

**Information Technology** Department

Code: xxxxxxxxxx

**SPECIALTY**

**INFORMATION SYSTEMS DEVELOPMENT**

## END OF STUDIES PROJECT

**ENTITLED**

**WOMENUP V**

**HOST ORGANIZATION**

**DIGIART LIVING LAB**

**REALIZED BY**

**AHMED YOUNES AND Ahmed Bougatf**

**ACADEMIC SUPERVISOR PROFESSIONAL SUPERVISOR**

**SAMIA CHELBI SAMIA CHELBI**

**COLLEGE YEAR 2020/2021**

# Dedication

I dedicate this work to my whole family.

My parents who have supported me all my life, I thank you with all my heart for the sacrifices and efforts that you make.

My friend bilel bellili, I thank you very much for all your encouragement and advice.

To commemorate our time together, I also dedicated this job to all my friends. Thank you for your encouragement and help.

# Thanks

First of all, I want to thank the teachers and administrators of ISET Nabeul. I would also like to thank all those who contributed to the success of my end of studies project and helped me during the writing of this report.

I would like to thank Ms. Samia Chelbi, the supervisor of the DigiArt Living Lab project in charge of the CREATEC Association, for the welcome, the time she spent with me, and the sharing of her experience with me.And pushing to go forward.

Thanks to her certainty I could complete my tasks completely.

Also, I would Like to thank my project’s partners Dr.Yassmine .. a Psychologue and Dr. Rabiaa a sociologue for their help to understand the women's behaviors and the traits of an abused woman.

This period was an opportunity to meet several people. Some directly affected my progress with interest and understanding;

others encouraged and morally supported me.

Finally, I want to thank all those who helped me and advised me in the development of this work, in particular the other interns Salime, Aziz, and Montassar.

# Table of contents

[General Introduction 1](#_TOC_250023)

[Chapter 1: Project context 2](#_TOC_250022)

[Introduction 2](#_TOC_250021)

1. [Company presentation 2](#_TOC_250020)
   1. [DigiArt Living Lab 2](#_TOC_250019)
   2. [Components of DigiArt Living Lab 2](#_TOC_250018)
2. [General presentation of the project 3](#_TOC_250017)
   1. [Frame 3](#_TOC_250016)
   2. [Problematic 3](#_TOC_250015)
   3. [Solution 3](#_TOC_250014)
3. [Analysis of existing 3](#_TOC_250013)
   1. [Description 3](#_TOC_250012)
   2. [Criticism 4](#_TOC_250011)
   3. [Proposed solution 4](#_TOC_250010)
4. [Development methodology 5](#_TOC_250009)
   1. [The four core agile values 6](#_TOC_250008)
   2. [The twelve principals of agile method 6](#_TOC_250007)
5. [Provisional schedule 7](#_TOC_250006)

[Conclusion 8](#_TOC_250005)

Chapter 2: Needs analysis and specification 9

[Introduction 9](#_TOC_250004)

1. Specification of needs 9
   1. Functional needs 9
      1. Functional needs related to the student 9
      2. Functional needs related to the teacher 10
   2. Non-functional needs 10
      1. Non-functional needs related to the student 10
      2. Non-functional needs related to the teacher 10
2. Presentation of use cases 11
   1. Presentation of the actors 11
   2. Use cases by actor 11
   3. Description of uses cases 13
      1. Description of student’s use cases 13
      2. Description of teacher’s use case 21

1.2 Use case diagrams 27

Chapter 3: Conceptual study 28

[Introduction 28](#_TOC_250003)

1. Development of the static model 28
2. Development of the dynamic model 30
   1. Construction of the mobile application sequence diagram 31
   2. Construction of the video game sequence diagram 34

[Conclusion 35](#_TOC_250002)

Chapter 4: The realization 36

Introduction 36

1. Development environment 36
   1. Hardware environment 36
   2. Software environment 36
      1. Programming languages 36
      2. Game engine: Unreal Engine 4 37
      3. Visual scripting: blueprints 39
      4. VaRest plugin 39
      5. Android studio 39
      6. Microsoft visual studio 40

1.2.6 Firebase 40

1. Architectural view of the system 41
2. Main graphical interfaces 42
   1. Video game graphical interfaces 42
   2. Mobile application graphical interfaces 46

Conclusion 50

[General conclusion 51](#_TOC_250001)

[Bibliography 52](#_TOC_250000)

**List of figures**

**Figure 1:** The existing system 4

**Figure 2:** Proposed solution 5

**Figure 3:** Screenshot of our Trello board 7

**Figure 4:** Student's use case diagram 27

**Figure 5:** Teacher's use case diagram 27

**Figure 6:** Class diagram 28

**Figure 7:** Global system sequence diagram 30

**Figure 8:** Teacher's sign up sequence diagram 31

**Figure 9:** Add student by teacher sequence diagram 32

**Figure 10:** Update student by teacher sequence diagram 33

**Figure 11:** First part of the video game sequence diagram 34

**Figure 12:** Second part of the video game sequence diagram 35

**Figure 13:** Unreal engine 4 logo 37

**Figure 14:** Unreal engine framework class relationships 37

**Figure 15:** Diagram of basic classes in unreal engine 38

**Figure 16:** VaRest plugin logo 39

**Figure 17:** Android studio logo 40

**Figure 18:** Visual studio logo 40

**Figure 19:** Firebase logo 41

**Figure 20:** Deployment diagram 41

**Figure 21:** Prime number mission graphical interface 42

**Figure 22:** Modulo mission graphical interface 43

**Figure 23:** Pgcd mission graphical interface 43

**Figure 24:** Keys graphical interface 44

**Figure 25:** Generating keys graphical interface 44

**Figure 26:** Encryption graphical interface 45

**Figure 27:** Decryption graphical interface 45

**Figure 28:** Sign up graphical interface 46

**Figure 29:** Add student graphical interface 47

**Figure 30:** Students list graphical interface 48

**Figure 31:** Update student graphical interface 49

# List of tables

**Table 1:** "Manage system" student’s use case description 13

**Table 2:** "Authenticate" student's use case description 14

**Table 3:** "Play prime numbers mission" student's use case description 15

**Table 4:** "Play modulo mission" student's use case description 16

**Table 5:** "Play PGCD mission" student's use case description 17

**Table 6:** "Generate private and public keys" student's use case description 18

**Table 7:** "Encrypt the code" student’s use case description 19

**Table 8:** “Decrypt the code" student's use case description 20

**Table 9:** "Sign up" teacher's use case description 21

**Table 10:** "Authenticate" teacher's use case description 22

**Table 11:** "Add student" teacher's use case description 23

**Table 12:** "Update missions' values" teacher's use case description 24

**Table 13:** "Update student's status" use case description 25

**Table 14:** "Access dashboard" teacher's use case description 26

# General Introduction

Video games is a revolutionary and popular form of entertainment, with gamers spending a lot of hours per day in front of their screens. Playing video games may provide learning, problem-solving skills and social benefits according to several researches, playing video games also may lead to negative effects if it’s over used.

Implementing video games in education may provide a positive effect for students who suffers from traditional form of education but only if these games are well-designed and developed under several studies and researches.

Through games students learn to exercise resilience, critical thinking, and problem-solving skills by identifying numerous solutions for problems. By introducing them to educational video games, these games can make students more interested in technology and technical skills from an early age.

As a result of traditional education, mathematics has become for some students a hard subject with a lot of useless equations and theories, these don’t understand the importance of mathematics in their future careers especially in technology and science fields.

Overall, we favored the DigiArt Living Lab as the suitable place for the graduation project, where I can learn more about video games development, gamification with Unreal Engine 4, one of the most popular game engines based in C++.

The objective of this project is to make a solution based on a serious video game and a mobile application.

This report will detail the activities that we had the opportunity to accomplish during the internship in four chapters:

* Project context.
* Needs analysis and specification.
* Conceptual study.
* Project realization.

# Chapter 1: Project context

# Introduction

The preliminary study is the first phase of any successful project; this chapter will first serve to present the host organization DALL. We then define the subject and the main objective of the project. The second part will be devoted to the definition of the methodology and the formalism adopted during the realization of this project.

# Company presentation

# DigiArt Living Lab

DigiArt Living Lab is located in the center of Nabeul, next to the “Jarrah”, it is a space for the talents of the region wishing to develop their creative spirit and wanting to produce creative projects, innovative, having a social impact and using creative and digital technologies (3D, Video Games, Virtual Reality, Augmented Reality, IOT, ...)

The project is funded by the European Union through the local cultural engagement fund in Tunisia “TFANEN”.

## Components of DigiArt Living Lab

**CO-Working:** There are teams working on innovative projects, so it is a place for work and exchange for their subscribers, whether graphic designers, coders, gamers or others who share the same passion.

**Resource Center:** D.A.L.L provides its subscribers with equipment to test their projects. The resource center is composed of a VR headset, a 3D prototyping space equipped with a computer and 3D printer, a projection space and filming on a green background.

**Training Center:** Organization of Masterclass and free trainings for the community members so they have the privilege to benefit from a 20% discount on all training provided by one of our partners.

# General presentation of the project

## Frame

The project mainly concerns video games development. To discover and learn more about the tools needed to create a video game, including its different components and aspects, to know how to manage your time to finish the task in a timely manner. Also, test various features the game engine Unreal Engine 4, discover and get acquainted with the visual scripting (blueprints) and the programming language C++, that allows us to create different features such as animations, game events, functions and graphic interfaces.

## Problematic

Today a lot of students hates math lessons thinking it’s hard, useless and can’t be used in real life. This is caused by many reasons, like classic teaching methods and pedagogic in schools. This generation is obsessed with video games and innovated technologies, so our solution will be based on a serious video game to gamify a math lesson.

## Solution

The solution is mainly based on two parts, a video game will be played by students and a mobile application will be used by teachers.

# Analysis of existing

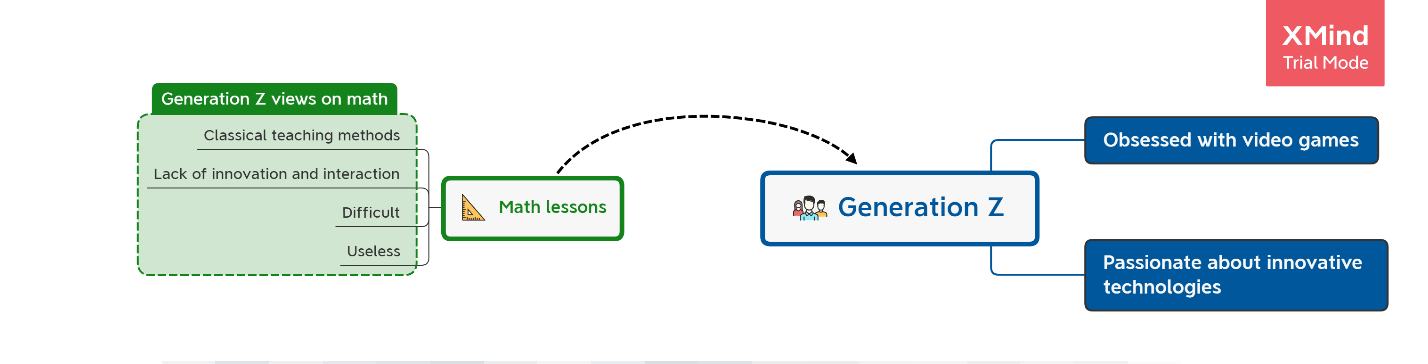
## Description

The study of the existing is an important phase to fully understand the current system. Its objective is to study and identify the shortcomings of the existing system and to propose the appropriate solutions and define the objectives to be achieved in terms of improvement.

There are two cases when we study the existing; either the product already exists, the it must be improved, or the product does not exist, it will therefore have to be created which is our case.

Started from a personal experience, many of high school students don’t know the importance of mathematics in our life -because of the traditional methods of education- which will lead to a failure and difficulties in our future careers especially in technology field.

We will analyze students’ point of view using surveys then using these data to extract a suitable gamified gameplay for certain mathematics chapters adopted by teachers and pedagogues.



## Criticism

Criticizing the system is a useful and important step. It aims to make an objective judgment in order to detect any shortcomings encountered during the study of the existing in order to offer a more reliable system.

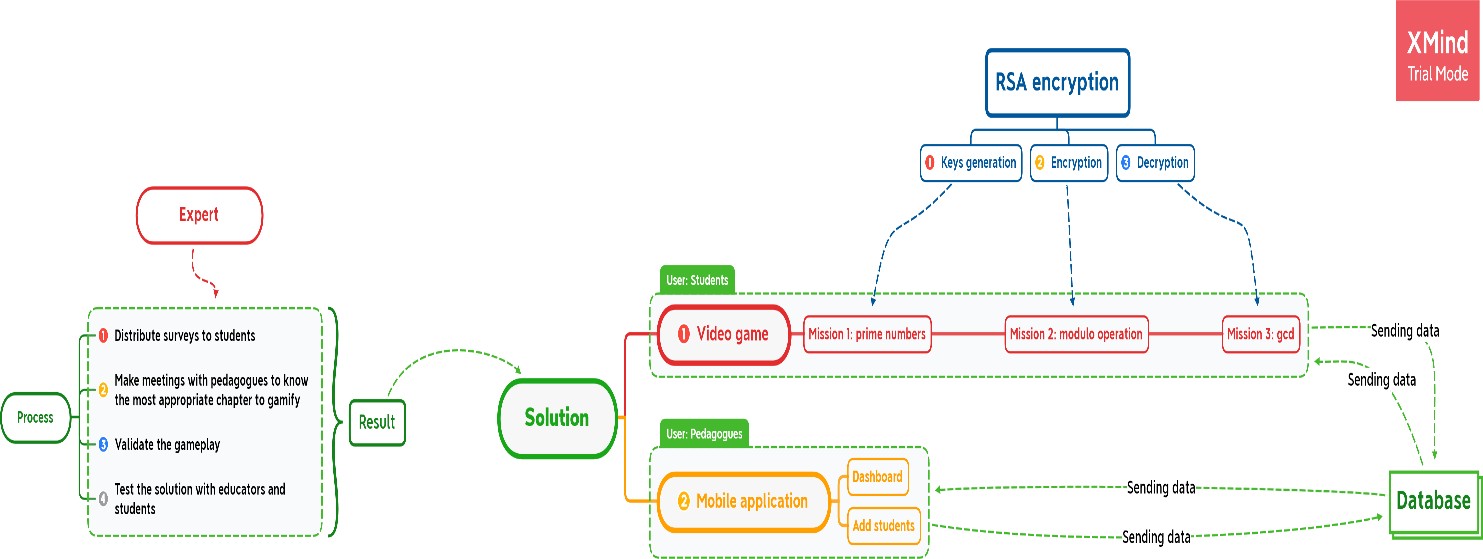
We found a critical problem in gamifying math lessons because of the misunderstood of teachers and pedagogues about gamification concept. This gap led to find and develop a suitable gameplay by ourselves.

Another problem is the current system of education is:

* + - Miss of innovation
    - Boring
    - Too much information
    - Lack of interactions
    - Less attention to real life applications

## Proposed solution

We propose a solution to avoid the critical problems listed above. To do this we want to create a system composed of a video game and a mobile application to help students learn mathematics in a way they like in the other side evaluated from their teachers.

* We will analyze the student’s behaviors using surveys.
* We will analyze top favorite games listed by the students to engage the suitable gameplay mechanics.
* Gamification of educational approach adopted by teachers.
* Implementation of a gameplay validated by the pedagogues in the form problem solving and mathematics calculations.
* Develop a mobile application for the teacher to monitor and evaluate the student in the video game.
* Dashboard for overall data extracted for the gameplay.

# Development methodology

To solve a problem and find solutions to manage any type of task, each person must follow a process to have an effective and well-structured result according to adopted methods.

This is why we must choose the best and most optimal solutions to have recourse to a powerful methodology which makes it possible to manage a life cycle of a project. There are several methods, we are interested in agile method. This method involves reiterating short cycles over time by dividing the project into multiple mini-projects and prioritizing them as needed.

Also, we used UML, short for Unified Modeling Language which is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems.

## The four core agile values

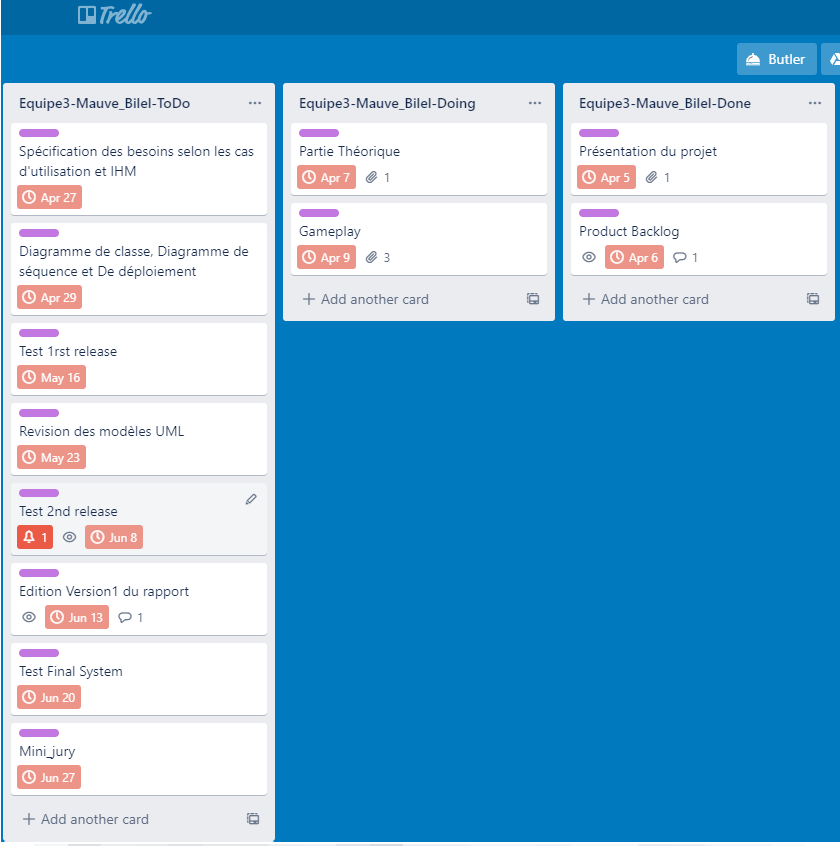
* + - Interaction between actors rather than processes and tools.
    - An operational product rather than a plethora of documentation.
    - Collaboration with the client rather than contract negotiation.
    - Responsiveness to change rather than following a plan.

## The twelve principals of agile method

* + - Satisfy the customer by delivering useful software early and regularly that offers real added value.
    - Accept change even in late development.
    - Frequently deliver a working app.
    - Collaborate daily between clients and developers.
    - Build the project around motivated people by providing them with environment and support and by trusting them.
    - Communicate through face-to-face conversations.
    - Measure progress with working software.
    - Maintain a sustainable pace of work.
    - Seek technical excellence and quality of design.
    - Let the team organize themselves.
    - Look for simplicity.
    - At regular intervals, think about ways to become more efficient.

# Provisional schedule

We used trello as our planning platform, it’s used to organize the work of a whole team or just to want to boost your own productivity.



# Conclusion

In this first chapter, we started by presenting our host organization DALL. Then, we described the context of the project, then we set out the problematic of our project, while providing the criticisms and possible solutions. Finally, we supported the choice of the working methodology used Agile and UML, thereafter we will naturally go towards planning and architecture.

# Chapter 2: Needs analysis and specification Introduction

This chapter presents the actors of our project as well as the analysis of the different needs of each.

# Specification of needs

Our project is used to provide students of generation z an educational video game controlled by a teacher from a mobile application in real time to help this generation understand the importance of math lessons in their future careers.

The video game must be fun to play with beautiful level and sound designs, and the mobile application must be easy to use with simple user interface and user experience.

## Functional needs

We will define here the actions that an actor can do using the application.

### Functional needs related to the student

The video game is available to the student after authentication by his name and his unique code, both provided by his teacher.

The video game will mainly offer these missions:

* Prime mission to unlock keys generation skill.
* Modulo mission to unlock encryption skill.
* PGCD mission to unlock decryption skill.
* Switching between two characters.
* Encrypt, send and decrypt specific code to unlock the case.
* Free exploration.

### Functional needs related to the teacher

The mobile application is available to the teacher after authentication by his email and password after he got whitelisted by the administration.

The mobile application will mainly offer these features:

* Generate unique codes for his students.
* Add student to give him access to the video game.
* Enable and disable specific student’s accessibility to the video game.
* Change missions’ numbers and values for a specific student.
* Access a dashboard for student’s performance in the video game.

## Non-functional needs

We will determine the set of implementation constraints to be respected to guarantee the proper functioning of both applications.

### Non-functional needs related to the student

* Performance: the video game should be at least 60 frames per seconds.
* Missions fun to play.
* Simple user interface.
* Beautiful level design and graphics.
* Good sound design.
* Instructions available.
* Interactive environment.
* Good storyline.

### Non-functional needs related to the teacher

* Performance: the mobile application must be designed to have a minimum response time.
* Reliability: the mobile application must have good quality of content as well as good adaptability to different screen sizes of mobile devices, on the other hand it Is necessary to ensure the correct operation without errors.
* Easy to use with simple user experience and user interface.
* Security: the mobile application should be secured to secure teacher’s data.

# Presentation of use cases

The use case diagram represents the actions performed by the system to obtain a result that meets the needs of a particular actor.

## Presentation of the actors

* Student: main actor of the video game application.
* Teacher: main actor of the mobile application.

## Use cases by actor

* Student will play with two different characters:
  + Authenticate using name and unique code provided by the teacher.
  + Achieve RSA encryption algorithm skill in three missions:

✔ Keys generation skill from prime numbers mission as first character.

✔ Encryption skill from modulo mission as second character.

✔ Decryption skill from PGCD mission as first character.

* + Use RSA encryption gained skill to encrypt, send and decrypt specific code to unlock the case in three steps:

✔ Generate public and private keys as first character.

✔ Send generated public key as first character to the second character.

✔ Encrypt the code using public key as second character.

✔ Send the crypted code as second character to the first character.

✔ Decrypt the code as first character.

* Teacher:
  + Sign up using whitelisted email.
  + Authenticate using email and password.
  + Add and generate unique codes for each student.
  + Change numbers and values of missions for each student.
  + Enable and disable student’s accessibility to the video game.

Access for a dashboard used to show students’ performance and play time in-game.

## Description of uses cases

### Description of student’s use cases

**Table 1:** "Manage system" student’s use case description

|  |
| --- |
| **Identification** |
| **Title:** Manage system.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * Video game is installed. |
| **Nominal scenario:**   1. The student starts the video game. 2. The student chooses start option from main menu. |
| **Alternative scenario:**   * The student chooses trailer option. * The student chooses quit option |
| **Post-condition:**   * The student is redirected to authentication interface. |

**Table 2:** "Authenticate" student's use case description

|  |
| --- |
| **Identification** |
| **Title:** Authenticate.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The student has a name and unique code provided by the teacher. * The student status is enabled by the teacher. * Game status is enabled by the administrator. |
| **Nominal scenario:**   1. The student enters his name and code. 2. The system checks the entered data. |
| **Alternative scenario:**   * Empty fields. * Invalid name or code. * Incorrect name or code. * Internet connection or database server is unavailable. * The student status is disabled by the teacher. * Game status is disabled by the administrator. |
| **Post-condition:**   * The student is authenticated. |

**Table 3:** "Play prime numbers mission" student's use case description

|  |
| --- |
| **Identification** |
| **Title:** Play prime numbers mission.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The student is authenticated. |
| **Nominal scenario:**   1. Search for a library. 2. Open locked door by aligning all keys in prime numbers. 3. Search and find for the key generation paper before times out. 4. Send mission data to database. |
| **Alternative scenario:**   * Keys are not aligned with prime numbers. * Can’t find the key generation paper. * Times out. * Internet connection or database server is unavailable. |
| **Post-condition:**   * Key generation skill is gained. |

**Table 4:** "Play modulo mission" student's use case description

|  |
| --- |
| **Identification** |
| **Title:** Play modulo mission.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The student is authenticated. * Prime numbers mission is completed |
| **Nominal scenario:**   1. Switch to the second character. 2. Enters an abandoned house. 3. Push specific number of cubes placed on walls to access hidden statue. 4. Take the remainder cubes. 5. Put the taken cubes in the statue to get the encryption paper. 6. Send mission data to database. |
| **Alternative scenario:**   * Pushed wrong number of cubes. * Didn’t take the reminder cubes. * Didn’t put the remainder cubes in the statue. * Internet connection or database server is unavailable. |
| **Post-condition:**   * Encryption skill is gained. |

|  |
| --- |
| **Identification** |
| **Title:** Play PGCD mission as first character.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The student is authenticated. * Prime numbers mission is completed. * Modulo mission is completed. |
| **Nominal scenario:**   1. Switch back to the first character. 2. Enters the forbidden room. 3. Generate the prime factors of two numbers. 4. Check the common prime factors of the both numbers. 5. Use the calculated PGCD value to pick up the correct key. 6. Unlock the case using the key and get the decryption paper. 7. Send mission data to database. |
| **Alternative scenario:**   * Checked common prime factors are wrong. * Taken key is wrong. * Times out. * Internet connection or database server is unavailable. |
| **Post-condition:**   * Decryption skill is gained. * RSA algorithm skill is gained. |

**Table 5:** "Play PGCD mission" student's use case description

|  |
| --- |
| **Identification** |
| **Title:** Generate private and public keys as first character.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * The student is authenticated. * All missions are completed. * RSA encryption algorithm skill is gained. |
| **Nominal scenario:**   1. Using key generation skill, generate public and private keys. 2. Send the public key to the second character. |
| **Post-condition:**   * Public and private keys generated. * Public key is sent to the second character. |

**Table 6:** "Generate private and public keys" student's use case description

|  |
| --- |
| **Identification** |
| **Title:** Encrypt the code as second character and send it to the first character.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * The student is authenticated. * All missions are completed. * RSA encryption algorithm skill is gained. * Generated public key sent from the first character. |
| **Nominal scenario:**   1. Switch to second character. 2. Using encryption skill and the received public key, encrypt the code. 3. Send the crypted code to the first character. |
| **Post-condition:**   * Code crypted. * Crypted code sent to the first character. |

**Table 7:** "Encrypt the code" student’s use case description

**Table 8:** “Decrypt the code" student's use case description

|  |
| --- |
| **Identification** |
| **Title:** Decrypt the code as first character.  **Actor:** Student |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The student is authenticated. * All missions are completed. * Crypted code sent from the second character. |
| **Nominal scenario:**   1. Switch back to the first character. 2. Using decryption skill and the received crypted code, decrypt the code. 3. Unlock the case using the decrypted code. 4. Send student data to database |
| **Alternative scenario:**   * Internet connection or database server is unavailable. |
| **Post-condition:**   * Case opened. * Game finished. |

### Description of teacher’s use case

|  |
| --- |
| **Identification** |
| **Title:** Sign up.  **Actor:** Teacher |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The teacher’s email is whitelisted by the administrator. |
| **Nominal scenario:**   1. The teacher enters his email. 2. The system checks if the email is whitelisted or not. 3. The teacher enters his password and information. 4. The system checks the entered data. |
| **Alternative scenario:**   * Empty fields. * Invalid email or information. * Email is not whitelisted. * Internet connection or database server is unavailable. |
| **Post-condition:**   * The teacher is signed up. |

**Table 9:** "Sign up" teacher's use case description

**Table 10:** "Authenticate" teacher's use case description

|  |
| --- |
| **Identification** |
| **Title:** Authenticate.  **Actor:** Teacher |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The teacher status is enabled by the administrator. |
| **Nominal scenario:**   1. The teacher enters his email and password. 2. The system checks the entered data. |
| **Alternative scenario:**   * Empty fields. * Invalid email or password. * Incorrect email or password. * Internet connection or database server is unavailable. * The teacher status is disabled by the administrator. |
| **Post-condition:**   * The teacher is authenticated. |

|  |
| --- |
| **Identification** |
| **Title:** Add student.  **Actor:** Teacher |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The teacher status is enabled by the administrator. * The teacher is authenticated. |
| **Nominal scenario:**   1. The teacher enters student name. 2. The teacher generates random and unique code for the student. 3. The teacher changes the default values of mission related to the student. 4. The system checks the entered data. |
| **Alternative scenario:**   * Empty fields. * Invalid name or missions’ values. * Incorrect email or password. * Internet connection or database server is unavailable. * The teacher status is disabled by the administrator. |
| **Post-condition:**   * Student is added. |

**Table 11:** "Add student" teacher's use case description

|  |
| --- |
| **Identification** |
| **Title:** Update missions’ values.  **Actor:** Teacher |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The teacher status is enabled by the administrator. * The teacher is authenticated. * Student is added |
| **Nominal scenario:**   1. The teacher access to students list. 2. The teacher selects specific student. 3. The teacher updates mission’ values related to the student. 4. The system checks the entered data. |
| **Alternative scenario:**   * Empty fields. * Invalid missions’ values. * Empty student list. * Internet connection or database server is unavailable. * The teacher status is disabled by the administrator. |
| **Post-condition:**   * Mission’ values are updated. |

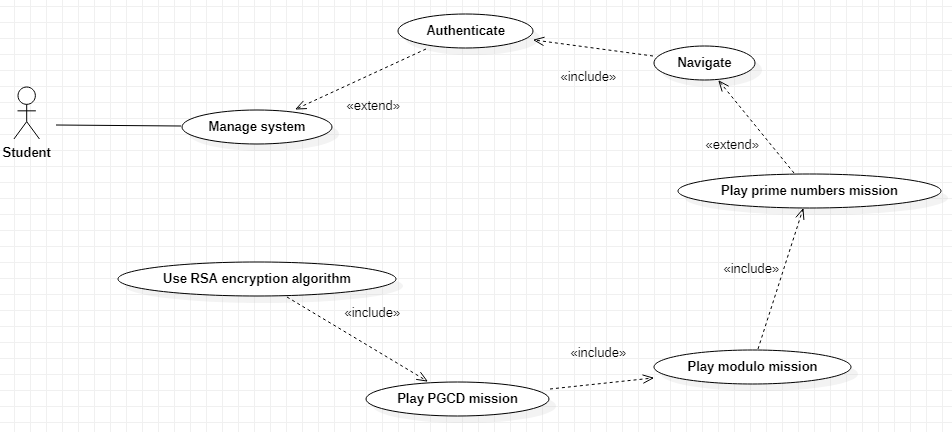
**Table 12:** "Update missions' values" teacher's use case description

**Table 13:** "Update student's status" use case description

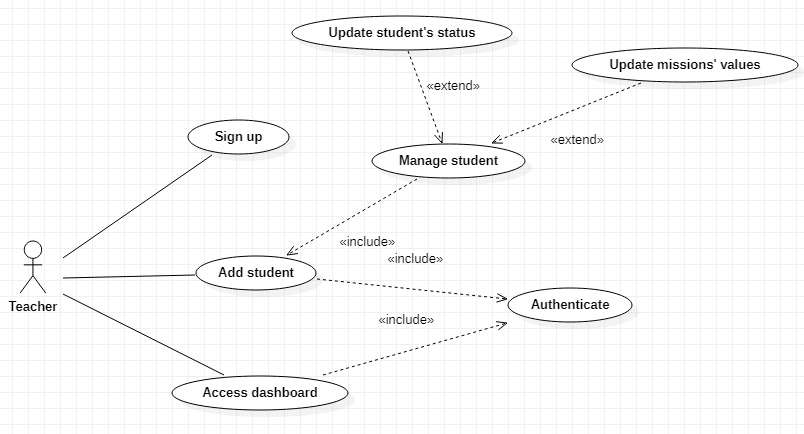
|  |
| --- |
| **Identification** |
| **Title:** Update student’s status.  **Actor:** Teacher |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The teacher status is enabled by the administrator. * The teacher is authenticated. * Student is added |
| **Nominal scenario:**   1. The teacher access to students list. 2. The teacher selects specific student. 3. The teacher updates the student’s values. 4. The system sends the new status to database. |
| **Alternative scenario:**   * Empty student list. * Internet connection or database server is unavailable. * The teacher status is disabled by the administrator. |
| **Post-condition:**   * Student’s status is updated. |

|  |
| --- |
| **Identification** |
| **Title:** Access dashboard.  **Actor:** Teacher |
| **Description of scenarios** |
| **Pre-condition:**   * Internet connection is available. * Database server is available. * The teacher status is enabled by the administrator. * The teacher is authenticated. |
| **Nominal scenario:**   1. The teacher access to dashboard tab. 2. The teacher checks overall performance for his students. |
| **Alternative scenario:**   * Empty dashboard. * Internet connection or database server is unavailable. * The teacher status is disabled by the administrator. |
| **Post-condition:**   * Dashboard accessed. |

**Table 14:** "Access dashboard" teacher's use case description



## 1.2 Use case diagrams



**Figure 5:** Teacher's use case diagram

# Chapter 3: Conceptual study

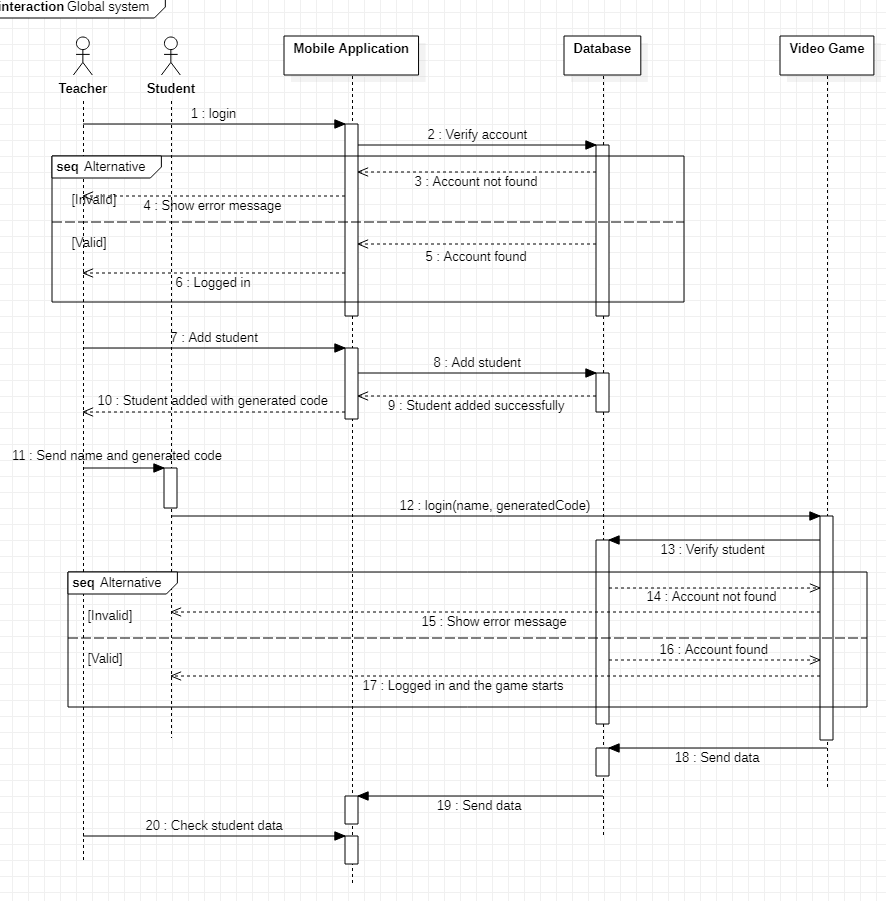
# Introduction

This chapter presents the design part which is used to realize different types of diagram which models the different parts of the project in order to better understand the system.

# Development of the static model

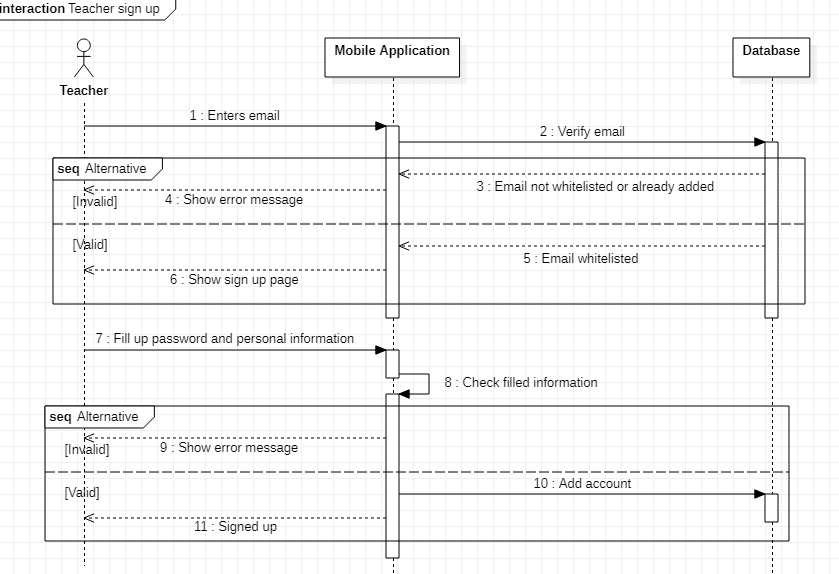


* **“Teacher” class:** is the class that represents the information of a teacher.
* **“Student” class:** is the class that represents the information of a student.
* **“GameInstance” class:** is the class that represents the information of a game instance which starts running by the video game’s launch.
* **“Level” class:** is the class that represents the information of a level in the video game.
* **“Character” class:** is the class that represents the information of different types of character.
* **“CharacterOne” class:** is the class that inherits the class “Character” and has additional attributes concerning the first character.
* **“CharacterTwo” class:** is the class that inherits the class “Character” and has additional attributes concerning the second character.
* **“Mission” class:** is the class that represents the information of different types of mission.
* **“PrimeNumbersMission” class:** is the class that inherits the class “Mission” and has additional attributes concerning the prime numbers mission.
* **“ModuloMission” class:** is the class that inherits the class “Mission” and has additional attributes concerning the modulo mission.
* **“PgcdMission” class:** is the class that inherits the class “Mission” and has additional attributes concerning the pgcd mission.



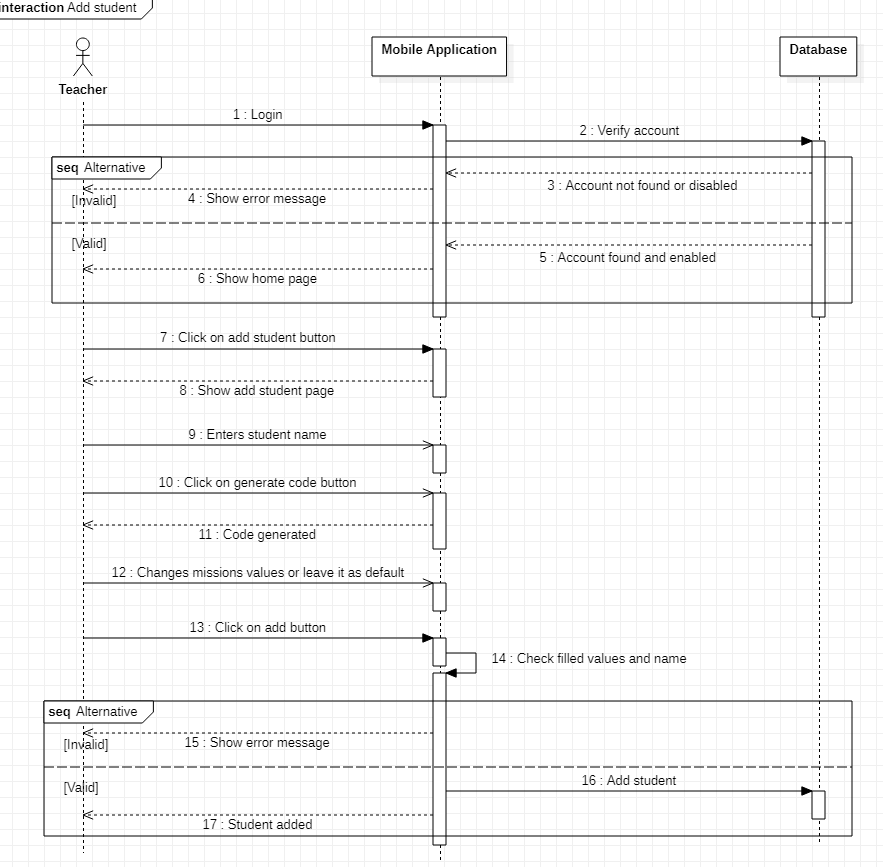
# Development of the dynamic model

## Construction of the mobile application sequence

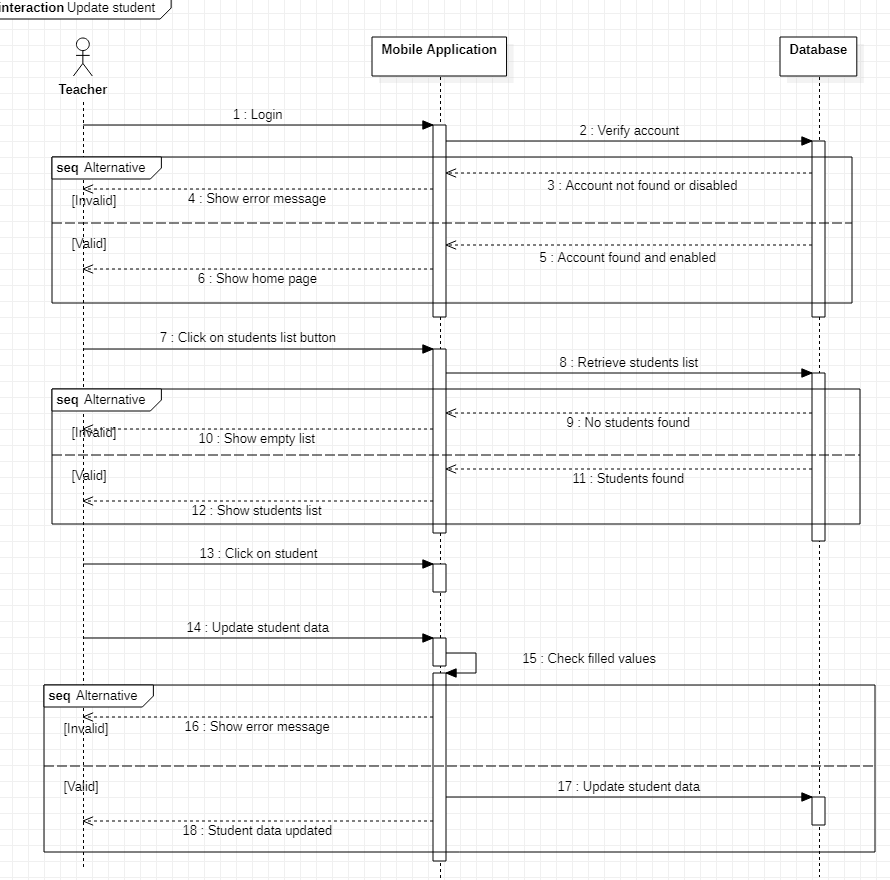


**diagram**



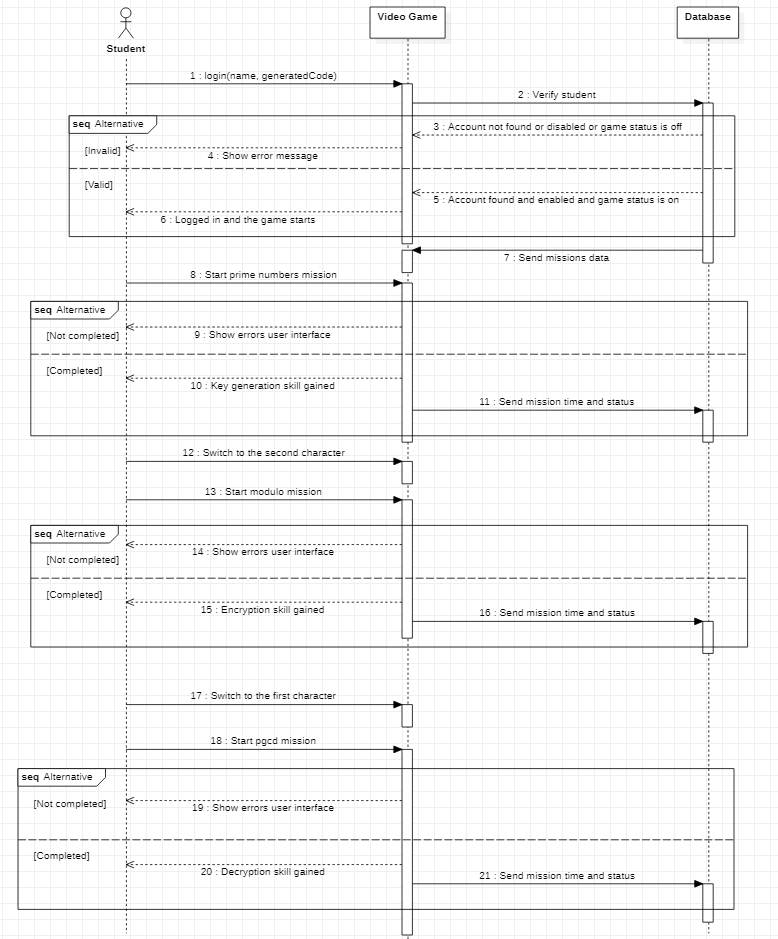




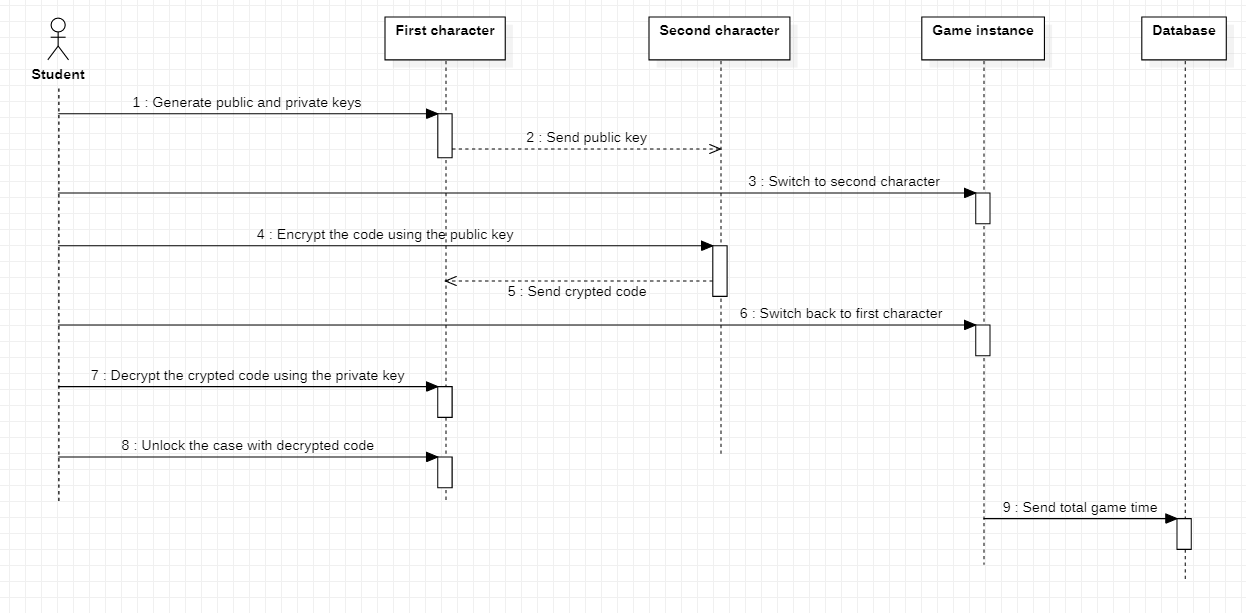




* 1. **Construction of the video game sequence diagram**







# Conclusion

 In this

chapter, we graphically represented the different parts of the project. Next, we will discuss the realization phase of the project.

# Chapter 4: The realization

# Introduction

In this last chapter, we will present the realization of our project, it’s the phase of the construction and implementation of the application.

In fact, we will start with a general presentation of the hardware, software setups and the tools used during development. Then we will expose the work with some screenshots.

# Development environment

## Hardware environment

During the realization phase of my project, I used a computer with the following specifications:

* **Laptop:** Asus ROG GL502VMK
* **Operating system:** Windows 10 Pro 64-bit
* **CPU:** Intel core i7-7700HQ @ 2.80GHz
* **RAM:** 16.0 GB
* **GPU:** NVIDIA GeForce GTX 1060 6.0 GB

## Software environment

### Programming languages

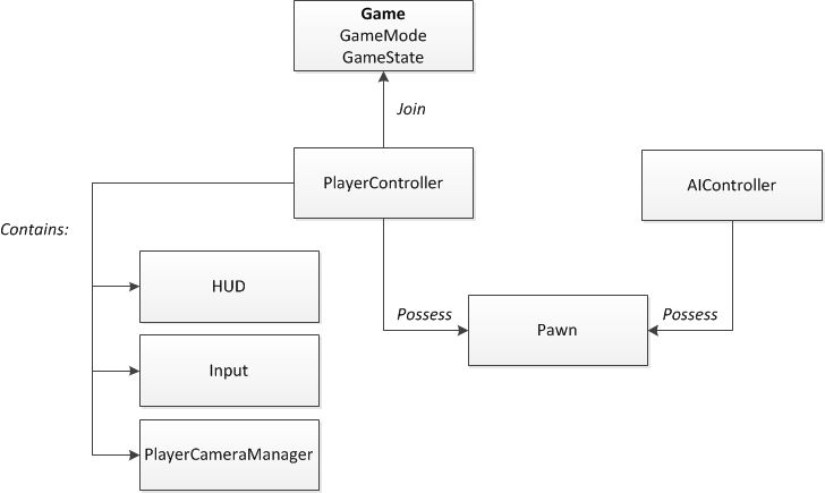
* **C++:** *is a powerful general-purpose programming language. It can be used to develop operating systems, browsers, games, and so on. C++ supports different ways of programming like procedural, object-oriented, functional, and so on. This makes C++ powerful as well as flexible.* [1]
* **Java:** *is a powerful general-purpose programming language. It is used to develop desktop and mobile applications, big data processing, embedded systems, and so on. According to Oracle, the company that owns Java, Java runs on 3 billion devices worldwide, which makes Java one of the most popular programming languages.* [2]

### Game engine: Unreal Engine 4

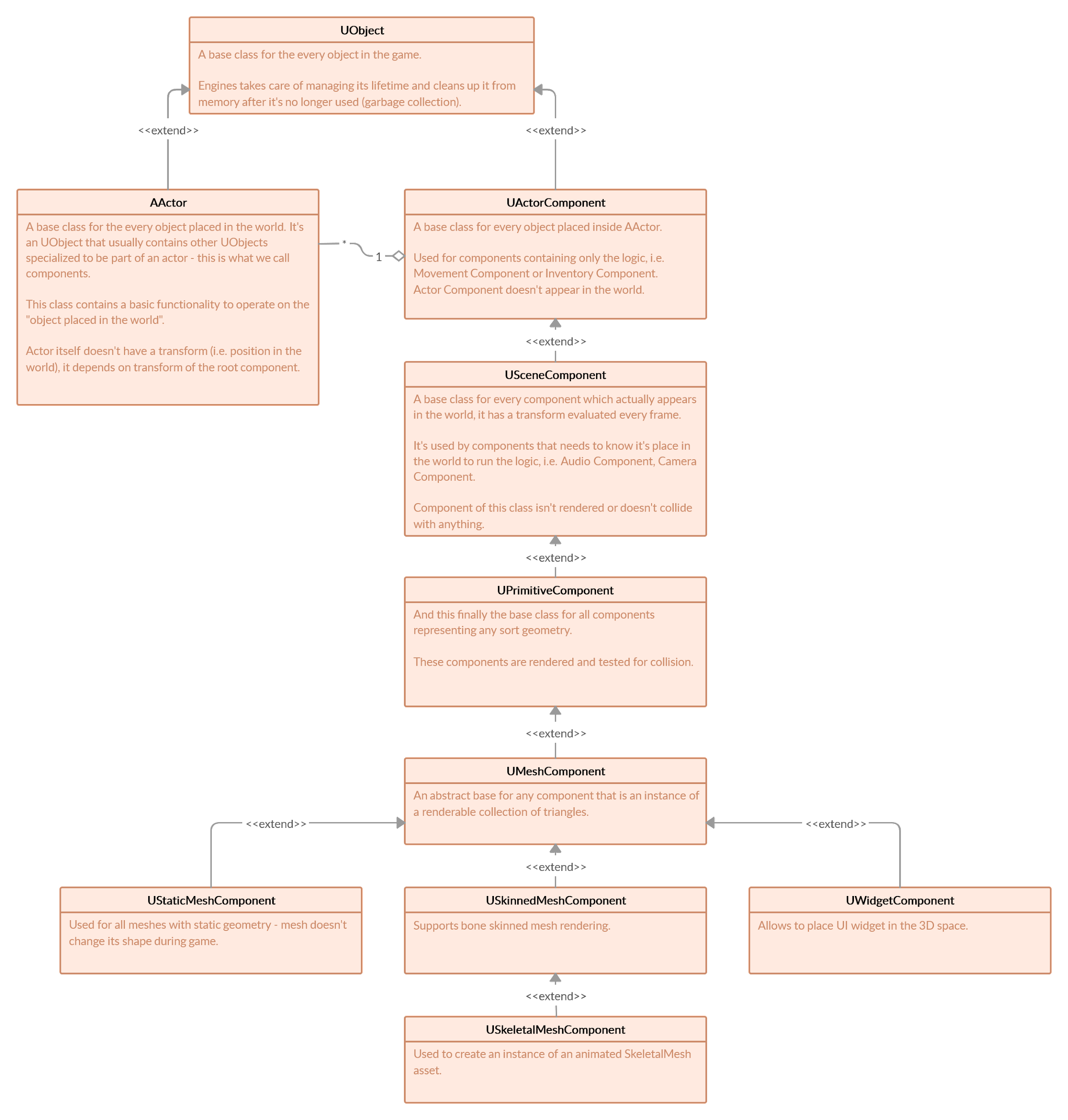
*The Unreal Engine is game engine developed by Epic Games, first showcased in the 1998 first-person shooter game Unreal. Although primarily developed for first-person shooters, it has been successfully used in a variety of other genres.*

*With its code written in C++, the unreal engine features a high degree of portability and is a tool used by many game developers today, with it being source-available. The most recent version is unreal Engine 4, which was released in 2014 under a subscription model. Since 2015, it can be downloaded for free, with its source code available on GitHub.* [3]











### Visual scripting: blueprints

*The Blueprints Visual Scripting system in Unreal Engine is a complete gameplay scripting system based on the concept of using a node-based interface to create gameplay elements from within Unreal Editor. As with many*

*common scripting languages, it is used to define object-oriented (OO) classes or objects in the engine. As you use UE4, you'll often find that objects defined using Blueprint are colloquially referred to as just "Blueprints."*

*This system is extremely flexible and powerful as it provides the ability for designers to use virtually the full range of concepts and tools generally only available to programmers. In addition, Blueprint-specific markup available in Unreal Engine's C++ implementation enables programmers to create baseline systems that can be extended by designers.* [4]

### VaRest plugin

VaRest is a plugin for Unreal Engine 4, which makes REST server communication easier to

use.



### Android studio

*Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems or as a subscription-based service in 2020.*

*It is a replacement for the Eclipse Android Development Tools (ADT) as the primary IDE for native Android application development.* [5]

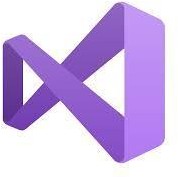




### Microsoft visual studio

*Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps.*

*Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.* [6]



### 1.2.6 Firebase

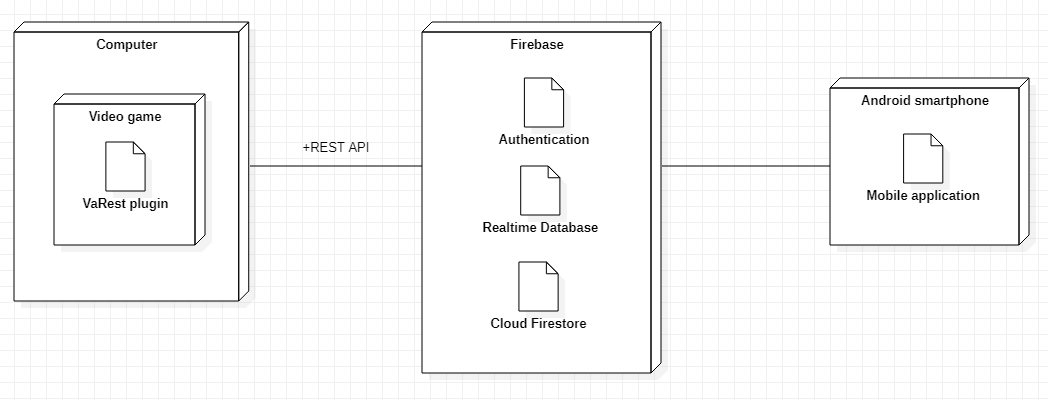
*Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014. As of March 2020, the Firebase platform has 19 products, which are used by more than 1.5 million apps.* [7]

* **Firebase Authentication:** Firebase Authentication is a service that can authenticate users using only client-side code. It supports social login providers and other service providers like. Also, it includes a user management system.
* **Firebase Realtime Database:** Firebase Real-Time Database is a cloud-hosted NoSQL database that allows you to store and synchronize data between users in real time.
* **Cloud Firestore:** Cloud Firestore is a NoSQL document database that allows you to easily store, synchronize and query data for



mobile and web applications

worldwide.

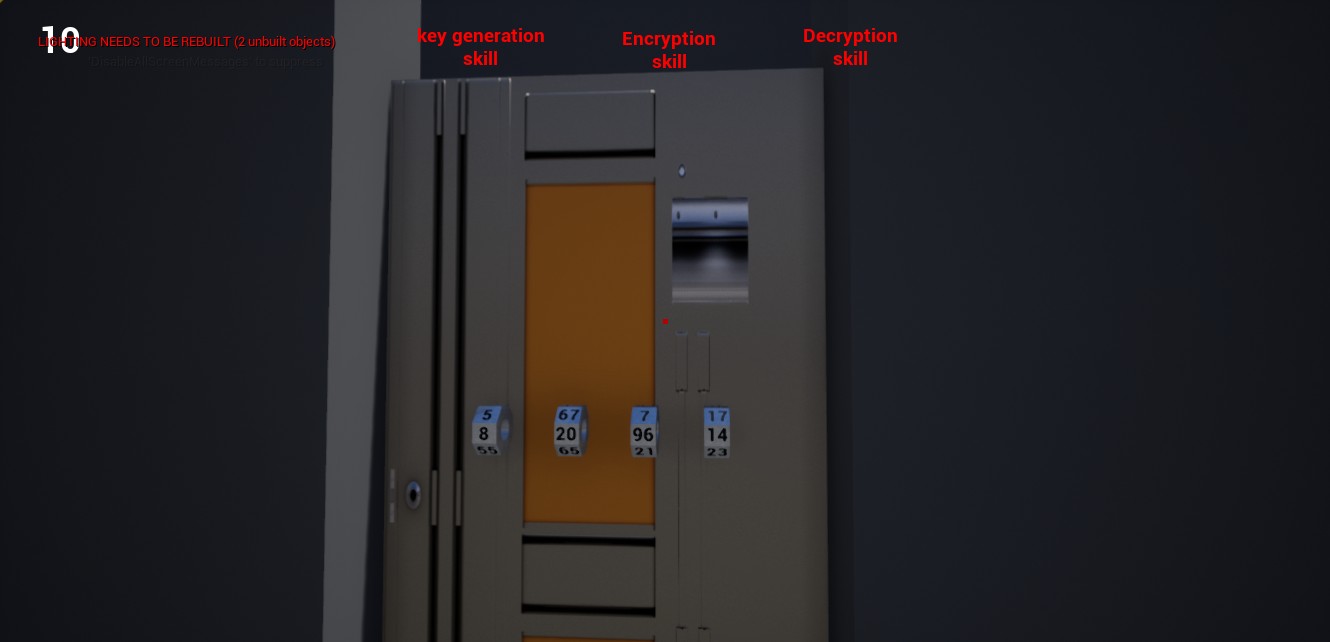


# Architectural view of the system

1. **Main graphical interfaces**

All screenshots are from development phase; design phase is still in work.

## Video game graphical interfaces

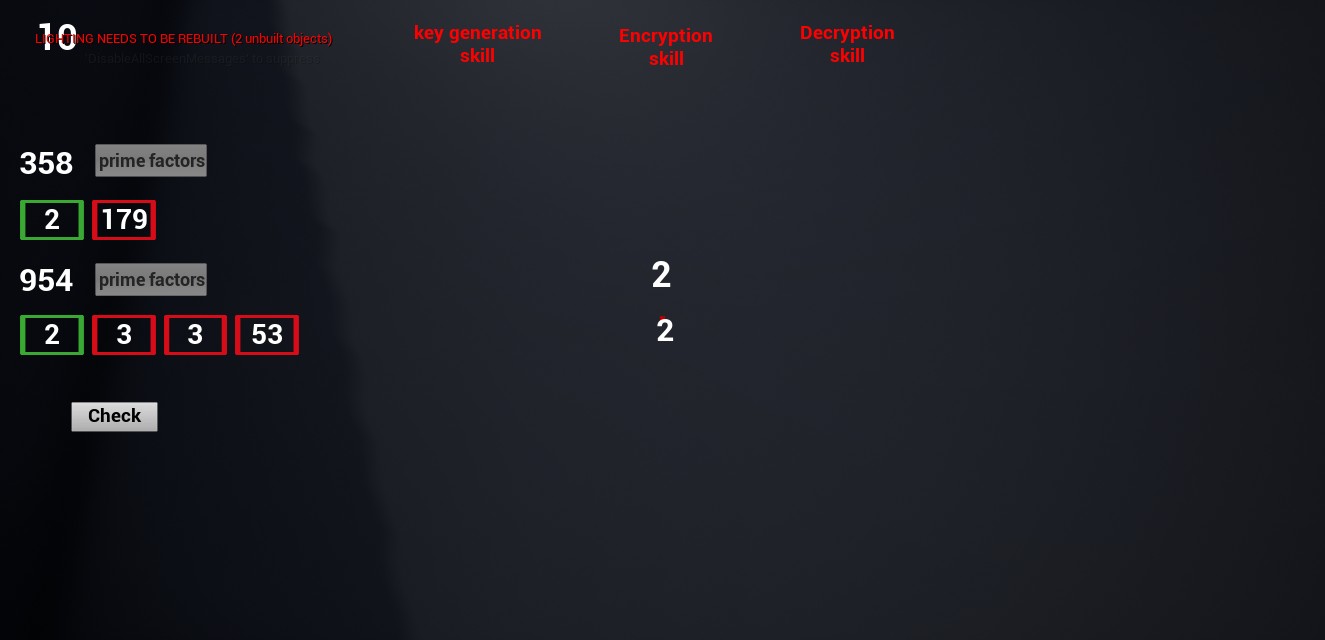


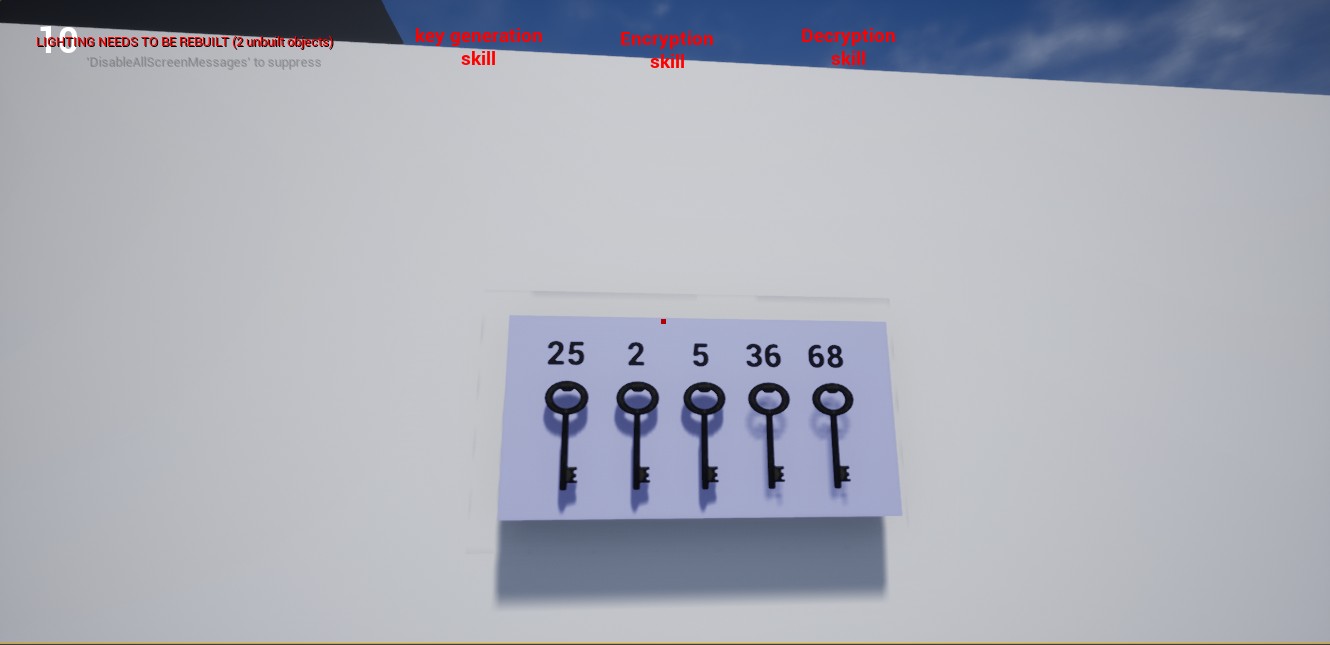
* **Prime numbers mission:** first mission in the game where you should unlock the door using prime numbers key combination before the times out, after you unlocked the door you should search for the key generation skill.



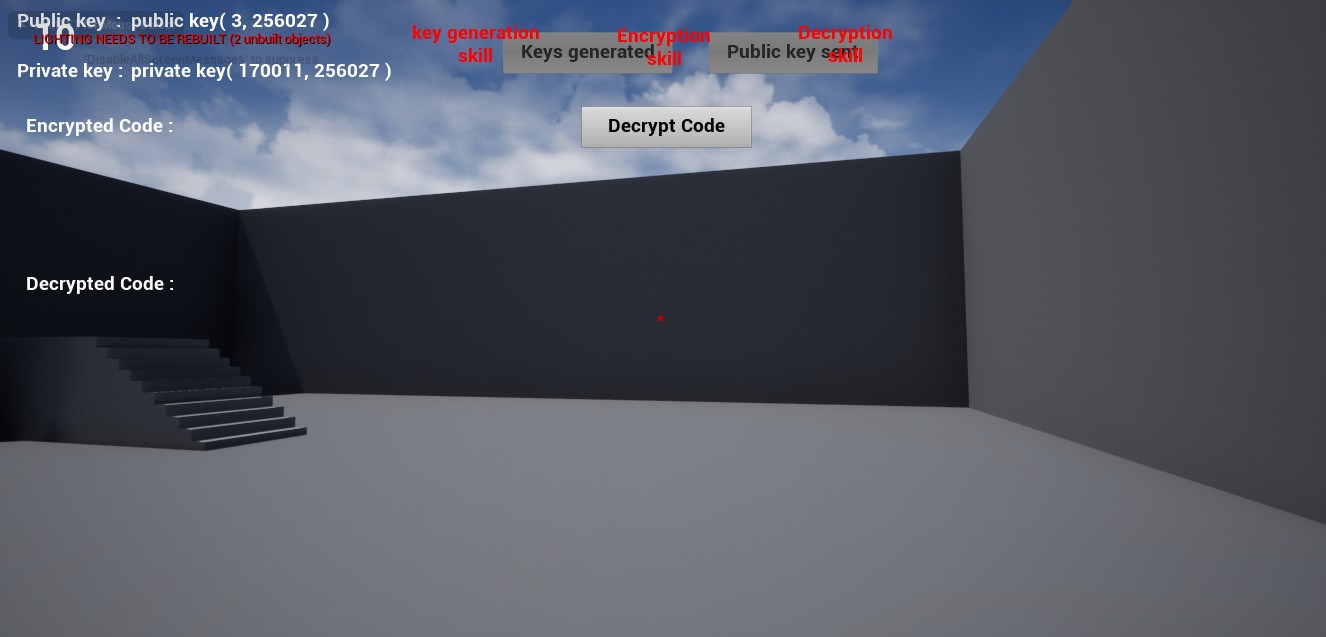
* **Modulo mission:** second mission in the game where you should to switch to the second character to play a specific modulo operation puzzle to turn out the status to get the encryption skill.
* **Pgcd mission:** third mission in the game where you should to switch back to the first character select the common prime factors of two numbers to calculate their pgcd to pick up the right key for the case to get the decryption skill.







* **Generating public and private keys:** generate both keys and send the public key to the second character.

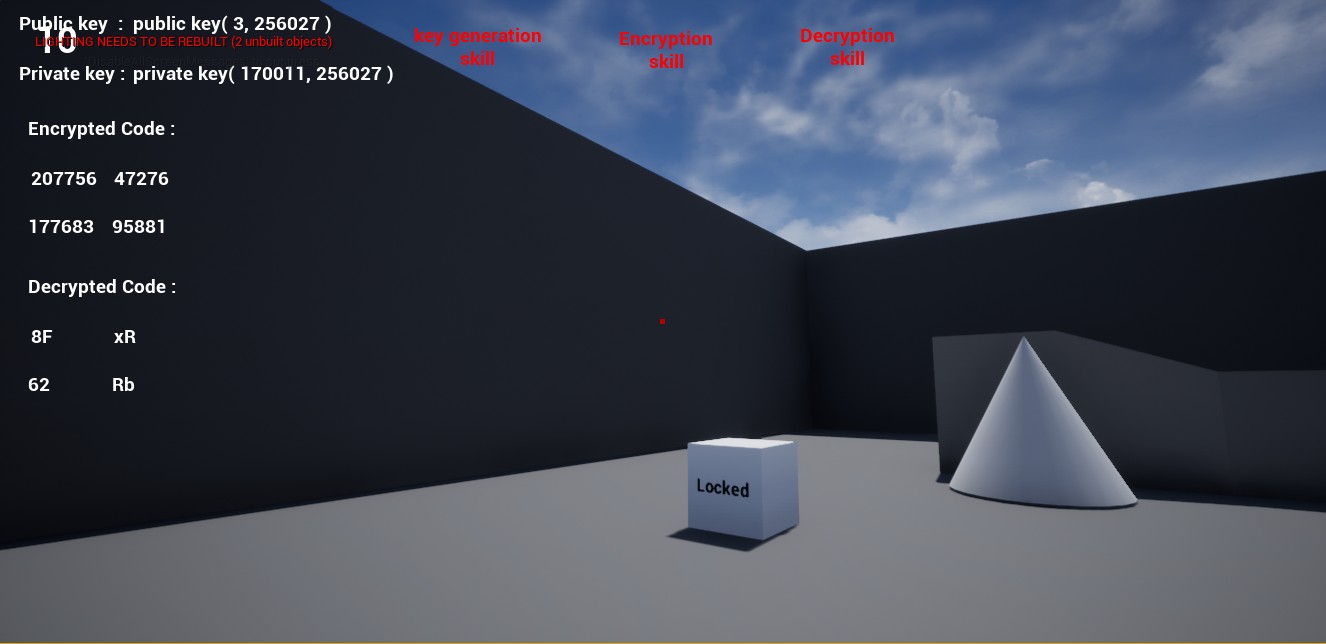


* **Encryption using public key:** here you need to switch to the second character to encrypt the code using public key sent from the first character.



**Figure 26:** Encryption graphical interface

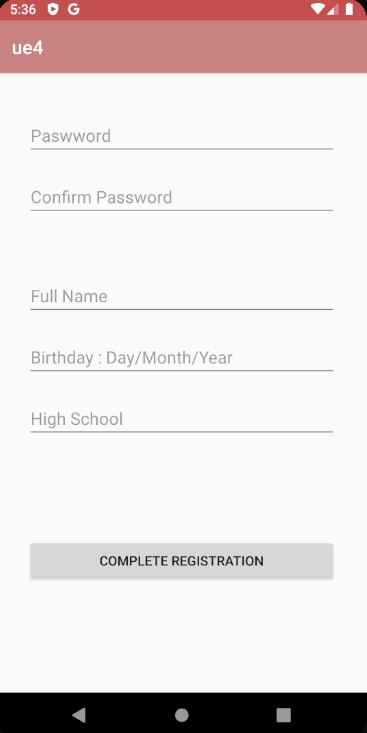
* **Decryption using private key:** here you need to switch back to the first character to



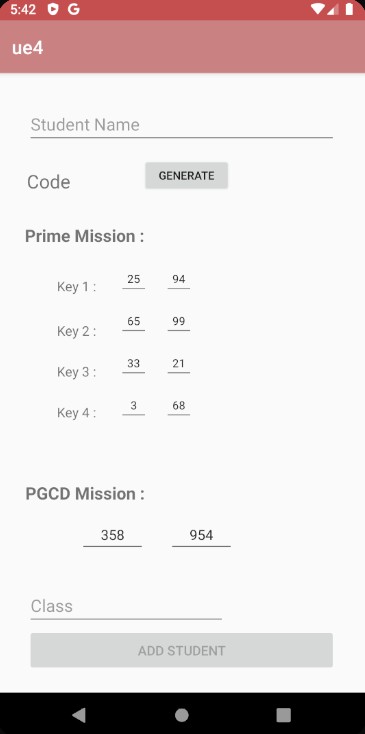
decrypt the encrypted code sent from the second character using the private key.

## Mobile application graphical interfaces

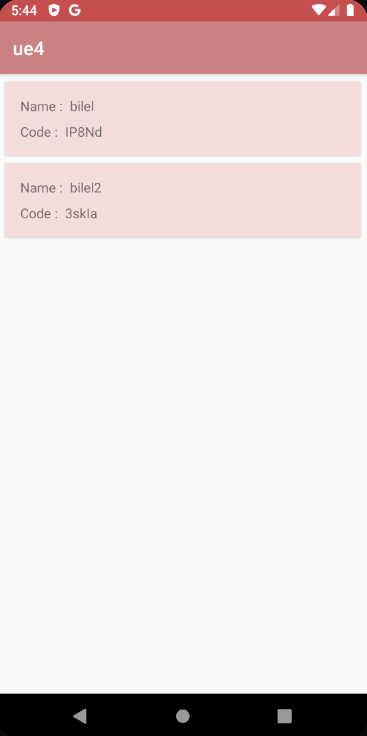
* **Sign up:** the teacher can sign up only after his email get whitelisted from the administrators.

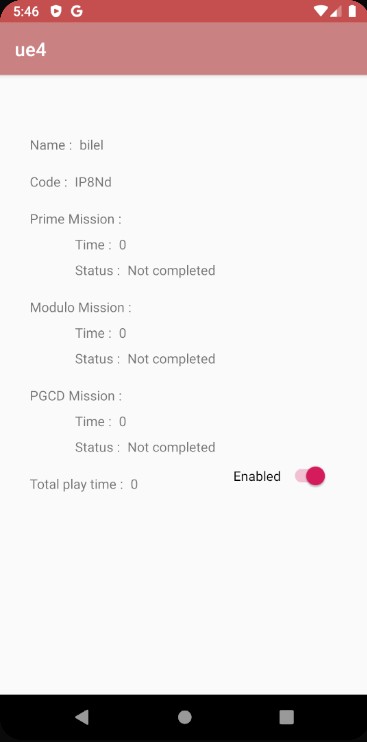


* **Add student:** the teacher can add students to access the video game.



* **Student list:** the teacher can check his student’s list.



* + **Update student:**

the teacher can update the

student status and missions’ values.



# Conclusion

In this chapter we presented the working environments for our project and then we described the different interfaces provided by our applications.

# General conclusion

The objective of our project was to develop a solution that offers both a serious video game about math lessons to students within generation Z and a mobile application for teachers to evaluate and monitor these students.

Before starting this project, we had no idea about creating web services using REST api alongside with unreal engine 4 and even less about networking in this game engine.

In the beginning, we spent some time to earn how to communicate the game engine with REST plugin and send https requests to Firebase but after some work and researches everything went well.

Also, we struggled to find a suitable gamified gameplay that is fun and educational for the students in the same time and this phase took the most of the time but after a lot of researches and surveys we found the right gameplay for our serious video game.

We managed to achieve all the main objectives and added some of secondary features such as side missions etc.

This project gave us an opportunity to learn a new programming language which is c++ and understanding more about video game development workflow.

This experience allowed us to familiarize ourselves with crunch time culture and professional life and work.

# Bibliography

1. <https://www.programiz.com/cpp-programming>
2. <https://www.programiz.com/java-programming>
3. <https://en.wikipedia.org/wiki/Unreal_Engine>
4. <https://docs.unrealengine.com/en-US/Engine/Blueprints/index.html>
5. <https://en.wikipedia.org/wiki/Android_Studio>
6. <https://en.wikipedia.org/wiki/Microsoft_Visual_Studio>
7. <https://en.wikipedia.org/wiki/Firebase>