
UNIVERSIDADE D
COIMBRA

Rafael Rainho Luís

# URSO VERDE

## PRIVACY, SECURITY AND MANAGEMENT FOR A SMART CITY PLATFORM

**Dissertation submitted to the Department of Electrical and Computer Engineering of the Faculty of Sciences and Technology of the University of Coimbra in partial fulfillment of the requirements for the Degree of Master of Science in Electrical and Computer Engineering oriented by Professor Doctor Jorge Miguel Sá Silva**

July 2024

1 2  9 0

UNIVERSIDADE D
COIMBRA

# URSO VERDE

## PRIVACY, SECURITY AND MANAGEMENT FOR A SMART CITY PLATFORM

**Rafael Rainho Luís**

Coimbra, July 2024

1 2  9 0

UNIVERSIDADE D
COIMBRA

# URSO VERDE

## PRIVACY, SECURITY AND MANAGEMENT FOR A SMART CITY PLATFORM


**Supervisor:**

Professor Jorge Miguel Sá Silva


**Jury:**

Professor João Pedro de Almeida Barreto

Professor Paulo José Monteiro Peixoto

Professor Jorge Miguel Sá Silva

Dissertation submitted in partial fulfillment for the degree of Master of Science in Electrical and Computer Engineering.

Coimbra, July 2024

The work presented in this dissertation was developed in collaboration with:

**University of Coimbra**


**Municipality of Coimbra**


**Department of Electrical and Computer Engineering**

# Acknowledgements

I would like to express my sincere gratitude to all those who made today possible.

To my family and friends for the incredible support and strength.

To Câmara Municipal de Coimbra, Coimbra City Lab and all the involved for the opportunity given to work in such an inspiring project.

To the university, the department and the city of Coimbra for being my home for the past five years.

To the all non-mentioned, you were not forgotten, thank you for the journey.

# Abstract

**Green Bear (*Urso Verde*) is a platform designed to transform Coimbra into a smart and digital city, encouraging users through rewards for sustainable actions that promote local commerce. However, it lacked a server to act as an intermediary between the user and the database, as well as the development of a dashboard capable of efficiently managing the entire system.**

**This study aims to address these gaps by developing a server and a dashboard. The server, designed to isolate the database, filter traffic, and standardise data, will be crucial for establishing effective communication with both the dashboard and users. In turn, the dashboard will enable a system administrator to define rewards and objectives for users, validate reward transactions, and analyse detailed usage statistics of the platform in real time. This technological advancement will not only optimise Green Bear's operation but also significantly contribute to its long-term effectiveness and sustainability.**

**This study also aimed to implement the project in the real world through a pilot project conducted at *Escola Secundária Jaime Cortesão*. Here, the functionalities and usability of the platform were tested, while essential feedback was collected for refinement before the official launch.**

**This feedback was evaluated through surveys, and the overall results were highly elucidative, demonstrating the viability of the platform and its potential positive impact on the community.**

***Keywords***— Smart city, Intermediary server, Dashboard

# Resumo

**O Urso Verde (*Green Bear*) é uma plataforma concebida com o intuito de transformar Coimbra numa cidade inteligente e digital, incentivando os utilizadores através de prémios por ações sustentáveis que promovam o comércio local. No entanto, carecia de um servidor que desempenhasse o papel de intermediário entre o utilizador e a base de dados, bem como de desenvolver um painel de controlo capaz de gerir todo o sistema de forma eficiente.**

**Este estudo tem como objetivo principal a resolução destas lacunas, através do desenvolvimento de um servidor e de um painel de controlo. O servidor, destinado a isolar a base de dados, filtrar o tráfego e padronizar os dados, será fundamental para estabelecer uma comunicação eficaz tanto com o painel de controlo como com os utilizadores. Por sua vez, o painel de controlo permitirá que um administrador do sistema defina prémios e objetivos para os utilizadores, valide transações de prémios e analise estatísticas detalhadas da utilização da plataforma em tempo real. Este avanço tecnológico não só otimizará a operação do Urso Verde, mas também contribuirá significativamente para a sua eficácia e sustentabilidade a longo prazo.**

**Este estudo visou também a implementação do projeto no mundo real através de um projeto piloto, conduzido na Escola Secundária Jaime Cortesão. Aqui, foram testadas as funcionalidades e a usabilidade da plataforma, enquanto se recolheram *feedbacks* essenciais para o seu aprimoramento antes do lançamento oficial.**

**Estes *feedbacks* foram avaliados através de inquéritos e os resultados globais foram amplamente elucidativos, demonstrando a viabilidade e da plataforma e o seu potencial impacto positivo na comunidade.**

***Palavras chave***— Cidade inteligente, Servidor intermédio, Painel de controlo

*"I have fought the good fight, I have finished the race, I have kept the faith."*

*— 2 Timothy 4:7*

# Contents

# List of Acronyms

| | |
|---|---|
| **API** | Application Programming Interface |
| **BLE** | Bluetooth Low Energy |
| **CCL** | Coimbra City Lab |
| **CMC** | Câmara Municipal de Coimbra |
| **CSSE** | Centre for Systems Science and Engineering |
| **DEEC** | Departamento de Engenharia Eletrotécnica e de Computadores |
| **DFR** | Dashboard Functional Requirements |
| **DNFR** | Dashboard Non-Functional Requirements |
| **DOM** | Document Object Model |
| **ESJC** | Escola Secundária Jaime Cortesão |
| **HTTPS** | Hypertext Transfer Protocol Secure |
| **HTTP** | Hypertext Transfer Protocol |
| **ID** | Identification |
| **IoT** | Internet of Things |
| **JSON** | Javascript Object Notation |
| **LoRaWAN** | Long Range Wide Area Network |
| **NFC** | Near Field Communication |
| **NoSQL** | Not Only Structured Query Language |
| **OAuth** | Web Authorization Protocol |
| **QR** | Quick Response |

**SFR** Server Functional Requirements

**TLS** Transport Layer Security

# List of Equations



# List of Figures

# List of Listings



# List of Tables

# 1

# Introduction

This chapter serves as an introduction to the entire document. It starts by explaining the purpose and motivation for this dissertation and then presents the state of the art with a summary of the literature. The chapter concludes by revealing the structure of the remainder document.

## 1.1 Motivation

Urso Verde presented an opportunity to apply the theoretical knowledge acquired in the course to a practical solution for smart cities. The challenge of stepping out of my comfort zone, exposing myself to real-life scenarios different from the controlled environments of academia, was very motivating.

Additionally, the project offered a high degree of autonomy, allowing a sense of ownership over the work by making important independent decisions while meeting deadlines and ensuring the project's success. This pressure, along with the experience of working in a team, provided an opportunity to develop important skills such as teamwork, communication, and problem-solving, contributing to my growth as a person and as a professional.

Collaborating directly with several institutions would offer the valuable opportunity to experience different dynamics, expectations, work cultures and areas of expertise.

Finally, the opportunity to be part of something for the community that contributes to a greener and more sustainable future was immensely motivating.

## 1.2 Purpose

The project aims to promote sustainable habits among the citizens of Coimbra. *Urso Verde* aims to incentives citizens willing to participate to engage in social activities tied to *Câmara Municipal de Coimbra*(CMC), such as blood donations, recycling, and the use of public transportation, as well as activities with other partners such as *Mercado Municipal D.Pedro V*, schools, or local commerce, by

rewarding their efforts with virtual points tied to a category that can be later exchanged for prizes of the same category. For example, borrowing a book from the public library will generate a certain amount of points to a wallet associated with the library, which can be traded for prizes provided by the CMC structures or partners associated with the same wallet. Therefore, the project can be considered a social and community project that requires the creation, adjustment, and ongoing work on a management dashboard to orient, track, and organise the live project and a server supporting the mobile app and the management dashboard. For this first attempt to test the idea, a pilot project was conducted to gather results, feedback, and other important information. A small group of participants was chosen from a class of students at *Escola Secundária Jaime Cortesão* (ESJC) as the control group. The field study focused on understanding how interested the students were in using the mobile app, how easy it was to use, and how motivating the application and the prizes were.

## 1.3 Related Work

### 1.3.1 Smart Cities

Cortés-Cediel et al. [6] analysed European smart cities, revealing that these cities emphasise social initiatives and open participation models, prioritising social well-being and democratic engagement. Information and communication technologies, such as mobile apps, Internet of Things (IoT) devices, and open data, are key tools for citizen engagement, complemented by offline methods like meetings and cultural events.

Dunayev et al. [9] concluded that a human-centric approach prioritises specific public services that meet the citizens' needs. The involvement of the community, local authorities and businesses is crucial to ensure coordination in active citizenship. Moreover, the importance of leveraging digital technologies such as IoT, big data, and artificial intelligence to enhance service delivery and citizen engagement within smart cities was highlighted.

### 1.3.2 Fraud detection systems

The literature presents studies on fraud detection across several areas such as credit card transactions [30] or e-voting [24] but they generally agree that the first layer of protection are prevention systems whose primary purpose is to prevent fraud from occurring in the first place.[3] Prevention against fraud can take multiple forms such as control, removal, suppression, destruction, restriction. Behdad et al.[3] explored different types of electronic fraud including email spam, phishing and network intrusions.

In the smart cities context there are three assets that need to be secured and protected: people, resources and services.[31] Attacks can be of two types: aimed to disrupt the lives of city residents or to disrupt the business operations in the city. Toh [31] focused his study on protection of different important assets related to smart cities such as connection or data, including encryption protecting the data content, authentication for the source, signatures that confirm the validity of the data, and privacy to disassociate data from the identity and location of the user.

The conclusions are that there are some challenges for governments and industries due to the emerging concerns with security in systems, devices, applications, providers, and users in the smart cities context.

### 1.3.3 Dashboards

Dashboards have emerged as essential tools for system configuration, data visualisation, analysis, and interpretation for individuals, organisations, governments, and companies, allowing users to make important and informed decisions by monitoring data trends, performance indicators, and other metrics. Wexler et al. [33] describe a dashboard as “a visual display of data used to monitor conditions and/or facilitate understanding”. Sarikaya et al. state that "the dashboard concept has evolved from single-view reporting screens to include interactive interfaces with multiple views and purposes, including communication, learning, and motivation, in addition to the classic notions of monitoring and decision support" and their study, conducting an extensive review of existing solutions and their characterisation reflects the differences between dashboards, highlighting the custom needs in terms of indicators, designs, components and goals [29]. The critical role of data visualisation has been increasingly recognised in recent years. A particularly notable example is the Johns Hopkins COVID-19 Dashboard[17] (figure 1.1), developed by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. This dashboard gathered attention and acclaim for its significance, innovation, and impact in providing real-time insights and updates on the progression of the COVID-19 pandemic. In recognition of its contribution, the CSSE was awarded the Making a Difference Award at the 2020 Esri User Conference [19].

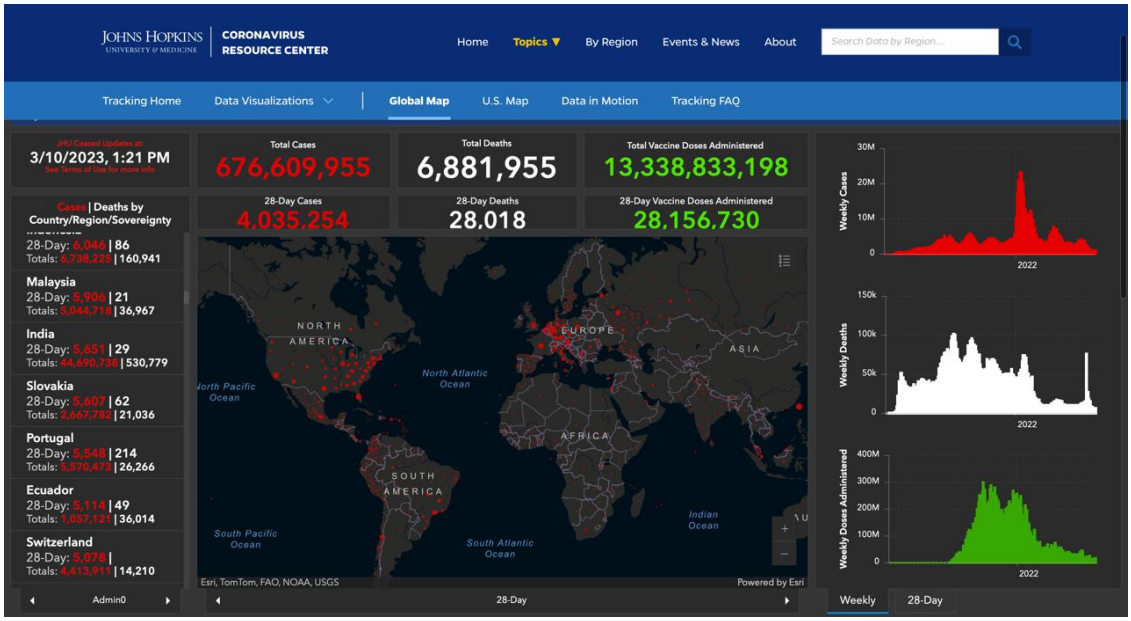



Figure 1.1: John Hopkins COVID-19 dashboard [17]

### 1.3.4 Servers for authentication and application programming interface purposes

Gammeter et al. presented a system for mobile augmented reality based on visual recognition [12]. The proposed architecture illustrated in figure 1.2 delegates recognition tasks to a server and executes tracking operations thereby improving overall operational speed by dividing object recognition and tracking between the server and the client.

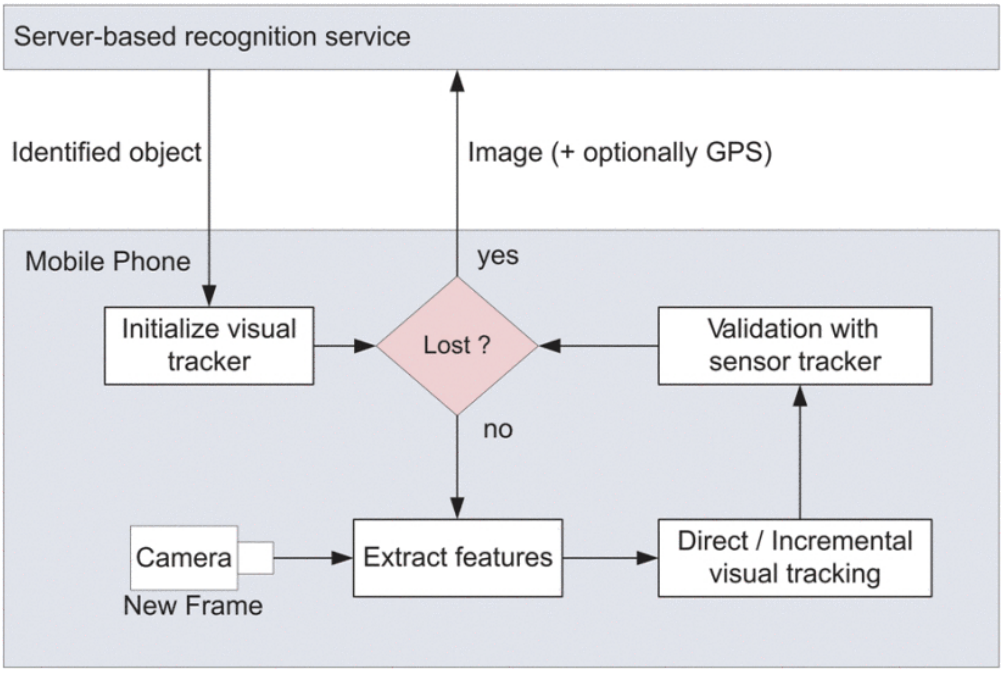



Figure 1.2: Proposed architecture [12]

Several studies indicate the advantages of an intermediary Application Programming Interface (API) when using mobile applications.

AL-atraqchi [4] proposed a model to connect a mobile application to a server through an API server.

A conceptual framework for a smart city middle-ware was framed by Goumopoulos [14]. This framework was based on twenty different middle-ware solutions that answered to questions related to architectural styles and programming paradigms. This framework outlines the key features necessary for developing a robust middle-ware platform that supports the creation of integrated and scalable smart city applications. It acts as a cornerstone in urban digitisation, providing an intermediary software layer that streamlines the development, deployment, and management of applications designed for smart urban environments.

## 1.4 Previous Work

Urso Verde was initiated in 2022. The aim of first version was to "explore and demonstrate the use of IoT, Long Range Wide Area Network (LoRaWAN) public networks, and HiTL technologies in a sustainable city environment." [27]. Leveraging Bluetooth Low Energy (BLE) technologies, the system encouraged the users to engage in outdoor activities (spending time in green areas) using bicycle paths and recycling.

An Android mobile application (figure 1.3) in the user's phone periodically sends a BLE beacon advertisement. The physical places where these activities took place were equipped with "Green Bear nodes", a set of sensors programmed to receive nearby beacons. When capturing one of these beacons, the information that the user performed a certain activity is sent over LoRaWAN protocols to a public gateway that later sends the information to a database over Message Queuing Telemetry Transport (figure 1.4).

This first version (figure 1.5) also featured a simple dashboard capable of listing the active users and their statistics, (figure 1.5a) as well as a general view of public gateways (figure 1.5b).

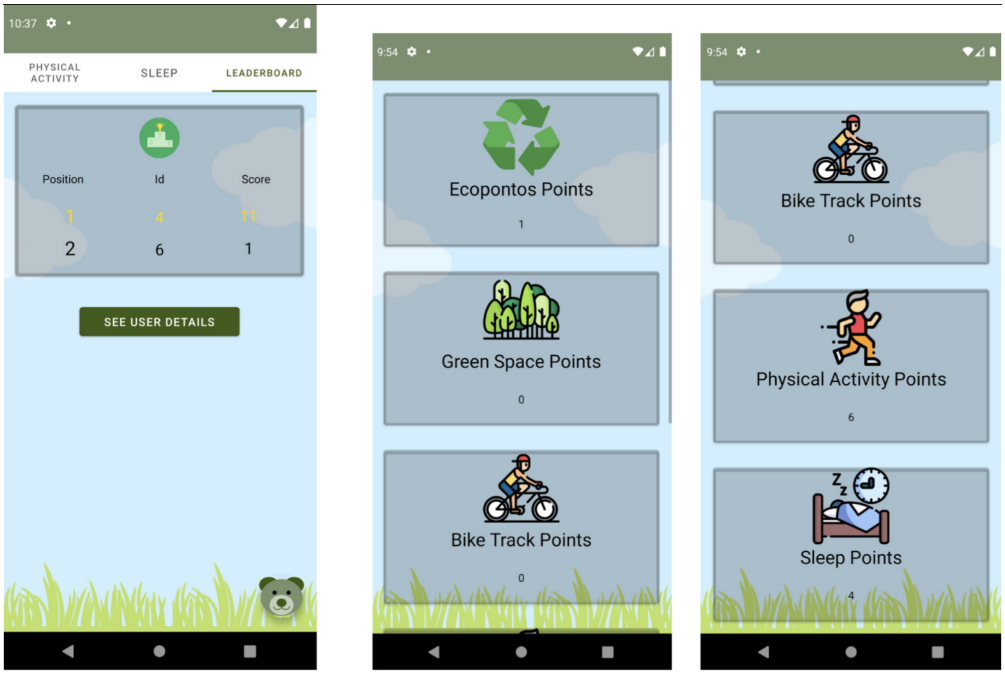



Figure 1.3: First version of the mobile app [27]

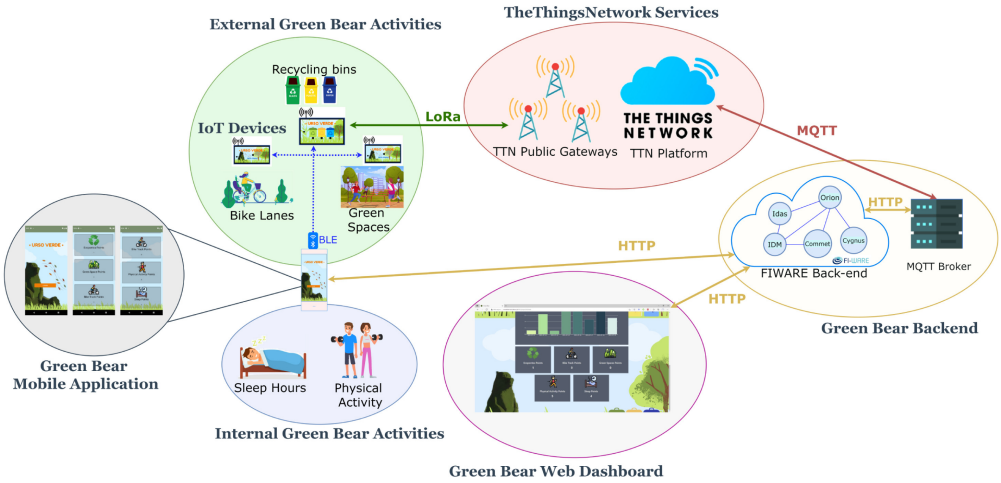
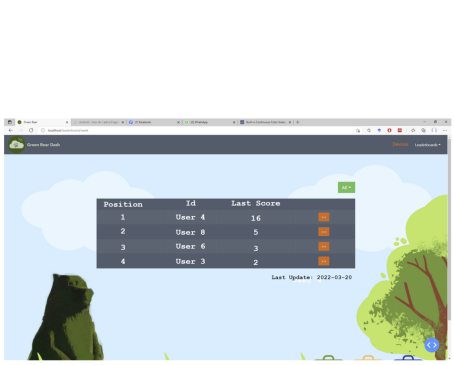
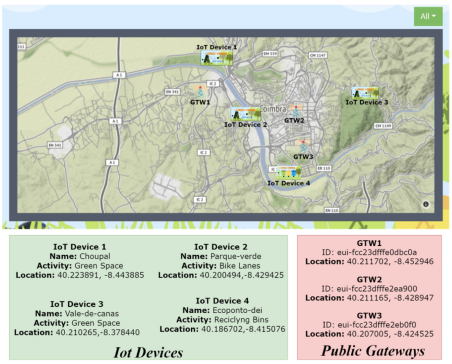



Figure 1.4: Architecture of the first version of Urso Verde [27]



(a) Dashboard - Active Users [27]




(b) Dashboard - Public gateways [27]

Figure 1.5: First version of the dashboard

This first version successfully implemented a way to track users' activities, however it had some issues. The application lacked a user authentication system, assigned a random unique identification (ID) to the users, and was not very engaging as it only collected information on the users' activities.

The second version of Urso Verde emerged to address the aforementioned issues. António Moisés Dias [7] migrated the Android application to a cross-platform framework to feature iOS devices, introduced the point reward system, the Quick Response (QR) reading system and generally changed the user interface, creating a logotype and improving the general design of the mobile application. However, the non-existence of an intermediary server or a dashboard required all the information to be hard-coded between the mobile application and the database (figure 1.6).This approach is inefficient and difficult to implement for a production version. Additionally, concerns related to laws and regulations (including the involvement of under-aged children in studies) necessitated revising, often changing, or improving some ideas and procedures, while still maintaining the initial idea of rewarding citizens for their civic behaviours.

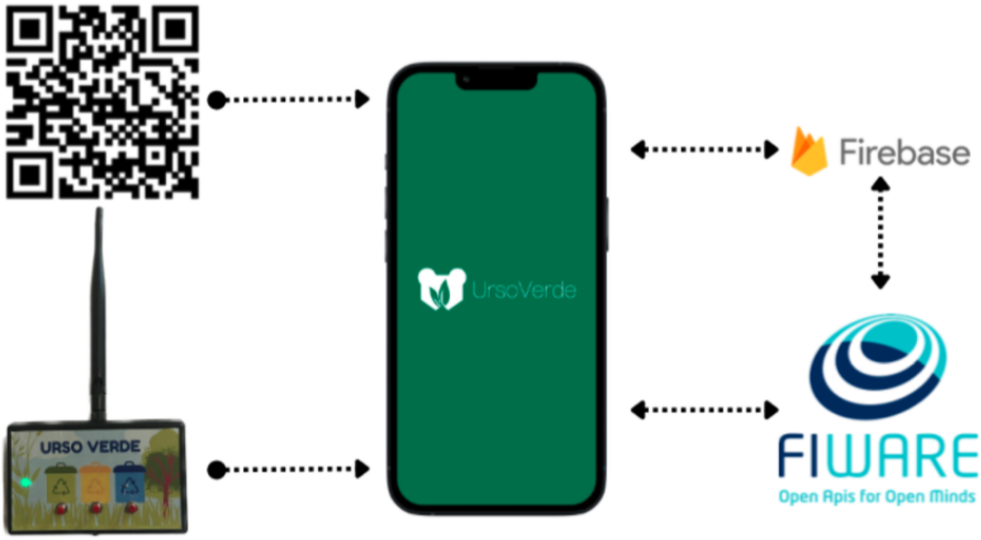



Figure 1.6: Architecture of the second version of Urso Verde [7]

## 1.5 Document Structure

This dissertation is structured in chapters as follows:

- **Chapter 1:** Presents the state of the art within smart cities, back-end applied to mobile apps, the purpose and motivation for this dissertation.
- **Chapter 2:** Describes the methodology and guidelines followed throughout this dissertation in order to successfully deliver and conclude the work.
- **Chapter 3:** Details the requirements the project must respect in order to ensure safety.

- **Chapter 4:** Clarifies the proposed solution to the presented problem and introduces the tools used in the conception of this dissertation.
- **Chapter 5:** Explains the work carried out, the implemented features and the final presented work.
- **Chapter 6:** Examines the challenges, milestones and problems that appeared.
- **Chapter 7:** Discusses the results and provides an overview of the successful and unsuccessful parts of the project.
- **Chapter 8:** Summarises the state of the project and suggests future improvements.

# 2

# Methodology

The investigation conducted for this dissertation, regarding its methodology, followed the principles of an agile approach where the work had iterative development. As a starting point, the working team had weekly meetings where each step was strategised and prepared. These meetings were followed by recurrent gatherings with *Coimbra City Lab* (CCL), ESJC, and CMC. From all these moments of planning and organising, the team was able to draft and organise a plan for the creation of the project structures, which entailed the different needs and purposes of each partner/structure.

The research group was formed by Master's and Doctoral students from different areas connected to human-in-the-loop, all under the supervision of Professor Jorge Sá Silva.

## 2.1 Work Team

### 2.1.1 Research group

The research group is composed of master's and doctoral students from different courses under supervision of Professor Jorge Sá Silva and Professor André Rodrigues. This group meets weekly, presenting the developed work in this period, gathering suggestions and opinions, providing assistance while sharing their weekly successes and setbacks, fostering a friendly and cooperative environment.

### 2.1.2 Coimbra City Lab

CCL is a CMC initiative that promotes and supports the development of innovative and sustainable technological ideas in a real-world environment. It acts as a facilitator in forming partnerships aimed at supporting the continuous development of projects. Urso Verde is a project inserted in this initiative and regular meetings between the development team and CCL were conducted in

order to access the state of the project to coordinate meetings with other partners or manages possible delays.

### 2.1.3 Escola Secundária Jaime Cortesão

The pilot project was conducted at ESJC from May 23, 2024, to June 14, 2024. Prior to this, a meeting was organized on the school grounds with the school headmaster and the class director regarding which students would participate in the experience and how their rights would be respected under the General Data Protection Regulation laws. Additionally, a visit to the installations was conducted, along with an on-field test, to ensure the correct functioning of the developed work. Other visits were scheduled for problem tracking and solving and to gather feedback at the end of the project.

## 2.2 SCRUM

In the fast-paced and ever-evolving industry of software development, choosing the right methodology is crucial for the success of a project.

Historically, management approaches that focus on hierarchical structures and standardised processes to maximise productivity such as *Taylorism* and *Fordism* have dictated industrial practices. However, due to their strictness these methods have been proven inefficient and inadequate and often struggle to keep the pace and falling short in meeting the expectations and changing demands for today's software projects.

Agile methodologies, prioritise development cycles and offer an iterative approach with focus on stakeholder engagement providing a framework for continuous improvement.

Scrum is one agile methodology that embodies flexibility, responsiveness, and collaboration. Scrum values short, time-boxed iterations called sprints. These sprints typically last two to four weeks, and at the end of each sprint, it is expected that there will be an increment in the software, addressing items from the product backlog (prioritised work items). There are regular reviews to ensure continuous feedback providing a constant improvement.

The key roles in Scrum include the *Product Owner* which represents the client, defining the priority of the work items, a *Scrum Master* responsible for organising the team and the *Development Team*, responsible for the delivery of the work. The work conducted in this dissertation followed this methodology, with continuous oversight from the Product Owner (Sandra Rodrigues, representing CMC and CCL), the Scrum Master (Professor Jorge Sá Silva), and the development team. Its features were implemented using an iterative approach that allowed for regular adjustments based on feedback, ensuring that the project remained aligned with expectations and requirements throughout

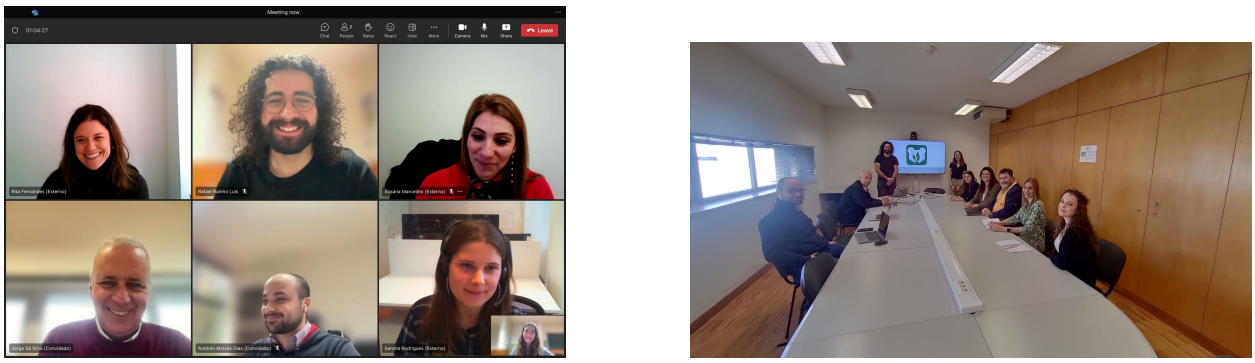
its development cycle.

## 2.3 Team meetings and work planning

Several meeting were organised between the involved parties in the project to ensure deadlines, debate ideas and discuss requirements. Apart from the weekly meetings with the research group, there were also meetings with CMC to present work progress, requirements, ideas, designs and functionalities, and with ESJC to discuss restrictions and responsibilities, present and demonstrate the project, evaluate flaws and errors and gather important feedback. These meetings are summarised as follows:

- Meetings with CMC and CCL (Figure 2.1)
  - November 10, 2023 - This first online meeting had the purpose of meeting the involved institutions in the project, discuss the main and general objectives and officialise the beginning of the work. (Figure 2.1a)
  - March 21, 2024 - The second online meeting had the objective of discussing the updates on the project and the deadlines for the next steps.
  - April 11, 2024 - In person meeting at *Departamento de Engenharia Eletrotécnica e de Computadores* (DEEC) facilities. Live demonstration of the prototype, demonstrating the project was ready to be implemented. This meeting set the official implementation schedule, leaving time for internal testing before the external tests. (Figure 2.1b)
  - May 22, 2024 - Meeting at Piscinas Municipais and Convento de São Francisco with the objective of teaching the staff how to use the dashboard, redeeming the user's prizes.
  - June 19, 2024 - Online meeting that served the purpose of discussing the results of the pilot project as well as plans for future implementations.
- Meetings with ESJC
  - May 6, 2024 - Meeting at ESJC with the school board. This meeting presented the project and discussed the data and image privacy of the students involved in the study and how that information would be kept anonymous. The parental consent can be found on appendix B.
  - May 21, 2024 - This meeting marked the beginning of the project as it was presented and explained to the students.
  - May 29, 2024 - This meeting was held to investigate alleged issues that were reportedly occurring with the application.

– June 13, 2024 - This meeting marked the end of the project. The students participated in a feedback survey and were thanked for being a part of the study.



(a) Online meeting

(b) In person meeting

Figure 2.1: Meetings with CMC

The work plan (figure 2.2) started with the definition of the proposal and the study of the most suitable tools and frameworks for this project. A few small projects were initiated with the objective of acknowledging and becoming familiar with these new protocols and programming languages. Shortly after the first meeting, the development of the project took place. The development of the server was initiated first to provide support for the development of both the dashboard and the mobile application. The internal tests started once the project was completed and lasted until the end of the pilot project along with bug fixing and error resolution. The implementation at ESJC took place in late June and the analysis of the results began after the project concluded.

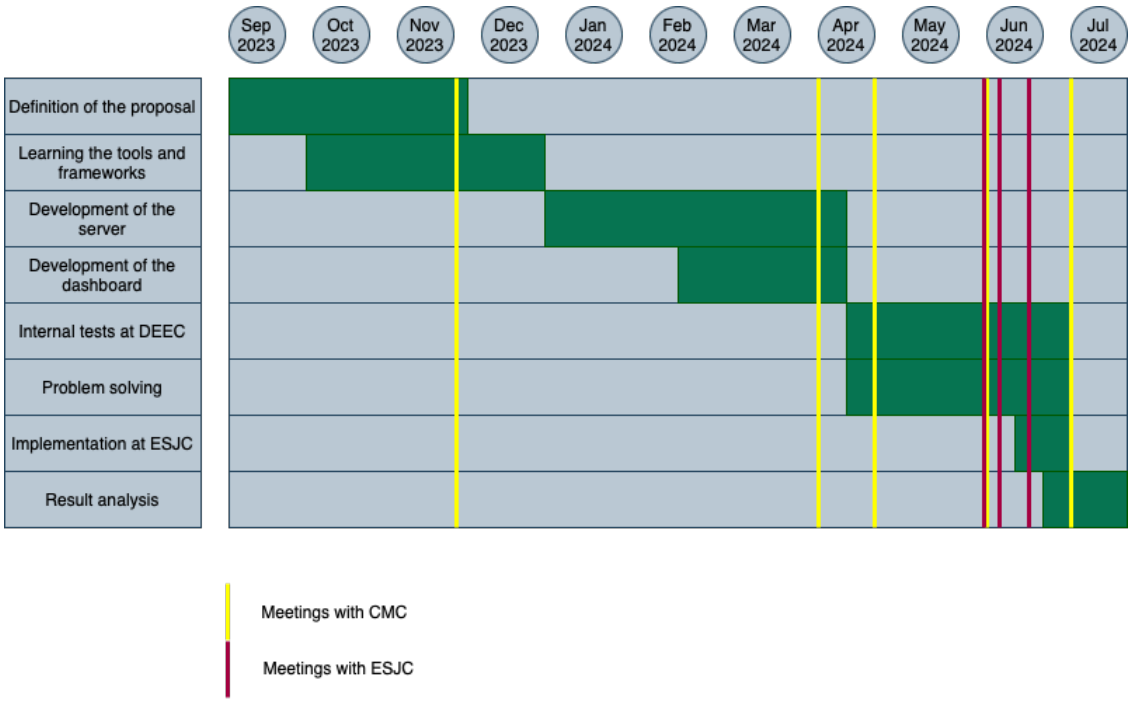



Figure 2.2: Gantt Chart

# 3

# Proposal and Development Tools

## 3.1 Proposal

The presented solution consists of a mobile application where the users can complete objectives. This can be achieved in different ways: scanning a QR code, a Near Field Communication (NFC) tag, a Bluetooth device, or any other viable alternative given the circumstances and conditions of the specific objective to be fulfilled. Additionally, there is a management dashboard for authorised administrators where it is possible to manage the ecosystem, whether by adding or removing objectives or prizes, or managing users and statistics, such as counting how many users are completing each objective or how many objectives each user has completed. A Flask server intermediates all the communication, connecting both interfaces to the database and handling all back-end logic and data management.

Given the budget for the project and its main objective of testing Urso Verde in the real world with a small, controlled group, the focus at this stage was on using QR codes only, while ensuring versatility for easy future upgrades to other methods of completing objectives.

The proposed custom dashboard should allow administrators on different levels, enabling CMC partners to validate the redeemed prizes by the users and empowering the main administrator of the platform to access and analyse critical information at a glance. Interactive visualisations will present data in a clear and compelling manner, facilitating the identification of trends, patterns, and anomalies. Real-time data access enables proactive decision-making by allowing users to monitor performance and identify potential issues as they arise.

A Flask server will act as the intermediary of this ecosystem. It will serve as a centralised data hub, providing a single point of access for both the dashboard and the mobile application to interact with the database. This way, data redundancy is eliminated and the whole system operates with up-to-date information.

The server will also enhance security by acting as a mediator for communication between the

database layer and the clients, handling user authentication and ensuring all accesses are trustworthy and authorised. It will handle all complex back-end logic, allowing the dashboard and the mobile application to focus on providing a user-friendly experience.

The server will be built with scalability in mind, capable of handling increased user traffic and data volume as the project grows. Security is one of the main concerns as the server should be employing robust authentication and authorisation mechanisms to protect sensitive data within the system such as Web Authorization Protocol (Oauth) 2.0. Flask, chosen as the server framework, offers several advantages. Its clean syntax allows for rapid development and maintainability, while lightweight nature ensures efficient resource utilisation. Additionally, Flask's flexibility allows for customisation and integration with various libraries and tools to address specific project needs.

This proposed solution offers several benefits. The dashboard administrators will be provided with intuitive interfaces to access and interact with data and real-time information allowing users to make informed decisions based on accurate data. The chosen technologies ensure the application can grow in the future with the needs while maintaining robust security measures.

In conclusion, this proposal outlines the development of a comprehensive user-friendly dashboard and a robust intermediary server. By making use of the strengths of React, and Flask, this solution is a good start to Urso Verde providing CMC with the tools needed to achieve success.

## 3.2 Development tools and technologies

### 3.2.1 Frameworks

#### React

The front-end interface for the dashboard was built with React[10], a popular JavaScript framework originally created by engineers at Facebook to solve the challenges involved when developing complex user interfaces with datasets that change over time[11]. It has component-based architecture which allows the creation of reusable components and elements. This approach simplifies the development process and the code maintainability. React utilises a virtual Document Object Model (DOM) to efficiently update the user interface.

Banks and Porcello explain that "The virtual DOM is made up of React elements, which conceptually seem similar to HTML elements, but are actually JavaScript objects. It is much faster to work directly with JavaScript objects than it is to work with the DOM API. We make changes to a JavaScript object, the virtual DOM, and React renders those changes for us using the DOM API as efficiently as possible." [5] By manipulating the virtual DOM, React minimizes the DOM manipulations required by the browser which results in a smoother and dynamic experience for the

user.

#### Flask

Flask is a framework that has three main dependencies[15]: the routing, debugging, and Web Server Gateway Interface subsystems come from Werkzeug[23]; the template support is provided by Jinja2[22]; and the command-line integration comes from Click[21]. Given it being a lightweight and versatile Python framework, it is ideal for handling server-side operations as it provides a clean syntax.

#### Third-Party libraries

To optimise the workflow some third party libraries were integrated. These correspond to open source code that provides reusable functions, methods and classes reducing development time and improving efficiency as they simplify otherwise complex tasks. The used libraries are listed in appendix D.

### 3.2.2 Tools

#### Visual Studio Code

Visual Studio Code was the main Integrated Development Environment for code development. Visual Studio Code offers a wide range of extensions such as syntax highlighting, debugging tools, and version control integration.

#### Docker

To ensure the deployment is consistent across various environments, Docker[8] containers were employed. Docker containers can package an application and its dependencies inside a virtual container and this technology is called as container technology. Those packages can be build, ship and run inside distributed environments by developers or system administrators with ease. [32] By containerising the application, it is safe to assume that the application will behave identically across different deployment environments.

#### Firebase

Firebase provides several products such as app hosting, authentication, cloud functions and storage and a Not Only Structured Query Language (NoSQL) database. This combination of tools facilitated the development of a well-structured, scalable, and maintainable web application.

Finally, Firebase, a comprehensive platform from Google, was chosen for hosting the application. Firebase provides robust hosting capabilities for both the static React front-end and the Flask back-end. Firebase's hosting infrastructure ensures high availability and scalability for the application. Additionally, Firebase offers a variety of features that can be integrated into the application, such as authentication and database management. This combination of tools facilitated the development of a well-structured, scalable, and maintainable web application.

#### GitHub

GitHub is a web-based platform that facilitates version control using Git. [13] It enables teams and individuals to collaborate on code and track the changes. It is one of the most popular platforms in the software development community valuing open source software making it easy to share code with others allowing user's to fork repositories, pull requests and track issues. GitHub desktop offered an interface that facilitated keeping the code updated with the team.

#### Site24x7

Site24x7 [1] is known for its monitoring solutions. With a global presence and cloud-native scalability, Site24x7 offers tools and solutions to maintaining websites up and running. Monitoring the availability of the server, server status, bottlenecks and performance metrics was essential to ensure minimal downtime which is crucial for maintaining user satisfaction and trust.

### 3.2.3 Technologies

#### HyperText Transfer Protocol Secure

Hypertext transfer protocol secure (HTTPS) is a secure version of Hypertext transfer protocol (HTTP). HTTPS uses Transport Layer Security (TLS), an encryption protocol to encrypt communications. TLS previously known as Secure Sockets Layer secures communications by using an asymmetric public key infrastructure[2]. This type of encryption uses two keys to secure the communications between two parties: the public key is available to everyone. It encrypts information that can only be decrypted by the private key which lives on a web server.

After the connection is secured, the client formulates a request message. This message has an overhead which indicates what is the action to perform. This action can be from one of the types in table 3.1. The response is a three-digit code (table 3.2) indicating the result of the request accompanied with the result itself (or not, if not successful).

Table 3.1: Some of HTTP methods

| **HTTP method** | **Description** |
|---|---|
| GET | Send named resource from the server to the client. |
| PUT | Store data from client into a named server resource. |
| DELETE | Delete the named resource from a server. |
| POST | Send client data into a server gateway application. |
| HEAD | Send just the HTTP headers from the response for the named resource. |

Table 3.2: HTTP status codes

| **HTTP status code** | **Description** |
|---|---|
| 1XX | Informational - Request received, continuing process |
| 2XX | Success - The action was successfully received, understood, and accepted |
| 3XX | Redirection - Further action must be taken in order to complete the request |
| 4XX | Client Error - The request contains bad syntax or cannot be fulfilled |
| 5XX | Server Error - The server failed to fulfil an apparently valid request |

### Oauth 2.0

OAuth 2.0 is the "industry-standard protocol for authorisation. OAuth 2.0 focuses on client developer simplicity while providing specific authorisation flows for web applications, desktop applications, mobile phones, and living room devices. This specification and its extensions are being developed within the Internet Engineering Task Force OAuth Working Group." [16] Oauth is a protocol for obtaining authorisation tokens to access protected resources with the resource's owner consent. This protocol as several functionalities such as token revocation, token exchange, dynamic client registration, and token introspection to ensure security when requesting access to private data. It uses the standardised JavaScript Object Notation (JSON) Web Token format and incorporates security measures like Proof Key for Code Exchange to prevent attacks. These tokens were used to ensure that users had secure and authorised access to the resources they needed, enhancing both security and user experience.

# 4

# Requirements

Before starting the project, a profound study was conducted to point out the dashboard functional requirements (DFR), server functional requirements (SFR) and dashboard non-functional requirements (DNFR). This was discussed amongst the weekly group meetings and throughout a literature study[25] [18] [26]. The dashboard must have the requirements presented in table 4.1.

## 4.1 Dashboard functional requirements

Table 4.1: Dashboard Functional Requirements

| Requirement (DFR) | Requirement Description |
|---|---|
| **1** | **Login with Admin Access** |
| **2** | **View Objectives** |
| **3** | **Add New Objectives** |
| **4** | **View Prizes** |
| **5** | **Add New Prizes** |
| **6** | **View History of Users** |
| **7** | **Validate Codes for Prize Redemption** |

The dashboard first functional requirement (DFR 3.1) represent the login. Administrators may be able to login with their email and password. If the user does not exist or has not administrator privileges, access to the dashboard shall not be conceded. DFR 2 indicates that the administrator may be able to verify all the objectives in the system.

DFR 3 can then be sub-divided into eleven sub-requirements (table 4.2). The administrator, by adding a new objective must give essential information for the correct functioning of the system.

The dashboard must take the name (DFR 3.1) of the objective which will be shown to the users in the mobile application as well as the values of each objective (DFR 3.2) and the description (DFR 3.8). DFR 3.3 specifies the wallet where the points will be credited once the objective is completed. This wallet is tied to certain objectives and certain prizes to allow different institutions or CMC partners to offer their respective prizes. DFR 3.4 defines a number of times that a user must complete the objective before the points are credited and DFR 3.5 defines the maximum number of times the user can receive the points from that objective. For example the user could receive points once has recycled paper three times but is limited to receive the points two times. DFR 3.6 defines a the time-out between objectives. If this time-out is defined to twenty-four hours the user could complete the objective only once a day. DFR 3.7 indicates the limit date for completing the objective. After the indicated date, the objective may no longer be completed. DFR 3.9 and DFR 3.10 are tied to each other as they respectively indicate a geographical centre point and a radius around this point. This radius delimits a circle where the objective may be completed. Finally, DFR 3.11 specifies the refresh rate for the QR code. If the refresh code is forty-eight hours, the QR code for completing the objective will change every two days.

Table 4.2: Functional Requirements for Adding Objectives

| **Requirement** | **Requirement Description** |
|---|---|
| **3** | **Add Objectives** |
| **3.1** | **Add name of the objective** |
| **3.2** | **Add the value in points for the objective** |
| **3.3** | **Select the type of wallet for the objective** |
| **3.4** | **Specify the number of times the objective must be fulfilled** |
| **3.5** | **Specify the maximum times each user can fulfill the objective** |
| **3.6** | **Define the time user must wait to repeat the objective** |
| **3.7** | **Set due date for the objective** |
| **3.8** | **Provide a description for the objective** |
| **3.9** | **Select the location on a map for the objective** |
| **3.10** | **Define the maximum distance from the location for the objective** |
| **3.11** | **Specify the refresh rate of the QR code referring to the objective** |

Similarly to DFR 2, DFR 4 indicates that the administrator may be able to verify all the prizes in the system. DFR 5 indicate the functional requirements to add a prize. This to can be sub-

divided into several sub-requirements (table 4.3). DFR 5.1, 5.2 and 5.6 are information (name, allusive image and description)that will be presented to the users in the mobile app to give context of the prizes available. DFR 5.3 define the number of points needed to retrieve the prize and DFR 5.4 defines which wallet will the points be withdrawn from. DFR 5.5, 5.7 and 5.8 define the limit date to retrieve the prizes, the available stock and the maximum number from the given price each user can retrieve. The users may not retrieve the prizes if any of these conditions are met.

Table 4.3: Functional Requirements for Adding Objectives

| **Requirement** | **Requirement Description** |
|---|---|
| **5** | **Add Prizes** |
| **5.1** | **Add name of the prize** |
| **5.2** | **Add image for the prize** |
| **5.3** | **Specify price of the prize** |
| **5.4** | **Select the type of wallet for the prize** |
| **5.5** | **Set due date for the prize** |
| **5.6** | **Provide a description for the prize** |
| **5.7** | **Specify availability/stock for the prize** |
| **5.8** | **Specify the maximum times each user can acquire each prize** |

There are two sub-requirements referring to view the history of the users (table 4.4). DFR 6.1 should allow to consult the view the completed objectives of each user while DFR 6.2 should allow to consult the retrieved prizes of each user.

Table 4.4: Functional Requirements to View History of Users

| **Requirement** | **Requirement Description** |
|---|---|
| **6** | **View History of Users** |
| **6.1** | **View completed objectives by a given user** |
| **6.2** | **View redeemed prizes by a given user** |

The last DFR is related to the need to assess whether or not the prize has been collected. There are two moments for the user to retrieve the prize. In the first moment, the user will retrieve the prize in the mobile application and get a random code. In the second moment, the user may go to the location where the prize will be delivered and validate the code. By doing this, the user's privacy and anonymity is preserved as the code is enough to retrieve the prize, without needing any

other personal details. DFR 7 allows the administrator to known if the prize was already handled to the user or if it is still reserved.

## 4.2 Dashboard non functional requirements

Table 4.5 defines the DNFR. It should have an intuitive and user-friendly interface (DNFR 1) and should be depoyed online in a website (DNFR 2) for easy access and use.

Table 4.5: Dashboard Non Function Requirements

| Requirement | Requirement Description |
|---|---|
| **1** | **User-Friendly Interface** |
| **2** | **Hosted on a website** |

## 4.3 Server functional requirements

SFR are stated in table 4.6. These include bi-directional communication with the dashboard (SFR 1), Android and iOS devices (SFR 2) and the database (SFR 3). These communications must support various data formats (SFR 4), including images and strings (JSON). The server must also do perform logical computations (SFR 5) for any device requiring it. These include point increments and deductions, geographical and temporal calculations, history tracking, handling authentication and authorisation, among others.

Table 4.6: Server Functional Requirements

| Requirement | Requirement Description |
|---|---|
| **1** | **Establish bi-directional communication with the dashboard** |
| **2** | **Establish bi-directional communication with both Android and iOS devices** |
| **3** | **Establish bi-directional communication with the database** |
| **4** | **Support the transmission of various data formats, including strings and images** |
| **5** | **Possess the capability to perform logical computations as required** |

# 5

# Implementation

The implementation of this project followed an agile methodology, with several sprints each of them featuring a few tasks. These tasks were weighted and adapted to fit the available time of each sprint.

## 5.1 Architecture

The adopted architecture (figure 5.1) is designed around an intermediary server that independently connects to the mobile application, the dashboard and the database, all through secure HTTPS connections. The server manages and routes the requests between these components ensuring a secure and efficient flow of data. The project was carried out in collaboration with Matilde Carvalho, a colleague who was responsible for the mobile application. More details about the mobile application can be found in her dissertation. [28]

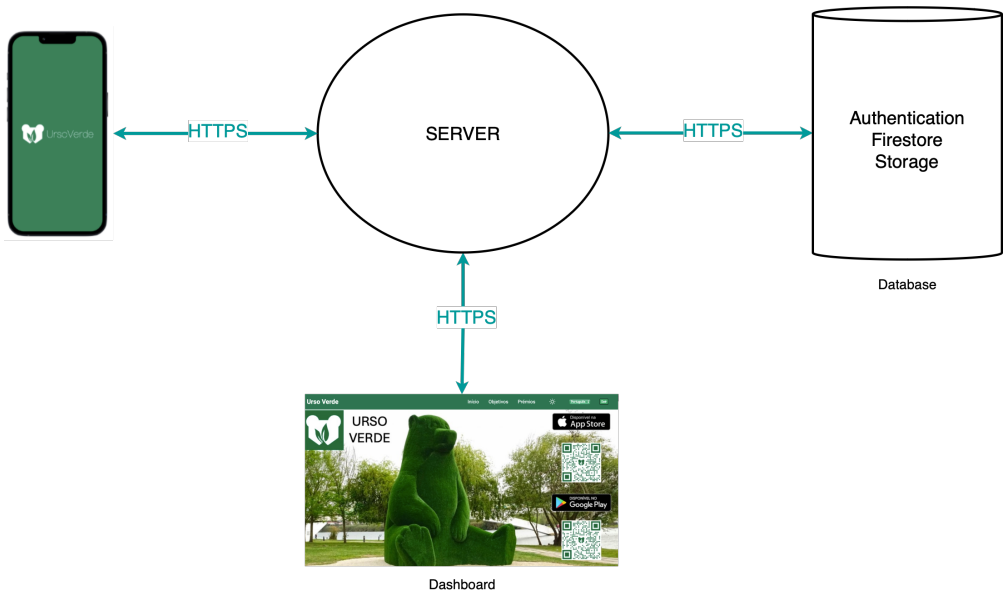



Figure 5.1: Architecture

## 5.2 Data structure

The data in this project was stored in Firestore, a NoSQL database. Firestore organises its data hierarchically, structuring its data in collections and documents. Documents are individual records that hold data in a key-value pairs format. Collections are containers for the documents. This particular system allows complex and nested data structures (Listing 5.1) as each collection can contain multiple documents and each one of these can include sub-collections. This project database included eight main collections (table 5.1) that include all the data.

Table 5.1: Database Collections

| Database Collections | |
|---|---|
| Admin | PrizeHistory |
| ObjectiveHistory | Prizes |
| Objetives | UserAdmin |
| PrizeHistory | Users |
| Prizes | ValidationCodes |

Each of the collections hold different types of information organised to facilitate efficient retrieval and management. The *Admin* collection holds the information of the full administrators of the dashboard in contrast of *UserAdmin* that holds the information of the administrators that have partial access. The *Objectives* and *Prizes* collections are store information about the various objectives and prizes available to users. Additionally, *ObjectiveHistory* and *PrizeHistory* collections maintain historical records of objectives and prizes, respectively, for both operational and statistic purposes. The *Users* collection stores all the personal information of the users, and finally, the *ValidationCodes* collection stores one-time-use codes to ensure that all participating users have parental consent to be part of the study. All of these collections, except for *ObjectiveHistory* and *PrizeHistory*, are organised by documents where each document has a unique, randomly generated ID and stores the information in key-value format. The *PrizeHistory* collection holds documents whose IDs correspond to the IDs of the specific users to whom the data refers. Each of these documents contains a sub-collection for each redeemed prize, containing its information. The *ObjectiveHistory* collection also holds documents whose IDs correspond to the IDs of the specific users to whom the data refer. Inside these documents, however, there is a document related to each of the objectives, with individual documents for every instance the objective was completed.

Listing 5.1: Database organisation example

```
Collection
{
    DocumentID-1 {
        key: value;
        anotherKey: anotherValue;
        subCollection {
            DocumentID-1.1 {
                key: value;
                anotherKey: anotherValue;
            }
            DocumentID-1.2{
                key: value;
                anotherKey: anotherValue;
                subSubCollection{
                    DocumentID-1.2.1{
                        key:value;
                    }
                }
            }
        }
    }
    DocumentID-2 {
        key: value;
    }
}
```

## 5.3 Server communication

The data communication between the server and the mobile app, the dashboard and the database is established over HTTPS. A self signed certificate was created for this stage of the project.

To ensure the tokens are valid and the server does not provide the requested information without a valid token, every communication includes a confirmation of the token's validity. If the token is not valid, the server sends a response to the mobile app, obligating it to generate a new token. If this second token is also not valid, the response code will force the mobile app to log out. If it is valid, the request will proceed and the requested resources will be delivered.

## 5.4 Server operations to mobile application

The server is responsible for all the logical operations. There are several routines associated with every POST/GET request. Every time there is an error the server will send an appropriate code status. There are code status for server errors, permissions denied or unauthorised, missing resources or resources not found and other errors such as incorrect location.

### 5.4.1 Register

The server receives the email, password, user type and a verification code. The email and password will be stored in Firebase Authentication, responsible for the authentication management. The user type determines which wallets, objectives and prizes the user will have access to within the application. Finnaly the verification code was implemented given the students involved in this study need a parental authorisation. These codes were stored in the database and associated with a boolean flag ensuring each code could only be used once and that all the students would have to have an authorisation to receive a code. If the register is successful, an email will be sent so that the account may be confirmed. If all the information is valid, the server creates an account and sends a confirmation email. After the email is confirmed, the user is able to access the account.

This secure registration process fosters responsible account creation and upholds ethical considerations regarding parental consent for student participation within the study.

### 5.4.2 Login

The login process takes place through several steps:

- Credential collection: The user inserts its credentials.

- Firebase Authentication: The server transmits the credentials over to Firebase Authentication, a service responsible for the authentication management.
- Email Verification: The email is verified to prevent unauthorised access through unverified emails.
- Token Generation: Once all verification steps are successful, a token is transmitted to the application. This token acts as a digital key, granting access to the user while establishing a secure connection.

### 5.4.3 Deleting the account

Deleting the account means that all information is completely removed. When the user initiates the account deletion, the process is definitive and irreversible. A sequential procedure ensures the removal of all the data tied to the respective user.

- History deletion: All user's historical data is erased including both completed objectives and earned prizes. This ensures that there is no trace of any activity within the system.
- Database deletion: Following history deletion, all personal information is permanently removed from the database.
- Authentication deletion: Finally the authentication credentials are removed from Firebase Authentication, the service responsible for the management of login credentials.

By implementing this multi-step procedure, the application protects its users information.

### 5.4.4 Viewing the objectives

When the user opens the mobile application page displaying the objectives, the app transmits a request to the server to which the server responds with a message containing two sets of information

- Objectives data: All data referring to the objectives such as names, descriptions or points. This data will provide the user with a clear understanding of the available objectives.
- User data: This data includes the balance in each user's wallet.

By including this information in the response, the server ensures the app can visually display the user's current point totals, providing valuable feedback and motivation.

### 5.4.5 Viewing the prizes

Similar to the process for retrieving objective data, when a user opens the prize page, the app also requests information to the server. The server responds with a JSON message containing a list of objects. Each object within this list represents a specific prize, encapsulating all its relevant details such as the title, the cost value, the image or the description.

### 5.4.6 Scanning an objective

When a user scans an objective, typically by reading a QR code, the server receives several data points. This information includes:

- The type of scan: This could be a QR code string read directly on the mobile device, an image containing a QR code for server-side decoding, or potentially NFC tags or Bluetooth readings (planned for future development).
- The user's ID.
- The user's current geographical coordinates.

If the received information is a Base64-encoded image,the server decodes the image is into Portable Network Graphics format and attempts to extract a QR code from the image. In cases where the image does not possess a QR code in the image or the QR code is invalid (does not correspond to any active objectives) the server transmits a specific error code back to the mobile application, indicating the encountered scenario.

However, if a valid QR code is received (either from the image or directly as a string), the server performs additional checks to ensure objective validity. The first step involves comparing the user's geographical coordinates with the designated location for completing the objective (as defined in DFR 3.9 and 3.10). This comparison is performed using the Haversine formula, a mathematical formula that calculates the distance between two points on a sphere (like the Earth) [20]:

$$a = \sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\Delta\lambda}{2}\right) \tag{5.1}$$

$$c = 2 \cdot \operatorname{atan2}\left(\sqrt{a}, \sqrt{1-a}\right) \tag{5.2}$$

$$d = R \cdot c \tag{5.3}$$

where:

- $\Delta\phi$ is the difference in latitudes between the two points: $\Delta\phi = \phi_2 - \phi_1$
- $\Delta\lambda$ is the difference in longitudes between the two points: $\Delta\lambda = \lambda_2 - \lambda_1$
- $\phi_1$ and $\phi_2$ are the latitudes of the two points in radians
- $\lambda_1$ and $\lambda_2$ are the longitudes of the two points in radians
- $R$ is the Earth's radius (mean radius = 6,371 km)
- $d$ is the distance between the two points

To check if the distance $d$ is within a certain radius $r$, we compare $d$ with $r$:

$$d \leq r \tag{5.4}$$

If Equation 5.4 holds true, the coordinate is within the specified radius.

In conclusion, to determine if a coordinate is within a certain radius, we first calculate $a$ using Equation 5.1, then determine $c$ with Equation 5.2, and finally find the distance $d$ with Equation 5.3. We then check if this distance $d$ satisfies the condition in Equation 5.4.

If the user is in a valid location to complete the objective, the system checks whether the user has previously completed the same objective. If so, the data from the last completion is retrieved. This data allows verification of whether the user can still complete the objective or has reached the maximum limit (DFR 3.5), and whether the user has waited the defined time between objectives (DFR 3.6). If all these conditions are fulfilled, the objective is considered valid. When the objective is valid, a history entry is created, saving all the relevant information. Additionally, if the number of times the objective has been completed is a multiple of the required number to earn points, then the points are credited to the respective wallet.

### 5.4.7 Redeeming a prize

When redeeming a prize, the server receives the user's ID and the prize ID. It then proceeds to retrieve data from the last prize of the same category and checks the stock availability, and whether the user has already reached the maximum limit for that specific prize. It also checks the user's balance and if all the conditions are met, the server updates the stock, records the number of redeemed prizes by the user, and adjusts the user's balance accordingly. Finally, for the purpose of anonymously identify the user that redeemed the prize, a random code is generated, which the user can use to claim the prize and a record of the transaction is created with all the relevant details stored.

### 5.4.8 Checking the redeem codes

When the user opens the page displaying all the codes, the server receives a request and retrieves all records of transacted prizes associated with the specific user from the database. Subsequently, the server responds to the mobile application with the codes for these prizes along with a boolean flag indicating whether each prize has been claimed or not.

### 5.4.9 Periodical changed of the QR codes

Currently, a routine runs every day at hourly intervals. This routine checks all the objectives in the database as specified in DFR 3.11. If a QR code needs to be changed, this routine will replace it with a new, unique QR code. This is not an ideal solution. A more efficient solution would be to utilise cloud functions. Cloud functions can be triggered within the cloud environment, eliminating the need for having physical hardware constantly running. However, cloud functions are part of a subscription paid plan and, for this stage of the project, this less optimal solution was adopted.

## 5.5 Dashboard Logical Operations

The approach for the development of the dashboard started by the definition of the requirements and discussion of what the goals were for this. What functionalities it should offer and what it should look like.

Once defined the functional requirements for the dashboard the first step was the creation of some templates for the front-end pages of the dashboard, in order to create a general idea of what functions should be created.

React is build in modules, so having something to work on all pages such as dark mode, we need it to wrap all the components inside. A top-bottom approach was followed creating the outer components first and proceeding to the specific inside components. First components to be created were the dark-mode and the language support components. The dashboard can load any language because every sentence visible to the user are stored in a JSON file. The next logical step was to create the login page. This should be visible to anyone, unlike the other pages that should not be visible to unauthenticated users even when trying to access directly through the direct Uniform Resource Locator link.

### 5.5.1 Dashboard - Server communication

All data processing occurs on the server. A JavaScript method has been developed to receive information from any React component and send it to the appropriate route on the server. This

approach consolidates data handling and communication between the front and back-end in an efficient manner.

### 5.5.2 Login credentials

The credentials (email and password) are requested from the user in the front-end (figure 5.2). After input,the credentials are sent to the server which validates them in FireBase authentication database using Oauth 2.0. Based on the database response the server raises an adequate flag that is sent over to the front-end. This flag can be of one of four:

- The credentials do not match (user does not exist or the password is incorrect)
- The user does not have administrator privileges
- The user is not a full administrator and has authorisation only to redeem prizes. These are the CMC partners responsible for the validation of the codes and hand the prizes over to the user.
- The user is a full administrator and has access to the entire functions of the dashboard. The user is able to manage the objectives, the prizes and the users.

When receiving the response, the front-end either concedes the access or displays an error message.

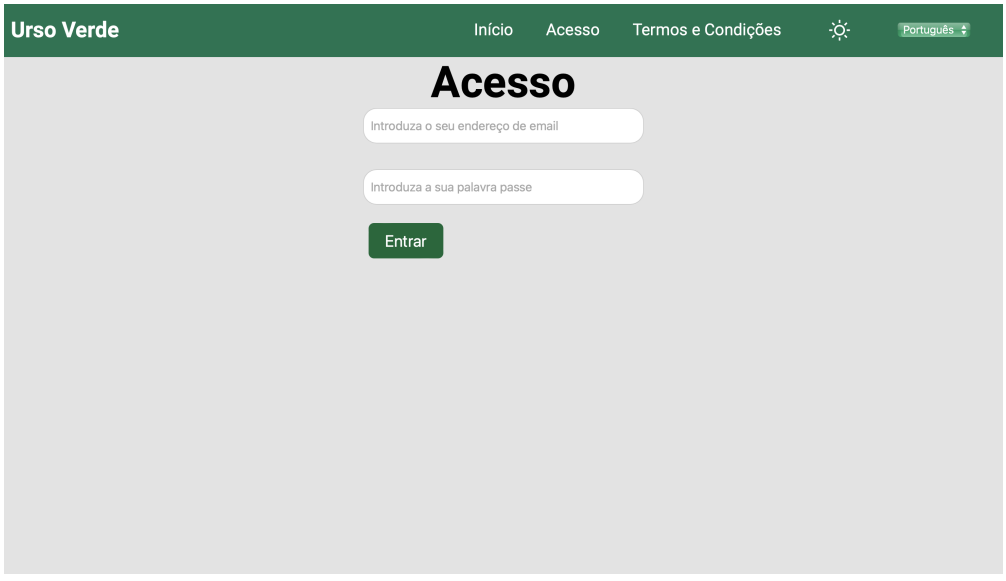
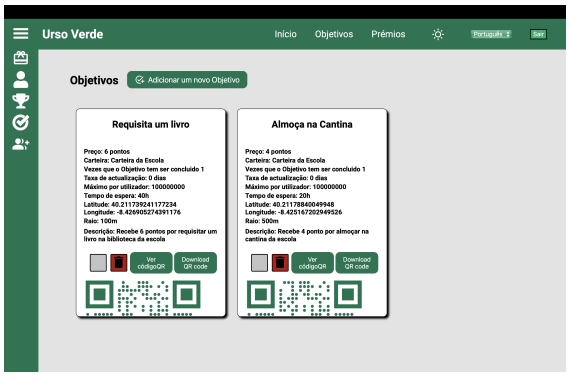



Figure 5.2: Login Screen - Dashboard

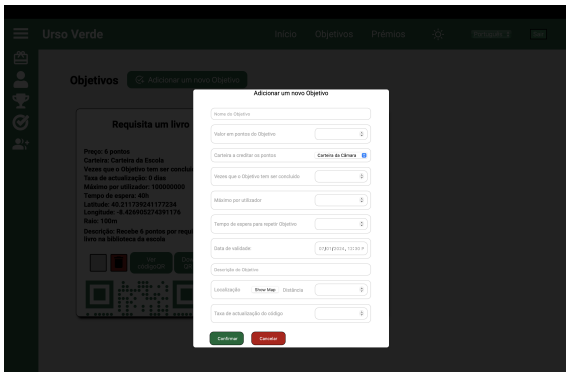
### 5.5.3 Viewing and adding objectives and prizes

When adding an objective or prize, several sets information are requested. These information contribute to both correct functioning and informative purposes. Leaving empty fields will, depending

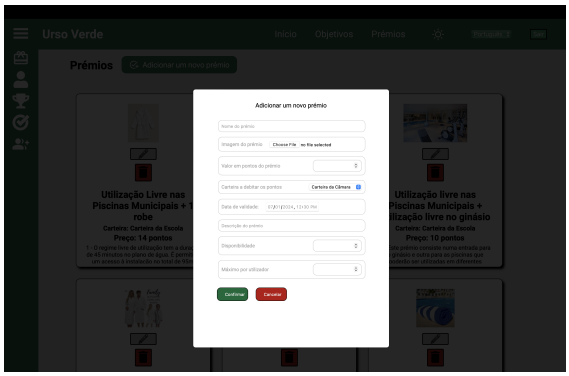
on the case, result in either an default value to be inputted or a raised flag, demanding the administrator to input a value in the accepted format. The administrator can then send the new objective or prize to the database through the server. In case of any error, an error message it will always be displayed. The pages that show the objectives and the prizes make a GET request to the server and map the information to different cards. These cards are custom made React components that adjust the information for better readability. Figure 5.3 represents some examples of the front-end graphic user interface visible in the dashboard for viewing and adding objectives or prizes. Figure 5.3a displays the main screen of the objectives page, where the administrator can add new objectives, delete existing ones or navigate to other page. Figures 5.3b and 5.3c represent, respectively, the forms that are required to be filled in order to add a new objective or prize to the database. Figure 5.3d illustrates an example of entering the location data, where the objective must be completed in order to be considered valid. Figure 5.4 is a close up of figure 5.3c, similar to figure 5.3b, provided for better understanding.



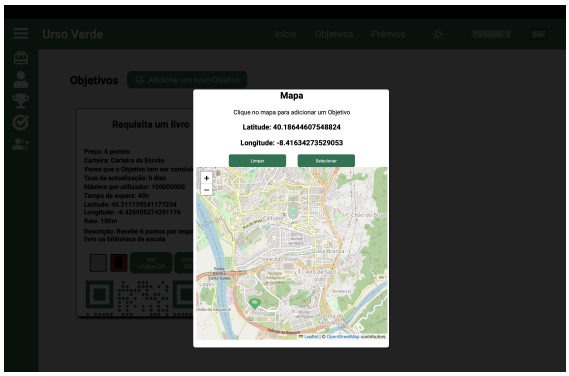

(a) Objective Page




(b) Add Objective Form




(c) Add Prize Form




(d) Map - Inserting Location

Figure 5.3: View and Add Prizes and Objectives

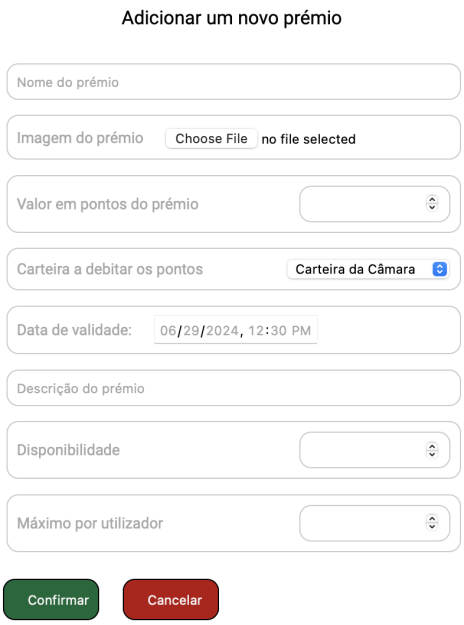



Figure 5.4: Add Prize Form (Close Up)

### 5.5.4 Validate redeem code

One of the most important features of the dashboard is the validation of redemption codes. When a user accumulates enough points to claim a prize and chooses to do so, a random code is generated and associated with both the prize and the user. The user can then visit the designated prize collection point and provide the code along with their email address. The staff member at the collection point, who is also an administrator of the dashboard with restricted access, can validate the code.

It was essential to ensure that the functionality of this page is as simple and intuitive as possible, minimising the workload for the staff.

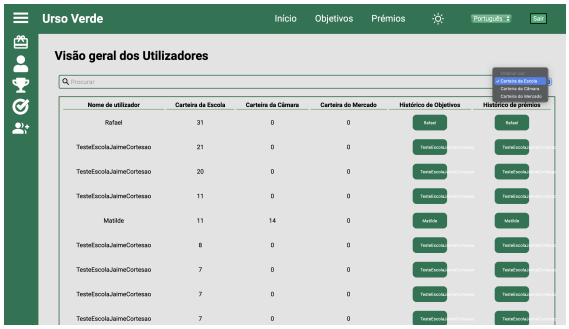
### 5.5.5 Statistics

The dashboard also features statistical views on various subjects. There are three menus that provide different information: objectives, prizes, and users. All objectives and prizes appear in these menus, regardless of whether they are expired, out of stock, or under other conditions that would typically exclude them from the main display.

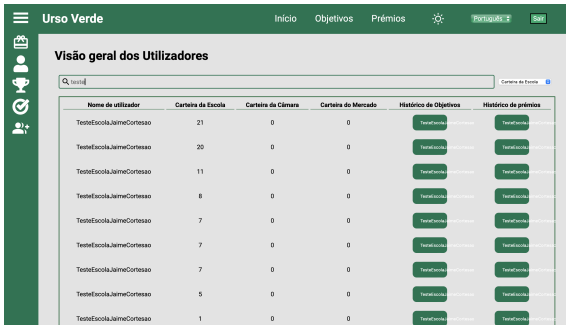
Regarding users, administrators can review individual histories related to completed objectives or redeemed prizes. Each menu allows searching by name and sorting by filters such as creation date, expiration date, or others.

Figure 5.5 shows some screenshots from the user interface when viewing statistical data. Figure 5.5a illustrates the user search menu, with figure 5.5b showing the same page when searching with

a term. Figure 5.5c shows the objective statistical page while 5.5d shows the statistical page the prizes. Figures 5.5e and 5.5f respectively show detailed information about the made objectives and prizes redeemed by a specific user.

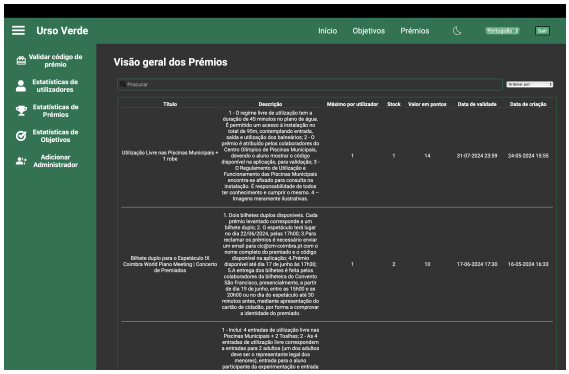



(a) User Search

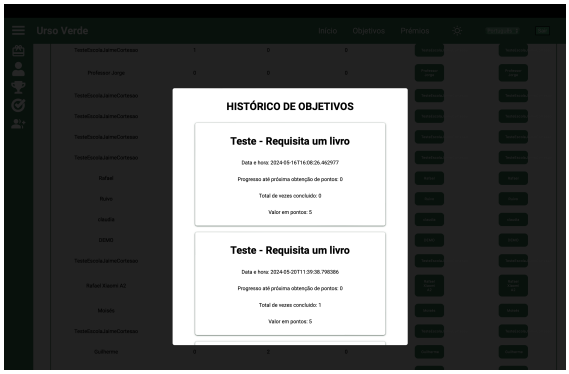



(b) User Search "teste"

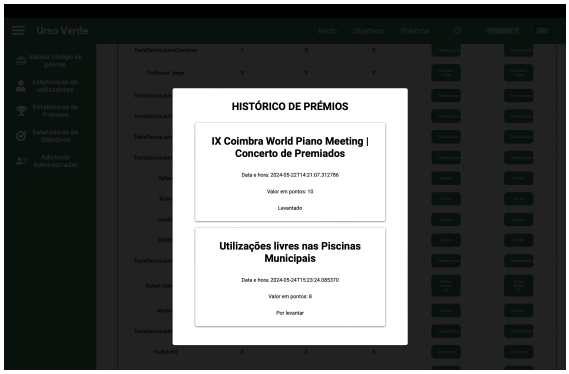

(c) Objective Search




(d) Prize Search




(e) Objective History




(f) Prize History

Figure 5.5: Statistics of Users, Prizes and Objectives

## 5.6 Other features

Some other non-functional features were included in the project, such as a top bar that can be visualised in most of the figures above. This top bar offers a simplistic yet very useful visual interface where the administrator can toggle between light and dark mode, change the display language, navigate through the main pages, and logout. The dark mode can be seen in figure 5.5d.

Also a foldable side bar where the administrator can navigate between the secondary pages. This side bar can be visualised in its reduced form in 5.5a and in its augmented form in figure 5.5c.

## 5.7 Server monitoring and infrastructure management

Ensuring continuous availability and performance of the server was essential to keep the whole ecosystem running throughout the duration of the project. This involved constant monitoring of the server's availability. With Site24x7, the server's availability was continuously monitored by periodically sending requests to a dedicated GET route every five minutes, tracking response times, identifying potential bottlenecks, and receiving real-time alerts and notifications for an immediate, adequate response. Figure 5.6 presents the availability of the server from May to July. The availability, tracked every five minutes throughout the project's duration (approximately 25,000 times) showed uptime of 99.838%. Additionally, the mean response time was 101 milliseconds which is perfectly acceptable given the context of web and mobile application services.


Figure 5.6: Server monitoring

## 5.8 Tests

The test were conducted in two distinct phases. Internal tests at DEEC and the tests at ESJC with a reduced number of twenty-two students.

### 5.8.1 Internal tests

The internal tests were conducted with members of the research group. They downloaded the application to their smartphones and proceeded to scan some QR codes and redeem test prizes. Examples of the objectives included, but were not limited to, closing the doors and windows of the laboratory or turning off the lights. This engaged the entire group in intensive testing of the application to identify any errors that could occur in the server or database. During this stage, some errors were reported and immediately fixed. Having multiple people with different smartphones and

different operative systems, helped ensure that the server could manage multiple connections at the same time.

### 5.8.2 Tests at ESJC

It was decided that in order to increase control of eventual errors that would appear these tests would only have two objectives available. The objectives, and their restrictions are depicted in table 5.2.

Table 5.2: Objectives on ESJC tests

| Objective | Points | Expiration | Time to wait between scans |
|---|---|---|---|
| Lunch at the school canteen | 1 | 28-07-2024 | 20 hours |
| Request a book from the school library | 5 | 28-07-2024 | 40 hours |

On the other hand there were multiple prizes to engage the participation.

Table 5.3: Prizes on ESJC tests

| Prize | Points | Stock |
|---|---|---|
| 1 Robes Family Pack = 4 free entries to the Municipal Pools (2 adults + 2 children) + 2 adult robes | 20 | 1 |
| 1 Towels Family Pack = 4 free entries to the Municipal Pools (2 adults + 2 children) + 2 towels | 15 | 1 |
| 1 double ticket for the show: IX Coimbra World Piano Meeting on 22/06/2024 | 10 | 2 |
| 1 free use of the Pools + 1 free use of the gym | 10 | 1 |
| 1 Free use of the Municipal Pools + 1 towel | 12 | 1 |
| 1 Free use of the Municipal Pools + 1 robe | 14 | 1 |
| 1 free use of the Municipal Pools | 8 | 22 |

After the start of the tests and after a couple of days, there were some active users but no records on the server of any attempts to register any objectives. The lack of attempts was concerning and it was difficult to determine whether there were no attempts because the students were unable to scan any objective or because they did not even try. Another meeting at the school was promptly organised to investigate these problems. From this meeting a few conclusions were drawn:

- Some students did not have parental authorisation to participate.
- Some students had incompatible mobile devices.
- The QR codes were placed inside plastic sleeves and some students couldn't scan the QR codes due to reflections.

To address the particular situation of the students that had incompatible smartphones (the QR codes were not being read by the devices), an alternative was quickly implemented. This alternative

consisted of encrypting the image captured by the smartphone, sending it to the server and, after decoding, extracting a QR code and processing it in server as detailed in chapter 5.4.6.

Unfortunately, this did not work as the incompatible smartphones were also incompatible with this solution. [28]

# 6

# Milestones and challenges

This chapter covers some of the main challenges encountered as well as some important milestones and achievements.

## 6.1 Learning about all the new technologies

One significant challenge encountered during the project was the lack of prior knowledge in JavaScript, HTML, or CSS. Learning React, a JavaScript framework for building user interfaces, proved particularly difficult in the early stages given this background. Initially, components, state, props, syntax and other concepts of React, were unfamiliar and complex. Several tutorials were completed to understand some of the peculiarities of the language. These tutorials were crucial in providing a basic understanding of how React works, offering practice skills and hands-on projects and contributed to archive the ability to program autonomously.

## 6.2 Connecting the server to the dashboard and mobile app

Another crucial aspect of the project involved establishing a secure connection between the server, the web dashboard, and the mobile application. This communication channel utilised HTTP at first to ensure data transmission. The implementation was not easy at first due to a lack of knowledge of data transmission protocols and certificates but eventually, after effectively establishing HTTP connection to both the server and the mobile application, further studies on HTTPS were undertaken to ensure enhanced security and proper certificates. This transition to HTTPS provided another layer of encryption, ensuring data security and protected transmission routes.

## 6.3 Teamwork and communication

Working in teams was challenging yet crucial for the success of the project. Communication was essential for the division of responsibilities and consequently the minimisation of overlap in work and alignment with the project's goals and progress. Additionally communication with CMC, CCL and ESJC was vital for organisation and definition of priority tasks, requirements and actions.

# 7

# Results and Discussion

At the conclusion of the experimentation at the secondary school, the points gathered by the students were as exhibited in Table 7.1. The students did not claim any prizes before the end of the allowed schedule to gather points from the proposed objectives. However, the points were still available for the exchange of prizes until the end of July (still ongoing). Therefore, the points in the table represent all the points gathered.

From this data, a few conclusions can be drawn. Eight students from the sample collected enough points to retrieve prizes: three students accrued seven points each, one student accumulated five points, and another student earned only one point. The interest in the participation was heterogeneous suggesting the prizes may have not been appealing for everyone. This observation is reinforced by the user satisfaction survey administered to all the students at the conclusion of this study. The research group accompanied by members of the CCL were present at the school to ensure that the participants completed the questionnaire. The survey responses can be found in appendix A. The main takeaway from the survey is that the majority if the students were not enthusiastic about the proposed prizes. Future initiatives should consider offering a wider variety and possibly tailored to the specific group to increase interest and consequently participation rates.

For the duration of the project neither the server nor the database client reported any issues or problems. However, it is important to consider the work load. In future tests, with increased app usage, more prizes and additional, upgrading the database to a paid subscription may be necessary. This upgrade would support increased read and write operations, as well as cloud functions, potentially reducing HTTP requests between the server and mobile app. Equipping the server with a Graphics Processing Unit could enable faster processing for scanning multiple QR codes simultaneously in the server instead of in the application. Overall, the study was successful, and the back-end for the application performed as expected during the pilot project. While the dataset was limited due to the study's small sample size, it provided valuable insights that can inform future initiatives with larger populations, expanded prize offerings, broader objectives, and diverse environments.

Table 7.1: Final points collected by the students in the study

| Student | Final accumulated points |
|---|---|
| Student 1 | 21 |
| Student 2 | 20 |
| Student 3 | 21 |
| Student 4 | 11 |
| Student 5 | 8 |
| Student 6 | 7 |
| Student 7 | 7 |
| Student 8 | 7 |
| Student 9 | 5 |

# 8

# Future Work

There is always something to improve and margin to progress. Overall the project has a lot of room for progression in different fields.

## 8.1 Dissemination of results

The work conducted in this dissertation will be part of the application for "Call for IC&DT Projects - Artificial Intelligence, Data Science and Cybersecurity relevant to Public Administration" within the scope of PRR Investment - C05-i08 - More Digital Science.

The work developed has also resulted in a paper (Appendix C) that will be published in "IEEE Internet Computing".

## 8.2 Partners

In the future getting more partners with more appealing prizes and objectives is essential. One of the conclusions of this study is that the interest in Urso Verde was highly correlated with the offered prizes. More objectives is also fundamental for this project. With more objectives to complete, the engagement will also increase. Targeted marketing to a specific public is also something to consider given that it may increase the interest and consequentially, the number of active users.

## 8.3 Server

To aim for more people using the app, the server needs to be able to accommodate the extra connections and the increasing data processing. If the number of users (and data) grows significantly, in the future it may be useful to train machine models to target and recommend certain objectives or prizes to different people given the collected data.

## 8.4 Functionalities

More functionalities can be added in the near future. These include other forms of completing the objectives such as the Bluetooth beacons (implemented in past versions of Urso Verde), NFC or complements of more than one.

Also the re-inclusion of the chatbot could be interesting if capable of informing the users of the conditions to claim prizes or complete objectives.

The inclusion of a leaderboard could lead to an increase of interest and consequentially engagement from the users.

Unlocking the already implemented functionality of multiple wallets tied to different partners could lead to an increment of prizes and objectives, making the application more appealing. In this scenario, these partners must have access to the dashboard features of managing the objectives and prizes (only the ones that are tied to them).

# Bibliography


[1] Site24x7. https://www.site24x7.com/website-monitoring.html?src=ProdMenu&pg=home. Accessed: 2024-06-15.

[2] What is https? | cloudflare. https://www.cloudflare.com/learning/ssl/what-is-https, 2024. Accessed: 2024-03-05.

[3] Aisha Abdallah, Mohd Aizaini Maarof, and Anazida Zainal. Fraud detection system: A survey. *Journal of Network and Computer Applications*, 68:90–113, 2016.

[4] Osama MA AL-atraqchi. A proposed model for build a secure restful api to connect between server side and mobile application using laravel framework with flutter toolkits. *Cihan University-Erbil Scientific Journal*, 6(2):28–35, 2022.

[5] Alex Banks and Eve Porcello. *Learning React: modern patterns for developing React apps*. O'Reilly Media, 2020.

[6] María Cortés-Cediel, Iván Cantador, and Manuel Pedro Rodríguez Bolívar. Analyzing citizen participation and engagement in european smart cities. *Social Science Computer Review*, 39:089443931987747, 10 2019.

[7] António Moisés Tomás Dias. Urso verde smart city platform for healthy and sustainable living. Master's thesis, University of Coimbra, 2023.

[8] Inc. Docker. Docker documentation, 2024. Accessed: 2024-02-26.

[9] Igor Dunayev, Nataliia Gavkalova, and Aleksander Kud. Designing a platform-based model of civic participation within the smart-city concept for post-war ukrainian cities. *Eastern-European Journal of Enterprise Technologies*, 4:46–56, 08 2023.

[10] Inc. Facebook. React documentation, 2024. Accessed: 2024-02-26.

[11] Cory Gackenheimer. *Introduction to React*. Apress, 2015.

[12] Stephan Gammeter, Alexander Gassmann, Lukas Bossard, Till Quack, and Luc Van Gool. Server-side object recognition and client-side object tracking for mobile augmented reality.

In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 1–8, 2010.

[13] Inc. GitHub. Github, 2024. Accessed: 2024-03-29.

[14] Christos Goumopoulos. Smart city middleware: A survey and a conceptual framework. *IEEE Access*, 2024.

[15] Miguel Grinberg. *Flask web development.* " O'Reilly Media, Inc.", 2018.

[16] IETF OAuth Working Group. Oauth 2.0, 2024. Accessed: 2024-04-04.

[17] Johns Hopkins University. COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. `https://coronavirus.jhu.edu/map.html`, 2024. Accessed: 2024-06-15.

[18] Nikolas Martinos. Functional and non-functional requirements for a performance dashboard in cs: Go. B.S. thesis, University of Twente, 2021.

[19] Greg Milner. Creating the dashboard for the pandemic. *ArcUser*, pages 56–59, 2020.

[20] Dwi Arman Prasetya, Phong Thanh Nguyen, Rinat Faizullin, Iswanto Iswanto, and Edmond Febrinicko Armay. Resolving the shortest path problem using the haversine algorithm. *Journal of critical reviews*, 7(1):62–64, 2020.

[21] Pallets Projects. Click documentation, 2023. Accessed: 2024-02-26.

[22] Pallets Projects. Jinja2 documentation, 2023. Accessed: 2024-02-26.

[23] Pallets Projects. Werkzeug documentation, 2023. Accessed: 2024-02-26.

[24] Ines Levin R. Michael Alvarez and Yimeng Li. Fraud, convenience, and e-voting: how voting experience shapes opinions about voting technology. *Journal of Information Technology & Politics*, 15(2):94–105, 2018.

[25] Reza Rabiei and Sohrab Almasi. Requirements and challenges of hospital dashboards: a systematic literature review. *BMC Medical Informatics and Decision Making*, 22(1):287, 2022.

[26] Samad Samad and Albertus Laurensius Setyabudhi. Design of dashboard administration of the public sector. *Engineering and Technology International Journal*, 5(01):71–80, 2023.

[27] Oscar Torres Sanchez, José Marcelo Fernandes, André Rodrigues, Jorge Sá Silva, Fernando Boavida, Jorge Eduardo Rivadeneira, Afonso Viana de Lemos, and Duarte Raposo. Green bear - a lorawan-based human-in-the-loop case-study for sustainable cities. *Pervasive and Mobile Computing*, 87:101701, 2022.

[28] Matilde Saraiva de Carvalho. Urso verde: A reward-based system for a smart city platform. Master's thesis, 2024.

[29] Alper Sarikaya, Michael Correll, Lyn Bartram, Melanie Tory, and Danyel Fisher. What do we talk about when we talk about dashboards? *IEEE Transactions on Visualization and Computer Graphics*, 25(1):682–692, 2019.

[30] KR Seeja and Masoumeh Zareapoor. Fraudminer: A novel credit card fraud detection model based on frequent itemset mining. *The Scientific World Journal*, 2014(1):252797, 2014.

[31] Chai K Toh. Security for smart cities. *IET Smart Cities*, 2(2):95–104, 2020.

[32] Tuomas Vase. Advantages of docker. B.S. thesis, 2015.

[33] Steve Wexler, Jeffrey Shaffer, and Andy Cotgreave. Front matter. In Steve Wexler, Jeffrey Shaffer, and Andy Cotgreave, editors, *The Big Book of Dashboards*. Wiley, 2017.

# Appendix A

## Feedback Survey

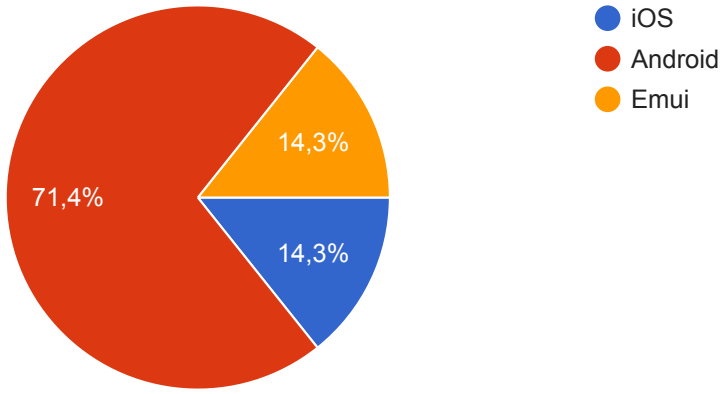
# Inquérito de Satisfação - Aplicação URSO VERDE

7 respostas

## Qual o sistema operativo do teu telemóvel/tablet?

Copiar

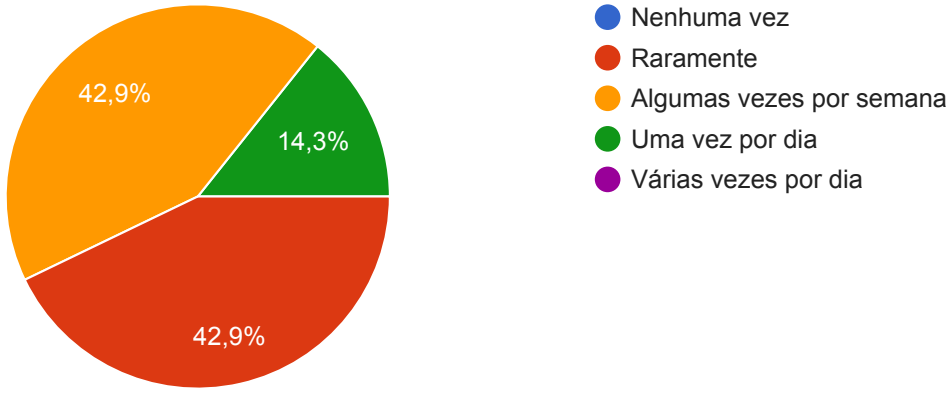
7 respostas




## Com que frequência usaste a aplicação?

Copiar

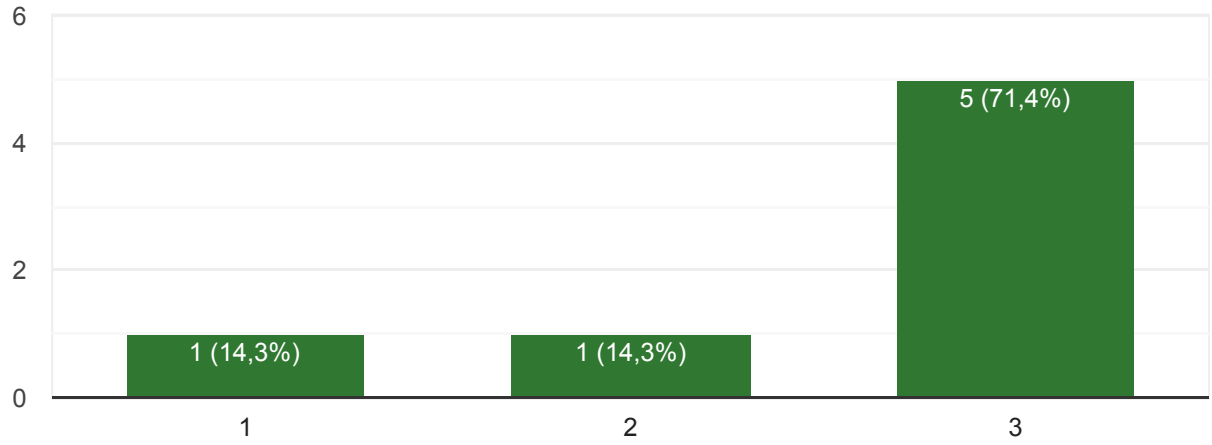
7 respostas



## Como avalias a facilidade de uso da aplicação?

Copiar

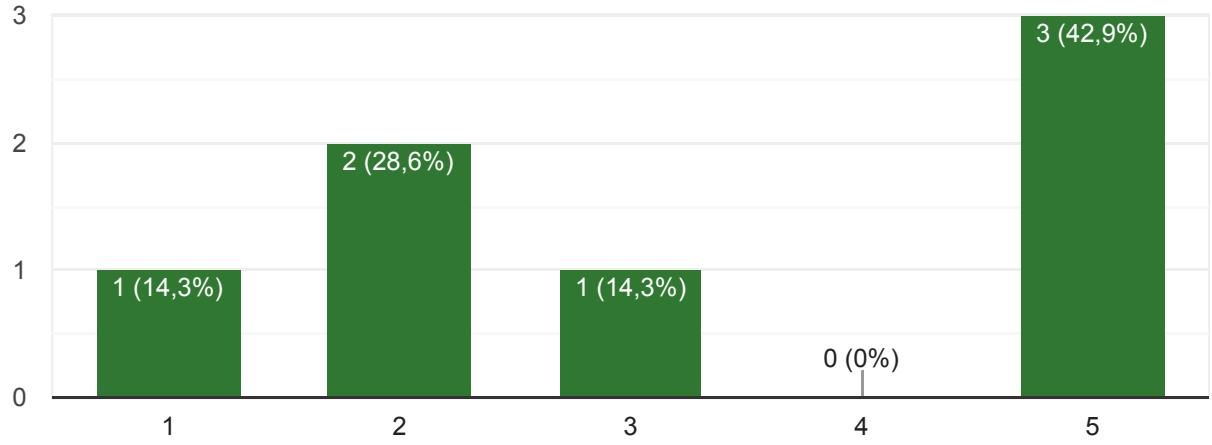
7 respostas




## Qual a tua satisfação com a funcionalidade de digitalização de QR Codes?

Copiar

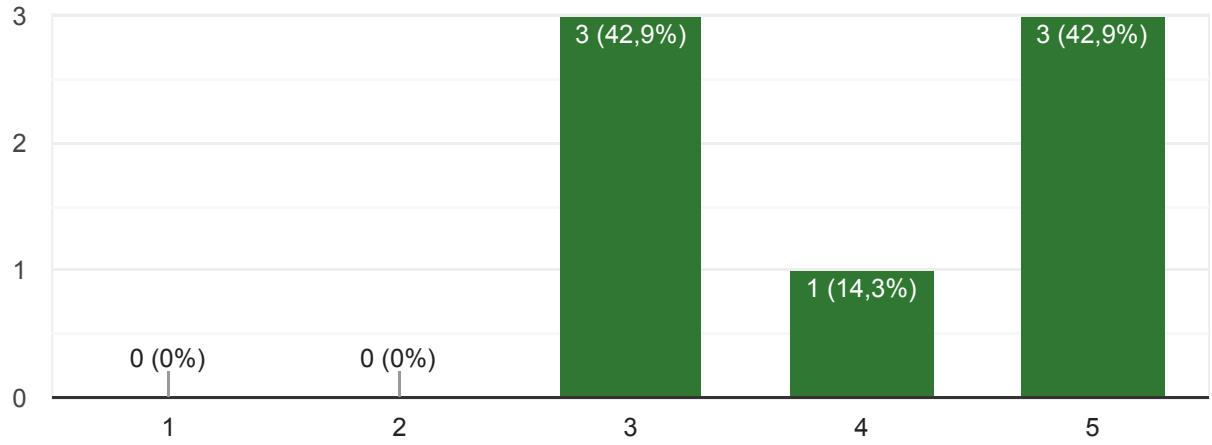
7 respostas



## Quão satisfeito estás com o sistema de pontos da aplicação?

Copiar

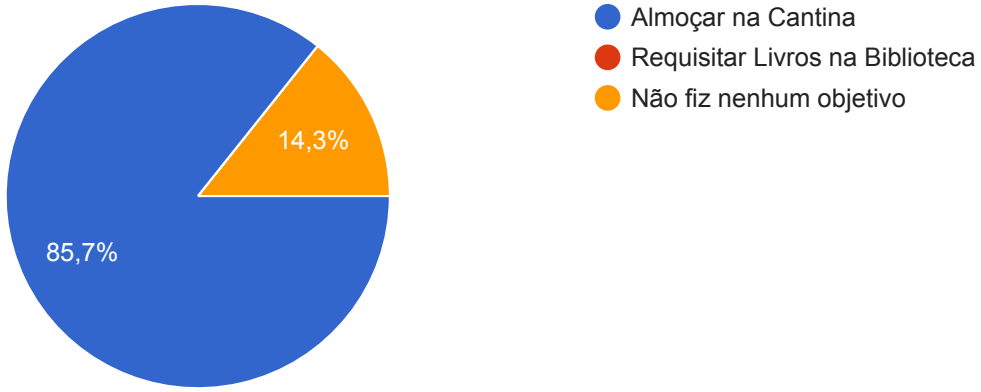
7 respostas




## Qual objetivo fizeste mais vezes?

Copiar

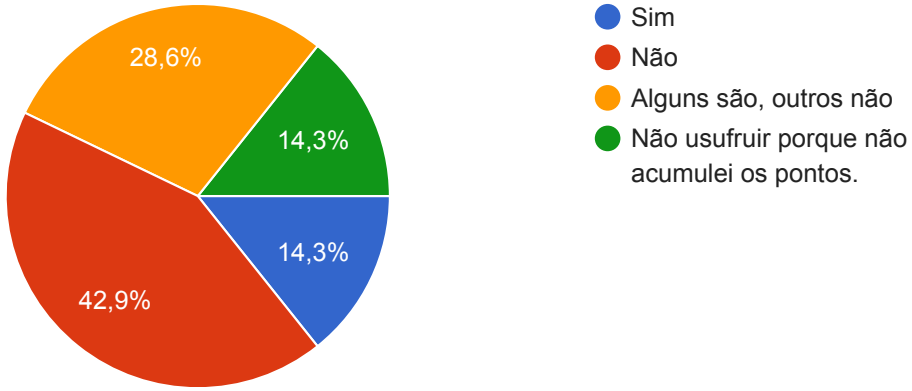
7 respostas




## Os prémios oferecidos foram atrativos para ti?

Copiar

7 respostas



## O que mais gostaste na aplicação Urso Verde?

7 respostas

- Os prêmios
- O sistema de recompensas por pontos
- Ser premiado.
- Em si a ideia só não teve muito hesito a ser aplicada
- os prémios
- É fácil de usar
- Fácil de usar e intuitiva

## O que menos gostaste na aplicação Urso Verde?

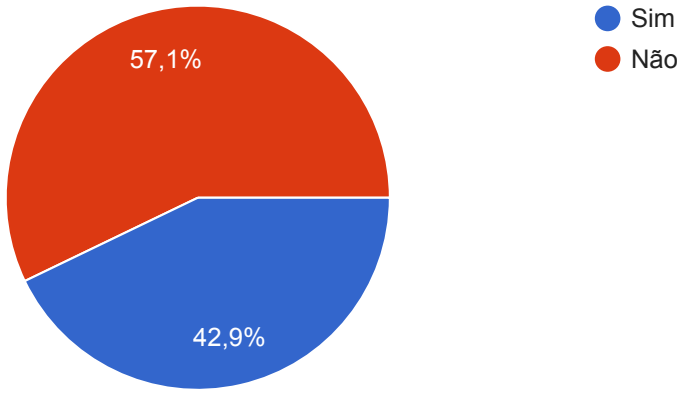
7 respostas

- Gostei de tudo só acabei esquecendo de resgatar os pontos hahaha
- Leitura do QR code
- Os prémios .
- É relativo já que não muito atractivo por mais que a ideia seja ótima
- o não funcionamento da digitalização dos qr codes
- não tenho nada para reclamar
- Os prémios

## Tiveste problemas a utilizar a aplicação?

Copiar

7 respostas




## Se tiveste problemas a utilizar a aplicação, descreve:

3 respostas

Leitura do QR code

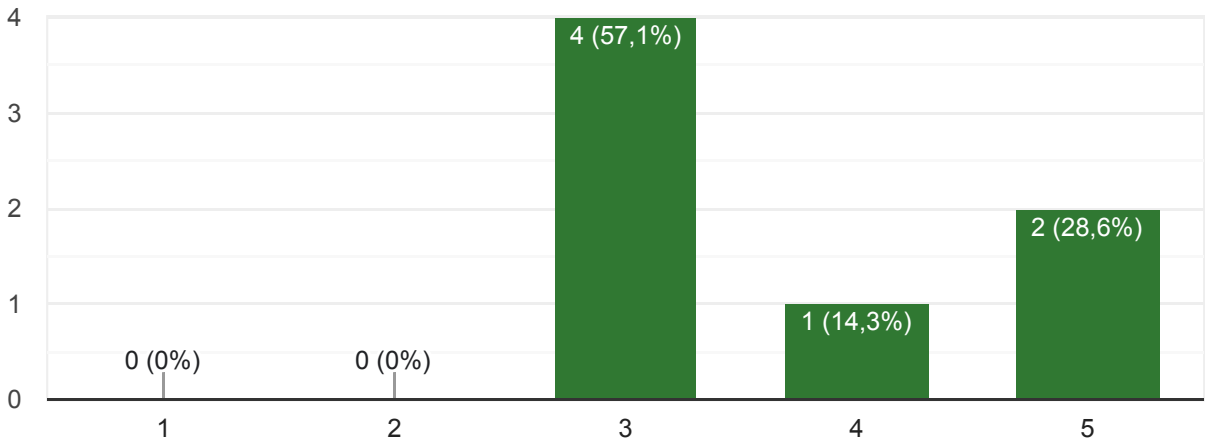
A scanner o código

quando tentava digitalizar o código, ficava a escanear mas sem sucesso, quando usava o novo meio de escanear (tirar fotografia) o telemóvel saia da aplicação

## Qual a tua satisfação geral com a aplicação?

Copiar

7 respostas



## Sugestões de melhoria

6 respostas

Acho que não tenho nenhuma sugestão de melhoria

Melhorar a leitura de QR code, pois era chato ter que enviar emails assim fazendo com que eu perdesse o interesse.

Melhores prémios.

Premios mas atrativos e mais incentivos as pessoas ,porque a ideia em si é brilhante só que é muito atrativa para nós

Não tenho, pelo menos no meu telefone funciona muito bem.

Os prémios deveriam ser mais atrativos para nós para existir competição entre nós

# Appendix B

## Parental consent

# CONSENTIMENTO INFORMADO

Por favor, leia com atenção a seguinte informação. Se considerar que algo está incorreto ou que não está claro, não hesite em solicitar mais informações. Se concorda com a proposta que lhe foi feita, queira por favor assinar este documento.

**Título do estudo:** Projeto Urso Verde

**Enquadramento:** O Projeto Piloto Urso Verde, acolhido no *#CoimbraCityLab*, laboratório vivo de inovação tecnológica e de experimentação do Município de Coimbra, está a ser desenvolvido por alunos de mestrado e de doutoramento do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra, sendo coordenado pelo Professor Doutor Jorge Sá Silva.

**Explicação do estudo:** O Projeto Urso Verde traduz-se em uma plataforma que tem como objetivo recompensar os munícipes por boas práticas de cidadania e hábitos de vida saudáveis, como por exemplo reciclar, doar sangue, adotar um animal no canil/gatil municipal, entre outros.

O projeto consiste numa aplicação móvel que permite a leitura de códigos QR. Esta aplicação móvel, desenvolvida em âmbito académico enquanto projeto de mestrado, encontra-se em fase final de desenvolvimento, sendo necessário proceder à sua experimentação, no sentido de se aferir a sua viabilidade e adequabilidade.

Neste sentido, o projeto Urso Verde encontra-se acolhido no *#CoimbraCityLab*, uma iniciativa do Município de Coimbra, para apoio ao desenvolvimento e experimentação de ideias e projetos inovadores e sustentáveis de base tecnológica, em ambiente real, através da disponibilização do espaço físico urbano e do conhecimento técnico e experiência dos serviços municipais, e que serve como facilitador na criação de parcerias que permitam dar suporte e continuidade ao desenvolvimento dos projetos.

O Departamento de Educação e Saúde (DES), da Câmara Municipal de Coimbra, está a colaborar no desenvolvimento e na experimentação do projeto piloto Urso Verde, tendo solicitado, para o efeito, a colaboração da Escola Jaime Cortesão, no sentido de o mesmo ser experimentado por um grupo de alunos/as da escola. Pretende-se, inicialmente, realizar uma experimentação dirigida a um público-alvo com um número mais reduzido, envolvendo, para o efeito, uma turma de 22 alunos, do 10º ano da Escola Jaime Cortesão.

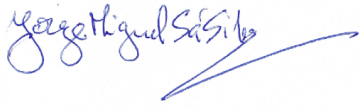
A experimentação do projeto decorrerá no período temporal compreendido entre 27 de maio e 14 de junho de 2024. Para o efeito, cada aluno deverá descarregar a aplicação para o seu telemóvel de forma a poder cumprir os objetivos propostos, que serão a "frequência da cantina escolar" e a "requisição de livros na biblioteca escolar". Por cada objetivo cumprido, os alunos participantes acumulam pontos na aplicação que serão, posteriormente, convertidos em prémios. Os prémios serão assegurados pelo Município de Coimbra, durante um período temporal que estará indicado na aplicação, e consistirão em entradas gratuitas para espetáculos no Convento São Francisco (a indicar) e entradas gratuitas no Complexo Olímpico de Piscinas Municipais.

**Condições e financiamento:** A participação é voluntária e gratuita. Em qualquer momento, poderá livremente recusar ou interromper a participação na experimentação, sem qualquer tipo de penalização por esta decisão.

**Confidencialidade e anonimato:** A confidencialidade e anonimato dos dados serão garantidos, sendo que o único dado solicitado é um e-mail (qualquer) para registo na aplicação. A aplicação cumpre com todos os requisitos legais impostos pelo Regulamento Geral de Proteção de Dados (EU)2016/679 do P.E. e do Conselho de 27 de abril (RGPD).

Informo que estou disponível para quaisquer esclarecimentos adicionais e manifesto o meu agradecimento pela sua participação.

**Assinatura:**



Professor Dr. Jorge Sá Silva
(Responsável do Projeto)

-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-o-

Eu, _____, Encarregado(a) de Educação do(a) aluno(a) _____, Nº. ___, do ___ ano, Turma ___:

☐ Declaro que aceito que o meu/minha educando/a participe neste projeto e permito a utilização dos dados que de forma voluntária forneço, confiando em que apenas serão utilizados para esta investigação e nas garantias de confidencialidade e anonimato que me são dadas pelos investigadores.

☐ Declaro que, para os devidos efeitos, autorizo/não autorizo (riscar o que não pretende) a recolha de imagens vídeo e fotografia do meu/minha Educando/a durante o experimento piloto, para divulgação das mesmas na comunicação social.

Assinatura: _____

Data: ___ /___ /_____

# Appendix C

## UrsoVerde: A Reward-Based Smart City Platform with Enhanced Privacy, Security, and Management

# *UrsoVerde*: A Reward-Based Smart City Platform with Enhanced Privacy, Security, and Management

Rafael Rainho Luís[†], Matilde Saraiva de Carvalho[†], J. Sá Silva[†‡], André Rodrigues[#‡], Guilherme Borges[†], A. Moisés Dias[†], Óscar Torres[*] and Carolina Pedro[+]

[†] *Department of Electrical and Computer Engineering, University of Coimbra, Coimbra, Portugal*
[*] *Department of Informatics Engineering, University of Coimbra, Coimbra, Portugal*
[+] *Department of Physics, University of Coimbra, Coimbra, Portugal*
[#] *Polytechnic University of Coimbra, Coimbra, Portugal*
[‡] *INESC Coimbra, Coimbra, Portugal*

***Abstract*—Reward-based systems are a promising tool to promote sustainable behavior in smart cities, using gamification and incentives to encourage environmentally friendly habits among citizens. This paper focuses on the *UrsoVerde* project, a reward-based system implemented in Coimbra, in partnership with the Coimbra City Council (CMC), aiming to promote good citizenship practices and sustainable living habits by rewarding residents for participating in activities such as recycling, using public transportation, supporting local businesses, visiting green spaces, and walking.**

**The UrsoVerde mobile application, compatible with iOS and Android devices, uses a point-based system. Users accumulate "Citizen Points" by scanning Quick Response (QR) codes or interacting with Internet of Things (IoT) devices at designated locations across the city. These points can be redeemed for rewards like discounts at participating stores, tickets to museums, or entries to the municipal pool.**

**While *UrsoVerde* was designed to transform Coimbra into a smart and digital city, encouraging users through rewards for sustainable actions that promote local commerce, it lacked a server to act as an intermediary between the user and the database, as well as a dashboard capable of efficiently managing the entire system. This study addressed these gaps by developing a server and a dashboard. The server, designed to isolate the database, filter traffic, and standardise data, is crucial for establishing effective communication with both the dashboard and users. The dashboard enables a system administrator to define rewards and objectives for users, validate reward transactions, and analyse detailed usage statistics of the platform in real time.**

**The project was implemented in the real world through a pilot project conducted at *Jaime Cortesão* School, where the functionalities and usability of the platform were tested, while essential feedback was collected for refinement before the official launch. This feedback was evaluated through surveys, and the overall results were highly elucidative, demonstrating the viability of the platform and its potential positive impact on the community. This technological advancement not only optimises *UrsoVerde*'s operation but also significantly contributes to its long-term effectiveness and sustainability.**

***Index Terms*—Smart Cities; Mobile Application; Dashboard; Internet of Things; Sustainability; Privacy**

## I. Introduction

The *UrsoVerde* project is a reward-based system implemented in Coimbra, Portugal, in partnership with the Coimbra City Council (CMC). The project aims to promote sustainable living habits and good citizenship practices among residents by rewarding them for engaging in activities such as recycling, using public transportation, supporting local businesses, visiting green spaces, and walking.

The *UrsoVerde* mobile application, available on iOS and Android devices, utilizes a point-based system where users earn "Citizen Points" by scanning QR codes or interacting with Internet of Things devices at specific locations throughout the city. These points can then be redeemed for rewards such as discounts at participating stores, tickets to museums, or entries to the municipal pool.

The project is a continuation of previous work and represents a collaborative effort between the Coimbra City Council, researchers, and students. The project underwent real-world testing through a pilot study at Jaime Cortesão School, providing valuable feedback for further development and improvement.

This paper is organized as follows. Section II identifies the related work, mainly reward-based systems and smart cities. Section III provides an overview of the system. Section IV details the *UrsoVerde* implementation. An assessment of the system is presented in section V, essentially covering the testing phases and results. The conclusions are presented in section VI.

## II. Related Work

### A. Smart Cities

Smart cities are urban areas that use technology to improve the quality of life for their residents. Human-in-the-loop refers to systems where humans and machines work together to

achieve a goal. In the context of smart cities, Human-in-the-loop means involving citizens in the design and implementation of smart city solutions [1].

Cortés-Cediel et al. [2] analysed European smart cities, revealing that these cities emphasise social initiatives and open participation models, prioritising social well-being and democratic engagement. Information and communication technologies, such as mobile apps, IoT devices, and open data, are key tools for citizen engagement, complemented by offline methods like meetings and cultural events.

Dunayev et al. [3] concluded that a human-centric approach prioritises specific public services that meet the citizens' needs. The involvement of the community, local authorities and businesses is crucial to ensure coordination in active citizenship. Moreover, the importance of leveraging digital technologies such as IoT, big data, and artificial intelligence to enhance service delivery and citizen engagement within smart cities was highlighted.

United Nations (UN) Sustainable Development Goals (SDG) reports emphasize that smart cities are pivotal in achieving sustainable development [14]. The UN provides platforms like the United Smart Cities (USC) initiative that aims to support cities in their transition towards becoming more sustainable, resilient, and inclusive through the use of technology and data [13]. This project also recognizes that the success of smart city projects is dependent of Human-in-the-loop and citizen involvement. By integrating citizen involvement throughout the process, USC aims to create more inclusive, equitable, and sustainable urban environments. This approach strengthens social cohesion and ensures smart city solutions truly serve the needs of the people.

### B. Reward-based applications

Reward-based applications (RBAs) are digital platforms that utilize gamification techniques to motivate user engagement. These applications typically offer rewards like points, badges, or leaderboard rankings for completing desired actions or tasks. The rewards are designed to provide immediate gratification and encourage repeated use of the application [4].

These platforms have emerged as promising tools for influencing user behavior, with applications spanning health, education, or marketing. Some examples of these applications:

- Sweatcoin: this app rewards users for physical activities such as walking, running, or achieving fitness milestones. Sweatcoin then converts steps into a digital currency that can be redeemed for goods and services [12].
- Google Opinion Rewards: this app allows users to win points by completing surveys. These points can be exchanged by credits to buy games, movies, e-books and Google Services subscriptions [5, 10].
- Acerta e Recicla: this portuguese app by *Sociedade Ponto Verde*, encourages and raises awareness about recycling through a gamification experience. It then gives vouchers in supermarkets [9].
- MB Way: this portuguese payment app developed by SIBS, features a challenge "Raspa e Ganha" where users can earn vouchers for making purchases. These vouchers can be redeemed for prizes such as cash or other rewards. The more frequently a user utilizes the app for transactions, the more vouchers they accumulate, increasing their chances of winning rewards [7].

A reward-based system can improve sustainability and healthier habits by using points making it more fun and engaging, trackable progress to motivate the users, social interactions as encouragement, and a variety of rewards [6].

### C. Dashboards

Dashboards have emerged as essential tools for system configuration, data visualisation, analysis, and interpretation for individuals, organisations, governments, and companies, allowing users to make important and informed decisions by monitoring data trends, performance indicators, and other metrics. Wexler et al. [15] describe a dashboard as "a visual display of data used to monitor conditions and/or facilitate understanding". Sarikaya et al. state that "the dashboard concept has evolved from single-view reporting screens to include interactive interfaces with multiple views and purposes, including communication, learning, and motivation, in addition to the classic notions of monitoring and decision support" and their study, conducting an extensive review of existing solutions and their characterisation reflects the differences between dashboards, highlighting the custom needs in terms of indicators, designs, components and goals [11]. The critical role of data visualisation has been increasingly recognised in recent years. A particularly notable example is the Johns Hopkins COVID-19 Dashboard, developed by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University. This dashboard gathered attention and acclaim for its significance, innovation, and impact in providing real-time insights and updates on the progression of the COVID-19 pandemic. In recognition of its contribution, the CSSE was awarded the Making a Difference Award at the 2020 Esri User Conference [8].

## III. *UrsoVerde* Overview

### A. Mobile Application

The UrsoVerde mobile application was developed using .NET MAUI, a C# framework that supports both iOS and Android. The application was built using .NET MAUI to replace the previous Xamarim version since the technology was no longer supported. The new application retains the user-friendly features and usability of its predecessor, now with a focus on the reward system, and the ability to add new rewards by including the City Hall to the workflow of reward management using a Dashboard. The main features of the platform include user creation and login, a rewards system for practicing sustainable habits from different activities, as well as tracking and displaying user statistics.

Sustainable habits are assigned to users through Objectives. Each Objective is distributed throughout the city in specific locations and can be completed by scanning QR Codes using the smartphone camera at the respective Objective location.

The app accesses the smartphone's location to verify the scan location during QR Code reading, in order to prevent fraud when the user is not near the perimeter of the QR Code site. Each time an Objective is complete, the user wins "Citizen Points."

The points earned can be exchanged for rewards. Each reward obtained is associated with a code generated when redeeming the reward in the application, which is validated by operators at the locations described in the reward details.

A preview of the application and its different pages can be seen in Figure 1.

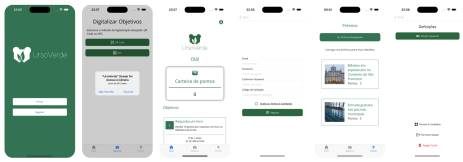


Fig. 1. *UrsoVerde* App on iPhone

### B. Dashboard and Server

Additionally, there is a management dashboard for authorised administrators where it is possible to manage the ecosystem, whether by adding or removing objectives or prizes, or managing users and statistics, such as counting how many users are completing each objective or how many objectives each user has completed. A Flask server intermediates all the communication, connecting both interfaces to the database and handling all back-end logic and data management.

Given the budget for the project and its main objective of testing *UrsoVerde* in the real world with a small, controlled group, the focus at this stage was on using QR codes only, while ensuring versatility for easy future upgrades to other methods of completing objectives.

The proposed custom dashboard should allow administrators on different levels, enabling CMC partners to validate the redeemed prizes by the users and empowering the main administrator of the platform to access and analyse critical information at a glance. Interactive visualisations will present data in a clear and compelling manner, facilitating the identification of trends, patterns, and anomalies. Real-time data access enables proactive decision-making by allowing users to monitor performance and identify potential issues as they arise.

A Flask server will act as the intermediary of this ecosystem. It will serve as a centralised data hub, providing a single point of access for both the dashboard and the mobile application to interact with the database. This way, data redundancy is eliminated and the whole system operates with up-to-date information.

The server will also enhance security by acting as a mediator for communication between the database layer and the clients, handling user authentication and ensuring all accesses are trustworthy and authorised. It will handle all complex back-end logic, allowing the dashboard and the mobile application to focus on providing a user-friendly experience.

The server will be built with scalability in mind, capable of handling increased user traffic and data volume as the project grows. Security is one of the main concerns as the server should be employing robust authentication and authorisation mechanisms to protect sensitive data within the system such as Oauth 2.0. Flask, chosen as the server framework, offers several advantages. Its clean syntax allows for rapid development and maintainability, while lightweight nature ensures efficient resource utilisation. Additionally, Flask's flexibility allows for customisation and integration with various libraries and tools to address specific project needs.

This proposed solution offers several benefits. The dashboard administrators will be provided with intuitive interfaces to access and interact with data and real-time information allowing users to make informed decisions based on accurate data. The chosen technologies ensure the application can grow in the future with the needs while maintaining robust security measures.

In conclusion, this proposal outlines the development of a comprehensive user-friendly dashboard and a robust intermediary server. By making use of the strengths of React, and Flask, this solution is a good start to *UrsoVerde* providing CMC with the tools needed to achieve success.

### C. Activities

After the first meeting with the City Hall, the requirements for the project were discussed and defined. In contrast to previous work, the City Hall opted for a more budget-conscious approach for the first phase, utilizing QR codes instead of IoT devices. It was decided to implement the activities in Figure 2 in the project.

| ACTIVITIES |
| --- |
| Participation in extracurricular activities |
| School Library Book Request |
| Participation in volunteering actions at school level |
| Shopping at D. Pedro V Market |
| Adoption of Animals from the Municipal Kennel/Cattery |
| Meals at restaurants at Mercado D. Pedro V |
| Use of the School Cafeteria |
| Shopping at School Stationery |
| Use of Recycling bins |
| Use of Public Transportation (SMTUC) |

Fig. 2. Activities

Activities that required an interaction with the person responsible for the space or activity such as participation in extracurricular activities or volunteering, school library book requests, adoptions at the Municipal Kennel/Cattery, meals at the school cafeteria, shopping at the school stationary, and activities at Mercado D. Pedro V would be registered using QR codes and would be implemented in the first phase of the

pilot tests. QR codes change periodically, and the application uses location services to avoid fraud. Activities that cannot be monitored such as the use of recycling bins and public transportation would be registered through the IoT devices using Bluetooth and would be implemented in a second phase of the pilot tests. The number of points attributed would change according to the activity. An activity such as adopting an animal would be more weighted than requesting a book from the school library.

### D. Architecture

The app communicates securely with a server that interacts with a Firebase database to store and retrieve data using HTTPS, as shown in Figure 3. The system also has a dashboard that allows administrators of the system to add or edit objectives and prizes, as well as to see user and system statistics.

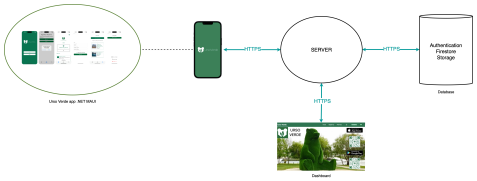



Fig. 3. Architecture

## IV. *UrsoVerde* Implementation

The UrsoVerde app establishes a secure and efficient connection with a HTTPS server developed using the Flask framework. Flask is a lightweight and flexible Python web framework, it enables rapid development of the server-side logic, including API endpoints for data exchange, request handling, and authentication mechanisms.

The app communicates with the Flask server through a REpresentational State Transfer API (REST). REST provides a standardized way for apps to send requests (e.g., GET, POST, PUT, DELETE) and receive structured responses in JavaScript Object Notation (JSON) format from the server.

This server, hosted in the Department of Informatics Engineering of Coimbra University, acts as the central hub for data processing, user authentication, and other critical backend operations for the app. For example, the server does the QR Code processing to verify if a QR Code is valid, relieving the app of the computational burden.

The users' mobile application allows the viewing of all the objectives and prizes. On each of these pages, by clicking on a specific objective or prize, it is possible to consult the individual information for each objective or prize. The mobile application also features a scanning page, allowing the users to scan the QR codes. In order to avoid potential fraud, these QR codes must obey some rules such as proper location, the maximum number of times each user is allowed to perform a certain objective, and a pre-defined time that the user must wait between performing the same objective two different times. If the user respects all the requirements, the points will be credited to a wallet tied to a category. Both the objectives and prizes are tied to a category, which means that to claim a prize, the user has to perform actions related to that same category. When redeeming the prizes, the users will receive a random code, which they can consult in the app. With one of these single-use codes, the users can redeem a prize directly with the person responsible for handling that specific prize. The pages of the application are displayed in Figures 4 and 5.

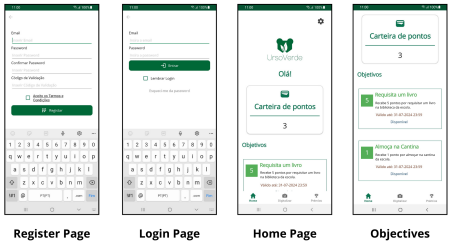



Fig. 4. Application Pages

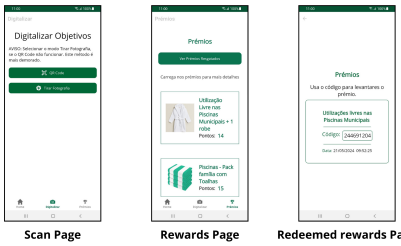



Fig. 5. Application Pages

The people responsible for handling the prizes to the users have access to a limited version of the dashboard6, which allows them to insert the users' codes and register them as handled. The full version of the dashboard allows the management of the system, adding or removing objectives and prizes, and viewing the history of the users.

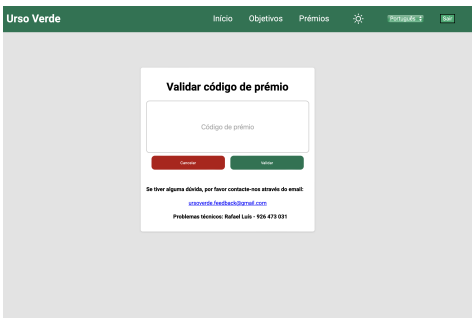



Fig. 6. Dashboard Redeeming Prize Page

### A. Application Flux

Application flux provides a comprehensive overview of the app's internal workings from a user-centric perspective. The diagram illustrated in Figure 7 shows the application's flux.

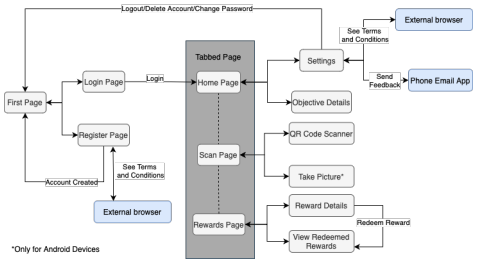



Fig. 7. Application Flux Diagram

The flux of the application is as follows:

1) The user launches the app and is presented with the First Page. He chooses either to navigate to the Login Page (if they have an existing account) or to the Register Page(to create a new account).
2) Upon successful login, the user is directed to the main app area Tabbed Page, landing on the default page: Home Page where information about the user's wallets and the objectives are displayed.
3) From the Tabbed Page, users can freely navigate to the Scan Page where they can Scan QR Codes and Take a Picture of the QR Code (only on Android) and to the Rewards Page where the they can view and redeem rewards.
4) The Settings Page can be accessed from the Home Page. In this Page users can see the terms and conditions of the app where it opens an external browser, send feedback where it opens the email app and take actions regarding their account:change password, delete account and logout. When one of this actions is done, the app goes back to the First Page.

## V. Assessment

This project incorporated two distinct testing phases: internal testing conducted at the Department of Electrical and Computer Engineering (DEEC) with the research group, followed by a pilot test implemented at School Jaime Cortesão with a select group of students.

### A. Internal Tests

The internal testing phase involved a group of six students and two professors from DEEC, who utilized beta versions of the app to assess its functionality and usability.

The application's Objectives were modified for the participating group, incorporating tasks such as turning off lights and closing doors scattered throughout the group's room. This adaptation enhanced engagement and allowed for a more comprehensive evaluation of the app's functionality in a controlled environment. Additionally, the rewards system was modified to make the experience more enjoyable and incentivizing for testers.

No significant issues were encountered during the internal tests.

### B. Tests at School Jaime Cortesão

The pilot test at School Jaime Cortesão started on May 27th and concluded on June 14th. Following a meeting with the school principal, parameters for the application were established, and necessary bureaucratic considerations were addressed. It was determined that only the "Lunch at the school cafeteria" and "Request a book at the school library" Objectives, would be active and they needed to be scanned only once to be completed. The Rewards would be from the City Hall, and only available during the months of June and July. A total of 31 rewards were available for the pilot test, allowing for students to redeem more than one prize. A single class of twenty-two tenth-grade students was selected to participate. Given the students' minor status, parental consent was required for app usage and test participation. The app was further modified to restrict access solely to authorized students through the implementation of a validation code on the registration page.

Ultimately, nine out of the twenty-two students participated in the pilot test.

The pilot test encountered initial challenges due to a miscommunication between the responsible teacher and the students regarding the app installation links. Some students were also unaware of the application due to their absence during the initial presentation, and a lack of communication from either their classmates or the teacher regarding the project's existence.

In the first week of the tests, it was discovered that certain Chinese-brand Android phones experienced difficulties reading QR codes. To address these issues, it was added an option only to Android devices to take a photo of the QR Code to send to the server alongside with the location of the phone, so the server can decode the QR Code instead of the phone. However, this proved unpopular due to taking a long time and needed precise photo capture. Further investigation revealed that QR code placement significantly impacted the correct scan of the QR Codes. To the students with problems in the first days, the points were manually added so no student was disadvantaged.

The photo-based solution also led to crashes on some Chinese-brand devices, and it was necessary to find a short notice solution: emailing QR code photos for manual point updates in the Database. While functional, this method was less efficient, which lead to less interaction with the application.

Following the initial week of testing, participation rates were lower than anticipated. As a result, the points awarded per Objective were increased to incentivize further engagement.

The pilot test concluded with a final meeting where students were invited to share their feedback and overall experience, providing valuable insights of the application.

The highest score achieved was 21 points, while the lowest was 1 point, attributed to a student who did not actively participate in the testing.

A week after the pilot test concluded, only one student redeemed a reward.

### C. Results

From the data gathered from tests and student inquiries, an analysis of the results is important to see how the *UrsoVerde* project performed.

The application was installed on 16 devices, with a majority on Android (12 devices) and a smaller number on iOS (4 devices). The Android version encountered two distinct failures during testing, one related to permission checks on older Android versions and the other related to image uploads. No failures were detected on Apple devices, indicating good performance on this platform during the pilot phase.

Feedback from seven students from Jaime Cortesão School highlighted both positive and negative aspects of their experience. Most of the students found the app easy to use (Figure 8), but satisfaction with QR code scanning was mixed (Figure 9), because of the technical issues identified. The points system was generally well-received (Figure 10), but rewards were considered unappealing by most (Figure 11). Despite these concerns, overall satisfaction for the app was moderate, with an average rating of 3.7 out of 5 (Figure 12). Users suggested improvements in QR code scanning, more appealing rewards, and more incentives.

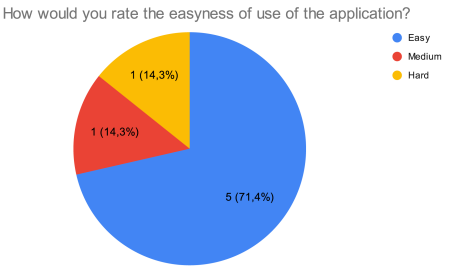



Fig. 8. Easiness of Use

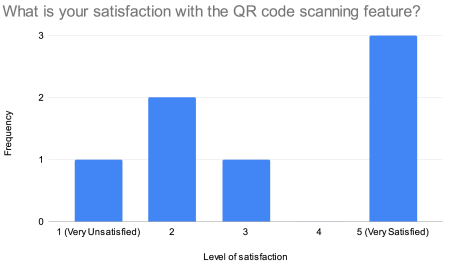



Fig. 9. Use of QR Code Feature

For the duration of the project neither the server nor the database client reported any issues or problems. However, it is important to consider the work load. In future tests, with increased app usage, more prizes and additional, upgrading the database to a paid subscription may be necessary. This upgrade would support increased read and write operations, as well as cloud functions, potentially reducing HTTP requests between the server and mobile app. Equipping the server

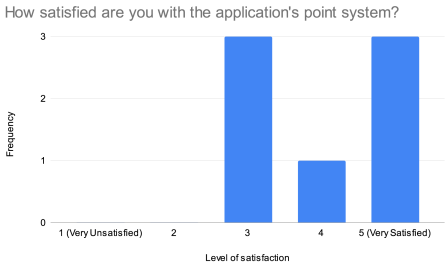



Fig. 10. Point System Satisfaction

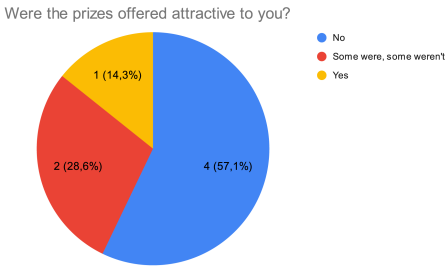



Fig. 11. Rewards Satisfaction

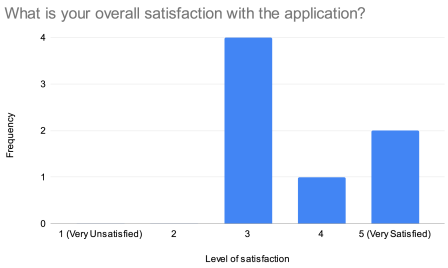



Fig. 12. App Satisfaction

with a Graphics Processing Unit (GPU) could enable faster processing for scanning multiple QR codes simultaneously in the server instead of in the application. Overall, the study was successful, and the back-end for the application performed as expected during the pilot project. While the dataset was limited due to the study's small sample size, it provided valuable insights that can inform future initiatives with larger populations, expanded prize offerings, broader objectives, and diverse environments.

## VI. Conclusions

The *UrsoVerde* app shows potential due to its ease of use and by being a reward-based app, but improvements are needed.

In future versions, more functionalities can be added in the near future. These include other forms of completing the objectives such as the Bluetooth beacons (implemented in past versions of Urso Verde), NFC or complements of more than one.

Also the inclusion of a chatbot could be interesting if capable of informing the users of the conditions to claim prizes or complete objectives.

The inclusion of a leaderboard could lead to an increase of interest and consequentially engagement from the users.

Unlocking the already implemented functionality of multiple wallets tied to different partners could lead to an increment of prizes and objectives, making the application more appealing. In this scenario, these partners must have access to the dashboard features of managing the objectives and prizes (only the ones that are tied to them).

Finally, getting more partners with more appealing prizes and objectives is essential. One of the conclusions of this study is that the interest in Urso Verde was highly correlated with the offered prizes. More objectives is also fundamental for this project. With more objectives to complete, the engagement will also increase. Targeted marketing to a specific public is also something to consider given that it may increase the interest and consequentially, the number of active users.

## References


[1] A. Adel. "Unlocking the Future: Fostering Human–Machine Collaboration and Driving Intelligent Automation through Industry 5.0 in Smart Cities". In: *Smart Cities* 6 (2023), pp. 2742–2782. DOI: 10.3390/smartcities6050124. URL: https://doi.org/10.3390/smartcities6050124.

[2] María Cortés-Cediel, Iván Cantador, and Manuel Pedro Rodríguez Bolívar. "Analyzing Citizen Participation and Engagement in European Smart Cities". In: *Social Science Computer Review* 39 (Oct. 2019), p. 089443931987747. DOI: 10.1177/0894439319877478.

[3] Igor Dunayev, Nataliia Gavkalova, and Aleksander Kud. "Designing a platform-based model of civic participation within the smart-city concept for post-war Ukrainian cities". In: *Eastern-European Journal of Enterprise Technologies* 4 (Aug. 2023), pp. 46–56. DOI: 10.15587/1729-4061.2023.285448.

[4] Diego Garaialde, Anna Cox, and Benjamin Cowan. "Designing Gamified Rewards to Encourage Repeated App Selection: Effect of Reward Placement". In: *International Journal of Human-Computer Studies* 153 (Apr. 2021), p. 102661. DOI: 10.1016/j.ijhcs.2021.102661.

[5] Google. *Google Opinion Rewards*. Accessed: 20 June 2024. 2024. URL: %7Bhttps://surveys.google.com/google-opinion-rewards/%7D.

[6] Hsi-Peng Lu and Hui-Chen Ho. "Exploring the impact of gamification on users' engagement for sustainable development: A case study in brand applications". In: *Sustainability* 12.10 (2020), p. 4169.

[7] MB Way. *Raspa e Ganha*. Accessed: 20 June 2024. 2024. URL: %7Bhttps://www.mbway.pt/mb-way-up/#:~:text=Raspa%20e%20ganha-,Raspa%20e%20ganha,-Pague%20com%20MB%7D.

[8] Greg Milner. "Creating the Dashboard for the Pandemic". In: *ArcUser* (2020), pp. 56–59.

[9] Recicla. *Esta Nova App Incentiva a Reciclar e Dá Vales em Supermercados*. Accessed: 20 June 2024. 2023. URL: %7Bhttps://recicla.pt/ideias-sustentaveis/esta-nova-app-incentiva-a-reciclar-e-da-vales-em-supermercados/%7D.

[10] Santander. *Aplicações para Ganhar Dinheiro*. Accessed: 20 June 2024. 2024. URL: %7Bhttps://www.santander.pt/salto/aplicacoes-para-ganhar-dinheiro%7D.

[11] Alper Sarikaya et al. "What Do We Talk About When We Talk About Dashboards?" In: *IEEE Transactions on Visualization and Computer Graphics* 25.1 (2019), pp. 682–692. DOI: 10.1109/TVCG.2018.2864903.

[12] Sweatcoin. *Sweatcoin*. Accessed: 20 June 2024. 2024. URL: https://sweatco.in.

[13] United Nations. *United Smart Cities*. Accessed: 20 June 2024. URL: %7Bhttps://sdgs.un.org/partnerships/united-smart-cities-usc%7D.

[14] United Nations Statistics Division. *Sustainable Development Goals Report 2023: Special Edition*. Accessed: 20 June 2024. URL: %7Bhttps://unstats.un.org/sdgs/report/2023/%7D.

[15] Steve Wexler, Jeffrey Shaffer, and Andy Cotgreave. "Front Matter". In: *The Big Book of Dashboards*. Ed. by Steve Wexler, Jeffrey Shaffer, and Andy Cotgreave. Wiley, 2017. DOI: 10.1002/9781119283089.fmatter. URL: https://doi.org/10.1002/9781119283089.fmatter.

# Appendix D

## Third party Libraries

- Server third party libraries
  - **appnope** 0.1.4
  - **APScheduler** 3.10.4
  - **asttokens** 2.4.1
  - **attrs** 23.2.0
  - **backcall** 0.2.0
  - **beautifulsoup4** 4.12.3
  - **bleach** 6.1.0
  - **blinker** 1.7.0
  - **CacheControl** 0.14.0
  - **cachetools** 5.3.3
  - **certifi** 2024.2.2
  - **cffi** 1.16.0
  - **charset-normalizer** 3.3.2
  - **click** 8.1.7
  - **contourpy** 1.2.1
  - **cryptography** 42.0.5
  - **cycler** 0.12.1
  - **decorator** 5.1.1
  - **defusedxml** 0.7.1

- **docopt** 0.6.2
- **executing** 2.0.1
- **fastjsonschema** 2.19.1
- **filelock** 3.14.0
- **firebase-admin** 6.5.0
- **Flask** 3.0.3
- **Flask-Cors** 4.0.0
- **Flask-SSLify** 0.1.5
- **fonttools** 4.52.4
- **fsspec** 2024.5.0
- **gcloud** 0.18.3
- **google-api-core** 2.18.0
- **google-api-python-client** 2.125.0
- **google-auth** 2.29.0
- **google-auth-httplib2** 0.2.0
- **google-cloud-core** 2.4.1
- **google-cloud-firestore** 2.16.0
- **google-cloud-storage** 2.16.0
- **google-crc32c** 1.5.0
- **google-resumable-media** 2.7.0
- **googleapis-common-protos** 1.63.0
- **grpcio** 1.62.1
- **grpcio-status** 1.62.1
- **gunicorn** 21.2.0
- **httplib2** 0.22.0
- **idna** 3.6
- **ipython** 8.12.3
- **itsdangerous** 2.1.2
- **jedi** 0.19.1

- **Jinja2** 3.1.3
- **jsonschema** 4.21.1
- **jsonschema-specifications** 2023.12.1
- **jupyter_client** 8.6.1
- **jupyter_core** 5.7.2
- **jupyterlab_pygments** 0.3.0
- **jwcrypto** 1.5.6
- **jws** 0.1.3
- **kiwisolver** 1.4.5
- **MarkupSafe** 2.1.5
- **matplotlib** 3.9.0
- **matplotlib-inline** 0.1.6
- **mistune** 3.0.2
- **mpmath** 1.3.0
- **msgpack** 1.0.8
- **nbclient** 0.10.0
- **nbconvert** 7.16.3
- **nbformat** 5.10.4
- **networkx** 3.3
- **numpy** 1.26.4
- **oauth2client** 4.1.3
- **opencv-python** 4.9.0.80
- **packaging** 24.0
- **pandas** 2.2.2
- **pandocfilters** 1.5.1
- **parso** 0.8.4
- **pexpect** 4.9.0
- **pickleshare** 0.7.5
- **pillow** 10.3.0

- **pipreqs** 0.5.0
- **platformdirs** 4.2.0
- **prompt-toolkit** 3.0.43
- **proto-plus** 1.23.0
- **protobuf** 4.25.3
- **psutil** 5.9.8
- **ptyprocess** 0.7.0
- **pure-eval** 0.2.2
- **py-cpuinfo** 9.0.0
- **pyasn1** 0.6.0
- **pyasn1_modules** 0.4.0
- **pycparser** 2.22
- **pycryptodome** 3.20.0
- **Pygments** 2.17.2
- **PyJWT** 2.8.0
- **pyparsing** 3.1.2
- **pypng** 0.20220715.0
- **Pyrebase4** 4.7.1
- **pytesseract** 0.3.10
- **python-dateutil** 2.9.0.post0
- **python-jwt** 4.1.0
- **pytz** 2024.1
- **PyYAML** 6.0.1
- **pyzbar** 0.1.9
- **pyzmq** 25.1.2
- **qrcode** 7.4.2
- **qrdet** 2.4
- **qreader** 3.12
- **quadrilateral-fitter** 1.8

- **referencing** 0.34.0
- **requests** 2.29.0
- **requests-toolbelt** 0.10.1
- **rpds-py** 0.18.0
- **rsa** 4.9
- **scheduler** 0.8.5
- **scipy** 1.13.1
- **seaborn** 0.13.2
- **shapely** 2.0.4
- **six** 1.16.0
- **soupsieve** 2.5
- **stack-data** 0.6.3
- **sympy** 1.12
- **thop** 0.1.1.post2209072238
- **tinycss2** 1.2.1
- **torch** 2.3.0
- **torchvision** 0.18.0
- **tornado** 6.4
- **tqdm** 4.66.4
- **traitlets** 5.14.2
- **typeguard** 4.2.1
- **typing_extensions** 4.11.0
- **tzdata** 2024.1
- **tzlocal** 5.2
- **ultralytics** 8.2.25
- **uritemplate** 4.1.1
- **urllib3** 1.26.18
- **wcwidth** 0.2.13
- **webencodings** 0.5.1

  - **Werkzeug** 3.0.2
  - **yarg** 0.1.9

- Dashboard Third party
  - **crypto-js** – v4.2.0
  - **dotenv** – v16.4.5
  - **firebase** – v10.11.1
  - **flowbite-react** – v0.7.2
  - **html2canvas** – v1.4.1
  - **https-browserify** – v1.0.0
  - **i18next** – v23.8.2
  - **js-cookie** – v3.0.5
  - **leaflet** – v1.9.4
  - **lucide-react** – v0.321.0
  - **react** – v18.3.1
  - **react-dom** – v18.3.1
  - **react-i18next** – v14.0.1
  - **react-icons** – v5.2.1
  - **react-leaflet** – v4.2.1
  - **react-lorem-ipsum** – v1.4.10
  - **react-modal** – v3.16.1
  - **react-qrcode-logo** – v2.9.0
  - **react-router-dom** – v6.21.3
  - **react-scripts** – v5.0.1
  - **web-vitals** – v2.1.4