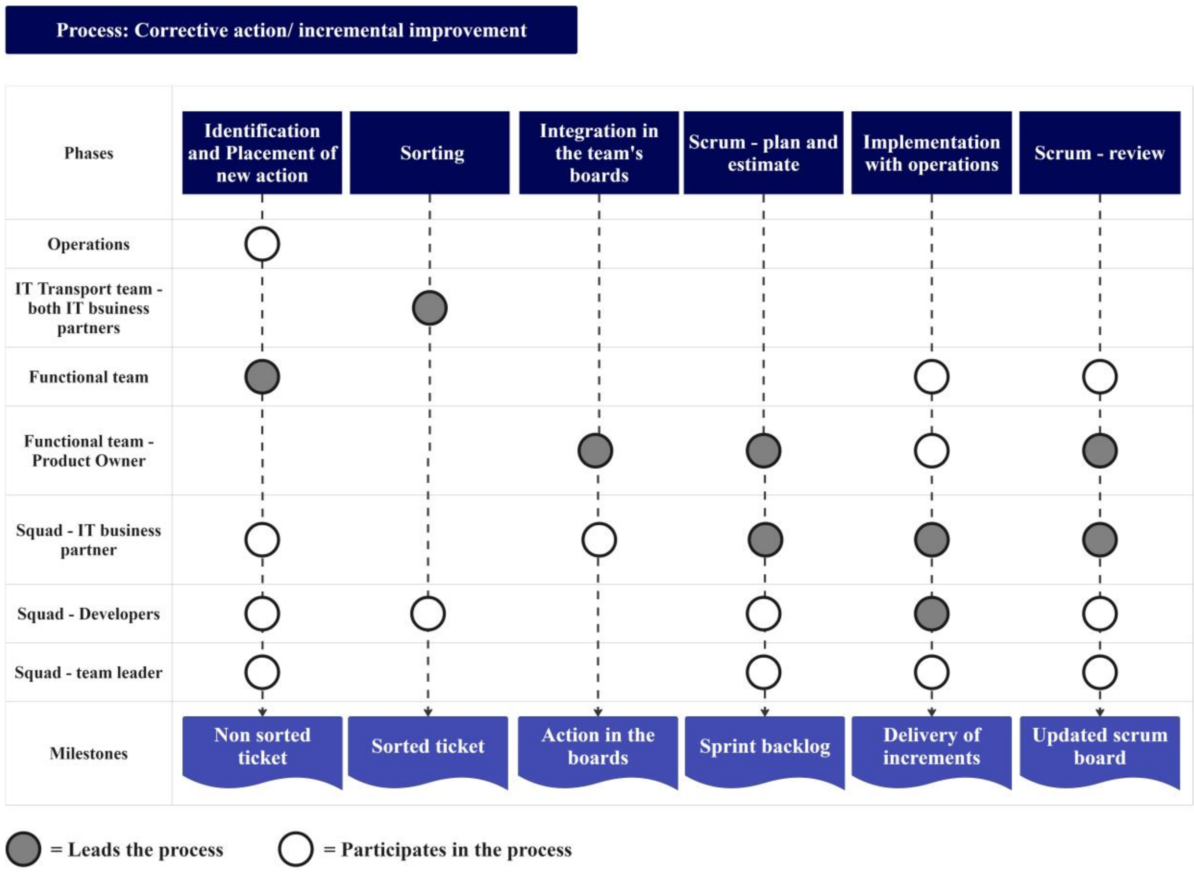

Figure B.2 - TO BE: Responsibility matrix representing both the core process “Incremental improvement” and the support process “Corrective action”

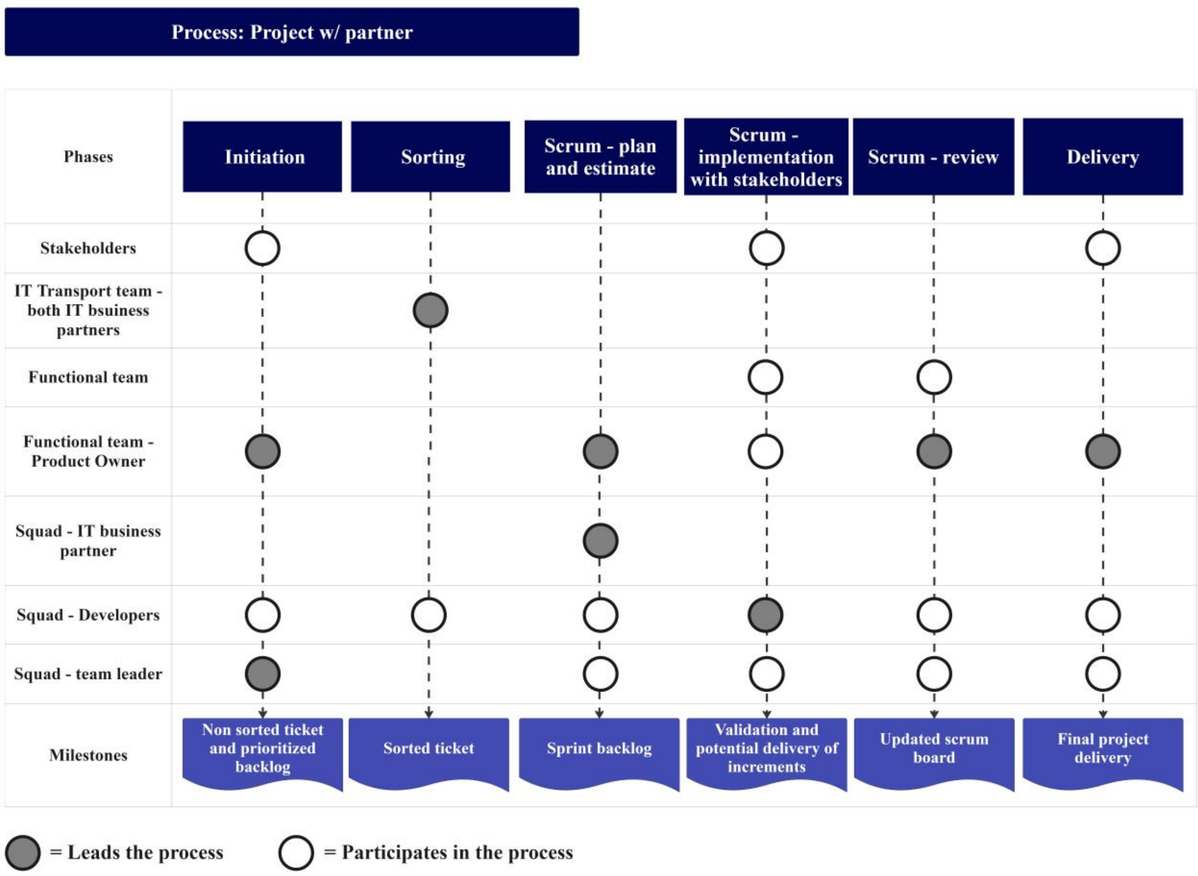

Figure B.3 - TO BE: Responsibility matrix representing the core process “Project with Partner”

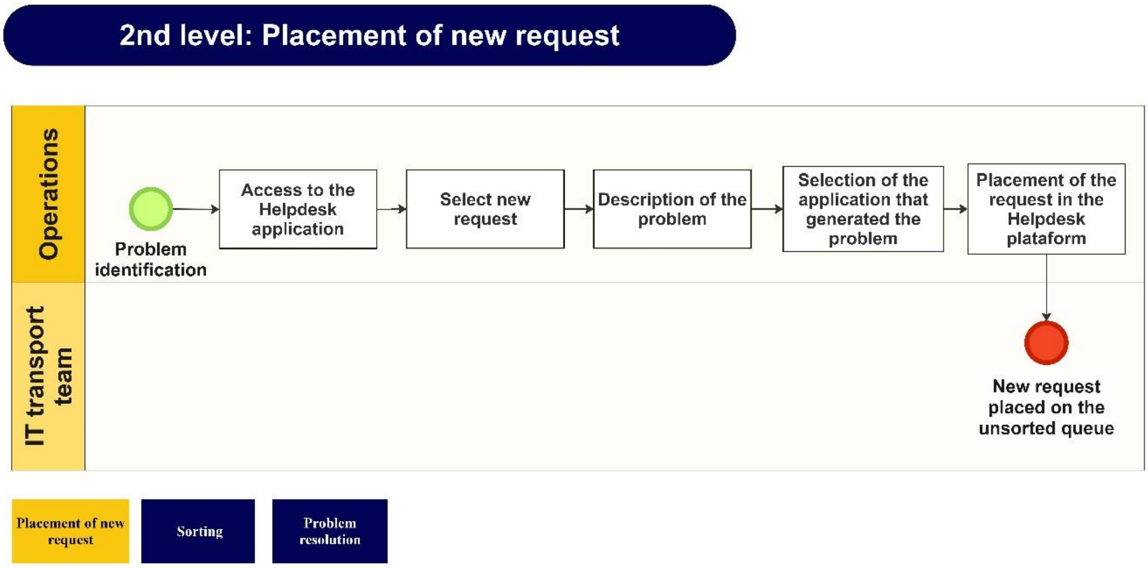

Figure B.4 – TO BE: Swimlane diagram representing the phase “Placement of new request”, from the support process “Business support and incident handling”

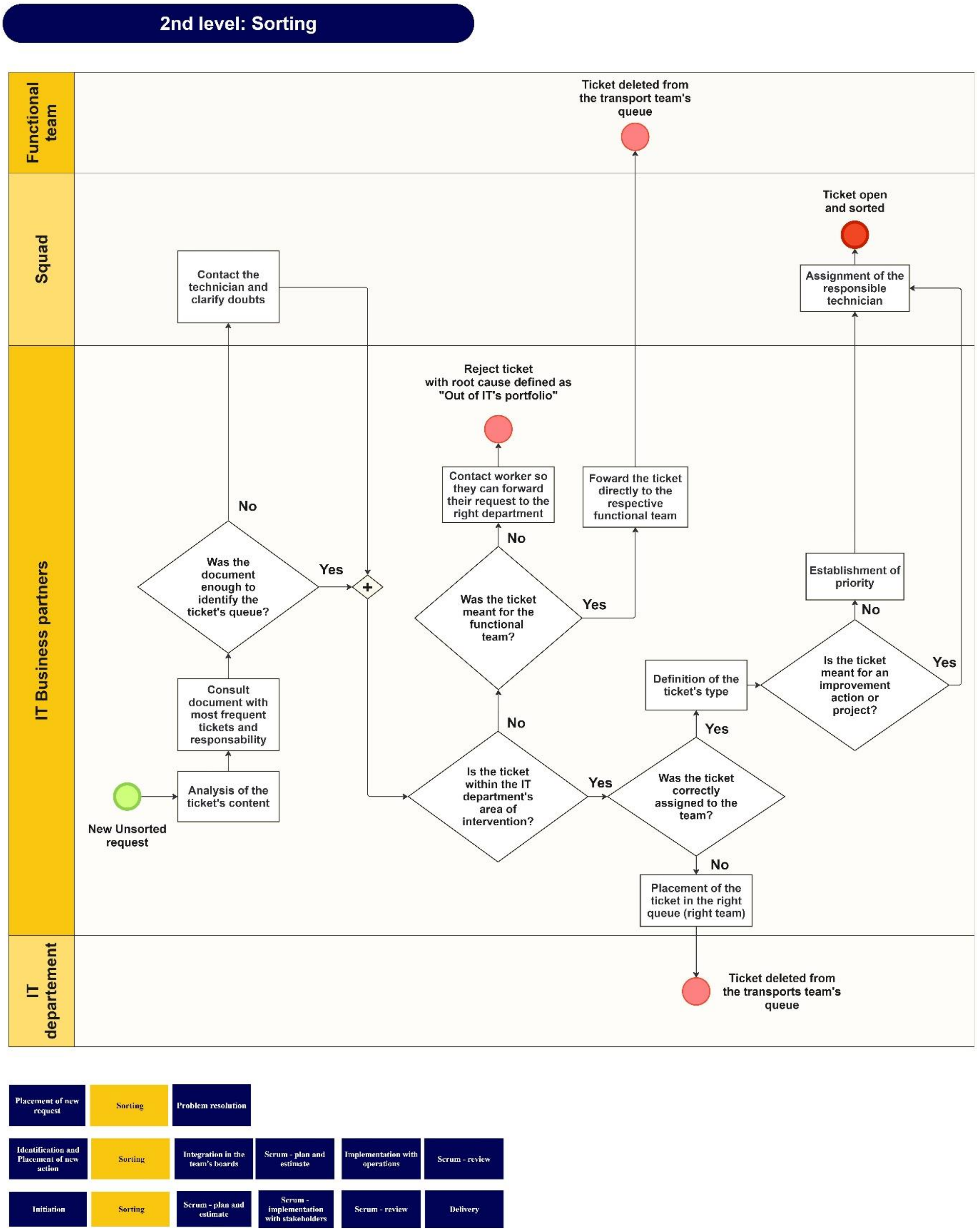

Figure B.5 - TO BE: Swimlane diagram representing the phase “Sorting”, common to all the processes

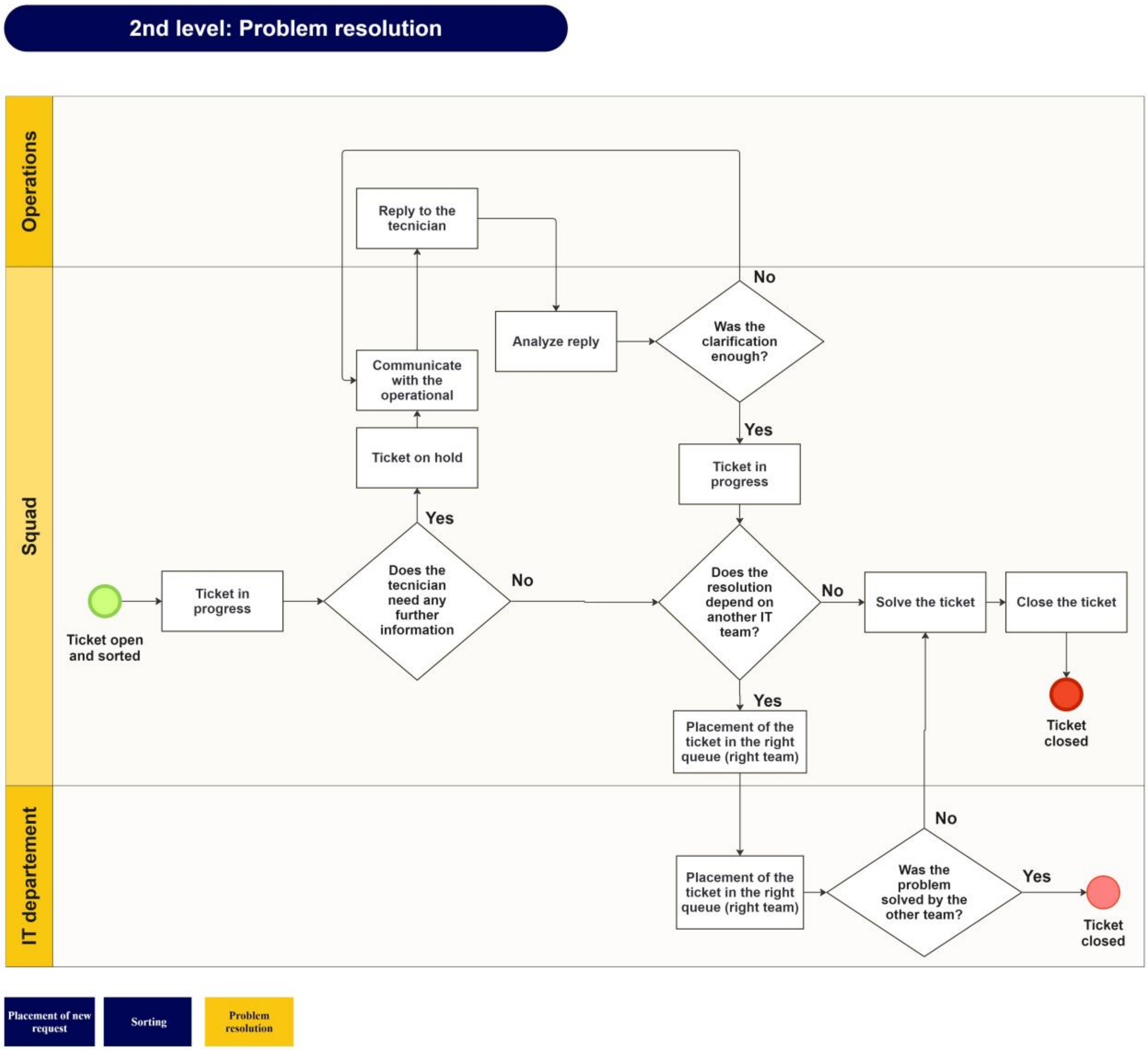

Figure B.6 – TO BE: Swimlane diagram representing the phase “Problem Resolution”, from the support process “Business support and incident handling”

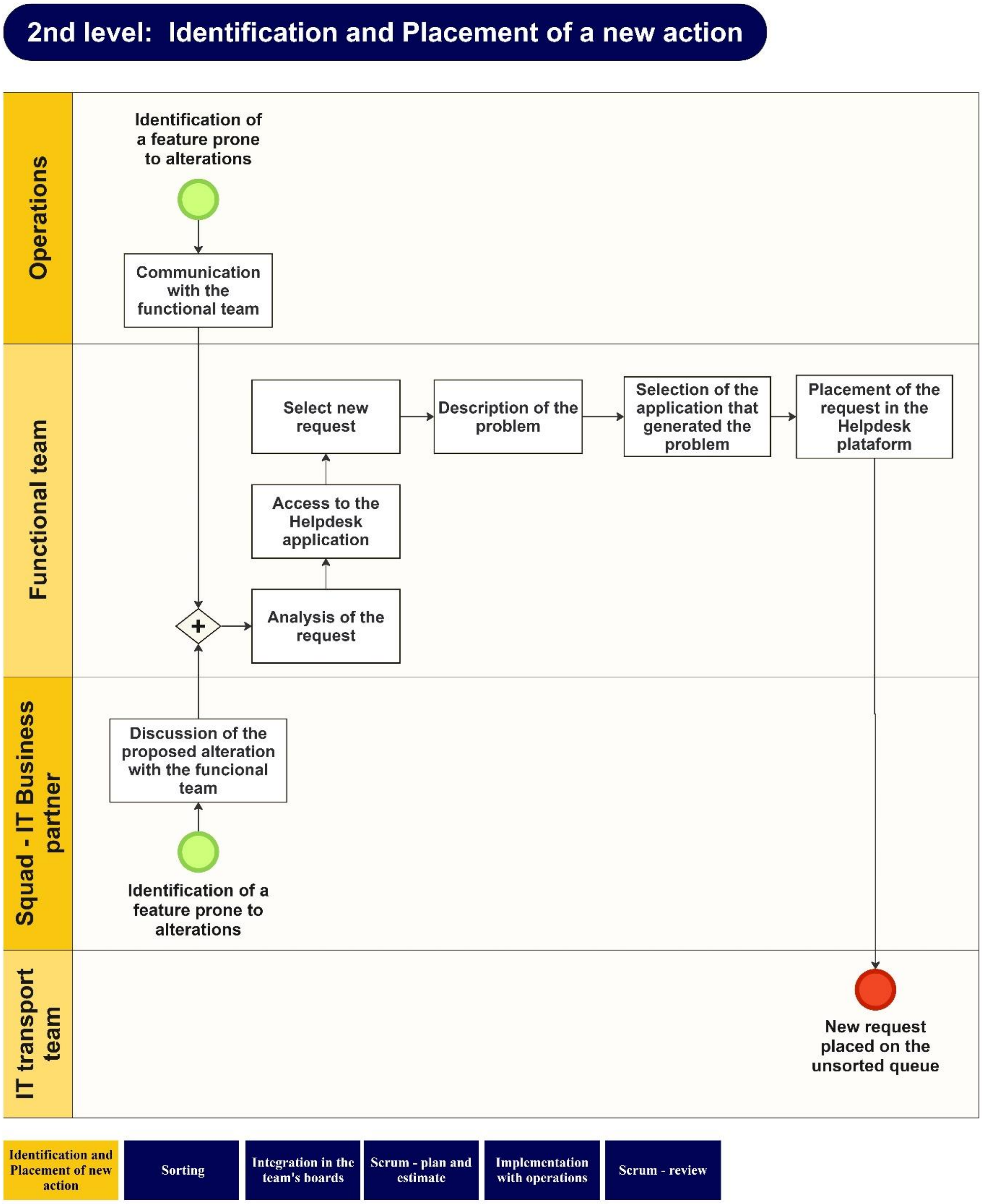

Figure B.7 – TO BE: Swimlane diagram representing the phase "Identification and Placement of a new action", common to the core process "Incremental improvement" and the support process "Corrective action"

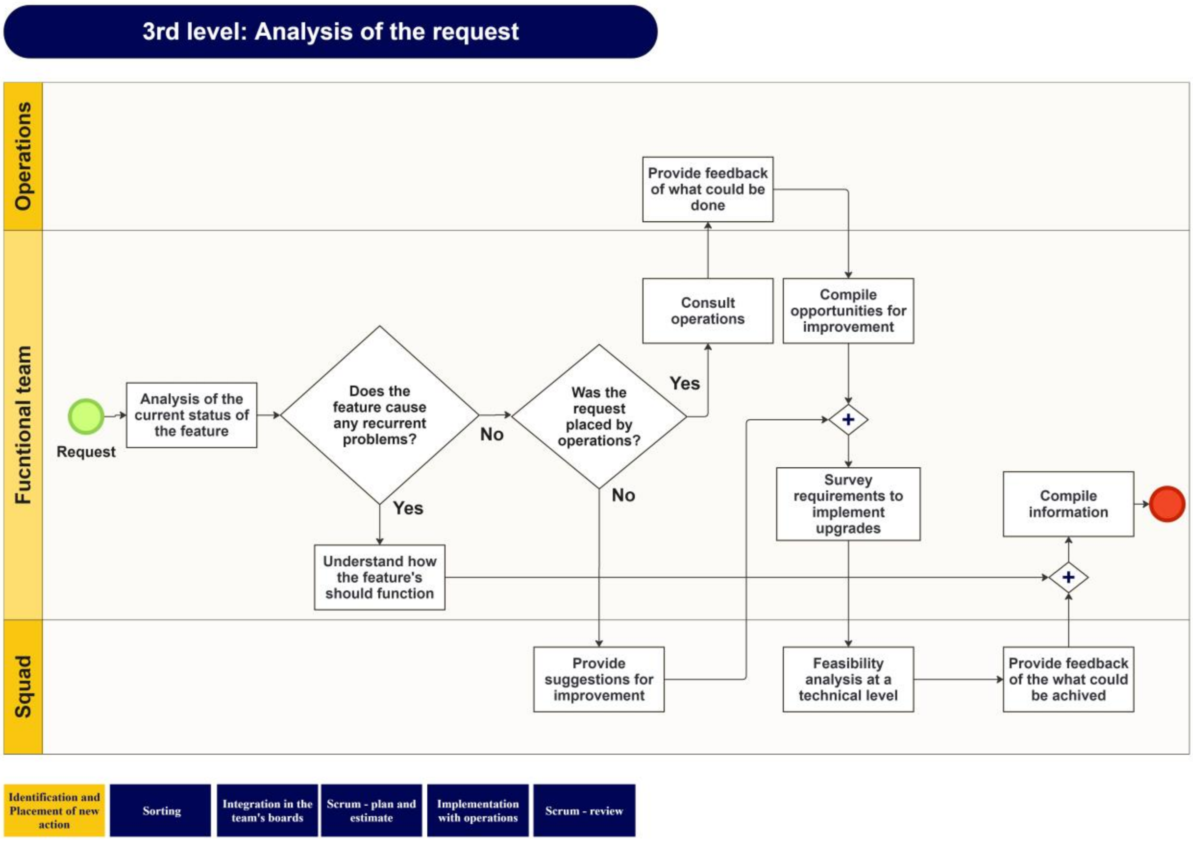

Figure B.8 – TO BE: Swimlane diagram representing the task "Analysis of request", common to the core process "Incremental improvement" and the support process "Corrective action"

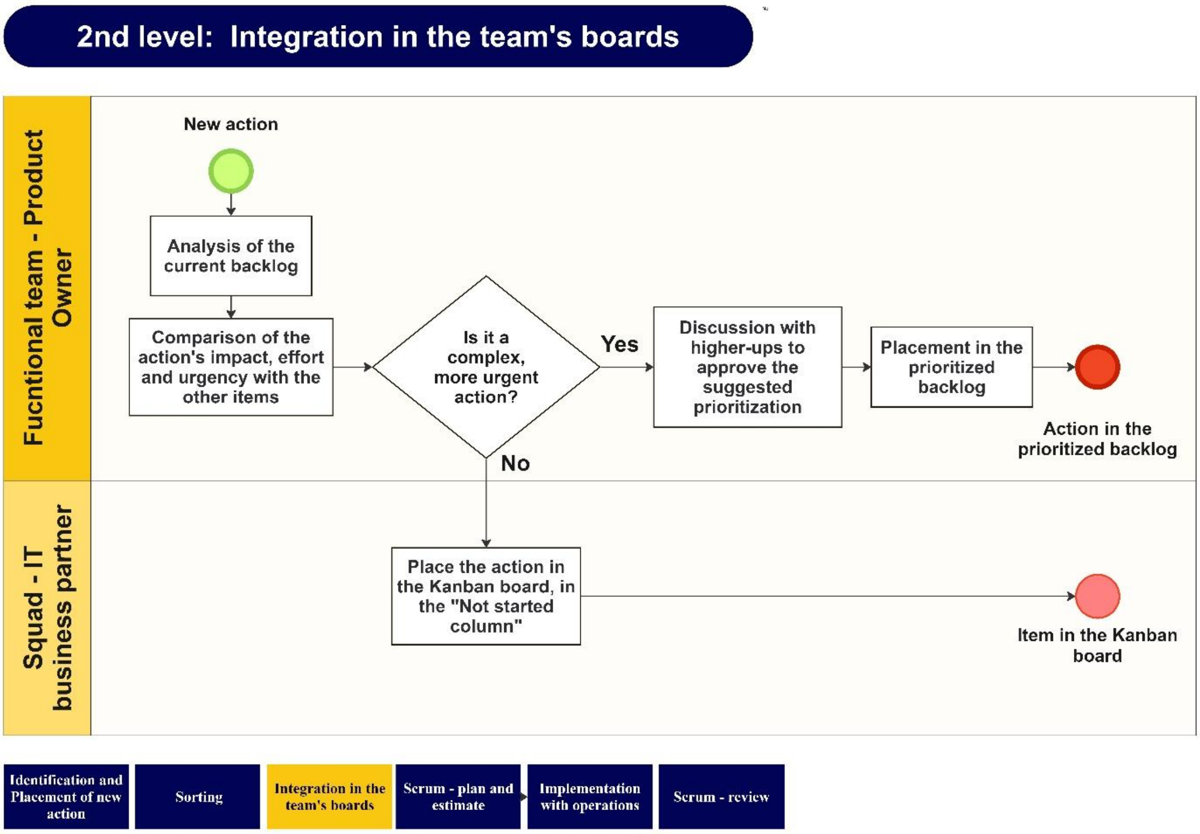

Figure B.9 – TO BE: Swimlane diagram representing the phase "Integration in the team's boards", common to the core process "Incremental improvement" and the support process "Corrective action"

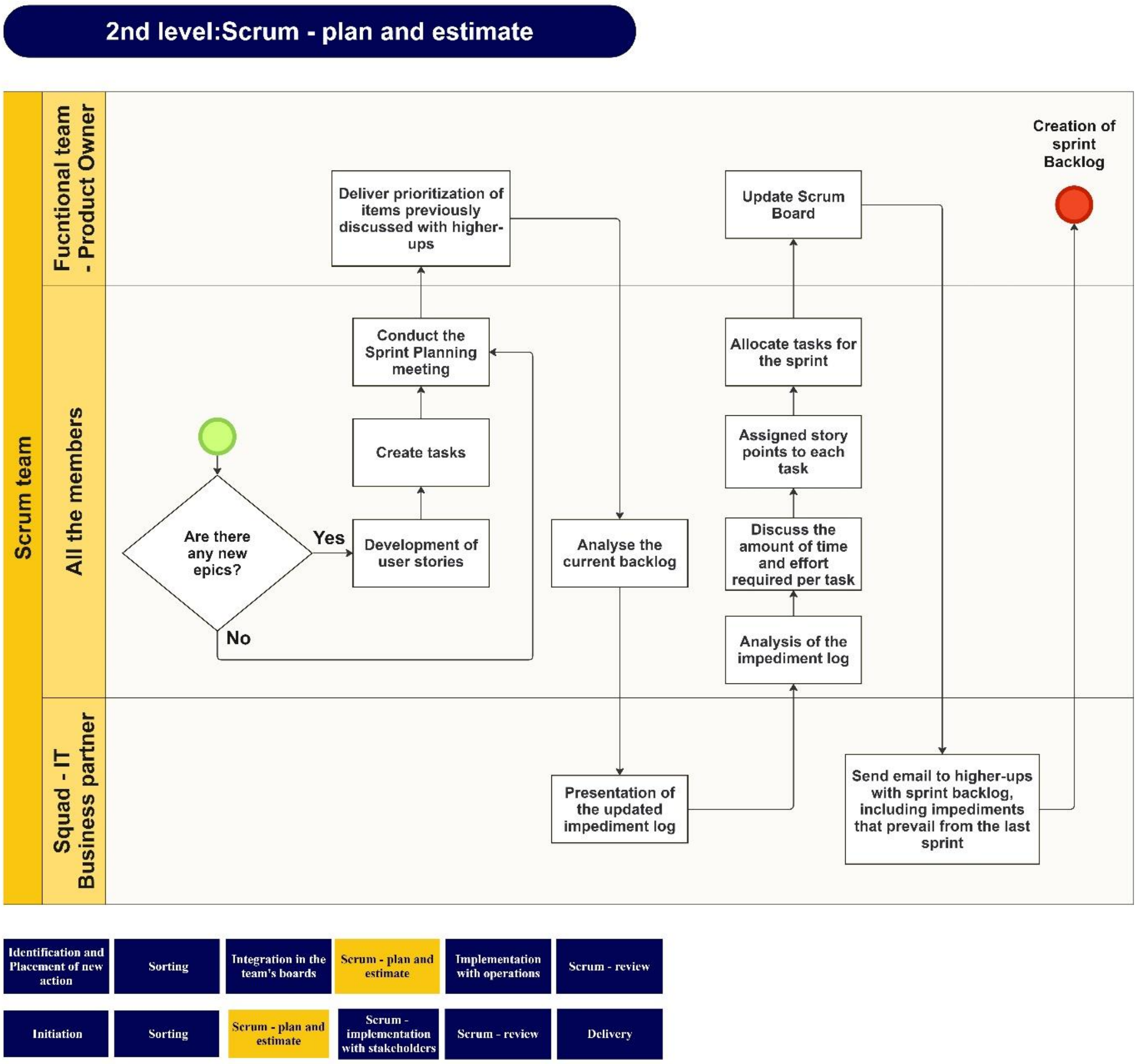

Figure B.10 – TO BE: Swimlane diagram representing the phase “Scrum – plan and estimate”, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

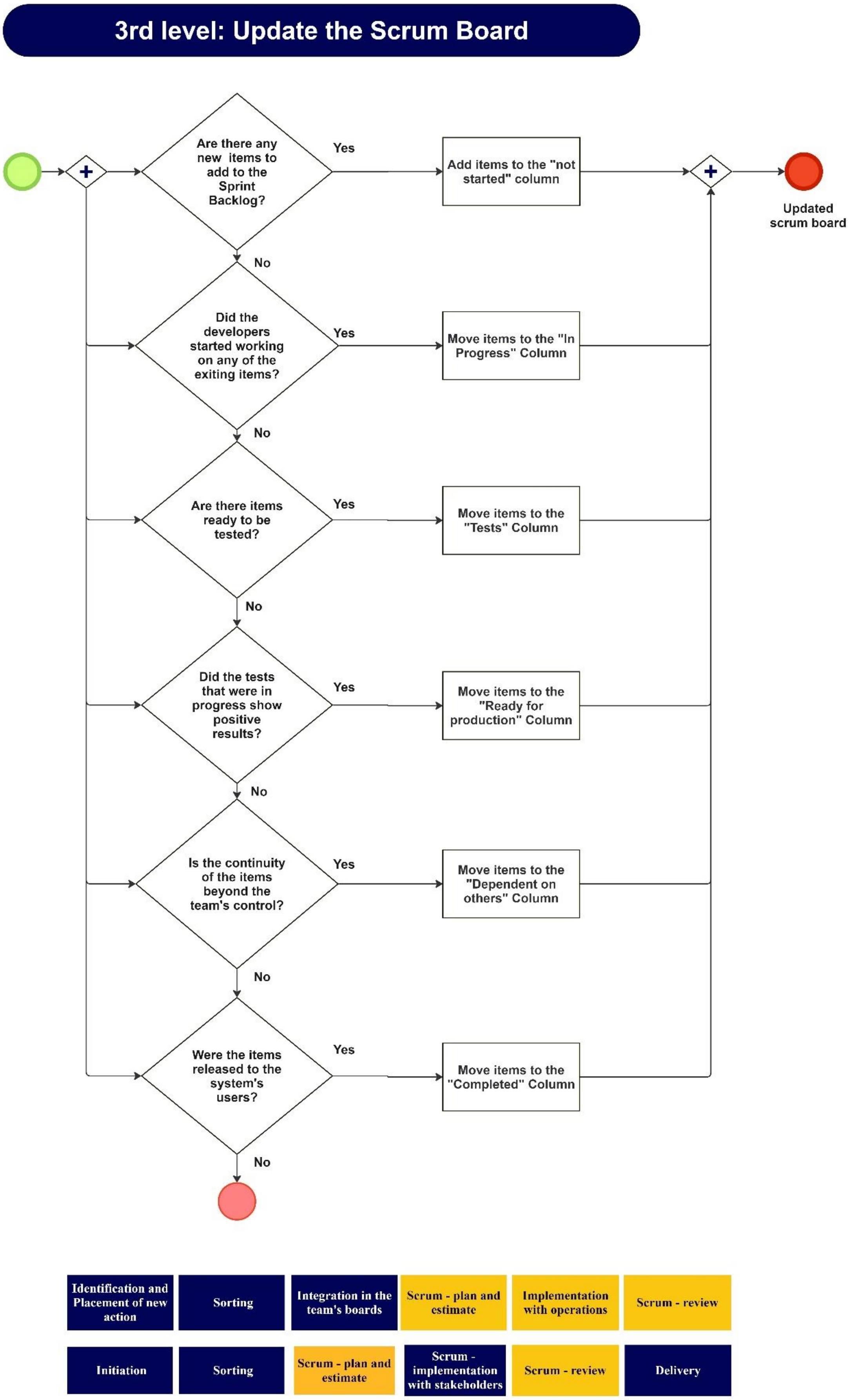

Figure B.11 – TO BE: Flowchart diagram representing the task “Update Scrum Board”, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

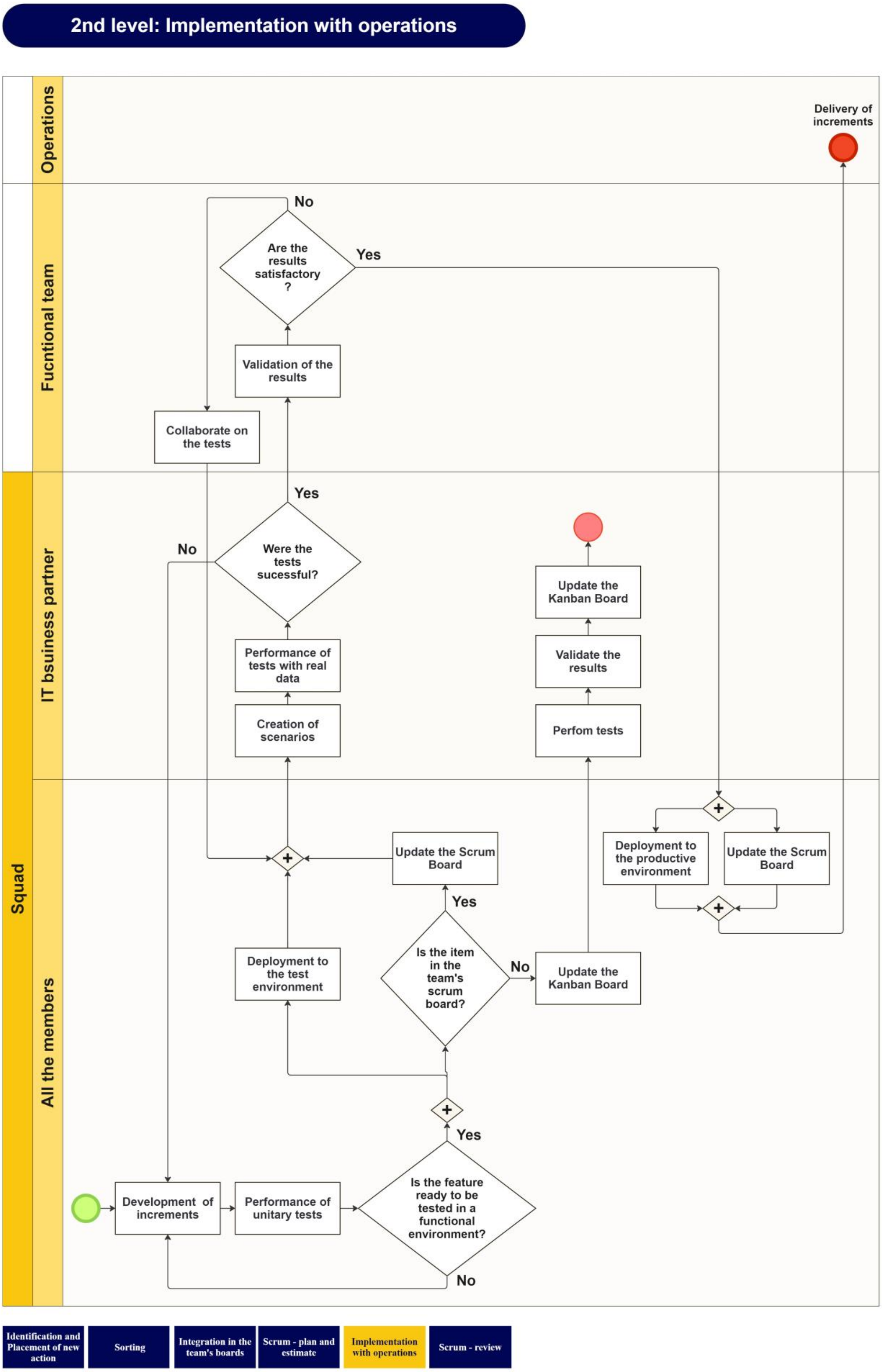

Figure B.12 – TO BE: Swimlane diagram representing the phase “Scrum – implementation with operations”, common to the core process “Incremental improvement” and the support process “Corrective action”

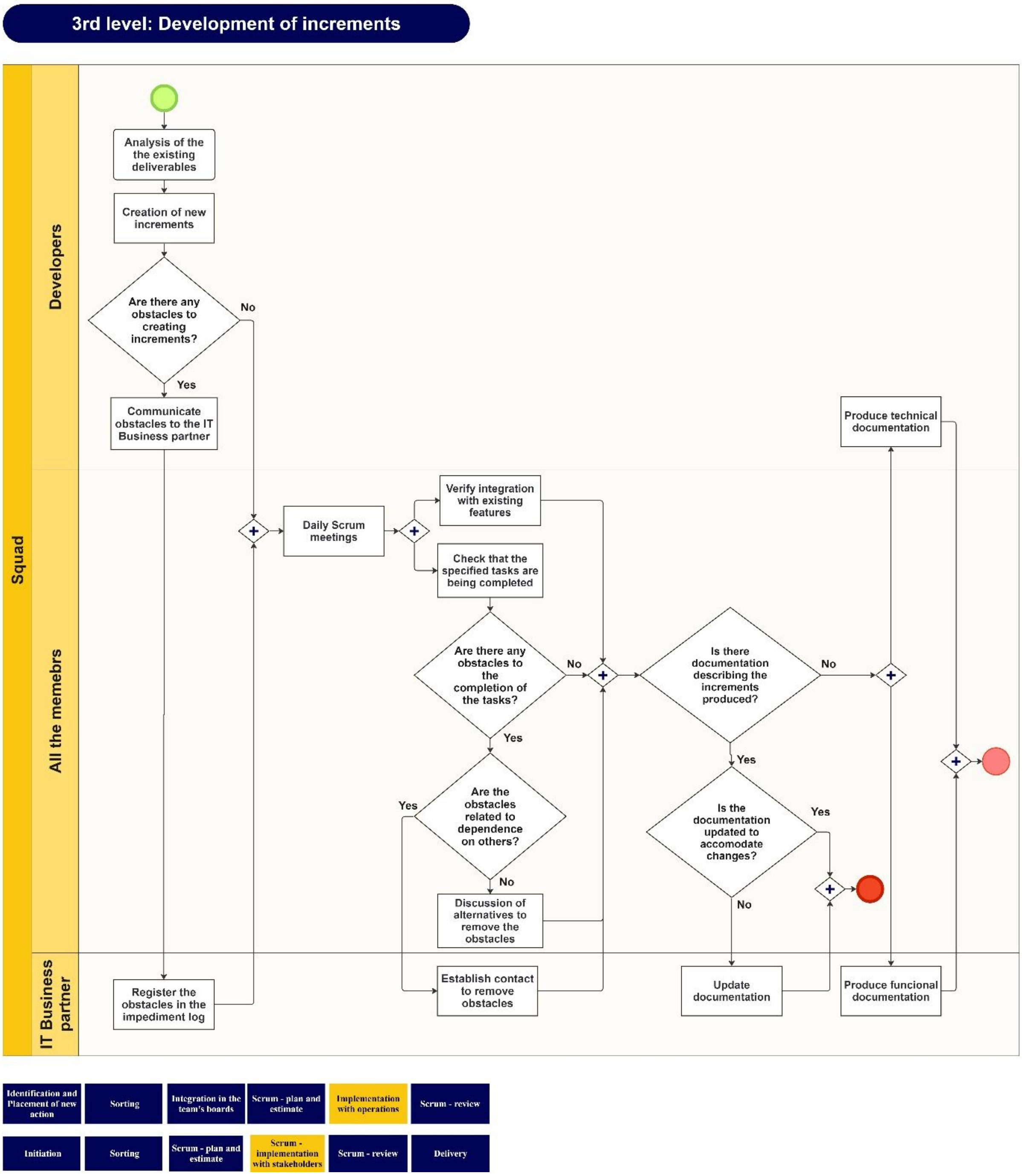

Figure B.13 – TO BE: Flowchart diagram representing the task “Development of increments”, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

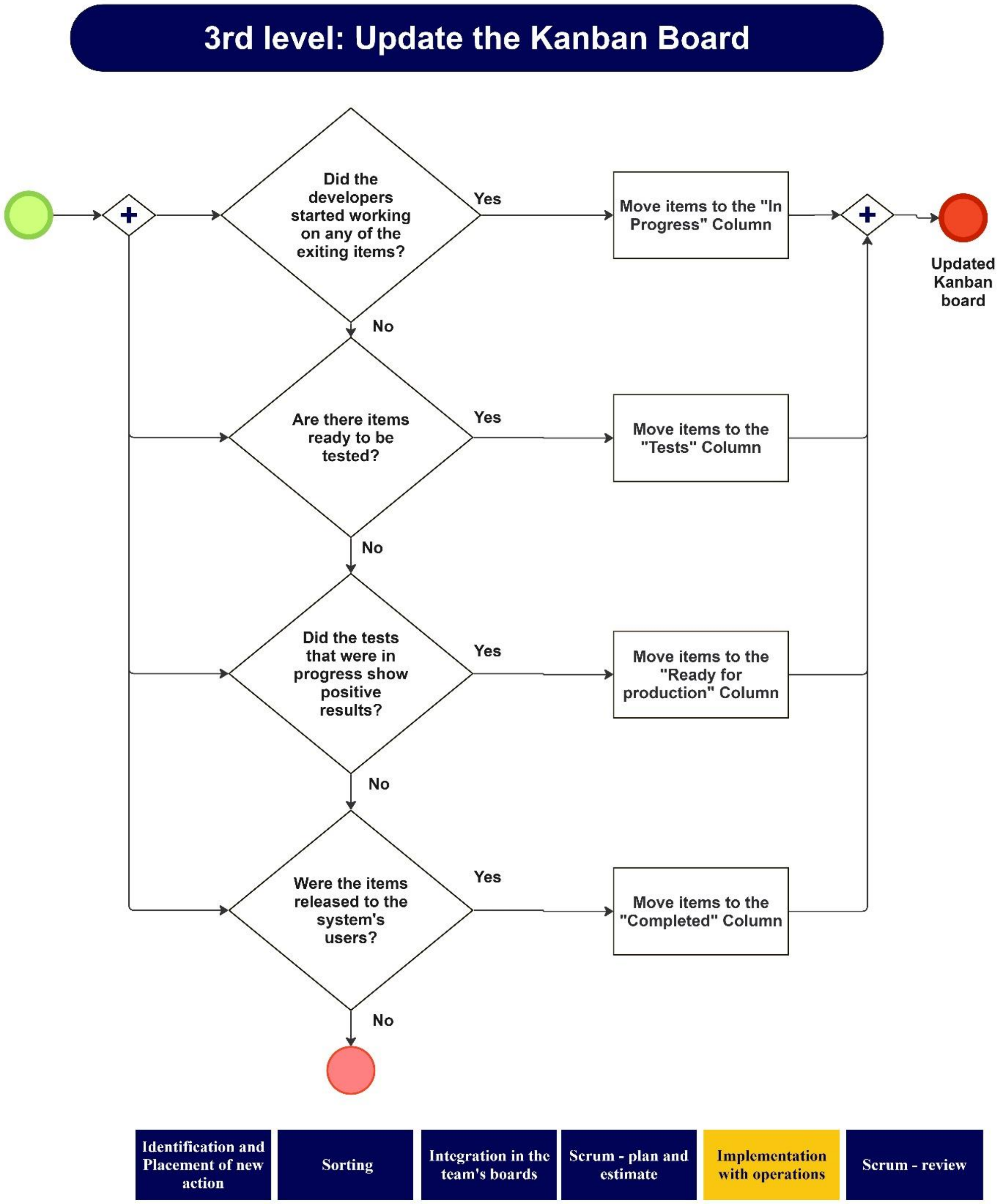

Figure B.14 – TO BE: Flowchart diagram representing the task “Update Kanban Board”, common to the core process “Incremental improvement” and the support process “Corrective action”

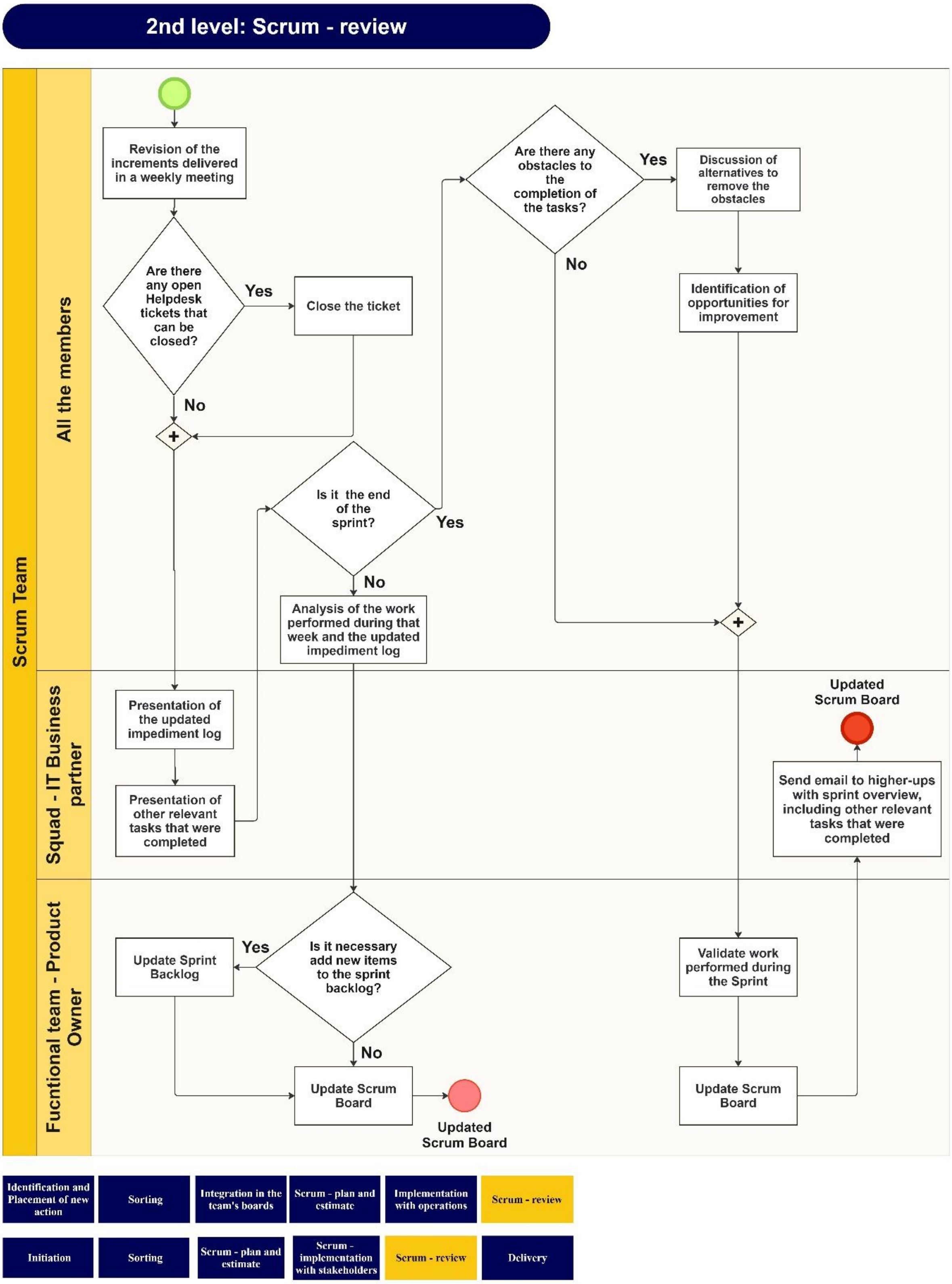

Figure B.15 – TO BE: Swimlane diagram representing the phase “Scrum – review, common to the core processes “Project with partner” and “Incremental improvement” and the support process “Corrective action”

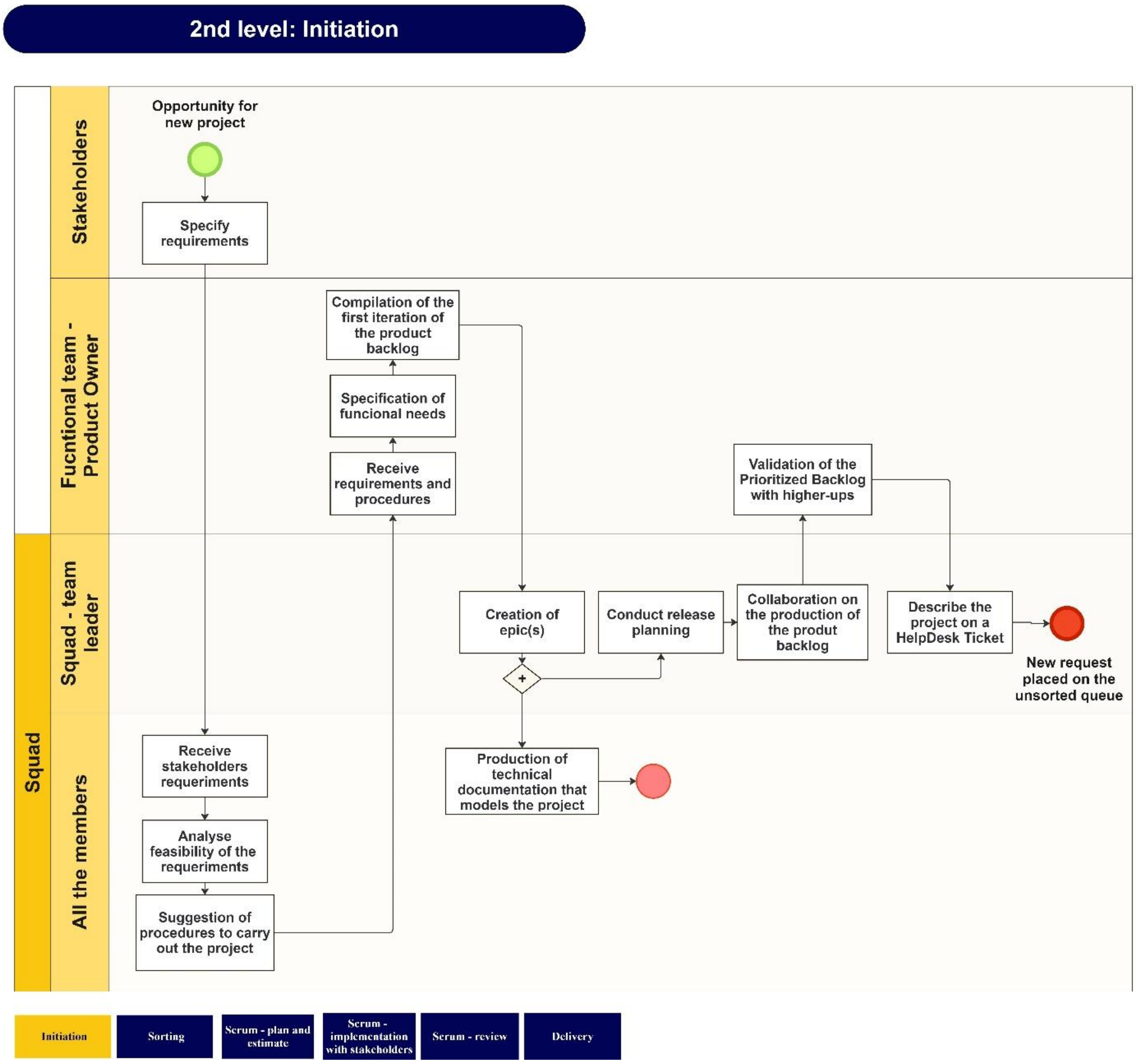

Figure B.16 - TO BE: Swimlane diagram representing the phase “Initiation”, from the process “Project with Partner”

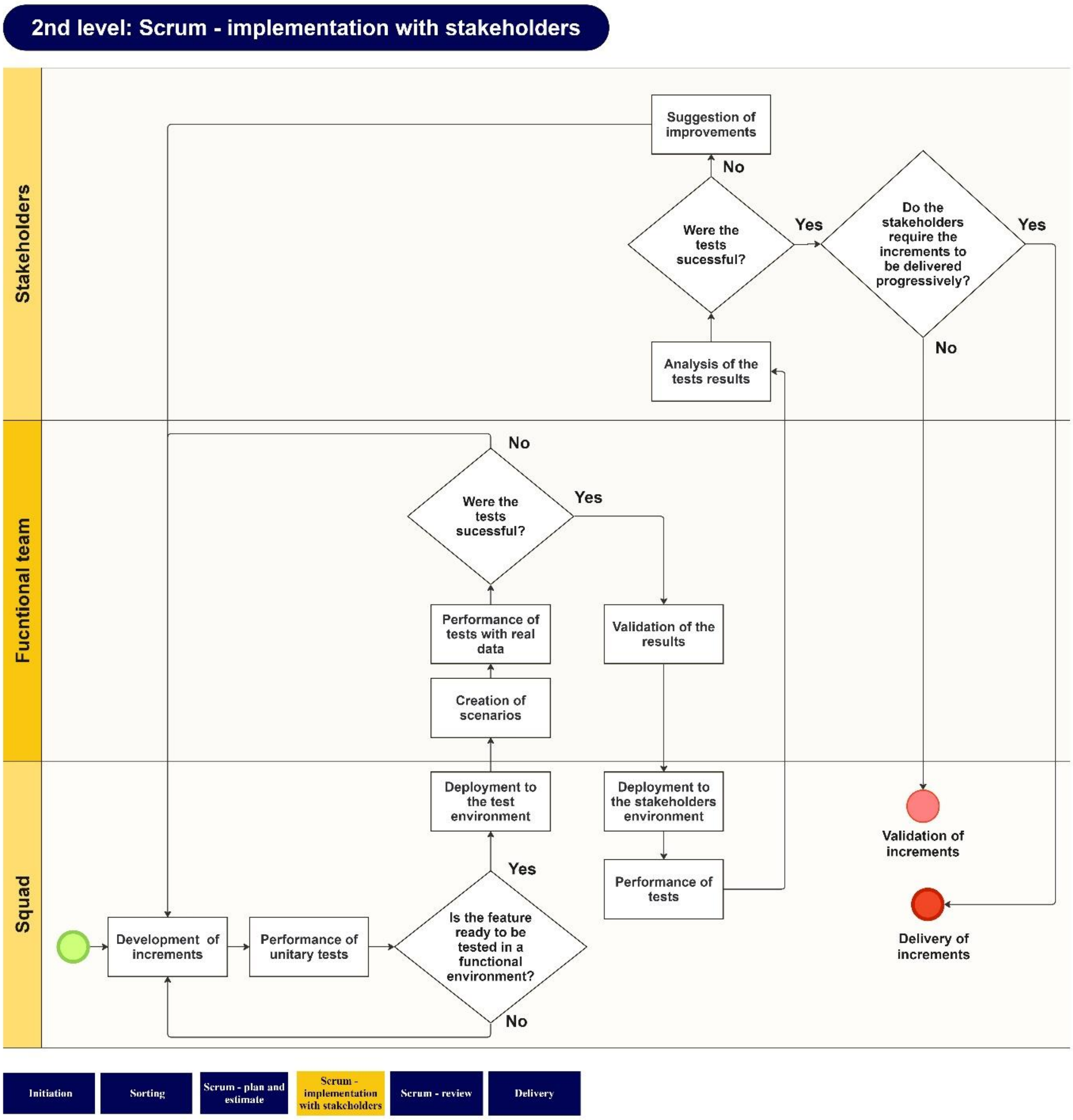

Figure B.17 - TO BE: Swimlane diagram representing the phase “Scrum – implementation with stakeholders”, from the process “Project with Partner”

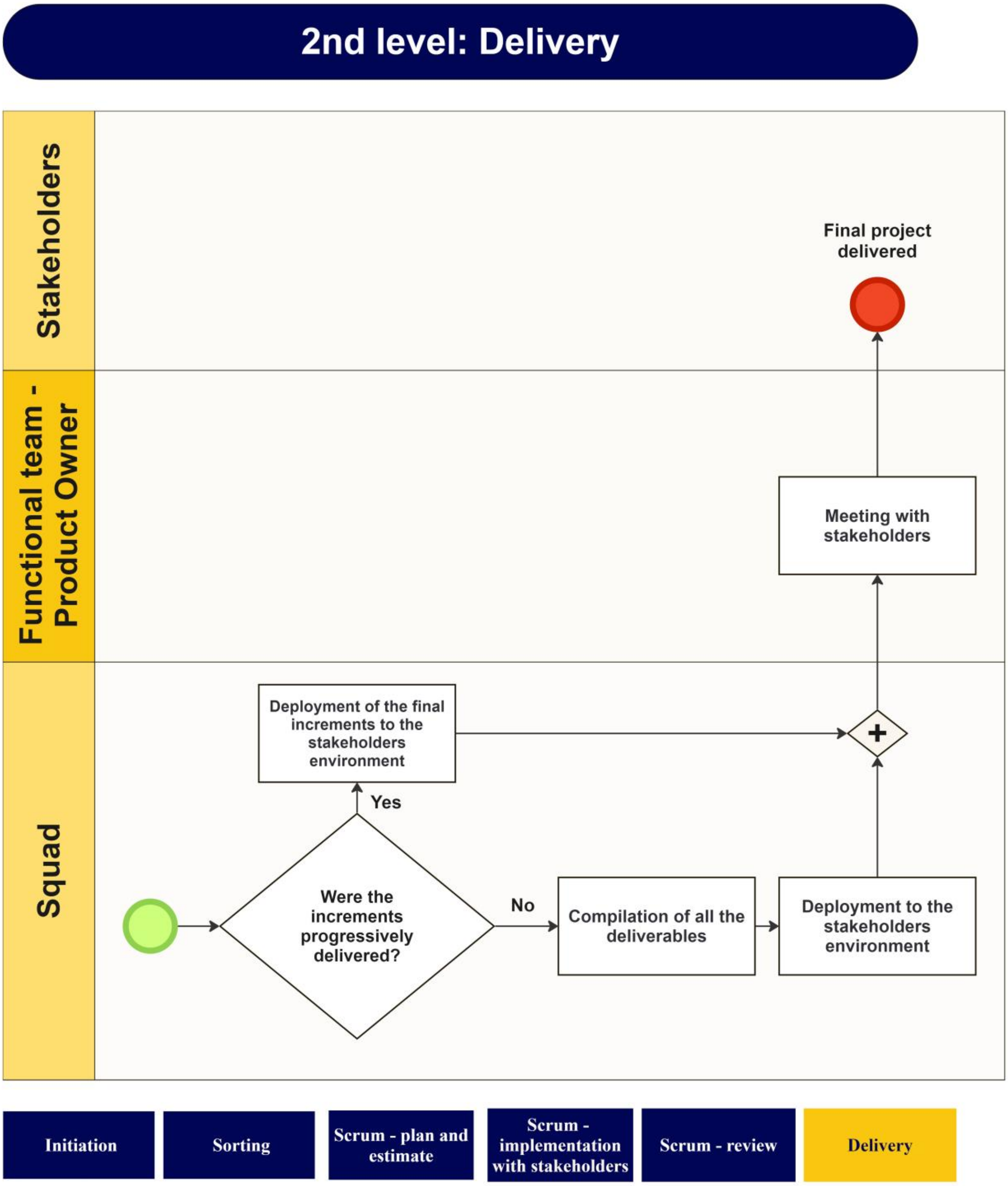

Figure B.18 - TO BE: Swimlane diagram representing the phase “Delivery”, from the process “Project with Partner”

Table B.1 –Leading indicators calculated for the support process “Business support and incident handling” for the 5 weeks of May 2024

CLUSTERS	TOTAL TIME SPENT PER CLUSTER (HOURS)	NUMBER OF TICKETS	PERCENTAGE OF TICKETS PER WEEK	TIME SPENT PER CLUSTER PER WEEK
0	**8.50**	**10**	**23.81%**	**34.34%**
WEEK 1	3.25	3	30.00%	38.24%
WEEK 2	4.42	5	50.00%	51.96%
WEEK 3	0.50	1	10.00%	5.88%
WEEK 4	0.33	1	10.00%	3.92%
1	**2.08**	**5**	**11.90%**	**8.42%**
WEEK 1	1.00	1	20.00%	48.00%
WEEK 2	0.33	2	40.00%	16.00%
WEEK 3	0.50	1	20.00%	24.00%
WEEK 4	0.25	1	20.00%	12.00%
2	**6.67**	**12**	**28.57%**	**26.94%**
WEEK 1	0.17	2	16.67%	2.50%
WEEK 2	4.00	4	33.33%	60.00%
WEEK 3	0.25	1	8.33%	3.75%
WEEK 4	1.50	3	25.00%	22.50%
WEEK 5	0.75	2	16.67%	11.25%
3	**2.00**	**1**	**2.38%**	**8.08%**
WEEK 3	2.00	1	100.00%	100.00%
4	**1.92**	**4**	**9.52%**	**7.74%**
WEEK 2	1.17	1	25.00%	60.87%
WEEK 4	0.25	1	25.00%	13.04%
WEEK 5	0.50	2	50.00%	26.09%
5	**0.50**	**2**	**4.76%**	**2.02%**
WEEK 2	0.25	1	50.00%	50.00%
WEEK 3	0.25	1	50.00%	50.00%
6	**1.83**	**4**	**9.52%**	**7.41%**
WEEK 2	0.33	1	25.00%	18.18%
WEEK 3	0.50	1	25.00%	27.27%
WEEK 5	1.00	2	50.00%	54.55%
7	**1.25**	**4**	**9.52%**	**5.05%**
WEEK 3	0.42	2	50.00%	33.33%
WEEK 5	0.83	2	50.00%	66.67%
GRAND TOTAL	**24.75**	**42**	**100.00%**	**100.00%**

Table B.2 - Lagging indicators calculated for the support process “Business support and incident handling” for May 2024

CLUSTER	TOTAL TIME SPENT PER CLUSTER (HOURS)	AVERAGE TIME SPENT PER CLUSTER (HOURS)	TIME SPENT PER CLUSTER CONSIDERING TOTAL SUPPORT TIME [%]	NUMBER OF TICKETS PER CLUSTER
0	0.5	0.5	2.02%	1
1	7	0.538462	28.28%	13
2	1.916667	0.479167	7.74%	4
3	8.5	0.85	34.34%	10
4	2.083333	0.416667	8.42%	5
5	3	1	12.12%	3
6	0.5	0.25	2.02%	2
7	1.25	0.3125	5.05%	4

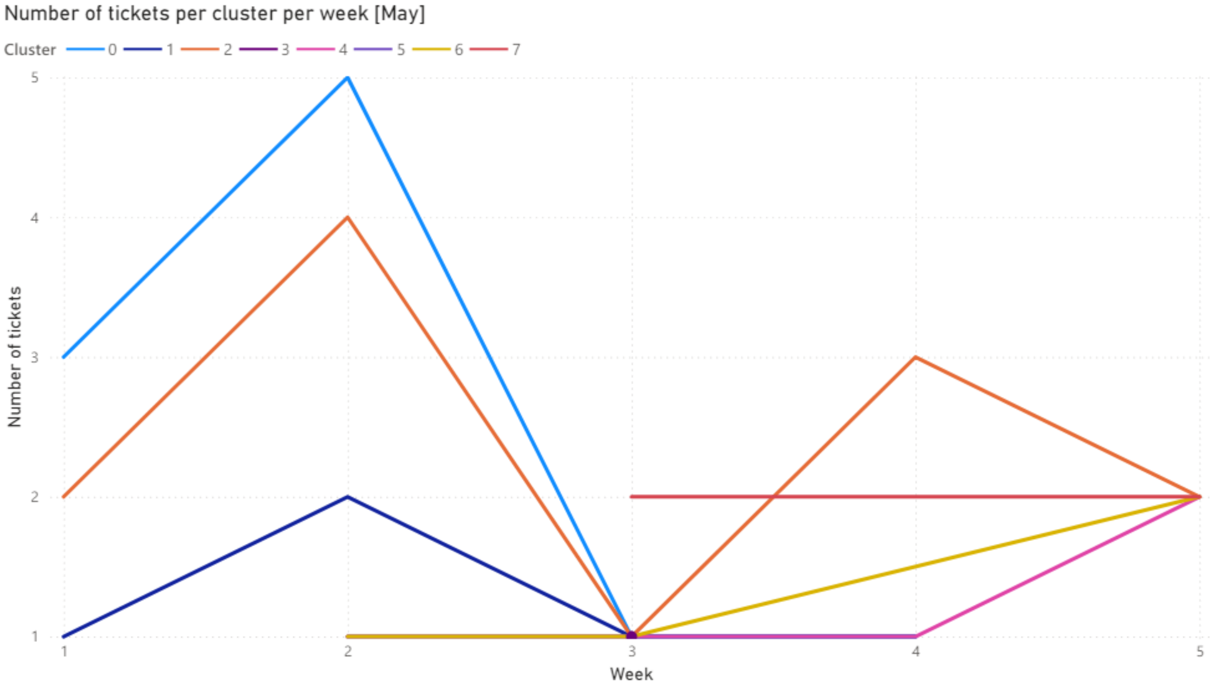

Figure B.19 – Graph for the support process “Business support and incident handling” representing the evolution of the number of tickets per cluster for the weeks of May 2024

Table B.3 - Lagging indicators calculated for the support process “Business support and incident handling” from February to May 2024

CLUSTERS	NUMBER OF TICKETS	TOTAL TIME SPENT PER CLUSTER (HOURS)	PERCENTAGE OF TICKETS PER MONTH	TIME SPENT PER CLUSTER PER MONTH
0	**14**	**5.75**	**8.14%**	**6.30%**
MONTH 2	6	2.67	42.86%	46.38%
MONTH 3	3	1.17	21.43%	20.29%
MONTH 4	4	1.42	28.57%	24.64%
MONTH 5	1	0.50	7.14%	8.70%
1	**38**	**24.68**	**22.09%**	**27.04%**
MONTH 2	7	11.67	18.42%	47.27%
MONTH 3	9	3.58	23.68%	14.52%
MONTH 4	9	2.43	23.68%	9.86%
MONTH 5	13	7.00	34.21%	28.36%
2	**12**	**3.92**	**6.98%**	**4.29%**
MONTH 2	6	1.50	50.00%	38.30%
MONTH 3	2	0.50	16.67%	12.77%
MONTH 5	4	1.92	33.33%	48.94%
3	**29**	**17.33**	**16.86%**	**18.99%**
MONTH 2	6	1.75	20.69%	10.10%
MONTH 3	3	1.42	10.34%	8.17%
MONTH 4	10	5.67	34.48%	32.69%
MONTH 5	10	8.50	34.48%	49.04%
4	**20**	**11.27**	**11.63%**	**12.34%**
MONTH 2	6	3.52	30.00%	31.21%
MONTH 3	4	2.58	20.00%	22.93%
MONTH 4	5	3.08	25.00%	27.37%
MONTH 5	5	2.08	25.00%	18.49%
5	**22**	**14.67**	**12.79%**	**16.07%**
MONTH 2	7	2.17	31.82%	14.77%
MONTH 3	3	3.25	13.64%	22.16%
MONTH 4	9	6.25	40.91%	42.61%
MONTH 5	3	3.00	13.64%	20.45%
6	**16**	**6.33**	**9.30%**	**6.94%**
MONTH 2	6	1.58	37.50%	25.00%
MONTH 3	4	1.83	25.00%	28.95%
MONTH 4	4	2.42	25.00%	38.16%
MONTH 5	2	0.50	12.50%	7.89%
7	**21**	**7.33**	**12.21%**	**8.03%**
MONTH 2	5	1.92	23.81%	26.14%
MONTH 3	4	1.83	19.05%	25.00%
MONTH 4	8	2.33	38.10%	31.82%

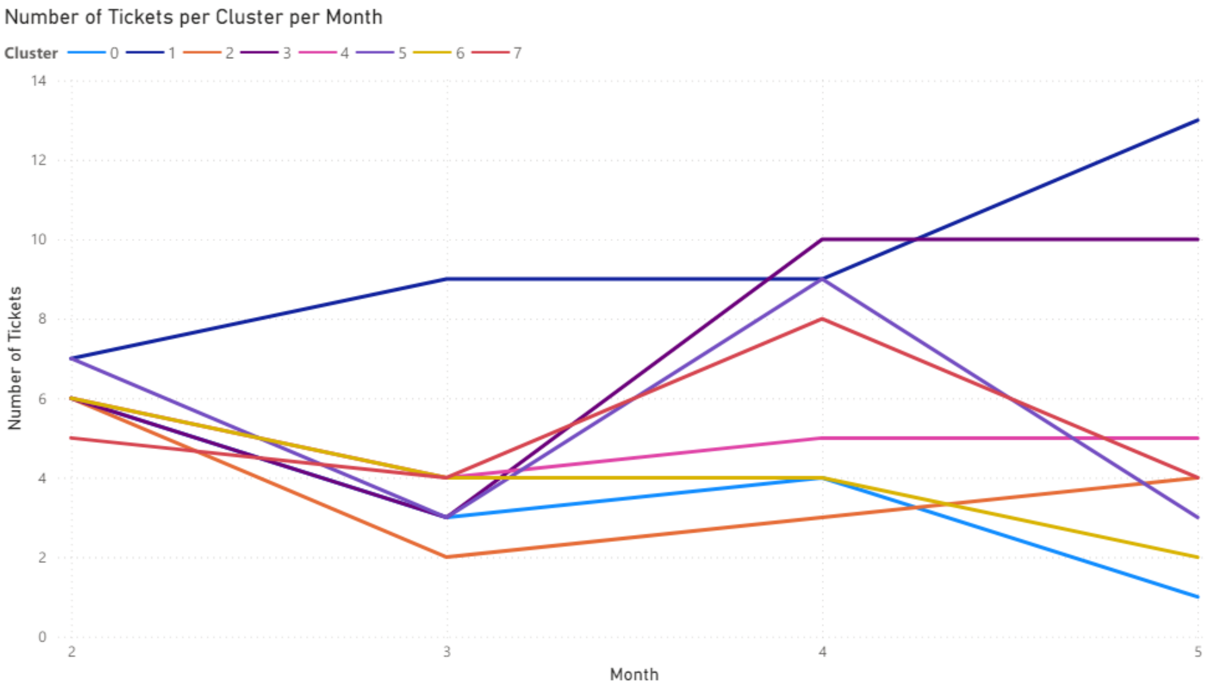

Figure B.20 - Graph for the support process “Business support and incident handling” representing the evolution of the number of tickets per cluster per month from February to May 2024

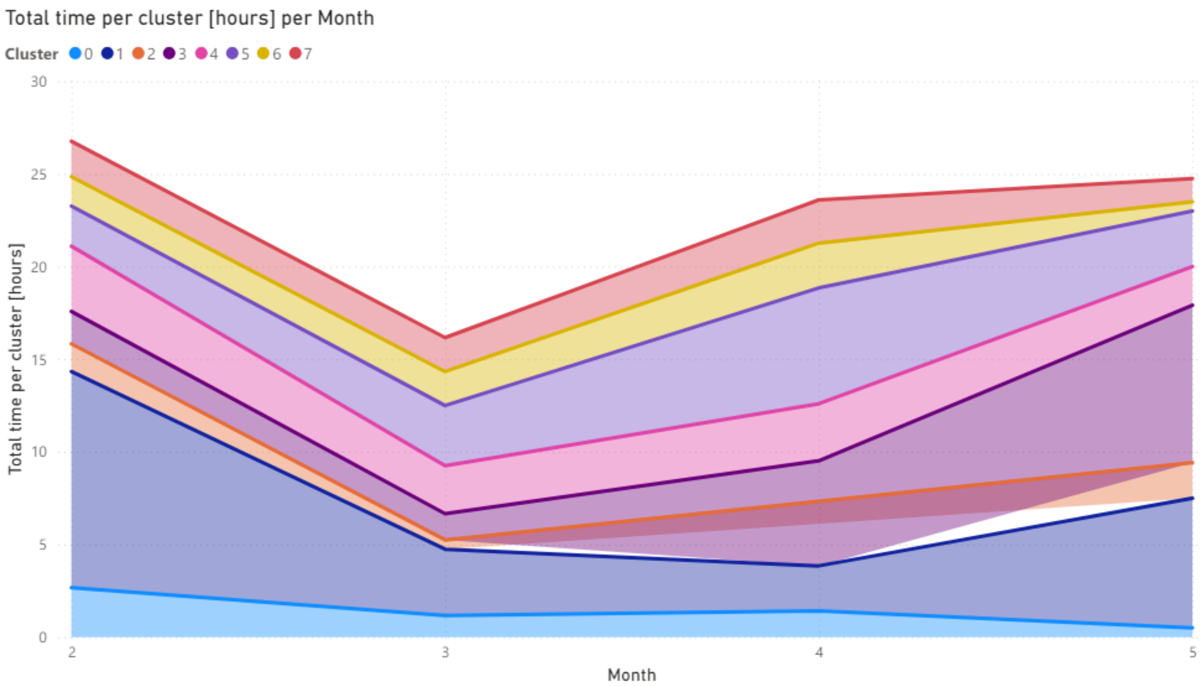

Figure B.21 - Graph for the support process “Business support and incident handling” representing the evolution of the total time spent per cluster per month from February to May 2024