ISEL
INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Desenvolvimento de Aplicações Móveis
Mobile Application Development
DAM

Final Project - Game

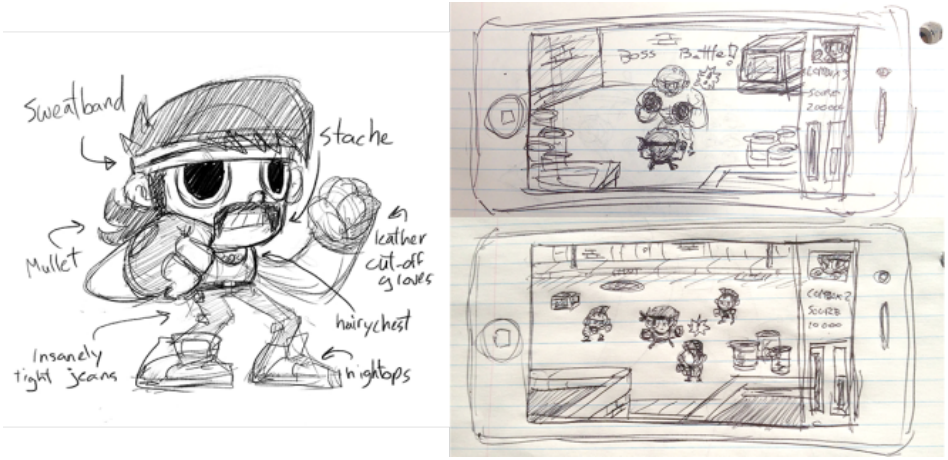
Ana Duarte Correia
ana.correia@isel.pt

Pedro Fazenda
pedro.fazenda@isel.pt

Abstract

This type of final project aims to develop an Android game, based on a Business plan and using a development process centred on the user experience.

Deadline: June 27st, 2021



Android Game

May 25st, 2021

Contents

List of Figures

1 Introduction

This kind of project aims to design, implement and evaluate a mobile game for the Android operating system.

The development process should follow the methodology centred on the player/user experience, analysed in UC slides, which have the phases of:

1. Concept;
2. Pre-Production;
3. Production; and
4. Post-Production.

The game should have a business plan since from the beginning. The game concept should be conceived to have at least one key feature to engage users to pay a certain amount (assume 1€) per month to have access to that feature. A game where users can pay 1€ per month to have access to 3 new skins each month (the skin extra feature is no longer acceptable), is an example of payable features.

The game must use the Firebase platform to **save game login and scores information** and to share it between users. Students must be proactive and self-sufficient to gather all the necessary information to use this platform through on-line information (documentation, examples and forums).

Game development should be based on the *LibGDX* game engine.

Students must deliver a fully functional marked-ready version of the application and evaluate it in terms of usability, functionality and business model.

2 Application Grading and Features

The developed game will be **graded** based on the following **guidelines**:

- Game **based on an existing game**
  - Minimum features graded up to **13**/20
- Game **based on an existing game**, but with **significant variations**
  - Minimum features graded up to **16**/20
- **Original Game**
  - Minimum features graded up to **18**/20
- **Extra features** add up to 1/20 for each feature, over minimum requirements grade

- Features are evaluated in terms of its originality, complexity and implementation quality.

The game **minimum requirements** are:

- Code from scratch - you should build all the code
- Must have user **login**
- Must have **Firebase** support to save, or share, game login and score data
- Must be a 2D game (or similar)
- Should have levels, lives and scores (depending on game genre)
- Must use sounds
- Must have at least **one animation**
- Must have start menu, help, about (with author identification and photo), settings (account information) and multilingual support for Portuguese, English and one more language (use a translator)

The game **extra features** may be:

- Be **CoViD related**
- Include multimedia content: video, more animations (of different types)
- Include 3D content
- Use location awareness or location services

An extra feature may be a group of requisites – not just one single item. Any extra feature must be approved by the teacher.

3 Development Methodology and Scheduling

The *game* should be designed and implemented focused on user experience and on a business model. Students should see information in class slides.

The *game* development should abide by the Development Methodology introduced, in the slides, and respect the following requisites and dates.

3.1 Concept phase - game concept

This phase aims to define the **game concept** and its main artefacts and study and test new technologies that will be used. For this phase it is required:

- Define the **name** and **general concept** of the *game*
- Define the experience that the *game* will promote, in a paragraph with 3 to 4 lines maximum
- Define the main characteristics of the *game* (maximum 10 characteristics)
- Define the rules of the *game*
- Define *game* main audience/users
- Identify at least one **key feature** that will engage users to pay a certain amount (1€) per month to have access to that feature
- Define *game* **layouts**
- Define *game* storyboards for the existing layouts
- Identify, justifying, the genre(s) of the game
- Identify the fun activities provided by the game
- Build the **Game Concept Document**(**GCD**) with 2 to 4 pages
- Get **teacher approval** for game concept and business model - **this is mandatory** (without teacher approval students can not proceed)
- Define the **Entity-Association diagram** (without attributes) for the *game* database to be used
- Create a **Firebase** project (https://firebase.google.com) and implement a **Firebase Android client** test application with user login using Firebase User Authentication with Email/Password and, optionally, as an extra, with another third-party authentication provider such as Google, Facebook or Twitter (worth 1/20 points)
- Make the changes to enable the **Firebase Android client** test application to save and get text and images.
- If the *game* will use **other technologies** not tested in the classes, the student should present a minimal report with an initial test (like a "Hello World" with caffeine) of each one of them
- Build a **report** with all these points

The documents and artefacts produced should be delivered until June 14^{th}, 2020.

3.2 Pre-production phase - game details and initial prototype

This phase aims to define all ***game* details** and produce an **initial functional prototype**. For this phase it is required:

- **Make user evaluation**, questioning 4 users (minimum). After a short introductory explanation, ask them to see the layouts and read the concept document and then to share their feedback, answering questions like these ones:
  - What is the purpose of the *game*?
  - Was it easy to get the *game* concept?
  - What do you think of the *game* in general?
  - Do you like it?
  - What does the player need to do to succeed?
  - When does the *game* end?
  - Do you find it challenging?
  - Would you use it?
  - Do you think there is any unsuitable element?
  - What would you like to see changed is the *game*?
  - How much are you willing to pay (monthly), to have access to payable features?
  - What are the payable features you would like to be added and you would pay with a smile?
- Improve the *game* concept and details, based on users feedback
- Design *game* **mock-ups** (layouts with detailed look) using a prototyping tool of your choice
- Define the *game* user' profiles
- Define the **full Entity-Association diagram** for the Firebase database
- Build the **Game Design Document** (GDD)
- Build a **minimal version of the *game*** (prototype) like implementing level one using only main features. Use hard-coded (simulated) data, if needed.

The documents and artefacts produced, in this phase, should be delivered until June 28^{th}, 2020.

Students should finish the first two phases as soon as possible, in order to have more time for the final phases.

3.3 Production phase - pre-market version

This phase aims to focus on the **development of the *game*** and should end with a **pre-market version**. For this phase it is required:

- Complete the *game* layouts/levels
- Build *game* auxiliary classes to handle existing data
- **Build the *game* pre-market version**, which should support all *game* main aspects
- Make **production usability evaluation** with at least 4 users (not from the concept ones), with these questions:
  - What are the actions you can do in the game?
  - Did you play one level more than once?
  - What is maximum level that you played?
  - Are the actions well mapped to command buttons?
  - Would you like to change the mapping between actions and buttons?
  - Do you think the game is fair/adequate?
  - Do you think the game is more difficult than it should?
  - Can you complete levels in time?
  - What are the levels that are found too much easy?
  - Did you find any situation that frustrated you?
  - What are your strategies to progress or win the game?
  - Which parts of the game are more fun?
  - What characters do you prefer?
  - How much are you willing to pay (monthly), to have access to payable features?
  - What are the payable features would you like to be added and would you pay with a smile?
- **Refine** *game* if needed

3.4 Post-production - market version and final report

This phase aims to **finish *game* development** and end with a **ready to market version**. For this phase it is required:

- Add *game* **final details** (multi-language, help, about, settings, ...)
- Make **post-production *gameplay* and usability evaluation**, repeating the production questionnaire with at least 6 users (not the same that already answered)
- Create the **final report** with all the documents

The final report and app market version must be delivered until July 31^{st}, 2020.

4 Final Report

The final report is mandatory and should include:

- introduction to the work
- the entire game development process
- initial layouts and storyboards
- initial mock-ups
- final screen-shots
- final full entity-association diagram of database
- final full data-base schema (no data)
- final full UML class diagram
- results obtained
- discussion of the most important issues
- conclusions

It should also include, in appendices, all the documents produced throughout the *game* development (e.g., the game concept document), which should be referenced in the report chapters as outcomes of the work, at their time.

5 Submission and Evaluation

The final evaluation of the work will be done after delivery of the final project (game marked version and report).

The final grade of the curricular unit includes all the grades of the tutorials executed during the semester and the final project (final product and report) and it is obtained in an oral discussion.

The submission of the work includes the report in digital format and the source code of the game and should be done: through the Moodle platform if possible; if not (due to the size limitations), through the OneDrive. If OneDrive is used the students must also deliver a zip, in Moodle, containing the link to the OneDrive zip file and the report (PDF) if possible.

6 Wire-frames, mock-ups and prototypes

A **wire-frame** (or wireframe) is a rough **sketch** about how a website/app/game will look like. Usually is conceived in black and white, and focused on contents, not on visualization details. It should describe the existing screens, their contents and navigation details. It should be rough, to avoid waste time in details that could reveal to be not suitable.

Figure 1: Wire-frame.

A **mock-up** (or mockup) is a high fidelity wire-frame. It focuses on the visual aspects, like colours, shapes, effects, spacing, text and form style and navigation. It should offer a complete visual guide for the prototype phase.

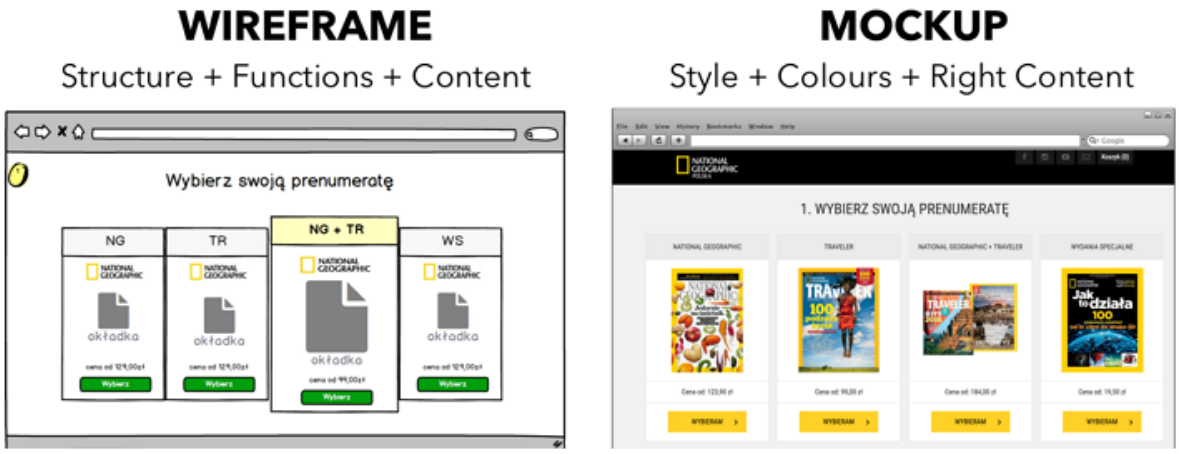


Figure 2: Wire-frame and Mock-up.

A **prototype** is a real implementation of the product, but in an early stage. It should implement, incrementally, the major use cases. Then, it should evolve to contain all of them.

Note: information and images from: www.mockplus.com, /www.fiverr.com and cover image from https://www.pinterest.ca.

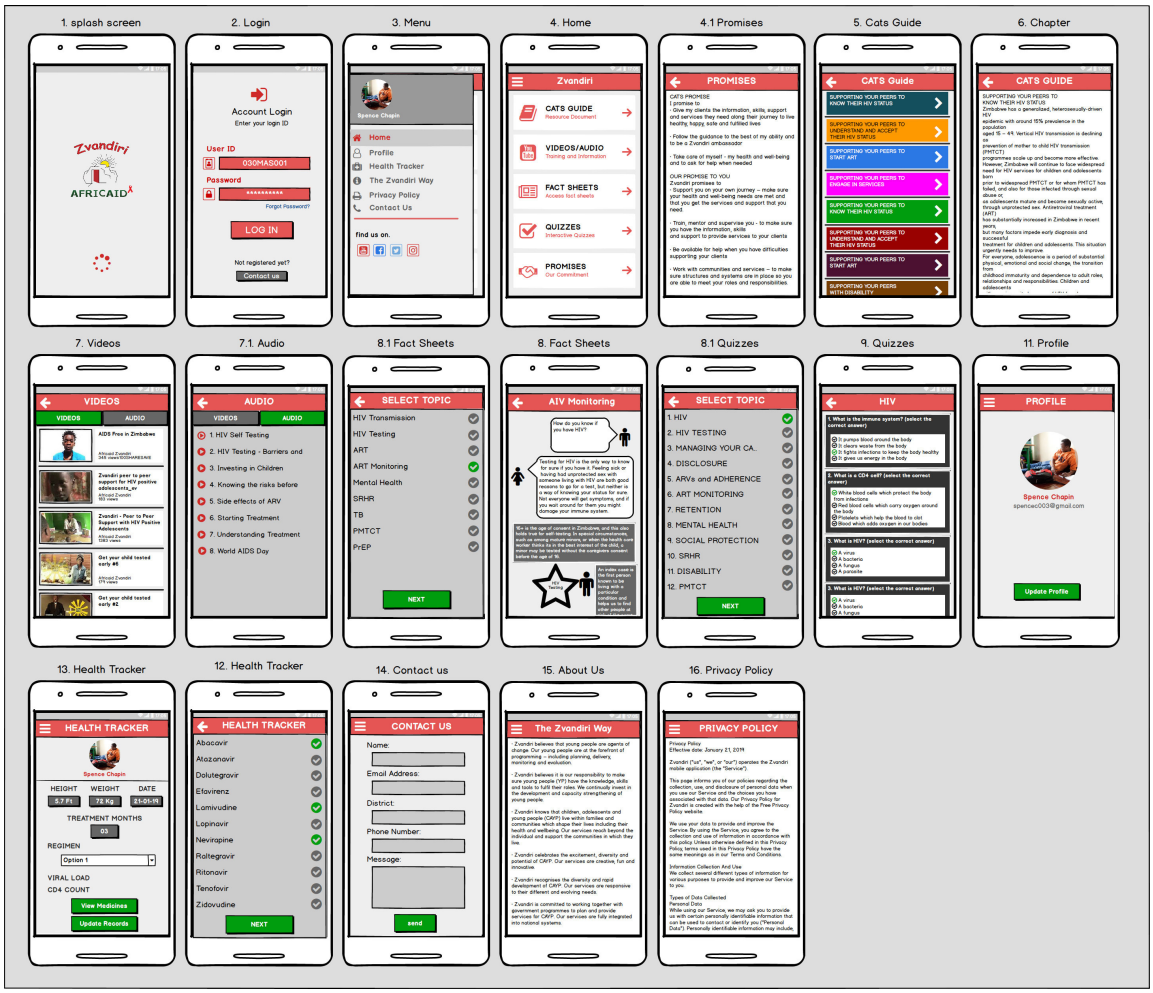


Figure 3: App full Mock-up.

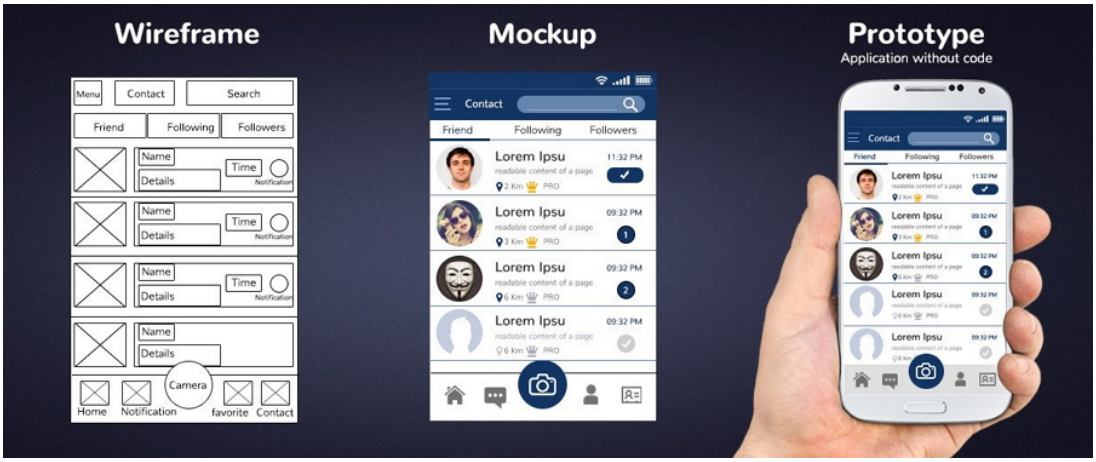


Figure 4: Wire-frame, Mock-up and Prototype.