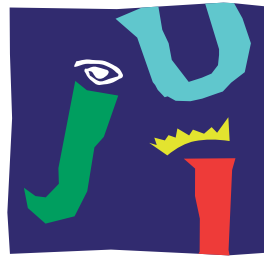# Project Incandescence
Design and development of a geolocated experience

Guzmán Bermejo Rodríguez

Advisor: Dr. Raúl Montoliu Colás

Department of Engineering
Universitat Jaume I de Castellón

This dissertation is submitted for the bachelor's degree of
Video Game Design and Development

Castellón de la Plana   September 2016

Abstract

As years go by, more and more everyday objects are getting interconnected and aware of their environment. This technological fabric being woven across our lives opens up opportunities never seen before. The communication and personal nature of smartphones makes them the ideal flagship in charge of clearing the path and heading towards the Internet of Things era. Their embedded sensors together with their broad adoption makes them the perfect devices for experimenting on novel ways of creating fresh interactive experiences.

In this paper, Project Incandescence is proposed; an application which aims to experiment with storytelling by the exploitation of up-to-date technologies present in smartphones. The intended target audience are students of the last years of high school or college students. In other words, people from 16 to 24 years old which are willing to try new things. Project Incandescence is a global video game project but efforts will be mainly focused on programming an app which integrates and makes use of the information provided by recent technologies. Specifically, geolocation, augmented reality and haptic feedback will capture our attention. Another key part of the project is game design. However, it will be tackled with lesser dedication. The artistic aspect of the experience will not be neglected either, but it will be the least important aspect of the three main ones.

The development will target the Android platform which is the most extended one among smartphones. This software stack has also postulated as key one in the future since it does not target a specific device, and currently supports from watches to cars.

Keywords: *Video Game Design and Development, Storytelling, Geolocation, Augmented Reality, Haptic Feedback, Android.*

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

$2D$ Two Dimensions

$3D$ Three Dimensions

$AIEP$ Academia de Idiomas y Estudios Profesionales

API Application Programming Interface

APK Android Application Package

AR Augmented Reality

bpm Beats Per Minute

CPU Central Processing Unit

dp Density-independent Pixel

ES Embedded Systems

FPS Frames Per Second

GB Gigabyte

GDD Game Design Document

GHz Gigahertz

$GLSL$ OpenGL Shading Language

GPS Global Positioning System

GPU Graphics Processing Unit

HOG Histogram of Oriented Gradients

IDE Integrated Development Environment

Inc. Incorporation

JNI Java Native Interface

km Kilometer

km^2 Square kilometer

LBP Local Binary Patterns

m Meter

MHz Megahertz

MIT Massachusetts Institute of Technology

MP Megapixel

ms Millisecond

MVP Model View Projection

NDK Native Development Kit

NPC Non-Playable Character

OpenGL Open Graphics Library

OS Operating System

QR Quick Response

RT Real Time

SURF Speeded Up Robust Features

UI User Interface

U.S. United States

UX User Experience

VSYNC Vertical Synchronization

Chapter 1

Technical Proposal

In this chapter, an overview of the project will be presented. This general analysis will include the motivation backing Project Incandescence, a list of the principal objectives to be fulfilled during the elaboration of the project together with the risks and planning associated with it.

1.1 Motivation

Since the fifties, we have seen a video games evolution in complexity and features. However, the traditional schema where a player sits in front of a screen has rarely been altered. This behavior has probably been inherited from the way we used to consume other media like television, movies, plays or books. Far for being that model the only one, we can observe other types of consumption as for example when we go for a walk with someone and they told us a story, or when we plug in our headphones and start swimming. In the interactive world scenario, we could consider another type where the screen is used as our controller/guide while the real world is our playground.

Following traditional schemas might also incur in the omission of new kind of experiences which can only be brought by modern technologies. One device which allows us to take advantage of these opportunities are smartphones. They are portable devices holding a permanent connection to the Internet which also contain a collection of sensors that makes them aware of the environment in several ways. Smartphones started as a premium device but they have suffered a democratization due to the fact that each year they become cheaper and more powerful. The availability of them keeps increasing and last year we reached 3,400 million smartphone subscriptions worldwide [11]. This means that there is about one smartphone per two inhabitants in the world [47].

Project Incandescence will try to exploit some of the technologies made available by smartphones in order to tell a story in a different way, taking advantage of the joy we feel when discovering new places. The principal objective of this project is to unify different kinds of inputs and outputs to generate a unique experience. In other words, bend popular types of games such as exploration, adventure, geocaching, treasure hunt and scavenger hunt with new technologies such as geolocation, augmented reality and haptic feedback. By doing so, a story bonded to a real location, in our case the Riu Sec campus of the Jaume I University, will be created. This kind of experience aims to revitalize an area by spotting and reusing architectural elements regularly ignored.

Regarding the visual aesthetics of the application, it will be kept simple and clean. This is going to be achieved by the use of material design principles for interfaces and a low poly approach for 3D models, unless better options are discovered during the development. Nevertheless, the part regarding the art of the video game is considered as low priority when compared with the game design one and even lesser compared with the programming one. A smaller importance does not mean none since game design and art will provide a sense of identity to the global project. Taking into account time constraints, the expected output of the corresponding Bachelor's Degree Final Project is a minimum viable product rather than a production-ready one.

Android Studio will be used as the main developing tool since its the official IDE for developing apps in the Android platform. The following section will clarify the preference of the Android platform while Section 1.6 describes more precisely the tools to be employed throughout the different phases of the game.

1.2   State of the Art

A theoretical introduction to the concepts dealing with the system should be presented before continuing with this technical proposal. In addition, different lines of existing work in the targeted area should also be discussed.

1.2.1   Adventure video games

According to the classification made by Scott Rogers an adventure game is the one which *"focuses on puzzle solving, and item collection and inventory management"* [77]. In [78] the author goes further and introduces exploration into the definition. Considering or not exploration as a key factor in adventure games, a subgenre of adventure games which focuses on this aspect should be considered as the most related to Project Incandescence.

Exploration games

Exploration games refers to the adventure video game's subgenre where players discover a story through exploration and discovery of a virtual world. The narrative is usually embedded in elements which needs to be discovered instead of being placed in cutscenes or interactions with NPCs. Some examples of this subgenre are: Gone Home, Thirty Flights of Loving and Dear Esther.

1.2.2 Outdoor locating games

This category comprises all the games that take place outdoors and involve some kind of navigation through the environment. If the game includes virtual aspects it is normally called a transreality game since it combines real and virtual worlds. For Project Incandescence three subcategories need to be described.

Treasure Hunt

A treasure hunt is a type of game where a group of people compete again each other in order to be the first ones to find a real treasure hidden somewhere in the physical world. Normally, players begin from a particular starting point where a clue or direction of an intermediate location is given. When they reach this intermediate location, they gather another clue or location that leads them to another intermediate point. Sometimes in order to gather this clue or location, the player must overcome a challenge like a riddle or a puzzle. Following the entire series of clues leaves the player at the treasure's location where he can recover a prize.

This concept is inherited from the stories that narrate how pirates hid their fortune in inhospitable places which could only be found following the precise steps written in an eccentric map.

Scavenger Hunt

Scavenger hunt is a game similar to treasure hunt but the importance is shifted from the final destination to intermediate points. In this case, a list is provided at the beginning and the player must complete it by performing some kind of action at intermediate points. These actions vary from taking a picture to gathering a leaf of the tree that is placed at the intended point.

Geocaching

[44] defines geocaching as *"a high-tech, worldwide treasure hunt (...) where a person hides a cache for others to find. Generally, the cache is some type of waterproof container that contains a log book and an assortment of goodies"*. This activity started on mid 2000 after the U.S. government discontinued its use of Selective Availability in order to make GPS more responsive to civil users worldwide [46]. The first first documented placement of a GPS-located cache was done by Dave Ulmer the 3rd May 2000 in Oregon [33]. Nowadays, there are more than 10 millions of users and more than 2.5 million geocaches are waiting to be found in over 180 countries [34].

The initial containers at specific coordinates have evolved in a variety of caches [32]. For the contents of this project, the most important ones are:

- **Mystery or Puzzle Caches:** solving puzzles gives you the correct coordinates of the cache.

- **Virtual Cache:** these caches are virtual so the location is is the goal rather than a container.

1.2.3   Android

Android is the leading smartphone operating system with about an 80.7% market share followed by iOS, the second most popular one, with a 17.7% [12]. This OS was initially developed by Android Inc. founded in 2003 and two years after it was purchased by Google. Android is released under open source licenses and it is based on the Linux kernel. The first commercial version was released on 2008 and since then 13 official versions have been released [14]. It began as an operating for mobile phones but it has been adapted to other devices, and it currently supports wearables, smartphones, tablets, televisions and cars.

There are two proper ways of developing an Android application. The first one is using Java which takes advantage of all the Android framework APIs available to developers. The NDK toolset offers us another method which consists on implementing parts of an app using native-code languages like C or C++ for achieving a performance improvement. Typically, this is just done on CPU-intensive applications such as game engines, signal processing, and physics simulation [51].

1.2.4   Augmented Reality

As described by Ronald T. Azuma [4], augmented reality are systems *"in which 3D virtual objects are integrated into a 3D real environment in real time"*. In other words, they are

systems which display virtual images on top of a view of the real world. There are several libraries that allow building AR systems for the Android platform like the open source ARToolkit [9] or the proprietary Vuforia [75] from PTC Inc.

As described in Section 1.4, one of the objectives of this project is to explore and learn new technologies. There is not a better way of acquiring an expertise in a field than experimenting and building modules from a lower lever. AR system can be divided in two fundamental parts: marker detection and display of 3D virtual objects. For these two tasks the usage of two existing technologies will be preferred over building it from scratch.

OpenCV

"Open Source Computer Vision Library is an open source computer vision and machine learning software library" [41]. It is a library that holds more than 2,500 computer vision and machine learning algorithms which allow tasks like identifying objects, tracking moving objects, recognizing scenery and establishing markers to overlay it with augmented reality, etc. OpenCV is written natively in C++ and gives support to Android.

OpenGL ES

OpenGL ES is a cross-platform API specification for rendering graphics on embedded systems which descends from the OpenGL rendering graphics managed by Khronos Group [35] [36]. Android includes support for the OpenGL ES API [64].

1.2.5   Geolocation

Geolocation is pinpointing a person's or object's real-world geographic position on the surface of the Earth. It can be done with different approaches like by satellites orbiting around the Earth, by wireless networks or even by mixing several approaches [10].

Fused Location Provider

Google Play Services is a proprietary background service which runs in the Android OS and allows the usage of APIs for popular Google services [24]. The Fused Location Provider, an API included in these services, allows developers an easy addition of location awareness to their apps [63]. It uses a hybrid geolocation approach which translates in higher accuracy and low-power consumption [52].

Geofences

Geofencing consists in creating a virtual geolocated area of interest. For simplification, usually a point with its longitude and latitute is specified together with a radius, in order to generate a virtually fenced circular area. An event will trigger each time the device enters, leaves or stays for a time period in such area.

1.2.6   Haptic Feedback

Ordinary smartphones are devices with a lot of sensors for input purposes but not so many for output ones. Output sensors can be divided according to the human sense they target: sight (screen, notification led and flashlight), hearing (speaker) and touch (vibrator). Sight is the most developed human sense so improvements are normally targeted in that direction instead of the probably most important one which is touch.

Regarding sensors which aim touch, or haptic sensors, not much have been improved since ancient mobile phones. We still just have a vibrator leaving behind other types of cutaneous sensing such as temperature or pressure variations.

Vibrator

A small mechanical device present in almost every smartphone which generate physical perturbations. This module usually rumbles following a pattern for a number of times determined by users or developers.

1.3   Game Preview

As described in Section 1.7, the conceptual design of the game is a phase of the project. Nevertheless, a high-level idea of the intended game design has been formed but it might be subject to modifications. Currently, there is not a well-known application similar to Project Incandescence so an analysis of the whole concept must be performed from the very beginning.

The first thing the user will see when he initializes the application is a brief introduction to the story and mechanics of the game. Then, the player will be headed towards the starting point of the adventure. In that starting point and thanks to augmented reality, a non-playable character will be introduced. This NPC will be the one in charge of presenting the conflict in the story and stating how the player could help solving it. From there, the player will have to explore the campus and, by doing so, meet different NPCs. This discovery will not be

done by showing the locations on a map but by feeling different vibration patterns on the smartphone. On each encounter with these characters, a riddle that lead us to a location will be provided. When we find the solution to a riddle and visit the indicated location, a virtual object will be automatically picked up thanks to geofencing. Gathering every object in the adventure will somehow uncover a final riddle pointing at a spot. This final destination will only be activated when the collection is completed. There, the last NPC will be presented and it will serve as the ending of the story.

From the last paragraph, it can be assumed that Project Incandesce follows a non-linear progress model in an open gameworld. At this moment, this approach is considered the most appropriate one but it could be changed during the game design phase. This game will not present any losing condition so the adventure will take place from the moment it is started until the completion of it. The only conditions that will interrupt the adventure are: manually, by closing the application, and automatically, when exiting the gameworld region which is encompassed by vast geofence. The reuse of architectural elements as AR markers will be explored. In our case, the silhouettes of horses spread though the Riu Sec campus will be our first target. Finally, it should be mentioned that the game will only have a single difficulty level and a single ending since it is developed under tight time constraints.

1.4   Objectives

The course syllabus for VJ1241 - Bachelor's Degree Final Project [81] states that the generic and specific competence to be acquired is:

- *Ability to perform individually and present and defend in front of a university tribunal an original exercise, consisting of a project in the field of video game design and development of a professional nature that synthesizes and includes the skills acquired in the degree.*

The related learning outcomes are:

- *Individually plan and implement an original project of a professional nature in the field of video games and where the skills acquired in the degree are synthesized and included.*

- *Write a report in English, orally present and defend it in front of a university tribunal an original project of a professional nature in the field of video games and where the skills acquired in the degree are synthesized and included.*

From a high-level perspective the principal objective of this project is to learn to develop a professional project which adjusts to the time requirements stated. Moreover, it must include proper documentation. If Project Incandescence is analyzed from a more distinguishing angle, the design and development of an interactive adventure that takes place in a specific physical place, could be defined as the main objective.

From an even more illustrative point of view, the principal objectives targeted in this project can be divided in:

1. **O1 - Technical proposal:** learn to write a document which describes precisely all the work to be done during the development of the project. The proposal must include a plan adequate to the existing restrictions.

2. **O2 - Degree's courses:** the project must include a set of skills learned though the different subjects of the bachelor's degree of video game design and development.

3. **O3 - Geolocated experience:** develop an application which make players experiment a unique adventure bonded to a physical location.

4. **O4 - Recent technologies:** be able to experiment and learn new technologies which could be applied in future projects.

5. **O5 - Android skills:** improve the knowledge on the Android ecosystem and some of its compatible libraries.

6. **O6 - Technical report:** write a technical document that summarizes the work done stating clearly every aspect of the app in conjunction with an evaluation of how it could evolve in the future.

1.5   Justification

In this section, the relation between the proposed project and the courses taught in the pertinent bachelor's degree will be stated. Since Project Incandescence is a global project, almost every subject attended during these years could be mentioned. However, only the most related ones from each branch will be listed; making emphasis on the most related one, VJ1234 - Advanced Interaction Techniques.

1.5.1 VJ1234 - Advanced Interaction Techniques

The main aim of the course was to explain interaction techniques that will be useful for the development of interactive video games [17]. The learning outcomes achieved in that subject were:

- *IR09, IR17 - Describe the different types of existing sensors both input and output.*

- *IR 17 - Build basic programs able to interact with all kinds of sensors.*

- *IR17 - Explain interaction techniques based on digital images.*

The app proposed is a mixture of different interaction techniques that have been made available for users in the last years thanks to smartphones. In other words, the application will combine input and output sensors present in the majority of such devices to provide a singular experience. The fundamentals of computer vision were learned in this subject and inevitably some techniques are going to be applied in the augmented reality module of this project.

1.5.2 VJ1218 - Hyper Media Narrative and Video Games & VJ1222 - Video Game Conceptual Design

These are the only two subjects present in the bachelor's degree related with game design. The learning outcomes achieved in these subjects were:

- *E11, G01 - Use new emerging audiovisual technologies for the production of audiovisual discourses in the field of advertising, graphic design, infographies and multimedia products.*

- *E11, G01 - Understand the basic mechanisms of audiovisual discourses and construction procedures through the assimilation of the expressive and narrative resources that make them possible.*

- *E15 - Define rules that harmonize with the available technological possibilities and which provide fluidity. Set game dynamics as principal elements when making a design.*

- *E15, G10 - Design systems which balance the game mechanics, the objectives to be achieved and possible conflicts inside and outside the game.*

- *G04 - Write a video game's technical document on English.*

These outcomes could be summarized as a correct design of an audiovisual discourse coherent with game mechanics as it is required in Project Incandescence.

1.5.3   VJ1212 - Graphic Expression & VJ1216 - 3D Design & VJ1226 - Character Design and Animation

The learning outcomes achieved in these subjects were:

- *E02 - Understand the different methods of three-dimensional representation on a flat surface.*

- *E04 - Use three-dimensional modeling applications for video games.*

- *E06 - Explain the basics of computer animation.*

- *E06 - Explain the principles of character modeling.*

- *E06, IR07, IR13 - Use applications for character modeling and animation in the context of video games.*

For the creation of correct 3D models to be used in AR module of the game, these learning outcomes are strictly needed.

1.5.4   VJ1204 - Artistic Expression & VJ1223 - Video Game Art

- *E01 - Master various graphical tools and expressive resources.*

- *E01 - Produce freehand sketches of video games elements.*

- *E09, G09 - Control tools for creating graphical representations with high finish levels.*

- *E09, G10 - Create characterized and textured characters.*

Sketches of the characters to model, also called 3D model sheets, are needed prior to the creation 3D models. Moreover, the creation of different 2D elements for the application's interface is needed, for example the logo.

1.6   Tools

This section lists the tools that will be used during the design and development of this application. These tools can be grouped according to the topic they help us with.

1.6.1 Programming

Android Studio: is the main tool for the development of the game. Android Studio is the official IDE for Android application development which allow developers to build apps from scratch.

Android Debug Bridge: tool that lets us communicate with a connected Android-powered device.

Notepad++: an open source program that allows the edition of source code in multiple languages.

1.6.2 3D Modeling and texturing

Autodesk 3ds Max 2016: software for 3D modeling which has been used in VJ1216 and VJ1226.

Blender: an open source 3D graphics and animation software used in VJ1212.

1.6.3 2D and UI Design

Adobe Photoshop: the raster graphics editor used in VJ1204 and VJ1223.

Inkscape: an open source vector graphics editor.

Layout Theme Editor: tools included in Android Studio which help developers in the production of user interfaces.

1.6.4 Debugging, Testing and Profiling

Android Device Monitor: a tool that provides a graphical user interface for several Android application debugging and analysis tools.

Performance Profiling Tools: tools included in Android Studio which record and visualize the rendering, compute, memory and battery performance of an Android app.

1.6.5 Documents

MiKTeX: an open source implementation of TeX/LaTeX and related programs for Windows.

Texmaker & TeXstudio: open source cross-platform integrated writing environments for creating LaTeX documents.

Google Slides & Microsoft PowerPoint: programs for creating and editing slide show presentations.

1.6.6 Version Control System

Git: an open source distributed version control system.

GitHub: a Git repository hosting service which allows students to host up to 5 private repositories for free.

1.6.7 Video

SCR Screen Recorder Pro: an Android screen recorder app with plenty of customization options.

Adobe Premiere Pro: the video editing application used in VJ1230.

1.7 Project Plan

1.7.1 Project Schedule

This section presents the intended schedule for Project Incandescence. Figure 1.1 summarizes the contents of this section in a Gantt chart. The totality of the project can be divided in five different phases, each one containing a different set of tasks.

Phase 1 - Documentation

Elaboration of all documentation related with the project including a technical proposal, a technical report and a presentation which contains a video for the project's defense.

Table 1.1 Phase 1 - Tasks Breakdown.

Documentation				
ID	Task	Period	Total Days	Total Hours
TP	Technical Proposal	30-03-2016 to 07-04-2016	9	12
TR	Technical Report	02-09-2016 to 22-09-2016	18	36
PDV	Project Defense Video	23-09-2016	1	1.5
PDP	Project Defense Presentation	23-09-2016 to 26-09-2016	4	3.5
Total				**53**

Phase 2 - Game Development

The development of the actual application. This phase will be divided in 3 subphases: base application, augmented reality and geolocation. The augmented reality subphase can be furthermore divided into two modules, marker detection and 3D model display, while the geolocation subphase can be divided into geofencing and haptic feedback modules.

Each of the 5 game's modules will be composed of four common tasks: analysis, design, implementation and testing. For simplicity, Table 1.2 shows a breakdown of the days and hours to be spent in each module but not on each specific task.

Table 1.2 Phase 2 - Modules Breakdown.

Game Development				
ID	Module	Period	Total Days	Total Hours
MD	Marker Detection	02-05-2016 to 15-05-2016	14	30
3DD	3D Model Display	16-05-2016 to 29-05-2016	14	30
GF	Geofencing	30-05-2016 to 14-06-2016	16	36
HF	Haptic Feedback	15-06-2016 to 19-06-2016	5	9
BA	Base Application	25-07-2016 to 14-08-2016	21	45
Total				**150**

Phase 3 - Game Design

Conceptual design of game aspects such as story, controls, game world division, riddles, tutorial, game screens and game flowcharts. All this aspects will be comprised in a brief GDD and added as a chapter in the technical report.

Table 1.3 Phase 3 - Tasks Breakdown.

Game Design				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
GDD	GDD production	20-06-2016 to 10-07-2016	21	45
Total				**45**

Phase 4 - Game Art

Elaboration of all visual assets present in Project Incandescence.

Table 1.4 Phase 4 - Tasks Breakdown.

Game Art				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
VI	Visual Identity	11-07-2016	1	1
LG	Logo	11-07-2016	1	2
UX	User Experience Design	12-07-2016	1	1
UI	User Interface Design	12-07-2016	1	2
TS	Tutorial Screens	13-07-2016 to 18-07-2016	6	12
3DM	Modeling a 3D character	19-07-2016 to 21-07-2016	3	9
3DT	Texturing of the 3D model	22-07-2016 to 24-07-2016	3	3
Total				**30**

Phase 5 - Game Evaluation

Integration tests and evaluation of the developed application.

Table 1.5 Phase 5 - Tasks Breakdown.

Game Evaluation				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
IT	Integration testing	31-08-2016 to 19-09-2016	3	9
EA	Evaluation of application	20-09-2016 to 21-09-2016	2	6
Total				**15**

Fig. 1.1 Project Schedule Gantt Chart.
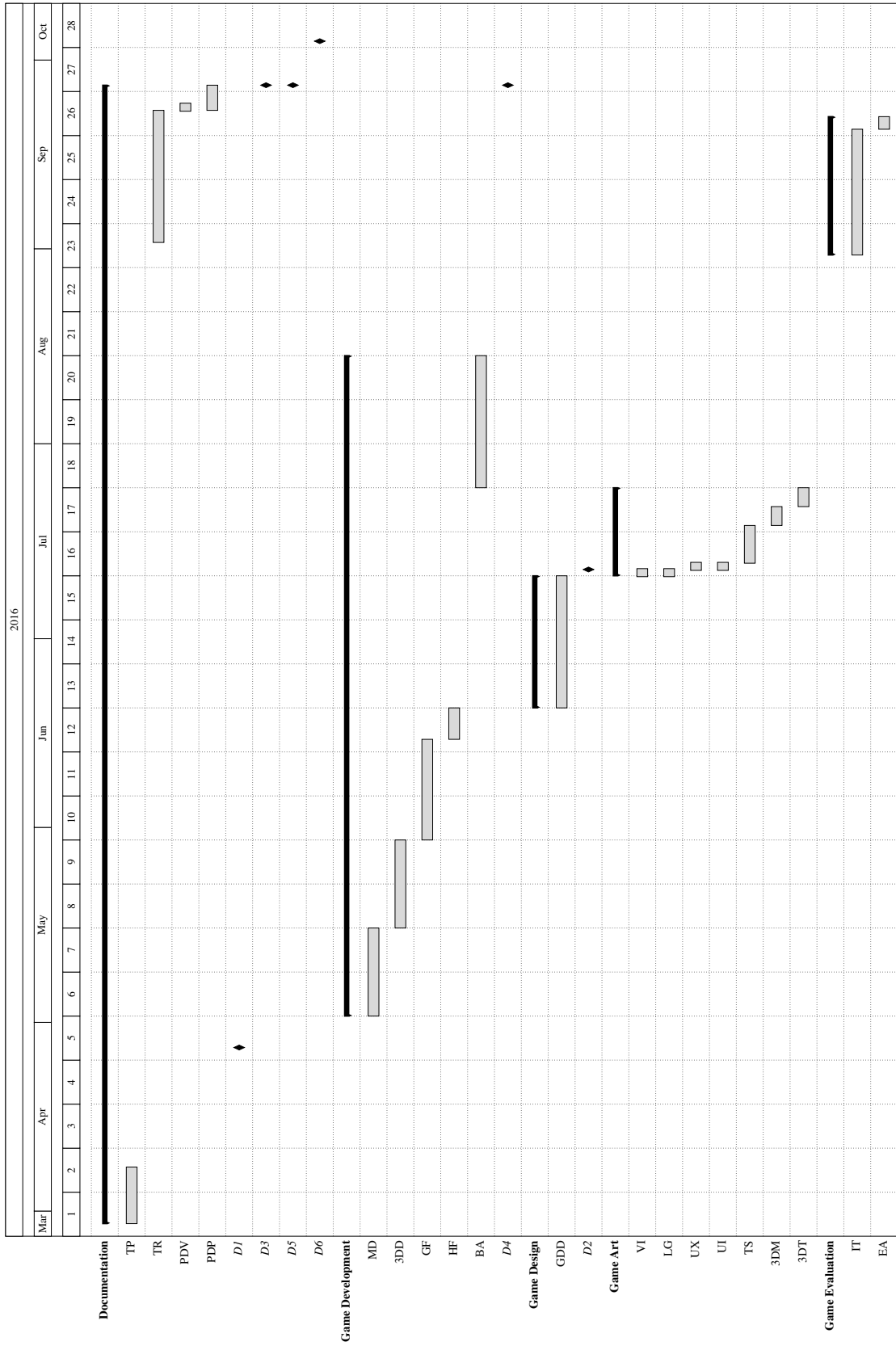
1.7.2 Deliverables

This project will have six different deliverables as outcomes of the project's development.

1. **D1 - Technical Proposal:** a technical document describing the intended work for the final project. *Due date: 26th April 2016.*

2. **D2 - Game Design Document:** a short but descriptive document giving an overall view of the developed game including narrative, mechanics, art and other aspects. *Due date: 11th July 2016.*

3. **D3 - Technical Report:** a technical document describing all the work done in this final project. *Due date: 26th September 2016.*

4. **D4 - Android Application:** the Android compatible app resulting from the completion of this project. *Due date: 26th September 2016.*

5. **D5 - Project Defense Video:** a two-minutes video showing a gameplay of the game developed. *Due date: 26th September 2016.*

6. **D6 - Project Defense Presentation:** A visual presentation with a set of slides in front of the evaluation tribunal. *Due date: 3rd October 2016.*

Table 1.6 Deliverables Breakdown.

Deliverables				
ID	*Deliverable*	*Related Objectives*	*Related phases*	*Due Date*
D1	Technical Proposal	O1	1	26-04-2016
D2	Game Design Document	O3	3	11-07-2016
D3	Technical Report	O6	1	26-09-2016
D4	Android Application	O2, O3, O4, O5	2, 3, 4, 5	26-09-2016
D5	Project Defense Video	O2, O6	1	26-09-2016
D6	Project Defense Presentation	O6	1	03-10-2016

1.7.3 Software Development Methodology

The software part of this project will be developed under an iterative waterfall model. Our application can be divided in 5 independent modules. Each of these modules can be divided in a particular iteration including test and evaluation of each module separately. The provision of feedback and adjustments is also available to preceding modules thanks to the methodology

selected. The whole development could be summarized as being more flexible and less risky than if chosen a pure waterfall model.

In all the software development, Git will be used as the distributed version control system. Moreover, the project will be hosted in a GitHub private repository for backup purposes. Code conventions stated in the Android Code Style for Contributors [17] will be followed for easing the task of source code readability and maintenance.

1.7.4 Budget

This section includes a breakdown of all the costs related with the project execution.

Human Resources

The cost associated with human resources is obtained from the average and the maximum 2016 contribution bases and rates stipulated by the Government of Spain[1], for the computer engineer and supervisor respectively, plus some supplements associated to the specialization in several topics such as augmented reality or geolocation (25% of base). The accomplishment of this project lasted an overall of 300 hours, 3 hours per working day, distributed along 5 months as detailed in Section 1.7.1.

Table 1.7 Human Resources Costs.

Human Resources					
Role	*Base/h*	*Supplements/h*	*Total/h*	*Hours*	*Cost*
Engineer	14.72€	3.68€	18.40€	300	5,520.00€
Supervisor	22.76€	5.69€	28.45€	6	170.70€
Total					**5,690.70€**

Hardware

Hardware owned not only for the development of Project Incandescence but also for the evaluation of it.

Software

All the software needed during the project, which includes: operating systems, development software, testing software, and the software required for the elaboration of this document,

---

[1]http://www.seg-social.es/Internet_1/Trabajadores/CotizacionRecaudaci10777/Basesytiposdecotiza36537/index.htm

Table 1.8 Hardware Costs.

Hardware owned			
Concept	*Price per unit*	*Amortization (3 years) per month*	*Cost (5 months)*
Personal Computer (x 1)	400.00€	11.11€	55.55€
Smartphone (x 1)	200.00€	5.55€	27.77€
Total			**83.32€**

is either open source, free, student-licensed or the university owns licenses for it. In other words, the associated expenses of them are 0.00€.

Consumables

This is the cost of all the consumables that will be spent. Office material includes items like sheets, pens, staples and printer ink.

Table 1.9 Consumables Costs.

Consumables	
Concept	*Cost*
Office material	50.00€
Total	**50.00€**

Other Costs

Encompasses any other cost which cannot be fitted in previous sections. Indirect costs incorporate expenses linked to the project (phone, maintenance, electricity, internet, etc.) and are represented by 20% of the human resources costs.

Table 1.10 Other Costs.

Other Costs	
Concept	*Cost*
Indirect costs	1,13814€
Work place	0.00€
Total	**1,138.14€**

Total

Table 1.11 comprises all the costs to be incurred from the first to the last day of development. The final budget includes a 15% associated with risks and a 20% associated with benefits. The price before taxes and after taxes (21%) is also included in Table 1.11.

Table 1.11 Total Costs.

Total	
Concept	*Cost*
Human Resources	5,690.70€
Hardware	83.32€
Software	0.00€
Consumables	50.00€
Other Costs	1,138.14€
Total before risks and benefits	6,962.16€
Risks (15%)	1,044.32€
Benefits (20%)	1,392.43€
Total before taxes	9,398.91€
Total after taxes (21%)	**11,372.68€**

1.8   Risk Management

The next tables show the most probable risks associated with Project Incandescence including their related contingency plans.

Table 1.12 Risk - 01.

Risk ID	R_01	Title	Schedule of professional internships
Category	Project Team	Likelihood	Medium
Description	Working full-time during professional internships.		
Contingency Plan	Readjust the project's schedule in order to recover hours before or after doing professional internships.		

Table 1.13 Risk - 02.

Risk ID	R_02	Title	Wrong time estimation
Category	Project Development	Likelihood	Medium
Description	Inadequate project's schedule regarding time constraints.		
Contingency Plan	If lack of hours, focus on essential aspects. If excess of hours, add additional features.		

Table 1.14 Risk - 03.

Risk ID	R_03	Title	Wrong costs estimation
Category	Project Development	Likelihood	High
Description	Wrong budget estimation associated with this project.		
Contingency Plan	Try to lower the costs and abandon the project if costs rise too much.		

Table 1.15 Risk - 04.

Risk ID	R_04	Title	Missing skills
Category	Project Team	Likelihood	High
Description	Some abilities are missing for the completion of the project.		
Contingency Plan	Learn missing skills.		

Table 1.16 Risk - 05.

Risk ID	R_05	Title	Project files deletion
Category	Project Team/Hazard	Likelihood	Low
Description	Some or the totality of files are unintentionally lost.		
Contingency Plan	Restore a backup.		

Table 1.17 Risk - 06.

Risk ID	R_06	Title	Equipment fault
Category	Project Development	Likelihood	Medium
Description	Working or testing devices break or malfunction.		
Contingency Plan	Repair or buy new equipment.		

Table 1.18 Risk - 07.

Risk ID	R_07	Title	RT performance shortfalls
Category	Project Development	Likelihood	High
Description	Real-time requirements on the AR module are not met.		
Contingency Plan	Perform optimizations to reach the requirements.		

Table 1.19 Risk - 08.

Risk ID	R_08	Title	Complexity of functionalities
Category	Project Development	Likelihood	Medium
Description	Some functionalities are more complicated than expected.		
Contingency Plan	Reduce the complexity of implementation by means of libraries or other mechanisms.		

Table 1.20 Risk - 09.

Risk ID	R_09	Title	Missing AR markers
Category	External	Likelihood	Low
Description	Intended AR markers get removed or relocated in the campus.		
Contingency Plan	Readjust the game's design if a few AR markers are moved. Ask for permission, print and place placeholder markers if they are removed or needed.		

Table 1.21 Risk - 10.

Risk ID	R_10	Title	Technical proposal rejection
Category	Project Development	Likelihood	Medium
Description	The technical proposal document get rejected by reviewers.		
Contingency Plan	Consider and adjust weak points of the technical proposal. Resubmit the technical proposal with modifications.		

Chapter 2

Game Design Document

This chapter details the game design aspects of Project Incandescence. It resembles the Game Design Document present during the pre-production and production of video games. Attention, this chapter spoils a lot of details regarding the application itself and the narrative behind it. It is advisable that you experience Project Incandescence before reading this chapter in order to form a pure and unbiased opinion of it.

2.1 Story Overview

Anais is a young Chilean lady living in the colorful city of Valparaiso. One of her early memories is from Vitalia, a neighbor of her that used to live across the street. Some years ago, Anais's father got sick so she had to go the pharmacy in order to buy him his daily medicines. This happened during the last days of January which are the hottest ones of the year. The temperature encouraged Anais to stay in the establishment more than necessary because they had air conditioner. Vitalia took advantage of that brief time periods and illustrated the child with some of the beauty present in different places of South America. Anais loved hearing such stories so it quickly turned into a habit. Each day she told her a story about her early travels through the continent when she was learning folkloric remedies from shamans while becoming a chemist.

One day, about 3 years ago, Vitalia died and his descendants began a dispute for her inheritance. Recently, a judge resolved this conflict and her house was sold. The new owner threw everything that he considered it was not valuable. Some weeks ago, Anais was getting back home from her classes when she spotted a box besides the waste container with a unmistakable handwriting. It was from Vitalia, so she grabbed it and discovered that the box was composed of smaller boxes and folders.

On one of those folders included a strange disk with a ciphered message and an annotation that said, "feel the outflow emanating on the river's bank since the 8 make visible the splitting of the 4".

She made an extensive research and found the uniqueness of the Riu Sec campus. It is the place with the thinnest Earth's crust and which at the same time is really close to a river basin. Thanks to global warning and to the proximity of the asthenosphere, she developed an experimental component that employing a normal camera allows the processing of signals obtained in micro-mirages caused by heated air outside the visible spectrum. This heat hazes only occur on places where the lithosphere is remarkably thinner which are areas prone to microseisms that can also be translated thanks to the phone's accelerometer.

Anais cannot afford a travel to the exceptional location in Spain so, would you kindly help her decipher the truth?

2.2   Gameplay

In this adventure, the player will undergo the following sequence of steps:

1 Travel to the starting point: a map is presented with a marker pointing at the starting point so that the player can go there for the adventure to begin.

2 Call to adventure: set of slides which introduces the player to the story.

3 Tutorial: near the starting point, a meeting point will be activated so that the player can get in touch with location and haptic mechanics. In this meeting point he will have the first contact with the AR system. There, the NPC dialog will lead to a location near the starting point where the player will learn the second sets of mechanics: gathering collectibles and deciphering a part of the ciphered message.

4 Open world: from this point on, all the meeting points will be activated allowing the gathering of the rest seven collectibles.

5 Decipher the message: deciphering the totality of the text will lead to the first ending and unlock the final meeting point.

6 Last meeting point: the last encounter with a forebear which is the second ending.

2.3   Character Profiles

The following tables describe precisely each of the characters which are part of this story.

Table 2.1 Anais Gónzalez.

Name	Anais Gónzalez	Age	19
Occupation	Developer of Project Incandescence	Relevance	Lead
Role	Sender	Physical Appearance	🧑🏽
Background	Raised in a worker family without much income. She lives with her parents and does not have siblings. His father has a chronic disease so she works as a shop assistant in the mornings in order to help to her family economically and to pay her studies. She is doing a degree of Computer Programming Technician at the AIEP.		
Personality	Extremely curious and intelligent. Loves literature and documentaries specially of myths and physics. She is quite extrovert but at the same time she enjoys time alone.		

Table 2.2 Vitalia Rojas.

Name	Vitalia Rojas	Age	92
Occupation	Traditional pharmacist	Relevance	Support
Role	Mentor	Physical Appearance	👵🏻
Background	When she was young she traveled through all South America learning home remedies from shamans. On her twenties she suffered hallucinations with some brews and for some people that shady reputation accompanied her until the day of her death.		
Personality	Naive, endearing and dreamer. She was always smiling and embraced life as no one else.		

Table 2.3 Forebears.

Name	Forebears	Age	Unknown
Occupation	Carry tectonic plates	Relevance	Secondary
Role	Helper	Physical Appearance	🗿
Background	Once they were alive but now they revere nature and living creatures.		
Personality	Incredulous. Each one have a slightly different personality and form of expression. The final one seems much closer to the player (like a direct ancestor). They do not retain any corporeal characteristic from their mankind lives like genre, race...		

2.4 Game World

Crucial aspect of the game since dealing with a real space and cannot create or drastically alter it as in a traditional videogame. Anyway, we can enhance some aspects and make the player discover new places.

Fig. 2.1 Area of the Riu Sec campus.

Figure 2.1 shows that about 0.55 km^2 are available and useful at the Riu Sec campus. This convenient area must hold 19 locations: for each of the 8 elements a location for the ancestor plus another for gathering the correspondent collectible, the starting point, the second ending meeting point and a collectible area that is duplicated. In other words, we must scatter those 19 locations within a 0.567 km^2 region. The precise spots are specified in Table 2.4, Table 2.5 and Table 2.6.

Table 2.4 List of special points.

Special Points				
ID	*Building*	*Latitude*	*Longitude*	*Duty*
SP_01	ESTCE	39.9927254	-0.0664335	Start
SP_02	CS	39.9959526	-0.0661575	End

Table 2.5 List of meeting points.

Meeting Points					
ID	*Building*	*Latitude*	*Longitude*	*Element*	*Symbol*
MP_01	ESTCE TD	39.9923363	-0.0665829	Crystal	
MP_02	ESTCE TC	39.9927864	-0.0660407	Water	
MP_03	ESTCE TI	39.9935722	-0.0678160	Metal	
MP_04	Library	39.9936562	-0.0691405	Earth	
MP_05	FCJE C1	39.9945717	-0.0670862	Energy	
MP_06	Rectorate	39.9949112	-0.0687373	Wood	
MP_07	FCS	39.9951439	-0.0716149	Fire	
MP_08	Espaitec 2	39.9941940	-0.0735810	Air	

Table 2.6 List of gather points.

Gather Points						
ID	*Name*	*Latitude*	*Longitude*	*Related MP*	*Distance from MP*	*Symbol*
GP_01	Pyramid	39.9921580	-0.0661220	MP_01	125 m	
GP_02	Fountain	39.9935470	-0.0683910	MP_02	215 m	
GP_03	Bolts	39.9951172	-0.0665804	MP_03	215 m	
GP_04	Ant	39.9953590	-0.0679890	MP_04	215 m	
GP_05	Solar Tree	39.9949379	-0.0690358	MP_05	175 m	
GP_05'	Solar Tree	39.9939156	-0.0732810	MP_05	530 m	
GP_06	Surrounded Tree	39.9955510	-0.0712090	MP_06	220 m	
GP_07	Fire Stallion	39.9935990	-0.0736490	MP_07	245 m	
GP_08	Deciduous	39.9940710	-0.0676410	MP_08	515 m	

The locations present in the previous tables are shown in Image 2.2. This map also shows the two most probable itineraries that players will follow. Both itineraries are about 2.25 km long in straight line which implies a distance of about 3 to 4 kilometers in a normal gameplay. Also, both routes have a common resting points near meeting points and gathering points for example GP_05 and GP_8. Not a direct reference of the resting point will be stated but hopefully players that need a rest will take advantage of them. Initial and final locations will create a strong bond between the game story and the physical environment. This link will supply coherence and cohesion to the whole story, reducing the barriers that exist between the real world and the experience's narrative. Finally, a geofence will encompass all the playing area so that resources can be freed automatically if the player leaves the region.

Fig. 2.2 Locations with most probable itineraries [1].

[1]https://drive.google.com/open?id=1riLOVGSlBZGlQ4HQU4QeZiOjxMU

2.5   Player Controls

In Project Incandescence controls can be divided into two groups:

1 **Interaction with the screen:** ordinary interactions with interfaces available in touch devices. It includes gestures like taps, scrolls and flicks.

   (a) Navigation: normal navigation between screens.

   (b) Disk screen: is where players must decode the ciphered message. They will achieve the goal by selecting which segment of text they want to decipher and spinning the inner disk.

   (c) Dialogue: advance in the dialogue of a forebear by touching a button.

2 **Rest of interactions:** interactions less common that are available thanks to sensors.

   (a) Augmented Reality: point the smartphone's camera at the marker so that the forebear can appear and interact with the user.

   (b) Geolocation: depending on the player's location in the real world he will receive vibrations and/or gather collectibles.

   (c) Haptic Feedback: get locations feedback not by staring at a map but by feeling vibrations.

2.6   Game Experience

Project Incandescence aims to offer a fresh way of consuming adventures. This feature should be seen as double-edged sword. At first we can think of it as an appealing experience for players thanks to its newness. However, humans tends to be cautious and reluctant to changes. In order to address this problem, players must explore and learn core game mechanics in an easy way but by themselves without explicit instructions.

The game contains an concealed tutorial at the beginning of the adventure which is embedded with the discovery of the first two locations. By taking this approach, we make sure that the users understand essential functions at the starting point. In our case, this is accomplished by fixing the initial location where the first cutscene is triggered. After a brief introduction to the story, the first meeting point is activated, and since players are inside the influence area, they are introduced to haptic feedback. In the map which determines the

starting point, players will see the area where they should return to pick the first collectible. When reaching that first meeting point, they will discover the mechanics related with the camera and augmented reality. There will be two possible signs that trigger AR mechanics in the area of the first meeting point. Finally, they will just have to return to the previously seen pyramid and gather the first collectible.

The adventure created have a series of aspects which can be seen as belonging to serious games. The first one of them is the rise of awareness on how humans are destroying nature with references to phenomenons like global warming or extinction of species. Another aspect, and probably the most direct one, is to inform in a subtle way of the existence and location of meeting points in case a fire arise. Probably this aspect may be seen as a nonsense but the truth is that nowadays almost no student seems aware of them. Finally, this game introduces cryptography, specifically substitution ciphers, to an audience that might find it useful on some bachelor's degrees.

It should also be mentioned that this app does not include any kind of audio neither music nor sound effects since it is played outside. It have been designed following this principle since sometimes is difficult to perceive sounds in the streets while other we could bother the rest of pedestrians.

2.7   Game Mechanics

Game mechanics present in this experience have been described in previous sections, anyhow now they will be disassembled into finer details.

- **Augmented Reality**: is the mechanic in charge of making the connection between the creatures of the underworld and us. Dialogs will work like a teleprompter, touching the screen will advance the text so that every player can consume it at its own path. The usage of AR with markers allows a finer sensation of depth for 3D characters. Models will scale as the marker scale varies simulating the distance to the marker.

- **Geolocation**: mechanics that are related with the real-world geographic location of the player.

    – Continuous Distance: distance between the player and the closest meeting point, when the player is within the influence area (120 m) of at least one. This distance will be translate to vibrations following a heart beat pattern, so as we get closer the heart rate will increase. The heart rate will have a range from 10 to 130 beats per minute.

      – Geofences: virtual barriers that let us discriminate when we trespass them. They
are useful for the pickup of collectibles and to delimit the playable area.

- **Puzzle**: for this application the Alberti Cipher is going to be chosen as the singular puz-
zle through the adventure. This decision was taken thanks to it being a straightforward
puzzle of easy comprehension, very visual and it is related to the story period. At the
beginning of the 20th century, the Mexican Army was using a more complex version
of this polyalphabetic cipher [80]. That means that is was somehow known, at least a
simplified version, during the years that Vitalia spent traveling through the continent.
In our case, the two disks consists on 32 symbols corresponding to the 27 letters of the
Spanish alphabet plus the 5 accented vowels. These 32 symbols are divided in 8 groups
corresponding to each of the elements present in the adventure. When a collectible
is picked, each element will be automatically placed in the corresponding inner disk
position limiting to only 4 possibilities of deciphering for the fragment corresponding
to that element.

2.8   Game Script

Dialogs will be a crucial part of this experience because they will be the ones in charge of
giving directions to users. For each of the forebears, we include a main speech including a
final hint that is "deciphered" in the background and given to players if they do not gather
the correspondent collectible during a time interval. Dialogs are written in Spanish since the
game will just be released in that language.

<div align="center">

Listing 2.1 Script Meeting Points

MEETING POINT 1

FOREBEAR 1

</div>

...y esa es la clave de la vida. Vaya cara se le ha quedado. Cualquiera
    diría que me puede ver con ese cachivache. No le parece fascinante
    esa cristalera. Cómo algo opaco como pueden ser granos de arena son
    purificados para dejarlos transparentes. Quién sabe, aunque por aquí
    se hacen azulejos de forma similar.

Me pregunto qué hubiese pasado si los egipcios hubieran podido producir
    grandes cantidades de fayenza. Bueno le tengo que dejar que hoy nos
    han convocado. Eso sí, tenga usted claro que...

FOREBEAR 1
(ciphered)

... a pesar de ser transparentes , los cristales ocultan grandes secretos , fíjese en el Louvre . Bueno quizá sea cosa de las pirámides y no de los cristales .

MEETING POINT 2

FOREBEAR 2

Así que este es mi sitio para hoy . ¡ Este , este , este ! , todos me han dicho , aunque hubiese preferido que estuviese un poco más al este de verdad . Anda , me dirá que no le gustaría estar refrescándose debajo del agua . Vaya cosas que le estoy diciendo cuando ustedes ni nos quieren poder ver , como a tantas otras verdades . Menos mal que no sufren tanto calor como nosotros , por ahora . Eso , y que han aprendido a canalizar el agua para hacer puntos de refresco donde habitan .

Cambiando de tema , que cosas tienen los lenguajes . Todos los años mueren palabras mientras otras denotan varios conceptos . Menos mal que hay algunos lenguajes inequívocos ...

FOREBEAR 2
(ciphered)

... como si le digo que halle la x para encontrar lo que anda buscando . Bueno no quiero que piense que le mando a multiplicarse por algo , pero estamos a punto de comenzar .

MEETING POINT 3

FOREBEAR 3

Vamos a ceñirnos al soporte , como si de ello dependiesen nuestras vidas . Y es que siempre ha sido igual . Para tirarnos al vacío nos atamos hasta a una cuerda quebrada buscando una seguridad que debería brotar de nuestro interior . Este concepto es aplicable desde a gente como tú , estancada en su zona de confort , hasta al miedo de las instituciones que intentar usar anclajes al suelo para mantener siempre erguida su posición .

A veces es necesario soñar y arriesgar para uno ser consciente de hasta dónde puede llegar. Por un momento supongamos que eres capaz de cruzar...

FOREBEAR 3

(ciphered)

...ese jardín. Quizás alguna clase de magia te espera al otro haciendo que todo encaje como si fuese un engranaje.

MEETING POINT 4

FOREBEAR 4

¡Qué suerte tengo! Siempre me dejan los sitios más pesados. Menos mal que la fe mueve montañas porque si no... A cargar con el peso del conocimiento.

Y pensar que con el intelecto que tenemos, estamos sentenciando a especies enteras. Ya ni debajo de la tierra se pueden esconder. Bueno que le voy a contar, si hasta nos sentenciamos a nosotros mismos según la apariencia.

A mí me arrebataron...

FOREBEAR 4

(ciphered)

...a mi fiel compañera. La encarcelaron y emplazaron a las puertas de la muralla que separa el mundo de las ninfas del que usted se haya.

MEETING POINT 5

FOREBEAR 5

Sea usted bienvenido. Al final va a ser ciertos los chismes que rondan entre mis hermanos. Un numen que parece capacitado para avistarnos aunque no tanto para tratarnos. Será cosa de ese atavío moderno que posibilita la visión de los sempiternos. O de ese singular faro que por el día es avaro mas por la noche exhibe sus bienestares sin vestigio alguno de fracaso.

A veces las cosas vienen...

FOREBEAR 5
(c i p h e r e d)

...en pares y este parece ser otro caso. Una apaciblemente cerca del río se halla, mientras la otra intenta huir de la petrificada garra.

MEETING POINT 6

FOREBEAR 6

Y pensar que pude vivir sin hacer nada fingiendo que a mi alcance aquello no estaba. Ahora cada día me siento más atrapada cumpliendo penitencia sin estar a ello obligada. Quisiera estar ahí para por última vez inspirar el aire como si estuviese en mi niñez. Aunque eso ya no ocurrirá ni aunque me traslade a mi añorada Panamá.

Me cuesta imaginar que llegó el punto en el que...

FOREBEAR 6
(c i p h e r e d)

...el germen de vida y ciencia está rodeado en el ojo de lo que parece la llave a la transparencia.

MEETING POINT 7

FOREBEAR 7

Left, right, left, right, left, left, left, left.

Bueno ya estamos otra vez. Que no, que no hay ninguna criatura de esas por aquí. ¿Cómo? ¿Me estabas buscando a mí? No, no me lo creo. Sinceramente te digo que hace un rato que no veo a ninguno de esos animalitos que os da por recluir en espacios angostos. No puedo entender como la gente disfruta capturándolos, encerrándolos, transfiriéndolos a un profesor tenebroso que fomenta el canibalismo para acabar entreteniéndoos con luchas entre ellos. Se ve que a día de hoy seguimos estancados desde la época romana.

Como veo que no se va a marchar hasta que le diga algo, le confesaré que uno de esos caballos con llamas como crines fue al último que vi, y si mal no recuerdo...

FOREBEAR 7

(c i p h e r e d)

. . . había conseguido trepar por un jardín vertical .

MEETING POINT 8

FOREBEAR 8

Por fin te dignas a aparecer . Ya había perdido toda esperanza y estaba a
    punto de marcharme . Pero que estoy diciendo , si no tengo ninguna
    prisa . Ha resucitado en mí la vieja manía de luchar contra el tiempo .
    Créeme , al final lo mejor es disfrutar del momento ya que de un
    plumazo puedes perder desde la salud hasta el trabajo . Recuerda lo
    que le pasó al árbol mejor cuidado .

En un día de otoño , una bocanada de aire le arrebató sus hojas y desde
    aquel momento no ha podido remontar el vuelo . Y eso que . . .

FOREBEAR 8

(c i p h e r e d)

. . . siempre está rodeado de gente y disfrutando del placer que le brindan
    los sentidos . ¡Ay! , si te contase yo lo que he sufrido .

SPECIAL POINT 9

FOREBEAR 9

Tanto tiempo sin verte . ¿De verdad eres tú? Recuerdo como si fuese ayer
    cuando empezaste a hablar . También recuerdo como te gustaba que te
    hicieran cosquillas y el sonido de las carcajadas que brotaban al hac
    értelas . Podría pasarme días rememorando buenos momentos que
    compartimos pero el tiempo apremia .

Lo primero es confesarte que no eres la única persona que conoce nuestra
    existencia . Si miras a la fachada del edificio a tu izquierda podrás
    comprobar que algunos te llevan años de ventaja . Aun así no te
    desesperes porque tienes toda una vida para alcanzar la verdad ,
    siempre que te enfrentes a ello sin miedo y con algo de tenacidad . Y
    así por fin revelar , lo que lleva casi tres milenios oculto solo
    visible para los que estamos en el otro mundo .

Noto como el vínculo que nos une se está debilitando . . . antes de que me vaya recuerda, ni las cosas ni las personas son siempre lo que parecen.

2.9   Cutscenes

Only two cutscenes will be present in Project Incandescence. They will take the form of a sideshow as they are commonly present in smartphone's apps as their introduction. Players will be able to swipe left/right to view the previous/next slide.

Crucial parts of a story, the beginning and the end, are going to be told with these brief presentations so information must be carefully picked and arrange in order to tell as much as possible with so little. A textual summary of these sideshows is presented in Table 2.7 and Table 2.8.

<div align="center">Table 2.7 Beginning Cutscene.</div>

1	2	3	4
Anais presentation	She lives in Valparaiso, Chile	Father's illness some years ago	Each day after her classes, she went for pills in spite of the hot weather
5	6	7	8
There she met Vitalia (description)	Vitalia told Anais stories of her travels	Some years ago Vitalia died	Dispute of inheritance
9	10	11	12
Judge resolved and house sold	New owner threw belongings	Grabbed a box and in one file a cipher	Vitalia handwritten annotations
13	14	15	16
Connection with 4 classical elements	Special location and global warming	Developed a exclusive app	Help her to uncover if there is something more

Table 2.8 End Cutscene.

1	2	3	4
Anais does not understand	She is sorry for not getting in touch before	Which remainder? Doubt about adventure	Examination of Vitalia's files
5	6	7	8
Inspect your personal data	Please, stop investigating	Thanks for collaborating	Farewell

2.10   Art

Material design is the design language that has been chosen for developing the UI of this app following the recent trend in this kind of devices. Emojis [23] will be a central part of the interface since they are embedded in the operating system which allows a reduction of the space requirements. Also, the use of emojis makes available the presence of rich glyphs in the notifications issued by the application and reduces the effort to be exerted in the art production phase.

Following this minimalist approach, a low poly pattern is embraced for our 3D meshes. We can resemble those 3D meshes with traditional origami figures. This approach suits the capabilities of graphics rendering on not very powerful smartphones. As in paper folded pieces, textures will be simple just including a composition of plain colors.

Usually in thermal imaging, a color palette ranging from purplish colors to yellowish colors is used. These color scheme will be adopted for the application theme and a green tone might be also employed for giving further contrast.

For further details, Section 4.1 presents a more complete description of Project Incandescence visual aesthetics.

Chapter 3

Game Development

This chapter corresponds to details regarding the second phase of Project Incandescence. The game development phase is the one which holds half of the total time invested in Project Incandescence from the five ones contained. From time perspective, it can be considered the most important one of the whole project. This chapter starts with a high-level overview of the application and consecutive sections provides insights of each one of the modules developed at a lower-level.

3.1 General Overview

The overall system could be seen as one receiving 3 types of inputs and responding to them with 3 types of outputs.
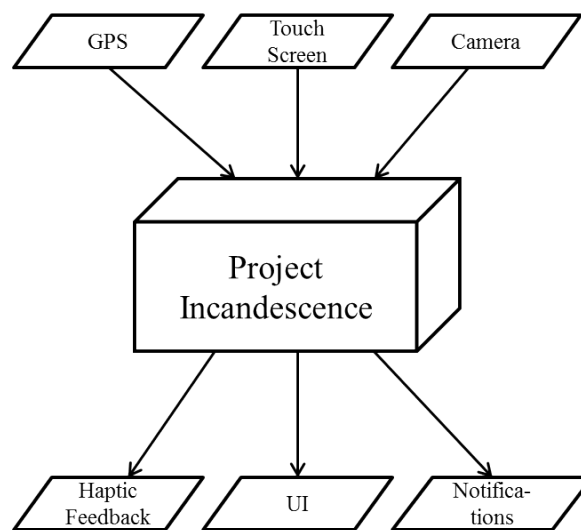
Fig. 3.1 Project Incandescence as a black box.

As represented in Figure 3.1, we can consider Project Incandescence as a black box which collects inputs from GPS, camera and touch screen. Those signals get processed accordingly and the system delivers output in form of haptic feedback, user interfaces and notifications.
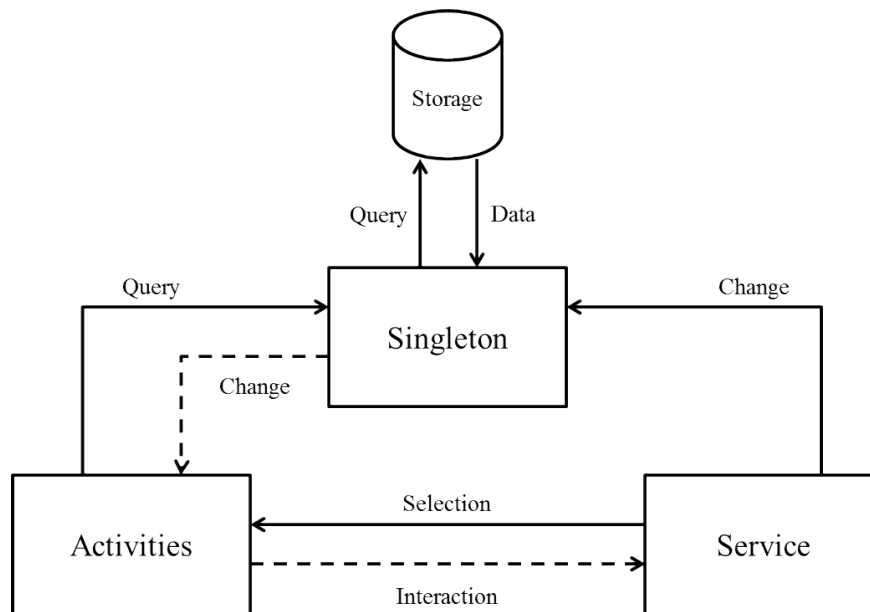
Fig. 3.2 Architectural design of the application.

Getting inside the black box will reveal the system's architecture. Figure 3.2 shows a high-level representation of the interconnected components which constitute the application. The application's architecture was designed taking as basis the Model-View-Controller pattern [76]. In our design, the model component corresponds with a singleton instance that preserves the game state, our view correspond to the variety of activities available in the app, and the controller is the background service in charge of presenting activities and making changes on the model. The diagram also includes a abstraction of the persistent storage which is available only to the model. These components will be described in detail on the sections of this chapter.

3.2   Base Application

This section consists on a detailed description of the application which communicates and serves as a base for more specialized modules. Each one of the activities described in this section have been developed with support libraries [70] so that they could bring material design to pre-lollipop devices.

3.2.1 Main Activity

Every application developed usually have a main entry point that opens whenever the application is executed. On games, this screen usually consists on a main menu that acts as a gateway for starting a new game, continue with a saved game or modify some settings. In contrast with this approach, smartphone's apps use this principal activity for presenting the most frequently accessed information and the most common actions to be performed. For example, in an email client app the main activity normally present the most recent emails received in the inbox and the preferred action is writing a new email. In Project Incandescence, the main activity contains a list of forebears which have been at least perceived once. The list displays a summary of the information for that forebear depending on the current state of it: just perceived, interacted with it or fully disclosed.

(a) Master - Smart-phone
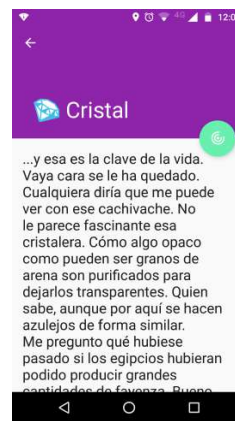
(b) Detail - Smart-phone
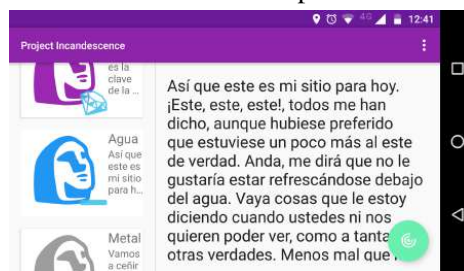
(c) Combined - Tablet

Fig. 3.3 Supports to multiple screens.

When a user click on one of the items of the collection, another view is shown with the complete information of that forebear. This master/detail navigation flow is extremly common on mobile devices due to space limitations in such screens. Some years ago, the appearance of hand-held devices with relatively big screens called tablets made necessary the creation of different user interfaces to take advantage of the increase of space. Nowadays,

the boundaries between a phone and a tablet have gotten narrower causing the appearance of the new term *phablet*. Project Incandescence sets that boundary on 480dp of width. On devices which current screen orientation surpass that mark, a side panel is attached to the main activity which allow the presentation of the list and details of forebears simultaneously. On the other case, the main activity just shows the collection of forebears and tapping on ones opens another activity for presenting the details. In both cases, the predominant action that users want to perform is travel to the disk activity described below. The floating button that allow this navigation automatically swaps permitting a direct launch of the AR activity when the player is near a meeting point. Figure 3.3 shows the difference between the main activity's appearance on different sized screens.

3.2.2   Map Activity

This activity is the first one presented by the application. It uses the Google Maps Android API [20] to display the starting point of the adventure. Thanks to the features provided by the API, the user's location is also displayed on screen. This marker stating the position of the player gets updated continuously giving a hint on the route to be followed in order to reach the starting point. A satellite map type was chosen since the target point is not placed in a street and players that are not used to the play area may orient easier with environmental hints.

3.2.3   Slideshow Activity

Regularly, Android apps display an intro screen composed of slides as a first taste to the capabilities of that application. Mimicking the same idea, Project Incandescence includes a series of slides as introduction to the game's story. This activity has been implemented with a custom subclass of *FragmentStatePagerAdapter* [60] which allows displaying and managing a number of fragments as pages with a low memory cost. Users are able to swipe left or right to advance through the slide show at their own pace. They can also tap on the buttons at the bottom bar to advance or go back in the slide set. Furthermore, the bottom bar contains a series of breadcrumbs that indicates the current progress. It must be expressed that this activity is the sole one that only supports portrait orientations of the device.

3.2.4   Disk Activity

A virtual Alberti cipher disk is implemented in this activity. This implementation contains the most challenging gesture interaction of the whole application. The outer and inner disks

are made of two separate *ImageView* [62] components. In the activity, a custom touch listener was attached to the inner disk image that responds to touch gestures. User's interactions translate to animations that seamlessly rotates the inner disk. A release of the touching finger causes the disk to stop rotating and triggers decryption of a ciphered word taking into account the current position of the disks.

The inner disk was also implemented to cast material design shadows. Floating button were discarded as an interface component to be used since images can not be stretched to fill the whole area of the button. These scarcity was overcame by setting a clipping oval outline to the *ImageView* by means of the class *ViewOutlineProvider* [74] available from API level 21. The inner disk also fills itself with the collectibles gathered so that players can associate the fragment they are decrypting with a section of the disk. This bound is created when the user taps on a word of the cipher text since automatically the corresponding collectible is displayed on the central part of the disk. This behavior was possible by attaching *ClickableSpan*s [56] to each word of the corresponding *TextView* [71].

3.2.5   Augmented Reality Activity

Apart from the proper implementation of the augmented reality module described in Section 3.4, we also included some visual and interaction components in the activity dealing with that module.

Thermal filter

A filter that imitates the output of a thermographic camera was implemented with OpenCV. It uses a look-up table [40] for applying a custom color map taking into account the different shades present in a grayscale frame. The custom color palette can be observed in Figure 3.4. It resembles a well-known palette used in thermal imaging and was generated by modifying some values of the color palette published in [8]. The 256 colors are mapped to a grayscale image being brighter pixels linked to yellowish color and darker pixels linked to purplish colors. The thermal filter is enabled with a switch at the top left part of the augmented reality screen and forebears are just observable when this filter is enabled.

Fig. 3.4 Thermal color palette.

Dialogue

Section 2.6 states that we will prevent the usage of audio in the app for a number of reasons. However, the character must communicate their speech to the players somehow. Nowadays, news presenters use a device called teleprompter to advance in the text of their script while they read. They are regularly activated by a foot pedal which makes teleprompters display the next piece of the speech. Teleprompter are the evolution of cue cards which are still used in some shows. In Project Incandescence, we tried to merge both concepts for transmitting dialogues. When a forebear is shown in the screen, a *CardView* [55] and a floating button are also displayed at the bottom area of the screen. As users press and hold on the floating button, the dialogue appears in the screen as if it was written by a typewriter. This effect pretends to emulate that the signal emanating from the subsoil is being received and processed on the spot. In order to achieve such effect, a custom *TextView* was implemented which continuously adds letters to the view with a slight delay.

Immersive Full-Screen

With Android KitKat a true full-screen mode was introduced for apps. There are several approaches [72] although in Project Incandescence sticky immersion is used. This mode of immersion hides system bars but semi-transparent bars are revealed with an inward swipe along the region where the system bars normally appear. After a few moments, the system bars automatically hide again thanks to the *SYSTEM_UI_FLAG_IMMERSIVE_STICKY* flag.

3.2.6 About Activity

This is the most simple activity of the experience. It just shows a view which states the author, version and java related libraries as if it was done from Anais perspective. There is a floating button that hides that view and displays another which states the real developer, version and native libraries.

3.2.7 Hint Alarms

The last part of each of the dialogues will appear encrypted as if the signal has weakened and the application needed more time to process it. This last phrase always contains a clearer clue indicating where the gather point is placed on the campus. As we want players to rely on their ability to reach some locations without straightforward hints, they will get decrypted and available for players after 12 minutes from the encounter with the forebear. Android alarms [66] provided the exact mechanisms to implement this behavior.

3.2.8   Encryption

One of the key processes used through all the application is the encryption of information. This process is not only indispensable in the disk activity but also on hiding information until the disclosure point is achieved. The Alberti cipher is the only one implemented and used though all the experience. Basically, a key states the inner disk rotation which allow the correct substitution of characters. Since our outer and inner disk contains the same ordered alphabet, it could be considered a simplified version of the Alberti cipher. It can be seen as a Caesar cipher which shifts are determined per each forebear gathered token providing the polyalphabetic nature to it. This simplification was performed for easing the comprehension of the mechanism to players but the general version is the one implemented.

3.2.9   Permissions Request

Starting from Android 6.0, users are capable of granting and denying permissions at run time [65]. This feature has been available for years in iOS and provides users a greater control over their data. In Android, system permissions are divided into two categories, normal and dangerous, taking into account the direct risk of users privacy. Project Incandescence uses 3 permissions: vibration, access to fine location and camera. The first one falls in the normal category whereas the other two fall in the dangerous category. The app gives support up to API level 23 which is Android 6.0 as described in Section 3.5. In other words, dangerous permissions must be requested at run time.

The user could have been asked to grant all the needed permissions when the app boots for first time but it is a better approach to ask for them when they are needed for the first time. Sometimes the requested permissions are not mandatory for the proper functioning of the application. In our case, both permissions are strictly needed for progressing in the experience. The location permission is requested in the map activity so that it can display the player's position on the map. Even if they do not enable it in the first request, they will have to enable it when they reach the starting point. The first geofence event must get triggered or it will prevent advancing in the adventure. A similar situation occurs the first time the AR activity is initiated, requesting the camera's permission. In both scenarios, a permanent snackbar is placed in the bottom of the screen requesting the permission in case the user have not granted them. Users can revoke the permissions at any time in the settings screen but it will prevent advancing in the experience for the reasons stated above.

(a) Snackbar   (b) Dialog
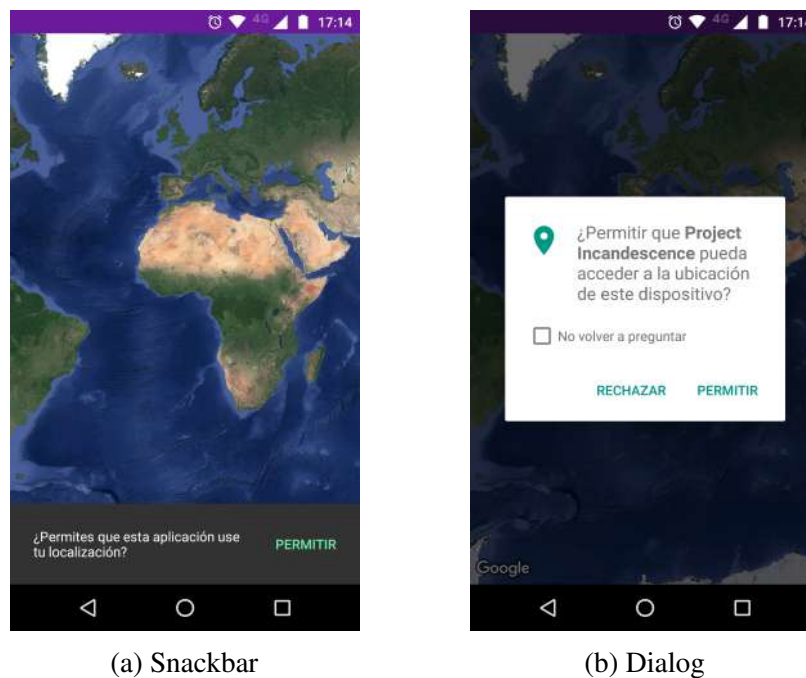
Fig. 3.5 Location permission request.

3.2.10   Model

The Android framework endorses several ways to store persistent data locally [28]. When a big amount of data is managed databases are the way to go. Android provides full support for SQLite databases which is normally accessed by a content provider. Project Incandescence uses a reduced amount of data so it was agreed to invest time in other aspects rather than in data management. That is why on the application all persistent data is stored in a class which tries to resemble a content provider.

A singleton class is instantiated, which globally holds the game's state, when the service is created. The constructor of this class is responsible for reading the forebear data from the content provider. From that point on, this instance is the one in charge of holding and modifying the forebears' state.

3.2.11   Notifications

A crucial user input in this game is the current player's location. This application does not rely on maps or other visual output regarding the current location. We want players to traverse the game world by themselves just with some clues obtained on encounters and haptic feedback to find new encounters. One of the main outcomes that Project Incandescence wants to achieve is that players can use a smartphone as a game device without constantly

sticking their eyes on the screen. It tries to direct that looks to the beauty of the real world. Anyhow, some kind of feedback needs to be provided when players are inside an area of interest. Notifications have been chosen as the way that the app have to inform users of events and progress.

There are 5 different notifications issued through a gameplay: when reached the starting point, when on a meeting point area, when a token is gathered, when a hint has been disclosed and two minutes after the disk puzzle is solved. All five share a common structure and properties. They all leave the default notification light, sound and vibration, render a emoji as a bitmap for the picture of the notification and use a common small icon to identify themselves as coming from Project Incandescence. Also all of them open the activity that is more closely related to the event and notifications are backward compatible with older versions of Android.

3.3 Geolocation

This module can be seen as the key one since is the one in charge of orienting players through the game world and at the same time is the one which triggers events that enable interaction.

3.3.1 Service

The geolocation module must always be running in a background task with no UI so an Android *service* [27] was chosen for this duty. This component is in charge of listening to user location updates and to geofences trigger events, and act accordingly. In Android, services runs on the main thread so it can become a performance pitfalls if we perform long and resource intensive tasks on it. In that case we should create threads inside the service. However, for Project Incandescence those tasks consist just on performing changes in the vibration, changing the model and issuing notifications. For that reason we do not create threads and handle all tasks in the main thread.

The service's lifespan was another aspect to take into account. Normally, tracking apps like fitness ones are in charge of collecting data continuously for example the speed, distance and route of a running session. Those functions are performed in what is called a *foreground service* [26]. This type of services are also selected for other kind of tasks like a music player but shares the same characteristic among them, users is explicitly aware of its ongoing operations. The main characteristics of these services is that they are preserved when the system is on low memory, and that it displays a sticky notification while it is running in the foreground. In other words, the user must explicitly kill the service in order for it to finish.

At first, this behavior should be considered as the right one for our application. Anyhow, there are two potential pitfalls. The first one is that notifications are a key mechanism in our game. The application is continuously issuing notification when we are in a AR available zone, when a token is gathered or when a hint is deciphered. For players, having a permanent notification may be misleading when another one is triggered. The second pitfall is the lifespan itself. If the system issues a cleanup in order to free resources, it will start from the older ones to the newer ones. Assuming that the player is still playing the game, this will not be the case so the application and service will be preserved. All in all, we have attached the lifespan of this service to the lifespan of the application. So the service will only live while the app is running on the device. If the system or player decide to kill the game, the service will also be deceased. The flag *START_NOT_STICKY* [67] is the appropriate one for achieving this behavior on a service.

The final important aspect of the service is the communication with activities. We have several mechanisms to perform such communication [16]. The communication is really sporadic between the service and other components in Project Incandescence. Since the communication can be asynchronous and we already included a *broadcast receiver* [54] in the service to listen for geofence triggers and alarms, we decided to employ this kind of communication as the sole one for both directions. Activities use this channel for requesting a vibration stop/start or a geofence placement. On the other side, the main activity is the only one listening for intents coming from the service. Those received intents inform of model changes and if the player has entered an AR area in order to refresh the UI.

3.3.2   Geofences

Google Play Services [24] provide a mechanism to create virtual fences that trigger events when a user enter, leaves or strays in a region for an amount of time [57]. This feature fits perfectly in augmented reality and geocaching applications since it emits events when the user enter an area of interest. The location APIs are really flexible allowing developers to set the circular region of interest, the duration and transition type of these geofences. Moreover they are really easy to use and that is why they were selected for the starting point, gathering points and limiting the playable area of the game.

3.3.3   Location

Despite the fact that geofences are a powerful tool for developers they have limitations. The first one is that they are inside the Google Play Services instead of forming part of Android, making it available just to those devices with them enabled. [57] also states more

limitations like the accuracy or a maximum of 100 active geofences per device user. Project Incandescence wants to guide players through the game area just by vibrations. For achieving this behavior, a mechanism that adapts to slight location changes must be provided. In order to attain this with geofences, a great number of them should have been placed to detect where the user is gracefully. Due to previously mentioned limitations, it could not be done it in such way so the continuous detection of the precise user's location was added in the application.

In Android, there are mainly two ways of adding location awareness to an application: through Google Play Services location APIs [19] or through Android framework location APIs [53]. The latest usually provides better accuracy when a valid signal in the GPS hardware is available and does not need an extra client library. Nevertheless, the first approach yields unified location updates from different providers and usually consumes less energy. Since we already use the Google Play Services APIs for geofencing and the campus have a free WiFi connection which allows more accurate location updates, we decided to use the fused location provider.

With a continuous stream of precise user locations we can determine accurately the distance from the current user's position to the closest meeting point. This information allows the precise tuning of the vibrator in order to provide players exact haptic clues on where the forebear is placed.
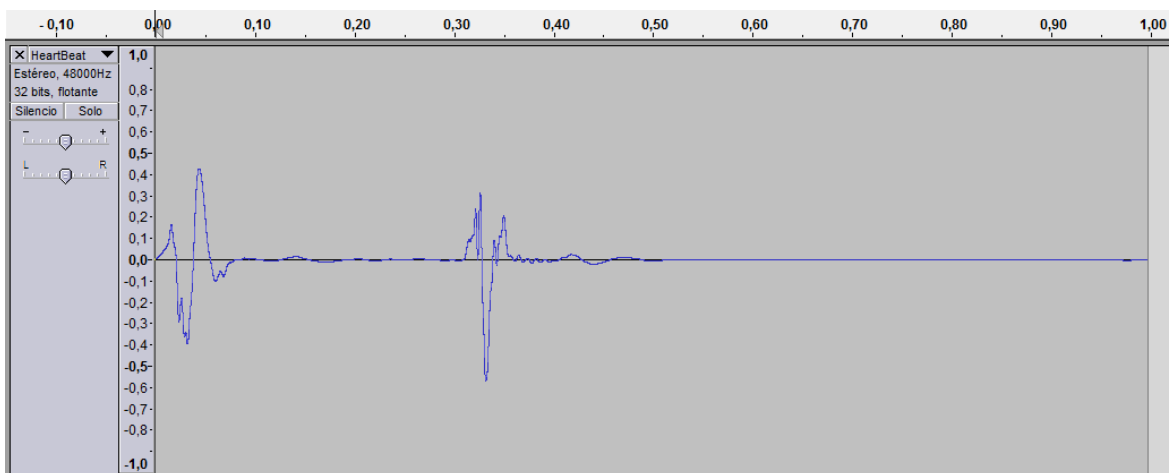
3.3.4 Haptic Feedback

Fig. 3.6 Cardiac cycle wave.

Location inputs are transformed in location outputs in order to provide haptic feedback. In a similar way on how a metal detector works, proximity information is provided but

not as sound output. As we are living beings, there are some patterns that we always tend to recognize. One of them is the sequence of cardiac cycles. The heart rate is typically expressed in beats per minute, and varies depending on the animal and its body current state. Project Incandescence tries to mimic this pattern but with vibrations rather than sounds. In Figure 3.6, we can observe a wave which represents a cardiac cycle from a sound recording. An approximation of the wave's behavior is presented in Equation 3.1. Each of the terms composing the equation corresponds to a sequence of vibrations and idles, corresponding the first one to a vibration.

$$wave = 1000ms = 200ms + 100ms + 200ms + 500ms \tag{3.1}$$

This approximation is great simulating cardiac cycles but is bonded to cycles of 1 second of duration or the ones corresponding to a 60 bpm rate. Also, a vibration pattern following the first approximation gets annoying over the minutes. That is the reason that lead to an insertion of bigger idle times on patterns with lower bpm values. Equation 3.4 represents the approximation used for creating vibration patterns in Project Incandescence. The range of rates are from a minimum of 10 bpm to a maximum of 130 bpm. Lower beats per minute corresponds to a target that is placed at a great distance while high rates correspond to a nearer target.

$$u = \frac{60}{bpm}100 \tag{3.2}$$

$$idle = (\max BPM - bpm)^2 \tag{3.3}$$

$$pattern = 2u + u + 2u + (5u + idle) \tag{3.4}$$

The vibrator API [73] included in the Android framework is quite limited. It lets issuing a vibration with a precise pattern during a fixed number of times or indefinitely. Anyway, it still does not allow the modification of such pattern once the vibration is ongoing. Continuous location changes need the modification of such pattern. The first approach that we could take is stop the current vibration and issue a new one with the new bpm. The problem with this approach is the concatenation of vibrations or wait periods between both patterns which can be misleading. In order to achieve an seamless haptic feedback, we have a runnable that issues a unique cardiac cycle with the current distance and recursively calls itself when that vibration ends. In each of the recursive calls, the pattern gets changed with the updated distance.

3.4   Augmented Reality

The following section describes the most complex and resource demanding component of the whole game. Augmented reality is a complex process that can be easily implemented on an Android app by means of some libraries like Vuforia, Wikitude or ARToolKit. Each of them have their own advantages and disadvantages but they are all high-level libraries which limit the understanding of AR in detail. For that reason, we decided to go into a lower level and decompose the whole process of overlaying virtual objects in a real-time camera stream into two separate phases.

3.4.1   Detection

The initial step of augmented reality is obtaining frames from the camera and perform a detection of the area where 3D models will get rendered. This decision is usually done by predefined signs called markers. Marker-based tracking is the most common approach for detection since once the marker is detected we can easily obtain the correct scale and pose of the camera from the image. Nevertheless, this method needs the placement of those outrageous predefined signs in the real world. The rest of detection techniques are normally called markerless even though they tend to use visual features for the detection.

The next decision taken was to determine what we were going to detect in order to overlay correctly information. In Chapter 1, we determined that horses silhouettes were going to be our "markers". However, we changed to meeting point signs for the following reasons. The first one is that these sculptures are absent in some regions of the game area so we will underuse all the space available. On the other hand, meeting point signs are scattered surrounding all the buildings in the campus. Horses are horizontal figures while character are normally vertical figures more similar to street signs. Furthermore, meeting point signs denote a point where you meet someone. These signs are predefined so they can be used as markers but can also be used with feature-based detection. As we will learn from next subsections, Project Incandescence follows a hybrid tracking approach. [79] states that hybrid tracking are systems which "combines two or more tracking methods". Our approach combines feature-based tracking with sensor tracking. It uses GPS to determine the user's position on earth in order to determine which forebear is going to be met while feature-based tracking figures out the local position of the object to be detected.

Image processing consists on a series of algorithms that apply signal processing techniques to images in order to obtain a result. Since Project Incandescence is a global project and not just an AR centered one, a library which contains computer vision algorithms was chosen as the baseline for marker detection. There are also different candidates for computer

vision libraries like FastCV or BoofCV. Hovever, OpenCV was chosen for this task since it is open source, cross-platform, mature and consolidated.

Approaches

OpenCV is a broad library containing different methods to achieve our goal. The following approaches were the most adequate ones and therefore they were the ones considered for the detection task.

1. **Template matching:** compare a template over every section of an input image.

2. **Feature detection, description and matching:** detect a points of interest or features like corners, describe them with algorithms like SURF and match these key points between the template and the analyzed image to find the desired region.

3. **Classifiers:** a machine learning based approach where a function is trained from a set of images and then used to detect objects in other images.

One of the approaches had to be selected from these possibilities. Template matching was the first one discarded since it does not behave correctly with scale variations. That will translate to a small range of positions where player are at an specific distance with respect to the object. The second approach is more scale-independent but it is limited to a particular marker. In other words, it detect a specific object but not a type of object. For example, it can detect the face of one particular person but not all human faces. This approach is the one available in libraries like Vuforia. The final classifiers approach is the most general and selected one since it does not present any of the previously mentioned limitations.

Cascade Classifiers

Normally a classifier works in the following way. First, there is a training phase composed of a set of images containing and not containing the desired object to be detected. From those images, a set of features is extracted with a threshold associated that determines if the feature is present or not in the targeted object. Features that best detects the presence of an object are retained while the rest are discarded. These finest features are called weak classifiers. By themselves, they cannot classify the image since they focus on a very specific characteristic but together they form a strong classifier. The final classifier is commonly a weighted sum of each one of the weak classifiers. Finally, features are applied to an image in order to determine if the object is present or not.

A cascade classifier is an optimization on regular classifiers. Instead of applying a number of features on each image's window, features are grouped in stages that compose a tree hierarchy. The first levels are composed of small sets of features while leaves contain larger sets. Applying separately each of the stages lets us discard rapidly those regions of the image and just focus on those which could be part of the object to be detected.

Cascade Training

The most important phase when using a cascade classifier is the adequate training of the cascade. Underfitting and overfitting can lead to the appearance of false positives and negatives while other aspects may have significant effects on the performance. The first element to be considered is the image set used in the training process. In any training process, two kind of images are required, positive and negative. Positive images refers to pictures which contains only the object to be detected. In our case, that means pictures where the meeting point sign is cropped in an arbitrary background. On the other hand, negative images, which are ones that does not contain the object to be detected, are also needed. The CBCL StreetScenes from the MIT [6] was selected as the dataset where negative images were obtained since they resemble the landscape of the game world. Pictures of the meeting point sign where obtained with a smartphone camera varying the resolution per each sample and manually cropped. With this process, 24 different images of the object to be detected were obtained. Anyhow this is not a sufficient number of images in order to achieve good results. In [45] the training set was composed of 5,000 positive images and 3,000 negative images. OpenCV contains an utility called *opencv_createsamples* [39] which applies perspective transformation for creating a large set of positive samples. Using as background the negative data set we created multiple positive images. During consecutive training stages some positive images get discarded so we need to produce more positive samples than needed. In our case, a 40% more was created by generating 207 samples per each of the 24 original images making a total of 4968 positive images.

With our images ready, the next step is to determine the feature type of the cascade. OpenCV currently only allows 3 different types of features: HOG, Haar and LBP. The detection rates of the three patterns are really similar as observed in [3]. However, our goal is to select the fastest one when detecting. That priviledge is held by LBP since it works with integers instead of floating-point numbers for computations [82]. This hypothesis is empirically tested on [1] where the author state that LBP is 10-15% faster than Haar while LBP is 10 times faster than HOG. Local binary pattern features were chosen as our feature types for the trained cascade. Regarding the classifier, OpenCV just support boosted classifiers so we stick with the default Gentle AdaBoost classifier.

(a) Negative sample

(b) Positive sample

(c) Generated positive sample

Fig. 3.7 Training samples.

The *opencv_traincascade* [39] program allows the generation of the desired cascade. Following an approximate 2:1 ratio used during training in [45], we set the number of positives as 3,548 and 1,774 as negative and LBP was set as the feature type. Some other parameters are available for tuning and they are described on Section 6.2.

Modifications in the library

Several modifications were performed in the section of OpenCV that deal with Android.

1. **Support for portrait orientation.** By default the Android wrapper of OpenCV just let developers use the camera in landscape mode. It was implemented in such way since Android devices always return the stream of frames in horizontal orientation. Frame rotation is an expensive operation so they just allowed such orientation for providing a better sense of motion in these limited devices. Project Incandescence characters are taller than wider, so the portrait orientation fits better. That is why the feature was developed and now it supports both orientations. The rotation of the frames makes the

AR module drop FPS but on modern devices the sense of motion is not completely lost.

2. **Selection of frame size.** As in the previous point, by default it picked the biggest one that fitted on the screen. Anyhow, cameras in new devices provides a great number of available frame sizes. Some of those sizes does not match the proportions of the screen so we end having a black region that limits our sensation of immersion. In order to alleviate that problem, a modification to the library was made in order to the select the biggest frame size which retains the screen's proportion.

3. **A unique JNI call.** There is an overhead when making calls to native code. That is why all JNI calls were merged into a single one which from an array of bytes makes the detection, applies the thermal filter and returns the final bitmap.

3.4.2   3D Model Display

After the area where the contextual information is determined, such information must be rendered on screen. In contrast with detection, Android devices support natively OpenGL ES [64] which allows us the creation of computer graphics. Despite the fact that this task could be implemented in either Java or C++ with similar results, we decided to do it with the latest since the detection phase is done in C++.

Everything dealing with this submodule was done from scratch except a helper class for performing math operations on matrices. This helper class, *vecmath* [49], was developed by The Android Open Source Project under the Apache License.

Environment

The first aspect we must take a decision on are camera's characteristics. Typically, we use perspective projections when making renders that look similar to the real world. That optical effect is achieved when objects are subject to foreshortening, and they appear larger the closer they are to the camera. As our vision of the real world is tied to this effect, perspective projection can be conceived as the appropriate one for the application. Notwithstanding that these seems the right choice, computing the distance from an object to a camera and adjusting the dimensions of it tend to be harder on perspective projections. In Project Incandescence, we only show a unique object in our scene with a transparent background. Using an orthogonal projection should not be visually strange for users since they do not have another object in the scene that could reveal the use of such projection method. An orthogonal projection was selected as the one for rendering the 3D character in the scene.

Orthogonal projection allows the object to be equally seen if closer or further from the camera when it is placed inside the frustum's boundaries. In other words, we must manually vary the scale to exert a sensation of depth. The detection phase yields a bounding box that determines the position and proportions of the detected object in the screen space. On the rendering phase, the model is translated and scaled in such a way that it covers the bounding box.

Shaders

Implementing shaders is necessary for displaying 3D shapes and coloring them. Specifically, a vertex shader is needed for rendering vertices of a model, and a fragment shader for rendering the face of a model with colors or textures. In OpenGL, these shaders are implemented in OpenGL Shading Language (GLSL). A key part of shading is the lightning model employed. We wanted forebears to express a sense of mystery which can be achieved by casting uncommon shadows. To achieve this effect, a low angle frontal light was placed. In our case, the type of light selected was a directional white light.

A custom shading model was implemented in GLSL. This model included ambient and diffuse lighting but not specular. The diffuse reflection is determined by means of Lambert's cosine law [42] whereas the ambient component is a plain dark color. As described in Section 2.10, we want a low poly approach when rendering our model. In other words, a uniform color must be applied to each face. By default, OpenGL sets a smooth shading that interpolate color values calculated at each vertex across corresponding faces, like in Gouraud shading. In OpenGL ES 2.0 modifying the shading technique to a flat shading is unavailable with *glShadeModel* [37]. For that reason, we had to duplicate vertices in order to store per-face normals. Figure 3.8, presents the difference between default smooth shading and custom flat shading. Moreover, every vertex's color of the model is the same on each encounter. This translates to different shades when rendered as the result of the lighting model implemented.

Model Storage

Another crucial aspect of this submodule is the way that the 3D model is represented. File formats such as obj, fbx or collada are the main ones used to store 3D information. Time requirements has not let an implementation of a file loader for any of those formats. Vertices, normals and color information of our 3D model has been stored as different inline arrays in a separate project's file.

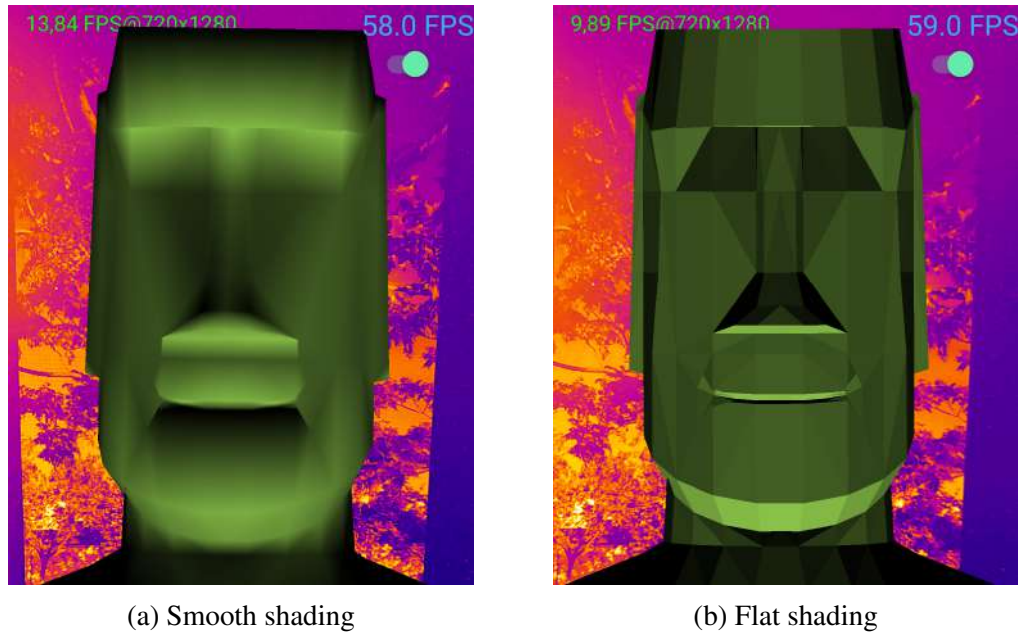(a) Smooth shading   (b) Flat shading

Fig. 3.8 Shading techniques.

3.4.3   Implementation

This subsection explains accurately the operations performed when the augmented reality is turned on. Figure 3.9 details the different operations that compose the AR module. Specifically, it details each one of the operations which are applied to a raw frame and conforming a pipeline.

Submodules operations are performed in three different threads. The main thread is in charge of capturing and feeding a worker thread with fresh frames. When a new frame arrives it gets processed in the worker thread. This procedure is done in the application's native code portion and can be split into the following operations.

1. **Memcpy:** copying the frame to the native part from a buffer.

2. **Rotation:** since the stream of frames is always obtained in a landscape orientation, a rotation is needed if the activity is in portrait orientation. A 90 degrees rotation operation can be split into a matrix transposition and a posterior flip it in the y-axis.

3. **Process:** apply computer vision techniques described in prior subsections. Tracking refers to determining regions of the image where it is probable that a targeted object is present. Detection refers to applying the trained cascade to find the object giving a higher priority to regions coming from the tracking stage.
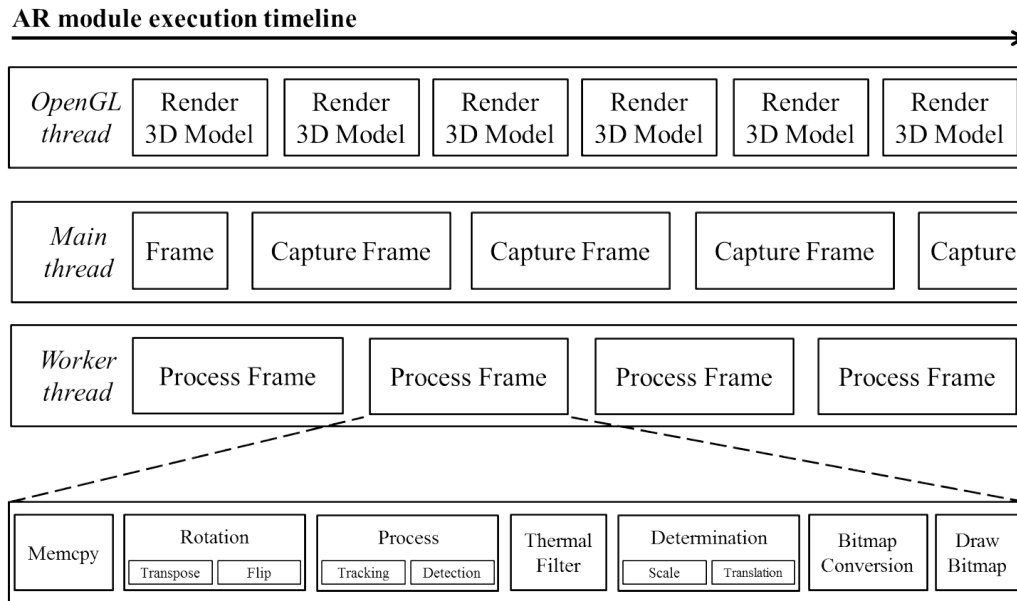
AR module execution timeline

Fig. 3.9 AR pipeline.

4. **Thermal filter:** map shades of gray to the color map described in Subsection 3.2.5 with a look-up table transform.

5. **Determination:** determine the position and size of the 3D model to be rendered.

6. **Bitmap conversion:** convert the frame to a bitmap.

7. **Draw bitmap:** draw the bitmap obtained in the previous operation in the canvas.

Simultaneously, a thread dealing with OpenGL renders the 3D model into a custom *GLSurfaceView* [61] with a transparent background. Rendering is divided in 3 phases: compute a MVP matrix with the scale and translation values determined in the worker thread, update all parameters needed by shaders and render the model on the GPU.

3.5 Build

Building is a key phase during the development of any kind of computer program but it usually gets neglected or at least does not retain the relevance it should. Compilers sometimes allow the tweaking of parameters that makes improvements in products with a minimum time investment. Android building process contain a number of phases which warrant from code shrinking to generation of builds per device architecture.

3.5.1   Gradle

Gradle [31] is an advances build toolkit which manages and automates the process of building an final product. It not only in charge of compilation but also of packaging resources in an APK file and signing this package. Project Incandescence uses Gradle experimental plugin which comes with NDK integration for building native applications [59]. Version 0.6.0 was chosen since it was the latest stable version when the development began.

3.5.2   Dependencies

The most notable dependency is determining compatible versions of the Android platform. We wanted the application to be running on at least one of each two devices. Project Incandescence is available from devices with Lollipop (API level 21) to ones with Marshmallow(API level 23), which is the 53.7% of the market [58]. Support to Android Kitkat (API level 19) was considered but it was not possible as established in Subsection 3.2.4. OpenGL ES versions are independent of the OS. That makes possible that devices running one of the latest Android versions rely on older versions of OpenGL. In order to support every possible device, the minimum OpenGL ES version was set to 2.0.

Table 3.1 specifies the libraries which the Java part of Project Incandescence depends. The selected versions were the latest ones when the development phase started. The native part of the app developed in C++ have the following library dependencies: android, log, jnigraphics and GLESv2.

Table 3.1 Java libraries used.

Library	Version
Design Support Library	23.4.0
v4 Support Library	23.4.0
v7 Appcompat library	23.4.0
v7 Cardview library	23.4.0
v7 Recyclerview library	23.4.0
v13 Support Library	23.4.0
OpenCV	3.10
Google Maps	9.0.2
Google Location and Activity Recognition	9.0.2

Regarding OpenCV, there are two possible ways of initializing the library in a custom application [38]. The recommended way is using async initialization which uses the OpenCV

Manager to access OpenCV libraries externally installed in the device. This is similar to dynamically linking the library which is automatically updated and shared among APKs. However, this approach requires that users manually install the OpenCV Manager application which they may not want to do. On the other hand, we can pack OpenCV inside the application's package. Similar to static libraries, this method increases the size of the application. Anyhow, the latest one was selected because it does not require players to install an additional application during the first encounter.
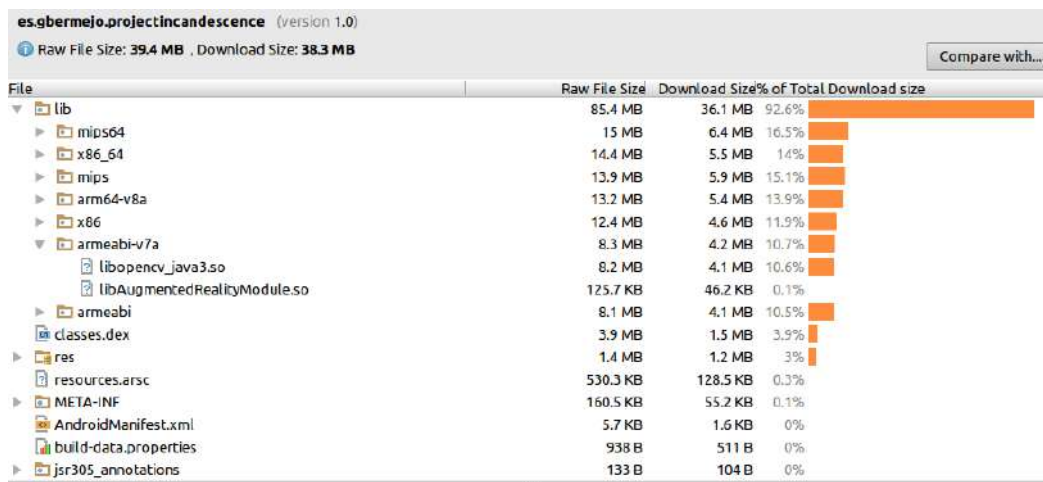
3.5.3   APK Splits

Fig. 3.10 Analysis of APK.

Figure 3.10 reflects the results obtained when using APK Analyzer [50] to determine the size of each entity inside the package. A 92.6% of the total size corresponds to the folder containing native libraries. Going deeper in the hierarchy reveals that OpenCV dynamic libraries per each architecture consumes the majority of space inside the APK file. Since the most popular architecture on smartphones is ARM version 7 or superior, a reduced APK only suitable for that devices' family was generated. A universal APK suitable for all architectures was also produced but is almost 6 times bigger. Another universal APK which is compatible with the recently released Android Nougat was created despite not having been tested.

3.5.4   ProGuard

ProGuard [43] is a Java class file shrinker, optimizer and obfuscator that can be used to improve generated APKs [68]. Developers just have to turn on the *minifyEnabled* option for the appropriate build type in the app's gradle file. When dealing with JNI, the process of

obfuscation inhibits to correct linking between java and native code functions. Also, some warnings appeared related to android to v4 Support Library and enumerations. Custom ProGuard rules were implemented as shown in Listing 3.1 in order to address these problems.

Listing 3.1 ProGuard custom rules.

```
−dontwarn android.support.v4.**

−keepclassmembers enum * {
    public static **[] values();
    public static ** valueOf(java.lang.String);
}

−keepclasseswithmembernames, includedescriptorclasses class * {
    native <methods>;
}
```

3.5.5 Signing

Android requires that all APKs are digitally signed with a certificate before they can be installed [69]. Debug builds are automatically signed with a debug certificate generated automatically by build tools but APKs signed with this type of certificate are not accepted when publishing. Release builds require the generation of a independent keystore including at least one key. The produced key enabled signing release builds of Project Incandescence although it provoked the request of an additional Google Maps API key just for release APKs.

Chapter 4

Game Art

The brief sections composing this chapter reveal the decisions made concerning visual assets' aesthetic present in Project Incandescence.

4.1 Visual Identity

4.1.1 Colors

As described in Chapter 2, material design principles will be used as the baseline for our UI design. This implies that the selection of colors presents in these interfaces needs to be compliant with that design language. Following the color palette in [18], the selected primary, secondary and accent colors are showed in Table 4.1. We include an alternate accent color since we need to discriminate between very different actions and states in some of the app's interfaces as described in subsequent sections.

Table 4.1 Project Incandescence color scheme.

Purpose	Name	Value
Primary	Purple 600	#8E24AA
Secondary dark	Purple 800	#6A1B9A
Secondary light	Purple 100	#E1BEE7
Accent	Green A200	#69F0AE
Accent alternate	Deep Orange A400	#FF3D00

4.1.2   Font

[30] states that Roboto is the standard typeface used on Android devices. Since Project Incandescence is developed exclusively for such platform, Roboto was adopted as our unique font for any kind of visual asset that needs to be created. Text colors will correspond to the default ones in Android's material themes.

4.1.3   Logo

The logo of a product should communicate the core idea in the simplest way possible. The logo is the entrance not only for a download but also for every usage so it needs to be attractive and friendly for users. Following the reference for material icon design [21] several candidates were produced. Taking as a starting point the color palette used for the thermal filter described in Section 3.4 we tried to incorporate core concepts in our logo. The evolution of the logo's design can be observed in Figure 4.1. The connection between two worlds was the fist approach to it. Then, it evolved into a letter '*i*' denoting incandescence and implicitly making a connection with the information sign. Eventually this design lead to the final logo which is more planar, includes the letter '*i*' and resembles a device which is illuminating something. At the same time, the whole design looks as planet Earth with some visible inner layers.

(a) First approach   (b) Second approach   (c) Final logo
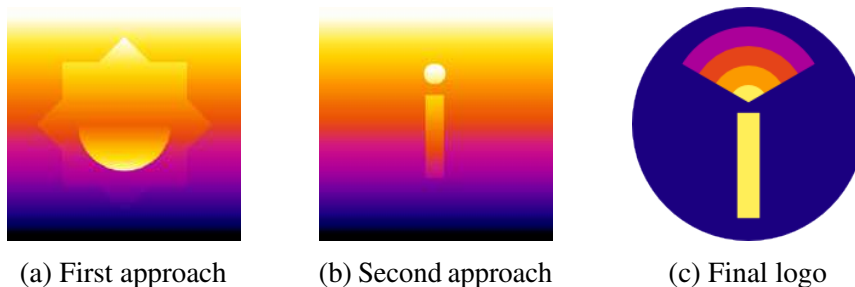
Fig. 4.1 Evolution of logo.

4.2   User Interface Design

Defining a clean and friendly layout makes users willing to interact with an interface. Mixing these idea with the principles present in [25] gave us an understanding on how to design all the interfaces present in Project Incandescence. Figure 4.2 show sketches of the principal interfaces existing in the final app. Toolbars and floating buttons are the principal ways of navigation and triggering actions.

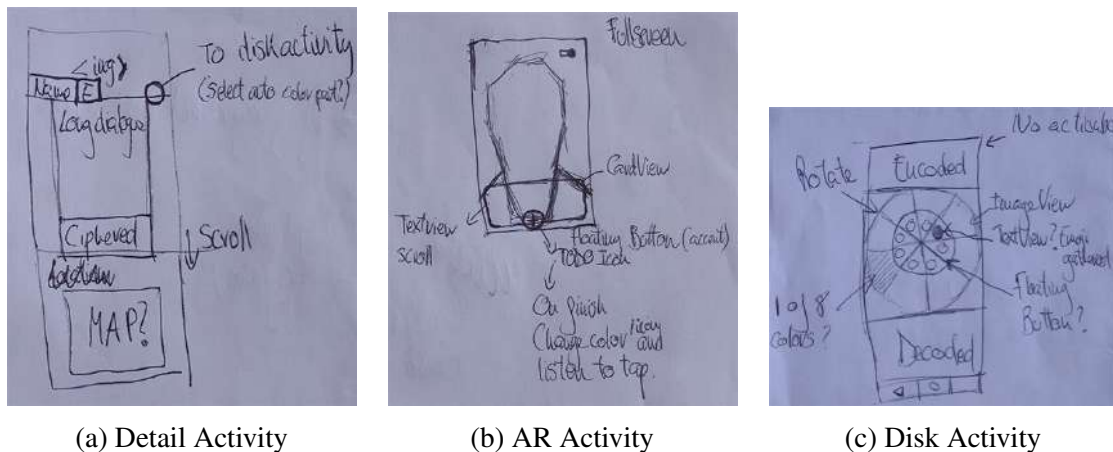(a) Detail Activity   (b) AR Activity   (c) Disk Activity

Fig. 4.2 Sketches of some activities.

Tablets are supported by default when making an Android app. However, we should take advantage of the screen size present in such devices when designing interfaces for them. Creating responsive layouts ensures that the most information can be show gracefully depending on the screen's scale. In our case, we took advantage of wider screens in order to represent more information in the main activity as explained in Subsection 3.2.1.

4.2.1 Icons

Some UI elements like floating buttons or app bar normally contains icons denoting actions that can be performed. Thanks to time constraints, icons used to represent such actions were taken from the open source material icons collection from Google [22]. Moreover, for the notification logo we also used an icon of such collection. The fan shape and the letter *'i'* are the key elements of Project Incandescence's logo, so we chose the best one from the collection that fused those concepts. Table 4.2 determines all the icons which are part of the final application.

4.3 User Experience Design

Although user experience is widely defined as "encompassing all aspects of the end-user's interaction with the company, its services, and its products" [48], we will focus not on the broad concept of it but on the part that deals with UI and usability. The fist modification done to improve UX is on the main screen. There is a floating button for accessing the cipher disk activity. This button toggles if we are on an area with a forebear in order to provide a shortcut linked to the camera activity. Instead of having two buttons which activates and deactivates,

Table 4.2 Material icons used from Google's collection.

Action	Name	Image
Cipher Disk	Wifi tethering	
AR Camera	Camera	
Enlighten	Wb incandescent	
Return	Exit to app	
Swap About	Import contacts	
Notification	Perm scan wifi	

we minimize it to a only one. This aspect is the one that lead us to pick an alternate accent color.

The second enhancement done boosting user experience is in the disk activity. Instead of letting the users type the corresponding decipher message, this process was automated. We have decreased the learning curve by doing so. At the same time, we have provided visual feedback for user when they select a snippet of the message which triggers a display of the emoji corresponding to such snippet.

Finally, the tablet design of the interface may also be underlined as an improvement in the user experience category. Allowing the visualization in the main activity of not only of the list of forebears but also the details of each one speeds up the interaction in a mid game scenario.

4.4   Emojis

Project Incandescence do not use any kind of video nor sound to express the story backing this game. Anyhow, we should encompass a mechanism of presenting such story to the player. Following the trend on most recent Android applications, we included some tutorial screens not for explaining how to use the app but for introducing the story. These screens consist on a plain color background with a top image and some underneath text. The movement between the current and the next or previous slide can be done by swiping left/right or tapping at actions in the bottom bar. This bar also include breadcrumbs to indicate the current visible slide of the whole slideshow. For the same reason as for the icons generation, already created icons were employed. In this case, an open source project has also been selected, the Noto

Color Emoji font [23]. It provides hundreds of colored icons representing a wide variety of concepts, all of them ordered and accessible in the official GitHub repository [13].

These emojis compose a font which is present in every Android device. If we take advantage of such reality, we can decrease space requirements since those images are already included in the device. For these reason, we also make use of emojis for representing other things in Project Incandescence like each of the eight elements or the gathered collectibles.

4.5 Disk Screen

Apart from the logo, the disk screen contains the rest of 2D assets created for Project Incandescence. It is made up of two vectorial images which consists on two different disk containing the Spanish alphabet including the five accented vowels as shown in Figure 4.3. Both disks letters are divided by radial lines in order to make a distinction between each others. All these lines meet up at the center, dividing this inner disk region in eight slices where emojis of gathered elements are placed.

(a) External disk   (b) Internal disk

Fig. 4.3 Image assets composing Alberti's cipher disk.
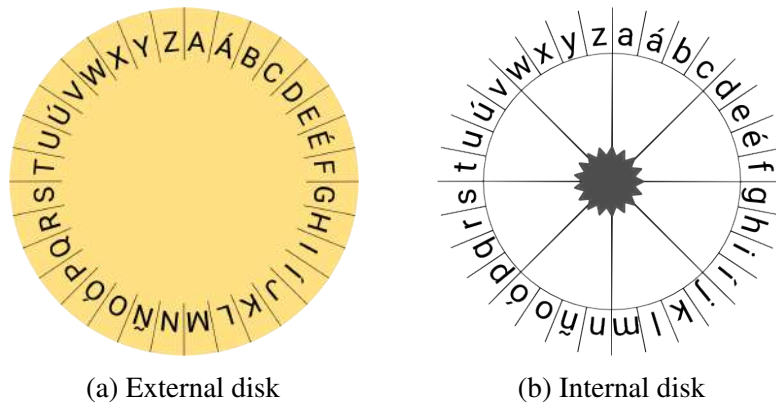
4.6 3D Character

4.6.1 Modeling

The modeling of the character was done in Autodesk 3ds Max 2016 obtaining a mesh composed of 576 triangles. Afterwards, it was optimized using the *MultiRes* modifier to a mesh of 266 triangles. This model was extremely polygonal when inserted inside the AR module so we ended up with a 468 triangles mesh optimized and fixed in Blender.

Figure 4.4 shows a quick render of the final model. The same program allowed the export of it to an obj file that was transformed to an array of points with the script available at [5].
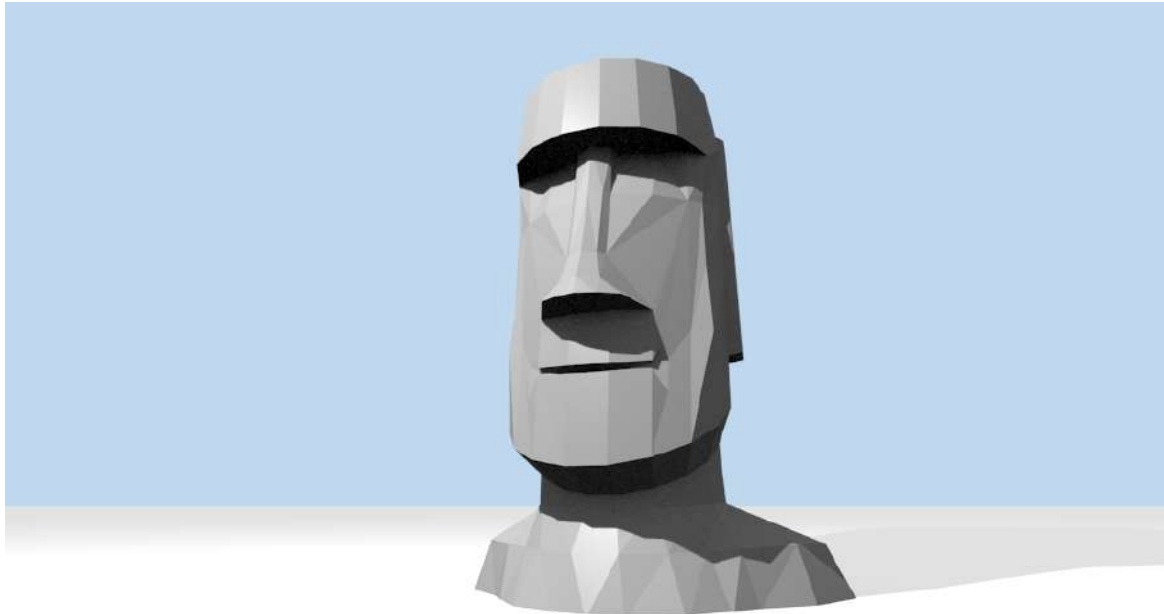
Fig. 4.4 Render of the 3D model representing a forebear.

4.6.2   Texturing

The model was not textured since when rendered on the augmented reality module it is shaded with a plain color as described in Subsection 3.4.2. The plain color changes depending on the forebear visited following the distribution indicated in Table 4.3.

Table 4.3 3D model colors on each encounter.

3D Model Colors			
ID	*Building*	*Element*	*Color*
MP_01	ESTCE TD	Crystal	#9C27B0
MP_02	ESTCE TC	Water	#2196F3
MP_03	ESTCE TI	Metal	#9E9E9E
MP_04	Library	Earth	#795548
MP_05	FCJE C1	Energy	#FFC107
MP_06	Rectorate	Wood	#8BC34A
MP_07	FCS	Fire	#FF5722
MP_08	Espaitec 2	Air	#00BCD4
SP_02	CS	Aether	#F5F5F5

Chapter 5

Results

In Chapter 1, a series of deliverables were presented. These deliverables can be considered as the tangible outputs or results from this project. This chapter consist on a presentation of the final state of each one of Project Incandescence's intended results.

5.1 Technical Proposal

The document describing the intended work for this final project which is the first chapter of this dissertation.

5.2 Game Design Document

As with the Technical Proposal, the GDD consisting on an overall of Project Incandescence's game design aspects makes part of this document as Chapter 2.

5.3 Technical Report

Subsection 1.7.2 states that the Technical Report will consist on "a technical document describing all the work done in this final project" which refers to the thesis you are currently reading.

5.4 Android Application

The Android Application is the developed experience which channeled all the efforts in order to attain the objectives targeted in Section 1.4. The version which is compatible

with all architectures and the one just suitable for devices with ARM version 7 or 8 can be downloaded from https://drive.google.com/open?id=0B4dO-M20_iy_RzJBYjNmZjk5NW8. The downloaded apk can be installed manually with the install option present in Android Debug Bridge or directly by allowing the installation of apps from unknown sources in the security related options on the smartphone.

Fig. 5.1 QR code linked to the download site.

Figures 5.2, 5.3, 5.4, 5.5, 5.6, 5.7 and 5.8 shows the final appearance of the activities described in Section 3.2 which constitute the final version of the application. In particular, Figure 5.2 reveals the final state of the Main Activity, Figure 5.3 reveals the final state of the Map Activity, Figure 5.4 reveals the final state of the Slideshow Activity, Figure 5.5 reveals the final state of the Disk Activity, Figure 5.6 reveals the final state of the Augmented Reality Activity, Figure 5.7 reveals the final state of the About Activity, and Figure 5.8 reveals the final state of the 5 different notifications issued by Project Incandescence.

5.5   Project Defense Video

Videos showing parts of the developed game can be watched at https://drive.google.com/drive/folders/0B4dO-M20_iy_QkQ3d0M5Vy0zZW8. They were recorded in a device under heavy load so the results are not as fine as in a normal gameplay. *Portrait.m4v* shows the augmented reality module on portrait mode while in *Landscape.m4v* you can sense haptic feedback resulting from location changes as well as AR in landscape mode. The music of *Portrait.m4v* corresponds to [2] and [83] to the other one. Both songs are published under Creative Commons' licenses.

5.6   Project Defense Presentation

The presentation to be used during the project defense is currently under construction but is accessible at https://goo.gl/AJMrVB.

(a) Master   (b) Detail
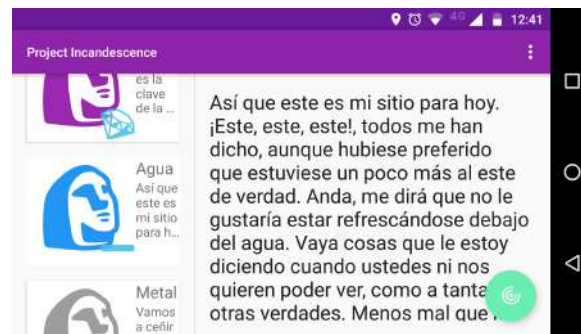
(c) Combined

Fig. 5.2 Main Activity result.
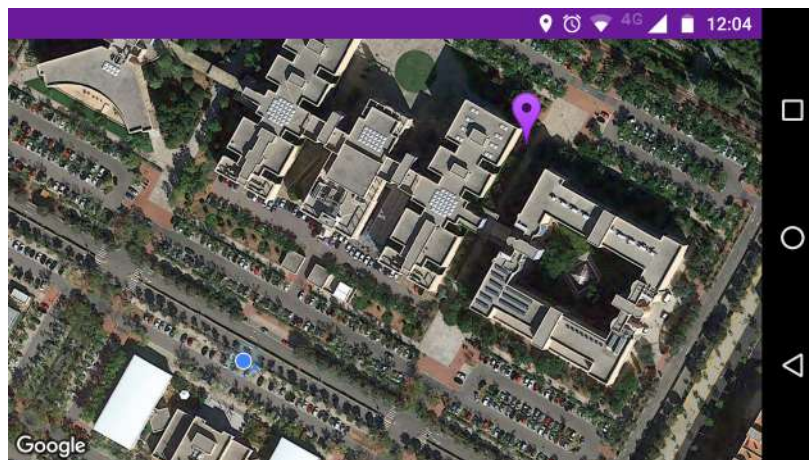
Fig. 5.3 Map Activity result.
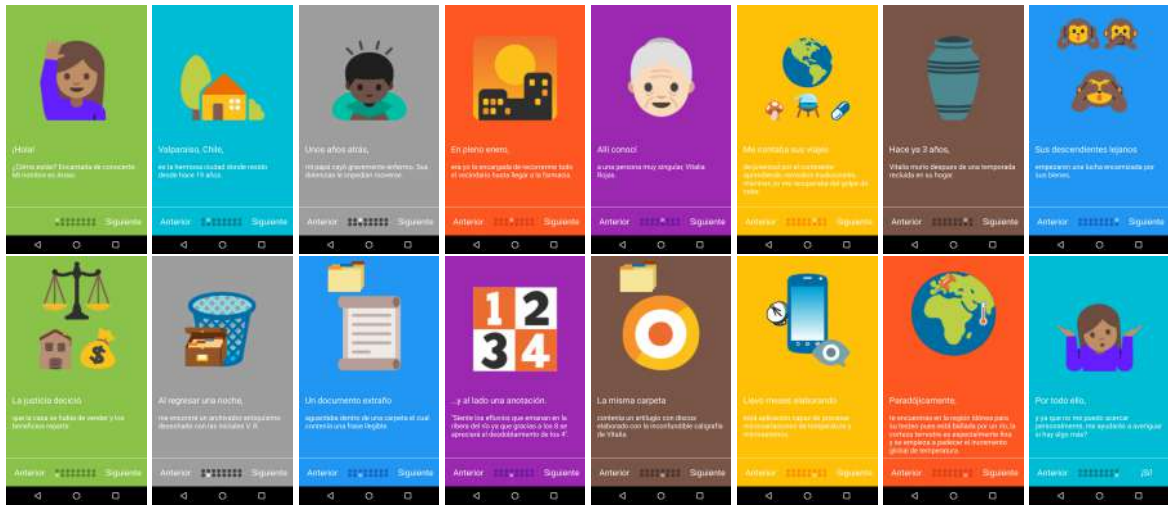
(a) Start
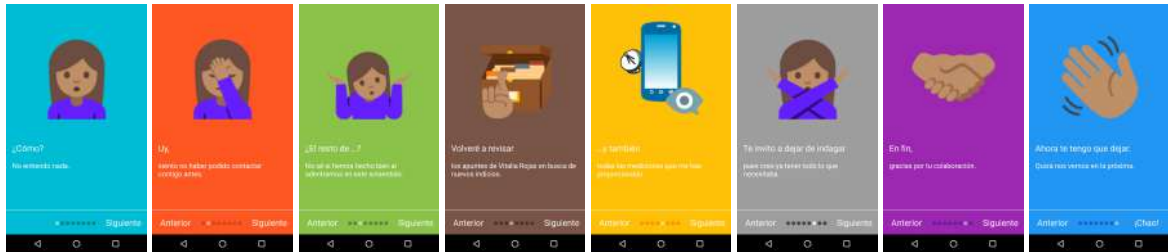
(b) End

Fig. 5.4 Slideshow Activity result.
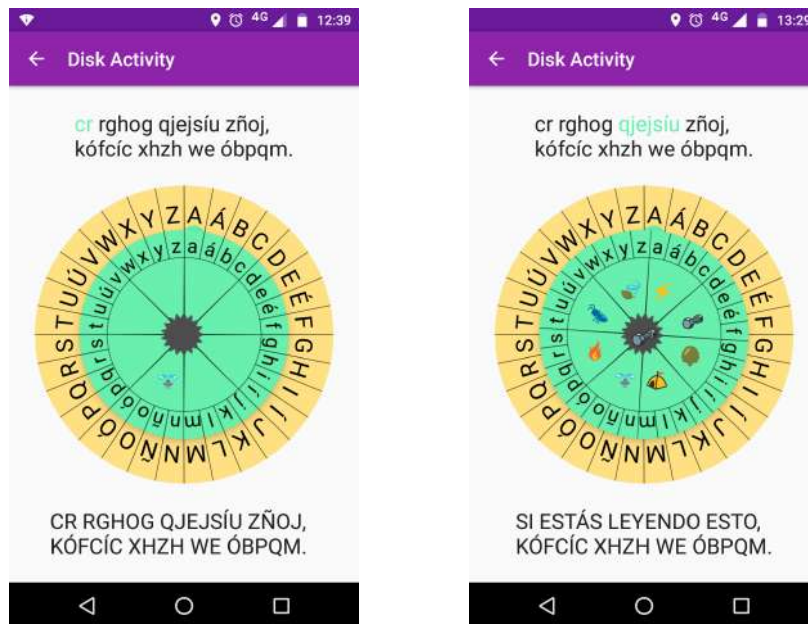
(a) Just a collectible gathered

(b) All collectibles gathered

Fig. 5.5 Disk Activity result.

(a) AR Landscape

(b) Normal

(c) AR Portrait

Fig. 5.6 Augmented Reality Activity result.

(a) Normal   (b) AR Portrait
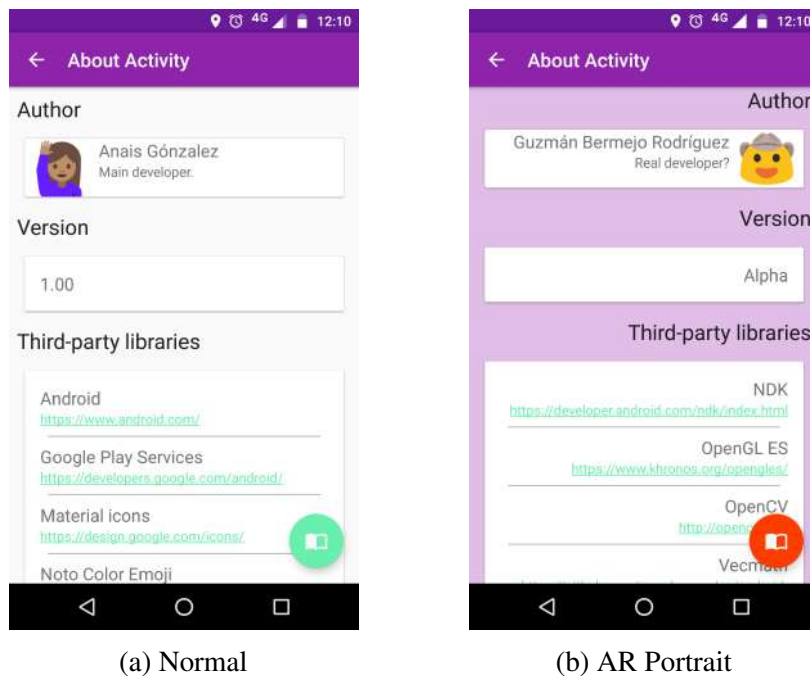
Fig. 5.7 About Activity result.

(a) Starting point   (b) Meeting point   (c) Gather point   (d) Hint disclosed   (e) Final message

Fig. 5.8 Notifications result.

Chapter 6

Testing & Evaluation

This chapter is divided in two well-defined sections. On the first one, we test the application ensuring the adequacy of Project Incandescence. On the second section, an evaluation of its performance is carried out on the app's most demanding module.

6.1   Testing

Ensuring the correctness of the system is not sufficient when testing games. Entertaining, a transmission of emotions and a development of skills are purposes inherent in games that must be examined too. The analysis of such aspects is particularly important on games which explore novice ways of attaining those objectives.

6.1.1   Verification

The verification phase guarantees that the application is free of any kind of bugs. This process is normally performed on 4 distinguishable phases: development phase, alpha, beta and release. On each phase different sets of actors test the product including developers, professional game testers and users. In our case, thanks to limitations in time and budget, the assurance on the correctness of the application was performed exclusively by the developer.

6.1.2   Validation

The process of checking that a game accomplishes its intended purposes can be labeled as validation. Developers are typically biased to some extension with their game so validation must be performed by external parties. There was not time left for performing a validation

with beta testers during the development of this project. Nevertheless, a playtest questionnaire was prepared and annexed as Appendix A.

6.2 Evaluation

In this section we will analyze the performance of the augmented reality module on a real-world scenario.

6.2.1 Hardware

The characteristics of the hardware used during the evaluation process can be found in Table 6.1.

Table 6.1 Specifications of evaluation devices.

Name	Moto G1	Moto G4 Plus
Screen Size (inches)	4.5	5.5
Android version	5.1	6.01
Memory (GB)	1	2
CPU	Qualcomm Snapdragon 400	Qualcomm Snapdragon 617
Cores	4	8
CPU Clock frequency (GHz)	1.2	1.5
GPU	Adreno 305	Adreno 405
GPU Clock frequency (MHz)	450	550
OpenGL ES version	3.0	3.1
Rear Camera Resolution (MP)	5	16

6.2.2 Marker Detection

OpenCV contains an utility called *opencv_performance* [7] to evaluate the performance of a classifier. The program takes a train set, applies the classifier and outputs the hits, misses and false positives. This program is prepared for performing benchmarks on desktop platforms but it is not really prepared for being executed on mobile ones. For that reason, all evaluations performed were done in a real-world scenario with the AR module itself. In other words, the assessment is not 100% rigorous but it have been done as strictly as possible.

Cascade Training Parameters

Subsection 3.4.1 explain how cascades were generated with the command *opencv_traincascade*. However, this program allows the tweaking of some parameters in order to improve the resulting classifier. In our case, the default value corresponding to the maximal desired false alarm rate for each stage of the classifier (0.5) was decreased and the minimal desired hit rate for each stage of the classifier was increased from the default value (0.995). The evaluation was performed on the Moto G1 device with a frame size of 1280x720 and outputting gray frames.

Table 6.2 Evaluation of different cascades.

ID	Cascade	Stages	FPS	Quality
C_1	Default	16	13	Negligible false positives
C_2	maxFA = 0.3	10	16.5	Some false positives
C_3	maxFA = 0.1	7	15	Some false negatives
C_4	minHR = 0.999	18	14.5	Little false positives
C_5	maxFA = 0.3 & minHR = 0.999	11	15.5	Some false positives
C_6	maxFA = 0.1 & minHR = 0.999	8	13	Some false negatives

Table 6.2 summarizes the different cascades generated with a variety of values. The first result it can be extracted is the absence of a correlation between the number of stages and the FPS. Common sense, make us think that traversing a deeper cascades will result in a detriment of performance since more nodes have to be evaluated. This is not always the case because each node can contain more or less weak classifiers. The comparison between C_1 and C_6 clearly demonstrates this behavior since both cascades yield 13 FPS but C_1 contains twice as much stages as C_6. Another aspect to be aware of is the appearance of false positives and false negatives. The first term relates with occasions when the classifier determines that an object have been detected on a frame which does not include the targeted object. In contrast, false negatives are related with occasions when the frames that contain the sign do not get detected. Both variants of misclassification can be remarkably confusing for players. Examining the quality of the totality of cascades, C_1 and C_4 postulate as the best ones. C_4 is slightly faster while C_1 give better results. It was considered that at this stage of Project Incandescence quality is more important than the frame rate so C_1 was the cascade chosen to be used in the augmented reality module.

Devices Comparison

In this evaluation, the final version of the augmented reality module was assesed. It compares the performance of the same classifier in all the devices described in Subsection 6.2.1.

For the Moto G1 the frame size was 1280 x 720 while for the Moto G4 Plus the frame size was 1920x1080.
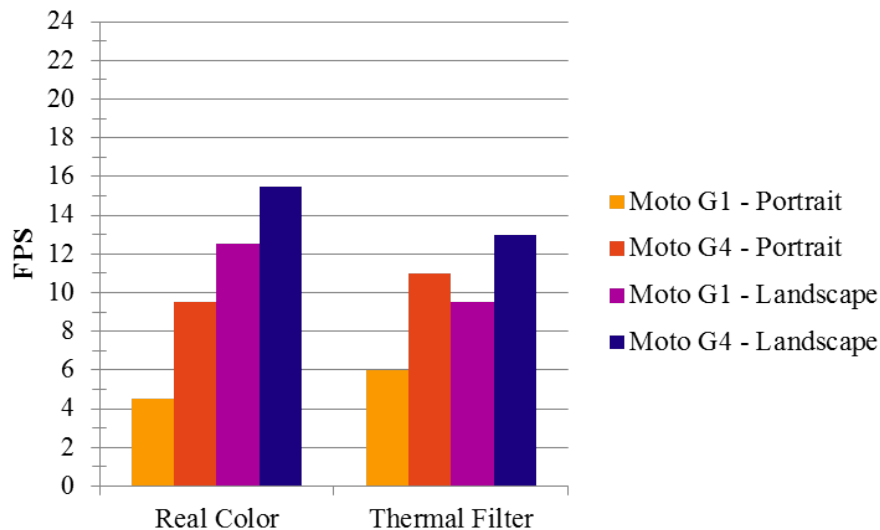
Fig. 6.1 Comparison between older and newer hardware.

As we can observe in Figure 6.1, every Moto G4 Plus test outperforms the corresponding Moto G1 one even considering that bigger frames are getting processed on the first device. Also, landscape configurations always have higher frame rates in view of saving on the costly rotation operation as underlined in Subsection 3.4.1. The interesting aspect to be spotted is that object detection and application of the thermal filter is significantly slower on landscape orientations compared to the performance gap between portrait and landscape orientations on real-colored frames. On real-colored frames the landscape processing of frames runs from 63 to 170% faster while in thermal filtered the difference is only from 18 to 58%.

6.2.3   3D Model Display

Around 60 frames per second were achieved when rendering the 3D model in both test devices. This frame rate is the one considered as providing smooth motion on apps [15] so manufacturers regularly set the screen's refresh rate and VSYNC to that frequency.

Chapter 7

Conclusions

This document presented Project Incandescence, an interactive experience which takes place in a specific region on Earth. The project not only included the design, development and proper documentation of the adventure but also had to be adjusted to a tight time schedule.

First, we presented the Technical Proposal which described precisely the work to be done throughout the project. This document included an explanation indicating how skills acquired during some degree's courses are employed during several phases. Afterwards, we designed and developed the application bonded with the Riu Sec campus. This process allowed the practice and improvement on each of the three branches a video game relies on: art, design and programming. The latest made possible the experimentation and learning of new technologies such as geolocation and augmented reality which might be useful in future projects. The acquisition of knowledge relative to such technologies was not only done from a unique path since every possible approach was carefully considered. The creation of an application for the Android platform has favored the gain of skills and experience in such ecosystem. Finally, we summarized everything related with Project Incandescence in this document.

From the paragraphs above we can extract the fulfillment of all the objectives presented in Chapter 1. In addition, we are able to extract additional outcomes. The first one of them is the feasibility of creating interactive prototypes, which use certain cutting-edge technologies, with limited resources. Game jams are events which encourage developers to try new concepts or mechanics. However, there are just a few ones that promote the adoption of new technologies. Fresh experiences which make use of recent advances can be inexpensively delivered to players and could even help increasing video games scope. Related with this idea, the toughness of some technologies might be conceived as entrance barriers. Obviously, there is an expertise to be adapted during the design and development of such projects but also there are additional costs incurred during testing and evaluation phases.

Finally, the application shows that in some cases accessibility can be provided without expense. Project Incandescence permits deaf players to fully experience it. Paying some attention during a game's design phase can bring that game to audiences with disabilities or special needs. Frequently, the creation of universal designs tend to help and provide benefits to everyone.

7.1 Future Work

There are numerous ways Project Incandescence can be improved and evolved. The following two sections provide insights on future work depending on the degree of abstraction.

7.1.1 Specific

There are numerous aspects that could be improved taking into account the current status of the application.

AR module performance

Although special care was taken during the development of the augmented reality module, the performance it yields is far from being the optimal one. Rendering camera frames directly with OpenGL ES, automatic adjustment of the frame size or a correct parallelization with a thread pool instead of just one worker thread will provide a boost in performance.

AR immersion

The sense of immersion provided currently with augmented reality is really limited. The migration from orthogonal to perspective projection, addition of animations to models and inclusion of view transformations with information gathered from sensors like gyroscopes or accelerometers can help creating a better sense of immersion. Also, a simulation of the mirage effect on "hot spots", the ones where the sign is visible, with linear filters on the colored camera view can reduce the gap between real and virtual worlds.

Older Android versions

Giving support to lower versions of Android is almost mandatory when publishing an app. In our case, Android Kitkat is still used by a 27.7% of users so we should support that version. Since the circle outline and immersive full-screen mode are the only aspects on

Project Incandescence that limit the support to older versions of the platform, we just have to implement them in another way to fix this limitation.

iOS

Migrate Project Incandescence to Apple's mobile operating system so that users with iPhones or iPads can run it.

Wearables

Allow pairing the application with wearable technology like fitness tracker or smartwatches for providing feedback. It can be easily developed for devices running Android Wear but arduous when dealing with each single fitness tracker in the market.

Improve design

Game design and art were important aspects of the project but not the principal one. Improvements on narrative, visual assets and mechanics will enhance the experience. Also, a more intuitive user experience design could help in the learning of mechanics present in first usages.

Complete testing

To this day the verification and validation of the application is limited to the developer. A group of beta testers will help spotting bugs and making improvements in the game's design.

7.1.2 General

From a higher level, we can identify how this project could evolve.

Change of device

Each year, new devices emerges while others become more autonomous. Location and augmented reality experience will be adopted by these devices at some point. As previously mentioned, wearables keep arriving which provides new sources for inputs and outputs. On the other hand, devices likes cars are starting to become autonomous, at least in determined conditions, so we could take advantage and create experiences for travels done in such conditions or for passengers.

Virtual events

Augmented reality permits the generation of virtual events. For example, a virtual art exhibition in a public space like a park or train station.

Routes

Since GPS signals are strong outdoors, creating hiking and/or sea routes perfectly fits as an evolution of Project Incandescence. However, other AR detection approaches will suit better this evolution like mixing location with cardinal direction on high seas.

Group

Playing a video game alone is great but is even greater when sharing the experience with loved ones. Creating experiences that allow cooperation or competition with other players should also be considered.

Remote

The insertion of a back-end in the system enabling remote addition and modification of adventures is crucial when targeting a global market of experiences.

Upcoming Technologies

New technologies could bring extra possibilities to augmented reality capabilities. For example, the recently released Tango platform [29] which "uses computer vision to give devices the ability to understand their position relative to the world around them". Tango uses new sensors like a motion tracking camera and a 3D depth-sensing camera to give mobile devices that type of understanding. Smartphones which include those sensor are currently rare on the market but it may become mainstream soon.

Chapter 8

Project Deviations

One of the greatest challenges faced during the development of a project is to follow the intended plan. In every project, some kind of deviation appears in constraints like time, cost or scope. The iron triangle is a model that illustrates how a deviation in one constraint affects other constraints. Project Incandescence incurred in some deviations that lead some adjustments in the project schedule and budget as described in the succeeding sections.

8.1   Project Schedule

Section 1.7 presented the intended schedule for this project estimating the duration of each task. An underestimation and overestimation of time was committed in a great set of this task due to the fact that most task were to a certain degree unfamiliar. But this wrong estimation associated with risk R_02 was not the only problem faced with regard to the time invested in the project. Risk R_01 from Section 1.8 also materialized making necessary a change in the project schedule. Even though the duration of tasks suffered a variation, the phases and task were accurately spotted in the Technical Proposal. Tables 8.1, 8.2, 8.3, 8.4 and 8.5, and Figure 8.1 are a modified version of the ones present in Subsection 1.7.1.

Table 8.1 Phase 1 - Actual Tasks Breakdown.

Documentation				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
TP	Technical Proposal	30-03-2016 to 07-04-2016	9	12
TR	Technical Report	24-08-2016 to 26-09-2016	34	85.5
PDV	Project Defense Video	23-09-2016 to 24-09-2016	1	2.5
PDP	Project Defense Presentation	27-09-2016 to 11-10-2016	4	3.5
Total				**103.5**

Table 8.2 Phase 2 - Actual Modules Breakdown.

Game Development				
ID	*Module*	*Period*	*Total Days*	*Total Hours*
MD	Marker Detection	02-05-2016 to 15-05-2016	14	28.5
3DD	3D Model Display	16-05-2016 to 02-06-2016	18	19
GF	Geofencing	03-06-2016 to 06-06-2016	4	8.5
HF	Haptic Feedback	07-06-2016 to 09-06-2016	3	5
BA	Base Application	24-07-2016 to 23-08-2016	31	68
Total				**129**

Table 8.3 Phase 3 - Actual Tasks Breakdown.

Game Design				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
GDD	GDD production	10-06-2016 to 17-07-2016	38	53.5
Total				**53.5**

Table 8.4 Phase 4 - Actual Tasks Breakdown.

Game Art				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
VI	Visual Identity	27-07-2016	1	1.5
LG	Logo	18-07-2016	1	2
UX	User Experience Design	30-07-2016	1	1
UI	User Interface Design	23-07-2016	1	2
TS	Tutorial Screens	19-07-2016 to 23-07-2016	5	10.5
3DM	Modeling a 3D character	18-09-2016	1	6.5
3DT	Texturing of the 3D model	18-09-2016	0	0
Total				**23.5**

Table 8.5 Phase 5 - Actual Tasks Breakdown.

| Game Evaluation				
ID	*Task*	*Period*	*Total Days*	*Total Hours*
IT	Integration testing	06-09-2016 to 26-09-2016	21	18
EA	Evaluation of application	21-09-2016	2	4.5
Total				**22.5**

8.2   Budget

As probably observed in Section 6.2, the application was evaluated in two smartphones instead of just one. The second device incurred the unique modification on the project's budget.

Table 8.6 Hardware Costs.

Hardware owned			
Concept	*Price per unit*	*Amortization (3 years) per month*	*Cost (5 months)*
Personal Computer (x 1)	400.00€	11.11€	55.55€
Moto G1 (x 1)	200.00€	5.55€	27.77€
Moto G4 Plus (x 1)	269.00€	7.47€	37.36€
Total			**120.68€**

This increase does not have an effect on the total cost since it is assumed as part of the 15% associated with risks.

Fig. 8.1 Project Schedule Gantt Chart with deviations.
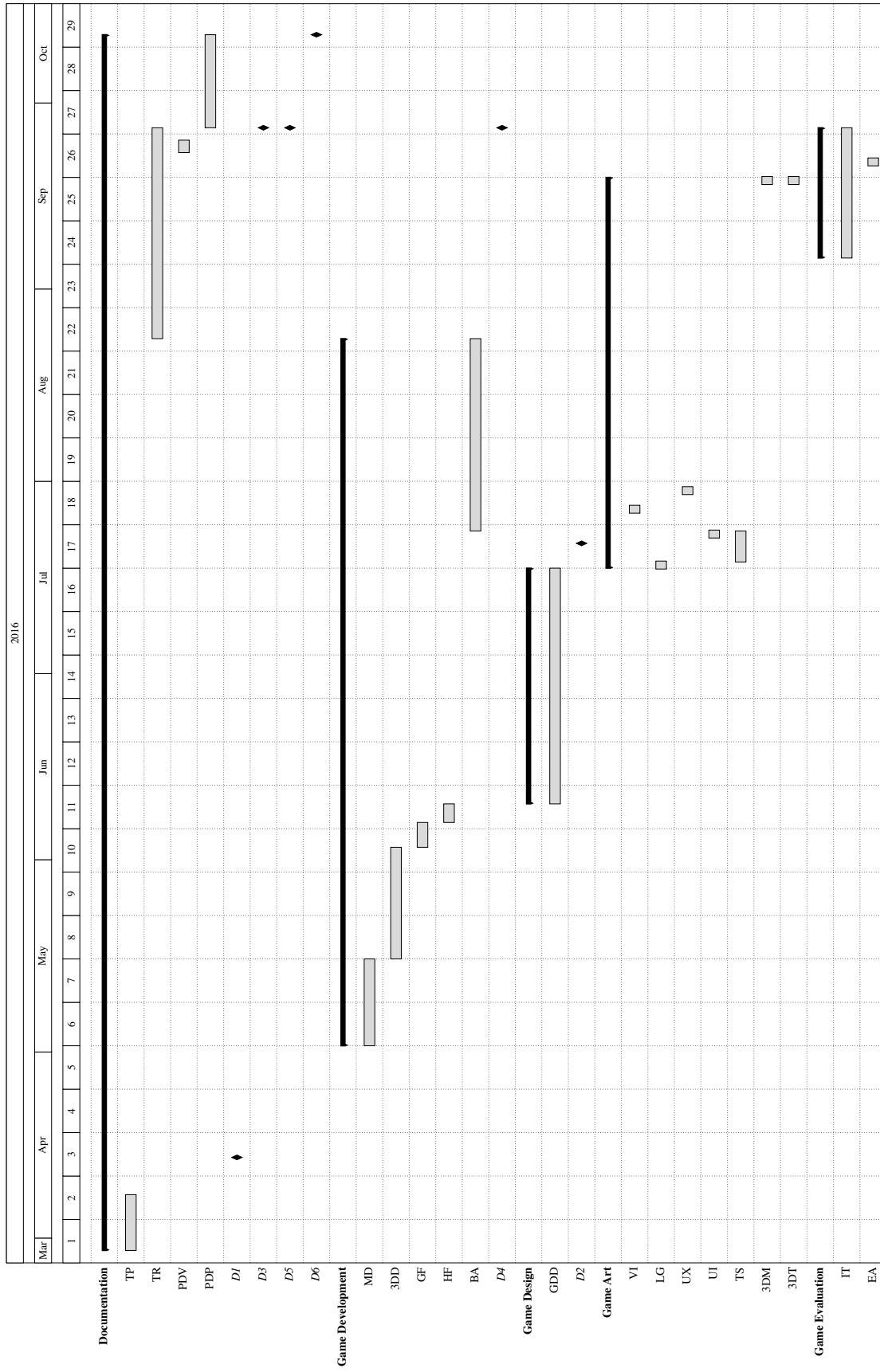
References

[1] (2016). Haar lbp and hog - experiments in opencv object detection | the robot diaries. http://robotics.hobbizine.com/opencv.html. (Online; Last accessed on 15-09-2016).

[2] 10, K. (2016). Bromas " take five" by k10.. | k 10 | free listening on soundcloud. https://soundcloud.com/k10kansas/bromas-take-five. (Online; Last accessed on 25-09-2016).

[3] Ahmad, F., Najam, A., and Ahmed, Z. (2013). Image-based face detection and recognition: "state of the art". *CoRR*, abs/1302.6379.

[4] Azuma, R. T. (1997). A survey of augmented reality. *Presence: Teleoperators and virtual environments*, 6(4):355–385.

[5] Behrens, H. (2012). obj2opengl/obj2opengl.pl at master · hbehrens/obj2opengl · github. https://github.com/HBehrens/obj2opengl/blob/master/obj2opengl.pl. (Online; Last accessed on 19-09-2016).

[6] Bileschi, S. (2007). Cbcl streetscenes database download page:. http://cbcl.mit.edu/software-datasets/streetscenes/. (Online; Last accessed on 11-09-2016).

[7] Canonical (2016). Ubuntu manpage: opencv_performance - evaluate the performance of the classifier. http://manpages.ubuntu.com/manpages/xenial/man1/opencv_performance.1.html. (Online; Last accessed on 26-09-2016).

[8] Center, I. T. (2005). Infrared training center - message board archive. http://www.infraredtraining.com/community/boards/thread/3019/. (Online; Last accessed on 02-09-2016).

[9] DAQRI (2016). About artoolkit's history and team | artoolkit.org. http://artoolkit.org/about-artoolkit. (Online; Last accessed on 08-03-2016).

[10] Djuknic, G. M. and Richton, R. E. (2001). Geolocation and assisted gps. *Computer*, 34(2):123–125.

[11] Ericsson (2016). Ericsson mobility report november 2015. http://www.ericsson.com/res/docs/2015/mobility-report/ericsson-mobility-report-nov-2015.pdf. (Online; Last accessed on 29-03-2016).

[12] Gartner (2016). Gartner says worldwide smartphone sales grew 9.7 percent in fourth quarter of 2015. http://www.gartner.com/newsroom/id/3215217. (Online; Last accessed on 08-03-2016).

[13] GitHub (2016). Github - googlei18n/noto-emoji: Noto emoji fonts. https://github.com/googlei18n/noto-emoji. (Online; Last accessed on 27-08-2016).

[14] Google (2016a). Android - history. https://android.com/intl/en_en/history/#/marshmallow. (Online; Last accessed on 08-03-2016).

[15] Google (2016b). Android performance patterns: Why 60fps? - youtube. https://www.youtube.com/watch?v=CaMTIgxCSqU. (Online; Last accessed on 08-09-2016).

[16] Google (2016c). Bound services | android developers. https://developer.android.com/guide/components/bound-services.html. (Online; Last accessed on 26-09-2016).

[17] Google (2016d). Code style for contributors | android open source project. https://source.android.com/source/code-style.html. (Online; Last accessed on 06-03-2016).

[18] Google (2016e). Color - style - material design guidelines. https://material.google.com/style/color.html. (Online; Last accessed on 27-08-2016).

[19] Google (2016f). com.google.android.gms.location | google apis for android | google developers. https://developers.google.com/android/reference/com/google/android/gms/location/package-summary. (Online; Last accessed on 26-09-2016).

[20] Google (2016g). Google maps android api | google developers. https://developers.google.com/maps/documentation/android-api/?hl=en. (Online; Last accessed on 26-09-2016).

[21] Google (2016h). Icons - style - material design guidelines. https://material.google.com/style/icons.html. (Online; Last accessed on 27-08-2016).

[22] Google (2016i). Material icons - google design. https://design.google.com/icons/. (Online; Last accessed on 27-08-2016).

[23] Google (2016j). Noto color emoji – google noto fonts. https://www.google.com/get/noto/help/emoji/. (Online; Last accessed on 27-08-2016).

[24] Google (2016k). Overview of google play services | google apis for android | google developers. https://developers.google.com/android/guides/overview#how_it_works. (Online; Last accessed on 29-03-2016).

[25] Google (2016l). Principles - layout - material design guidelines. https://material.google.com/layout/principles.html. (Online; Last accessed on 27-08-2016).

[26] Google (2016m). Running a service in the foreground | android developers. https://developer.android.com/guide/components/services.html#Foreground. (Online; Last accessed on 26-09-2016).

[27] Google (2016n). Services | android developers. https://developer.android.com/guide/components/services.html. (Online; Last accessed on 28-08-2016).

[28] Google (2016o). Storage options | android developers. https://developer.android.com/guide/topics/data/data-storage.html. (Online; Last accessed on 26-09-2016).

[29] Google (2016p). Tango. https://get.google.com/tango/. (Online; Last accessed on 21-09-2016).

[30] Google (2016q). Typography - style - material design guidelines. https://material.google.com/style/typography.html. (Online; Last accessed on 27-08-2016).

[31] Gradle (2016). Getting started with gradle for android build l gradle. https://gradle.org/getting-started-android-build/. (Online; Last accessed on 26-09-2016).

[32] Groundspeak (2016a). Geocaching > getting started with geocaching > geocache types. http://www.geocaching.com/about/cache_types.aspx. (Online; Last accessed on 08-03-2016).

[33] Groundspeak (2016b). Geocaching > getting started with geocaching > the history of geocaching. https://www.geocaching.com/about/history.aspx. (Online; Last accessed on 08-03-2016).

[34] Groundspeak (2016c). Geocaching > press > media faqs. https://www.geocaching.com/press/faq.aspx. (Online; Last accessed on 08-03-2016).

[35] Group, K. (2014). Faq - opengl.org. https://www.opengl.org/wiki/FAQ. (Online; Last accessed oon 08-03-2016).

[36] Group, K. (2016a). Opengl es - the standard for embedded accelerated 3d graphics. https://www.khronos.org/opengles/. (Online; Last accessed on 08-03-2016).

[37] Group, K. (2016b). Replacement for glshademodel on opengl es 2.0. https://forums.khronos.org/showthread.php/6949-Replacement-for-glShadeModel-on-OpenGL-ES-2-0. (Online; Last accessed on 21-09-2016).

[38] Itseez (2015a). Opencv: Android development with opencv. http://docs.opencv.org/3.1.0/d5/df8/tutorial_dev_with_OCV_on_Android.html. (Online; Last accessed on 17-09-2016).

[39] Itseez (2015b). Opencv: Cascade classifier training. http://docs.opencv.org/3.1.0/dc/d88/tutorial_traincascade.html. (Online; Last accessed on 26-09-2016).

[40] Itseez (2015c). Opencv: Operations on arrays. http://docs.opencv.org/3.1.0/d2/de8/group__core__array.html#gab55b8d062b7f5587720ede032d34156f. (Online; Last accessed on 26-09-2016).

[41] Itseez (2016). About l opencv. http://opencv.org/about.html. (Online; Last accessed on 08-03-2016).

[42] Kessenich, J., Sellers, G., and Shreiner, D. (2016). *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.5 with SPIR-V*. OpenGL. Pearson Education.

[43] Lafortune, E. (2011). Proguard. https://stuff.mit.edu/afs/sipb/project/android/sdk/android-sdk-linux/tools/proguard/docs/index.html. (Online; Last accessed on 21-09-2016).

[44] Lary, L. M. (2004). Hide and seek gps and geocaching in the classroom. *Learning & Leading with Technology*, 31(6):14–18.

[45] Lienhart, R., Kuranov, E., and Pisarevsky, V. (2003). Empirical analysis of detection cascades of boosted classifiers for rapid object detection. In *In DAGM 25th Pattern Recognition Symposium*, pages 297–304.

[46] National Coordination Office for Space-Based Positioning, N. and Timing (2013). Gps.gov: Selective availability. http://www.gps.gov/systems/gps/modernization/sa/. (Online; Last accessed on 08-03-2016).

[47] Nations, U. (2016). World population prospects - population division - united nations. http://esa.un.org/unpd/wpp/. (Online; Last accessed on 29-03-2016).

[48] Norman, J. N. . D. (2016). The definition of user experience (ux). https://www.nngroup.com/articles/definition-user-experience/. (Online; Last accessed on 27-08-2016).

[49] Project, A. O. S. (2013). android-ndk/vecmath.cpp at 75638054e96ef7a6b2f91905d3802b8b11deb574 · googlesamples/android-ndk · github. https://github.com/googlesamples/android-ndk/blob/75638054e96ef7a6b2f91905d3802b8b11deb574/teapots/common/ndk_helper/vecmath.cpp. (Online; Last accessed on 26-09-2016).

[50] Project, A. O. S. (2016a). Analyze your build with apk analyzer | android studio. https://developer.android.com/studio/build/apk-analyzer.html. Online; Last accessed on 22-09-2016).

[51] Project, A. O. S. (2016b). Android ndk | android developers. http://developer.android.com/intl/es/tools/sdk/ndk/index.html. (Online; Last accessed on 08-03-2016).

[52] Project, A. O. S. (2016c). android.location | android developers. http://developer.android.com/intl/es/reference/android/location/package-summary.html. (Online; Last accessed on 29-03-2016).

[53] Project, A. O. S. (2016d). android.location | android developers. https://developer.android.com/reference/android/location/package-summary.html. (Online; Last accessed on 26-09-2016).

[54] Project, A. O. S. (2016e). Broadcastreceiver | android developers. https://developer.android.com/reference/android/content/BroadcastReceiver.html. (Online; Last accessed on 26-09-2016).

[55] Project, A. O. S. (2016f). Cardview | android developers. https://developer.android.com/reference/android/support/v7/widget/CardView.html. (Online; Last accessed on 26-09-2016).

[56] Project, A. O. S. (2016g). Clickablespan | android developers. https://developer.android.com/reference/android/text/style/ClickableSpan.html. (Online; Last accessed on 26-09-2016).

[57] Project, A. O. S. (2016h). Creating and monitoring geofences | android developers. https://developer.android.com/training/location/geofencing.html. (Online; Last accessed on 28-08-2016).

[58] Project, A. O. S. (2016i). Dashboards | android developers. https://developer.android.com/about/dashboards/index.html#Platform. (Online; Last accessed on 17-09-2016).

[59] Project, A. O. S. (2016j). Experimental plugin user guide - android studio project site. http://tools.android.com/tech-docs/new-build-system/gradle-experimental#TOC-Ndk-Integration. (Online; Last accessed on 17-09-2016).

[60] Project, A. O. S. (2016k). Fragmentstatepageradapter | android developers. https://developer.android.com/reference/android/support/v4/app/FragmentStatePagerAdapter.html. (Online; Last accessed on 26-09-2016).

[61] Project, A. O. S. (2016l). Glsurfaceview | android developers. https://developer.android.com/reference/android/opengl/GLSurfaceView.html. (Online; Last accessed on 26-09-2016).

[62] Project, A. O. S. (2016m). Imageview | android developers. https://developer.android.com/reference/android/widget/ImageView.html. (Online; Last accessed on 26-09-2016).

[63] Project, A. O. S. (2016n). Making your app location-aware | android developers. http://developer.android.com/intl/es/training/location/index.html. (Online; Last accessed on 29-03-2016).

[64] Project, A. O. S. (2016o). Opengl es | android developers. http://developer.android.com/intl/es/guide/topics/graphics/opengl.html. (Online; Last accessed on 08-03-2016).

[65] Project, A. O. S. (2016p). Requesting permissions at run time | android developers. https://developer.android.com/training/permissions/requesting.html. (Online; Last accessed on 26-09-2016).

[66] Project, A. O. S. (2016q). Scheduling repeating alarms | android developers. https://developer.android.com/training/scheduling/alarms.html. (Online; Last accessed on 26-09-2016).

[67] Project, A. O. S. (2016r). Service start_not_sticky | android developers. https://developer.android.com/reference/android/app/Service.html#START_NOT_STICKY. (Online; Last accessed on 26-09-2016).

[68] Project, A. O. S. (2016s). Shrink your code and resources | android studio. https://developer.android.com/studio/build/shrink-code.html. (Online; Last accessed on 21-09-2016).

[69] Project, A. O. S. (2016t). Sign your app | android studio. https://developer.android.com/studio/publish/app-signing.html. (Online; Last accessed on 23-09-2016).

[70] Project, A. O. S. (2016u). Support library features | android developers. https://developer.android.com/topic/libraries/support-library/features.html. (Online; Last accessed on 26-09-2016).

[71] Project, A. O. S. (2016v). Textview | android developers. https://developer.android. com/reference/android/widget/TextView.html. (Online; Last accessed on 26-09-2016).

[72] Project, A. O. S. (2016w). Using immersive full-screen mode | android developers. https://developer.android.com/training/system-ui/immersive.html.

[73] Project, A. O. S. (2016x). Vibrator | android developers. https://developer.android.com/ reference/android/os/Vibrator.html. (Online; Last accessed on 26-09-2016).

[74] Project, A. O. S. (2016y). Viewoutlineprovider | android developers. https://developer. android.com/reference/android/view/ViewOutlineProvider.html. (Online; Last accessed on 26-09-2016).

[75] PTC (2016). Vuforia developer portal. https://developer.vuforia.com/. (Online; Last accessed on 08-03-2016).

[76] Reenskaug, T. M. H. (2010). Mvc xerox parc 1978-79. http://heim.ifi.uio.no/~trygver/ themes/mvc/mvc-index.html. (Online; Last accessed on 16-09-2016).

[77] Rogers, S. (2010). *Level Up!: The Guide to Great Video Game Design*. Wiley.

[78] Rollings, A. and Adams, E. (2003). *Andrew Rollings and Ernest Adams on Game Design*. NRG Series. New Riders.

[79] Siltanen, S. (2012). *Theory and Applications of Marker-based Augmented Reality*. VTT science.

[80] Summers, D. W. (2005). Cipher machines: From antiquity to the enigma machine. http: //csc.columbusstate.edu/summers/Research/cipher-machines.ppt. (Online; Last accessed on 17-07-2016).

[81] UJI (2016). Consulta de guies docents - vj1241 - trabajo fin de grado. https://e-ujier.uji.es/pls/www/gri_www.euji22883_html?p_curso_aca=2015&p_ asignatura_id=VJ1241&p_idioma=ES&p_titulacion=231. (Online; Last accessed on 08-03-2016).

[82] Worawat Lawanont, Pornchai Mongkolnam, C. N. M. I. (2016). A comparison between two image detection algorithms on neck angle detection and a prolonged usage classification concept. (Online; Last accessed on 15-09-2016).

[83] Zplit (2015). Tea walk by zplit | free listening on soundcloud. https://soundcloud.com/ zplit/tea-walk. (Online; Last accessed on 25-09-2016).

Appendix A

Playtest questionnaire

Project Incandescence - Playtest questionnaire

Demographics

Q1	What is your age?		Q2	What is your gender?	
Q3	How often do you play games?		Q4	Years playing video games?	

Quantitative

	Rate from 1 to 5. **(1-Strongly disagree, 3 - Neutral, 5 - Strongly agree)**	**1**	**2**	**3**	**4**	**5**
Q5	Did you enjoy the experience?					
Q6	Were you able to learn how to play quickly?					
Q7	Did you feel engaged the whole time?					
Q8	Did you feel lost at some point?					
Q9	Would you consider playing a similar adventure in another location?					
Q10	Would you recommend Project Incandescence?					

Qualitative

Q11	How long did it take the experience?	
Q12	What was the objective of the game?	
Q13	What emotions did you feel though the game?	
Q14	What did you like about the experience?	
Q15	What did you dislike about the experience?	
Q16	Was anything confusing?	
Q17	Which locations were the hardest to find?	
Q18	Did you feel tired at any time? If so, when?	
Q19	In your opinion, what needs to be improved?	
Q20	Name the game you have played that is most similar to the one you playtested.	
Q21	Any final thoughts or comments you would like to add?	