CA400 FINAL YEAR COMPUTER APPLICATIONS PROJECT

FUNCTIONAL SPECIFICATION

LINGOLUDUS

Gamifying the learning the experience of a language

Group Members

- Yann Ndjatang: 18337813
- Fawwaz Kekere-Ekun: 18408814

SUBMISSION DATE

19/11/2021

0. Table of contents

1. Introduction

1.1 Overview

The project we are working on is a Computer Assisted Language Learning game called LINGOLUDUS. This application's main goal is to create a fun and engaging experience for people who are aiming to learn a new language. The app will aim to teach things like different tenses in the language, grammar and numbers.

As briefly mentioned before, our game will be used to teach different languages. We will create the framework in such a way that a user would have an option of languages to pick from when starting the game.

Each of the levels in the game will focus on learning and testing oneself on different aspects of the language. For example one level may focus on teaching numbers to players of the game. This level could be something like counting the number of platforms that a character must jump on and choosing the correct number out of a list of options.

The game will have different levels like this where there is an incentive to getting correct answers to getting answers correct. Our hope for this is that players will try to remember better what different numbers and words mean so that they can succeed in completing different levels.

1.2 Business Context

Here are few examples of what could be done with this application in terms of business:

In relation to our project, a possible business context would be to potentially upload it to the Google Play Store.
Our app could contain advertisements from other companies which would allow a steady flow of revenue to be generated.

Schools could integrate customised versions of our app into their network in which teachers and students could use it to interact remotely with each other.

2. General Description

2.1 Product / System Functions

Here we have the main functions that the app is expected to have by the end of our development cycle. Although this list of features is ideal to a well working and fully operational web app, the features are subject to change at any time over the course of the development. These functions will require the user to have a created a profile via a username

Players:

- Select language
- Start New Game
- View topic of the game
- View controls
- Load Game
- View profile and progress

2.2 User Characteristics and Objectives

Describes the features of the user community, including their expected expertise with software systems and the application domain. Explain the objectives and requirements for the system from the user's perspective. It may include a "wish list" of desirable characteristics, along with more feasible solutions that are in line with the business objectives.

The users that we plan to target for this product are beginners wishing to learn a language for the first time. This will allow us to focus on teaching less complex topics more effectively and at the same time focusing on learning while gaming aspect of our project. There is a chance that if we were to focus the app on more experienced speakers in a language that we would need to focus a lot more on the complexity of the language as opposed to the implementation of different learning techniques to teach users. Despite this being said, if we feel that we are able to add more later on in the production of the project we will incorporate a

higher range of difficulty in the language to suit intermediate language speakers also.

The user interface will allow for a simple, yet appealing user friendly experience. We expect the app to be as simple and straightforward as possible so that users can focus on actually learning rather than spending time trying to figure out the layout of the game.

2.3 Operational Scenarios

User signs up/ User loads up a created profile
Following the user having installed the application on their pc, that user can then proceed to create a new user profile or if the user already created one, he would then be able to load a user profile by choosing from a list of already existing profiles.

User selects language
Following sign up, the user will be taken to the language page where they will be able to select which language they would like to play related games for and challenge their knowledge of the language with a selection of different games.

User selects game
After selecting a language, the user will be then brought to a screen where they will have a choice of mini games to play. Each mini game will be related to a different topic based on the selected language.

User plays game
User can then play a host of mini games e.g hangman, wordsearch, scrabble etc all with the goal of learning the language.

User views controls
Before a game begins, the user will be able to view the controls of the game and how to play it

User views scores and achievements
Users can view their scores and achievements, seeing how well they're doing on different topics.

2.4 Constraints

Time constraints
As we are quite restricted on our game development process due to the time, we will focus on implementing the basic functionality and ensure it all works according with sufficient testing conducted. Over time we can grow the application to incorporate more features and functionality to it. However the management of time will need to be a top priority to our team. Especially with the additional juggling of other lectures and coursework.

3. Functional Requirements

3.1 Registration

Description

Registration will be the first step and a requirement to using this application. A simple sign up form will require the new user to enter a username. This step is crucial as it is how we can tie the users progress to that username. This avoids the possibility of a user losing his progress and scores

Criticality

Without registration, potential users wouldn't be able to access its services. Through this function we want to enforce and make sure users keep track of their progress.

Technical issues

For the usernames, they will be stored in a sqlite database

Dependencies with other requirements

The user must have the application installed on their pc

3.2 Selecting the language

Description

Following registration the user will prompted to a screen where they will have the option to select a language. For now this will only display one language with the hope of adding another language if time permits.

Criticality

This step is critical as this will allow the users gaming data to be properly stored and tied to their username

Technical issues

This step allows the chosen language games to be displayed and will help keep track of the users progress on a game for the chosen language.

Dependencies with other requirements

Must have created a user with a unique username.

3.3 Games Selection

Description

Once the user has selected the language they would like to challenge themselves on they will be brought to the games selection screen. From here the user will be able to choose between a selection of minigames created to help the user learn different language topics such as verbs.

Criticality

This gives the user a choice of games to play and challenge themselves on different topics which is the main purpose of our application.

Technical issues

Creating a user friendly selection menu where a user can easily choose a game to play

Dependencies

Must have created a user with a unique username.

3.4 Playing the game

Description

The user will be able to play a selected games such as hangman, wordsearch, scrabble etc all with the goal of learning the language.

Criticality

This is the main purpose of our application. Users will be able to play and learn at the same time.

Technical issues

Technical aspect of it will be to create games that are easy to understand as our main purpose is to teach the user.

Dependencies

Must have created a user with a unique username.

3.5 Viewing game controls

Description

Before a game begins, the user will be able to view the controls of the game and how to play it

Criticality

This will allow the user to quickly learn how to play the game and see the topics they will be tested on.

Technical issues

This will require us to create a screen showing the details of each game before the user commences a game

Dependencies

Must have created a user with a unique username, selected a language and picked a mini to begin playing.

3.6 Viewing Scores and rewards achieved

Description

The user will then be able to view their scores and achievements, seeing how well they're doing on different topics.

Criticality

This helps the user track their progress on different topics of the language with which the user can then

Technical issues

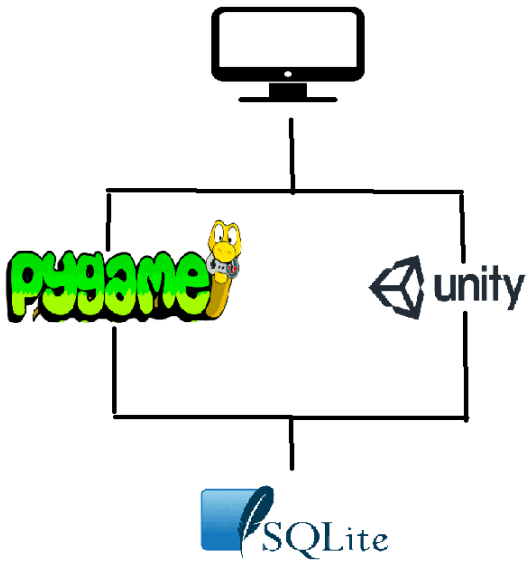
After each game session we will have to update and store the users score in a database and tie it to their username so they can keep track of their progress .

Dependencies

Must have created a user with a unique username and played at least one game for a selected language

4. System Architecture

4.1 System Architecture Diagram



Description
The above system architecture diagram shows the relationship between the different internal and external components of our system.

At the top level of the diagram, it shows the hardware components. Which is represented by a personal computer device which will be used to run our application.

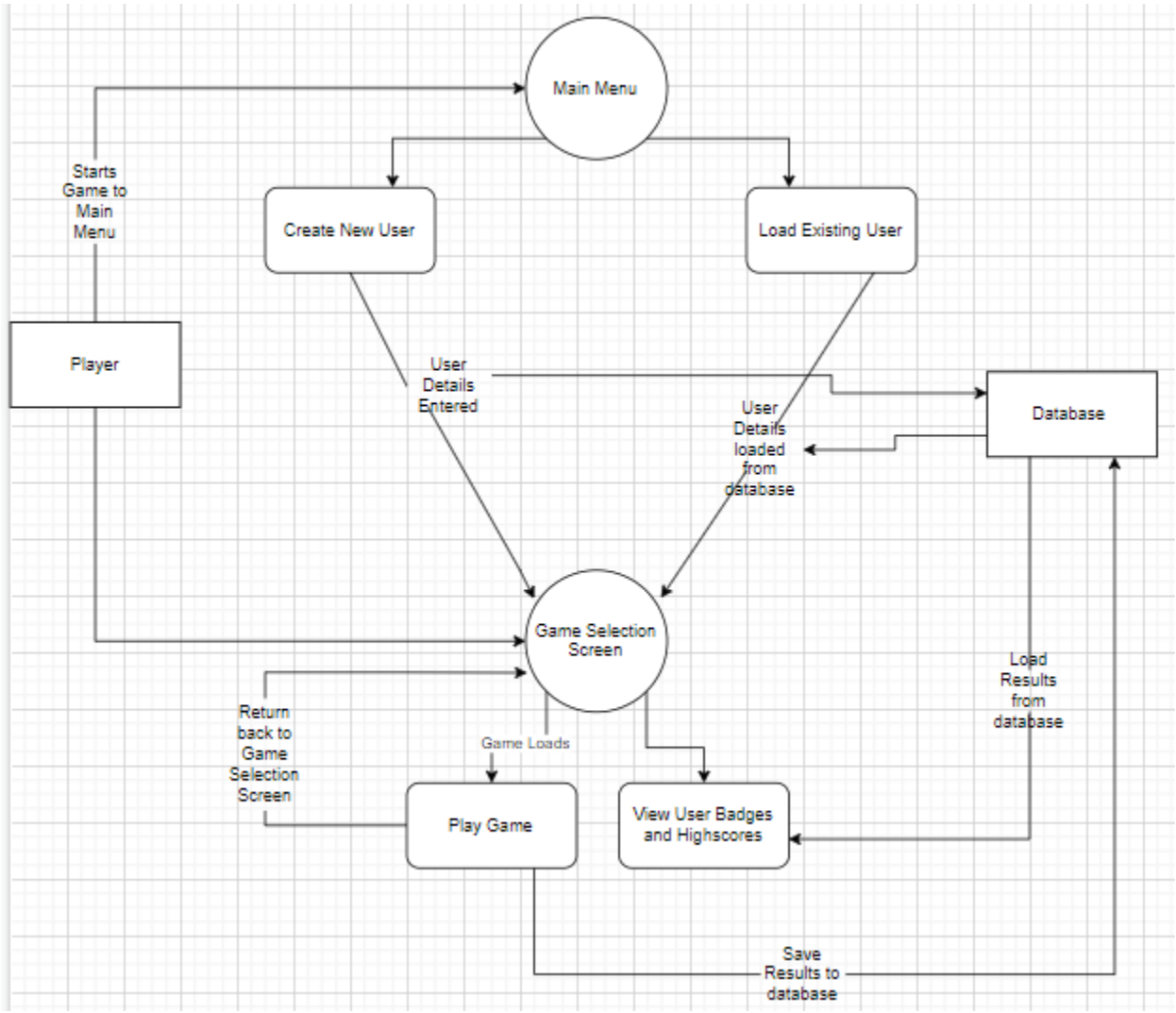
Below that shows the main software components of this application and what will be used to create it. The logos show the following: pygame and unity. We're still undecided on which one we will use as we don't have any experience with either

but these are currently our two options. Our codebase will live within github and will be written within VSCode where we can take advantage of its extensions and leverage its testing tools.

At the bottom level of this diagram is our application's backend. Here the main component in this will be a Sqlite database to store the users data.

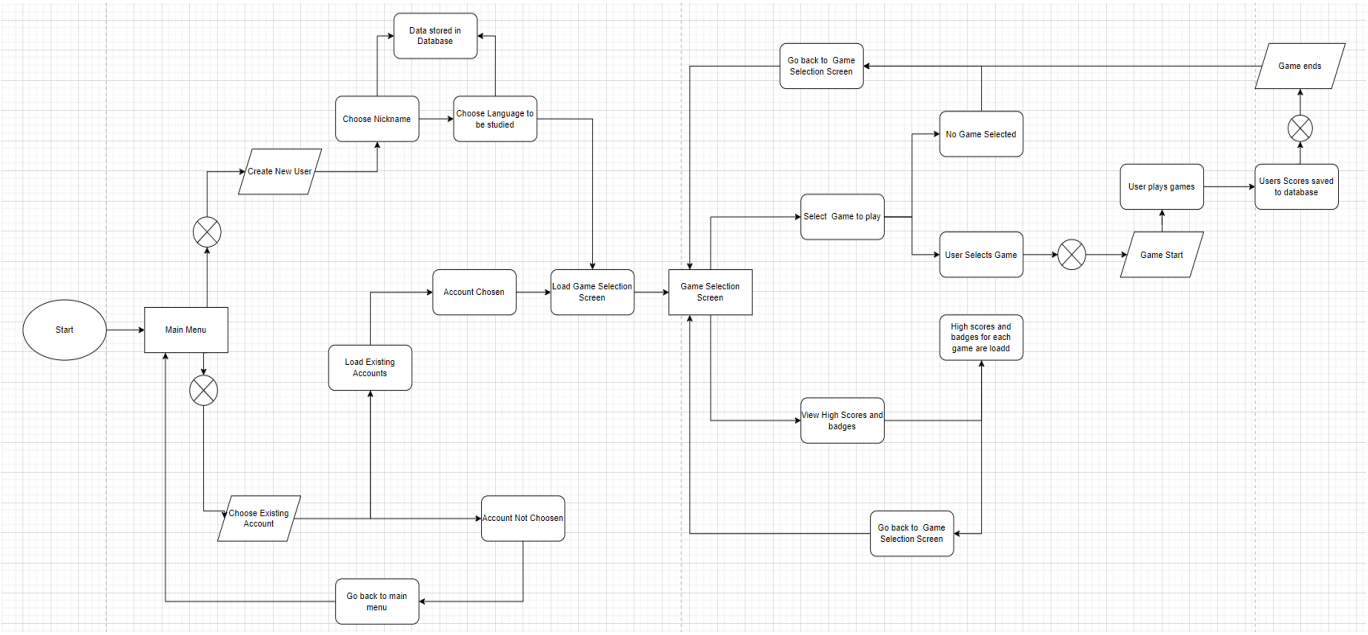
5. High-Level Design

5.1 Context Diagram



The context diagram shows the interactions and requirements of our various entities with the application. It showcases an overview of the data flow at the highest level.

5.2 Logical Data Flow Diagram

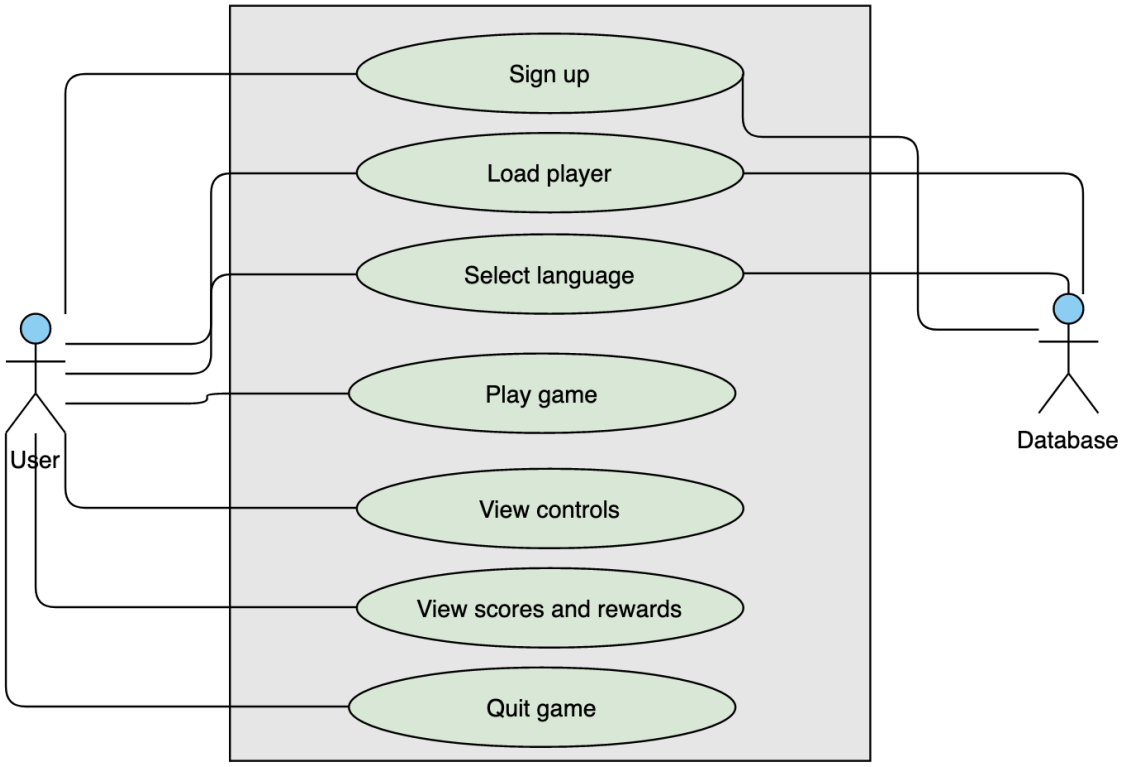


This is the logical data flow diagram which goes in details of the different activities happening.

5.3 Class Diagram

This is a diagram of all the classes we will implement within our source code. For each class, we listed the class variables and its methods. We tried to include all the classes and functions that we think we will use, however, we may need to add more as our development progresses. For now this is our basis to follow.

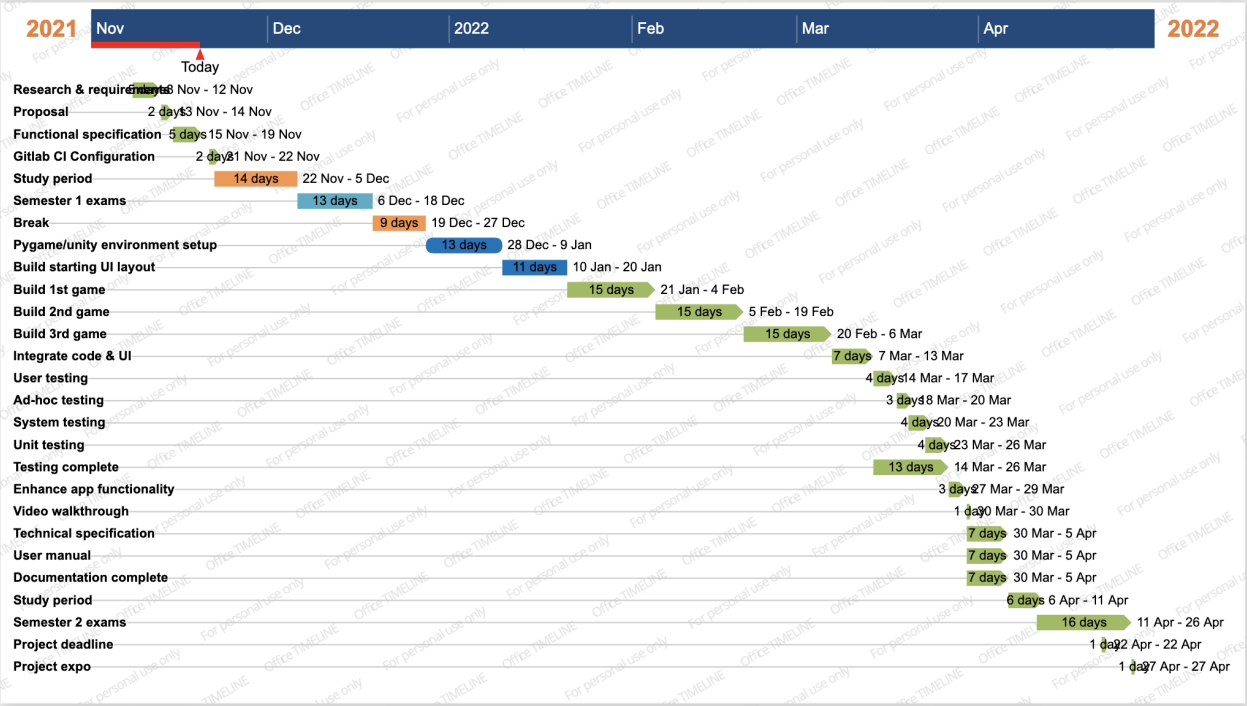
5.4 Use Case Diagram



The above use case model diagram shows the high level dynamic behaviour of our system with a user. The actor on the far left shows the user of the application and relationship between the various events of the system with the internal and external actors.

6. Preliminary Schedule

6.1 Gantt Chart



The above gantt chart is sectioned according to the semester. They show our ideal schedule for the course of this project. However this is subject to change. Saying that, our goal is to stick tightly to the deadlines that we have set out because we started late due to us not having an approved project.

Predicting the time frame for developing our source code can be tricky as it's hard to determine the exact time we will need, giving the unexpected errors and bugs that may arise later. We've attempted to incorporate time for that nevertheless. But if we're being realistic, it may require some extra time in our schedules.

We created the above chart through our pre-existing task list shown in the next section.

6.2 Task list

Task	Start Date	End Date	Duration
Research & requirements	8/11/2021	12/11/2021	4
Proposal	13/11/2021	14/11/2021	1
Functional specification	15/11/2021	19/11/2021	4
Gitlab CI configuration	21/11/2021	22/11/2021	1
Study period	22/11/2021	5/12/2021	7
Semester 1 exams	6/12/2021	18/12/2021	12
Break	19/12/2021	27/12/2021	8
Pygame/unity environment setup	28/12/2021	9/1/2022	12
Build starting UI layout	10/1/2022	20/1/2022	10
Build 1st game	21/1/2022	4/2/2022	14
Build 2nd game	5/2/2022	19/2/2022	14
Build 3rd game	20/2/2022	6/3/2022	14
Integrate code & UI	7/3/2022	13/2/2022	6
User testing	14/3/2022	17/3/2022	3
Ad-hoc testing	18/3/2022	20/3/2022	2
System testing	20/3/2022	23/3/2022	3
Unit testing	23/3/2022	26/3/2022	3
Testing complete	14/3/2022	26/3/2022	12
Enhance app functionality	26/3/2022	29/3/2022	3
Video walkthrough	30/3/2022	30/3/2022	1
Technical specification	30/3/2022	5/4/2022	6
User manual	30/3/2022	5/4/2022	6
Documentation complete	30/3/2022	5/4/2022	6

Study period	5/4/2022	10/4/2022	5
Semester 2 exams	11/4/2022	26/4/2022	15
Project deadline	22/4/2022	22/4/2022	1
Project expo	27/4/2022	27/4/2022	1
Project demonstration	TBC	TBC	TBC

The above task list shows the preliminary list of all our upcoming deliverables and milestones. We've attempted to predict our tasks as best as possible. But as previously mentioned, the timeline of the completion of each task is not set in stone.

Through both of our INTRA placements we used the scrum agile framework approach when it came to working with our teams. In an effort to follow and put into practice what we learnt, we plan to incorporate that methodology into our own project. Not only will this provide us structure, but it'll help us to keep track and be punctual. The goal is to have sprints every two weeks where both members of the team would be assigned tasks from the task list. Doing this, we could update one another on our progress and help each other through any obstacles.