FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

MobiAlert: A Data-driven Embedded System to Alert Cyclists About Critical Urban Zones

José Miguel Araújo Martins Ferreira

Master's in Electrical and Computer Engineering

Supervisor: Daniel Gouveia Costa

July 19, 2024

Resumo

Atualmente, as cidades estão a passar por uma expansão notável, e os sucessivos confinamentos durante a pandemia de COVID-19 forneceram uma perspetiva prática do que as cidades poderiam ser se fossem adotadas abordagens mais sustentáveis. Em geral, a congestão do tráfego e a poluição do ar originadas por veículos com motores a combustão continuam a ser dos fatores mais problemáticos em cidades superlotadas. Neste contexto, o uso de meios de transporte alternativos tem aumentado rapidamente. No entanto, a maioria dos ciclistas ainda enfrenta condições adversas nas cidades modernas, desde ciclovias degradadas e falta de infraestrutura adequada, até acidentes de trânsito e poluição causada pela quantidade excessiva de veículos nas estradas. Muitos estudos têm desenvolvido formas de avaliar condições adversas em ciclovias e trilhos, procurando proporcionar uma melhor experiência e mais segura para os ciclistas. Além disso, cada vez mais se verifica um aumento do número de bases de dados digitais que permitem a visualização de dados geoespaciais, como, por exemplo, o Google Earth e o OpenStreetMap. Portanto, este trabalho propõe um sistema embebido que, combinado com tecnologias *Internet of Things* (IoT), abordagens de mineração de bases de dados *Geographic Information Systems* (GIS) e localização geoespacial em tempo real, pode fornecer alertas rápidos e assertivos aos ciclistas quando eles entram em zonas perigosas. Assim, esta dissertação apresenta uma solução inovadora, com o uso de um sistema embebido de custo reduzido, que tem como objetivo a melhoria da experiência do ciclista durante o seu percurso em zonas urbanas, tendo dados geoespaciais como *input*. Com o uso da ferramenta CityZones e uma Raspberry Pi Zero W, esta solução é acessível, eficiente e facilmente reproduzível. Para testar a veracidade do *hardware*, foram obtidos resultados na cidade do Porto, Portugal, que mostram a implementação prática do sistema e a sua efetividade na procura pela segurança do ciclista e da mobilidade urbana.

Abstract

Nowadays, cities worldwide are undergoing a significant expansion, and the successive lockdowns during the COVID-19 pandemic have provided a practical perspective on what cities could become if greener approaches were adopted. In general, traffic congestion and air pollution caused by combustion-engine vehicles remain some of the most problematic factors in overcrowded cities. In this context, the use of alternative means of transportation has been rapidly increasing. However, most cyclists still face adverse conditions in modern cities, ranging from degraded cycling paths and lack of adequate infrastructure, to traffic accidents and pollution caused by the excessive number of private vehicles on roads. Many studies have been developing ways to assess adverse conditions on cycling paths and trails, seeking to provide a better and safer experience for cyclists. Additionally, there is a growing number of digital databases that allow the visualisation of geospatial data, such as Google Earth and OpenStreetMaps. Therefore, this project proposes an embedded system that, combined with Internet of Things (IoT) technologies, Geographic Information Systems (GIS) database mining approaches, and real-time geospatial location, can provide quick and accurate alerts to cyclists when they enter dangerous zones. This dissertation presents an innovative solution, using a low-cost embedded system aimed at improving the cyclists' experience during their route in urban areas, with geospatial data as input. Using the CityZones tool and a Raspberry Pi Zero W, this solution is affordable, efficient, and easily reproducible. To test the hardware effectiveness, results from the city of Porto, Portugal, demonstrate the practical implementation of the system and its effectiveness in enhancing cyclist safety and urban mobility.

Sustainable Development Goals

Table 1: Sustainable Development Goals (SDG) 11 e 13

ODS	Goal	Contribution	Performance Indicators
ODS 11	To make cities and communities more sustainable, providing access to safe, accessible, and sustainable transportation systems for all. This includes improving road safety and promoting the use of non-polluting modes of transportation, such as bicycles (11.2.)	Increasing road safety and promoting the use of non-polluting transportation, such as bicycles, contributing to more sustainable cities	Reduction in the number of traffic accidents as well as the severity of the accidents (11.2.1.)
ODS 13	Combating climate change and its impacts, including reducing greenhouse gas emissions. Promoting the use of bicycles as a sustainable mode of transportation contributing to climate change mitigation by reducing air pollution and decreasing dependence on fossil fuels (13.2.)	Reducing greenhouse gas emissions, improving air quality, and decreasing dependence on fossil fuels	Reduction of CO2 emissions (13.2.2.)

Agradecimentos

Primeiramente agradeço aos meus pais e irmão, Ângela F., Óscar F. e João Pedro F., por todo o apoio que me deram durante o meu percurso universitário e por nunca terem desistido de mim, mesmo nos momentos mais difíceis. A vossa presença e incentivo foram fundamentais para que eu chegasse até aqui.

De seguida agradeço à minha namorada, Juliana R., por uma motivação inigualável, a tua presença constante foi essencial para que eu mantivesse o foco e a determinação necessários para concluir esta etapa.

Gratidão aos meus amigos da UBI, André L., Conrado G., André P e Jorge B, pela amizade e camaradagem demonstradas ao longo deste percurso.

E por fim, um obrigado ao Professor Doutor Daniel Costa Gouveia, por todo o apoio e por me ter aturado durante estes meses com todas as minhas dúvidas e incertezas, estando sempre disponível para me ajudar a selecionar um caminho para conclusão da dissertação.

Um obrigado a todos, a vossa presença tornou esta jornada mais leve e gratificante.

José Miguel Araújo Martins Ferreira

"A winner is a dreamer who never gives up"

Nelson Mandela

Contents

List of Figures

List of Tables

Abbreviations and Symbols

AI	Artificial Intelligence
ARM	Advanced Risk Machine
CAD	Computer Aided Design
CO_2	Carbon Dioxide
CSV	Comma Separated Values
EPC	Electronic Product Code
GIS	Geographic Information Systems
GPIO	General-Purpose Input/Output
GPS	Global Positioning System
GUI	Graphical User Interface
HDMI	High-Definition Multimedia Interface
ICT	Information and Communication Technology
IOT	Internet of Things
IIOT	Industrial Internet of Things
IP	Internet Protocol
IPSO	Internet Protocol for Smart Objects
IS	Information Systems
IT	Information Technology
ITS	Intelligent Transportation Systems
LED	Light-Emitting Diode
MA	MobiAlert
MZ	Mitigation Zone
NMEA	National Marine Electronics Association
OSM	Open Street Maps
PETG	Polyethylene Terephthalate Glycol
POI	Point of Interest
RFID	Radio Frequency Systems
RPI	Raspberry PI
RT	Real Time
SBC	Single-Board Computer
SDG	Sustainable Development Goals
SLDASM	SOLIDWORKS Assembly File
TCP/IP	Transmission Control Protocol/Internet Protocol
UDP	User Datagram Protocol
UN	United Nations
UV	Ultraviolet
USB	Universal Serial Bus
WWW	World Wide Web
3D	Three Dimensions

Chapter 1

Introduction

Our cities have been experiencing unprecedented growth in the last decades, with about 55% of the worldwide population now residing in urban areas, which is expected to increase to 68% by 2050. Urban areas are becoming overcrowded, and problems such as pollution, traffic jams, and road accidents are becoming common (Sharifi and Salehi [2022]). The future of urban cities and the need for resilient and sustainable cities is a problem that has been gaining prominence among researchers. In 2020, due to COVID-19, the world witnessed practical cases of an alternative and greener future, experiencing a significant pollution reduction, accompanied by a noticeable shift towards sustainable modes of transportation. This period emphasised the importance of smart mobility, with recent studies affirming that this is one of the pillars of smart cities (Bıyık et al. [2021b]).

Despite the need for more sustainable cities highlighted by the COVID-19 pandemic, there remains a strong inclination towards the use of non-sustainable motor vehicles. This tendency is leading cities to high congestion rates and a substantial carbon footprint from internal combustion engine vehicles. In addition, the world is dealing with the urgent issue of climate change, which is becoming more challenging in urban areas. According to the World Bank, 13% of global carbon dioxide (CO_2) emissions come from transportation (Mitieka et al. [2023]).

To handle the problems caused by urbanization, one simple solution that has been gaining prominence among city inhabitants is cycling. Promoting the use of bicycles or e-bikes as the main mode of transportation is a step toward making cities more sustainable and resilient. However, this trend also brings some challenges in multiple areas. Among them, the lack of adequate cycling infrastructures has been perceived in most cities, bringing negative consequences of the promotion of bicycles as an effective transportation modal. Entities like the United Nations (UN) have supported the development of better cycling routes to address environmental concerns, emphasizing that embracing cycling can lead to significant reductions in CO_2 emissions (Oliveira et al. [2021a]).

To address the cycling challenges, the concept of smart cycling has gained prominence among cities. Furthermore, the evolution of the IoT concept has revolutionized conventional perceptions of bicycles, enabling researchers to allocate embedded systems and sensors on them. This advancement facilitates the assessment of cycling infrastructures by selecting relevant factors (Costa and Duran-Faundez [2018]). On this basis, by integrating Global Positioning Systems (GPS) and data mining algorithms within embedded systems, bicycles can obtain a new resource to support the formulation of road evaluation strategies. Doing so, it is possible to alert cyclists about urban zones that pose potential risks to their health and safety (Mitieka et al. [2023]).

This dissertation proposes the development of the MobiAlert system, an innovative solution that uses the IoT and geospatial data to enhance cyclist safety. MobiAlert integrates sensors, real-time communication, and data analysis. With the use of the CityZones tool, it is possible to alert the cyclist about the risk of the zone where he is crossing (Peixoto et al. [2023]). Through data mining algorithms and relevant hardware, the cyclist can be informed in advance to make him more conscious of the risk and gravity of an accident in the specific zone he is cycling.

Therefore, the main objective of this project is the design, development, and evaluation of a functional prototype of the MobiAlert system. It aims to provide accurate and near real-time information about adverse conditions that may pose a risk to cyclists. This dissertation addresses the problem structure, a review of the state of the art, the system development methodology, and the analysis of the results obtained through practical tests and case studies.

The innovation and relevance of this project lies in the application of emerging technologies to solve a practical and urgent problem in modern cities. MobiAlert not only contributes to cyclist safety, but also promotes the use of bicycles as a viable and safe alternative to traditional transportation. It is also aligned with sustainable development goals and the promotion of smart cities.

1.1 Research goals and contributions

The main aim of this research is to create MobiAlert, an embedded system designed to improve the cycling experience. To accomplish this, the specific objectives of this master thesis are as follows:

- Multi-domain data exploitation for smart cycling;
- Data formatting for large irregular maps;
- Implementation of a low-cost system for cycling alerts;
- Validation of the proposed solution and prototype with real-life case scenarios in the city of Porto;
- Development of scientific papers to validate the proposed MobiAlert unit.

1.2 Structure

This report is structured into six chapters, facilitating a clearer explanation of the phases and concepts encompassed within the dissertation.

Chapter 1, provides a contextualization of the problem's context, the dissertation's structure, and its primary objectives.

Chapter 2 outlines the theoretical framework, aligning it with the research area and addressing the core issues. Additionally, it provides a review of the state-of-the-art in cycling technologies and big geospatial data processing.

Chapter 3 details all the layers of the MobiAlert systems and subsystems developed for this thesis.

Chapter 4 explores the 3D modelling of the MobiAlert cover and support, designed to be attached to the bicycle.

Chapter 5 presents the results obtained from the testing phase of the MobiAlert unit, including three real test cases conducted in the city of Porto, Portugal.

Chapter 6 provides a brief discussion on the findings and developments achieved throughout the research, along with the anticipated future steps for the project.

Chapter 2

Background and literature review

This chapter investigates the foundational concepts that support the development of the dissertation. Specifically, it explores the theoretical framework of MobiAlert, focusing on domains such as smart cities, IoT, and data driven cities. Additionally, it examines the different aspects of Data-Driven cities and conducts a comparative analysis of embedded platforms, including various models of the Raspberry Pi, within the context of IoT applications.

2.1 Cities and Mobility

Since 2018, about 55% of the world's urban population has been concentrated in urban areas and this number is anticipated to go up to almost 68% by 2050 (United Nations and Affairs [2019]). This rapid urbanization is delaying the Sustainable Development Goals (SDGs) of 2030, posing great difficulty in developing sustainable cities.

The drift of people migrating into the urban areas is bringing a series of problems that include, but are not limited to, environmental degradation, social inequality, crime, pollution, and traffic congestion. This delays the development of sustainable cities which are often located near the areas of rapid urban development. Furthermore, the urban plans are often incompatible with the urbanization speed, hence making the area less sustainable (Sharifi and Salehi [2022]).

In big cities, the main mode of transportation is the car, and the high number of private vehicles in the streets highlights the problem of traffic accidents, due to rapid growth in mobility and the number of people driving. In the context of personal vehicles being the most preferable mode of transportation, a great number of global entities have been increasingly worried about sustainability and the surplus of issues that personal vehicles causes for the urban centers (Oliveira et al. [2021b]).

In response to problems brought by these urbanization, the concepts of smart and sustainable cities have gained traction, especially since the COVID-19 pandemic. These types of cities are still underdevelopment and their main goal is to reduce the dependence on private vehicles and promote sustainable mobility. This can be achieved through efficient public transportation, improved infrastructure, electric cars, vehicle sharing, IoT technologies, and intelligent urban planning (Paes et al. [2023]).

2.1.1 Cities and Sustainability

The idea behind sustainability is to develop a city that can sustain the current population, while not compromising the rights of future generations. Greener sustainability, social justice, and economic development should be the main focus of sustainable cities. Moreover, the decreased usage of fossil fuels, the reduction in waste and pollution should also be a priority. Taking into account the planning and management of urban areas, three fronts of sustainability can be achieved: the social, the economic and the environmental front. (Sadowski [2023], Bayulken and Huisingh [2015]).

Moreover, cities that succeed make sure that everyone has a fair chance to live a dignified life, while preserving the nature (Sadowski [2023], Bayulken and Huisingh [2015]). As mentioned before and as defined by the UN Sustainable Development Goals, sustainability should be developed in three fields: green technologies, social parity, and economic opportunities (Stokstad [2015]).

Sustainable cities were built to minimize CO_2 emissions and to advance towards healthier eco-spaces. According to Li [2022], the top five issues to be addressed will be smart economies, sustainable lifestyles, educated citizens, efficient transportation, and a responsible government by the newest environment designs. However, cities that have been applying these principles are undergoing through some challenges, like the absence of a roadmap for a sustainable city or attempts at implementing such projects. It is important to point out that while major technological advancements have come, they caused externalities that still have to be addressed and solved (Williams [2009]).

2.1.2 Integrated Transport Systems

Multimodal transportation systems or Integrated Transport Systems (ITS) include various kinds of transport such as air, land, rail, and maritime (Babić et al. [2022], A. Tavasszy et al. [2015]). This concept is environmentally friendly, as it sustains the employment of public transport and non-motorized transportation, like cycling and walking. This will make public transport more attractive to the public and it will decrease car dependency (Babić et al. [2022], A. Tavasszy et al. [2015]).

ITS is a possible solution to successful city planning, as it does not only facilitate cooperation, but also makes the movement of vehicles more efficient. This translates as a benefit to both the transportation service providers and the population that gets more convenient and reliable transportation services. As a principle, ITS has the most advantages in suburban and regional places where there is a wide array of transport modes that are interconnected (Babić et al. [2022]).

Moreover, ITS contributes to economic development, since it ensures that labour markets and employment are not restricted. However, the latter is a prerequisite for ITS sustainability and must be given through the construction of infrastructure (Babić et al. [2022], A. Tavasszy et al. [2015]).

2.1.3 Challenges in Urban Cycling

The use of bicycles in urban areas has increased over the years, but motor vehicles are still the main mode of transport in large cities. Additionally, the infrastructure is inadequate, and accessibility to bicycles is limited. In this way, cyclists still lack road safety and face difficulties during their commutes (Naim and Felix [2022]).

A report from the "European Road Safety Observatory" shows worrying data regarding cycling, where it is possible to see that since 2010, all modes of transport have had a reduction in road deaths, except for bicycles, which have maintained an annual rate of 2035 deaths (Observatory [2021]).

Therefore, it is essential to enhance the safety of cyclists on the roads and also motivate the use of bicycles. A bicycle, unlike fossil fuel powered vehicles, is one of the greenest and most sustainable modes of transportation worldwide (Oliveira et al. [2021a]). Additionally, the advantages of this transport method are vast, ranging from the mitigation of traffic congestion caused by car traffic, to the minimization of gases produced by them, while enhancing cyclists' health. Furthermore, riding a bicycle produces 0% gas emissions, being a favorable choice for non-polluting transportation (Kapousizis et al. [2023]).

2.2 Internet of Things

The IoT history dates back to the 19th century and has evolved to today's Information and Communication Technology (ICT). The process began with the introduction of the telephone in the early 1800s, which later allowed for the existence of the communication network. The manufacture of computers in the 1920s developed the processing power required for IoT devices. Thus, this laid the foundation for the modern-day IoT technologies. A big advancement was made in 1991, with the remarkable fast development of networking technologies, led by the World Wide Web (WWW) (Ande et al. [2020]). The latter changed worldwide communication and kick-started the growth of connected digital systems.

In the decade of the 2000s and the beginning of the 2010s, there was a significant development in the field of IoT devices collaboration. This happened due to the industrial progress produced by intelligent devices, smartphones, cloud computing, and sensors. In 2014, there was a significant growth in the IoT sector, with the entrance of smart home products by key players in the industry, such as Google and Apple. This growth in the usage of connected devices, contributed to a change in the daily life of citizens. Simultaneously, Industrial IoT (IIoT) brought into the picture the reshaping of the manufacturing and industrial sectors, by improving processes, and enabling predictive maintenance Ande et al. [2020].

Looking forward, IoT is expected to evolve with a focus on applying human-centric principles, emphasizing user experience and ethical considerations. However, this growth also brings concerns around cybersecurity and data privacy, encouraging efforts to develop robust security frameworks and regulations to safeguard IoT ecosystems (Ande et al. [2020]).

2.2.1 Internet of Things definition

The IoT is a paradigm of ICT that creates a global system, linking different objects around the world, using common standards. According to the definition of Atzori et al. [2010], IoT is a network of devices that collect data from various sources and feed it to the cloud where it is further analyzed and shared among the network devices.

According to the author of Canton [2011], the IoT definition is divided into three main perspectives: "Things oriented vision" , "Internet oriented vision" and "Semantic oriented vision".

The "Things oriented vision" perspective is such that objects are being monitored through the extensive application of sensors and radio-frequency identification (RFID) technologies. The identification of each item is done through an Electronic Product Code (EPC). Thus, these networks are the basis for the further development of RFID, sensor arrays, computational devices, and global connectivity, being key success factors of the whole framework (Mehta et al. [2018]). This particular viewpoint, serves as a pillar for smart city development (Peixoto and Costa [2017]).

The "Internet oriented vision" perspective looks into a network-centric approach, with the Internet Protocol for Smart Objects (IPSO) developed in 2008. The protocol capitalized the possibility to link up different devices thoroughly (Mehta et al. [2018]).

The "Semantic oriented vision" is the concept of increasing the understanding and utilization of the data within the IoT realm. In turn, this concept addresses data representation, storage, connectivity, searchability, and organization issues that are related to big datasets. The use of these datasets in the semantics approach brings forth specially designed models and strategies whose goal is to bring the benefit of these methods to the consumer (Mehta et al. [2018]).

Therefore, in the realm of IoT hardware, sensors and actuators devices will join forces, making this a complex process. However, this can be facilitated through the identification and tracking technologies, using both wired and wireless sensors, advanced communication protocols, and intelligent processing techniques. The IoT has a notable impact on daily lives of citizens, which can be noticed in the fields of automation, manufacturing, logistics, and business, as well as the creation of intelligent transportation systems (Atzori et al. [2010]).

Some of the technologies behind IoT are RFID, sensors, and wireless devices. They aim to change the current systems into intelligent frameworks. This feature is what facilitates the connection of millions of physical objects into distributed networks, retrieving actionable insights from raw data.

2.2.2 Architecture

Concerning society, the Internet of Things remains as an unknown paradigm where there are issues and questions to solve. It can have a remarkable effect on how people live through IoT systems (e.g. exercising applications, smartwatches that measure heart rate). With these systems, it is possible to create much more convenient scenarios with smarter and interconnected devices.

Most scientists and developers use the three-layer definition as a basic architecture. This one is structured into a perception layer, a network layer, and an application layer, illustrated in the

Figure 2.1. This one is a type of client-server architecture, where an application is divided into three individual areas that are still connected. Each one of them performs a specific purpose and has been designed to be as independent as possible from the others (Zhong et al. [2017]).

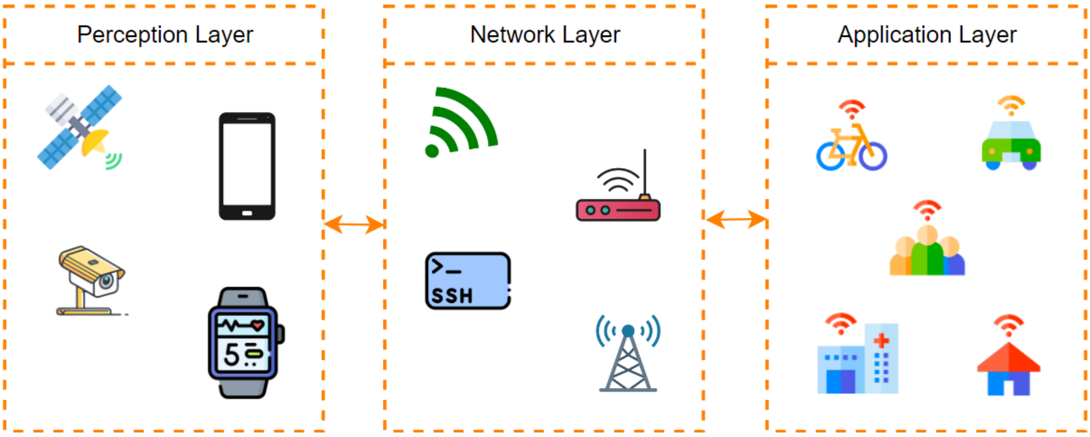

Figure 2.1: IoT three-layer architecture.

The physical perception layer is about sensing the environment signs and capturing the data. The system hardware is equipped with several sensors, actuators, and other devices that can be used to gather data in different aspects, such as temperature, humidity, and movement. This layer has place in the IoT network where the physical state is connected with the digital world (Zhong et al. [2017]).

The network layer promotes communication among devices by making sure that the data carrier is passed from one device to another, regulating the transportation of data packets among devices, and guaranteeing that the data it is reliable. One of the necessary tasks this layer performs is the utilization of protocols such as Transmission Control Protocol/Internet Protocol (TCP/IP), User Datagram Protocol (UDP), and routing protocol (Zhong et al. [2017]).

In the application layer, it is possible to find raw data and strategies to handle the vast amount of data. It is made up of databases, operating systems, and any data storage means. This layer is responsible for data validation, storage, retrieval, and manipulation based on the application's requirements (Zhong et al. [2017]).

2.2.3 Embedded Platforms

One of the most important problems that are part of the planning of smart cities is the choice of devices that fit the complex and large scale of urban systems. The implementation of efficient solutions that allow people and the environment to live a sustainable life take into account the integration and performance of embedded systems. The latter is the reason why the buying process of embedded hardware should be based on factors, such as: cost, computing power, programming flexibility, input and output interfaces, and power consumption (Costa and Duran-Faundez [2018]).

IoT is getting more and more popular with what it seems to be not only the development, but also the need for embedded platform hardware in smart cities. Along with this, the role of these components in the IoT infrastructure is analyzed concerning the embedded system design trade-offs. The platforms of "embedded platforms hardware" type, like Raspberry PI and Arduino, have several advantages. These include compatibility with any type of programming language, great computational power, small dimensions, and low prices. Not only are they used to do programming conveniently, but can also instantaneously turn into a hardware device. These devices are programmable and are used for many different applications, including ecological and urban monitoring (Mathe et al. [2024], Costa and Duran-Faundez [2018]).

These embedded systems have a considerable processing power and storage, allow for communication with other devices, and can be programmable with auto tasks, giving them a solid basis for the development of smart cities. Also, the openness of the hardware can lead to more collaboration and innovation, a must for the fast pace of IoT technologies (Costa and Duran-Faundez [2018]).

Among the existing open source boards, the Raspberry PI is a widely used single-board computer (SBC) that can be used for the IoT, being indispensable to the smart cities development trend in the future (Costa and Duran-Faundez [2018]). This device has a compact size, a powerful processing unit with a high degree of efficiency and minimal power consumption (Karthikeyan et al. [2023]).

The number of applications that are data acquisition related is increasing by the day, some of them being home automation, ITS, and healthcare. These are examples that have proven the utility of the Raspberry PI with wireless connection to be a practical concept for data acquisition applications (Mathe et al. [2024], Karthikeyan et al. [2023]).

Among the models of the Raspberry PI, some are described in the table 2.1.

Table 2.1: Raspberry PI comparison (Foundation [2006]).

Model	PI 3B	PI Zero W	PI 4B	PI Zero W 2
Price (€)	39.87	15.99	49.99	17.99
Processor	Cortex-A53	ARM11	Cortex-A72	Cortex-A53
Clock	1.2GHz	1GHz	1.8GHz Quad	1GHz
RAM	1GB	512MB	1,2,4,8GB	512MB
Ports	4xUSB	1xMicro-USB	4xUSB	1xMicro-USB
Wi-Fi	Yes	Yes	Yes	Yes
Bluetooth	Yes	Yes	Yes	Yes
Memory	Micro SD	Micro SD	Micro SD	Micro SD
GPIO	40	40	40	40

Raspberry PI 3B has a high processing power, meaning it is suitable for more demanding applications. This PI family has their own wireless connection and Bluetooth functions, which are the main drivers for projects where performance and price balance matter (Mathe et al. [2024]).

However, because of its size, it is not recommendable for projects where the availability of space is constrained.

Raspberry PI 4B, apart from the Raspberry PI 5, is the most powerful device in the Raspberry PI series. It is the chosen device if there is a requirement to perform complex projects, such as, creating a web server, building a desktop or designing computational vision related projects. Nevertheless, this PI family has many heating issues, as informed by the Raspberry PI foundation (Kondaveeti et al. [2021], Mathe et al. [2024]).

The Raspberry PI Zero W is the smallest and cheapest of them, meant for projects that have lack of funds or limited space. On the one hand, it has a mini High-Definition Multimedia Interface (HDMI) port that allows the users to see the Graphical User Interface (GUI) of the Raspberry PI. Furthermore, it also features built-in Wi-Fi to enable wireless communication within projects. The device acts has a miniature computer, and it is suitable for embedded projects, intelligent appliances, and robotics (Mathe et al. [2024]). This device is suitable for students to have their first contact with SBC and learning embedded programming languages.

Due to its wide range of sensor implementation through General Purpose Input/Output (GPIO), inexpensive hardware, and constant product updates, this SBC comes as an ideal choice for hardware control for IoT prototypes (Kondaveeti et al. [2021]). When compared to other prototyping boards, such as Arduino, it has a more powerful processing power and storage, allowing for network connections (Kondaveeti et al. [2021]).

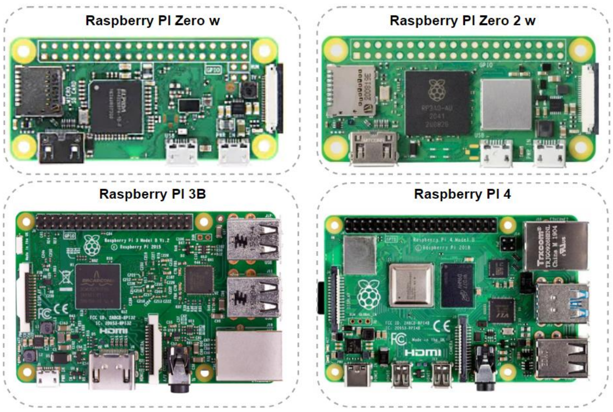

Figure 2.2: Raspberry PI models.

2.3 Smart Cities

Among the review of literature, there is no consensual definition of what a smart city is, as it is a concept that changes depending on the city needs and characteristics. However, according to Mohanty [2016], a general approach to define this term is a city where networks and services work together to obtain a more sustainable environment, through connection with digital Information

Technology and Communications. The main goal of these cities is to improve the quality of life for citizens, increase the efficiency of transportation, reduce energy usage and improve the public services. Furthermore, it has initiated healthcare use of technology to provide health services to the people (Mohanty et al. [2016]).

To better clarify what a smart city is, the table 2.2 provides a review of the literature on its definition, with several authors describing their perspective.

Table 2.2: Smart City definitions

Canton	2011	A smart city is an intelligently designed urban system that utilizes technologies to address challenges. By integrating innovations, it aims to create sustainability for its inhabitants.
Bibri and Krogstie	2017	A smart city integrates advanced ICT with various physical, infrastructural, operational, and ecological systems, across various scales and urban planning methodologies.
Ismagilova et al.	2019	Smart cities employ an Information Systems (IS) centric approach, utilizing ICT within an interactive infrastructure.
Kwak and Lee	2023	A smart city is defined as one where ICT is merged with traditional infrastructures and coordinated using new digital technologies.
Batra and Chhabra	2023	A smart city is defined as a city that adopts ICT to enhance well-being, urban services, and the environment, while meeting the social, environmental, and economic needs.

The main reason why smart cities are essential is the expected increase in the global population and the pressure inflicted on resources and the environment. This cities can alleviate the recurring complications of urbanization and urban population growth (Mohanty [2016], Mohanty et al. [2016]).

However, smart cities are confronted by complications that involve aspects of the technical, financial, legal, social, and cultural kind. Among these complications, the following can be listed: the lack of standardization and interoperability; the high implementation costs; the legal framework for data collection and sharing; the citizen participation and technology acceptance (Gracias et al. [2023]).

To handle these problems, cities should take steps in the main three areas: Policy and Governance, Partnerships and Collaboration, and Technology and Infrastructure. The switch to intelligent municipalities can be made by adopting a solid governance framework that supports citizen participation, provides public-private partnerships, secures digital infrastructure and new technologies (Gracias et al. [2023]).

Smart cities can provide many advantages to these issues, ranging from increasing the quality of life for residents, to promoting economic growth, and to addressing the environmental issues. Additionally, smart cities can benefit on a large scale in various ways through the implementation of IoT (Mahobia and Pawar [2024], Syed et al. [2021]):

- IoT promotes the integration of different technologies, allowing interaction between them;
- IoT facilitates the development of smart city systems, enhancing services such as transportation, energy management, healthcare, and more;
- IoT provides real data from sensors, allowing cities to make informed decisions;
- IoT facilitates reducing several challenges, such as: traffic management, pollution and resource management.

Overall, the application of IoT technologies in urban environments can benefit residents as well as governmental, public, and private entities. Smart cities have a wide range of applications: smart grids, smart transportation, smart buildings, IoT sensors, smart health, waste management, smart security, smart agriculture, among others (Mahobia and Pawar [2024], Syed et al. [2021]).

Figure 2.3 shows an overview of some smart city applications.

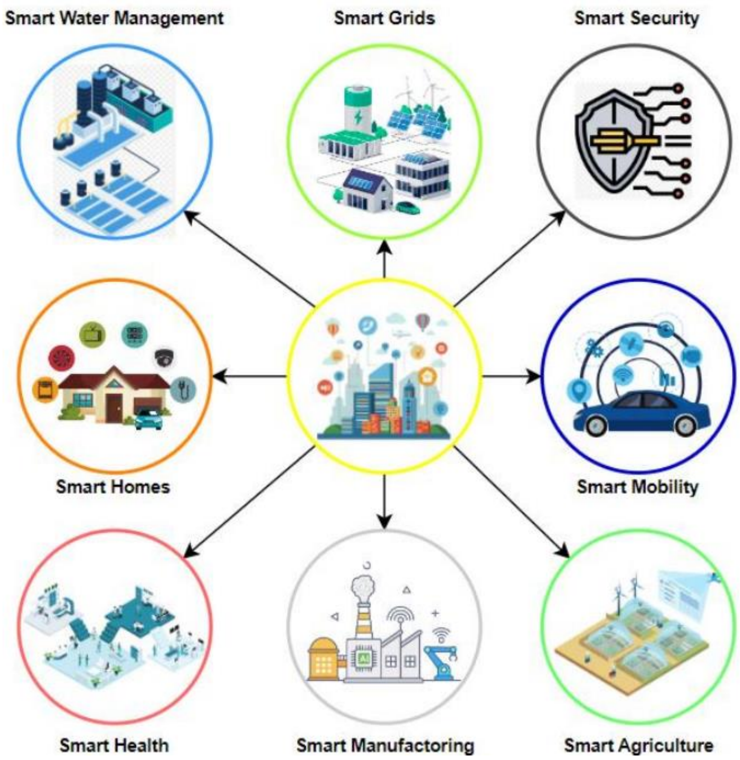

Figure 2.3: Common smart city applications.

Smart mobility is a new trend of connecting smart cities. The utilization of IoT and artificial intelligence (AI), along with other techniques (e.g. application of big data analysis), allows transportation systems to be more effective and sustainable. Everything is planned to take place in an environment that is indeed safer and more resilient, allowing better living standards for people. Some of the innovations in the smart mobility field are data sharing, interoperability and user-centric design (Mitieka et al. [2023], elettronica Federazione italiana di elettrotecnica et al. [2019]).

City-wide smart mobility projects are included in the smart city concept, as they are pursuing energy-efficient public transport, which is one of the actions to reduce air pollution (Mitieka et al. [2023], elettronica Federazione italiana di elettrotecnica et al. [2019]). The smart city infrastructure is upgraded when real-time data is integrated into public transportation, efficient traffic management, and sharing-ride platforms. In addition, the overall mobility in the city is improved through the use of technology. This approach brings an easier and safer journey, as well as a less costly option for citizens, which leads to the development of urban life quality (elettronica Federazione italiana di elettrotecnica et al. [2019]).

Consequently, through the use of IoT sensors, the flow of the traffic can be controlled by traffic signs. Moreover, the sensors of the smart parking lots are programmed to display the availability of parking, notifying the drivers of free parking spaces. Also, cars can be connected as well, to share with other road users data about the traffic or road conditions (Faria et al. [2017]).

Urban mobility receives a different focus through utilizing the combination of the IoT and real-time database analysis. The implementation of such an idea encounters various obstacles, such as: insufficient funding for roadways and intersections; absence of advanced traffic management programs and traveler services; absence of benefits for people choosing public transport over the shared bike alternatives. These problems are some of the numerous causes of the slow growth of smart mobility (Orlowski and Romanowska [2019]).

Of all the modes of transportation that smart mobility encompasses, the bicycle is considered the most sustainable, economical, and easily accessible. This kind of transportation has several benefits which seem to be an indispensable part of making the cities sustainable. Some of them are: health improvement, air pollution reduction, savings growth, and ecological sustainability (Bıyık et al. [2021a]).

With the introduction of smart cities, the phenomenon of "Smart Cycling", a concept that has been discussed by many researchers, was born. It involves not only the cycling part of the whole ecological system, but it is also linked to many other branches of IoT, that with the use of embedded hardware platforms, improve the biking experience. Fundamentally, smart cycling involves changing certain habits and patterns, as well as promoting collective transportation. This entails the introduction of new devices in bicycles, like apps, digital maps, operational systems, and sensors (Vieira et al. [2016]).

Smart cycling is being tracked in different regional researches, proving its necessity and reasons behind it. To add to this point, IoT has been applied in several cities across the world in numerous areas. With the contribution of sensors, actuators, and embedded hardware platforms, it is feasible to gather the cyclist's environmental data and caution them on the spot about the impending threat (Bıyık et al. [2021a]).

Figure 2.4 shows an overview of a generic IoT bicycle on the current times.

Devices such as GPS navigation and planning, real-time traffic information, health tracking, environmental monitoring, and data collection are some of the advancements related to smart cycling (Muhamad et al. [2020], Bıyık et al. [2021a]).

The table 2.4 shows some works in the area of smart cycling

Figure 2.4: Generic smart bicycle using IoT applications.

Table 2.3: Work related to smart cycling.

Work	Year	Work Description
Alam et al.	2018	Safety System with hand gesture recognition for turn indication, collision detection and accident detection.
Andres et al.	2019	Regulate speed to allow the cyclist to fetch a "green wave" of green traffic lights.
Makarova et al.	2020	Design of a monitoring unit for environmental conditions and cyclist health.
Oliveira et al.	2021a	Development of a multi-parameter system designed to evaluate the quality of cycling paths.
Theresa et al.	2022	Smart bike accident detection system implementation.
Munisamy et al.	2024	Design of a fallen detection system that sends an SMS with the cyclist's location to their next-of-kin
Bawa et al.	2024	Development of a safety monitoring and alert system.
Tamkhade et al.	2024	Display and control of the cyclist velocity.

2.4 Data-driven smart cities

Nowadays, sensor and data systems in smart cities record and provide the collection of a massive amount of data on things like traffic, energy consumption, pollution, and several other functions that keep an urban area operating (Sarker [2022]). Thus, this idea leads to "Data-Driven Smart

Cities", which are characterized by the compilation of data into manageable forms and actionable insights (Sarker [2022]).

The data-driven smart cities are coming up with applications that involve dealing with public health, public safety, environmental monitoring, and smart transportation (Sarker [2022], Bibri [2021]). Thanks to the deployment of IoT sensors and remote sensing, environmental data can be gathered (e.g. air quality, noise, rain, and temperature). Exclusive application of these data and technologies in different domains of the urban sphere, such as transportation, health, and energy management, leads to the offering of automatic, efficient, and personalized services (Bibri [2021]).

The data processing is necessary to achieve three major smart cycling fields: route planning and optimization, real-time monitoring of traffic conditions and cyclist safety. Sensors can capture cyclists' behaviors, and they can also be used for predictive analyses of risk areas and to ensure cycling infrastructure quality and reliability (Sarker [2022]). This introduces the non-motorized vehicle as a cleaner transportation option and also as a promoter of reduced congestion and CO_2 emissions, ensuring public safety (Bibri [2021]).

2.4.1 IoT as a database Generator

The use of IoT applications brings in real-time data acquisition from multiple sensors and devices, transforming the traditional data management procedure. It allows smart cities to store, analyse and expand their datasets. The virtual machines and platform services are scalable, allowing real-time and historical data gathering, which provides advanced analytics. Taking this into account, there is an increasing need for real-time processing of large volumes and flows of data across the grid, which, in turn, will lead to insightful decisions for organizations (Simmhan et al. [2018]).

For these fast data systems to work most effectively, handling the IoT data has to be done in the adequate way. Primarily, it should verify the data, to guarantee that the data obtained from IoT devices is reliable and safe. Second, using the method called data transformation, manipulating the data gathered in the proper format for processing. Reducing the processing time, leading to valuable insights a short period of time. The third component involves data validation at the time of processing, recognition and correction of incoherences that occur in the data capturing phase, making sure that the data is reliable (Simmhan et al. [2018], Sarker [2022]).

With IoT devices in smart cities, real-time data is being generated by sensors from different sources, including traffic sensors and transportation systems. This data acts as a valuable database for urban management, due to the growth of real-time data increase from ITS Kazmi et al. [2022].

These systems carry on interactions in their domain, which is the key to making the device connected, making sure that all the information is disclosed. In this process, the control and management tools are given to decentralize actions for shared purposes. The people that control the devices keep engaging in various roles and services, which is why they are said to be sustainable development players Kazmi et al. [2022].

The versatility of the real-time IoT data streams, is a crucial factor that enables to distribute comprehensive insights across transportation systems. There are several transport services and smart utilities that take advantage of this type of information in the urban system. By assessing its holistic and actionable nature, it is possible to make transportation better and convenient Kazmi et al. [2022].

2.4.2 Mapping Services

Mapping services are digital applications that obtain tools to examine, guide and interact with maps. These are a must for transportation; urban planning and development; emergency coordination strategies; environmental monitoring of projects; and business intelligence. These services are designed based on the data collection (GPS data), data processing (pillar of geographical information), and data presentation. This type of service is generally offered in the form of web mapping and GIS (Geographical Information System) databases (Carvalho and José [2022]).

Smart innovations present a particular smart mode of agile storage, retrieval, and management of the GIS databases, which are organized structures. They are often the foundation of various spatial data storage, such as maps, satellite images, and geospatial information. Additionally, these databases can be considered the place where all parameters and information are delivered. Hence providing the depth of information that is necessary for integrated analysis and decision-making structures (Luna-Reyes et al. [2012], Carvalho and José [2022]).

GIS is a state-of-the-art tool for smart city development that underpins the integration of location data to facilitate space utilities, public communication, innovation initiatives, and sustainable development. The combination of GIS technology with business intelligence tools, which is known as geospatial intelligence, results in a complete point view of operations that leverages spatial data to improve decision-making and control (Carvalho and José [2022], Luna-Reyes et al. [2012]).

In addition, database systems also contribute positively and might solve some big data issues in smart cities, by managing various spatial data types, reference systems, levels of detail, and reliability. These are very relevant abilities for dealing with the large amount of georeferenced data. Thus, daily functions of cities operations can be stable and decisions from different entities can be well-informed (Luna-Reyes et al. [2012]).

2.4.3 Smart Cycling

In the act of cycling, there are several factors associated with road bicycle accidents. It can be stated that, among the age groups of cyclists, the elderly have the highest rate of fatal accidents, suggesting that risky behavior or the inability to handle complex traffic situations contribute to these accidents (Sasaki et al. [2024]). Furthermore, as mentioned in section 2.1.3, the construction of good infrastructure for cyclists is directly linked to their safety (Kapousizis et al. [2023]).

To ensure rapid and efficient treatment in case of an accident, the cyclist distance to emergency Points of Interest (POI) is an important factor to be considered. Through the cyclist perspective the choice of the route is not highly correlated with the number of accidents in that same route. This means although cyclists prefer to avoid higher-risk routes, it is not the main factor deciding the route, thus affecting traffic security and cyclist safety on the roads (Huber et al. [2023]).

Hence, the cyclist choice can affect not only their safety, but also their health. The appropriateness of the routes to be chosen can be associated with several criteria, such as the quality of infrastructure, the conditions of traffic, weather, personal choice, and the accessibility to the hospital. Moreover, the threat levels varies for each cycling route and zone (Meng and Zheng [2023]).

Regardless of the cases, through Open Mapping Services and geospatial data, such as Open Street Maps (OSM) and GIS databases, it is possible to create methodologies that allow the extraction of viable information from an already computed dataset. Such tools have been emerging increasingly. As an example to be used in the development of the MobiAlert unit is the CityZones tool (Peixoto et al. [2023]).

This tool indicates an indirect perception risk according to the distance to an emergency structure. The consequences of bicycle accidents tend to be more dangerous when they occur in areas further from hospitals. Therefore, using an indirect risk to indicate the distance to these points does not aim to prevent or predict accidents, but rather to make cyclists aware of the risks, for them to adjust their level of attentiveness depending on the distance to the first aid centers.

2.4.4 Geospatial Data

Geospatial data is of high significance in such domains as urban planning, logistics, and natural resource management. As they provide information regarding the specific locations on earth, such as the features, infrastructure, and other data sources that can be used for further mapping (Lee and Kang [2015]). Even so, this kind of information presents a big challenge related to its storage and mining of sizable volumes of data, precision, and the integration of different information sources (Lee and Kang [2015]).

As it includes data on the exploration of the city, the urbanization of the space geospatial data is important in fields such as urban planning. The use of IoT devices to collect information, with the correct process and organization, can provide information regarding the specific locations on earth such as the features, infrastructure, and other data sources that can be used for further mapping (Lee and Kang [2015]). In spite of that, the information is presented in a very complex manner, once it is directly related to the storage and mining of large volumes of data, and its precision (Lee and Kang [2015]).

Spatial data generated by remote sensing technologies and IoT results in different kinds of formats (vector, raster, text). Normally resulting in complex relationships, calling the need to include topology, such as distance, and direction. (Lee and Kang [2015], Perumal et al. [2015]). Besides, the spatial autocorrelation is the process by which the nearby values are more similar, which does not comply with traditional data mining approaches. That is one of the main obstacles that need to be addressed regarding geospatial data mining (Perumal et al. [2015]).

One of the core problems of analyzing data for an extended period is its complexity, which directly affects the time it takes the user to understand and make a decision. Some of the topics that developers and researchers are investigating involve new methods and tools, on which they built new architectures (Perumal et al. [2015]).

The table 2.4 showcases which of the already conceived spatial projects are particularly suited for large-scale spatial projects. By contrast, this field of research is very limited, with few investigations and weak demonstration patterns, existing only a few notable trials being reported.

Table 2.4: Big Data Analytics

Work	Year	Work Description
Cui et al.	2014	The article proposes the Dynamic Pyramid R-tree as an indexing technique solution for managing large-scale remote sensing data.
Klein et al.	2015	It proposes a platform designed for scalable geo-spatial data, "PAIRS".
Yi et al.	2018	It showcases an indexing technique, MGIC and MTISIC, for management of massive data.
Lei et al.	2023	The authors propose a W-Hilbert curve, aiming to address the limitations of multiscale geospatial data.

Chapter 3

Proposed MobiAlert System

This thesis introduces MobiAlert, a model designed to monitor and evaluate the cycling experience by providing a visual representation of the risk associated with the zone a cyclist is currently crossing, aiming to enhance cyclist awareness and safety on the roads. The model is structured into two main components, focusing on the visualization and processing of large volumes of geospatial data. The risk of a zone is identified by the CityZones tool, which divides a city into small, square-shaped Mitigation Zones (MZ) and exports this information, including geospatial data and risk levels, into a CSV file.

In this context, the MobiAlert is divided into two modules: “PreProcessing” and "Cycling". The “PreProcessing” module handles the processing of large spatial datasets retrieved by the City-Zones tool, enabling fast data retrieval. The “Cycling” system focuses on the quick search algorithm, identifying risk zones (Z_i), visualizing these zones, and recording relevant information for further analysis.

The main goal of this chapter is to explain the proposed architecture of MobiAlert, as well as its design and implementation of all systems and sub-systems.

3.1 Proposed Architecture

The proposed model of the MobiAlert unit involves combining different systems to process a geospatial file with risk zones and show the risk level. The MobiAlert unit is divided into two processes, which are described in detail in the following subsections.

Overall, the first process, “PreProcessing”, corresponds to the manipulation of data from the GIS CityZones file. For each geospatial coordinate found in the file, a unique identification index is assigned, allowing for faster and more efficient searching. After assigning an index to all the coordinates, they are written into a CSV file, which is subsequently passed to the MobiAlert unit.

The second process, “Cycling”, is responsible for capturing the cyclist’s current position. Doing so, it is possible to identify the city where the cyclist is located, and conveying relevant information to the cyclist. It is composed of the “Cycling-NAV”, “Cycling-GPS Module”, and “Cycling-DataVisualization” subsystems, and is key to the MobiAlert unit’s impact on cyclist safety on the road, enabling the cyclist to take safer actions when in a higher-risk zone. Figure 3.1 illustrates the proposed approach for the systems and subsystems of the Mobialert unit.

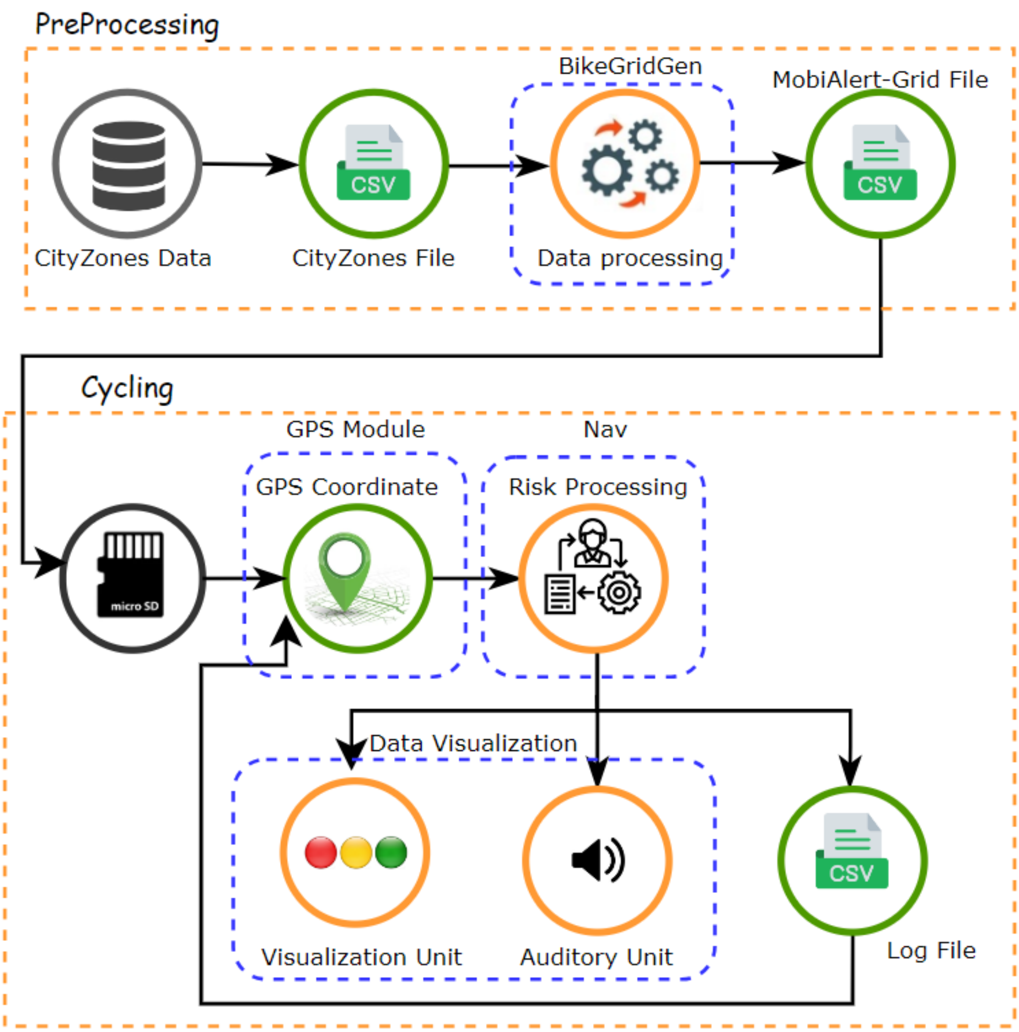

Figure 3.1: MobiAlert flowchart of the proposed architecture.

3.2 The CityZones tool

CityZones is a geospatial tool designed to compute urban risk zones for emergency management in smart cities. It uses real data to evaluate emergency response capabilities within urban areas by defining Areas of Interest (AoI) and categorizing them into low (green), medium(yellow), and high(red) risk levels based on distances to essential infrastructures such as hospitals, police stations, fire stations, and metro stations. The platform allows the user to specify an AoI into small, square Mitigation Zones (MZ) with predefined lengths, such as 100m per side. The results are visualized as heat maps with different colors for varying risk levels and can be exported in file formats, such as, Comma-Separated Values (CSV) for further analysis (Peixoto et al. [2023])

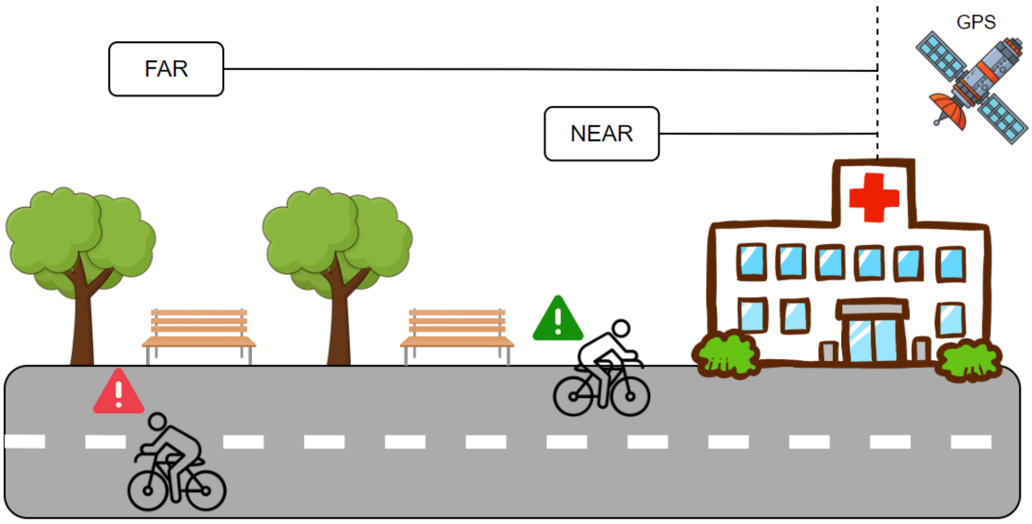

Figure 3.2: Risk Alerting in urban areas based on an emergy response infrastructure, with regions far from the hospital being riskier.

Generally speaking, the expected outcome of using small-shaped zones, usually lower than 100 meters in size, is that a typical city will have thousands of them. This way, the searching of GPS coordinates and the required identification of the current bicycle's zone may be a time-consuming task.

Since small refresh frequencies in the order of seconds will be considered, it is natural to expect that a bicycle will move to neighboring zones between subsequent measures: if in time $t0$ the bicycle is inside zone zi, the next time $t1$ might yield a movement location not far from that zone. This way, although zones are organized in a matrix with W columns and H lines, the searching within such a matrix should not be linear.

3.3 MobiAlert: PreProcessing

As mentioned, the "PreProcessing" system of the MobiAlert aims to manipulate the CityZones file. This section explains the general system of the "PreProcessing" unit and the mathematical model of "BikeGridGen".

The "PreProcessing" system takes a CSV file as input, provided by the CityZones tool, converting it into a new CSV file generated by the "BikeGridGen" subsystem. This system is depicted at the beginning of this chapter in Figure 3.1 and aims to index each coordinate found in the CityZones file to perform a simple and quick search. Although not shown in the figure, the creation of the MobiAlert: "Grid File" can be done on either a regular computer or the chosen embedded system.

It is known that the file provided by the CityZones will be named "city.csv". To facilitate searching, the MobiAlert "PreProcessing" unit simply creates a new file with the same name as the CityZones file but in a different folder to avoid any confusion in the embedded system and for the user. Therefore, it is expected that the data manipulation by the “PreProcessing” unit will follow the format illustrated in Figure 3.3.

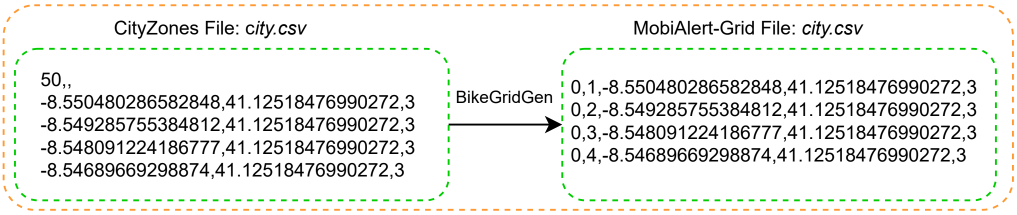

Figure 3.3: Examples of input and output of the “PreProcessing” system.

3.3.1 BikeGridGen: Model

In the system, the CityZones CSV data follows a format of *(lon, lat, r)*, where *lon* and *lat* represent longitude and latitude respectively. The primary objective of this subsystem is to organize this CSV data, with the approach being based on angles.

Given that we have coordinates *(lon, lat)* for each point, we can determine the bearing angle,θ, between different points using a specific formula denoted in 3.1. This angle provides with the direction in which a point is located relative to another. In that equation, y_{Pc} is the latitude of the first point found in the CityZones CSV file, and x_{Pc} is the longitude. This value will remain the same throughout the several steps of the “BikeGridGen”, ensuring coherence in the angles with other points in the file. On the other hand, x_{Point} and y_{Point} correspond to the latitude and longitude of a point in the CSV file.

This approach allows to systematically analyze the spatial relationships between points in the dataset, providing valuable insights into their relative positions and orientations.

$$\theta = \arctan\left(\frac{y_{\text{Pc}} - y_{\text{Point}}}{x_{\text{Pc}} - x_{\text{Point}}}\right) \times \frac{180}{\pi} \tag{3.1}$$

For initializing the grid creation, the first step of the MobiAlert "BikeGridGen" is to divide the CityZones CSV into four quadrants, with *Pc* as the reference point. The quadrants are established as it follows:

- 1st Quadrant: $\theta \in [0, 90]$;
- 2nd Quadrant: $\theta \in (90, 180]$;
- 3rd Quadrant: $\theta \in (-180, -90)$;
- 4th Quadrant: $\theta \in [-90, 0)$;

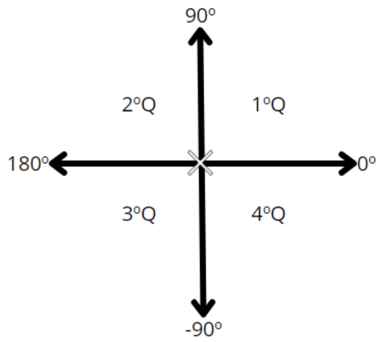

Figure 3.4: Step 1: Fetch the first CSV file point and calculate the quadrants.

Figure 3.4 represents the first step for the grid construction, where the white cross corresponds to *Pc*, the center of the *xy* referential.

After determining the quadrants, it is necessary to devise a method to index each point of the grid. So the second step consists in indexing each point. As there cannot be any repeated points, otherwise the search process may yield errors due to identical index values for different points, reference points were calculated with a bearing angle, θ, equal to θ = 0º, θ = 90º, θ = 180º, and θ = -90º relative to P_c , which in this case will be the center of the grid.

Figure 3.5 illustrates how the indexing scheme was assigned, with index i intersecting the y-axis and index j intersecting the x-axis.

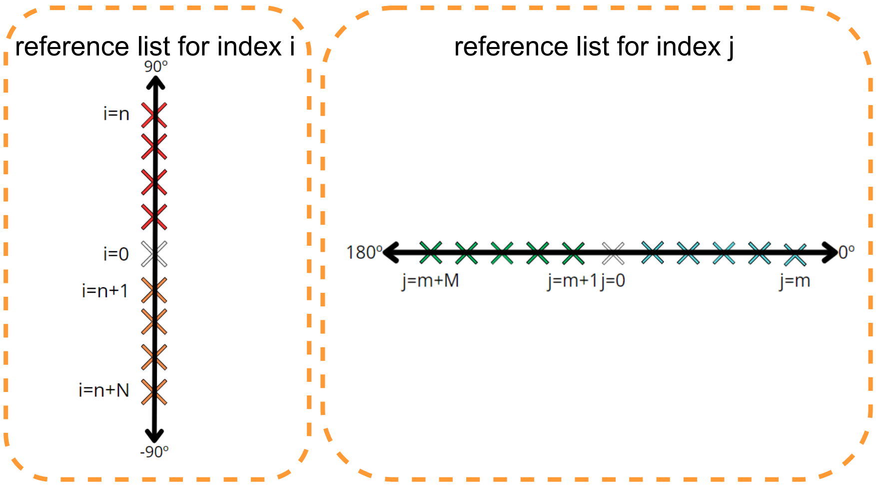

Figure 3.5: Step 2: Computing the reference lists in order to index the grid.

After determining the reference lists and quadrants, it comes the third step. This step involves assigning the indexes. For each reference list, points are examined respectively above, below, or to the side depending on the quadrant.

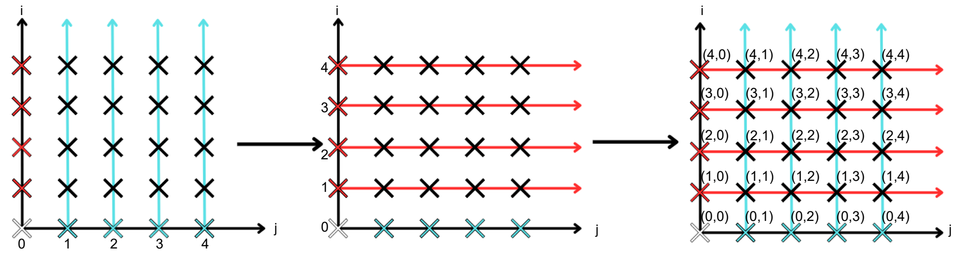

Figure 3.6: Step3: Grid calculation. In this example, the 1ºQ.

Taking the example of the first quadrant, depicted in figure 3.6, for each point on the reference list intersecting the x-axis, all points immediately above it (i.e., where θ = 90°) are checked, and the index of the point belonging to the reference list is assigned. Similarly, for the reference list intersecting the y-axis, points where θ = 0° are examined, and the index of the point belonging to the reference list is assigned. The final result for the first quadrant, considering a regular four-sided polygon, is illustrated in figure 3.6, this process is repeated for all quadrants, proceeding sequentially counterclockwise until the entire grid is filled as shown in figure 3.7

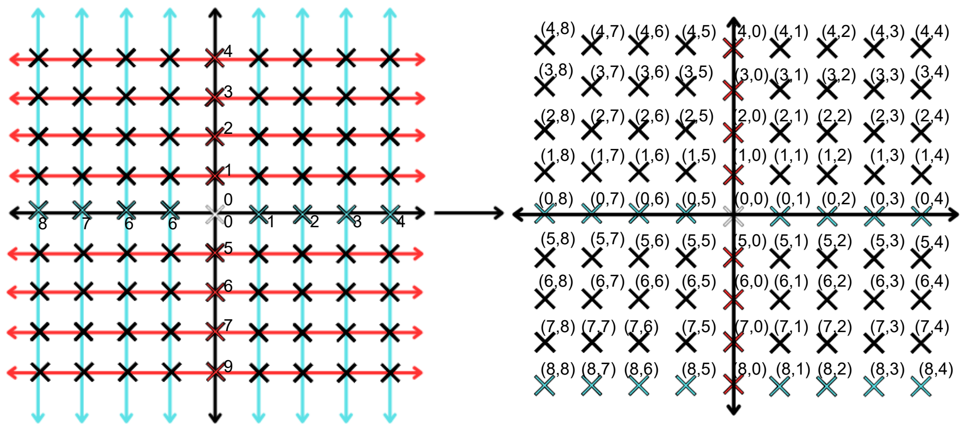

Figure 3.7: Step3: Computing the grid.

In the previous case, it was examined the simplest scenario involving a regular 4-sided polygon. However, the CityZones tool can generate various shapes, ranging from regular polygons to irregular ones. Relying solely on points with angles of 90º, -90º, 0º, and 180º relative to P_c may lead to an insufficient number of points in the MobiAlert: "Grid File".

Consider, for instance, a shape depicted in figure 3.8, where the first point retrieved from the CSV file, P_c, is shown in white, and the corresponding reference lists are depicted in red and blue. Following the approach used in the aforementioned case, it is anticipated that points colored light green will not be included in the final MobiAlert: "Grid File".

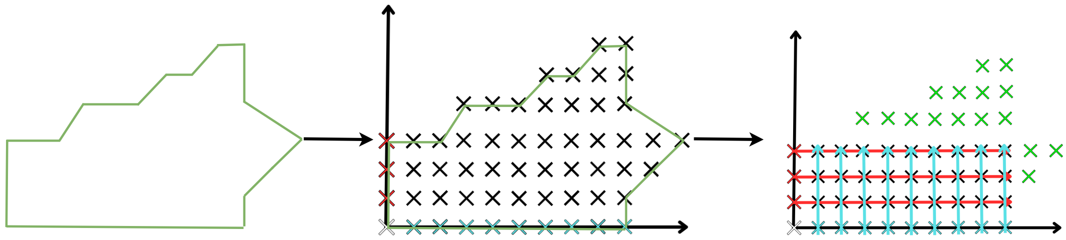

Figure 3.8: Computing the three steps proposed for an irregular shape.

To address this issue, it is necessary to consider not only points with specific angles relative to P_c, but also those lacking angles on certain sides. For example, let's consider the scenario illustrated in figure 3.8, where reference lists are established, and all points with angles θ = 90º or θ = 0º are selected. For each selected point, it is examined whether there are any points with a 180º angle. If not, the point is added to the appropriate reference list. Thus, in this scenario, and considering figure 3.9, points lacking angles on the left side will be added to the reference list for creating the i indexes. Conversely, if no angles are detected below ($\theta = -90$, the point will be added to the reference list for creating the j indexes.

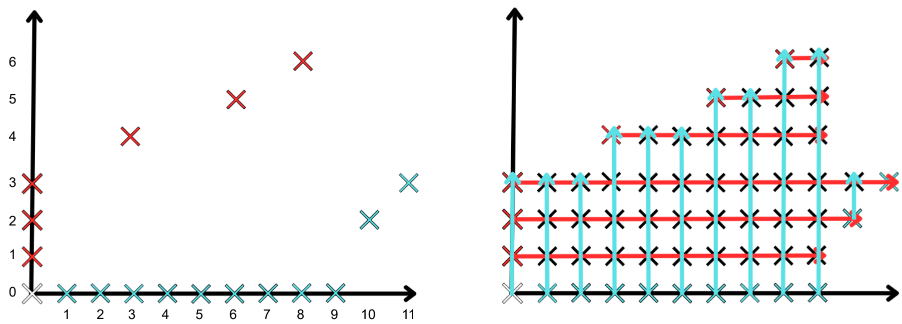

Figure 3.9: Computing the MobiAlert re-indexed grid for an irregular shape.

Once the MobiAlert Final Grid is created, the CSV file is ready to be exported to the MobiAlert "Cycling" unit.

3.3.2 BikeGridGen: Software

The previous subsection explained the logic behind the "BikeGridGen" subsystem. This subsection aims to describe in pseudocode the algorithm proposed to implement each proposed logic. Thus, the following three pseudocodes show a high-level representation of each previously proposed step

The first pseudocode, 1, corresponds to the first step described in the previous subsection, which is the division of the CityZones file into four quadrants according to its angle θ.

Algorithm 1 Step1 - BikeGridGen Initialization

```
Require: CityZones CSV File
Ensure: MobiAlert-Grid CSV File
  Q1, Q2, Q3, Q4, gridQ1, gridQ2, gridQ3, gridQ4 ← []
  for data in CityZones CSV File do                                   ▷ Step 1
      θ ← compute 3.1
      if θ ∈ [0, 90] then
          AddData ← Q1
      else if θ ∈ (90, 180] then
          AddData ← Q2
      else if θ ∈ (−180, −90) then
          AddData ← Q3
      else if θ ∈ [−90, 0) then
          AddData ← Q4
      end if
  end for
  gridQ1, gridQ2, gridQ3, gridQ4, ← compute Step2
  MobiAlert − GridFile ← joinGrids(gridQ1, gridQ2, gridQ3, gridQ4)
  return MobiAlert-Grid File                      ▷ Return MobiAlert-Grid File 3.1
```

The second pseudocode, 2, proposes an algorithm for creating reference lists, intersecting them, and verifying whether the MobiAlert: "Grid file" has all coordinates present in the City-Zones file indexed. If that is true, it simply returns the MobiAlert: "Grid File". However, if that is not true, the city provided by the CityZones file likely does not have a regular shape, so in the first iteration, some indexes will be missing. If this happens, as is the case in Figure 3.8, a pseudocode is described on the pseudocode 3, to index the remaining points.

Algorithm 2 Step2 and 3

```
Require: Quadrant, Pc
Ensure: Indexed quadrant
  indexi, j ← 0
  reflisti, j ← []
  for data in Quadrant do                                             ▷ Step 2
      θ ← compute 3.1
      if θ is 0 then
          reflistj ← (indexj, data)
          Increment indexj by one
      else if θ is 90 then
          reflisti ← (indexi, data)
          Increment indexi by one
      end if
  end for
  grid i,j ← createGrid(Quadrant, reflist)
```

```
indexed quadrant ← Compute Step3
if indexed quadrant ←points missing then
    indexedquadrant, indexes, reflists ← Compute alg. 3
    return indexed quadrant, indexes, reflists
end if
return indexed quadrant, indexes, reflists
```

Algorithm 3 Recalculate Grid

```
Require: Quadrant, reflists, indexes
Ensure: ReIndexed quadrant
  newpointsi, newpointsj = []
  for datai, dataj in reflistj, reflisti do
      for data in Quadrant do
          θ ← compute 3.1
          if θ is 90 then
              newpointsi ← dataj
          else if θ is 180 then
              newpointsj ← datai
          end if
      end for
  end for
  for data in QuadrantList do
      for datai, dataj in newpointsj, newpointsi do
          θ ← compute 3.1
          if θ is 180 then
              remove dataj from newpointsi
          else if θ is -90 then
              remove datai from newpointsj
          end if
      end for
  end for
  if newpointsj, newpointsi is not in reflists then
      Increment indexes by one
      reflists ← newpointsj, newpointsi
  end if
```

grid i,j ← createGrid(Quadrant, reflists)
return indexed quadrant, indexes, reflists

3.4 MA - Cycling

The primary concern of the "Cycling" system is to provide cyclists with a safe way to receive information from the CityZones tool without interfering with the cycling experience. For this purpose, a design of the connections of all components to the portable unit was necessary. The Fritzing tool was used for modeling the proposed hardware, as shown in Figure 3.10.

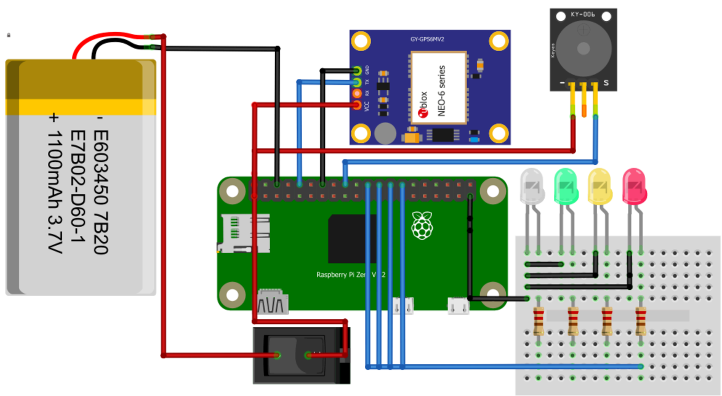

Figure 3.10: Proposed hardware architecture.

The proposed hardware schematic for MobiAlert includes a battery, a button to cut off the power to the embedded system, at least four LEDs (three to indicate the risk level for the cyclist and one to show the unit's operational status), and an auditory module to alert the cyclist of changes in the risk level of the area they are cycling in.

For the implementation of the modeled hardware, the flowchart, represented in Figure 3.11, provides a general approach for "Cycling", taking into account all the elements considered in Figure 3.10 and subsystems of Figure 3.1.

In the figure, the logical operation behind "Cycling" is illustrated. Oranges blocks correspond to the "GPSModule" of the "Cycling" system, used for fetching the cyclist's position. Red denotes the "Nav" subsystem, which executes the quick search algorithm to enable fast refresh rates. Yellow represents the "DataVisualization" subsystem, comprising the LED chosen for the risk zones and the buzzer for auditory support. Overall the system is cyclic, refreshing the "GPS module" every $t = 1s$. Further subsections will explain in detail every sub system and logic behind the "Cycling" operation.

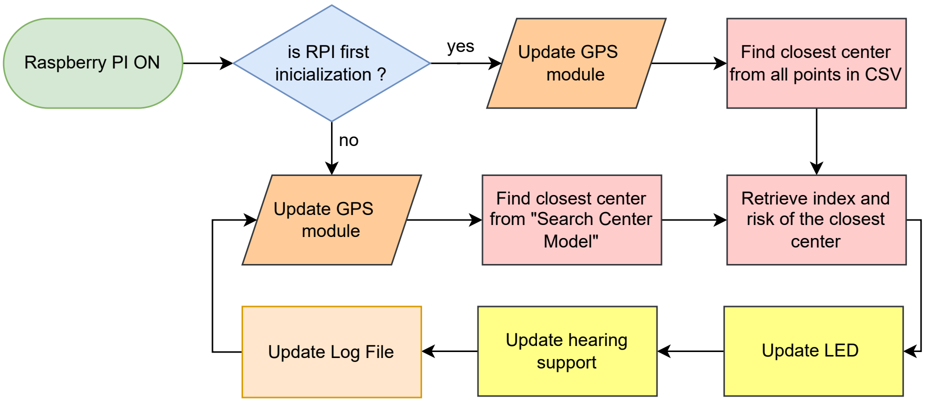

Figure 3.11: Proposed “Cycling” flowchart.

3.4.1 Sub-System: GPS Module

For that the MobiAlert starts by fetching the coordinate and comparing it to the MobiAlert “GPSModule”. To notice that the refresh of the “GPS module”, occurs every period, $T = 1$ second.

Generally the format of the coordinates retrieved by a GPS module will be in National Marine Electronics Association (NMEA) format, corresponding to a series of comma-separated values, where each value represents a specific type of data. The NMEA is a standard for interfacing marine electronic devices. It defines communication protocols and data formats used by navigation equipment on ships and boats. NMEA sentences provide information about position, speed, heading, and other navigational data. The standard ensures compatibility and consistency among different devices, allowing them to exchange data seamlessly (Banys et al. [2013]).

For retrieving GPS coordinates, the value typically returned by a GPS receiver module is a line that starts with the sentence identifier `GPGGA`, as exemplified in figure 3.12.

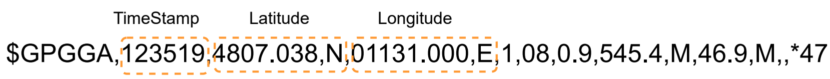

Figure 3.12: Typical line fetched by a GPS receiver.

3.4.2 SubSystem: NAV

The "Nav" system encompasses the search and computation of the zone in which the cyclist is presently riding. This approach requires determining the nearest center based on the cyclist’s coordinates and executing a swift search, typically accomplished within milliseconds. To enhance clarity, this section is subdivided into three subsections, offering an elaborate overview of the "Cycling" system and delineating the various steps involved in computing the risk zone within which the cyclist is traveling. Furthermore, the calculation of distances among centers is integral to this section.

$$d = R \cdot c \cdot 1000 \tag{3.2}$$

$$c = 2 \cdot \text{atan2}\left(\sqrt{a}, \sqrt{1-a}\right) \tag{3.3}$$

$$a = \sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right) \tag{3.4}$$

The Haversine formula ,3.2, calculates the great-circle distance between two points on the surface of the Earth, given their latitudes and longitudes. This formula accounts for the spherical shape of the Earth and provides an accurate measure of distance and it is described below (E Maria and Taruk [2020]).

Equation **(3.2)**: Distance Calculation (d)

$$d = R \cdot c \cdot 1000$$

- d: Distance between the two points in meters.
- R: Radius of the Earth, commonly taken as 6371 km.
- c: Angular distance between the two points in radians.

Equation **(3.3)**: Angular Distance (c)

$$c = 2 \cdot \text{atan2}\left(\sqrt{a}, \sqrt{1-a}\right)$$

- $\text{atan2}(y, x)$: The arctangent function of two variables. This function returns the angle between the positive x-axis and the point (x, y) in the plane. It ensures the correct quadrant of the angle is calculated, giving a result in the range $-\pi$ to π.

Equation **(3.4)**: Intermediate Variable (a)

$$a = \sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right)$$

- ϕ_1 and ϕ_2: Latitudes of the two points in radians.
- λ_1 and λ_2: Longitudes of the two points in radians.
- $\Delta\phi$: Difference in latitude between the two points:

$$\Delta\phi = \phi_2 - \phi_1$$

- $\Delta\lambda$: Difference in longitude between the two points:

$$\Delta\lambda = \lambda_2 - \lambda_1$$

3.4.2.1 Closest Center Calculation

It is recognized that the CityZones file divides a city into equally distributed squares with side length W, where each square corresponds to a zone risk, Z_i. Furthermore, the distances between the centers of the squares is equal to the side length of the squares, as illustrated in 3.13.

So, the easiest and quickest way to find the closest center, given an unknown position of the cyclist, is by doing a circle approximation. For a GPS coordinate to belong to a zone Z_i, an approximation to a circle of radius r was made. This can be mathematically represented as:

$$\sqrt{(x - x_i)^2 + (y - y_i)^2} \leq r \tag{3.5}$$

where (x, y) is the GPS coordinate and (x_i, y_i) is the center of zone Z_i, and r is the radius of the circle.

However, using a circle approximation leads to uncertainty of results, because if the radius it is equal to $w/2$ it will lead to a failure on the approximation once the square corners will be missed, as represented in figure 3.13. This can result in a misclassification of GPS coordinates that fall within the corner regions of the square zones. Since the circle approximation does not fully capture the square shape of the zones, there will be regions where GPS coordinates may be incorrectly assigned to neighboring zones.

To address this issue, adjusting the radius of the circle approximation was considered using the Pythagorean theorem. The maximum distance (max_distance) can be calculated as:

$$\text{max\_d} = \sqrt{2} \times w \tag{3.6}$$

where w is the length of the side of the square zone. The result for this change of radius can be visually seen in figure 3.13.

When the cyclist GPS coordinate (Px, Py) is fetched, the proposed solution involves creating a circle with radius $r = \text{max\_distance}$ and finding the closest center. In Figure 3.13, it is shown two possible situations. In situations where there are found four centers and all the distances are the same, it is simply chosen one of the centers with the highest level. However, situations with 2 centers or 3 centers are found occurs, the algorithm chooses the center with the smallest distance between the coordinate of the cyclist and the risk zone centers of the file.

To complement the explanation, a pseudo algorithm is proposed for calculating the closest center.

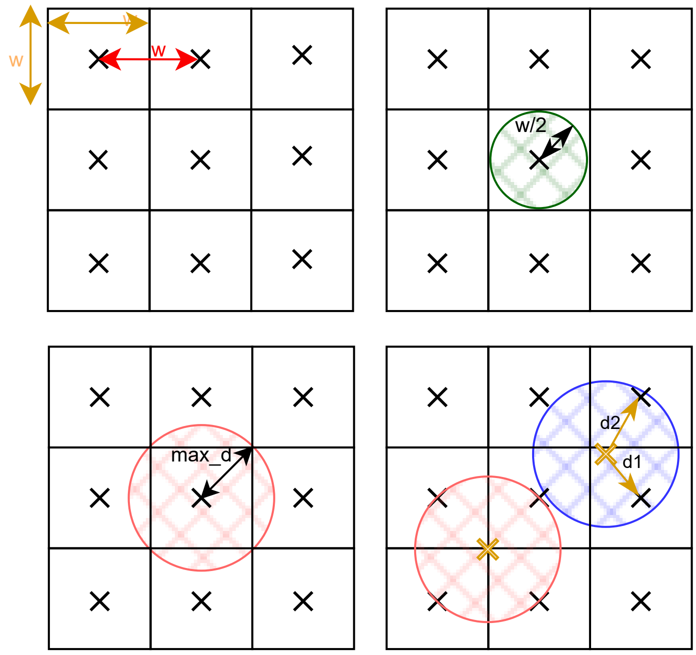

Figure 3.13: Computing the correct risk zone of the cyclist using harvesine distance and circles approximations.

3.4.2.2 Search Center

As illustrated in the flowchart 3.11, two distinct types of searches are visually apparent. One search occurs when the Raspberry Pi Zero W is initially powered on. Since it lacks information about previous cycling activities, it is imperative to first retrieve the indexes from the closest center found. This process follows the computation of algorithm 4, detailed in 3.4.2.1, where the distances between the cyclist's coordinates and all centers from the MobiAlert: "Grid File" are calculated.

Algorithm 4 Closest center calculation

Require: $coordinatesGPS, MobiAlert - GridFile, max\_d$
Ensure: Closest center coordinate
  $centers, distances \leftarrow []$
  for data in $MobiAlert - GridFile$ **do**
    distance ← Compute 3.2
    distances ← distance
    if distance < max_d **then**
      centers ← data
    end if
  end for

---

if size centers > 1 **then**
    closestcenter ← compute 3.2
    closestcenter
else if size of *centers* = 1 **then**
    closestcenter ← center
    return closestcenter
end if

Having obtained the center indexes (i, j) in the initial iteration, a quick search is simply performed by expanding the search area around the initially found center. To achieve this, a list of indices is constructed, including not only the initial center but also the surrounding centers, as shown in figure 3.14.

This approach allows for a more comprehensive and efficient search of the area near the cyclist's current position, thereby accelerating the GPS refresh rate, once that it doesn't need to compute the distance among thousand of centers, but only between nine centers.

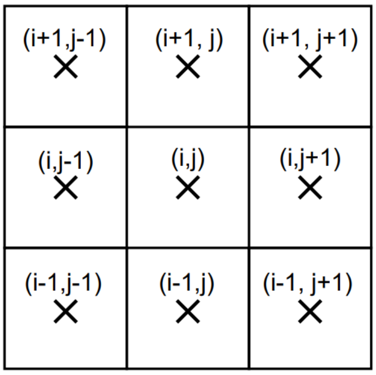

Figure 3.14: Computing the grid for the quick search.

As the refresh rate is in the order of seconds, this approach is deemed sufficient. Given that at time t_0, the cyclist is expected to be in zone 1, and shortly thereafter at time t_1, they would not have traveled a significant distance from their previous location. Hence, expanding the search area around the initial center provides an effective means of capturing the cyclist's updated position within a short timeframe.

3.4.3 Data Visualization

"Data visualization" is essentially what enables the MobiAlert unit to provide cyclists with visual and auditory support during their cycling experience. With the CityZones tool dividing the area into three risk zones, it becomes feasible to visually represent these zones using a color system: green for $Risk = 1$, yellow for $Risk = 2$, and red for $Risk = 3$. This approach allows cyclists to easily understand the risk level associated with the zone they are cycling in.

For easiest explanation, flowchart 3.15 provides a visual representation of the data visualization unit

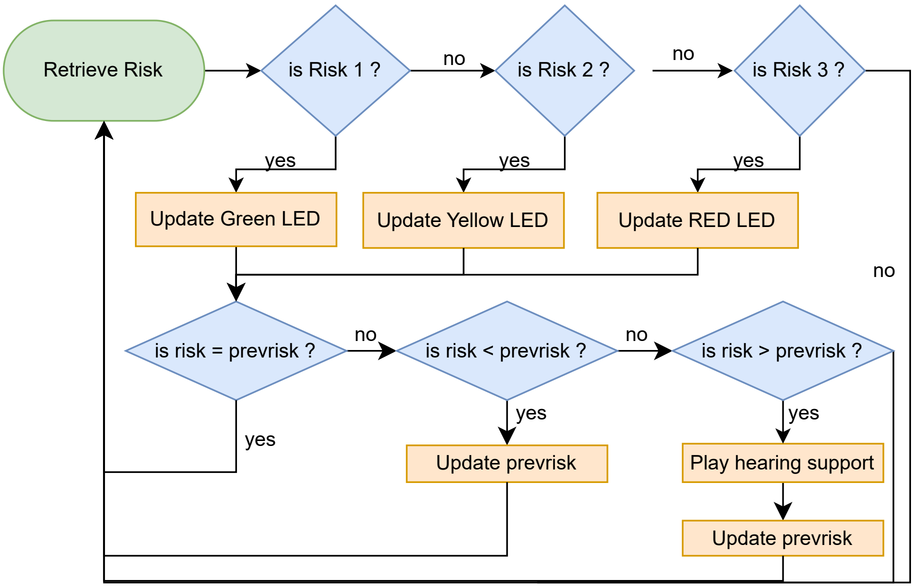

Figure 3.15: Cycling: "Data Visualization" flowchart.

3.4.4 Log File

The "Log File" is essentially an output generated after each Cycling cycle, following the hardware update. It serves the purpose of facilitating future processing, particularly for analyzing the results obtained from the cycling sessions. The "Log File" is formatted as a CSV, the structure is illustrated in Figure 3.16.

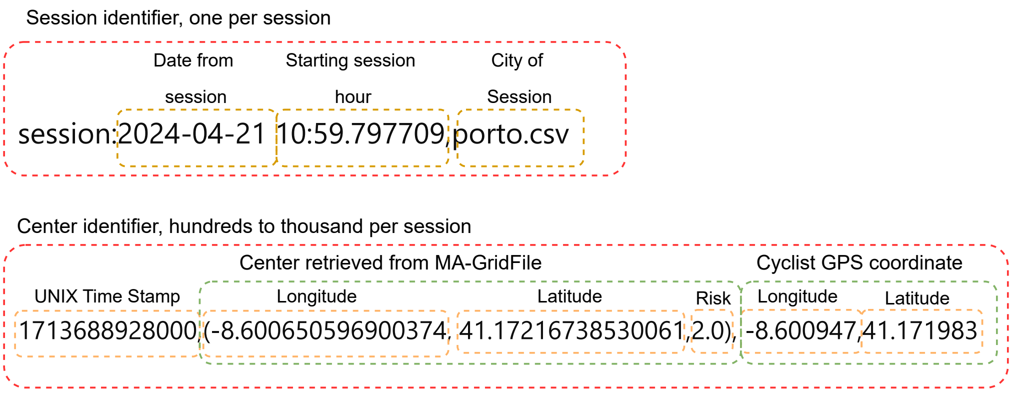

Figure 3.16: Log file structure.

Chapter 4

The Developed Prototype

Previous chapters described the proposed solutions to extract and manipulate data from a GIS database and to perform a quick search in order to provide the cyclist with relevant information as quickly as possible. Furthermore, a hardware architecture was suggested to show the cyclist the risk zone they are currently crossing, which involved the use of a button, a power supply, a GPS module, and two methods to provide relevant information to the cyclist.

This chapter proposes a proof-of-concept implementation of the proposed solutions. Such prototype includes the selection of hardware for the different subsystems, the development of a 3D case for the chosen embedded unit, and its attachment to the bicycle. Section 4.1 describes the chosen materials, while Section 4.2 illustrates the assembly and dimensions of the prototype.

It is important to note that the algorithms proposed in the previous chapter could be used with other types of hardware. This way, the developed prototype is just an effective implementation for validation purposes, but other configurations might be possible.

4.1 MobiAlert Components

One of the primary objectives of the MobiAlert unit is to assess and display the risk level of the area where the cyclist is crossing, providing warnings without disrupting the cycling activity and maintaining a low-cost price in terms of maintenance and purchase. To achieve these goals in the prototype, priority was given to selecting small, energy-efficient, and affordable components.

Figure 4.1 illustrates the proposed architecture, where each component was used for the assembly of the MobiAlert unit prototype. As shown in the figure, the visualization display, auditory unit, and sensor were chosen from the Grove line of products. This approach allows for a "boardless" and safer assembly, as the cables are inserted into the Grove Hat, without the need for soldering or complex wiring, accelerating the prototyping and development process (Bell [2021]).

For the embedded unit, the Raspberry Pi Zero W was chosen, along with a micro SD card. This card contains the Raspberry OS Lite system, the MA-Cycling algorithm, and a number of CSV files processed by the CityZones tool.

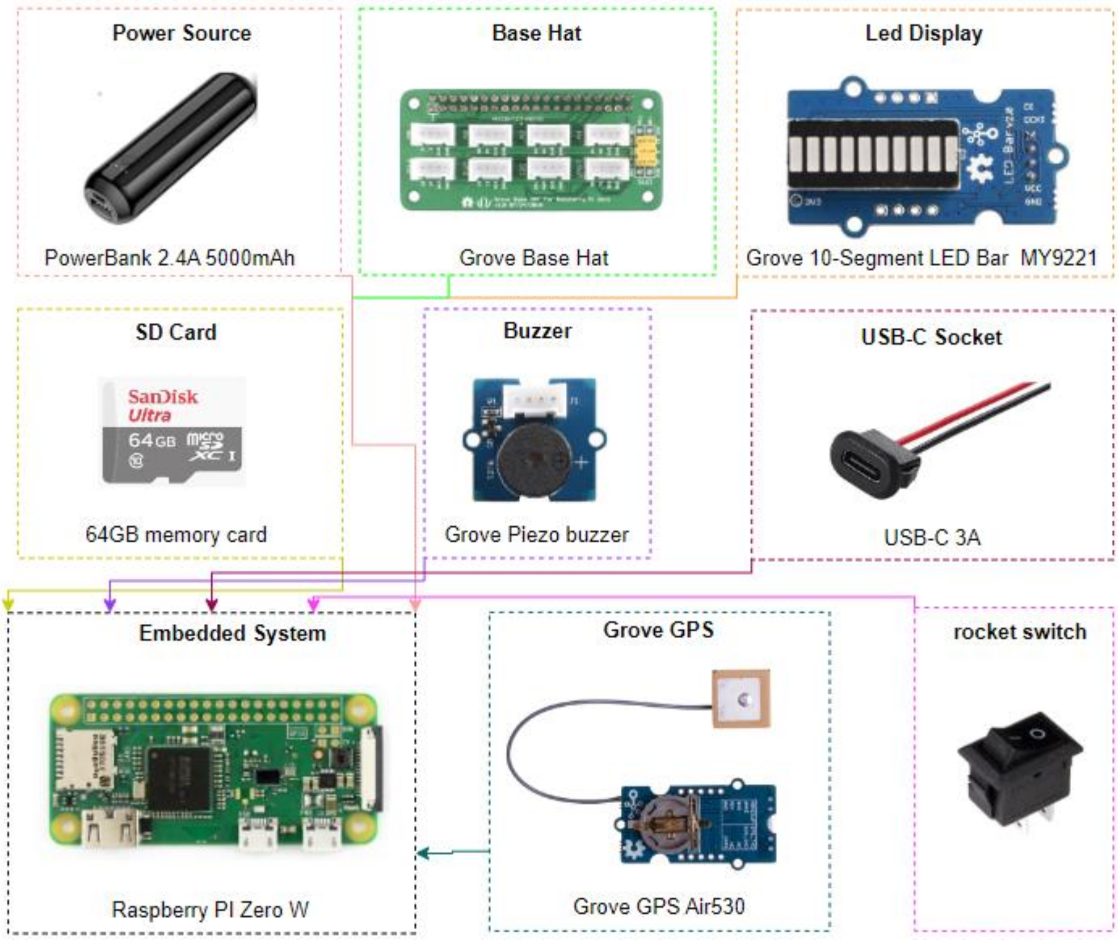

Figure 4.1: Hardware chosen for the implementation of the MobiAlert unit based on the proposed architecture.

The Raspberry OS Lite System was chosen due to its low memory requirements for installation. Furthermore, using the Raspberry OS Desktop is unnecessary for the MobiAlert unit, as the RPI Zero W supports SSH and Wireless connections (Toledo [2023]). With tools like Putty, it is easy to transfer files to the embedded unit from a simple laptop.

The Raspberry Pi Zero W is a SBC chosen due to the various advantages it offers, especially for projects that require a combination of processing power and low energy consumption. It has a small size, substantial processing capacity, low power consumption, and integrated wireless connectivity, eliminating the need for additional modules for wireless communication. This simplifies the project design and reduces additional costs.

It is important to mention that choosing the Raspberry Pi Zero W is not mandatory. Other models within the Raspberry Pi Zero family can be used. However, regardless of the model selected, using the Grove Hat with 40 pins is mandatory. This hat is essential for facilitating the connection of various Grove sensors and modules, which are standardized modular components designed to simplify the construction and expansion of hardware projects.

Table 4.1 provides a breakdown of the hardware costs, totaling approximately 70€. This selection ensures that the MobiAlert unit remains affordable while also boasting low energy consumption and adaptability.

Table 4.1: Detailed cost of MobiAlert components.

Component	Cost (€)	Quantity
Raspberrry PI Zero W	18.30	1
[Seeed] Grove GPS (Air530)	13.5	1
[Seeed] Grove LED Bar	5.90	1
[Seeed] Grove Buzzer	1.97	1
[Seeed] Grove Hat	13.7	1
USB-C Socket	3.2	1
USB-C Cable	5.99	1
Rocket switch	1.36	1
Total	**70.17 €**	

4.2 MobiAlert 3D Modulation

For the development of the case for the MobiAlert unit, the SolidWorks program was used. This is a computer-aided design (CAD) program used for 3D modeling and design, with the use of CAD tools enabling detailed modeling, thereby avoiding unnecessary prints and achieving detailed design while maintaining low manufacturing and maintenance costs. The primary priority of the case is to cause minimal impact on the bicycle and ensure safety during cycling, so as not to interfere with the act of pedaling. Additionally, Polyethylene Terephthalate Glycol (PETG) material was used, which is recommended for impact resistance as it is strong and resilient.

To avoid conflicts with the power source, in case it needs to be replaced, two cases were created: one for all the hardware components of the MobiAlert unit, and another for the power source. This way, the power source can be charged without having to remove the unit from the bicycle.

4.2.1 Cover Dimensions

The casing of the MobiAlert "Cycling" unit was designed to avoid any disruption to the cyclist. Various layouts were tested to achieve the most optimized design for the proposed hardware. Figures 4.2 show the dimensions of the casing for the battery and the "Cycling" unit's.

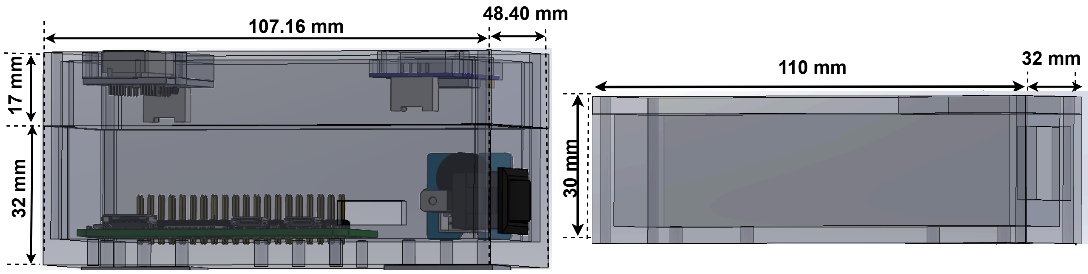

Figure 4.2: Dimensions of the MobiAlert cover.

4.2.2 Covers Assemble

The assemble of the MobiAlert unit consisted on retrieving existent components already made by GrabCad users, they display a ".SLDASM" file that allows you to assemble diverse modelated components. In this case, they were assembled to test the viability of the cover before printing, allowing for a reduction of the 3D fillment consumed.

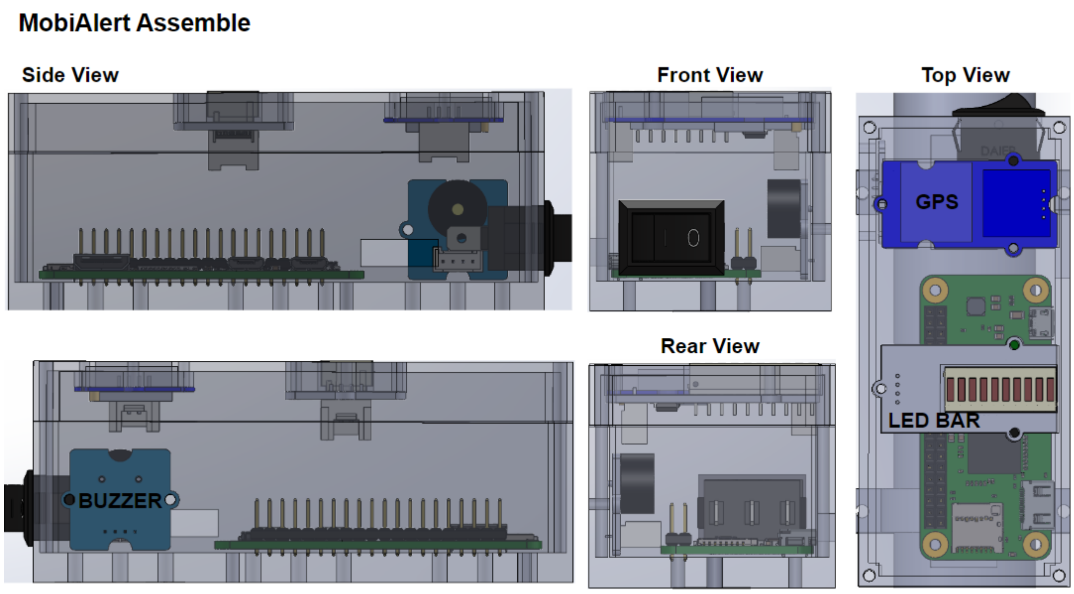

Figure 4.3: Assemble of the MobiAlert cover.

Even though the cables , HAT grove and USB Socket were not displayed on the figures, it is possible to see that the case modelated well-fits the hardware proposed.

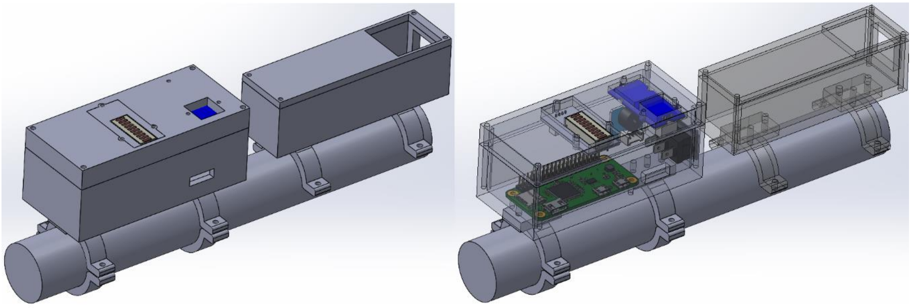

Figure 4.4: Final Assemble of the MobiAlert.

4.3 3D-printed cases

Figure 4.5: Real life assemble of the MobiAlert unit.

Figure 4.6: Real life final assemble of the MobiAlert unit attached to a bicycle.

4.4 CityZones Map

The CityZones tool, illustrated in Figure 4.7, allows for selecting a size for the zones, which, in this case, was set to 100 metres. This means the city will be divided into thousands of square zones, each 100 metres on a side. Additionally, the tool allows for assigning weights, f(p), to the city's POI, such as police stations, fire stations, and hospitals. In this specific case, hospitals were assigned a weight of 10, while the others received a weight of 5. This is due to the fact that, in the event of a road accident, a hospital will be the most relevant among these three POI.

Figure 4.8 presents the map exported from the CityZones tool, showing a good distribution of risk throughout the city according to the weights of the considered POI.

Thus, with the extraction of the map of Porto, the proposed hardware, and the printed casing of the unit, it is possible to test the viability of the algorithms and functional logic proposed in chapter 3. These results are described in the chapter 5.

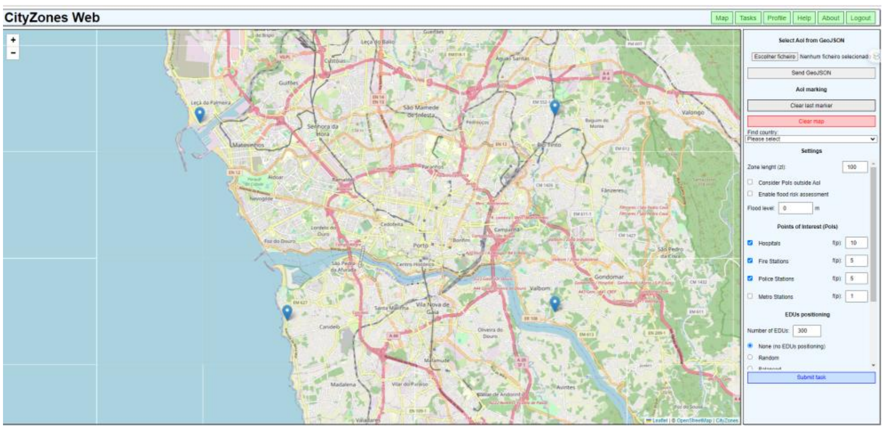

Figure 4.7: CityZones tool initial page.

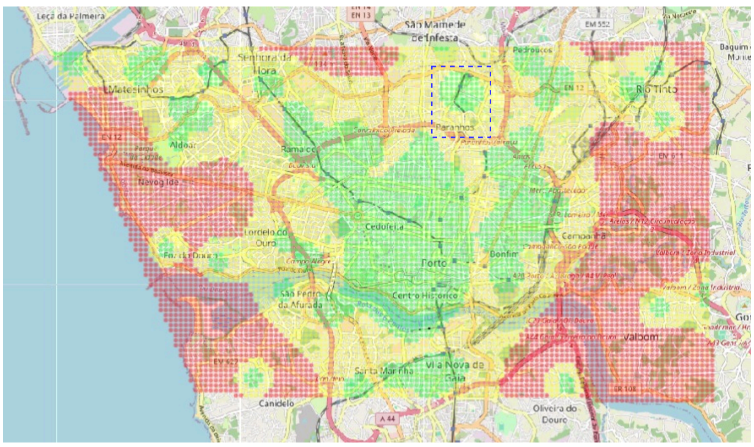

Figure 4.8: Exported risk map of Porto, Portugal.

Chapter 5

Experiments and Results

The primary objective of the MobiAlert unit is to alert the cyclist in near real-time about the risk zone they are crossing, using the CityZones tool, which divides a city into risk zones and exports this information into an unorganized CSV file. To achieve this, the proposed approach utilizes the "PreProcessing" phase to organize the CSV file into an indexed format, where each coordinate corresponds to an (i, j) position, allowing for a quick search of each zone in the city. Finally, the "Cycling" approach performs the search, updates the cyclist's coordinates, and alerts the cyclist about the severity of the risk zone they are in.

To validate the proposed solution, a generic approach was defined, enabling the implementation of the MobiAlert unit with its systems and subsystems, such as "BikeGridGen", "GPS module", "Nav", and "Data Visualization". This represents the first deployment of the MobiAlert unit.

This chapter presents the results obtained during well-defined experimental scenarios, focusing on the execution times of each system/subsystem and the impact on cycling. To this end, three tests were conducted in city of Porto, Portugal.

5.1 Using the CityZones tool

As previously mentioned, the CityZones tool allows the division of a city into risk zones and their extraction into a CSV file. For the validation of the algorithms and logic proposed in previous sections for subsystems, as well as hardware and modeling, the testing phase was conducted in the city of Porto.

The CSV file returned by the CityZones tool will be named "porto.csv" and has the format shown in Figure 5.1. The first line identifies the distance from the center to the side of the square; for example, if this distance was ten, the square would have an area of 400 m². The remaining lines contain the centers of the squares and their corresponding risk zones.

It is also evident that this is not a regular polygon, as depicted in Figure 4.8. Therefore, the recalculation of reference lists will need to be performed as explained in the subsection 3.3.1.

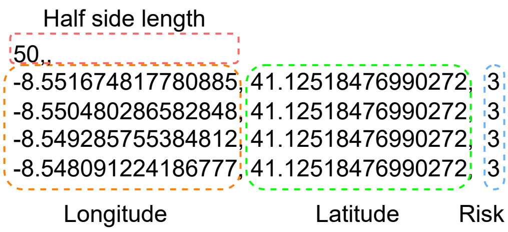

Figure 5.1: A segment of the CSV returned by CityZones.

The exported file contains 8,208 computed risk zones and occupies approximately 322KB, where each line corresponding to a zone has a maximum size of 39 bytes per line. Thus, the next step after exportation will be the manipulation of the file through the analysis of geospatial relationships between the various risk zones.

5.2 PreProcessing Results

Following the logic proposed in section 3.3, this system uses the "BikeGridGen" subsystem to place indexes $[i][j]$ on each center found in the CSV file. For the validation of each step, the following image illustrates step 1, corresponding to the division of quadrants, and step 2, the creation of reference lists (3.4, 3.5).

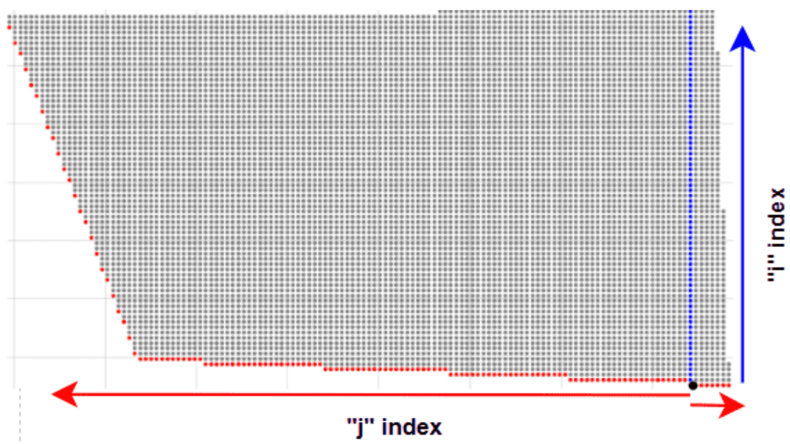

Figure 5.2: Computing the first steps for indexing the CSV file.

In the Figure 5.2, it is noted that the first point of the CityZones file is colored black and will be Pc. Using equation 3.1, the division of quadrants and the identification of reference point lists for the creation of the matrix it is made. The blue and red arrows indicate the orientation of the index increments, which increase as they move away from the origin (black point) and follow a sequential logic in a counterclockwise direction starting from the 1st quadrant.

For this specific AOI, 463 zone centers were found in the first quadrant, 7,745 in the second quadrant, and 0 in the third and fourth quadrants. The division of the quadrants it is shown on Appendix A, Figure A.1. The indexes for the reference list j range from [0,7] for the first quadrant and [8,122] for the second, thus the j index takes values between [0,122]. The indices for the reference list i range from [0,71].

The final step, described in Figure 3.7, involves assigning each index to each point in the reference list "j". The index value of that point is assigned to all points above it, while for the reference list "i", the index value is assigned to points with a θ of 0° or 180° and subsequently intersected.

For a better understanding, the Appendix A, Figure A.2 shows the values for the computed reference lists, Figure A.3 shows the computed grid with the i indexes and Figure A.4 shows the computed grid with the j indexes.

The final result is also a CSV file with 8,208 lines, following the format illustrated in Figure 5.3.

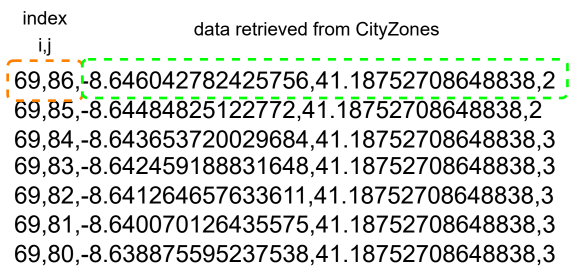

Figure 5.3: PreProcessing: MobiAlert: “Grid File”.

After creating the MobiAlert: “Grid File”, it is ready to be exported to the Raspberry PI Zero W. Compared to the CSV file extracted from CityZones, this file has a maximum line size of 46 bytes and occupies 370 KB, which imposes a low memory requirement.

To confirm the proposed architecture, tests were conducted for the “PreProcessing” algorithm to determine the best approach: executing the algorithm on the Raspberry PI or an external device. For the tests, the external device was a Lenovo Ideapad laptop. The results are described in Table 5.1.

Table 5.1: PreProcessing: “BikeGridGen” computing time.

Device	CPU Clock	RAM	Execution Time
Laptop Lenovo	2GHz	16GB	**75**s
Raspberry PI Zero W	1GHz	512MB	**1016**s

The processing time on the Raspberry PI is substantially longer than on the computer, with a difference of 941 seconds (16 minutes). Therefore, it is ideal to run the “PreProcessing” phase on a device with greater processing capacity to achieve faster indexing and then export it to the embedded system, as shown in Figure 3.1.

5.3 Cycling Results

For the Cycling system, algorithms and architectures were proposed to enable the “GPS module” to update within one-second intervals to provide information as quickly as possible. To achieve this, “Search Center” (3.3.2) was proposed, which constructs an index list in a $3x3$ matrix format.

Algorithm 4 is the algorithm that calculates the nearest risk zone to the cyclist’s coordinates. However, computing this with all points from the CSV file takes about 5 seconds per iteration, preventing the GPS from updating every second. This time was reduced to less than 300ms with the logic proposed in “Search Center”, allowing for rapid GPS and relevant hardware updates

5.3.1 Case Studies

To test the accuracy of the proposed algorithms and hardware, four case studies were implemented in the city of Porto and are demonstrated in the following subsections. For each of the results, an analysis table was created with the total distance traveled in each route (**d**), the total operation time of the route (**Op.Time**), the number of centers with risk caught during the cycling session (**Total Centers Nº**), the numbers of non-repeteaded centers with risk caught during the cycling session (**Nº of non-repeated centers**), the average time the cycling takes for a new cycle , this is, between GPS updates ($\overline{T}$ **Cycling**), and the time of search for the Search Center model proposed in subsection 3.4.2.2 ($\overline{T}$ **Search Center**).

On the figures 5.4, 5.5 and 5.6, the $y-axis$ corresponds to the latitude and the $x-axis$ to the longitude. Moreover, these values are summarized in Table 5.2

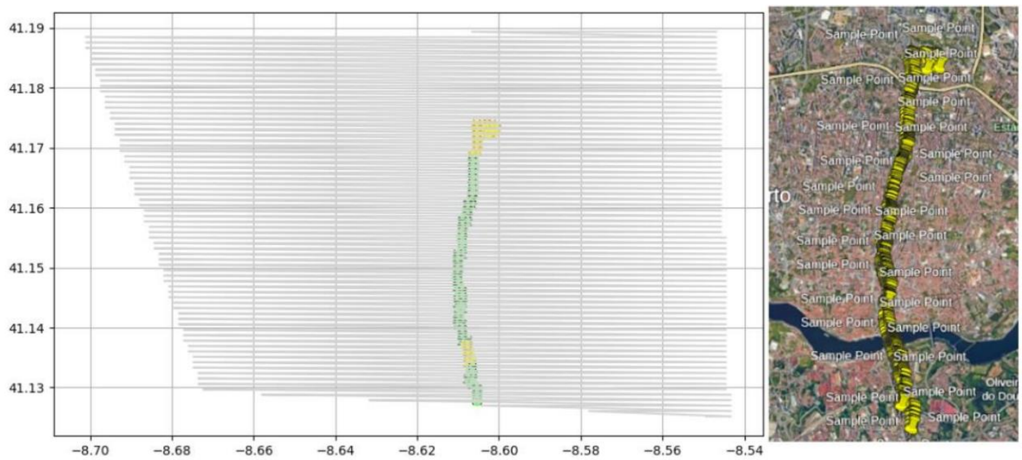

Figure 5.4: Cycling route 1.

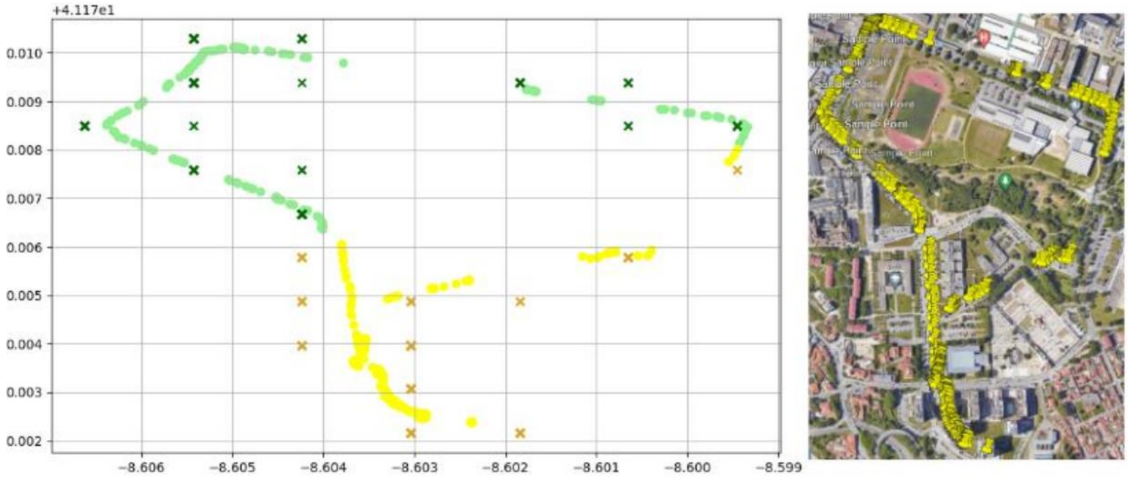

Figure 5.5: Cycling route 2.

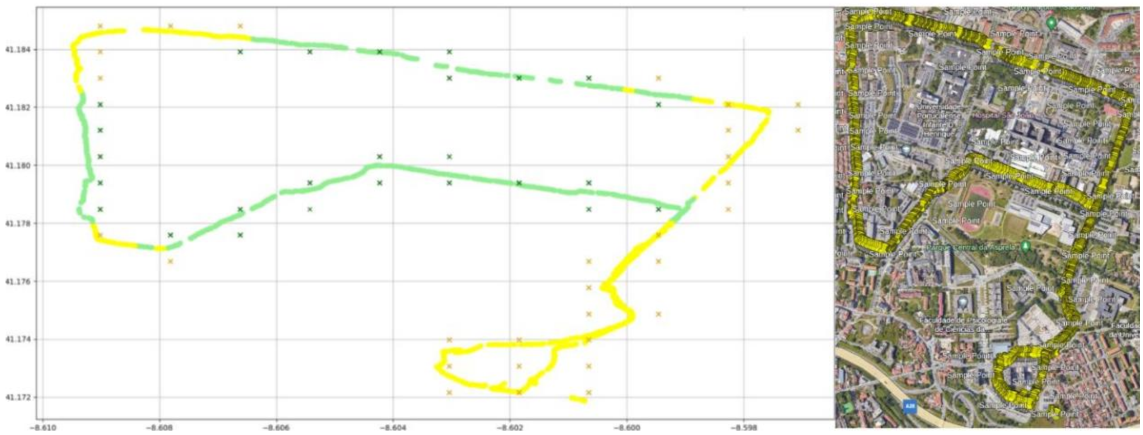

Figure 5.6: Cycling route 3.

Parameter	d [km]	Op.Time [min]	Total Centers Nº	Nº of non-repeated centers	$\overline{T}$ Cycling [ms]	$\overline{T}$ Search Center [ms]
Case 1 Result	6.3	55	1506	69	2201	0.284
Case 2 Result	2.8	54	1500	24	2183	0.278
Case 3 Result	5.4	78	2246	56	1535	0.255

Table 5.2: Summary of results for different cases in the MobiAlert system.

Relying solely on $\overline{T}$ Cycling and $\overline{T}$ Search Center to see the MobiAlert performance will tell us little about it, because values really high appear, disturbing the mean. Having this in account, further processing results can be found on Appendix A. Figures A.8, A.9 and A.10 represent the times for the Search Center model for each new cycle of GPS coordinates during a cycling session. In these images, green indicates iterations considered excellent (i.e., search times under $t = 300ms$), while yellow indicates iterations with search times over $t = 300ms$.

Figures A.5, A.6 and A.7 representing the period, T, of the MobiAlert unit, encompassing the GPS coordinate retrieval, risk search, hardware update, and repetition. The initial project goal was for the MobiAlert unit to have $T = 1000ms$. Thus, in these images, green points represent $T = 1000ms$, yellow points represent $1000ms < T < 5000ms$, and red points represent $T > 5000ms$.

Tables A.1 and A.2 in the same Appendix summarize the information from the images. The first table shows the execution times for the 'Search Center' model, displaying its maximum, minimum, and threshold values at $t = 300ms$. The second table shows the period values of the Cycling unit, demonstrating the consistency of results in each test.

It is important to note that period variations mainly occurred due to GPS signal failures. The program pauses until a coordinate is captured by the module because there is no need to proceed without GPS data.

After the cycling activity, the MobiAlert unit has a log file with stored metadata of the cycling route. This file has a maximum size of 101bytes for each entrance (line), following the format on 3.16.

5.3.2 Discussions

In Chapter 4, the MobiAlert unit is described as relatively inexpensive and easy to maintain, thanks to the Grove line of hardware. A compact and small casing design was achieved, which had minimal impact on the bicycle. However, the location of the USB-C input cable could be improved for greater convenience, as the cable exiting from the side may cause some inconvenience to the cyclist.

For the CityZones tool, the chosen weights provided a good distribution of risk zones, allowing for a diverse risk representation on the map.

Regarding the results, the "PreProcessing" unit takes some time to convert the city data returned by CityZones, when done on the Raspberry PI Zero W. It is therefore recommended to perform the conversion on an external device with higher processing power, for example, a simple laptop computer.

Additionally, the mobile unit MobiAlert, defined by the "Cycling" system, performed relatively well. This delay is primarily due to the GPS, which tends to be inaccurate and often fails. Furthermore, the linear model Search Center demonstrated its efficacy in performing quick searches based on the MobiAlert: "GridFile".

The verification analysis of centers was performed by comparing the detected centers with the risk map from the CityZones file, which provided positive feedback.

Overall, the MobiAlert unit is cost-effective, easy to maintain, and has a minimal impact on the bicycle. Moreover, the created algorithms are consistent and fast, which is a crucial factor.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

The growing encouragement to use bicycles as a healthy, fast, and sustainable mode of transportation makes the current global landscape favorable for constructing high-quality bike lanes in major urban centers. However, these cities are typically designed to accommodate conventional vehicle traffic, complicating the lives of cyclists who, besides facing environmental health risks, must compete for space on the roads with automobiles, making them vulnerable to severe and fatal traffic accidents.

Aware of these problems, this project proposed a new model that integrates innovative technologies into urban environments, addressing these safety challenges by leveraging cost-effective hardware and open geospatial data to provide real-time risk assessments and alerts, enhancing cyclist safety and promoting sustainable urban mobility.

The architecture proposed in this project, MobiAlert, was developed by creating and interconnecting several subsystems to perform specific tasks. The implementation of the MobiAlert unit was presented through its systems: "PreProcessing" and "Cycling". The first involves extracting the CSV file from the CityZones tool and converting it into an indexed city using a mathematical logic based on angles, carried out by the "BikeGridGen" subsystem, generating the MobiAlert: "Grid File" indexed city file. The second involves integrating various systems, including the "GPS module", which captures the cyclist's current coordinate. The "NAV" subsystem allows a quick search of the coordinate based on a 3x3 matrix range, the "Data Visualization" subsystem, which updates relevant hardware, and the "LogFile", which records data for future processing.

All these systems and subsystems were fully developed, and three case studies were conducted in Porto to confirm the accuracy and consistency of the proposed algorithms, logic, and prototype throughout this dissertation.

Given the results obtained in this research and its multiple contributions, the MobiAlert unit proved to be inexpensive, easy to maintain, and highly functional, with an operational period generally equal to 1 second, except when the GPS fails to acquire data. This makes it a valuable asset for cyclists on the road, alerting them to areas along their route that require greater attention and caution from the cyclist.

Finally, it is worth mentioning the publication of two scientific papers discussing results from this dissertation, described as follows:

- Ferreira, J. F., Bittencourt, J. C. N., Costa, D. G. MobiAlert: A Data-Driven Embedded System Approach to Enhance Safety for Cyclists. 2024 IEEE Smart Cities Futures Summit, Marrakech, Marocco, 2024.

- Ferreira, J. F., Costa, D. G. Enhancing Cycling Safety in Smart Cities: A Data-Driven Embedded Risk Alert System. *Smart Cities*, 2024, submitted.

6.2 Future Work

The MobiAlert unit has a significant growth potential. For this dissertation, the unit was simply proposed to index a city for quick searches and to assess certain risks. However, based on the various components of the MobiAlert unit, several suggestions are put forward.

Regarding the achievements of this dissertation, it is recommended to conduct more tests and develop an algorithm to calculate the accuracy rate of the risk zone where cyclists pass, ensuring the consistency of the algorithm. There is also a suggestion to design a power supply for the unit, considering that this thesis focused on timing and used a power bank as a power source. This could potentially be changed to a cheaper and more compact power source if designed by someone in the field.

Furthermore, it is suggested to implement the MobiAlert unit on other devices to determine if the Raspberry Pi Zero W is the best option in terms of energy consumption, processing power, and memory. Other recommendations include enhancing the unit with additional hardware. Given that the prototype had minimal impact on the bicycle, there is room for modifying or expanding the unit. For instance, incorporating accelerometers, gyroscopes, humidity/UV sensors, accident detection systems, etc., which are widely discussed in the literature and have shown positive impacts on cyclists.

Therefore, the MobiAlert unit has substantial potential for future development, where this dissertation provides a starting point for more challenging and newly state-of-the-art device for cyclist safety.

Appendix A

Appendix 1

A.1 MA-PreProcessingResults

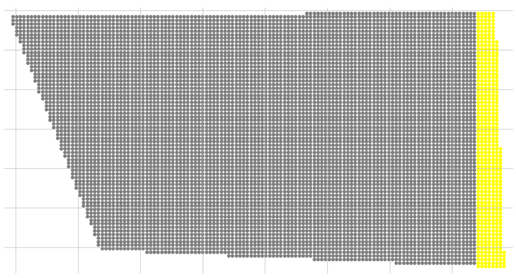

Figure A.1: Computed quadrants.

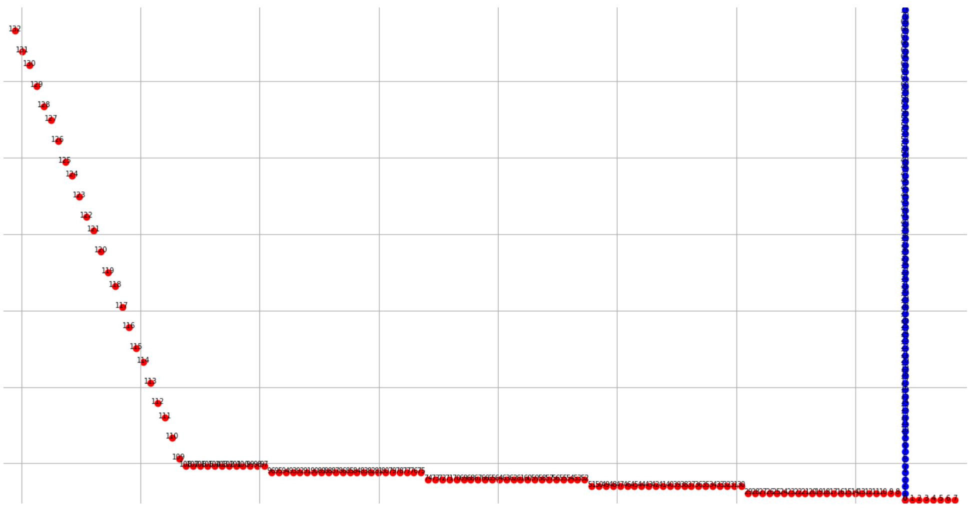

Figure A.2: Computed reference lists.

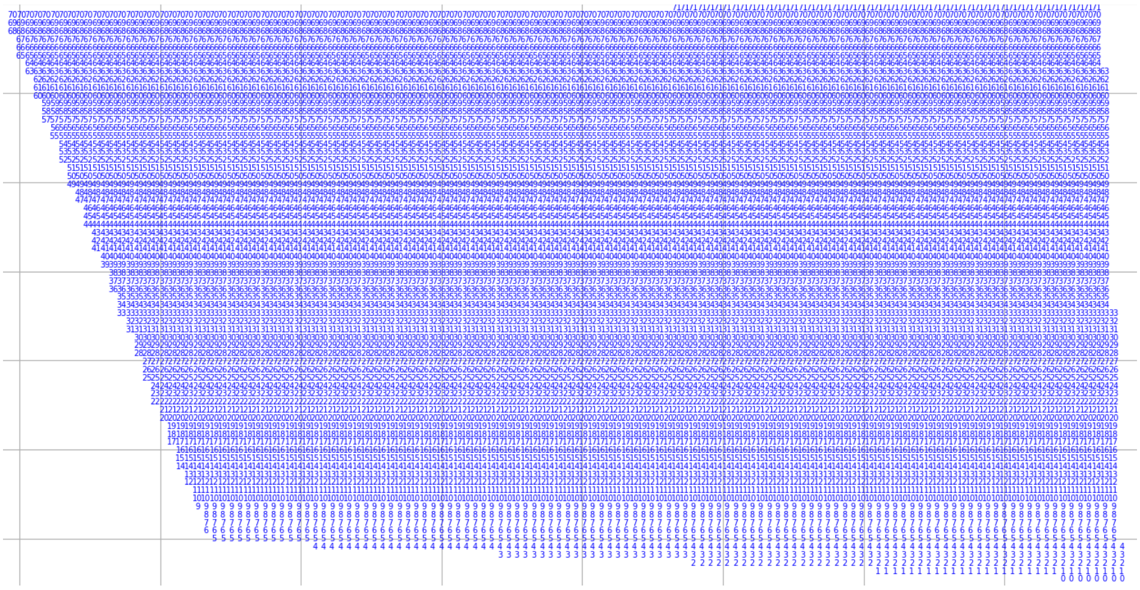

Figure A.3: Computed indexed grid for reflist i.

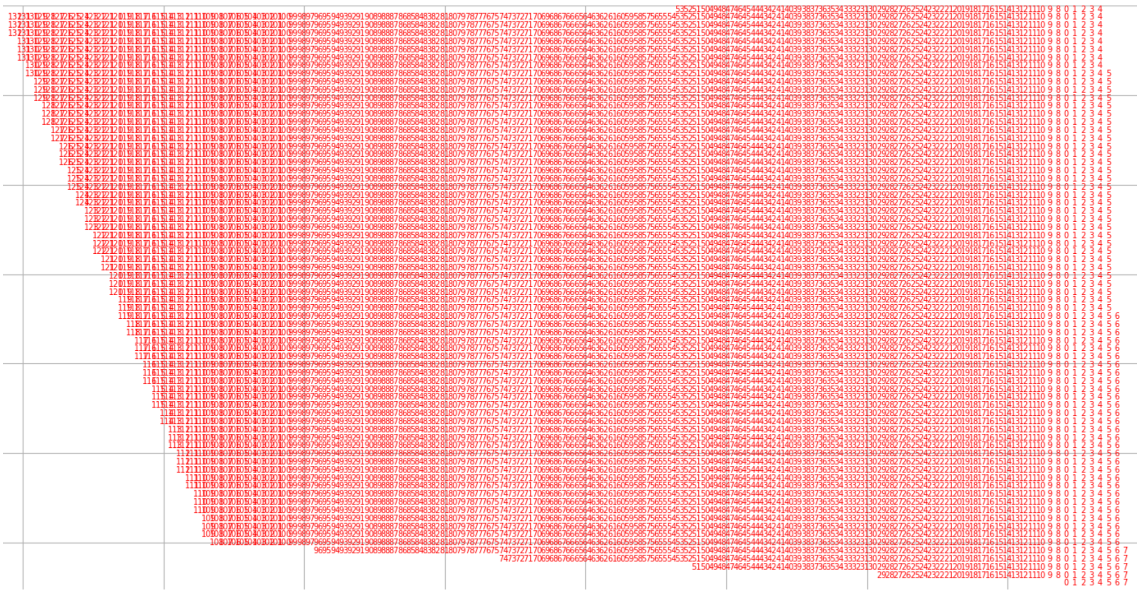

Figure A.4: Computed indexed grid for reflist j.

A.2 MA-Cycling Results

Parameter	MaxTime [ms]	MinTime [ms]	Values bigger than 0.3 [ms]	Values lower than 0.3 [ms]
Case 1 Result	0.328	0.237	27	1479
Case 2 Result	0.332	0.233	18	1482
Case 3 Result	0.329	0.227	24	2222

Table A.1: Time performance of the "Search Center" model proposed, 3.4.2.2,the time values are categorized by threshold (0.3 ms).

Parameter	Max T [ms]	Min T [ms]	Cycles with T bigger than 1000 [ms]	Cycles with T equal to 1000 [ms]
Case 1 Result	31000	1000	290	1216
Case 2 Result	46000	1000	278	1222
Case 3 Result	26000	1000	603	1643

Table A.2: Period of the "Cycling" model proposed, 3.11,the time values are categorized by threshold (1000ms).

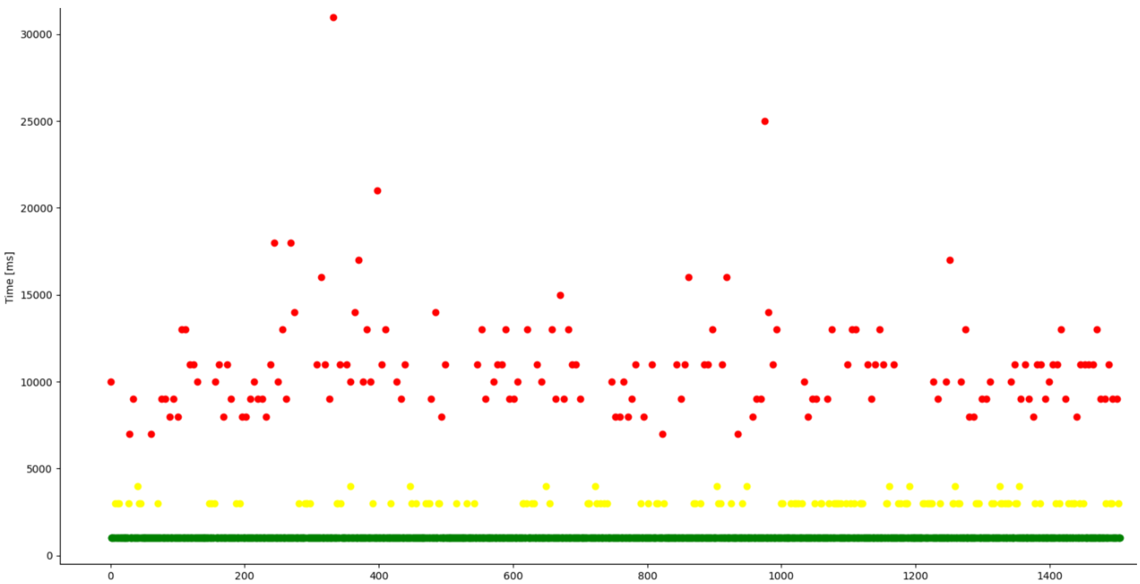

Figure A.5: MobiAlert period, T. Case Study Nº1.

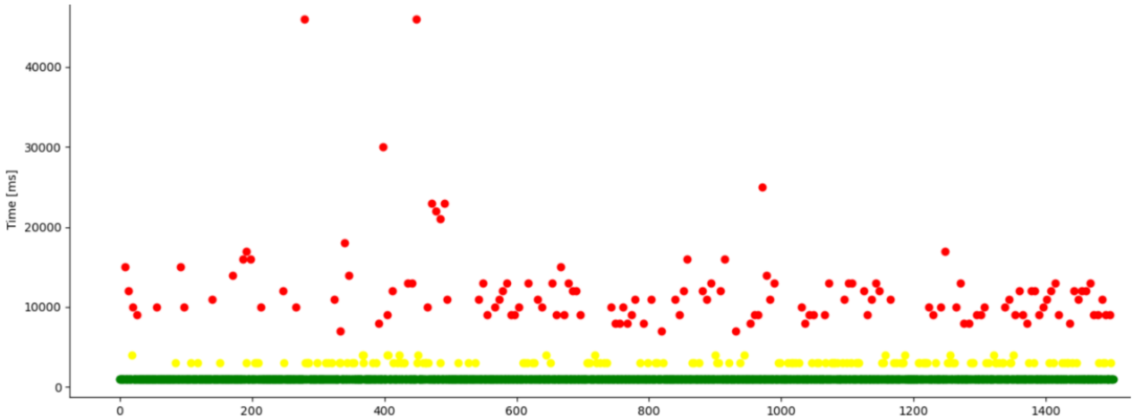

Figure A.6: MobiAlert period, T. Case Study Nº2.

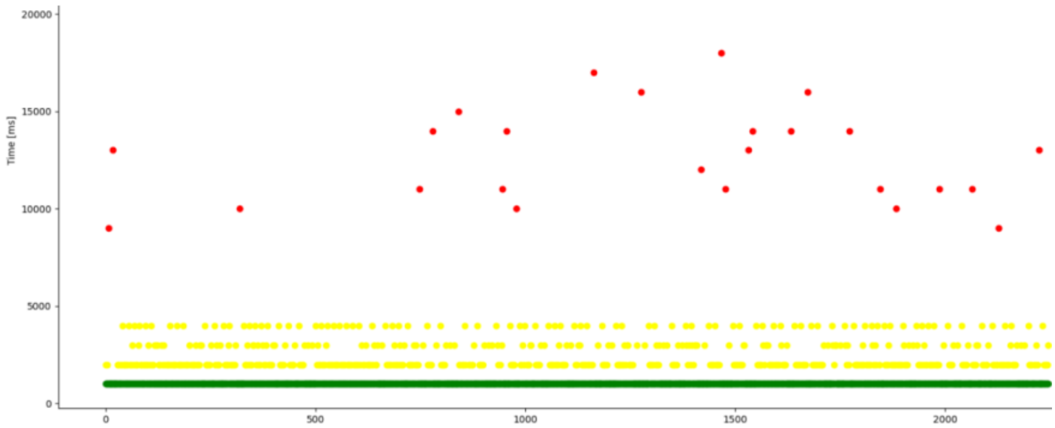

Figure A.7: MobiAlert period, T. Case Study Nº3.

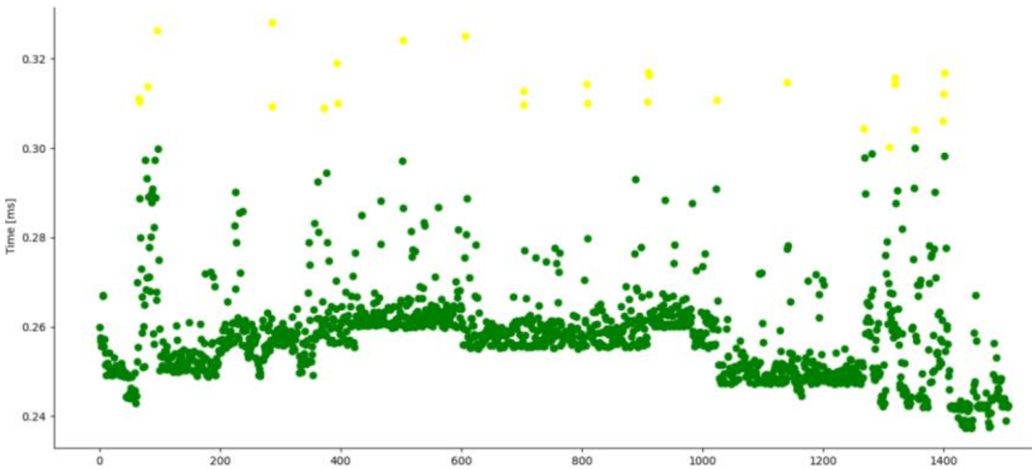

Figure A.8: Times for "Search Center" Model. Case Study Nº1.

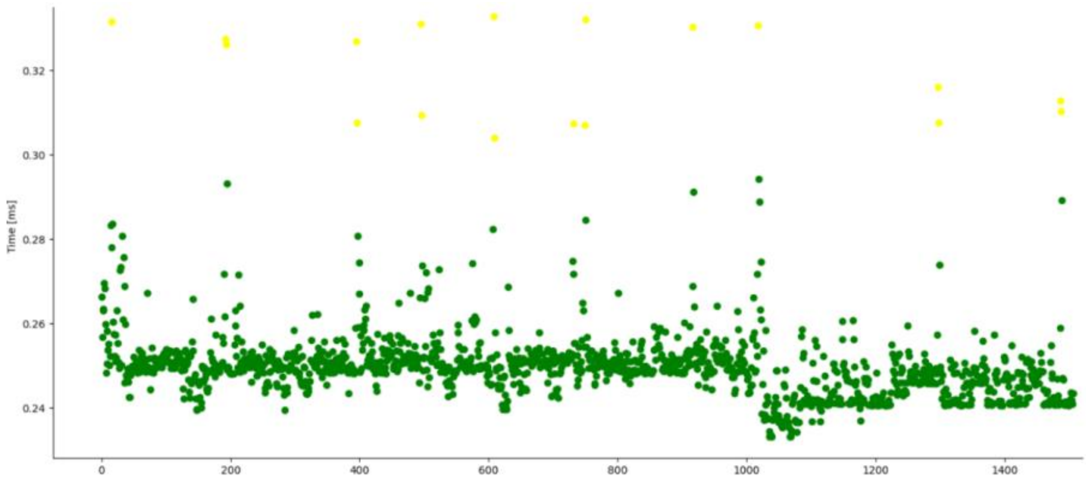

Figure A.9: Times for “Search Center” Model. Case Study Nº2.

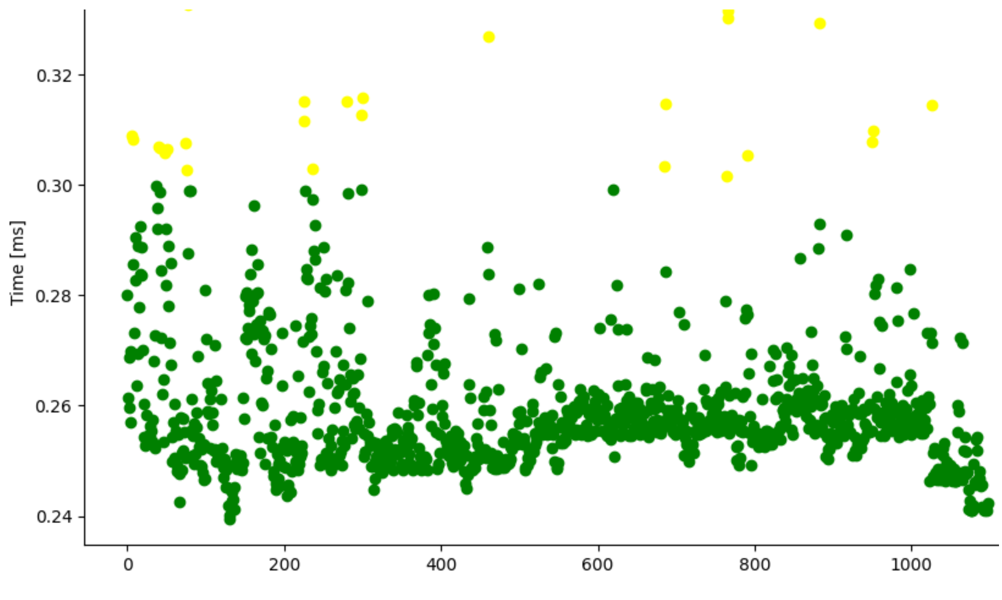

Figure A.10: Times for “Search Center” Model. Case Study Nº3.

References

Lorant A. Tavasszy, Behzad Behdani, and Rob Konings. Intermodality and synchromodality. *SSRN*, 2015. doi: https://dx.doi.org/10.2139/ssrn.2592888.

Ajmain Inqiad Alam, Mahfuzur Rahman, Sharmin Afroz, Mahbubul Alam, Jia Uddin, and Md. Ashraful Alam. Iot enabled smart bicycle safety system. *IEEE*, 2018. doi: 10.1109/ICIEV.2018.8641067.

Ruth Ande, Bamidele Adebisi, Mohammad Hammoudeh, and Jibran Saleem. Internet of things: Evolution and technologies from a security perspective. *Sustainable Cities and Society*, 2020. doi: https://doi.org/10.1016/j.scs.2019.101728.

Josh Andres, Juerg von Kaenel, Florian Mueller, and Tuomas Kari. Co-riding with my ebike to get green lights. 2019. doi: 10.1145/3322276.3322307.

Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 2010. doi: 10.1016/j.comnet.2010.05.010.

Danica Babić, Milica Kalić, Milan Janić, Slavica Dožić, and Katarina Kukić. Integrated door-to-door transport services for air passengers: From intermodality to multimodality. *Sustainability*, 2022. doi: 10.3390/su14116503.

Pawel Banys, Evelin Engler, Frank Heymann, and Thoralf Noack. Timestamp discrepancies in multisensor nmea environment during survey voyage. 2013. URL https://api.semanticscholar.org/CorpusID:46901107.

Ashish Batra and Pankaj Chhabra. Smart cities: An empirical study of definitions. *ShodhKosh: Journal of Visual and Performing Arts*, 2023. doi: 10.29121/shodhkosh.v4.i1.2023.357.

Gurinder Bawa, Zamen Latef Naser, Hayder Sharif, Laith H. A. Fezaa, Jasgurpreet Singh Chohan, and Ruby Pant. Development of safety monitoring and alert system for 2 wheelers. 2024. doi: 10.1109/ICRITO61523.2024.10522447.

Bogachan Bayulken and Donald Huisingh. A literature review of historical trends and emerging theoretical approaches for developing sustainable cities (part 1). *Journal of Cleaner Production*, 2015. doi: https://doi.org/10.1016/j.jclepro.2014.12.100.

C. Bell. Beginning iot projects: Breadboard-less electronic projects. 2021.

Simon Bibri and John Krogstie. Smart sustainable cities of the future: An extensive interdisciplinary literature review. *Sustainable Cities and Society*, 2017. doi: 10.1016/j.scs.2017.02.016.

Simon Elias Bibri. Data-driven smart sustainable cities of the future: An evidence synthesis approach to a comprehensive state-of-the-art literature review. *Sustainable Futures*, 2021. doi: https://doi.org/10.1016/j.sftr.2021.100047.

Can Bıyık, Ahmad Abareshi, Alexander Paz, Rosa Arce Ruiz, Rosaria Battarra, Christopher D. F. Rogers, and Carmen Lizarraga. Smart mobility adoption: A review of the literature. *Journal of Open Innovation: Technology, Market, and Complexity*, 2021a. doi: 10.3390/joitmc7020146.

Can Bıyık, Ahmad Abareshi, Alexander Paz, Rosa Arce Ruiz, Rosaria Battarra, Christopher D.F. Rogers, and Carmen Lizarraga. Smart mobility adoption: A review of the literature. *Journal of Open Innovation: Technology, Market, and Complexity*, 2021b. doi: 10.3390/joitmc7020146.

James Canton. The extreme future of megacities. *Significance*, 2011.

Carlos Carvalho and Rui José. The gap between cycling practices and mapping services for smart cycling. *IEEE*, 2022. doi: 10.1109/ISC255366.2022.9922089.

Daniel G. Costa and Cristian Duran-Faundez. Open-source electronics platforms as enabling technologies for smart cities: Recent developments and perspectives, 2018.

Jifeng Cui, Yong Zhang, Yubin Yang, Chao Li, and Chunxiao Xing. A dynamic pyramid rtree index for massive geospatial files. *2014 11th Web Information System and Application Conference*, 2014. doi: https://10.1109/WISA.2014.40.

Haviluddin E Maria, E Budiman and M Taruk. Measure distance locating nearest public facilities using haversine and euclidean methods. *IOP Publishing Ltd*, 2020. doi: https://10.1088/1742-6596/1450/1/012080.

elettronica Federazione italiana di elettrotecnica, International Microelectronics, and Packaging Society. *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE) : Turin (Italy), July 2-4, 2019*. 2019.

Ricardo Faria, Lina Brito, Karolina Baras, and José Silva. Smart mobility: A survey. *International Conference on Internet of Things for the Global Community (IoTGC)*, 2017. doi: 10.1109/IoTGC.2017.8008972.

Raspberry Pi Foundation. Raspberry pi, 2006. URL https://www.raspberrypi.com/.

Jose Sanchez Gracias, Gregory S. Parnell, Eric Specking, Edward A. Pohl, and Randy Buchanan. Smart cities—a structured literature review. *Smart Cities*, 2023. doi: 10.3390/smartcities6040080.

Stefan Huber, Paul Lindemann, and Bettina Schröter. Understanding the influence of accident risk and perceived safety on bicycle route choice. *Transportation Research Procedia*, 2023. doi: https://doi.org/10.1016/j.trpro.2023.11.416.

Elvira Ismagilova, Laurie Hughes, Yogesh K. Dwivedi, and K. Ravi Raman. Smart cities: Advances in research—an information systems perspective. *International Journal of Information Management*, 2019. doi: https://doi.org/10.1016/j.ijinfomgt.2019.01.004.

Georgios Kapousizis, Mehmet Baran Ulak, Karst Geurs, and Paul J. M Havinga. A review of state-of-the-art bicycle technologies affecting cycling safety: level of smartness and technology readiness. *Transport Reviews*, 2023. doi: 10.1080/01441647.2022.2122625.

S. Karthikeyan, R. Aakash Raj, Meenalosini Vimal Cruz, Lei Chen, J. L. Ajay Vishal, and V. S. Rohith. A systematic analysis on raspberry pi prototyping: Uses, challenges, benefits, and drawbacks. *IEEE Internet of Things Journal*, 2023. doi: 10.1109/JIOT.2023.3262942.

Aqeel Kazmi, Saqib Rasool Chaudhry, Hadi Tabatabaee Malazi, Sangita Dhara, and Siobhan Clarke. Interoperable internet of things for smart transportation systems in circular cities. *Computer*, 2022. doi: 10.1109/MC.2022.3202298.

Levente J Klein, Fernando J Marianno, Conrad M Albrecht, Marcus Freitag, Siyuan Lu, Nigel Hinds, Xiaoyan Shao, Sergio Bermudez Rodriguez, and Hendrik F Hamann. Pairs: A scalable geo-spatial data analytics platform. *2015 IEEE International Conference on Big Data (Big Data)*, 2015. doi: 10.1109/BigData.2015.7363884.

Hari Kishan Kondaveeti, Nandeesh Kumar Kumaravelu, Sunny Dayal Vanambathina, Sudha Ellison Mathe, and Suseela Vappangi. A systematic literature review on prototyping with arduino: Applications, challenges, advantages, and limitations. *Computer Science Review*, 2021. doi: https://doi.org/10.1016/j.cosrev.2021.100364.

Young Hoon Kwak and Jaehyun Lee. Toward sustainable smart city: Lessons from 20 years of korean programs. *IEEE Transactions on Engineering Management*, 2023. doi: 10.1109/TEM.2021.3060956.

Jae-Gil Lee and Minseo Kang. Geospatial big data: Challenges and opportunities. *Big Data Research*, 2015. doi: https://doi.org/10.1016/j.bdr.2015.01.003.

Yi Lei, Xiaochong Tong, Dali Wang, Chunping Qiu, He Li, and Youwei Zhang. W-hilbert: A w-shaped hilbert curve and coding method for multiscale geospatial data index. *International Journal of Applied Earth Observation and Geoinformation*, 2023. doi: 10.1016/j.jag.2023.103298.

Bing Li. Effective energy utilization through economic development for sustainable management in smart cities. *Energy Reports*, 2022. doi: https://doi.org/10.1016/j.egyr.2022.02.303.

Luis F. Luna-Reyes, Jing Zhang, Sehl Mellouli, Association for Computing Machinery, and ACM Digital Library. *14th Annual International Digital Government Research Conference : from E-Government to Smart Government*. 2012.

Siddharth Mahobia and Rahul Pawar. Comprehensive review of smart cities using iot in cloud. *International Journal of Innovative Research in Computer and Communication Engineering*, 2024. doi: 10.15680/IJIRCCE.2024.1205194.

Irina Makarova, Aleksey Boyko, Anton Pashkevich, and Eduard Tsybunov. Development of a monitoring system on environment and human health to control a smart bike. 2020. doi: 10.1109/ICCC49264.2020.9257239.

Sudha Ellison Mathe, Hari Kishan Kondaveeti, Suseela Vappangi, Sunny Dayal Vanambathina, and Nandeesh Kumar Kumaravelu. A comprehensive review on applications of raspberry pi. *Computer Science Review*, 2024. doi: https://doi.org/10.1016/j.cosrev.2024.100636.

Rishika Mehta, Jyoti Sahni, and Kavita Khanna. Internet of things: Vision, applications and challenges. *Procedia Computer Science*, 2018. doi: 10.1016/j.procs.2018.05.042.

Siyuan Meng and Hao Zheng. A personalized bikeability-based cycling route recommendation method with machine learning. *International Journal of Applied Earth Observation and Geoinformation*, 2023. doi: https://doi.org/10.1016/j.jag.2023.103373.

Douglas Mitieka, Rose Luke, Hossana Twinomurinzi, and Joash Mageto. Smart mobility in urban areas: A bibliometric review and research agenda. *Sustainability (Switzerland)*, 2023. doi: 10.3390/su15086754.

Saraju Mohanty. Everything you wanted to know about smart cities. *IEEE Consumer Electronics Magazine*, 2016. doi: 10.1109/MCE.2016.2556879.

Saraju P. Mohanty, Uma Choppali, and Elias Kougianos. Everything you wanted to know about smart cities: The internet of things is the backbone. *IEEE Consumer Electronics Magazine*, 2016. doi: 10.1109/MCE.2016.2556879.

Wan Norsyafizan W. Muhamad, Sayyidul Ainulfadhily bin Razali, Norfishah Ab Wahab, Meor Mohd Azreen, Suzi Seroja Sarnin, and Nani Fadzlina Naim. Smart bike monitoring system for cyclist via internet of things (iot). *IEEE 5th International Symposium on Telecommunication Technologies (ISTT)*, 2020. doi: 10.1109/ISTT50966.2020.9279377.

Sri Banu Munisamy, Roziyani Rawi, Hakim bin Mohd Yasim, Nur Farahwahida Ab Aziz, Sayed Aziz Sayed Hussin, and Norilawati Binti Md Jali. Enhancing cyclist safety: Real-time fallen cyclist detection and emergency notification system. 2024. doi: 10.1109/IMCOM60618.2024.10418329.

Boushra Naim and Mary Felix. Cycling safety problems in urban context. *APJ: Cycling Safety Problems in Urban Context*, 2022.

European Road Safety Observatory. Facts and figures – cyclists - 2021, Outuro 2021. URL https://road-safety.transport.ec.europa.eu/system/files/2022-03/FF_cyclists_20220209.pdf.

Franklin Oliveira, Daniel G. Costa, Luciana Lima, and Ivanovitch Silva. Ibikesafe: A multi-parameter system for monitoring, evaluation and visualization of cycling paths in smart cities targeted at cycling adverse conditions. *Smart Cities*, 2021a. doi: 10.3390/smartcities4030056.

Franklin Oliveira, Dilan Nery, Daniel G. Costa, Ivanovitch Silva, and Luciana Lima. A survey of technologies and recent developments for sustainable smart cycling. *Sustainability (Switzerland)*, 2021b. doi: 10.3390/su13063422.

Aleksander Orlowski and Patrycja Romanowska. Smart cities concept: Smart mobility indicator. *Cybernetics and Systems*, 2019. doi: 10.1080/01969722.2019.1565120.

Vítor de Castro Paes, Clinton Hudson Moreira Pessoa, Rodrigo Pereira Pagliusi, Carlos Eduardo Barbosa, Matheus Argôlo, Yuri Oliveira de Lima, Herbert Salazar, Alan Lyra, and Jano Moreira de Souza. Analyzing the challenges for future smart and sustainable cities. *Sustainability*, 2023. doi: 10.3390/su15107996.

João Paulo Just Peixoto, Daniel G. Costa, Washington de J.S.da Franca Rocha, Paulo Portugal, and Francisco Vasques. Cityzones: A geospatial multi-tier software tool to compute urban risk zones. *SoftwareX*, 2023. doi: 10.1016/j.softx.2023.101409.

João Paulo Just Peixoto and Daniel G. Costa. Wireless visual sensor networks for smart city applications:a relevance-based approach for multiple sinks mobility. *Future Generation Computer Systems*, 2017. doi: https://doi.org/10.1016/j.future.2017.05.027.

Mousi Perumal, Bhuvaneswari Velumani, Ananthi Sadhasivam, and Kalpana Ramasami. Spatial data mining approaches for gis – a brief review. *Advances in Intelligent Systems and Computing*, 2015. doi: 10.1007/978-3-319-13731-5_63.

Ryszard F. Sadowski. Remarks on the concept of a sustainable city in light of konrad lorenz's concept of man. *Sustainability*, 2023. doi: 10.3390/su15032443.

Iqbal H. Sarker. Smart city data science: Towards data-driven smart cities with open research issues. *Internet of Things*, 2022. doi: https://doi.org/10.1016/j.iot.2022.100528.

Yo Sasaki, Katsuya Fujiwara, and Kazutaka Mitobe. Risks that induce bicycle accidents: Measurement and analysis of bicyclist behavior while going straight and turning right using a bicycle simulator. *Accident Analysis Prevention*, 2024. doi: https://doi.org/10.1016/j.aap.2023.107338.

Ayyoob Sharifi and Pourya Salehi. *Resilient Smart Cities*. 2022. doi: 10.1007/978-3-030-95037-8.

Yogesh Simmhan, Pushkara Ravindra, Shilpa Chaturvedi, Malati Hegde, and Rashmi Ballamajalu. Towards a data-driven iot software architecture for smart city utilities. 2018. doi: 10.1002/spe.2580.

Erik Stokstad. Sustainable goals from united nations under fire. *Science*, 2015. doi: 10.1126/science.347.6223.702.

Abbas Shah Syed, Daniel Sierra-Sosa, Anup Kumar, and Adel Elmaghraby. Iot in smart cities: A survey of technologies, practices and challenges. *Smart Cities*, 2021. doi: 10.3390/smartcities4020024.

Jayashree Tamkhade, Jay Daflapurkar, Safa Kagad, Harsh Kalokhe, and Amey Agrawal. Design and development of controlling and monitoring bicycle speed remotely. *IEEE*, 2024. doi: 10.1109/I2CT61223.2024.10544355.

M. Theresa, P. Rajam, A. Benila, R. Priyadharshini, and J. Jacob. An intelligent bike safety and surveillance system. *AIP Conference Proceedings*, 2022. doi: 10.1063/5.0074199.

Sivan Toledo. Ssh tunneling to connect to remote computers. *Software impacts*, 2023. doi: https://doi.org/10.1016/j.simpa.2023.100545.

Department of Economic United Nations and Social Affairs. World urbanization prospects: The 2018 revision (st/esa/ser.a/420). Technical report, New York: United Nations., 2019.

Pedro Vieira, João P. Costeira, Susana Brandão, and Manuel Marques. Smartcycling: Assessing cyclists' driving experience. *IEEE Intelligent Vehicles Symposium (IV)*, 2016. doi: 10.1109/IVS.2016.7535561.

Katie Williams. Sustainable cities: research and practice challenges. *International Journal of Urban Sustainable Development*, 2009. doi: 10.1080/19463131003654863.

LEI Yi, TONG Xiaochong, LAI Guangling, and FAN Shuaibo. An efficient index for global massive remote sensing data. *2018 International Workshop on Big Geospatial Data and Data Science*, 2018. doi: https://10.1109/BGDDS.2018.8626834.

Chang-le Zhong, Zhen Zhu, and Ren-Gen Huang. Study on the iot architecture and access technology. 2017. doi: 10.1109/DCABES.2017.32.