Serious Game with Augmented Reality - Programming on Indoor/Outdoor systems: A case study on *Galeria da Biodiversidade da Universidade do Porto*

Diogo Alexandre Silva Teixeira

WORKING VERSION

U. PORTO

FEUP **FACULDADE DE ENGENHARIA**
UNIVERSIDADE DO PORTO

Serious Game with Augmented Reality - Programming on Indoor/Outdoor systems: A case study on *Galeria da Biodiversidade da Universidade do Porto*

Diogo Alexandre Silva Teixeira

Mestrado Integrado em Engenharia Informática e Computação

July 23, 2021

Resumo

Os museus são instituições culturais dedicadas a preservar e interpretar as provas tangíveis primárias da humanidade e do ambiente. Como um dos objetivos principais dos museus é educar a população, envolver os visitantes é fundamental e por isso têm surgido novas formas de captar a atenção dos visitantes. Os jogos pervasivos oferecem uma forma de melhorar a experiência de um utilizador em qualquer contexto, pois estendem a experiência do jogo para o mundo real, combinando o espaço físico e com o digital. Ao estender a experiência de jogo para o mundo real, tal como durante uma visita num museu, o utilizador torna-se mais do que um observador, tornando-se num jogador que interage com os artefactos no espaço à sua volta.

No entanto, atualmente os museus ainda adotam métodos tradicionais de exibição das suas exposições, que nem sempre captam a atenção dos visitantes de forma eficaz, especialmente para o público mais jovem. No caso da Galeria da Biodiversidade, as vitrines com folhas não permitem nenhum tipo de interação. Para além disso, a ligação entre as folhas em exposição e as árvores plantadas no exterior, no Jardim Botânico, passa despercebida se os visitantes não estiverem numa visita guiada.

O objetivo deste projeto é explorar soluções tecnológicas de Realidade Aumentada que estabeleçam uma ligação entre espaços interiores e exteriores. Isto requer diferentes tipos de acesso a localização para Realidade Aumentada (RA), tais como GPS ou beacons Bluetooth. Este projeto permite a ludificação da experiência do museu, e simultaneamente estabelecer a ligação entre as folhas em exposição na Galeria da Biodiversidade e as suas respetivas árvores plantadas no exterior, no Jardim Botânico, melhorando tanto a atratividade da exposição como o conhecimento adquirido pelos visitantes.

Um método de avaliação foi estabelecido para que os voluntários pudessem testar e partilhar comentários, enquanto simultaneamente a aplicação era observada para detetar possíveis erros. A aplicação foi testada na exposição das vitrines das folhas na Galeria da Biodiversidade e no Jardim Botânico da Universidade do Porto. Os resultados obtidos mostraram que os voluntários podiam experienciar a aplicação na totalidade sem problemas, com umas raras exceções, que foram resolvidas posteriormente. Os testes também mostraram que os voluntários se sentam mais envolvidos com a exposição das folhas e que eram capazes de associar as árvores com as folhas que tinham encontrado anteriormente.

Abstract

Museums are cultural institutions dedicated to preserving and interpreting the primary tangible evidence of humankind and the environment. Since one of the main goals for museums is to educate the population, engaging visitors is paramount and so ways to grab the attention of visitors are being invented. Pervasive gaming offers a way to enhance the experience of a user in any context, as they extend the game experience to the real world, combining the digital space with the physical space. By extending the gaming experience to the real world, such as a museum visit, the visitors become more than observers, they become players who interact with the artefacts in the space around them.

However, museums still adopt traditional methods of displaying exhibits, which not always succeed in capturing the attention of visitors effectively, especially when it comes to a younger audience. In the case of *Galeria da Biodiversidade*, the showcases displaying leaves do not allow any kind of interaction. Additionally, the connection between the leaves in display and the trees planted outside, on *Jardim Botânico*, goes unnoticed if visitors are not on a guided tour.

The aim of this project is to explore methods of Augmented Reality (AR) that connect indoor and outdoor environments. This requires different access to localization for Augmented Reality, such as GPS or Bluetooth beacons. This project allows for the gamification of the museum experience, while also drawing a connection between the leaves in the exhibition and their respective trees planted outside of the *Galeria da Biodiversidade*, on the *Botanical Garden*, improving both the attractiveness of the exhibit and the knowledge acquired by the visitors.

An evaluation method was established so that volunteers could test and provide feedback, while the application was observed for possible errors. The application was tested in the exposition area of the leaves showcases in *Galeria da Biodiversidade* and on the exterior garden of *Jardim Botânico da Universidade do Porto*. The results showed that the volunteers could experience the full application without issues, with a few rare exceptions, such as leaves not being detected, which were later resolved. The tests also showed an increase in the engagement felt with the showcases by the volunteers and they were also able to connect the trees to the leaves they had found previously.

Contents

List of Figures

Abbreviations

AR   Augmented Reality
CV   Computer Vision
PG   Pervasive Game
RSS   Received Signal Strength

Chapter 1

Introduction

1.1 Context and Motivation

Museums are spaces used to collect, communicate and exhibit objects of historical, scientific, artistic or cultural interest, for the purposes of education, study and enjoyment[3]. For this reason, engaging visitors is paramount and so ways to grab the attention of visitors are crucial.

One of the new technological approaches is with pervasive gaming, since these types of apps offer new ways to enhance the experience of a user in any context. By extending the gaming experience to the real world, such as a museum visit, the user becomes more than an observer, the visitors become players who interact with the space around them.

With new technologies being developed, such as Augmented Reality, new innovative ways to communicate are established, from which arise new challenges to the design of interactive experiences. [4]

And with mobile phones currently being the primary gaming device [5] and currently dominating the AR world [6], the interest in AR pervasive games is high, as shown by the success of games such as Pokemon Go.

The experience of playing an AR game in a museum provides the visitor with a ludic moment, making it an innovative way to capture the attention of new audiences. At the same time, the technology could be used to improve the way information of the exhibits is displayed, and possibly even extend it, regardless of whether the location is indoors or outdoors, and also be used to provide new ways to experience them.

1.2 Problem and proposed solution

The main problem lies in the room with the leaves exhibit in *Galeria da Biodiversidade do Porto*, which despite being an interesting room, the lack of interactivity makes it less engaging especially when doing so without a guide, since it only allows observation. If visitors are uninterested in an exhibit, then they will not be compelled to learn more about the subject of the exhibit, and may just walk by it.

By creating more ways to interact with the exhibit, it becomes more attractive to the visitor, increasing their interest in the subject. On a related note, the leaves in question are of the same species as some of the plants located outside *Galeria da Biodiversidade*, on *Jardim Botânico do Porto*. This fact opens up possibilities in connecting the two spaces, and creates a more memorable visit.

The proposed solution plans on taking advantage of the relationship between the leaves in exposition and the trees outside in order to create an interactable activity which aims to increase the interest in both the exhibit and the garden separately and in the relationship between them. In order to do so, a serious game will be developed, which can be divided in two parts:

- The first part takes place indoors, in the room with the vitrines that contain the various species of leaves. Here, a player is asked to take photographs of the leaves indicated by the game. When a photograph is taken, the app compares and verifies if it is the correct leaf. Once all leaves are found, the first part of the game terminates.

- The second part begins outside, after the player leaves *Galeria da Biodiversidade*. Here, the player is invited to find the trees of the same species as the leaves whose photograph was taken. To help the user, the app functions as a radar, warning the player once he's nearby a point of interest. For each completed objective, the player is rewarded with a collectible. The final showcasing will show their achievement.

1.3 Objectives

The objective of this dissertation is to develop a location based AR game, which takes place in *Galeria da Biodiversidade* and on *Jardim Botânico*, which is one of the Core Poles of the Natural History and Science Museum of the University of Porto (MHNC-UP). This project will use locally placed beacons for indoor location and use GPS for outdoor location. The main purpose is to associate the leaves placed in the vitrines found inside *Galeria da Biodiversidade* with the trees planted in *Jardim Botânico*, gamifying the exhibit, which in turn increases its impact and attractiveness.

The main scientific challenges are:

- Indoor location - With most museums being partly or fully indoors, resorting to the most common method of location, GPS, can prove to be unwise, as the infrastructure make the location inaccurate. As such, an alternative method will be used, beacons placed inside the museum.

- Outdoor location - Although the outdoors do not have the museum's infrastructure to make GPS data inaccurate, other obstacles have to be taken into consideration, such as zones which are at a lower altitude than their surrounding areas. In the case of *Jardim Botânico*, the trees could make the data inaccurate, especially if they are swaying with the wind.

- Leaf identification - Accurately identify and distinguish leaves of different species using computer vision may prove challenging, especially if they are similar in appearance.

This investigation project is being developed as part of a Doctoral Program in Digital Media.

1.4   Dissertation Structure

The structure of this document has a total of five chapters, described here:

- Chapter 1 is the introduction;

- Chapter 2 describes the state of the art, regarding game design, location and computer vision for plant identification;

- Chapter 3 presents the application developed, showing the multiple sections in the game;

- Chapter 4 explains the more relevant sections of the implementation, the architecture and the more vital operations;

- Chapter 5 covers user testing, describing what was done to test the application and the results obtained.

- And Chapter 6 concludes the document and discusses future work.

Chapter 2

Literature Review

The development of a project such as this one requires technologies that gather data from the real world. These technologies should retrieve data regarding the location of the user, as well as process images regarding photos of leaves. As such, they must be discussed to understand what can be used for each objective.

2.1 Location

This section covers technology regarding location. Since the accuracy of location data can be greatly influenced by external conditions such as physical obstacles, the usage of technologies is investigated on how best to handle them.

2.1.1 GNSS

The Global Navigation Satellite System (GNSS) is broad term used to describe any satellite constellation that provides positioning, navigation and timing services on a global or regional basis, the most common one being the GPS.

The Global Positioning System (GPS) is one of the global satellite systems that provides geolocation and time information to a GPS receiver anywhere on or near Earth where there is an unobstructed line of sight to four or more GPS satellites. However, obstacles such as building and mountains block the relatively weak GPS microwave signals, broadcasted by the GPS satellites [7]. Additionally, since GPS was designed to be used on open sky with minimum interference, and so the canopies and trunks of the trees negatively affect the reception of the broadcasting signal, increasing the error[8]. The wind causes this canopy to move, causing multi-path effects, where the receiver receives multiple copies of the same GPS signal, because the signal bounced off several features[9].

The conditions in which data is acquired can be improved by mounting GPS field antennas on telescopic rods, partially compensating some of the loss in quality and accuracy of GPS data, but this is option is not always feasible. Since most smartphones have GPS, WiFi and Bluetooth, when the GPS signal is not available, WiFi and Bluetooth can be used to make up for it[10], which

works well in cities, where there are WiFi signals in most places and bluetooth low energy beacons are present in some places.

2.1.2   WiFi positioning system

The WiFi positioning system is a geolocation system which determines the location of a device by using nearby wireless access points[11], useful when GPS data is inaccurate or unobtainable. Typically, one of two techniques are used for determining the location of a device. On one hand, there are the propagation based techniques, in which a base station measures either the received signal strength, the Angle-of-Arrival of the direction of the strongest received signal or the time difference of arrival, of the received signal. This information is then used to compute the device's location[12]. On the other hand, there are the fingerprinting techniques, in which, during the offline phase, several access points are collected before used, being stored in a server as a function for the user's location. Then, at the online phase, the received signal strength is matched in the database by the server, in order to determine the current location of an access point[13].

2.1.3   Bluetooth Low Energy Beacons

Bluetooth beacons are hardware transmitters and a class of Bluetooth low energy (BLE) devices that broadcast their identifier to nearby portable electronic devices, by using one of the two states in which BLE has the ability to exchange data in: advertising mode [14], which uses the Generic Access Profile (GAP) layer to broadcast data out to any device that is listening. These beacons transmit an universally unique identifier [14], that is picked up by a compatible app or operating system, which, together with several other bytes, can be used to determine the location of the device with an accuracy of a few meters [15].

Bahl and Padmanabhan[16] presented a radio-frequency system for locating and tracking users inside buildings called RADAR, in which it first, durin the offline phase, recorded and processed signal strength information at multiple locations, from which the radio map is created. During the online phase, the device collects the signal strength, which is matched with the radio map to determine the possible position (see Fig.2.1.

Based on this, Yeh et al.[17] proposed a positioning technology based on a mobile device with GPS and WiFi-based pattern matching, in order to improve precision in situations of bad weather or when there is no clear line-of-sight. In the offline phase, the WiFi radio map is similarly made, however, the GPS coordinates are also recorded in a different database (see Fig.2.2). In the online phase, there are three positioning point sources: GPS, WiFi and the Selective Weight position (see Fig.2.3), the latter one combining GPS and WiFi positioning methods to obtain an adjusted positioned point.
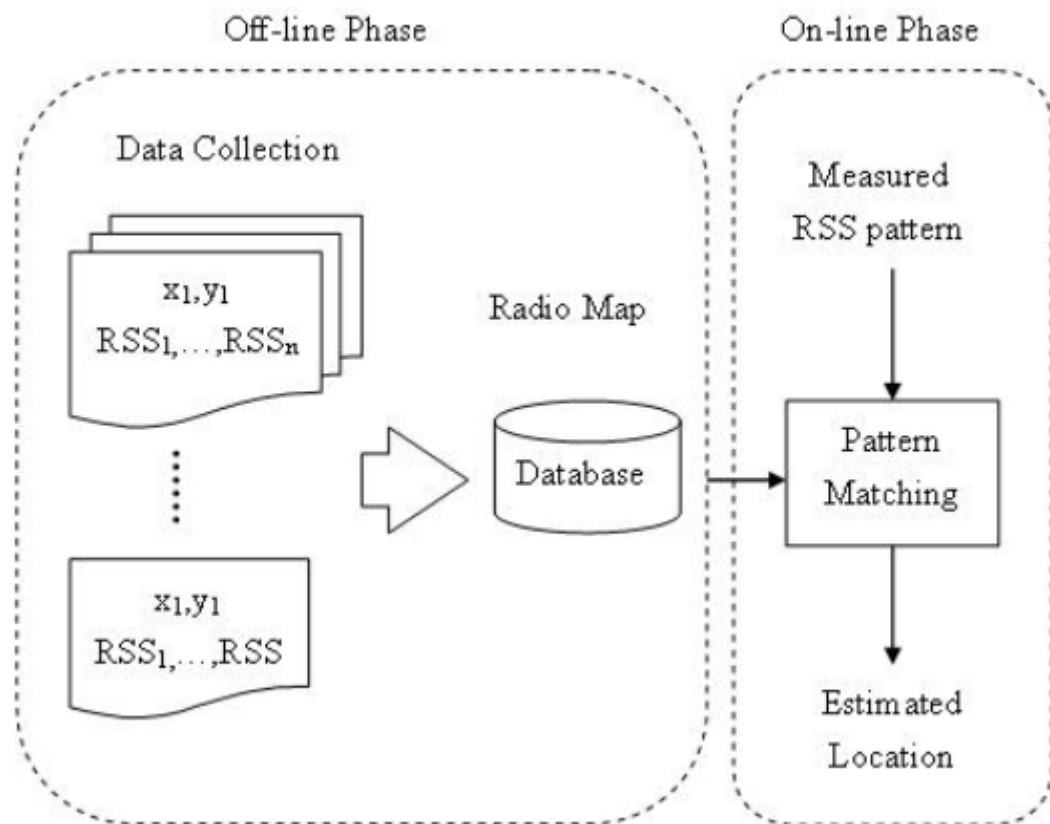
Figure 2.1: Method of Operation of the Radio Map
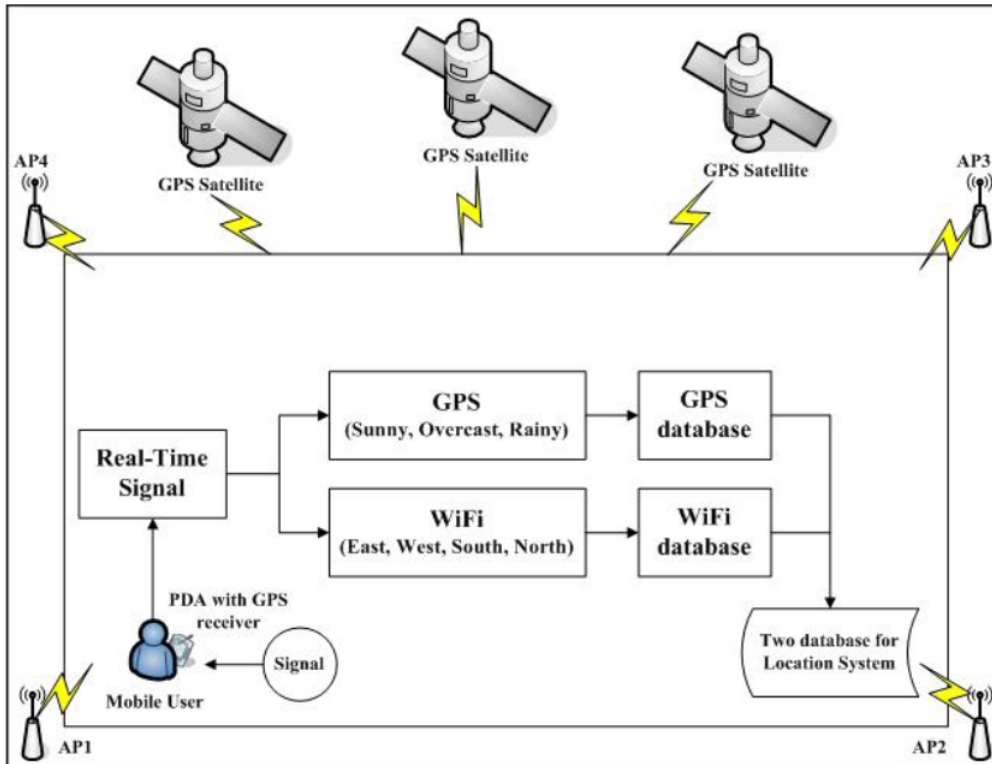
Figure 2.2: Offline Phase of Yeh et al.'s method
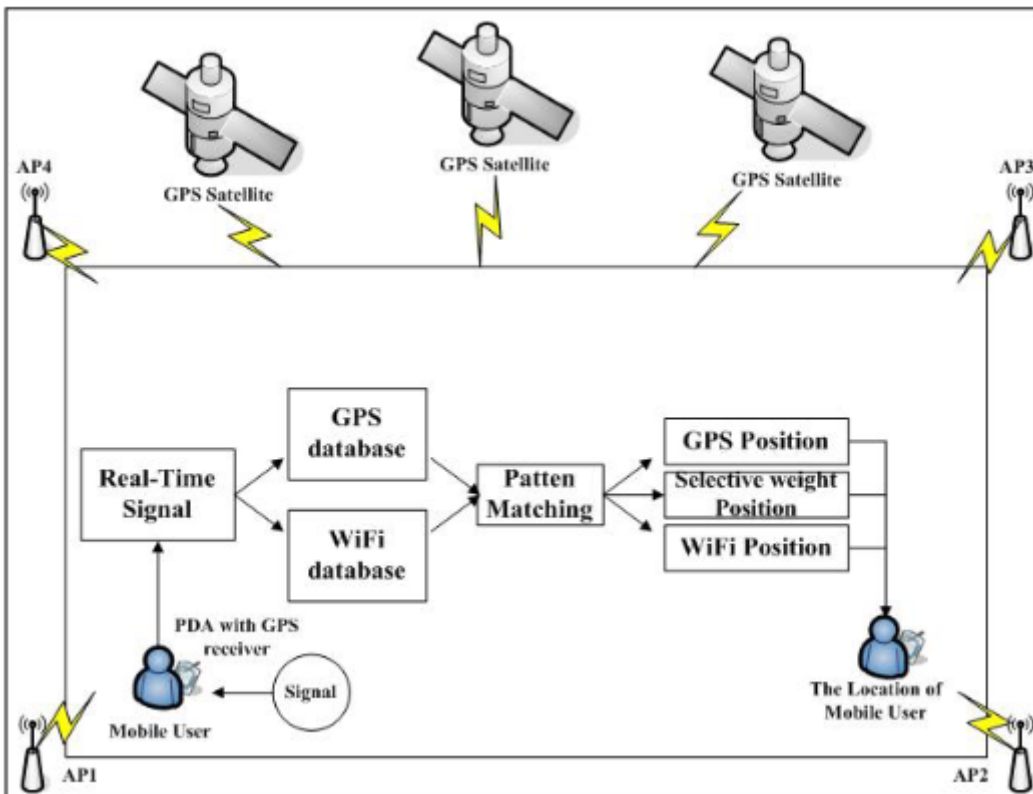
Figure 2.3: Online Phase of Yeh et al.'s method

2.2   Computer Vision for Plant Identification

Identifying plants manually is a time-consuming, complex task, which may be impossible to conduct by non-experts due to the use of specific botanical terms. Essentially, the process consists of answering a series of questions about the attributes of the unidentified plant, focusing on discriminating characteristics, eventually narrowing down the set of candidate species to one, the desired species.

Image-based methods were considered a promising approach for species identification [18] and nowadays, most automated identification systems rely on images depicting the species for identification [19].

An image classification process can typically be divided in the following steps [20]:

1. Image acquisition - Obtain an image of the whole plant or of its organs;

2. Preprocessing - Enhance image data to suppress distortions and emphasize relevant features by generating a modified image to be used in the next step. This step typically involves image denoising, content enhancement and segmentation.

3. Feature extraction and descriptors - Take measures of possibly segmented, meaningful regions in the image. Features are described by a set of numbers that characterize some property of the plant or its organ that was captured by the image.

4. Classification - Concatenate all the features that were extracted into a single feature vector and classify it.

Jana Wäldchen and Patrick Mäder [20] conducted a systematic literature review on this subject, analysing peer-reviewed studies between the years 2005 and 2015, and their findings relevant to this project are presented:

- Most studies used images with plain background, avoiding segmentation;

- Main research focus on leaf analysis for plant identification;

- Shape is the dominant feature for plant identification;

- Multi-feature fusion facilitates higher classification accuracy;

- Contour-based shape description more popular than region-based description;

- LeafSnap, Pl@ntNet, and Folia are the only publicly available implementations.

2.2.1   Image Acquisition

As mentioned above, the first step in image classification is acquiring said image. These are usually obtained as a photograph, whether on a simple background or a natural one, or through scanning plants. Most studies analysed by Jana Wäldchen and Patrick Mäder [20] avoid using photos on natural backgrounds as these types of photos require extra effort with dealing occlusions and overlaps (examples shown in fig. 2.4 by Yanikoglu et al. [21])

Figure 2.4: Samples from scan (first row), pseudo-scan (second row) and photo categories (third row).

2.2.2   Feature Detection and Extraction

Feature extraction is a process of dimensionality reduction by which an initial set of raw data is reduced to more manageable groups for processing by extracting characteristic patterns from the images, found in shapes, textures and color. There is a plethora of methods for describing features such as those described by Zhang et al. [22] for feature representation.

In feature extraction for images of leaves, shape was the pattern most used. Shape descriptors can be classified in two categories: contour-based and region-based. The former extracts shape features from the contour of the shape, while the latter obtains shape features from the whole region of a shape. There are other methods that don't fall in these categories but they are few. [20] Wu et al. [23] mentions five basic geometric features in leaves:

- diameter - the longest distance between any two points on the margin of the leaf;

- physiological length - The distance between the two terminals of the main vein of the leaf;

- physiological width - By drawing a line passing through the two terminals of the main vein, one can plot infinite lines orthogonal to that line. The number of intersection pairs between those lines and the leaf margin is also infinite. The longest distance between those intersection pairs is defined as physiological width. The relation between physiological length and width is illustrated in figure 2.5;

- leaf area - As the name indicates, it's the measurement of the surface of the leaf, on an image, it's the number of pixels that are of a leaf;

- leaf perimeter - Similar to the leaf area, the perimeter is calculated by counting the number of pixels consisting of leaf margin.

From these features, 12 digital morphological features used for leaf recognition are defined, such as aspect ratio, form factor, rectangularity and narrow factor and perimeter ratio of physiological length and physiological width.

Hossain, Javed and Amin, M. Ashraful [24] propose a more leaf-specific descriptor, called leaf width factor (LWF), which is extracted from leaves by slicing across the major axis and parallel to the minor axis. Then, the LWF per strip is calculated as the ratio of the width of the strip to the major axis (see Fig. 2.6). The major and minor axis mentioned are the equivalent to Wu et al.'s [23] physiological length and width, respectively.

Additionally, Yanikoglu et al. [25] proposes a slight variation of the LWF with the introduction of an area width factor (AWF), which given an isolated leaf image, it is first divided into n strips perpendicular to its major axis, similar to the LWF method. Then, for the final n-dimensional features, the area of each strip normalized by the global area is computed.

Finally, Prasad et al. [26] proposes a reduced shape and color feature extraction method for mobile devices, where the original image is reduced to similar aspect ratio which does not affect shape information but reduces the computation cost nearly up to half of the total cost. The
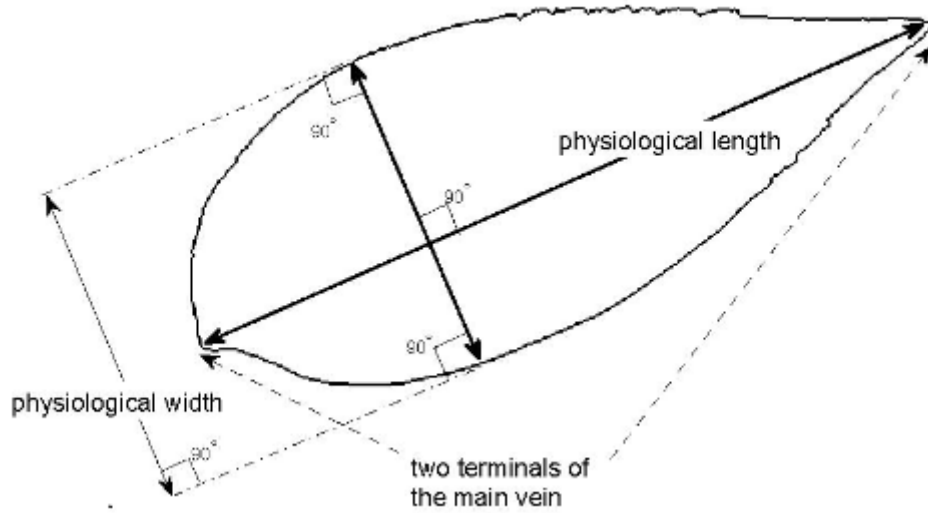
Figure 2.5: Relationship between Physiological Length and Physiological Width
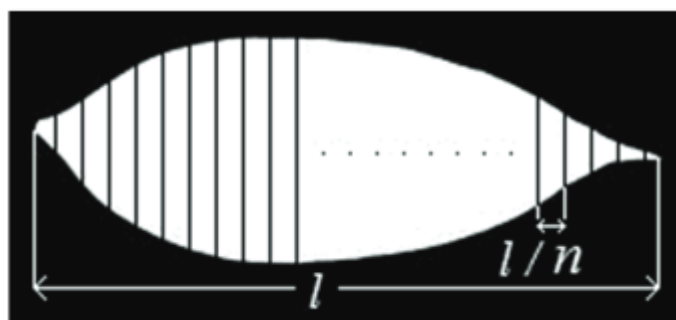
Figure 2.6: Leaf Width Factor (LWF) Extraction

proposed system effectively executed on Android mobile devices with an average response time of 0.52 seconds.

2.2.3 Mobile Applications

Leafsnap was one of the first apps to be developed in this domain, later followed by *Pl@ntNet* and *Folia*. Although their performance might not be enough for the identification of all plant species worldwide, they perform exceedingly well when regarding their databases, which host multiple images of hundreds of different plant species. In fact, according to an article [27] reports that "high levels of accuracy of *Leafsnap* identifications, as were confirmed by expert botanists, were used to map the geographic distribution of native and exotic species at a scale previously unachievable without this technology and without the aid of citizen scientists". However, these apps do require an internet connection as most of the work is performed in the server.

2.3 Game Design

Game design consists in applying design and aesthetics to create a game, whether they are board games, role-playing games or video-games. However, applications outside of games have been increasing in number, through gamification. Hunick et al.[1] proposed the MDA framework as a formal approach to understanding games, easing the process to decompose, study and design a broad class of game designs and artifacts. MDA stands for mechanics, dynamics and aesthetics, represented in Fig.2.7 and explained below:

- Mechanics - Mechanics are the actions, behaviours and control mechanisms players are afforded within the game context. The mechanics serve as support for the overall gameplay dynamics.

- Dynamics - Dynamics are the run-time behaviour of the game's mechanics acting on the player inputs and their outputs. Dynamics are responsible for creating the aesthetic experiences

- Aesthetics - Aesthetics is the desirable emotional response evoked in the player, when interacting with the game. The aesthetics are what turns the game into an enjoyable experience.

Figure 2.7: The three MDA levels[1]

2.3.1   Gamification

Gamification was summarised in an article published by Deterding et al. in 2011 as simply being "the use of game design elements in non-game contexts"[2], or, from the designer's perspective, gamification is different from other games, serious or for entertainment, because they are built with the intention of a system that uses game elements, but is not an actual game, even though the user can then experience them as if they were games[2]. This phenomenon is already applied in multiple aspects of life, such as loyalty points and travel miles, which aim to attract customers and increase engagement.

Deterding et al.[2] stresses that the affordances when implementing gamification have to be the same as the ones used in games, whereas Hamari and Huotari [28] emphasise the role of gamification lies simply in invoking the same psychological feeling that games deliver.

Deterding et al.[2] distinguish gamification from other related concepts via the two dimensions of playing/gaming and parts/whole, differentiating gamification from games and serious games with the former axis and from toys and playful design with the latter axis (see Fig.2.8), which shows gamification steers away from playing, defined by Caillois [29] as a more unstructured and spontaneous, and relates more to gaming, which places structure in play, with the use of rules.
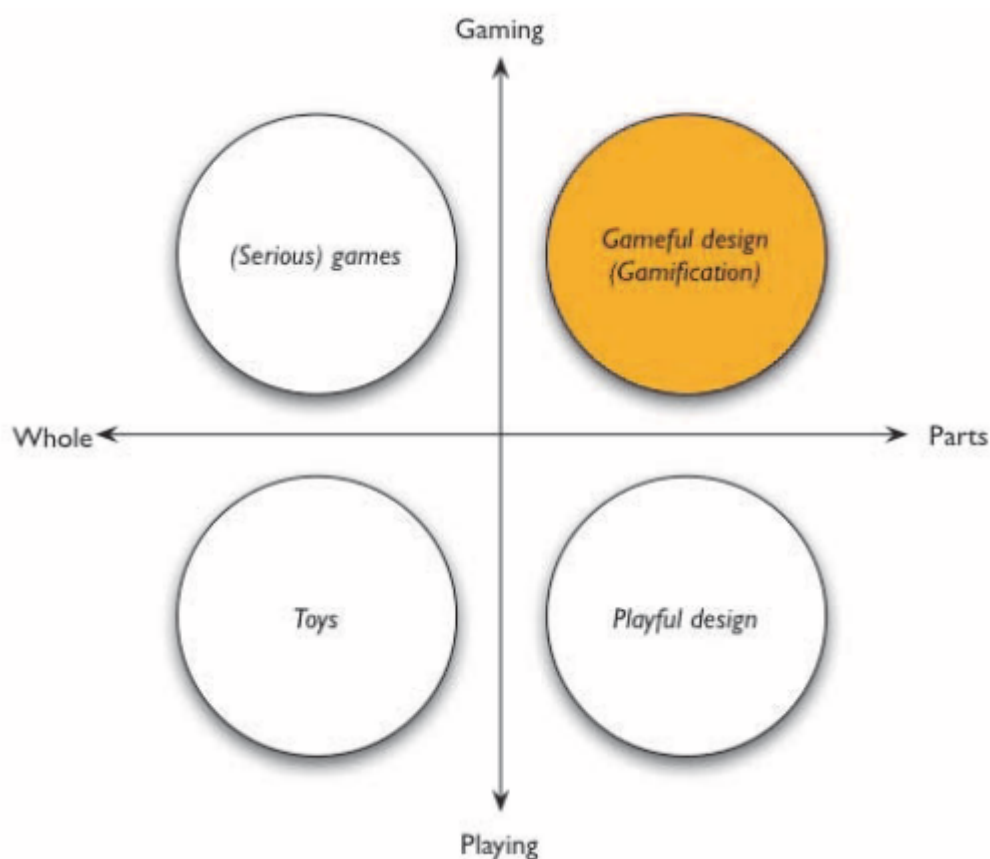
Figure 2.8: Situating gamification on game/play and whole/parts axis (by Deterding et al[2].

Deterding et al. also locate gamification in the larger field that is the ludification of culture,

better differentiating the definition from other concepts, as shown in Fig. 2.9, where the upper half shows the usage of games and bottom half the extension of games or the playful interaction. In the extension of games are the pervasive games, which are games that extend the concept of "magic circle"[30] of games, the space where the game world replaces the real world, to the surrounding environment of the player. As the Fig. 2.9 shows, full-fledged serious game can also be pervasive games. Finally, gamification is placed as the use of game elements, but not game technologies or game practices.
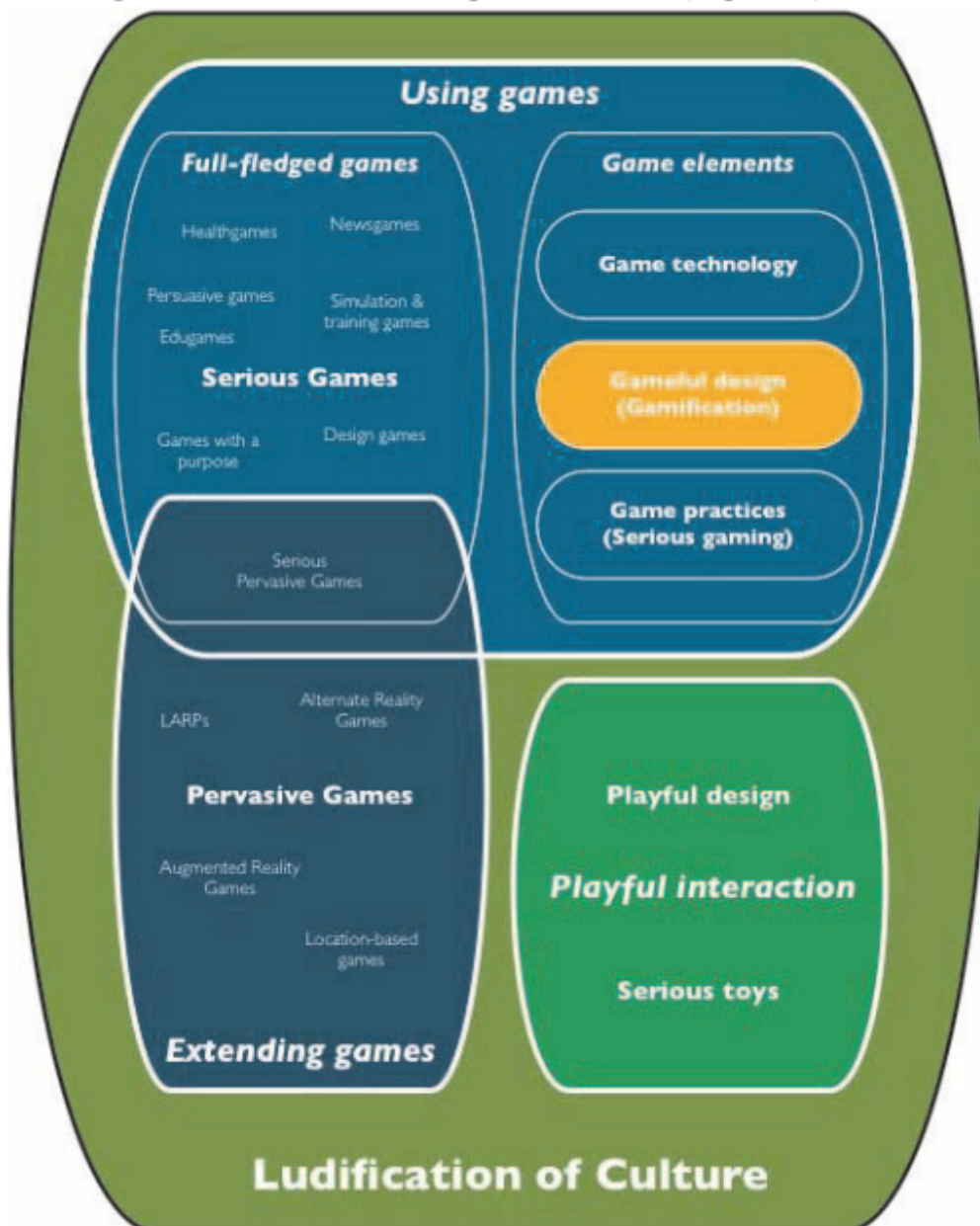
Figure 2.9: Gamification and the Ludification of Culture (by Deterding et al[2].

Robson et al.[31] defined gamification with "the application of lessons from the gaming domain to change behaviours in non-game situations", and reports that the increased interest in gamification

nowadays is due to three relatively recent developments:

1. Over the last two decades, research into what makes computer games engaging and successful was increased due to the growth and importance of the game industry, which developed theories and lessons about what motivates people to play.

2. The increased pervasiveness of social media and other applications changed the way individuals and organizations interact with one another, which made it possible to gather enormous amounts of data regarding people and their opinions, feeling and behaviour. which is useful to produce gamified employment or consumption experiences at scale, and create more data.

3. New knowledge about the design and management of gaming experiences, together with the increased interest in providing more engaging experiences and the advent of social media provide the possibility for new innovation in the way organisations connect and influence customers and employees.

Werbach et al.[32] structures game elements in a pyramid of three types of game elements, represented in :

- Dynamics - the dynamics are composed by the mechanics, related to the player experience in the game.

  – Constraints - the restrictions placed to the actions of the player.

  – Emotions - the emotions provided to the player.

  – Narrative - the narrative itself and the sequence of events that compose the plot.

  – Progression - the progress through the levels of the game and the sequential increase in difficulty.

  – Relationships - the relationships between players and between the player and the game.

- Mechanics - the mechanics develop the dynamics of the game, related to the rules of the game.

  – Challenges.   – Resource Acquisition.

  – Chance.   – Rewards.

  – Competition.   – Transactions.

  – Cooperation.   – Turns,

  – Feedback.   – Win States.

- Components - the components are used to create the mechanics.

- – Achievements.
- – Avatars.
- – Badges
- – Boss Fights.
- – Collections.
- – Combat.
- – Content Unlocking.
- – Gifting.

- – Leaderboards.
- – Levels.
- – Points.
- – Quests.
- – Social Graph.
- – Teams.
- – Virtual Goods.

A common approach to gamification is called "The PBL triad", which consists in points, badges and leaderboards, and are useful for multiple reasons:

- Points provide quick feedback on a user's performance as well as determine the progress in the game. Additionally, points can be used for determining win conditions and rewards, or to just simply keep track of the score of a player.

- Badges are normally used to represent an achievement the user has accomplished, a milestone they have reached or an obstacle they have overcome. As such, badges can be used to act as symbols of status and importance, or be used as conditions necessary to perform some actions, accessing content previously denied access. Moreover, the flexibility in its design can also increase the player's interest in acquiring them and are especially attractive to collectors.

- Leaderboards provide feedback on the competition by ranking the users. They can also be tailored to a user by ranking, for example, only between the user and their friends. Nevertheless, leaderboards come with the risk of possibly causing demotivation if the differences between the user and the competition are too large, especially in educational environments.

2.3.2 Serious Games

A serious game refers to a game designed for something other than pure entertainment purposes[33], used in areas such as education, health and politics. It is important to make the distinction that games that helped a player learn more about a subject does not make the game a serious game. For example, a shooter game in a WWII setting, whose main purpose is entertainment, but the player learned more about the history of the war, is not a serious game.

One early example of a serious video game used for education is Math Blaster, where the player is placed inside a spaceship and a mathematical equation is presented, with the result missing. The player then has to aim and shoot the object displaying the correct answer (see Fig.2.10). Simulator games are also common serious game , normally used for training, such as flight simulators.

Figure 2.10: Math Blaster!, a serious game for educational purposes.

As technology improves, serious games are more capable of contextualizing a player's experience in a realistic environment, which supports situated cognition [34]. This type of activity complements formal learning by encouraging exploration in various situations, with limited space and time[35]. Furthermore, since serious games can serve as simulations, training in situations which are dangerous, costly or in complex environments, can be performed in a more controlled manner[34], such as surgery simulations.

Emotion also takes part in the learning process. M.H. Immordino-Yang and A. Damasio wrote that while cognition and emotion should be distinguished as different aspects of the human functioning, it is unwise to separate them completely, as emotion is a part of both cognitive and sensory processes[36]. Additionally, the aspects of cognition that are used more often for education purposes, namely learning, attention, memory, decision making, motivation and social function, are not just greatly affected by emotion, but are actually included in the processes of emotion. And so, rational thought and logical reasoning cannot be recruited in a proper and useful manner in the real world without emotion, since emotions help direct our reasoning into the sector of knowledge that is relevant to the current situation [36]. This relationship between cognition and emotion is depicted in Fig.2.11, by M.H. Immordino-Yang and A. Damasio[36].
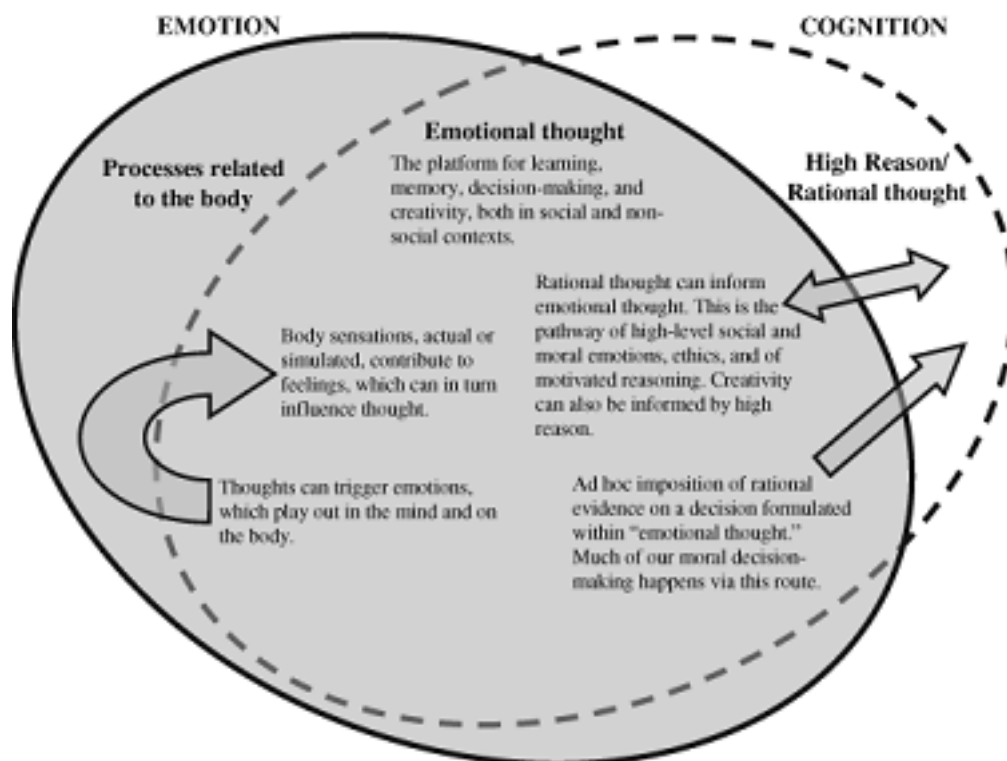
Figure 2.11: Neurological relationship between cognition and emotion.

So, serious games can enrich the learning experience beyond merely reading and listening, making room for a more active learning environment, but come with some concerns, especially in educational and training situations. De Gloria et al. warns that the 'suspension of disbelief', which normally happens when playing a game, may hinder the learning process; usability issues

and competition can cause frustration to the players; the desire to win the game can make the player to try to glaze over the knowledge being taught, defeating the purpose of the game, among other problems, reaching the conclusion that serious games in educational settings have to be organized properly, since designing a serious game is a complex challenge.

Nevertheless, the advantages for serious games are still present. The possibility for a more tailored learning experience that can be adapted to the progress of the user, together with a more immediate feedback to each action, a higher degree of knowledge retention and the attractiveness, especially to the younger audiences, make serious games become powerful educational tools[36].

2.3.3   Pervasive Games

Pervasive games, as defined by Benford et al.[37], are games in which the experience of the game is extended to the real world, whether indoors or outdoors. Sensors in mobile devices can be used to gather information such as the location of the player, and create a game that reacts to where the player is, unlike typical gaming platforms [37]. There are numerous examples of this type of game, probably the most famous one being *Pokemon GO*, where the player needs to physically move around the streets to find new *Pokemon* (see Fig.2.12, but the game also reacts to the weather and time of day of the user's location.
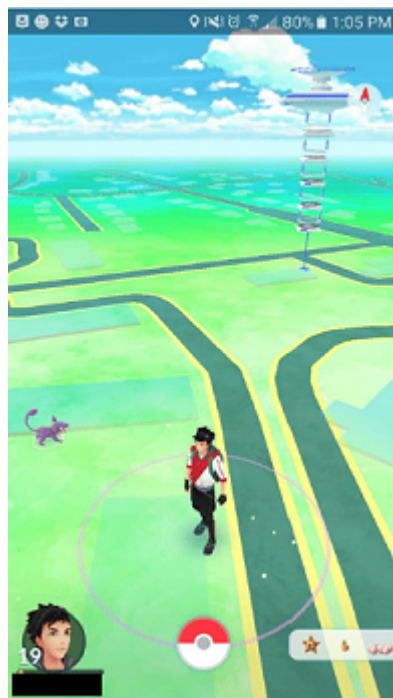
Figure 2.12: *Pokemon Go*'s mapping of the real world

Pervasive games of this type all use the following technologies[37]:

1. Sensors to gather information about the user's context, such as GPS positioning;

2. A wireless connection to a remote server (and other players, if applicable);

3. A display to deliver the digital content to the player, typically a mobile device.

Benford et al.[37] also reports one of the main problems with pervasive games, which they referred as "Dealing with uncertainty": since sensors and wireless communication are subjected to problems such as errors and jitter, and are greatly influenced by the conditions of the real world, for example, establishing a wireless connection in a congested area.

Sweetser and Wyeth[38] develop the GameFlow Model to be used to design and evaluate games focusing on how much is the player enjoying the game, show in Fig.2.13 and Fig.2.14.

However, L. Jegers[39] noted that this model was designed with traditional computer games in mind and does not take into account three dimensions related to pervasive gaming:

- Mobile/Place-Independent Gameplay - Pervasive gaming can be done outside the context of consoles and computers, but in nearly all contexts of everyday life;

- Social Interaction Among Players - Although not exclusive to pervasive games, as other online games allow for social interaction, the way social interaction can be done in pervasive games is unique;

- - Integrating Physical and Virtual Worlds - Unlike the exclusive focus given to the virtual world from traditional computer games, pervasive games integrate both worlds and expand their use.

This leads to Jegers' modified Gameflow Model, the Pervasive Gameflow Model, which has the following alterations[39]:

- Concentration - instead of trying to remove all distractions from the surroundings, pervasive games should facilitate switching concentration between the game and surrounding factors of importance;

- Challenge - pervasive games should support players in the creation of game scenarios as well as help keeping balance in the creation of paths and developments in the game world, without much constraint;

- Player skills - with the change of the previous point, which made progress more user-driven, the game should be very flexible and the pace at which players develop their skills should be set by the players themselves;

- Control - Since its in the nature of pervasive games for gameplay to be constant and ongoing, traditional shell control is out of the question, giving instead the ability for players to easily join the game and quickly recognize what changed since they last played;

- Clear goals - Pervasive games should allow players to form and communicate their intermediate goals;

- Feedback - no changes are needed;

Element	Criteria
Concentration Games should require concentration and the player should be able to concentrate on the game	- games should provide a lot of stimuli from different sources - games must provide stimuli that are worth attending to - games should quickly grab the players' attention and maintain their focus throughout the game - players shouldn't be burdened with tasks that don't feel important - games should have a high workload while still being appropriate for the players' perceptual, cognitive and memory limits - players should not be distracted from tasks that they want or need to concentrate on
Challenge Games should be sufficiently challenging and match the player's skill level	- challenges in games must match the players' skill levels - games should provide different levels of challenge for different players - the level of challenge should increase as the player progresses through the game and increases their skill level - games should provide new challenges at an appropriate pace
Player skills Games must support player skill development and mastery	- players should be able to start playing the game without reading the manual - learning the game should not be boring, but be part of the fun - games should include online help so players don't need to exit the game - players should be taught to play the game through tutorials or initial levels that feel like playing the game - games should increase the players' skills at an appropriate pace as they progress through the game - players should be rewarded appropriately for their effort and skill development - game interfaces and mechanics should be easy to learn and use
Control Players should feel a sense of control over their actions in the game	- players should feel a sense of control over their characters or units and their movements and interactions in the game world - players should feel a sense of control over the game interface and input devices - players should feel a sense of control over the game shell (starting, stopping, saving, etc.) - players should not be able to make errors that are detrimental to the game and should be supported in recovering from errors - players should feel a sense of control and impact onto the game world (like their actions matter and they are shaping the game world) - players should feel a sense of control over the actions that they take and the strategies that they use and that they are free to play the game the way that they want (not simply discovering actions and strategies planned by the game developers)
Clear goals Games should provide the player with clear goals at appropriate times	- overriding goals should be clear and presented early - intermediate goals should be clear and presented at appropriate times

Figure 2.13: GameFlow Criteria for player enjoyment (1/2)

Feedback Players must receive appropriate feedback at appropriate times	- players should receive feedback on progress toward their goals - players should receive immediate feedback on their actions - players should always know their status or score
Immersion Players should experience deep but effortless involvement in the game	- players should become less aware of their surroundings - players should become less self-aware and less worried about everyday life or self - players should experience an altered sense of time - players should feel emotionally involved in the game - players should feel viscerally involved in the game
Social Interaction Games should support and create opportunities for social interaction	- games should support competition and cooperation between players - games should support social interaction between players (chat, etc.) - games should support social communities inside and outside the game

Figure 2.14: GameFlow Criteria for player enjoyment (2/2)

- Immersion - Pervasive games should allow for a shift focus between the virtual and the physical sections of the game, as well as enable a smooth transition between everyday contexts and not demand player actions which violate social norms in everyday contexts;

- Social interaction - Pervasive games should enable meaningful social interactions within the game as well as include game elements that prompt the player to interact socially.

2.4 Related Work

This section presents some projects from which useful information can be gathered from.

2.4.1 Play JBT

Play JBT[40] takes place on the Tropical Botanical Garden of Lisbon, intent on developing an Android mobile app that had a set of tours, tailored for different types of visitors, with different paths and points of interest. The app features a map which displays the user's position as well as the marked points of interest, and serves to guide visitors to these points, as well as display details about them.

Stefan Postolache mentioned some challenges with the development of the project, namely the implementation of the map and the orientation of the visitor. The process of designing the map is summarized on Fig.2.15. In the map, it is possible to see that the maps were merged, the yellow circles marking the distinctions between them, and the bottom left map (which is displayed at the
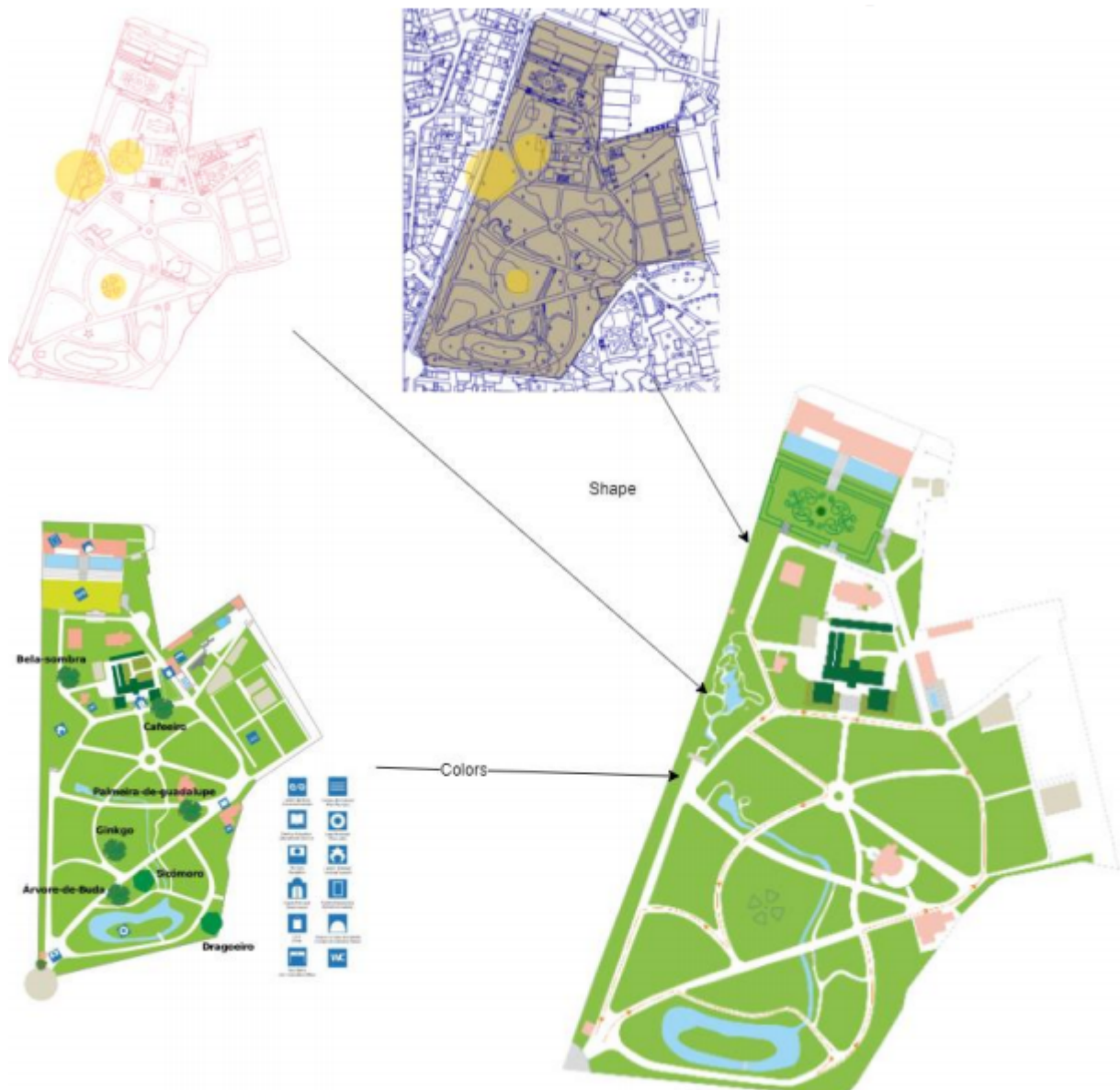
Figure 2.15: Summary of the building of the map for the Play JBT project

entrance of the garden) was used just for coloring, simplifying the map, which ensured the user would not be overwhelmed by unnecessary information.

Map navigation allows the user to locate the points of interest of the tour and themselves on the map, as well as a highlighted path, which is the recommended path to follow, using the GPS function on the user's phone. To help with perceiving what has been seen or not, the user is able to mark a point of interest as seen by tapping it.

As for the orientation of the user, the map is on a fixed orientation, and the icon representing the user rotates to represent their orientation, as shown in Fig.2.16, taking advantage of the gyroscope, accelerometer and magnetic field sensors to calculate the orientation.
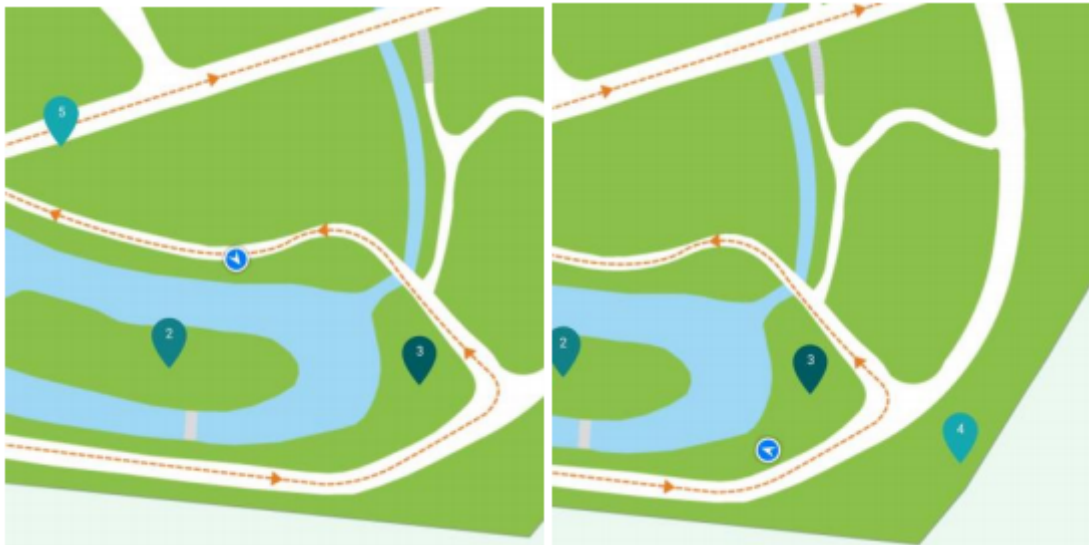
Figure 2.16: Rotation of the user position icon.

2.4.2 An approach to the project H2020 BEACONING

With the use of context-aware pervasive methodologies, the H2020 BEACONING project aimed to develop a different way of learning through gameful experiences. Coelho et al. describe the development of an app[41] that serves as a solution to the BEACONING project, integrating the BEACONING project games and authoring tools.

The app featured a radar-like guiding system, which displayed the locations in relation to the user's position, and included zoom in/out functions and map overlay, and so the user could find the different locations with game challenges. The user was not required to be at the location exactly, but simply near it, a feature implemented with accessibility in mind. The final iteration of the implementation of this app is displayed on Fig.2.17.

Despite all the main requirements being met, one of the obstacles mentioned was the limitations of the technology, as the Android Unity plug-in was not updated, which required work to find which versions could be used.
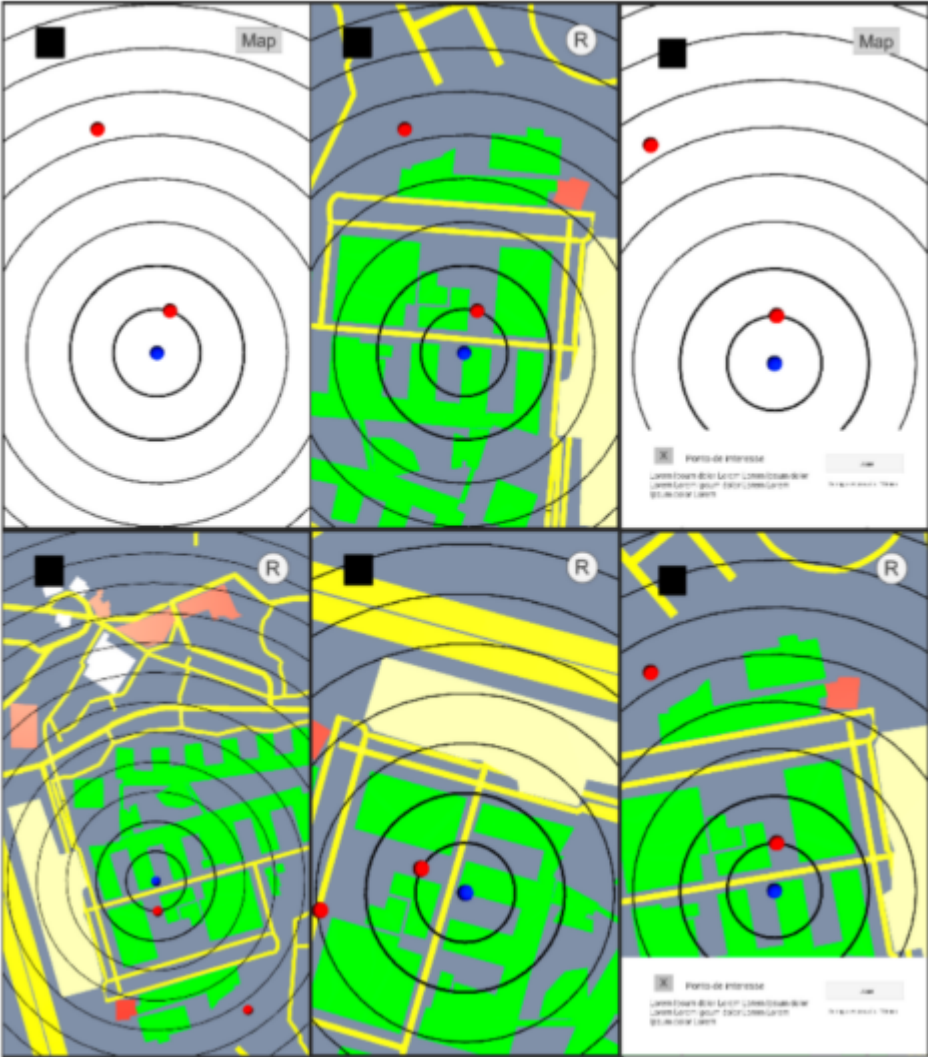
Figure 2.17: Radar system used to guide users.

2.4.3   e-Tracer AR App

The e-Tracer AR application was developed for the Silversmithing museum in the Castle of Ioannina, Greece, to allow museum visitors to explore the exhibitions by themselves, with their own smartphones [42]. The application was developed in Unity and uses ARCore for the AR components. The application, when detecting an object of interest, displays additional information about the artefact or, if available, a 3d projection of them. It takes advantage of the fact that the objects in the museum are mostly static, which allows to manage the expected positions users will be when using the application (see Fig. 2.18).

Figure 2.18: Screenshot of the application and a representation of the physical setting.

A feature that deviates from the normal in this application is the way the quiz was made. After the museum visit, visitors are prompted to answer questions regarding what they learned. However, instead of a choose the right answer, users are asked to go to the 'answer' and scan the artefact with their devices, proposing users to revisit exhibitions (see Fig. 2.19). To take into account the variety of different visitors, the level of difficulty is adjusted to each user based on the profile they create.

For the target markers, it was decided to use printed copies of the artefacts of the museum and not Quick Response Codes (QRC), mainly because the latter was not human readable and
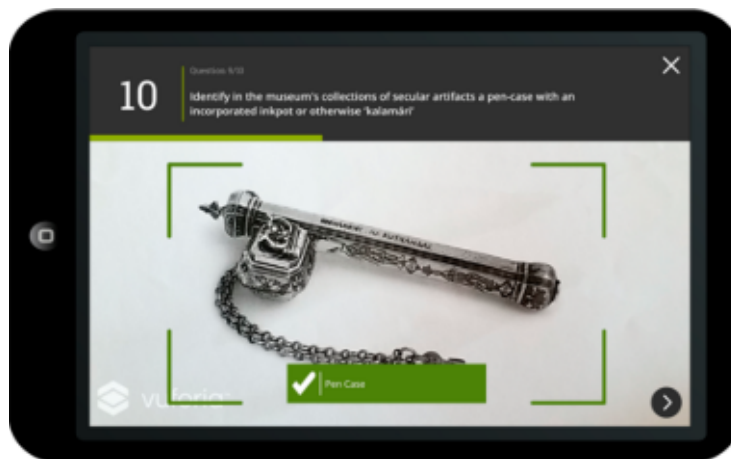
Figure 2.19: Example of answring thrquestion by finding the object.

conflicted with the aesthetic of the museum, unlike the printed copies. To improve the feature detection of the pictures of the artefacts that were going to be used as markers, some values were adjusted: Gamma correction, to make the shadows on the artefacts darker; Local contrast enhancement adjustments, by making the background lighter and the foreground darker and with more contrast; and Laplacian filter, to enhance edges. All these modifications to the images helped improve the number of feature points detected (see Fig. 2.20.
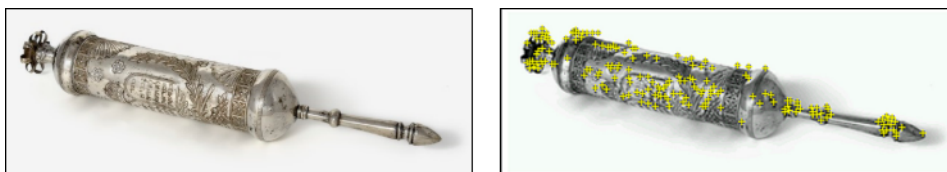
Figure 2.20: The object's image used as a marker on the left and its feature points highlighted on the right.

The results of testing the application allowed them notice some problems, namely the low lighting areas of the museum making it harder for the system to adapt to the position and orientation of the user. A solution to this would be to better illuminate the markers or alternatively, rearrange the exhibitions so that the markers would be in better illuminated areas. Additionally, the app required relatively recent hardware to run, otherwise it would not launch at all. Despite the issues mentioned above, this project showed to be in line with other AR projects, enhancing the experience of visitors in the museum, helping them learn more and become more engaged with the exhibitions.

Chapter 3

Application Overview

The application developed in Unity, named Flo[RA], can be divided in three main sections (and other smaller scenes):

- the Inside Scene, which takes place inside the Gallery, in the room where the leaves are in the showcase window;

- the Outside Scene, which takes place on the Garden surrounding the Gallery;

- the Creative AR Scene, a separate component which can be used anywhere.

The game is designed in a linear route. First, the player must go Inside, where the player finds the leaves and unlocks the ability to find their trees in the Garden. Then Outside, where the player finds the trees and gains the ability to use them in the Creative AR Scene. And lastly, the Creative AR Scene, where the unlocked stickers from finding the trees can be used. Despite its linearity, the game allows for the player to return to a previous stage and unlock what is missing. In the current version there are a total of 6 different species of trees which are used by the application, whose leaves are displayed in the Gallery.

3.1 Inside the Gallery

The leaves' exhibit main problem was the fact that its simplicity and lack of interaction made it hard to get the attention of the visitors and keep it for long. The game designed for this section was made so that these visitors, while they were trying to complete the challenge of finding the specific leaf, would have to pay attention to the various leaves in display and notice how different they were (see Figs. 3.1 and 3.2).

In the Inside scene takes place the leaf detection game, where an image of half a leaf is displayed, and the player must find the respective leaf in the exhibit. The challenge is made in sets of three leaves to be found, requiring the player to repeat the challenge to find all six (see fig.3.3). The player can choose between 3 levels of difficulty, which correspond to how detailed is the half leaf image (see fig.3.4). A timer and a score system runs hidden from the player, which serve as a measure of the player's performance in the challenge.

Figure 3.1: The four leaves displays inside the Gallery.

Figure 3.2: User looking for a leaf as part of the leaf challenge.

Figure 3.3: Introductory panel for the Inside section of the game and the difficulty selection.

Figure 3.4: The three levels of detail according to the difficulty selected.

When searching for the leaves, if the camera points at one of the six leaves, regardless of whether it is the leaf that the player is searching for or not, a question mark shows up, suggesting to the player that a leaf is being detected. When the player presses the camera button, if the correct leaf is not being detected, a red cross is displayed, a penalty is applied to the score and the player has to keep searching for the correct leaf. Finding the correct leaf displays a green tick (see fig.3.5).

Once the correct leaf is detected, the game asks for the scientific name of the leaf species, and presents three possible options. Answering right or wrong affects the score accordingly. Once

Figure 3.5: From left to right: the application detecting one of the leaves; feedback showing it was the wrong leaf; the application detecting another leaf; feedback showing it was the correct leaf.

answered, a panel with various information of the species is shown (see fig.3.6). In case the player has trouble finding the leaf, or is, in some other way, unable to detect the leaf, the player can use the option to skip finding the leaf, which marks it as found and moves on to the question.
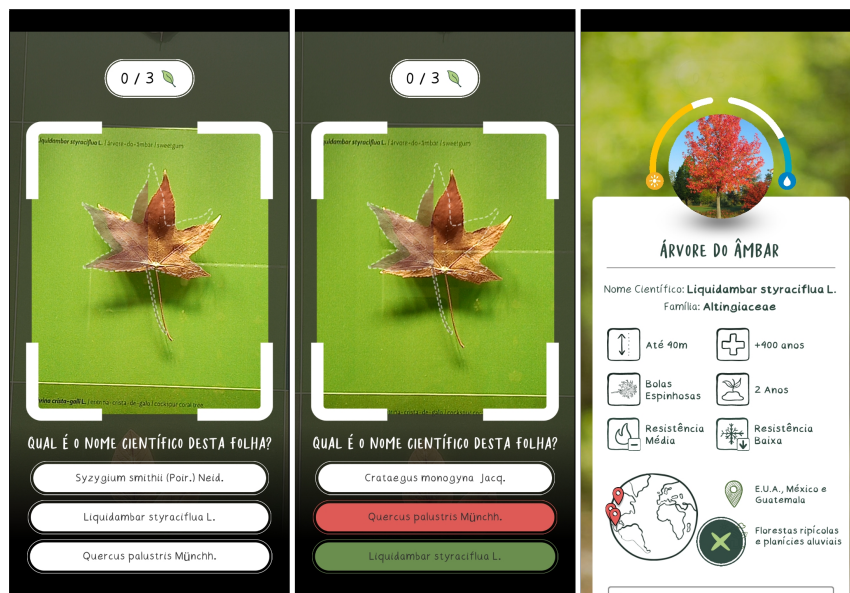
Figure 3.6: The question for the scientific name of the species of the leaf after it was found, and the information modal of the species.

Once three leaves of the challenge are found, the final panel is shown, displaying ratings regarding time and the number of mistakes the player made, as well as the overall score. Below, the player is given option to repeat the challenge if there are still leaves to be found, check the

Collection, or move on to the next section, the Outside scene (see fig.3.7).

Figure 3.7: The final panel of this section, showing the performance in the previous challenge, allowing to easily compare it to others.

3.2 Outside the Gallery

The Outside scene takes place in the Garden, where the player is tasked with finding the trees of the same species of the leaves that were found in the previous scene (see fig.3.8). As such, if the player has only found three leaves, then they will only be able to find their three respective trees.
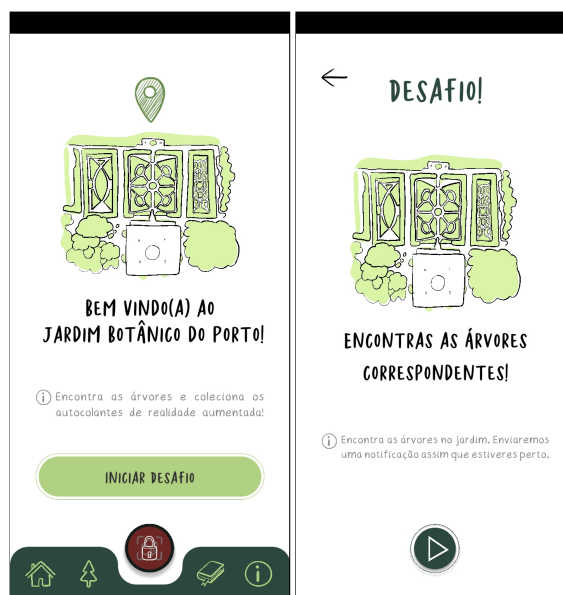
Figure 3.8: Introductory panels of the Outside Section.

A map of the Garden is displayed with pins showing the location of the trees, but does not display the position of the player. However, when the application detects that the player is approaching a relevant tree, the map centres the respective pin to highlight the tree in question, alongside a text hinting to how close the player is to the tree.

The player's distance to each tree is divided in 4 tiers: Closer than 7 meters away, 7 to 15 meters away, 15 to 30 meters away and further than 30 meters away (see fig.3.9). If more than one tree is in range, the closest one will be taken into consideration. Once a tree is found, a small dialogue box shows, allowing the tree to be confirmed found. Once all trees are found, the final panel is displayed, urging the user to experiment with the Creative AR scene (see fig.3.10).
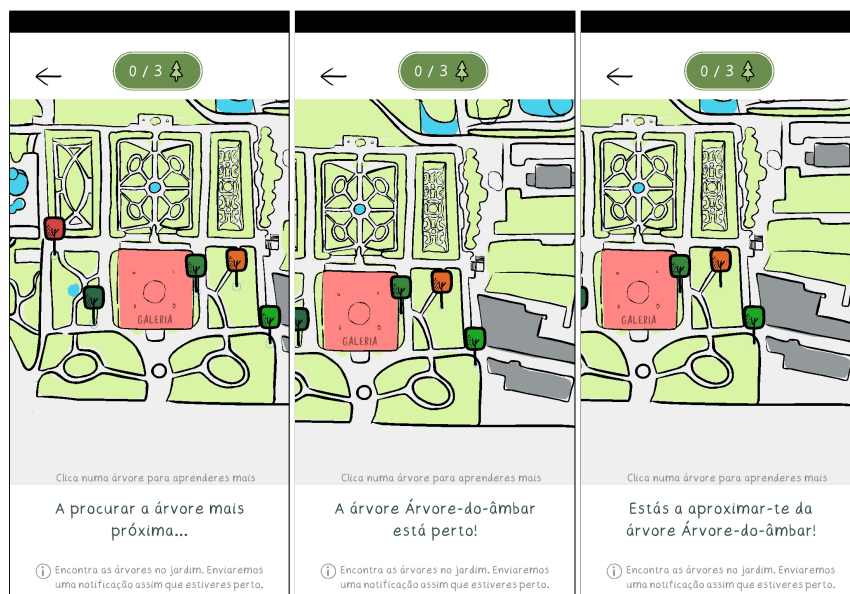
Figure 3.9: The various tiers of proximity before finding the tree.

Initially, this section was only supposed to have a sort of "radar", where it would simply notify the player as they got closer to a relevant tree, with the idea that the players would be walking around and enjoying the garden without paying much attention to finding the trees. However, this was reconsidered when we thought about the more objective driven players, as well as players that wished to play the game to its full extent but couldn't spend an uncertain amount of time looking for the trees blindly. With this in mind, the map was implemented in order to make it easier for players to find the trees.

3.3   Creative AR Scene

This scene is part of the application but is not a part of the scope of this dissertation. In here, the player is given the possibility to use their surroundings as a creative space.

The application allows the player to place digital leaves and to even grow a tree, allowing the player to express themselves creatively and share resulting pictures.

Details about this section can be found on my colleagues dissertation, David Soares.
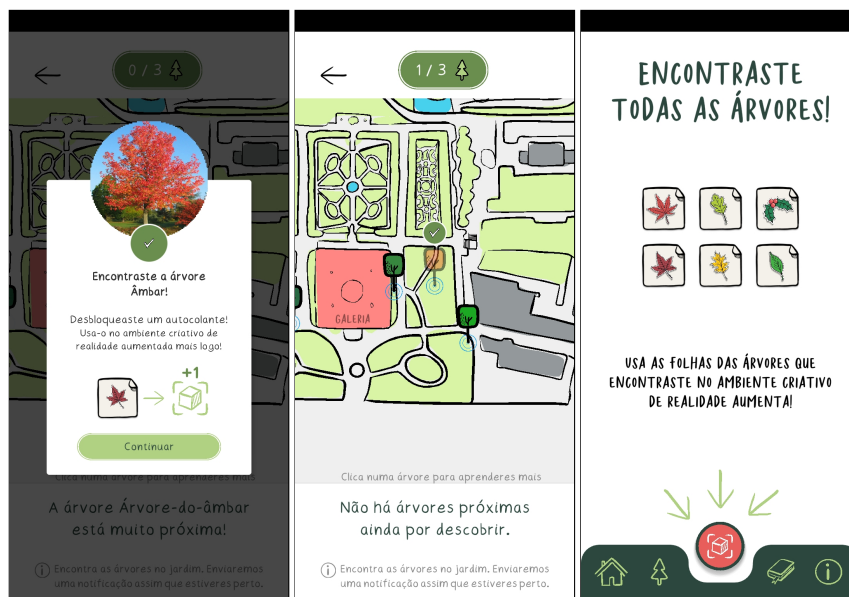
Figure 3.10: From left to right: the window pop-up to confirm finding the tree; the tree marked on the map as found; the final panel once all trees are found.

3.4 Other Scenes

These additional scenes serve as complements to the main section.

3.4.1 Permission Screen

The Permission Screen ensures the player understands that the application requires access to the Camera and Location of the device to work properly (see fig.3.11). It is also in this scene that the game information is loaded for the rest of the game.

3.4.2 Introduction Panels

The Intro Panels give the player a quick presentation of the flow of the game and what can be done in the application (see fig.3.12).

3.4.3 Collection

The Collection scene showcases the leaves that were found, as well as indicating if the respective trees were already found as well. Clicking on a found leaf displays the information previously seen when they were found (see fig.3.13).

3.4.4 About Screen

The About Screen presents the developers and the people who guided them in the development of this project, and also the option to reset the game data should the player wish to reset their experience (see fig.3.14).
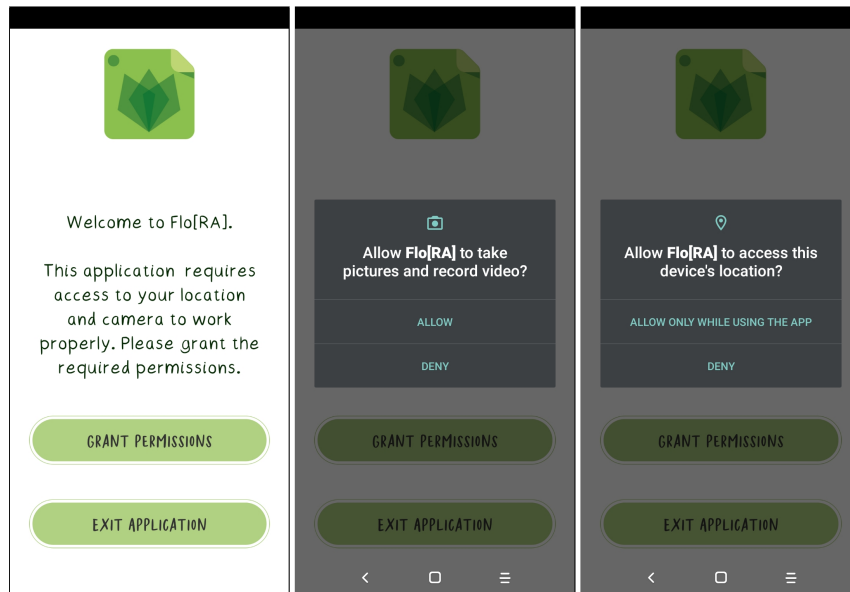
Figure 3.11: The application explaining that it needs to request permissions before the operative system shows the permission dialogue.
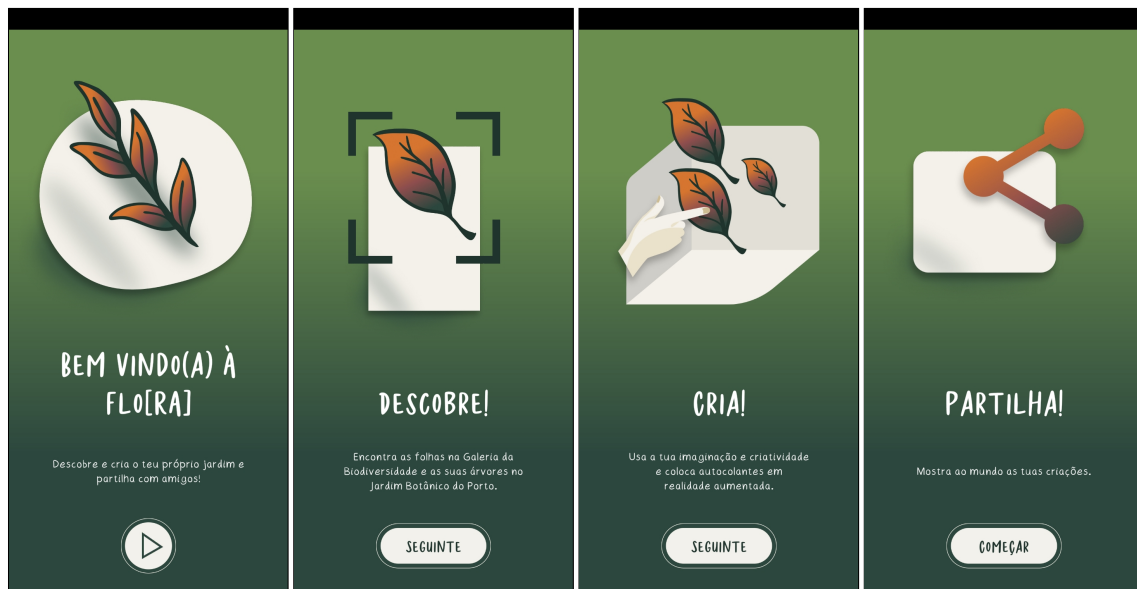
Figure 3.12: The Introduction panels for the application.
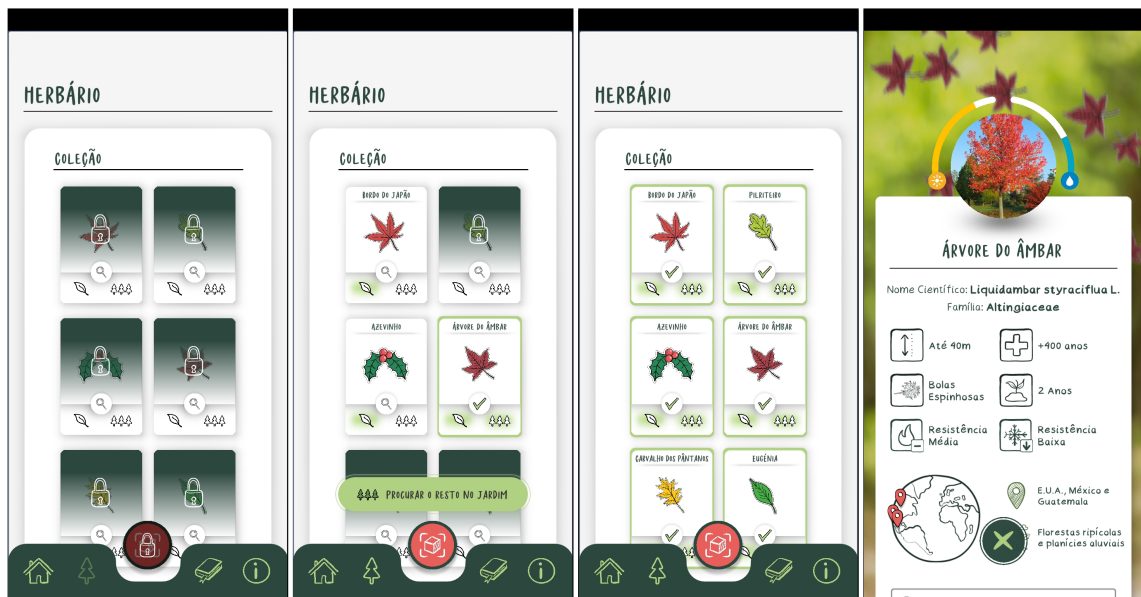
Figure 3.13: The Collection menu, showing the leaves locked, unlocked and those whose tree have also been found.
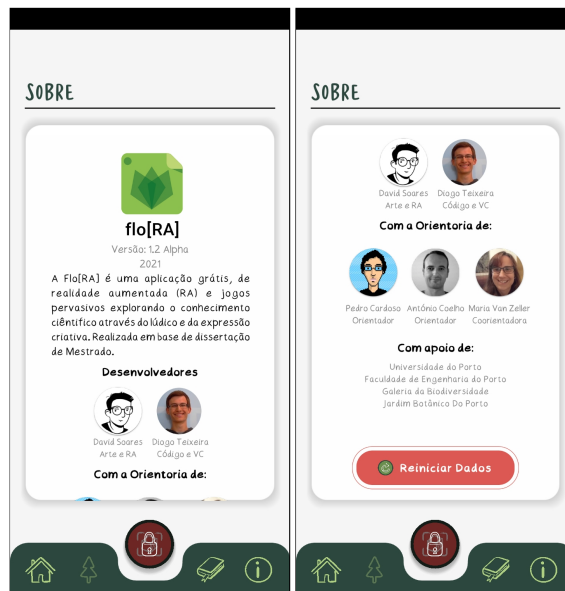
Figure 3.14: The About section with those involved in the development of the application, as well as the reset button.

Chapter 4

Implementation

This chapter describes the most relevant implementation aspects of the developed application.

4.1 Application Architecture

The application can be split into multiple components (see Fig. 4.1): GPS data is retrieved from the system with Location Service by Unity, used for localisation in the Botanical Garden; the Vuforia Engine compares the Camera image to the ones stored in the Target Manager database, used in leaf recognition; the rest of the application is on Unity.
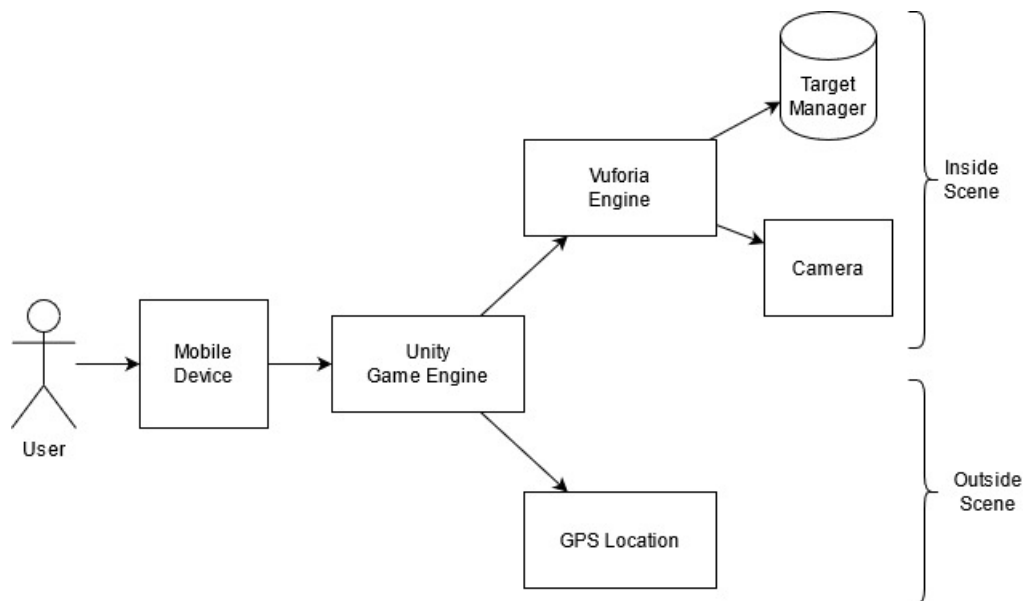
Figure 4.1: Architecture diagram of the application.

4.2   Game Controller

In order to keep data between scenes, a 'Game Control' object was created. This object Implis not destroyed on Scene change and is used to keeps all data relevant to maintain between scenes, such as what leaves were found.

This object is placed in every scene, so that its functionality can be used for quick testing in the Unity Editor without having to start in a different scene. However, since 'Game Control' is an object that is marked to not be destroyed when the scene changes, every time a scene changed, a new instance of 'Game Control' would be created, and all the previous ones would still be active. To avoid that, whenever an instance of 'Game Control' is created, it checks if another instance already exists, in which case it deletes itself immediately.

```
1    private void Awake()
2       {
3           //Let the gameobject persist over the scenes
4           DontDestroyOnLoad(gameObject);
5
6           //Check if the control instance is null
7           if (control == null)
8           {
9               //This instance becomes the single instance available
10              control = this;
11          }
12          else if( control != this)
13          {
14              //In case there is a different instance destroy this one.
15              Destroy(gameObject);
16          }
17       }
```

The 'Leaf' class contains all information regarding a single leaf, such as its common and scientific name, but also the coordinates of the tree in the Garden. To better present this information, and more easily expand the number of leaves, a 'LeafStruct' was implemented. This struct allows for the creation of the instances of the 'Leaf' class with the input in the Unity Editor's Inspector, displaying the information in a more organized way and making it more simple to add more leaves in the future (see fig.4.2).
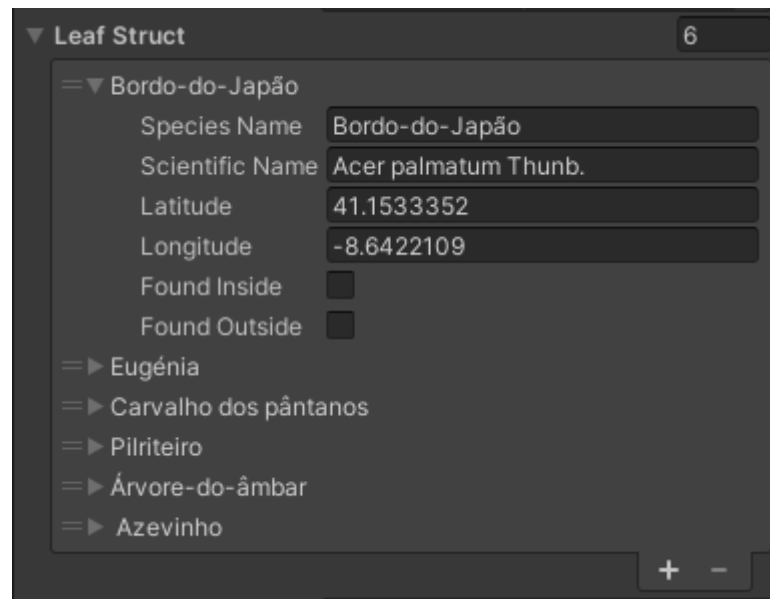
Figure 4.2: Leaf Struct variables shown in Unity Editor's Inspector.

4.3   Leaf Detection Challenge

The first section of the game picks three unfound leaves for the user to find them, by first filtering the leaves to those that are unfound and then randomly choosing three, with the following methods:

```
18        public void CreateLeafChallenge()
19    {
20        leafChallenge.Init();
21        WinStarsScript.ResetValues();
22
23        // Create a list of only unfound Leaves
24        List<short> leavesNotFound = new List<short>(Control.Leaves.Length);
25        for (short i = 0; i < Control.Leaves.Length; i++)
26        {
27            if(!Control.Leaves[i].IsLeafFound())
28            {
29                leavesNotFound.Add(i);
30            }
31        }
32
33        // Fill leafChallenge with random leaves to be found
34        short loop_max =
35                (short)Mathf.Min(leavesNotFound.Count, LEAVES_PER_CHALLENGE);
36        for (short i = 0; i < loop_max; i++)
37        {
38            int rnd = Random.Range(0, leavesNotFound.Count);
39            leafChallenge.leaves[i] = leavesNotFound[rnd];
40            leavesNotFound.RemoveAt(rnd);
```

```
41            }
42        }
43
44      public struct LeafChallenge
45      {
46          public short index;
47          public short[] leaves;
48          public bool isActive;
49
50          public string[] answers;
51          public short rightAnswer;
52
53          public float TimeStart;
54          internal void Init()
55          {
56              index = 0;
57              leaves = new short[LEAVES_PER_CHALLENGE];
58              answers = new string[3];
59              TimeStart = Time.time;
60          }
61
62          internal short GetLeaf()
63          {
64              return leaves[index];
65          }
66
67          internal bool Next()
68          {
69              index++;
70              if (index >= LEAVES_PER_CHALLENGE)
71              {
72                  index = 0;
73                  return false;
74              }
75              else return true;
76          }
77      }
```

The point system is calculated by penalizing wrong answers and time spent looking for the leaf, and rewarding correct answers and finding the correct leaf. The following are the values used to calculate the player's points:

```
78      private static readonly int CORRECT_LEAF_POINTS = 300;
79      private static readonly int WRONG_LEAF_POINTS = 50;
80      private static readonly int CORRECT_ANSWER_POINTS = 150;
81      private static readonly int WRONG_ANSWER_POINTS = 100;
82      private static readonly float POINTS_PER_SECOND = 11.0f;
```

For the recognition of detected leaves, the application made use of the Vuforia AR SDK for Unity. At first, photos of the leaves were taken, from which the feature points were extracted using Vuforia's Target Manager.

However, the different size and shape of the various leaves meant that some were much easier to detect than others. In truth, one particular leaf, whose species is commonly known as Eugenia, was at times impossible to detect. Initially, the photo was edited, increasing contrast with the intent that it would improve detection, and while it did, it didn't improve enough.

For this reason, it was decided to make use of the fact that the leaves greatly differ from one another, are still objects that aren't modified and can only be observed from one angle, and so the photos taken of the leaves were directly used as the target images, created at runtime, for detection, without processing them using the Target Manager.

4.4 GPS location

Unity Engine's built-in 'LocationService' was able to access Android's GPS data, which simplified the process of obtaining the data. First the service must be initialized, like this:

```
83  private IEnumerator StartLocationService()
84      {
85          if (!location.isEnabledByUser)
86          {
87              Debug.LogError("GPS is not enabled");
88              yield break;
89          }
90
91          location.Start(4.0f, 1.0f);
92          int waitCounter = 20;
93
94
95          while (location.status != LocationServiceStatus.Running && waitCounter > 0)
96          {
97              yield return new WaitForSecondsRealtime(1f);
98              waitCounter--;
99          }
100
101         if (waitCounter <= 0)
102         {
103             Debug.LogError("Location Service Initialization Timed Out");
104             yield break;
105         }
106
107         if (location.status == LocationServiceStatus.Failed)
108         {
109             Debug.LogError("Location Service Initialization Failed");
110             yield break;
```

```
111            }
112
113        locationServiceStarted = true;
114
115        StartCoroutine(UpdateCoordinates());
116
117        yield break;
118    }
119
120    private IEnumerator UpdateCoordinates()
121    {
122        while (true)
123        {
124            selfLatitude = location.lastData.latitude;
125            selfLongitude = location.lastData.longitude;
126            selfAccuracy = location.lastData.horizontalAccuracy;
127            location_updated = true;
128            yield return new WaitForSeconds(1.0f);
129        }
130    }
```

The 'LocationService.Start' takes two parameters: the first sets the desired accuracy in meters, and the second sets the update distance in meters, that is, how much distance does the player need to move before the player coordinates are updated. On average, the accuracy values were around 3.5 meters, which were perfectly adequate for the application.

In order to calculate the distance of the player to the trees, the method 'DistanceToTree', in the class 'Leaf', was implemented:

```
131    public double DistanceToTree(double lat, double lon)
132    {
133        double lat1 = lat;
134        double lat2 = treeCoordinates.x;
135        double lon1 = lon;
136        double lon2 = treeCoordinates.y;
137
138        double dLat = (lat2 - lat1) * Mathf.PI / 180;
139        double dLon = (lon2 - lon1) * Mathf.PI / 180;
140
141        lat1 = lat1 * Mathf.PI / 180;
142        lat2 = lat2 * Mathf.PI / 180;
143
144        double a = Mathf.Sin((float)dLat / 2) * Mathf.Sin((float)dLat / 2)
145                + Mathf.Sin((float)dLon / 2) * Mathf.Sin((float)dLon / 2)
146                * Mathf.Cos((float)lat1) * Mathf.Cos((float)lat2);
147        double c = 2 * Mathf.Atan2(Mathf.Sqrt((float)a), Mathf.Sqrt((float)(1 - a)));
148        return EARTH_RADIUS * c * 1000;
149    }
```

The changes caused by moving closer to the relevant trees include the map focusing on the tree, the text changing to reflect in which tier of distance is the tree, and the device vibrating. The vibration uses Unity's built-in 'Handheld' interface.

Finding a tree causes all the above to happen as well as receiving an app notification and a small panel popping up, allowing the player to confirm the discovery. The notification was implemented using the 'Unity Mobile Notification' package, available in the package manager, and used in the 'UpdateGPSText' class:

```
150  public class UpdateGPSText : MonoBehaviour
151  {
152      public Text coordinates;
153      public Text LeafName;
154      private GPSLocation gps;
155
156      void Update()
157      {
158          // The GPSLocation class is active
159          if (gps != null)
160          {
161              //The value is of a tree
162              if(gps.ActiveNotification != -1)
163              {
164                  //It is a new tree for the previous notification
165                  if(gps.ActiveNotification != gps.previousNotification)
166                  {
167                      string species = GameControl.control
168                              .Leaves[gps.ActiveNotification].speciesName;
169                      LeafName.text = "Arvore Perto: " + species;
170                      Handheld.Vibrate();
171                      MobileNotifications.CreateNotification(
172                          "Uma Arvore esta Perto",
173                          "Pareces estar perto da arvore " + species
174                          );
175                  }
176              }
177              else
178              {
179                  LeafName.text = "Nao ha arvores para descobrir por perto.";
180              }
181              gps.previousNotification = gps.ActiveNotification;
182          }
183          else gps = GPSLocation.Instance;
184      }
185  }
```

When the small panel opens, a congratulating sound plays, using an 'Audio Source' object

attached to the modal, which plays when the modal is activated. While this panel is active, the
coordinates are not updated in the application, to prevent the player from finding multiple trees
without acknowledging the first find.

```
187   private void CheckForTrees()
188       {
189           Proximity closest_proximity = Proximity.FARAWAY;
190           short closest_index = -1;
191
192           for (short i = 0; i < leaves.Length; i++)
193           {
194               if (leaves[i].IsLeafFound() && !leaves[i].IsTreeFound())
195               {
196                   Proximity prox = CheckIfInRange(leaves[i]);
197                   if(prox < closest_proximity)
198                   {
199                       closest_proximity = prox;
200                       closest_index = i;
201                   }
202               }
203           }
204
205           if (!isModalOpen)
206           {
207               foundLeafIndex = closest_index;
208               foundProximity = closest_proximity;
209               switch (closest_proximity)
210               {
211                   case Proximity.CLOSE:
212                       {
213                           RadarTreeText.text = "A arvore "
214                                   + leaves[closest_index].speciesName
215                                   + " esta muito proxima!";
216                           OpenFoundModal();
217                           // Notify nearby tree
218                           ActiveNotification = closest_index;
219                           break;
220                       }
221                   case Proximity.MEDIUM:
222                       {
223                           RadarTreeText.text = "Estas a aproximar-te da arvore "
224                                   + leaves[closest_index].speciesName + "!";
225                           break;
226                       }
227                   case Proximity.DISTANT:
228                       {
229                           RadarTreeText.text = "A arvore "
230                                   + leaves[closest_index].speciesName
```

```
231                                   + " esta perto!";
232                         break;
233                     }
234             default: //FARAWAY
235                     {
236                         RadarTreeText.text =
237                                 "Nao ha arvores proximas ainda por descobrir.";
238                         break;
239                     }
240         }
241
242         // Move map view to nearby tree
243         currentViewPortIndex = closest_index;
244         currentViewPortProximity = closest_proximity;
245         if (currentViewPortProximity < previousViewPortProximity
246             || currentViewPortIndex != previousViewPortIndex)
247             {
248                 TreeMap.FocusMapOnTree(currentViewPortIndex);
249                 Handheld.Vibrate();
250             }
251         previousViewPortProximity = currentViewPortProximity;
252         previousViewPortIndex = currentViewPortIndex;
253     }
254     return;
255 }
```

4.5   Requesting Permissions

Requesting Permissions to access certain components of a mobile device is necessary to use them. Since this application uses GPS Location as well as the device's camera, it is required that the user grants those permissions. By default, the Android operative system automatically presents the dialog requesting permissions when the application first tries to access those components. However, it is both more pleasant and recommended by Android to present an explanation about the need to access those components and only then request the permissions to access them. In this application, the request is made at the beginning, like so:

```
256 public void RequestPermission()
257 {
258     if (!Permission.HasUserAuthorizedPermission(Permission.Camera)
259     || !Permission.HasUserAuthorizedPermission(Permission.FineLocation))
260     {
261         string[] permissions = { Permission.Camera, Permission.FineLocation };
262         Permission.RequestUserPermissions(permissions);
263     }
264 }
```

It is important to note that requesting multiple permissions should be done using
'Permission.RequestUserPermissions(string[])' and not multiple calls of
'Permission.RequestUserPermission(string)'. This is due to some versions of Android automatically
deny all permissions that would be requested after the first one is requested. Using the permissions
above as example, if the requests were made as:

```
265  public void RequestPermission()
266      {
267          if (!Permission.HasUserAuthorizedPermission(Permission.Camera)
268          || !Permission.HasUserAuthorizedPermission(Permission.FineLocation))
269          {
270              Permission.RequestUserPermission(Permission.Camera);
271              Permission.RequestUserPermission(Permission.FineLocation);
272          }
273      }
```

In some versions of Android, the dialog window requesting the permission for the Camera would
pop up, but the permission request for FineLocation would be automatically denied.

4.6   Saving and Loading Data

Unity has two ways to save data: using the PlayerPrefs class, which can store data in the form of
strings, floats and integers between game sessions, or by serializing data and writing it to a file.
Since this application requires to store information of arrays, which leaves and trees were found,
serialization was the only option available.

First, a class 'SaveData' was created, which contained all the variables that needed to be saved.

```
274  [System.Serializable]
275  class SaveData
276  {
277      public static readonly string FILE_PATH = Application.persistentDataPath
278                                          + "/FloRA.dat";
279      public bool[] savedFoundInside;
280      public bool[] savedFoundOutside;
281      public bool SeenIntro;
282      public bool SeenTutorial;
283
284      public SaveData(int length)
285      {
286          savedFoundInside = new bool[length];
287          savedFoundOutside = new bool[length];
288      }
289  }
```

Saving data requires all the data to save to be copied to their equivalent variables in the 'SaveData' class and then serialize it and write it to the save file. Loading the save data is reversing this process, first deserializing from the save file and then copy the variables from the 'SaveData' class to, in this case, the variables in the 'GameControl' class. Resetting the data is simply to delete the save file and resetting the relevant variables to the default value.

```
290  public void SaveGame()
291      {
292          BinaryFormatter bf = new BinaryFormatter();
293          FileStream file = File.Create(SaveData.FILE_PATH);
294          Leaf[] leaves = control.Leaves;
295          SaveData data = new SaveData(leaves.Length);
296          data.SeenIntro = seen_intro;
297          data.SeenTutorial = seen_tutorial;
298          for (short i = 0; i < leaves.Length; i++)
299          {
300              data.savedFoundInside[i] = leaves[i].IsLeafFound();
301              data.savedFoundOutside[i] = leaves[i].IsTreeFound();
302          }
303          bf.Serialize(file, data);
304          file.Close();
305          Debug.Log("Saved Game Data: " + SaveData.FILE_PATH);
306      }
307
308      public void LoadGame()
309      {
310          if (File.Exists(SaveData.FILE_PATH))
311          {
312              BinaryFormatter bf = new BinaryFormatter();
313              FileStream file = File.Open(SaveData.FILE_PATH, FileMode.Open);
314
315              SaveData data = new SaveData(control.Leaves.Length);
316              data = (SaveData)bf.Deserialize(file);
317              file.Close();
318              for (short i = 0; i < control.Leaves.Length; i++)
319              {
320                  if (data.savedFoundInside[i])
321                      control.Leaves[i].FoundLeaf();
322                  if (data.savedFoundOutside[i])
323                      control.Leaves[i].FoundTree();
324              }
325
326              if (data.SeenIntro) seen_intro = true;
327              if (data.SeenTutorial) seen_tutorial = true;
328
329              Debug.Log("Loaded Data from: " + SaveData.FILE_PATH);
330          }
331          else
```

```
332          {
333              Debug.LogWarning("No save data on: " + SaveData.FILE_PATH);
334          }
335          load_finished = true;
336      }
337
338      public void ResetGame()
339      {
340              if (File.Exists(SaveData.FILE_PATH))
341              {
342                  File.Delete(SaveData.FILE_PATH);
343                  for (short i = 0; i < control.Leaves.Length; i++)
344                  {
345                          control.Leaves[i].ResetLeaf();
346
347                  }
348                  Debug.Log("Successfully Reset Data.");
349              }
350              else
351                  Debug.LogWarning("No save data found to delete.");
352      }
```

In this application, the save data is loaded when the app is started and is saved after completing a leaf challenge or finding a tree.

Chapter 5

User Testing

This chapter covers the tests that were performed on the application, in order to assess its usability and detect possible bugs in the system, as well as suggesting possible solutions.

5.1 Methods

To conduct these tests, we used the strategy Thinking Aloud, by asking the users to vocalise their thoughts while using the application. This way, it is possible to follow the users mental path of how should the application work and discover usability issues. Additionally, these test also allowed to observe if the application could work when handle in a different way by different users, that is, since different people try different things in different orders, these tests allowed to see if the application had bugs that had remained undetected when tested by those involved in the project. My colleague, David Soares, was the interviewer and focused on the usability issues while I, Diogo Teixeira, was the observer and focused on the technical problems and their causes.

Due to the nature of the application, all tests were conducted in *Galeria da Biodiversidade* and on *Jardim Botânico do Porto*, between 4th of June and 7th of June, during the working hours of the Gallery, between 10h00 and 18h00. In total, sixteen people volunteered to test the application, of which four were employees of the gallery and the garden.

5.2 Analysis

First, to better analyse data we would gather, we asked and noted some characteristics of our volunteers:

- 11 volunteers had experience with AR applications;

- 8 volunteers had some knowledge of the trees in the garden;

- 11 volunteers people had previously visited the area;

- Only one volunteer was not accompanied;

Then, they were to use the application, which was divided into the following tasks:

- Task 1 - The user was able to read and to accept or refuse the permission request.

- Task 2 - The user was able to see the Introduction panels.

- Task 3 - The user could access the Inside scene, using the respective button.

- Task 4 - The user could choose a difficulty for the leaf challenge in the Inside scene.

- Task 5 - The user could access the camera panel and the device's camera was being used.

- Task 6 - The leaves were being detected by the app as possible finds.

- Task 7 - When the player clicked the camera button with the right leaf being detected, the game provided feedback that it was correct.

- Task 8 - The leaf challenge question showed up after the correct leaf was chosen.

- Task 9 - The user could choose the correct answer or the wrong answer and the game provided the appropriate feedback (highlight green the correct answer, highlight red the chosen answer if it was wrong).

- Task 10 - The modal panel of the found leaf is displayed after answering the question.

- Task 11 - When the all three leaves of the current challenge were found, the game correctly showed the final panel (points and ratings displayed correctly, buttons working, option to repeat challenge available if all leaves had not been found).

- Task 12 - The user could repeat the leaf challenge if not all leaves had been found.

- Task 13 - The user could access the Collection scene to review found leaves.

- Task 14 - The user could tap a found leaf to learn more.

- Task 15 - The user could access the Outside scene once at least one leaf was found, using the respective buttons.

- Task 16 - The device correctly notified the user of a nearby tree and its proximity tier.

- Task 17 - The user could only find trees whose leaf had been found in the Inside scene.

- Task 18 - When close to a tree, the application showed a small dialog box so that the user could confirm the find.

- Task 19 - Finding all the trees, the device presented the final Outside panel.

- Task 20 - The AR Creative space scene became available after at least one tree had been found and the user could access it.

The success was noted for each task as:

- S - Success -> The task was completed successfully;

- P - Partial Success -> The task had some bug or did not act as expected but was possible to keep using the app.

- F - Failure -> The task had a bug or did not act as expected and made it impossible to continue without closing the application.

- NP - Not Performed -> The task was not performed.

The gathered data from these tasks is presented in the tables 5.1 and 5.2.

Tasks Users	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
U1	S	S	S	S	S	P	S	S	S	S
U2	S	S	S	S	S	P	S	S	S	S
U3	S	S	S	S	S	S	S	S	S	S
U4	S	S	S	S	S	S	S	S	S	S
U5	S	S	S	S	S	S	S	S	S	S
U6	S	S	S	S	S	S	S	S	S	S
U7	S	S	S	S	S	S	S	S	S	S
U8	S	S	S	S	S	S	S	S	S	S
U9	S	S	S	S	S	S	S	S	S	F
U10	S	S	S	S	S	S	S	S	S	S
U11	S	S	S	S	S	S	S	S	S	S
U12	S	S	S	S	S	S	S	S	S	S
U13	S	S	S	S	S	S	S	S	S	S
U14	S	S	S	S	S	S	S	S	S	S
U15	S	S	S	S	S	S	S	S	S	S
U16	S	S	S	S	S	S	S	S	S	S

Table 5.1: Test Sheet of Tasks 1 to 10.

Regarding task 6, users 1 and 2 were unable to detect the leaves in one of the displays, due to the lights in that display not working, which prevented the application from detecting the leaves in the display. Regarding task 10, user 9 was unable to proceed after answering one of the leaf challenge questions. Regarding task 19, users 8, 11, 13 and 14 could not complete the test due to personal constraints.

5.3 Results

Considering the gathered data, the application is robust and functional. The partial failures in task 6 were due to the lights in one of the leaves displays not working, which prevented the application from detecting the leaves in that display. New target images were added to help with the detection

Tasks Users	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20
U1	S	S	S	S	S	S	S	S	S	S
U2	S	S	S	S	S	S	S	S	S	S
U3	S	S	S	S	S	S	S	S	S	S
U4	S	S	S	S	S	S	S	S	S	S
U5	S	S	S	S	S	S	S	S	S	S
U6	S	S	S	S	S	S	S	S	S	S
U7	S	S	S	S	S	S	S	S	S	S
U8	S	S	S	S	S	S	S	S	NP	S
U9	S	S	S	S	S	S	S	S	S	S
U10	S	S	S	S	S	S	S	S	S	S
U11	S	S	S	S	S	S	S	S	NP	S
U12	S	S	S	S	S	S	S	S	S	S
U13	S	S	S	S	S	S	S	S	NP	S
U14	S	S	S	S	S	S	S	S	NP	S
U15	S	S	S	S	S	S	S	S	S	S
U16	S	S	S	S	S	S	S	S	S	S

Table 5.2: Test Sheet of Tasks 11 to 20.

in the different light. The error in task 10 was due to the modal panel that opens after answering the leaf challenge question not closing properly, which was later fixed.

The volunteers also provided some feedback: some mentioned that the half leaf displayed on the device made it unclear regarding the size of the actual leaf, that is, they had trouble figuring out if they were looking for a large or small leaf. Other volunteers were limited by their physique, our youngest volunteer, aged 10, could not scan one of the leaves by himself, and one couple could not complete the garden challenge due to the physical strain.

Furthermore, some technical problems also arised during these tests: on one instance, the lights in one of the displays had failed, which stopped the application from being able to detect the leaves in that display. On another instance, the GPS location seemed to be acting erratically, which seemed to be related to the device also being connected to the gallery's WiFi network, as it returned to normal once the WiFi was turned off, but further investigation is required.

Nevertheless, most of the volunteers showed interest in learning and seemed to be satisfied with the app's potential, namely the AR component.

Chapter 6

Conclusion and Future Work

Traditional methods of object displaying in museums have little to no interactivity, which make them less engaging to visitors, which as a result, are less compelled to learn more about the subject. In the particular case of *Galeria da Biodiversidade do Porto*, the leaves exhibit has multiple leaves in display, with no way to interact with them.

The aim of this Dissertation was the development of an application to counter this problem, that is, not only to increase the level of interaction with the exhibit, but also to improve and expand the learning experience by creating a connection between the exhibit inside, where the leaves are displayed, with the garden outside, where the full trees can be observed. As such, the exhibit becomes less of a viewing experience, instead becoming a game, engaging visitors to participate and learn more about it.

The game showed to be able to increase the interest of the visitors in learning more about the exhibition, and were able to associate the leaves inside with the trees in the garden, all the while having fun playing the game.

There are some limitations to the application. The display cases of the leaves in the gallery require to be properly lit up, as lighting can greatly affect image detection. If a row of lights goes out in one of the displays, the leaves in that display may not be detected. Additionally, the GPS location seems to act unpredictably when the device is also connected to the gallery's WiFi, possible due to a poor connection to it, but further investigation on this matter is required. Despite these issues, the application remains an effective way to have an engaging experience in the gallery.

The use of Bluetooth Low Energy beacons was planned but not implemented, as it required a native Android Library to be made to be able to access Bluetooth features of a device and developing such a library required both time and computing power.

In any case, further improvements to the application can be considered. Visitors typically are accompanied and were observed to interact with each other while using the app, whether competitively or cooperatively, despite it being a single-player game. As such, multiplayer mechanics could improve the user experience even further and increase interest. Another improvement could be the addition of other leaves beyond the original six, in order to extend the learning experience

to more species in display. One other change that could be made is making the application into a web-based system, allowing visitors to enjoy the game without the need to download the application. Finally, improving the leaf detection method to function with changes of lighting would safeguard against future display malfunctions, which would have kept players from fully experiencing the application.

Similar concepts of gamifying the museum experience can be applied to other museums, adapting the game to the content and theme of each museum. For example, city museums have stamps depicting historical buildings and important locations. A game could be made for players to visit the museum and later visit the locations they saw depicted in the stamps, bringing the museum experience to the outside and making it more memorable.

References

[1] R. Hunicke, M. LeBlanc, and R. Zubek, "Mda: A formal approach to game design and game research," in *Proceedings of the AAAI Workshop on Challenges in Game AI*, vol. 4, no. 1. San Jose, CA, 2004, p. 1722.

[2] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From game design elements to gamefulness: Defining "gamification"," in *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2011*, 2011, pp. 9–15.

[3] "Museum Definition - ICOM - ICOM." [Online]. Available: https://icom.museum/en/resources/standards-guidelines/museum-definition/

[4] M. Van Zeller, L. Santos, A. Coelho, and P. Cardoso, "Play the museum: um projeto sobre jogos em ambientes de aprendizagem informal na galeria da biodiversidade," in *5º Encontro sobre Jogos e Mobile Learning*, 2020, pp. 330–338. [Online]. Available: https://estudogeral.sib.uc.pt/jspui/handle/10316/25740

[5] The state of online gaming - 2019. [Online]. Available: https://www.limelight.com/resources/white-paper/state-of-online-gaming-2019/#mobile

[6] "The New Reality for Mobile Gaming: The VR/AR Opportunity | Newzoo." [Online]. Available: https://newzoo.com/insights/articles/the-new-reality-for-mobile-gaming-the-vr-ar-opportunity

[7] "What is a GPS? How does it work?" *Library of Congress, Washington, D.C. 20540 USA*. [Online]. Available: https://www.loc.gov/item/what-is-gps-how-does-it-work/

[8] A. S. Bastos and H. Hasegawa, "Behavior of gps signal interruption probability under tree canopies in different forest conditions," *European Journal of Remote Sensing*, vol. 46, no. 1, pp. 613–622, 2013.

[9] P. Sigrist, P. Coppin, and M. Hermy, "Impact of forest canopy on quality and accuracy of gps measurements," *International journal of remote sensing*, vol. 20, no. 18, pp. 3595–3610, 1999.

[10] K. Książek and K. Grochla, "Aggregation of gps, wlan, and ble localization measurements for mobile devices in simulated environments," *Sensors*, vol. 19, no. 7, p. 1694, 2019.

[11] T. Lindner, L. Fritsch, K. Plank, and K. Rannenberg, "Exploitation of public and private wifi coverage for new business models," in *Building the E-Service Society*. Springer, 2004, pp. 131–148.

[12] A. Ibrahim and D. Ibrahim, "Real-time gps based outdoor wifi localization system with map display," *Advances in Engineering Software*, vol. 41, no. 9, pp. 1080–1086, 2010.

[13] R.-H. Jan and Y. R. Lee, "An indoor geolocation system for wireless lans," in *2003 International Conference on Parallel Processing Workshops, 2003. Proceedings.* IEEE, 2003, pp. 29–34.

[14] A. Blackstone., "Understanding the different types of BLE Beacons," 2015. [Online]. Available: https://os.mbed.com/blog/entry/BLE-Beacons-URIBeacon-AltBeacons-iBeacon/

[15] Infsoft, "Indoor Tracking and Indoor Navigation Using Beacons." [Online]. Available: https://www.infsoft.com/technology/positioning-technologies/bluetooth-low-energy-beacons

[16] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on computer communications. Nineteenth annual joint conference of the IEEE computer and communications societies (Cat. No. 00CH37064)*, vol. 2. Ieee, 2000, pp. 775–784.

[17] S.-C. Yeh, W.-H. Hsu, M.-Y. Su, C.-H. Chen, and K.-H. Liu, "A study on outdoor positioning technology using gps and wifi networks," in *2009 International Conference on Networking, Sensing and Control.* IEEE, 2009, pp. 597–601.

[18] K. J. Gaston and M. A. O'Neill, "Automated species identification: why not?" *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, vol. 359, no. 1444, pp. 655–667, apr 2004. [Online]. Available: https://royalsocietypublishing.org/doi/10.1098/rstb.2003.1442

[19] J. Wäldchen and P. Mäder, "Machine learning for image based species identification," pp. 2216–2225, nov 2018. [Online]. Available: https://besjournals.onlinelibrary.wiley.com/doi/full/10.1111/2041-210X.13075https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13075https://besjournals.onlinelibrary.wiley.com/doi/10.1111/2041-210X.13075

[20] J. Wäldchen and P. Mäder, "Plant species identification using computer vision techniques: A systematic literature review," *Archives of Computational Methods in Engineering*, vol. 25, no. 2, pp. 507–543, 2018.

[21] B. A. Yanikoglu, E. Aptoula, and C. Tirkaz, "Sabanci-okan system at imageclef 2012: Combining features and classifiers for plant identification." in *CLEF (Online Working Notes/Labs/Workshop)*, vol. 23. Citeseer, 2012.

[22] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, jan 2004.

[23] S. G. Wu, F. S. Bao, E. Y. Xu, Y. X. Wang, Y. F. Chang, and Q. L. Xiang, "A leaf recognition algorithm for plant classification using probabilistic neural network," in *ISSPIT 2007 - 2007 IEEE International Symposium on Signal Processing and Information Technology*, 2007, pp. 11–16.

[24] J. Hossain and M. A. Amin, "Leaf shape identification based plant biometrics," in *Proceedings of 2010 13th International Conference on Computer and Information Technology, ICCIT 2010*, 2010, pp. 458–463.

[25] B. Yanikoglu, E. Aptoula, and C. Tirkaz, "Automatic plant identification from photographs," *Machine Vision and Applications*, vol. 25, no. 6, pp. 1369–1383, apr 2014. [Online]. Available: https://link.springer.com/article/10.1007/s00138-014-0612-7

[26] S. Prasad, S. K. Peddoju, and D. Ghosh, "Mobile plant species classification: A low computational aproach," in *2013 IEEE 2nd International Conference on Image Information Processing, IEEE ICIIP 2013*, 2013, pp. 405–409.

[27] W. J. Kress, C. Garcia-Robledo, J. V. B. Soares, D. Jacobs, K. Wilson, I. C. Lopez, and P. N. Belhumeur, "Citizen Science and Climate Change: Mapping the Range Expansions of Native and Exotic Plants with the Mobile App Leafsnap," *BioScience*, vol. 68, no. 5, pp. 348–358, 03 2018. [Online]. Available: https://doi.org/10.1093/biosci/biy019

[28] J. Hamari, J. Koivisto, and H. Sarsa, "Does gamification work? - A literature review of empirical studies on gamification," in *Proceedings of the Annual Hawaii International Conference on System Sciences*. IEEE Computer Society, 2014, pp. 3025–3034.

[29] R. Caillois, *Man, play, and games*. University of Illinois Press, 2001.

[30] J. Huizinga, *Homo Ludens: a study of the play-element in culture*, 30th ed. Boston: The Beacon Press, 2009, oCLC: 837203829.

[31] K. Robson, K. Plangger, J. H. Kietzmann, I. McCarthy, and L. Pitt, "Is it all a game? Understanding the principles of gamification," *Business Horizons*, vol. 58, no. 4, pp. 411–420, 2015.

[32] K. Werbach and D. Hunter, *For the win: How game thinking can revolutionize your business*. Wharton digital press, 2012.

[33] D. Djaouti, J. Alvarez, and J.-P. Jessel, "Classifying serious games: the g/p/s model," in *Handbook of research on improving learning and motivation through educational games: Multidisciplinary approaches*. IGI Global, 2011, pp. 118–136.

[34] A. De Gloria, F. Bellotti, and R. Berta, "Serious Games for education and training," *International Journal of Serious Games*, vol. 1, no. 1, feb 2014.

[35] E. Klopfer, S. Osterweil, K. Salen *et al.*, "Moving learning games forward," *Cambridge, MA: The Education Arcade*, 2009.

[36] M. H. Immordino-Yang and A. Damasio, "We feel, therefore we learn: The relevance of affective and social neuroscience to education," *Mind, brain, and education*, vol. 1, no. 1, pp. 3–10, 2007.

[37] S. Benford, C. Magerkurth, and P. Ljungstrand, "Bridging the physical and digital in pervasive gaming," *Communications of the ACM*, vol. 48, no. 3, pp. 54–57, Mar. 2005. [Online]. Available: https://dl.acm.org/doi/10.1145/1047671.1047704

[38] P. Sweetser and P. Wyeth, "Gameflow: A model for evaluating player enjoyment in games," *Comput. Entertain.*, vol. 3, no. 3, p. 3, Jul. 2005. [Online]. Available: https://doi.org/10.1145/1077246.1077253

[39] K. Jegers, "Pervasive game flow: understanding player enjoyment in pervasive gaming," *Computers in Entertainment (CIE)*, vol. 5, no. 1, pp. 9–es, 2007.

[40] S. Postolache, "Play jbt–mobile application for the tropical botanical garden of lisbon," Ph.D. dissertation, Faculdade de Engenharia da Universidade do Porto, 2020.

[41] A. Coelho, P. Cardoso, M. Camilo, and A. Sousa, "Designing of a mobile app for the development of pervasive games," in *2019 International Conference on Graphics and Interaction (ICGI)*.   IEEE, 2019, pp. 153–159.

[42] I. Paliokas, A. T. Patenidis, E. E. Mitsopoulou, C. Tsita, G. Pehlivanides, E. Karyati, S. Tsafaras, E. A. Stathopoulos, A. Kokkalas, S. Diplaris *et al.*, "A gamified augmented reality application for digital heritage and tourism," *Applied Sciences*, vol. 10, no. 21, p. 7868, 2020.