
Software Requirements Specification

for

SSADPro

Prepared by The Challengers

Nanyang Technological University, Singapore

April 5, 2020

Revision History

Name	Date	Reason For Changes	Version
Whole Team	13/02/2020	Initial Document	1.0

Table of Contents

<i>Table of Contents</i>	2
1. Contributors	4
2. Introduction	5
2.1. Purpose	5
2.2. Document Conventions	5
2.3. Intended Audience and Reading Suggestions	5
2.4. Product Scope	6
3. Overall Description	7
3.1. Product Perspective	7
3.2. Product Functions	7
3.3. User Classes and Characteristics	7
3.4. Operating Environment	8
3.5. Design and Implementation Constraints	9
3.6. User Documentation	9
3.7. Assumptions and Dependencies	10
4. External Interface Requirement	11
4.1. User Interfaces	11
4.2. Software Interfaces	30
5. Initial UI Mockup	32
6. Functional Requirements	34
6.1. System Functionality to be Performed	34
6.2. Use-Case Diagram	42
6.3. Use-Case Descriptions	43
6.4. Context Diagram	87
6.5. State Transition Diagram(s)	88
6.6. CRUDL Matrix	90
6.7. Decision Table(s)	92
7. Non-Functional Requirements	93
7.1. Reliability	93
8. Candidate Architecture	94
8.1. Rationale for Candidate Architecture	95
9. Subsystem Interface Design (Game Engine)	98

9.1.	Services Layer.....	98
9.2.	Framework Layer.....	104
10.	<i>Component Diagram</i>	105
11.	<i>Communication Diagram</i>	106
11.1	Student Management.....	106
11.2.	Instructor Management	107
11.3.	Compete Mode	108
12.	<i>Real Time Data Analytics</i>	109
12.1.	Student Mode	109
12.2.	Instructor Mode	111
13.	<i>Testing</i>	112
13.1.	Student Mode	112
13.2.	Instructor Mode	119
13.3.	Student Login	123
13.4.	Instructor Login	125
13.5.	Settings	127
14.	<i>Appendix</i>	129

1. Contributors

Name	Email
Kiran Mac Milin	macmilin001@e.ntu.edu.sg
Gupta Jay	jay002@e.ntu.edu.sg
Vaish Arjun	arjun014@e.ntu.edu.sg
Kanodia Ritwik	ritwik002@e.ntu.edu.sg
Bhatia Ritik	ritik001@e.ntu.edu.sg
Hasan Mohammad Yusuf	mohammad059@e.ntu.edu.sg
Koh Zhuang Chean	zkoh014@e.ntu.edu.sg
Mundhra Divyesh	divyesh001@e.ntu.edu.sg
Asok Kumar Gaurav	asok001@e.ntu.edu.sg
Bhadra Soham	bhad0001@e.ntu.edu.sg
Truong Quang Duc	duc015@e.ntu.edu.sg
Yap Jay Shen	jyap033@e.ntu.edu.sg

2. Introduction

2.1.Purpose

The document lays out a plan for the development of “**SSADPro**”, a game application by The Challengers. The intended readers of this document are current and future developers working on the “**SSADPro**” project. The plan will include, but is not restricted to, a summary of the system functionality, the functional and non-functional requirements of the system, use-case models, UI model and other data-flow diagrams.

2.2. Document Conventions

This document features the usage of the font Arial of size 12. It must be noted that each new title uses Times New Roman of size 14 and is stylized to be bold.

In numbering our detailed requirements and use cases we employ a nested (indented) numbering system whereby a title numbered 1 has subheadings 1.1, 1.2 and so on.

2.3. Intended Audience and Reading Suggestions

This document is intended for any developer, project manager, software tester or user involved in the design, development and testing of this application.

The sequence followed in this document is

1. Overall Description
2. External Interface Requirements
3. Initial UI Mockups
4. Functional Requirements
5. Non-functional Requirements
6. System Architecture
7. Appendix

Developers who are to review application capabilities, system features and other requirements may view sections 1, 2 or 4. For better understanding of terminologies, they may refer to the Data Dictionary under Section 4. Furthermore, those who wish to visualize the working of the application can view the interface in section 3.

2.4. Product Scope

SSADPro will be an application that will facilitate a user to learn the topics of Software Engineering through a gamified approach. Currently, the project is being developed for students and instructors in the School of Computer Science and Engineering at NTU. The larger goal for this project would be to incorporate gamified learning in Computer Science schools across many universities in Singapore and other countries.

3. Overall Description

3.1. Product Perspective

Our product aims to gamify and socialize teaching and learning of software engineering courses.

It is a follow up member of a family of products which seek to make the learning process interactive and fun and increase productivity with elements of ‘challenge’ and ‘Points’.

3.2. Product Functions

- The application presents the five stages of SDLC (Software Design Life Cycle) as worlds for players to navigate and answer questions in.
- Worlds are divided into levels, which consist of mini games that players complete to gain proficiency in the subject.
- In order to proceed to the next world, all players must pass a quiz.
- Players can challenge each other using the ‘Compete mode’, either by selecting from predefined questions or by creating their own tests.
- Instructors may monitor the progress of students through a dashboard wherein the progress of the players is visible.
- Instructors can also add or modify the content as they deem suitable.

3.3. User Classes and Characteristics

- Teachers who can monitor SDLC concepts taught in the game and all the learning modules. They can overlook the learning and teaching aspect of the game. They can update and modify the content weekly.
- Students who are experienced can use this game once early to revise the basic concepts and strengthen their knowledge base.
- Students who are new to SDLC can use this game daily to better understand each concept and practice from the given questions and improve their knowledge.
- Students who are in college may use this game daily in a semester as a part of their coursework.
- This game can be used in corporate training for new recruits and interns to increase their knowledge and give them a better idea. The game use would be 10-15 days in a year.

3.4. Operating Environment

This app has been built for both Android and iOS. The mobile operating systems with which it works are – Android Jellybean, v16, 4.1.x or newer, and iOS 13.4 or newer. Further, the mobile hardware compatibility is as follows – iOS devices (iPhone SE, iPhone 6S or newer) and ARM based Android devices.

In building this application, the following software dependencies required available from (<https://pub.dev/>) –

- environment:

```
sdk: ">=2.1.0 <3.0.0"  
sdk: ">=2.3.0 <3.0.0"
```

- dependencies:

```
flutter_date_picker: "^0.1.8"  
sdk: flutter  
cupertino_icons: ^0.1.2  
firebase_auth: ^0.15.5+3  
google_sign_in: ^4.2.0  
simple_animations: ^1.3.11  
path_provider: ^1.6.5  
cloud_firestore: ^0.13.4+2  
provider: ^3.1.0  
font_awesome_flutter: ^8.2.0  
charts_flutter: ^0.6.0  
volume: ^0.1.0  
url_launcher: ^5.4.2  
app_settings: 3.0.1  
share: ^0.6.3+6
```

- fonts:

```
- family: Open Sans  
  fonts:  
    - asset: assets/fonts/OpenSans-Bold.ttf
```

```
    weight: 700
    - asset: assets/fonts/OpenSans-ExtraBold.ttf
      weight: 800
    - asset: assets/fonts/OpenSans-Light.ttf
      weight: 300
    - asset: assets/fonts/OpenSans-Regular.ttf
      weight: 400
```

3.5.Design and Implementation Constraints

SSADPro requires several APIs to function properly. Following are SSADPro's third party API dependencies –

1. Google Firebase Auth REST API
(<https://firebase.google.com/docs/reference/rest/auth>)
2. Google Firestore API
(<https://firebase.google.com/docs/firestore/reference/rest>)

SSADPro requires a mobile application written in Dart language on Google's Flutter platform (along with auto-generated Ruby code from Dart language) that can be deployed either on iOS or Android platforms.

It also requires maintenance of the user information database. The application must adhere to the timing and speed performance guidelines stated by the non-functional requirements in this document.

3.6. User Documentation

Our user documentation will consist of a user manual, online help and a video description detailing the functionality of the app.

To make the structure of the user document clear and readable, format requirements are –

- Proper indentation
- Insert appropriate heading, section titles and page numbers
- Suitable text fonts and style
- Insert pictures for key steps and buttons
- Use bold to highlight certain operations

As our user may not have a technical background, our standards are –

- Simple words and short sentences
- Show operations instead of explaining a long process

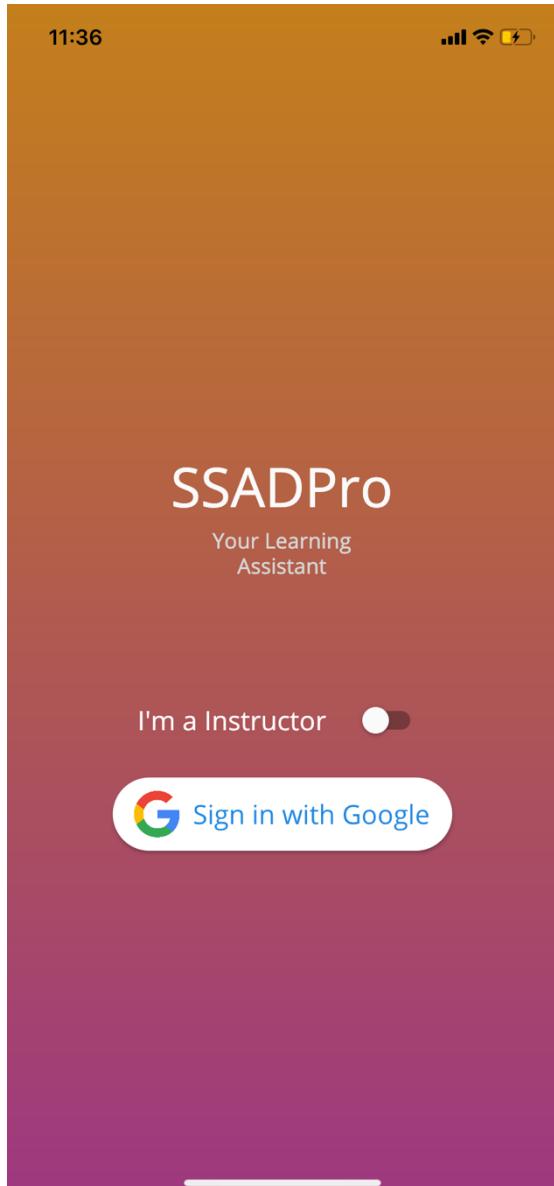
3.7. Assumptions and Dependencies

It is assumed that the user is aware of how to use web and mobile applications. In case of web application, it is assumed that the user knows how to navigate using the mouse and keyboard. In the case of mobile application, it is assumed that the user knows how to navigate using the touch screen. It is also assumed that the user will possess good internet connectivity.

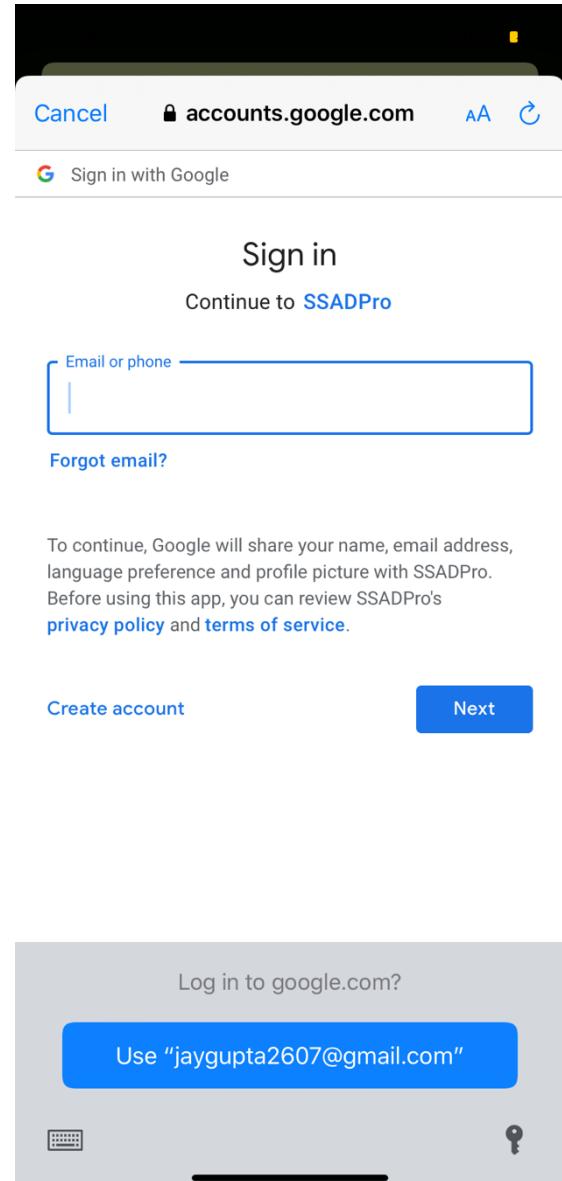
4. External Interface Requirement

4.1. User Interfaces

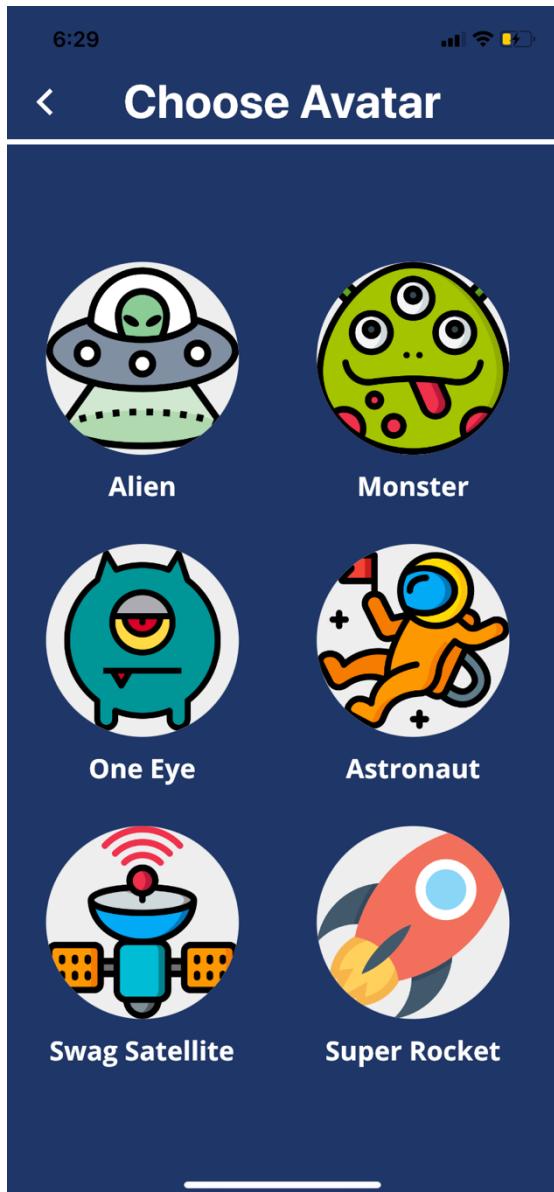
Attached are some screenshots of the application, that provide a basic insight on the user interface of the application, that is, the application in the user.



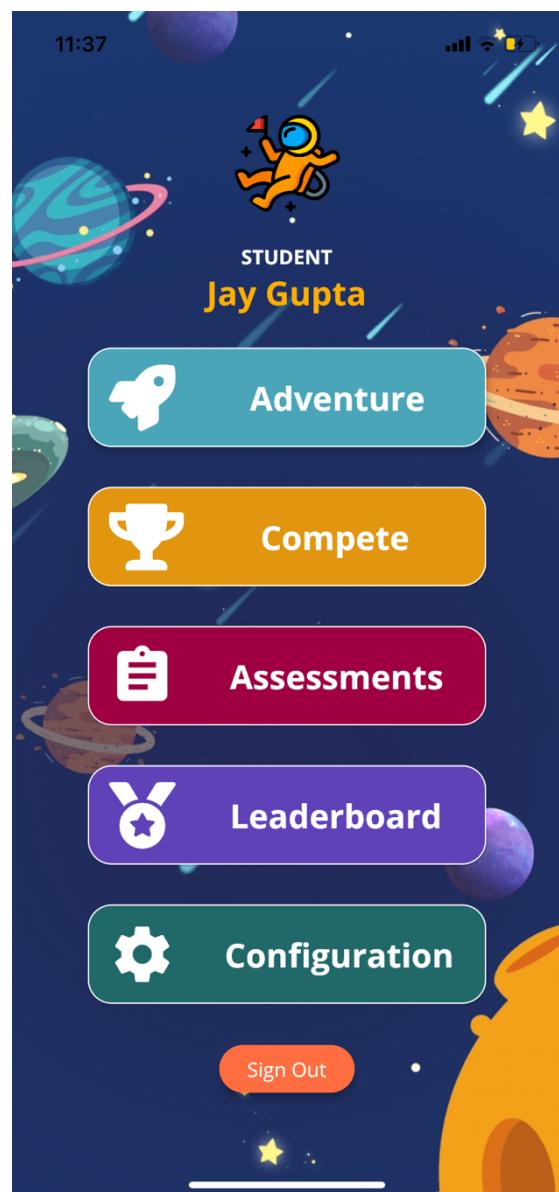
Entry Page



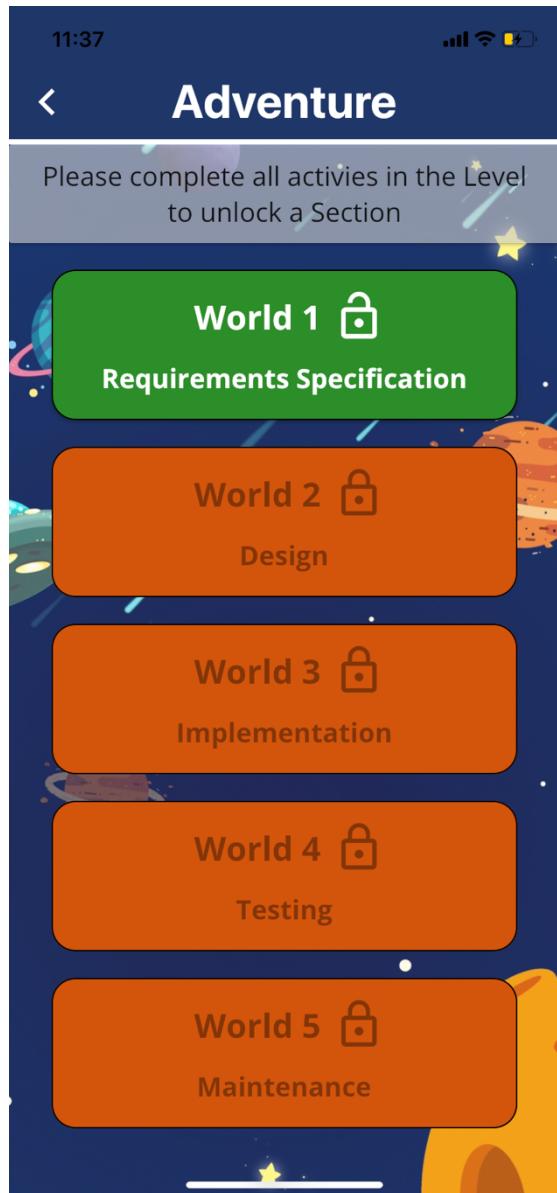
Login via Google



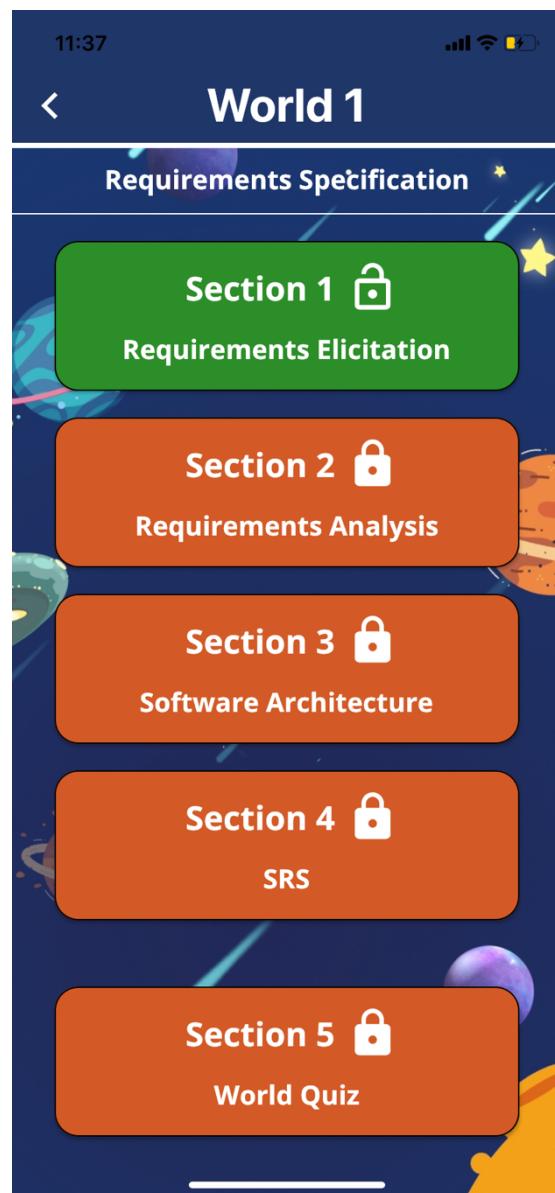
Choose Avatar (Student Mode)



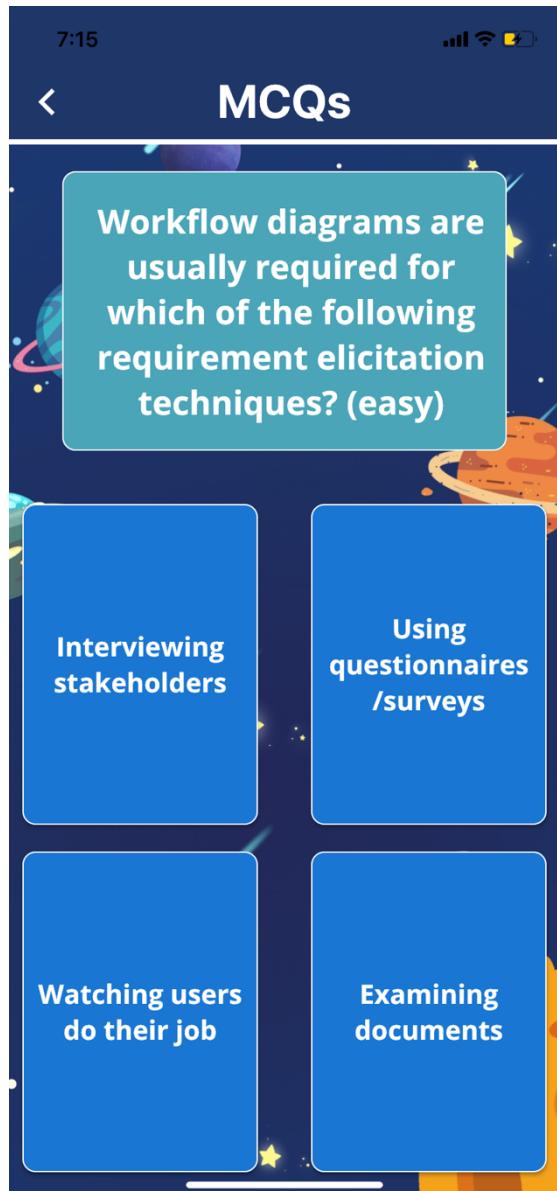
Dashboard (Student Mode)



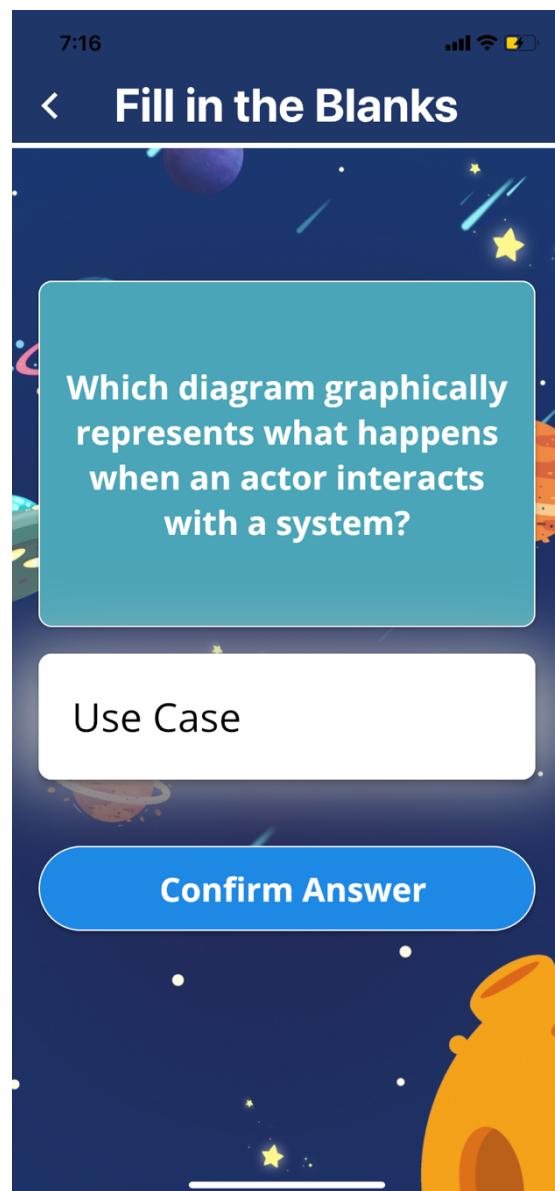
Adventure Mode (Student Mode)



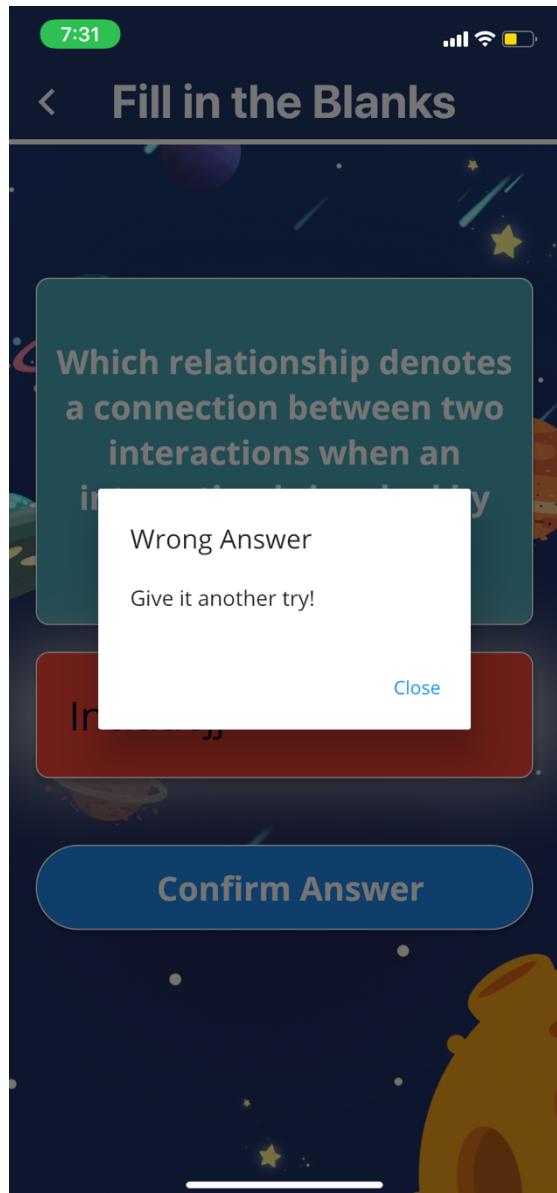
Section View (Student Mode)



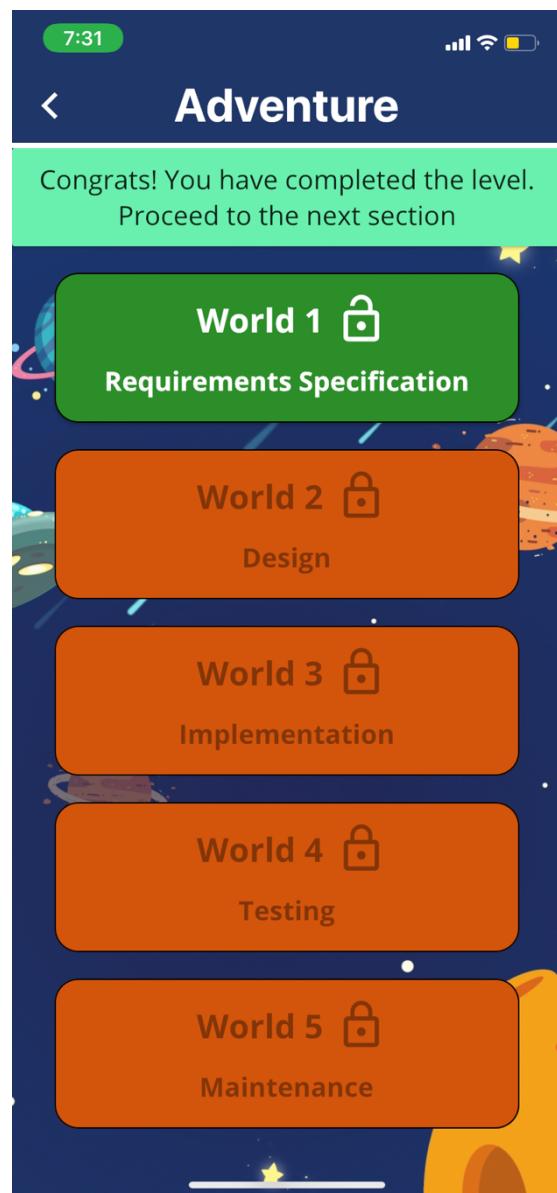
Adventure – MCQs (Student Mode)



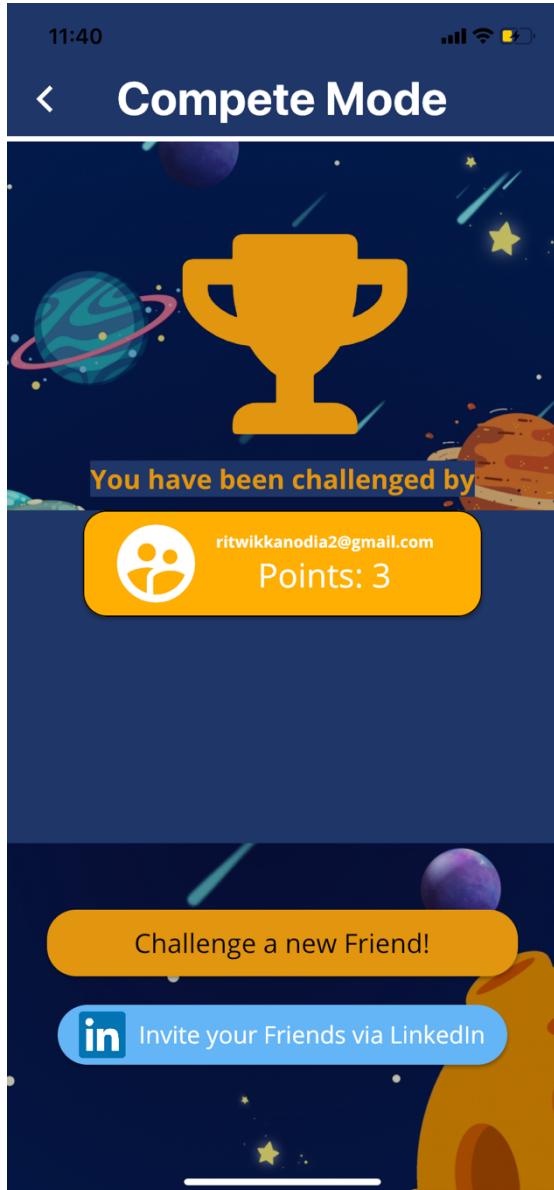
Adventure – Fill-in-the-Blanks



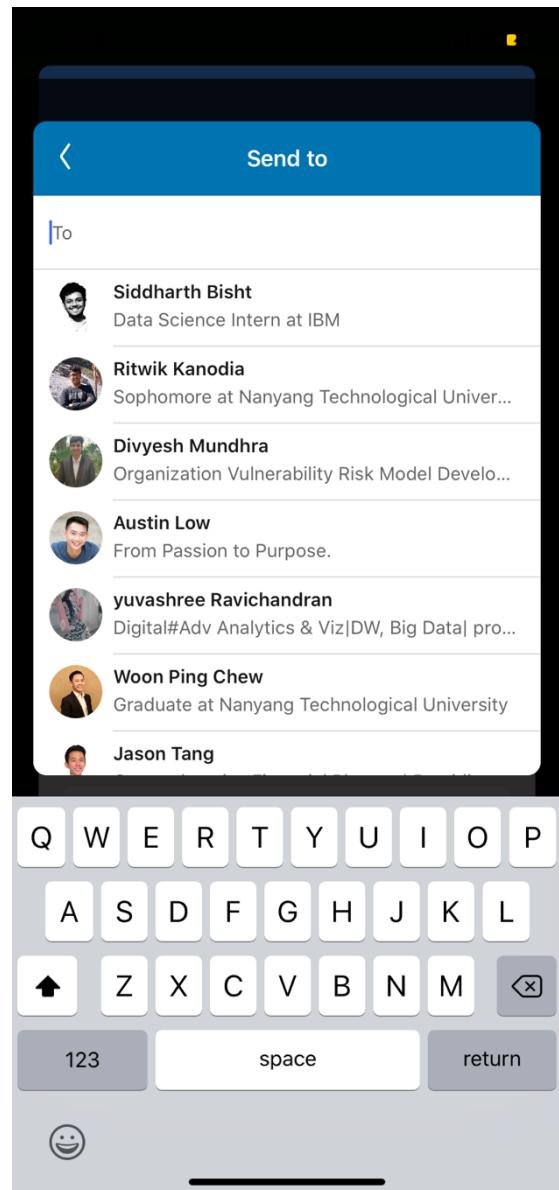
Adventure – Wrong Answer



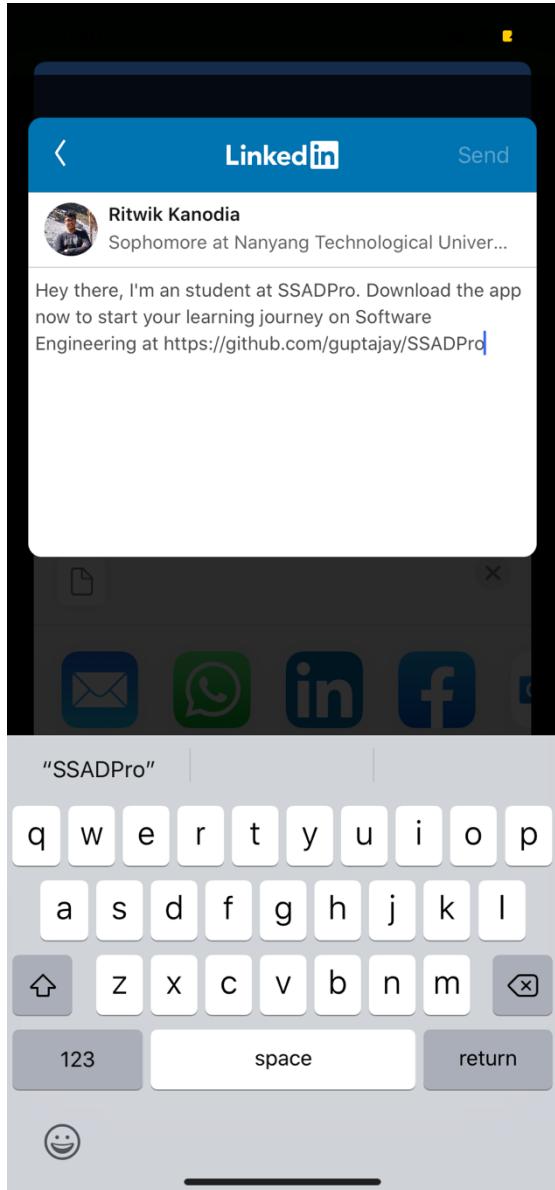
Adventure – Section Complete



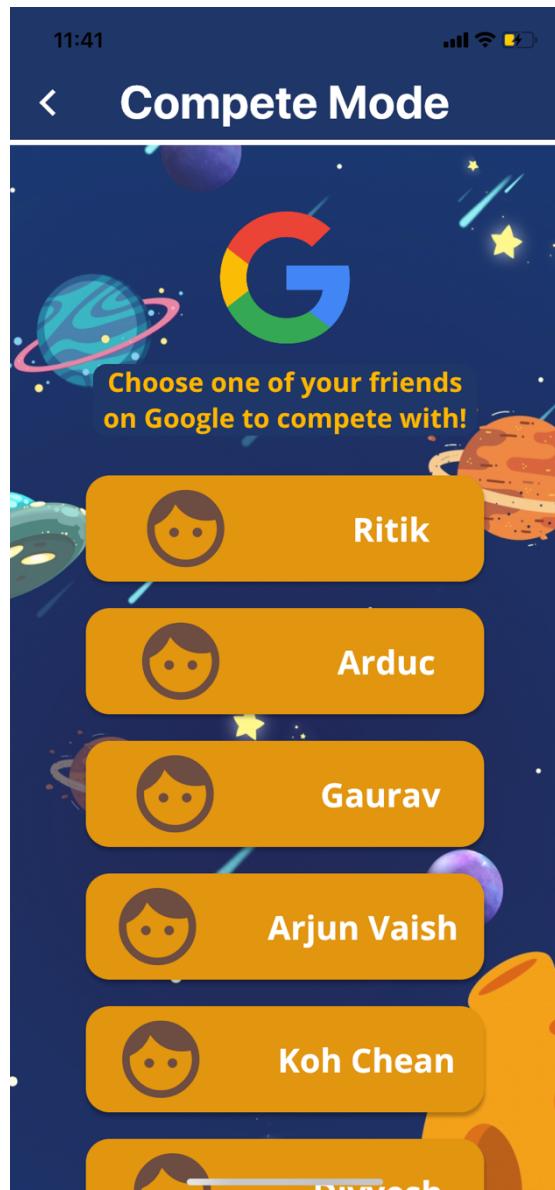
Compete Mode (Student)



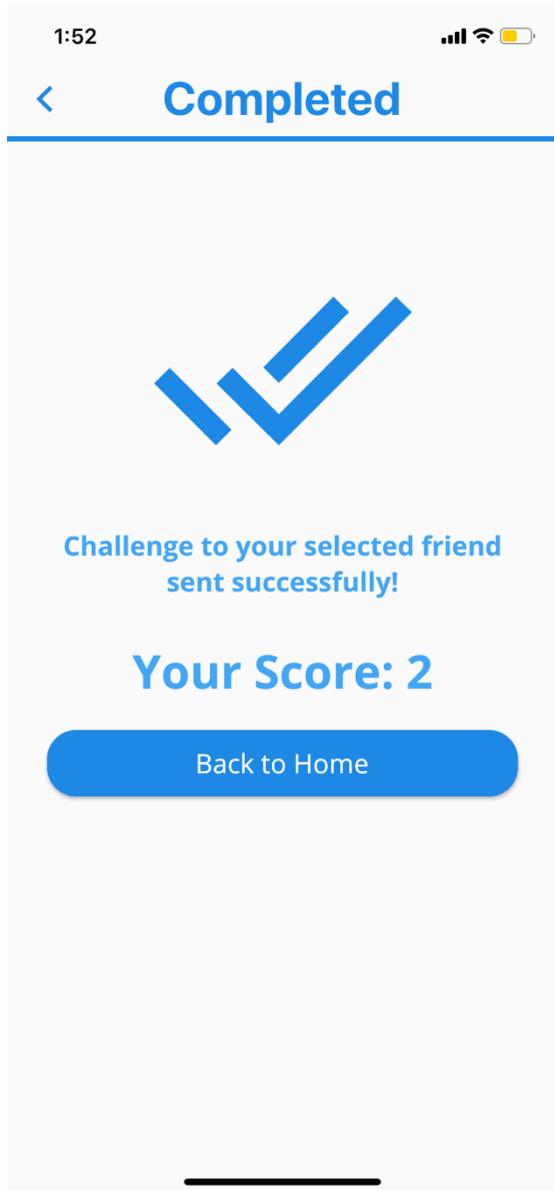
Challenge Friends via LinkedIn



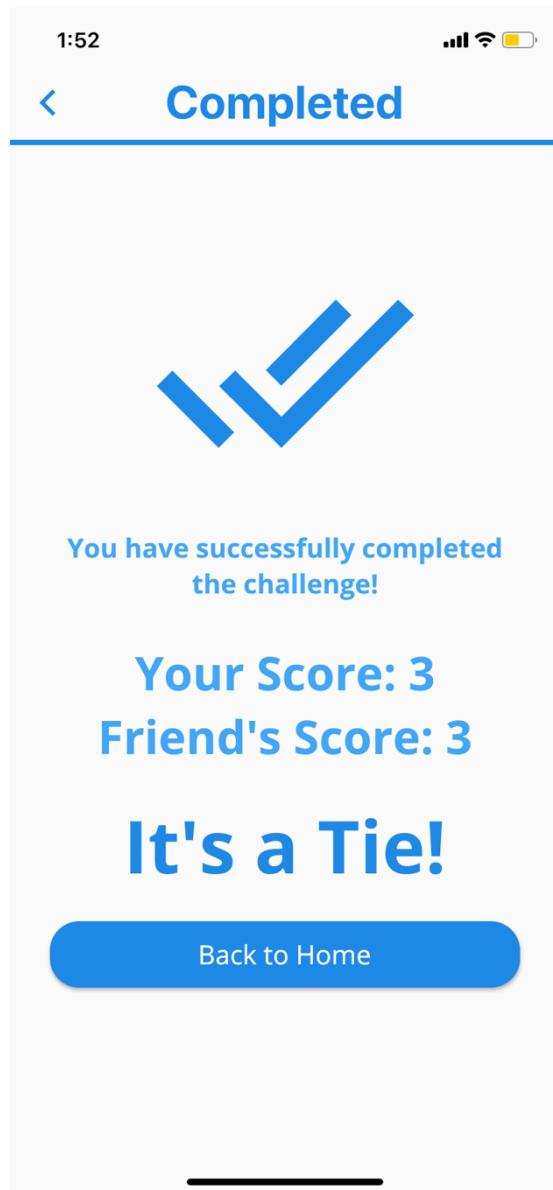
Challenge Friends via LinkedIn



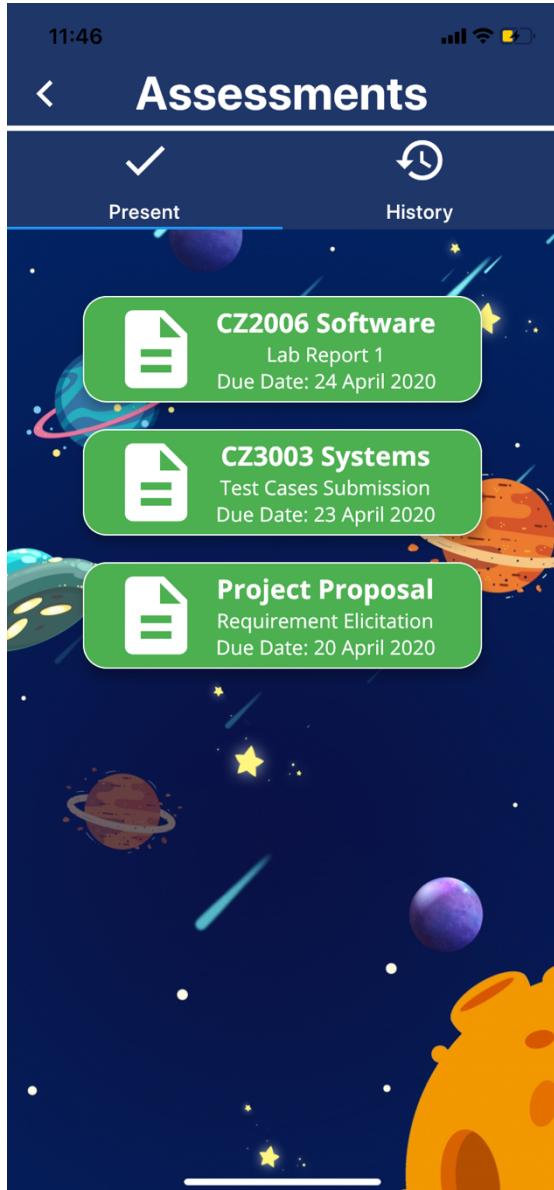
Compete Mode – Challenge Friends



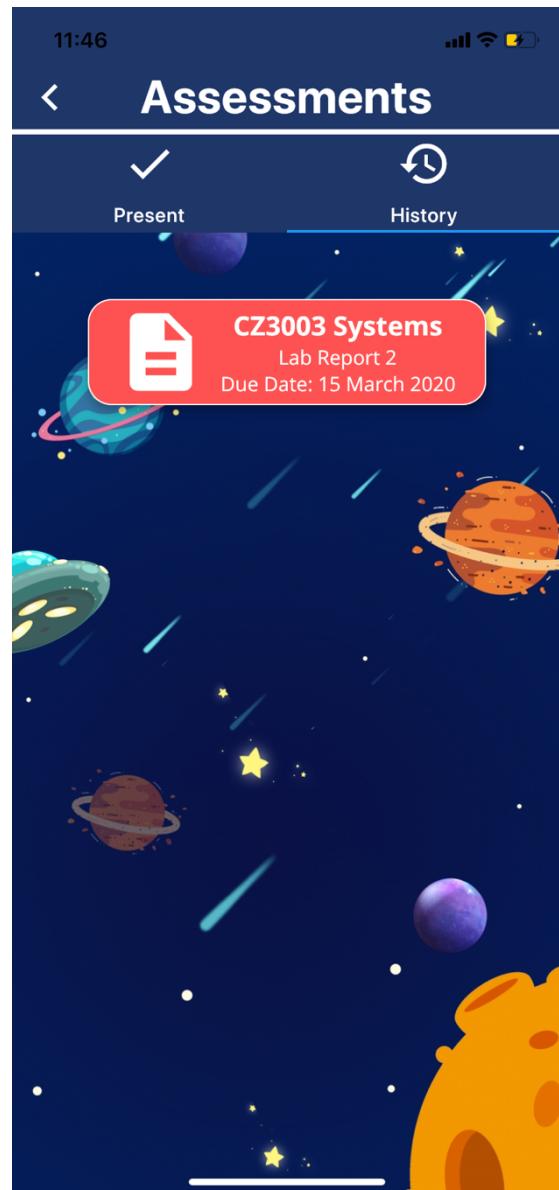
Challenge Complete (Compete Mode)



Existing Challenge Tie



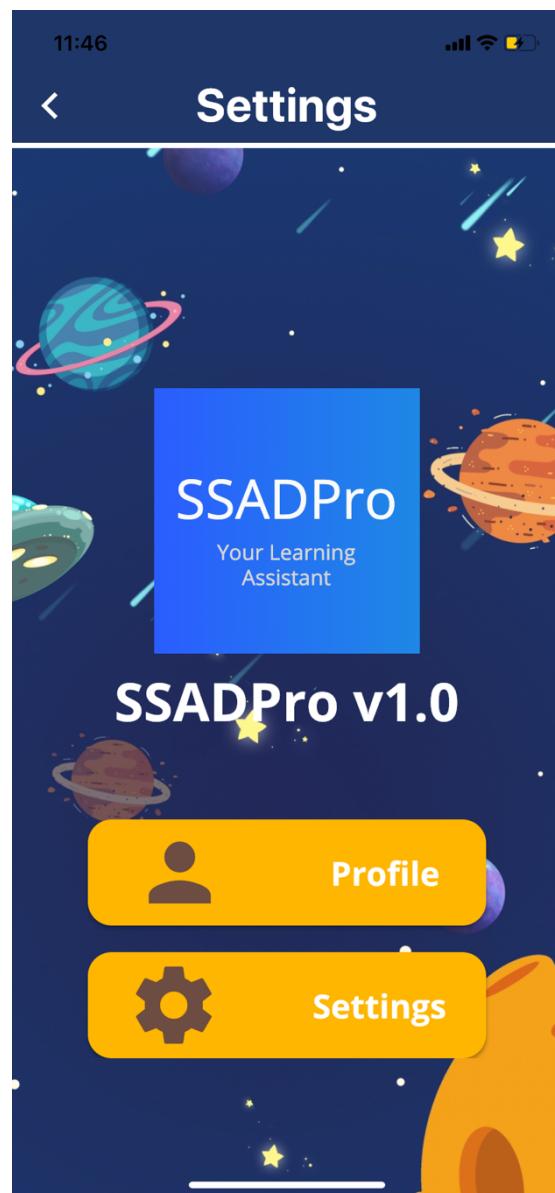
View Assessments (Student Mode)



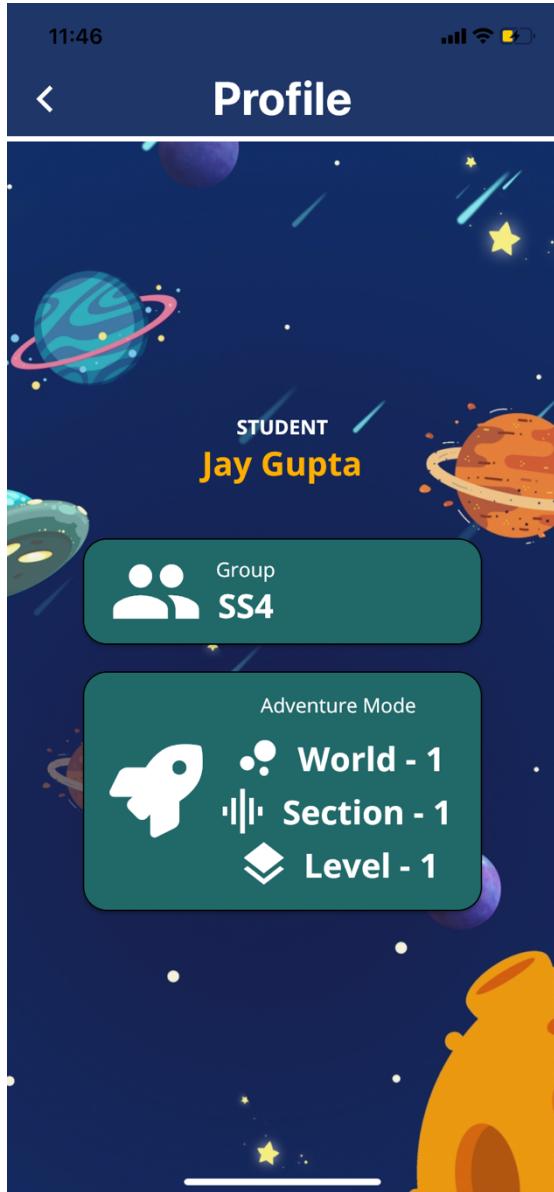
View Assessments (Student Mode)

Rank	Name	Point
1	Gaurav	Point: 6
2	Arjun Vaish	Point: 3
3	Koh Chean	Point: 3
4	Yap Jay	Point: 3
5	Mac Milin	Point: 3
6	Yusuf	Point: 3
7	SSAD Project	Point: 3
8	Divyesh	Point: 2

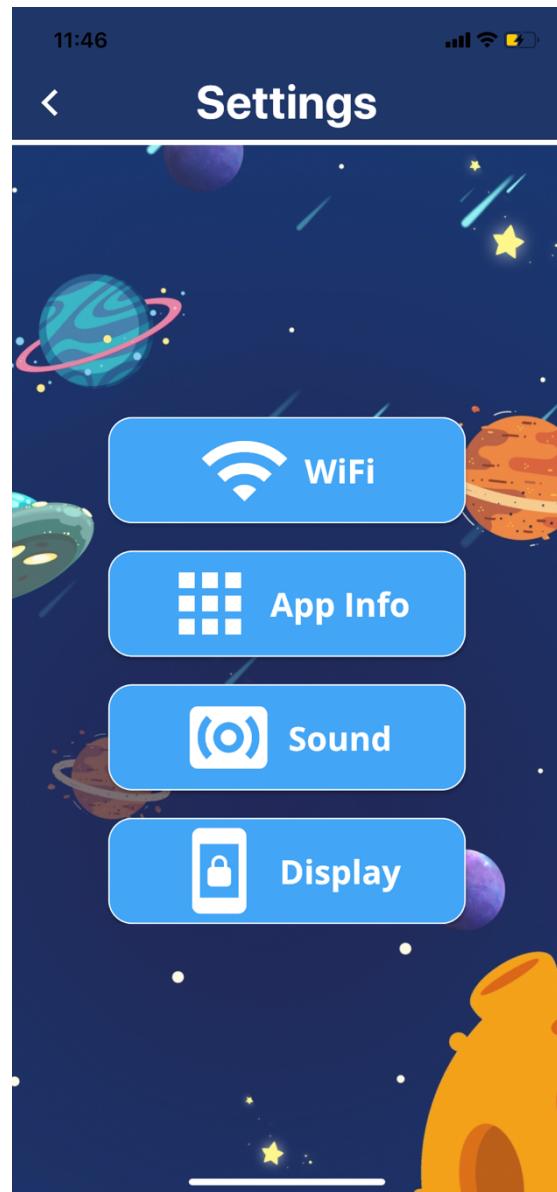
Leaderboard (Student Mode)



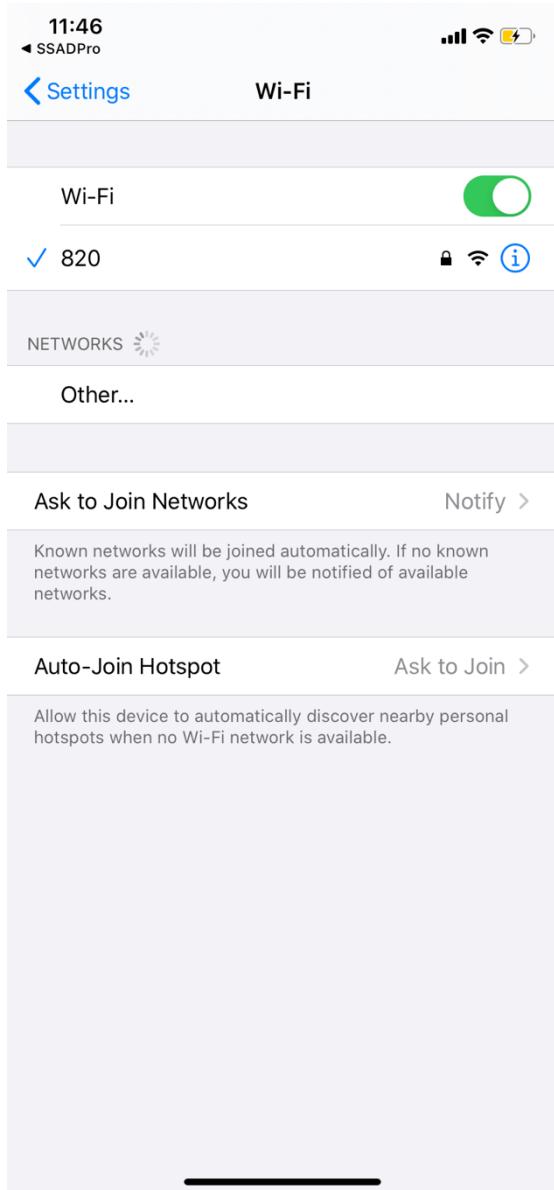
Settings



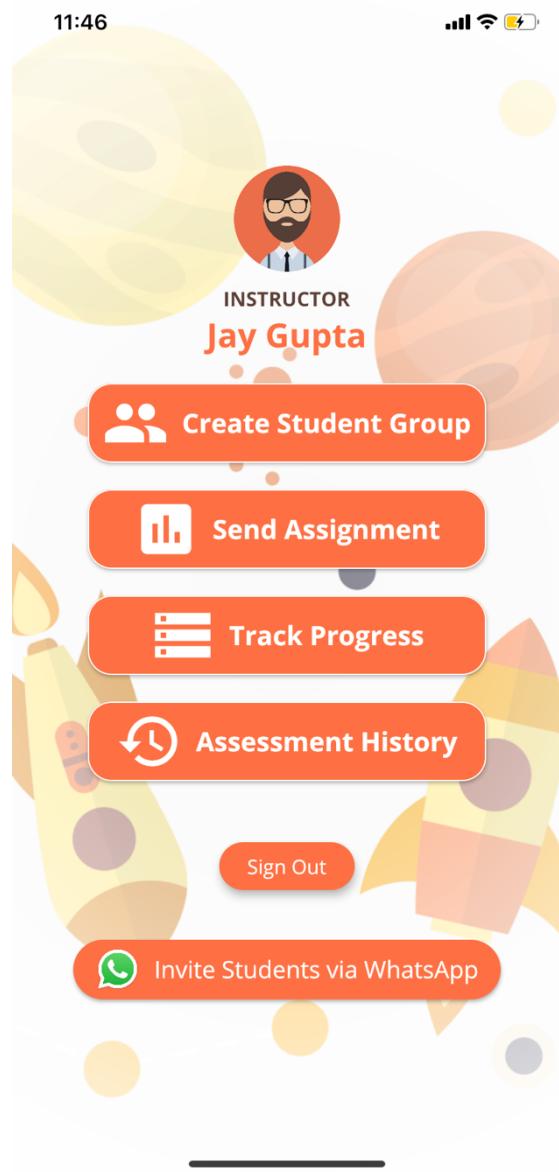
Profile Page



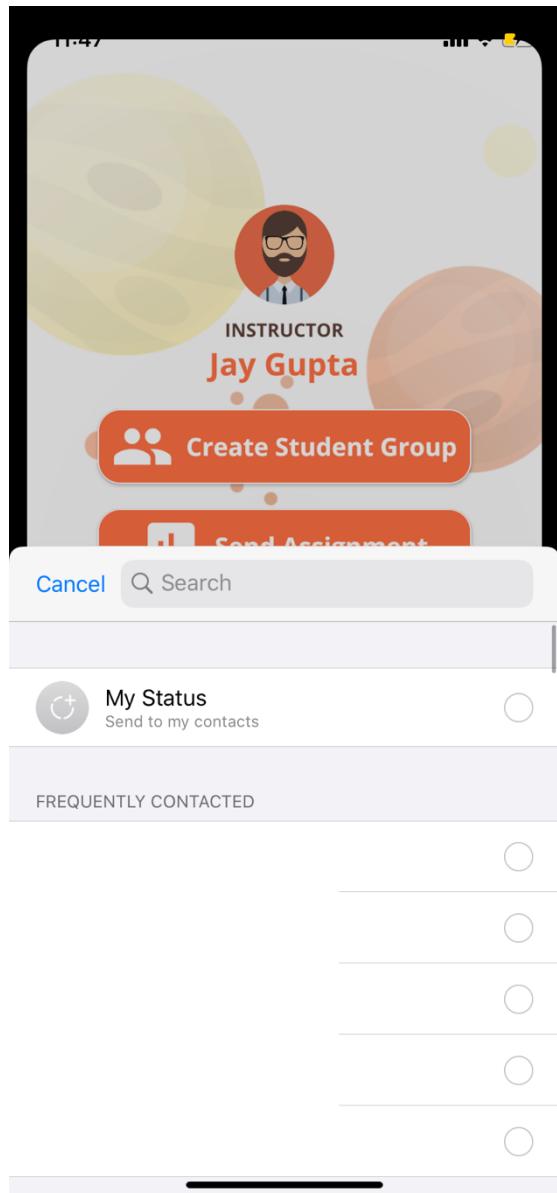
Application Settings



WiFi Settings – App Settings



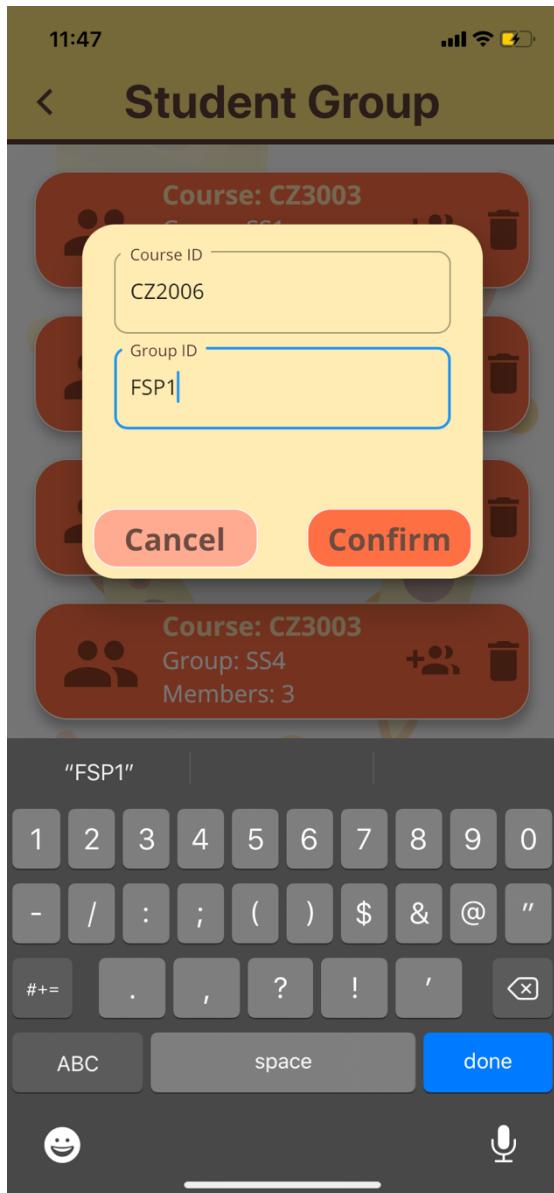
Dashboard (Instructor)



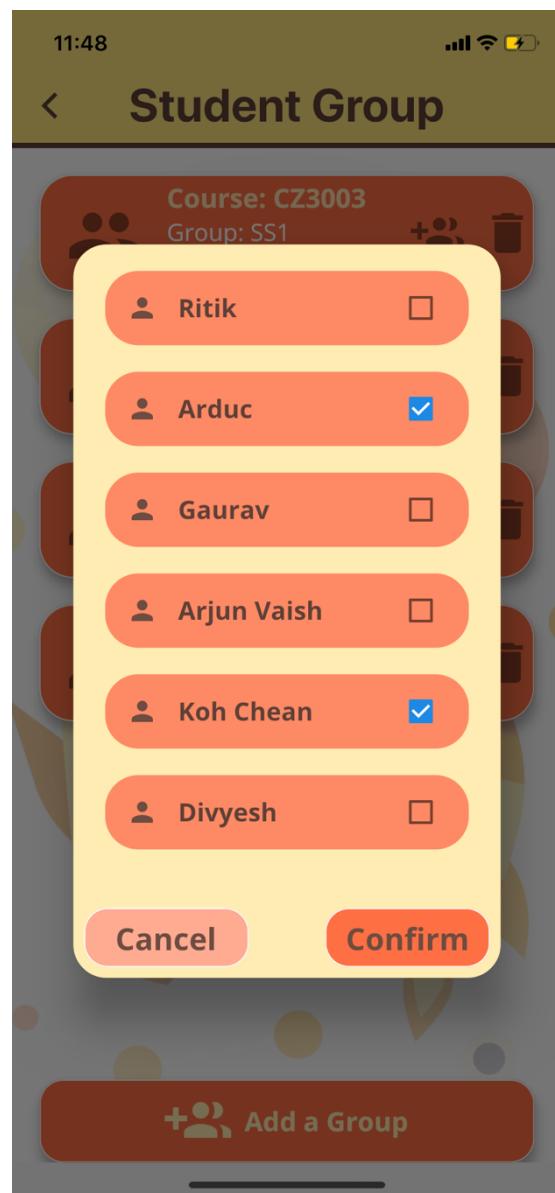
Invite Students via WhatsApp



Create Student Group (Instructor)



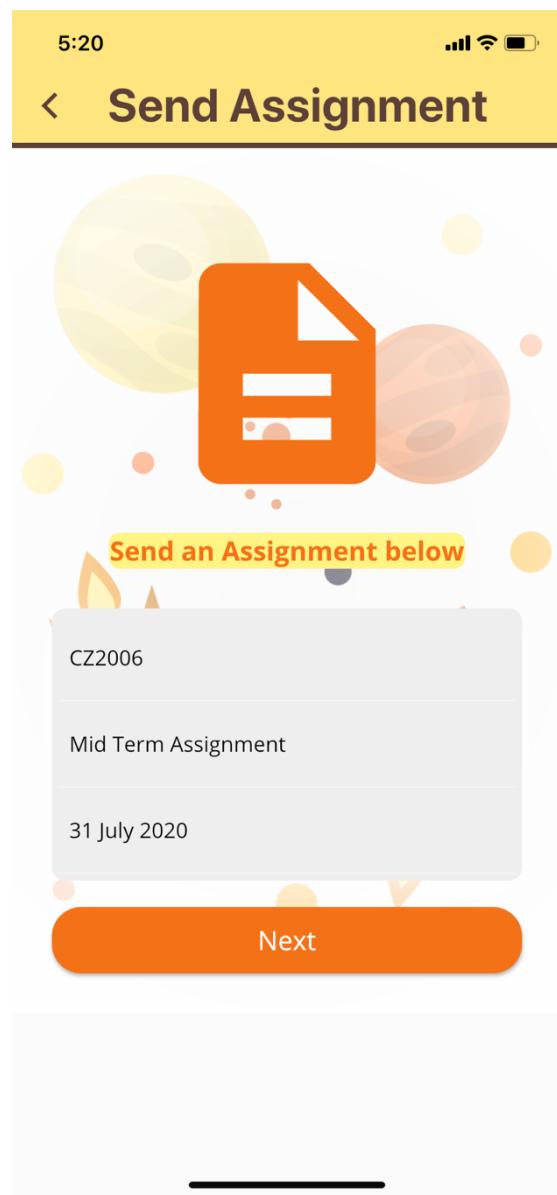
Create Student Group (Instructor)



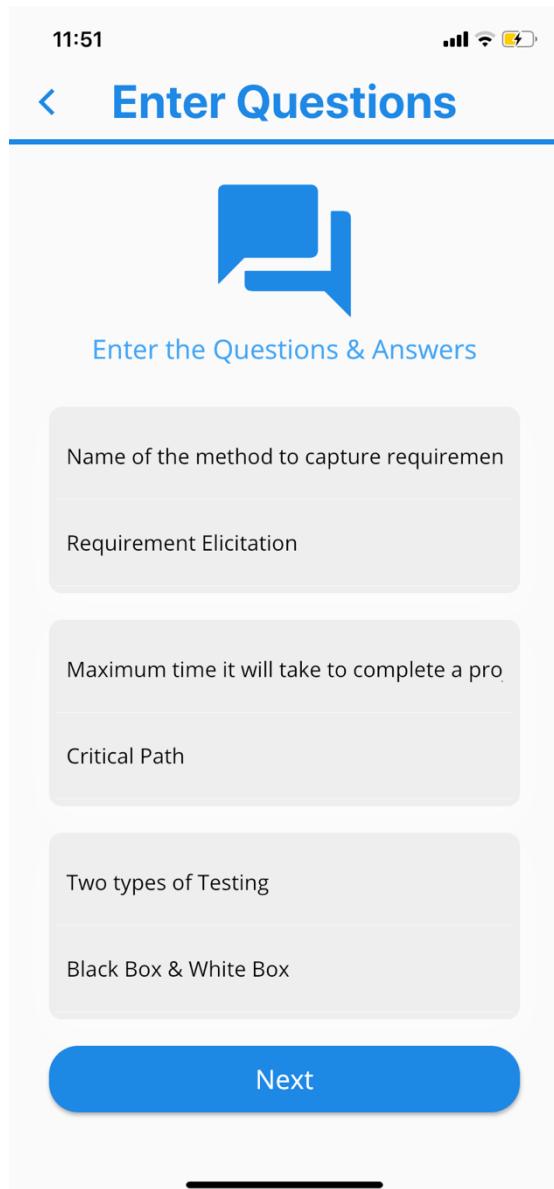
Create Student Group (Instructor)



Delete Student Group (Instructor)



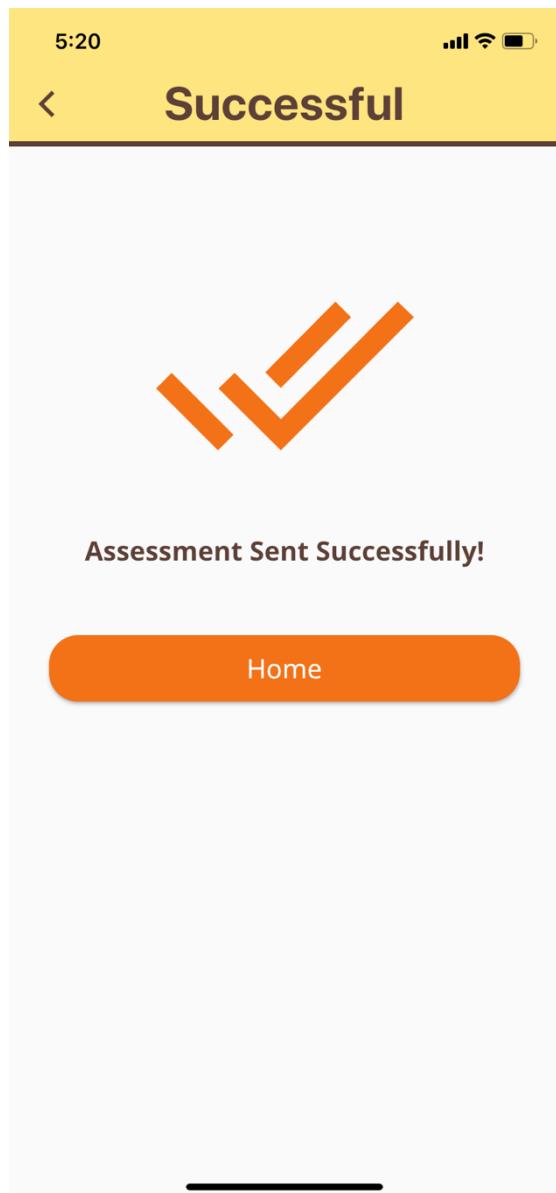
Send Assignment (Instructor)



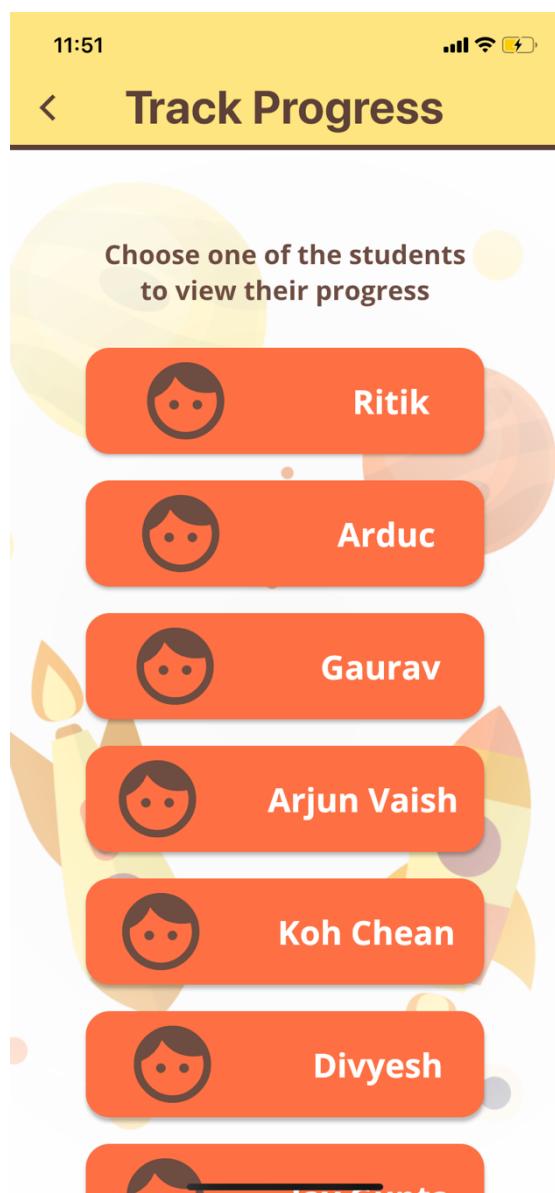
Assignments – Questions (Instructor)



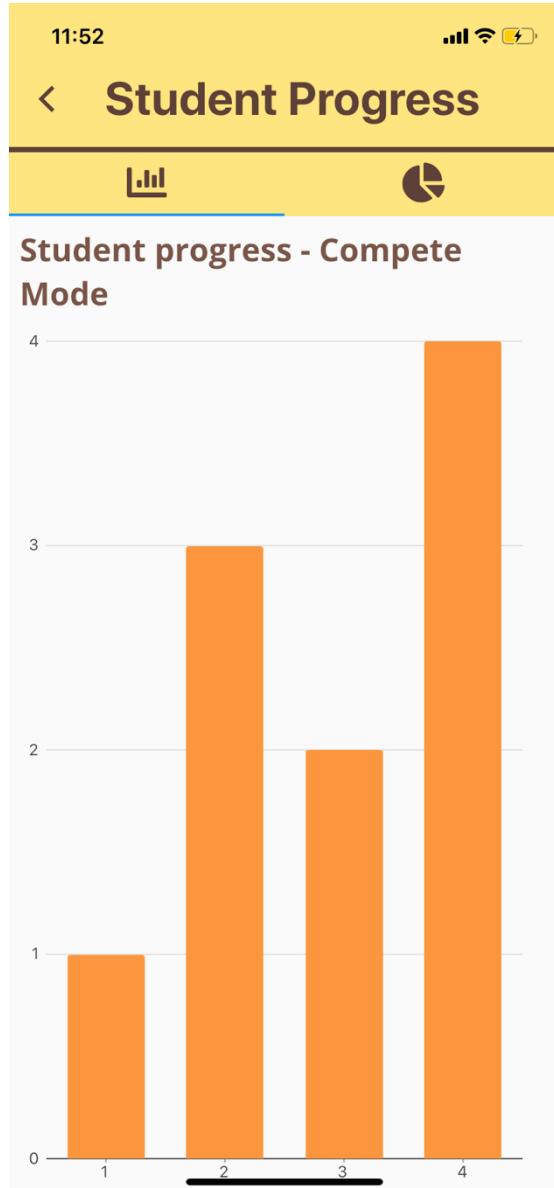
Assignments – Group (Instructor)



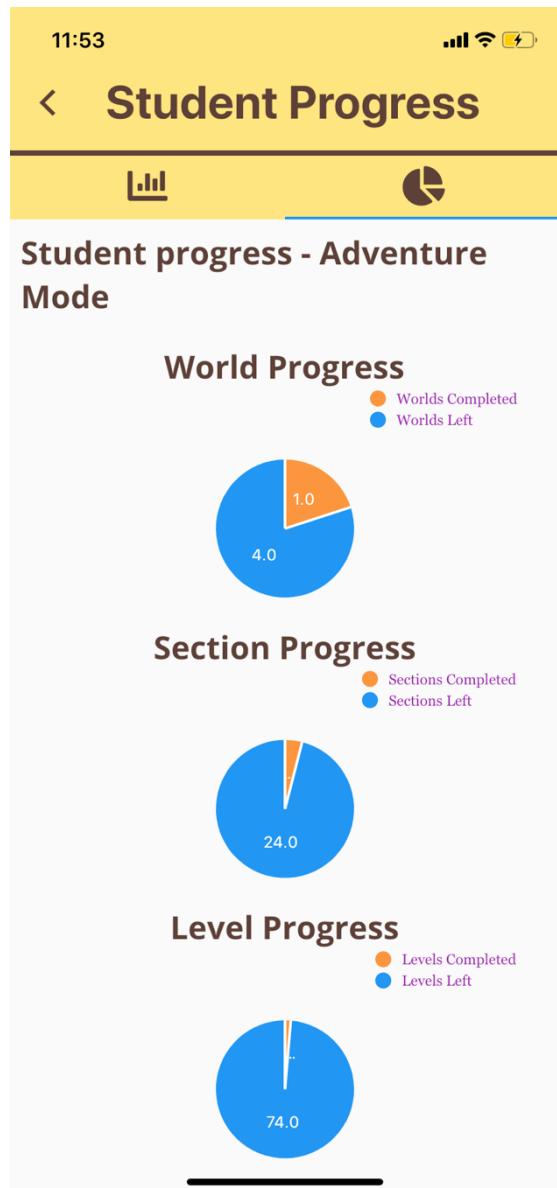
Assignment Sent (Instructor)



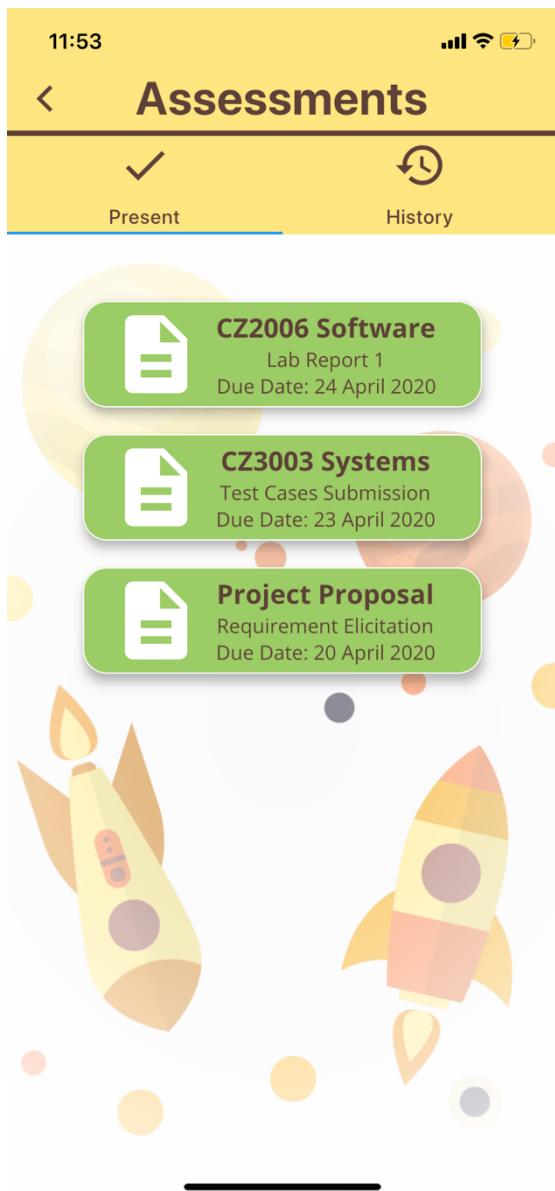
Track Progress (Instructor)



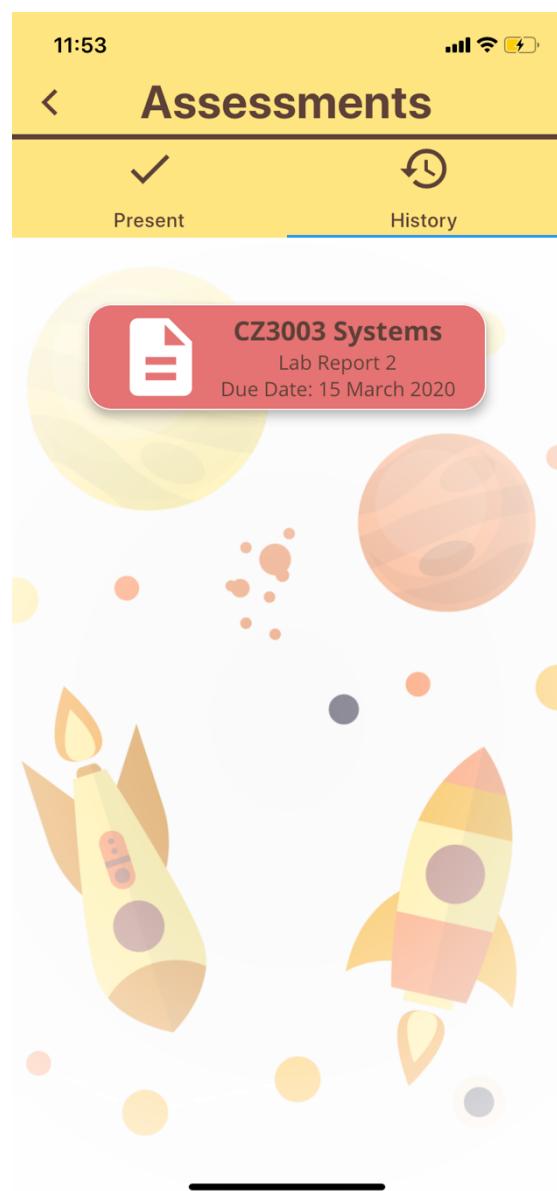
Track Progress (Instructor)



Track Progress (Instructor)



Track Assessments (Instructor)



Track Assessments (Instructor)

4.2. Software Interfaces

The application has the following software dependencies, indicated in the *pubspec.yaml* file:

- environment:

```
  sdk: ">=2.1.0 <3.0.0"  
  sdk: ">=2.3.0 <3.0.0"
```

- dependencies:

```
  flutter_date_picker: "^0.1.8"  
  sdk: flutter  
  cupertino_icons: ^0.1.2  
  firebase_auth: ^0.15.5+3  
  google_sign_in: ^4.2.0  
  simple_animations: ^1.3.11  
  path_provider: ^1.6.5  
  cloud_firestore: ^0.13.4+2  
  provider: ^3.1.0  
  font_awesome_flutter: ^8.2.0  
  charts_flutter: ^0.6.0  
  volume: ^0.1.0  
  url_launcher: ^5.4.2  
  app_settings: 3.0.1  
  share: ^0.6.3+6
```

- fonts:

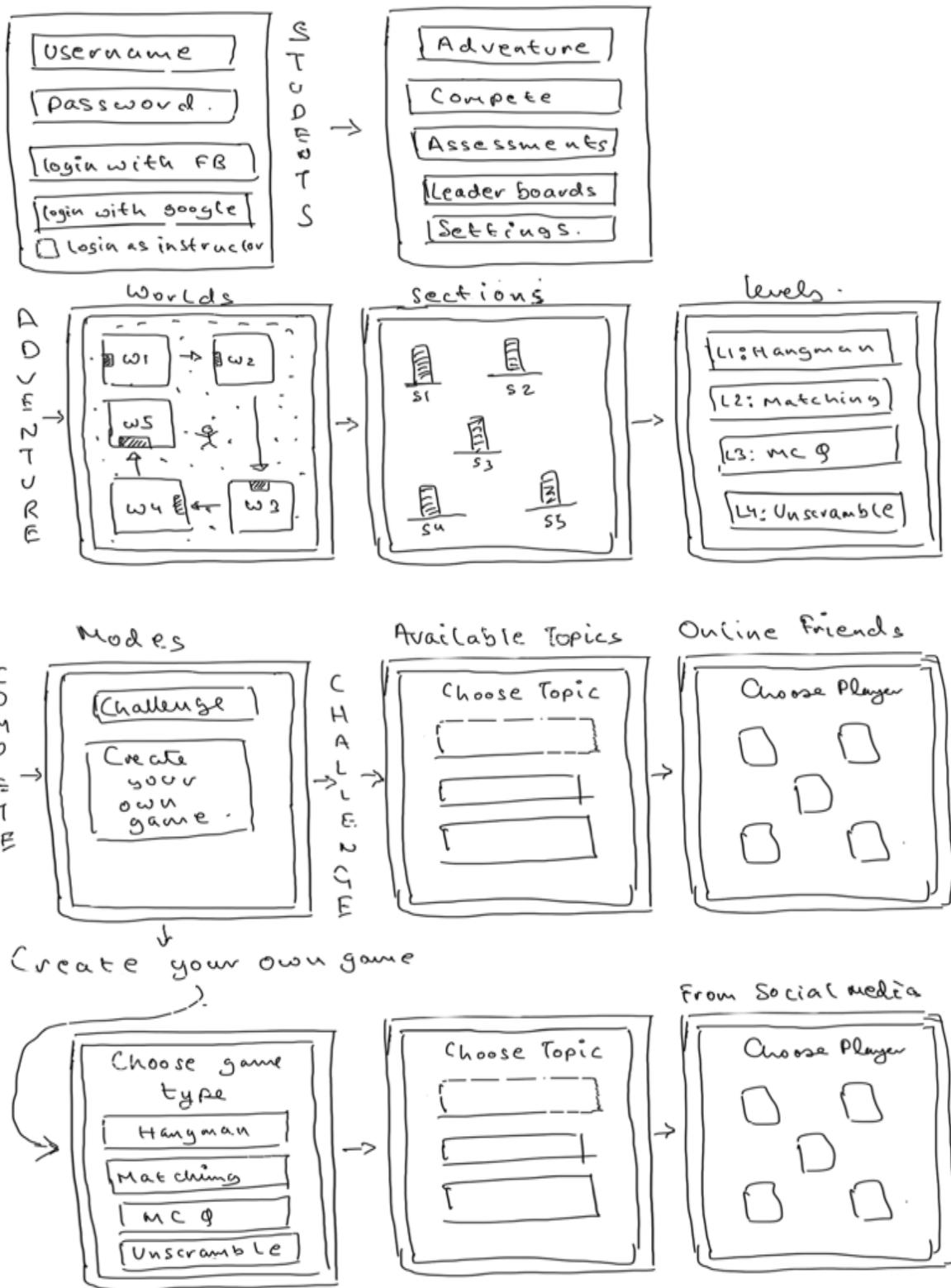
```
- family: Open Sans  
  fonts:  
    - asset: assets/fonts/OpenSans-Bold.ttf  
      weight: 700  
    - asset: assets/fonts/OpenSans-ExtraBold.ttf  
      weight: 800  
    - asset: assets/fonts/OpenSans-Light.ttf
```

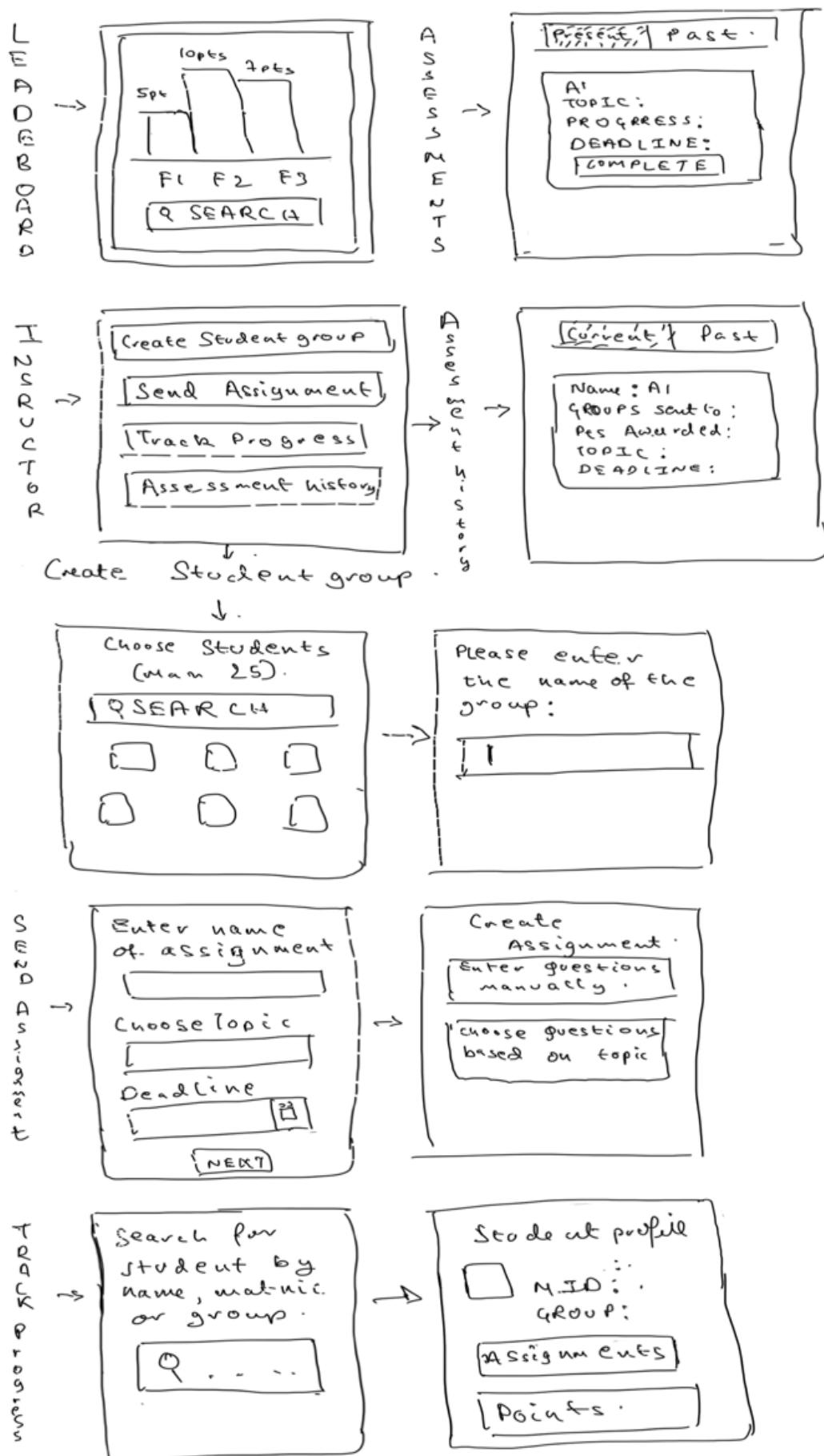
```
    weight: 300
- asset: assets/fonts/OpenSans-Regular.ttf
    weight: 400
```

The above dependencies need to be installed for successful execution of the code.

The data storage has been done using both local storage and Google Firebase Database Server. This has been achieved by using ***path_provider*** which creates txt files on the local storage of the user and reads/updates this same txt file throughout and ***firebase_auth*** which communicates with the Firebase Server.

5. Initial UI Mockup





6. Functional Requirements

6.1. System Functionality to be Performed

Login

1. The game must allow the user to be a player.
 1. The game shall allow the user to create an account through email or by linking to his/her Facebook or Google account.
 1. The system shall send a verification email with a temporary password.
 2. The system shall allow the user to change the default password.
 2. The game shall allow the player to select their desired avatar from a set of default avatars.

Game Modes

1. The application must have 2 main game modes, an assessment tab, a leader board and settings access on the main menu.
 - 1.1. In the first mode, which is the '**adventure mode**', the system shall allow the user to set out on a learning adventure on his own.
 - 1.2. In the second mode, which is the '**compete mode**', the system shall allow the user to challenge other players.
 - 1.3. In the '**Settings**' option, the system shall allow players to change the basic settings of the game and view their profile.
 - 1.4. The **leader boards** must display the points of the user and other players.
2. The game shall allow the player to choose the mode.
 - 2.1. Users must press the "Adventure" button on the main menu to access the Adventure mode of the game.
 - 2.2. The system must display a world map for the player to explore.

Game Modes - Adventure Mode

1. The adventure mode shall have 3 components in hierarchical order: worlds, sections and levels.
 - 1.1. The worlds represent different phases of the Software Development Life Cycle.
 - 1.2. Each world shall have sections representing specific topics of each SDLC phase.
 - 1.3. Each section shall have levels with questions of increasing difficulties.

Worlds

1. The game shall keep '**World 1**' unlocked and all the other worlds locked for the user by default.
2. The player must be able to explore only the unlocked worlds in the adventure-world map with the help of their avatar.
 - 2.1. The game shall allow the player to enter an unlocked world by moving their avatar to its door.
 - 2.1.1. The system shall allow the player to move their avatar with arrow keys.
 - 2.1.2. The system shall display a message when the user reaches a particular unlocked world's door: "Press Enter to explore".
 - 2.1.3. Upon pressing the enter key, the game shall allow the player's avatar to explore the sections of the world.
 - 2.2. The system shall not allow a player to enter a locked world.
 - 2.2.1. The system shall display a message when the user reaches a locked world's door: "This world is locked. Please complete the **world quiz** to unlock it!"
 - 2.2.2. The system shall allow the player to exit a world from the same door.
3. The game shall only unlock a world after the player passes the '**World Quiz**'.
 - 3.1. The game must set a quiz at the end of each world.
 - 3.1.1. The system shall display random questions based on the topics of the current world.
 - 3.1.1.1. The game must display 10 MCQ type questions.

3.1.2. After completion of the quiz, the system shall allow the player to proceed to the next world.

3.1.2.1. The game must allow the player to proceed only after the player answers 7 questions correctly.

3.2. The game shall allow the player to take the '**World Quiz**' anytime.

3.2.1. The game shall allow the player to attempt the quiz even if the player hasn't completed all sections.

3.3. If the player fails the quiz, the game must keep the next world locked.

Sections

1. When the player enters the world, the system shall display a section map of that world.

1.1. The player can navigate the section map through their avatar.

1.2. The game shall allow the player to enter any section that they desire in a particular world.

1.2.1. The game shall keep all the sections unlocked by default.

2. Once a player enters a particular section, the system must display a tutorial video on a particular topic.

2.1. The system must give the option to the player to either watch the video or skip it.

2.2. The system displays a side pane (menu) with information related to the levels in that section.

Levels

1. The system shall store a database of questions.

1.1. The system shall sort the questions by different phases of the SDLC life cycle and topics within each phase.

2. The game shall consist of 4 Levels in each section.

2.1. The system shall create a unique mini-game for each level.

2.2. The system shall extract questions from the database of questions.

- 2.3. The system shall allow users to access the levels as per their wish.
3. The game shall create a mini-game for each level.
 - 3.1. The system must use the following formats to test the player:
 - 3.1.1. Hangman - Fill in the blanks
 - 3.1.2. Matching - Question/Answer
 - 3.1.3. MCQ Quiz - Question/Answer
 - 3.1.4. Unscramble - Fill in the blanks
 - 3.2. For Hangman, the game shall accept one-word answers.
 - 3.2.1. The system shall allow the player to guess the letters.
 - 3.2.2. The system shall allow a maximum of 4 letters.
 - 3.2.3. The system shall also allow the player to guess the entire word.
 - 3.2.4. For Matching, the user has to match a question to its answer.
 - 3.2.4.1. The system shall display problem statements in one column and answers in another column.
 - 3.2.4.2. The system must allow the player to select a problem statement and its answer consecutively.
 - 3.2.4.3. The system must save the number of correct matches.
 - 3.2.5. For MCQ quiz, the game shall display 4 options for each question.
 - 3.2.5.1. The system must allow the player to select only one option for each question.
 - 3.2.5.2. The system must save the number of correct answers.
 - 3.2.6. For Unscramble, the user has to solve a jumbled word for the problem statement.
 - 3.2.6.1. The system must allow the user to rearrange the letters to form the answer.
 - 3.2.6.2. The system shall provide hints.
 4. The system shall award a fixed number of total 40 points for each section.
 - 4.1. The system shall allocate 2 points for each question, irrespective of the minigame they are playing.
 - 4.1.1. Therefore, the system must divide a total of 20 questions into 4 mini games.

- 4.1.1.1. In Hangman, the game must create 5 rounds with one question each round.
 - 4.1.1.2. In Matching, the game must create 1 round with 5 questions.
 - 4.1.1.3. In MCQ quiz, the game must create 5 rounds with one question each round.
 - 4.1.1.4. In Unscramble, the game must create 5 rounds with one question each round.
 - 4.1.1.5. The system shall award a maximum of 10 points for each minigame.
- 4.2. The user will be awarded partial points based on the number of questions they get correct.
 - 4.3. The system shall not take into account the time factor in points calculation.

Leader board

1. Throughout the learning experience, the system will have a button in the main menu.
 - 1.1. The system will display the player's social media friends.
 - 1.1.1. The system will display each friend's points.
 - 1.2. The system will also display other players who are online.
 - 1.3. The system calculates leader board by taking the sum of points earned in both adventure mode and the challenge mode.

Game Modes - Compete Mode

1. The game will allow the player to choose 'Compete' mode in the main menu.
 - 1.1. After selecting the 'Compete' mode, user must choose between 'Challenge' option or 'Create your own Test' option.
 - 1.1.1. On choosing 'Challenge', the system sets up a random game between the two players.
 - 1.1.2. The system will allow the initiating player to choose the topic of the challenge.
 - 1.1.3. The system will allow the player to search for a desired friend.

- 1.1.4. The system will create game-style questions using the database of questions.
- 1.1.5. The system will calculate total points earned for a player using specific rules.
 - 1.1.5.1. The player who completes the mini game faster gets 15 points.
 - 1.1.5.2. The player who completes the game slower gets 5 points.
2. The system will ask the player to challenge another player with a custom game using the "**Create your own game**" mode.
 - 2.1. This mode provides the challenging user with the flexibility of choosing the mini-game type. (Match, Hangman, MCQ or Unscramble).
 - 2.2. Subsequent rules will be the same as the 'Challenge' mode.

Assessments Tab for Students

1. The system shall allow the players to access the assignments sent by the instructors in their Assessment tabs.
 - 1.1. The system shall display a 'Track Progress' option.
 - 1.2. The system shall allow the player to check their progress.
 - 1.3. The system shall display the strengths and weaknesses of the player.

Instructors' Dashboard

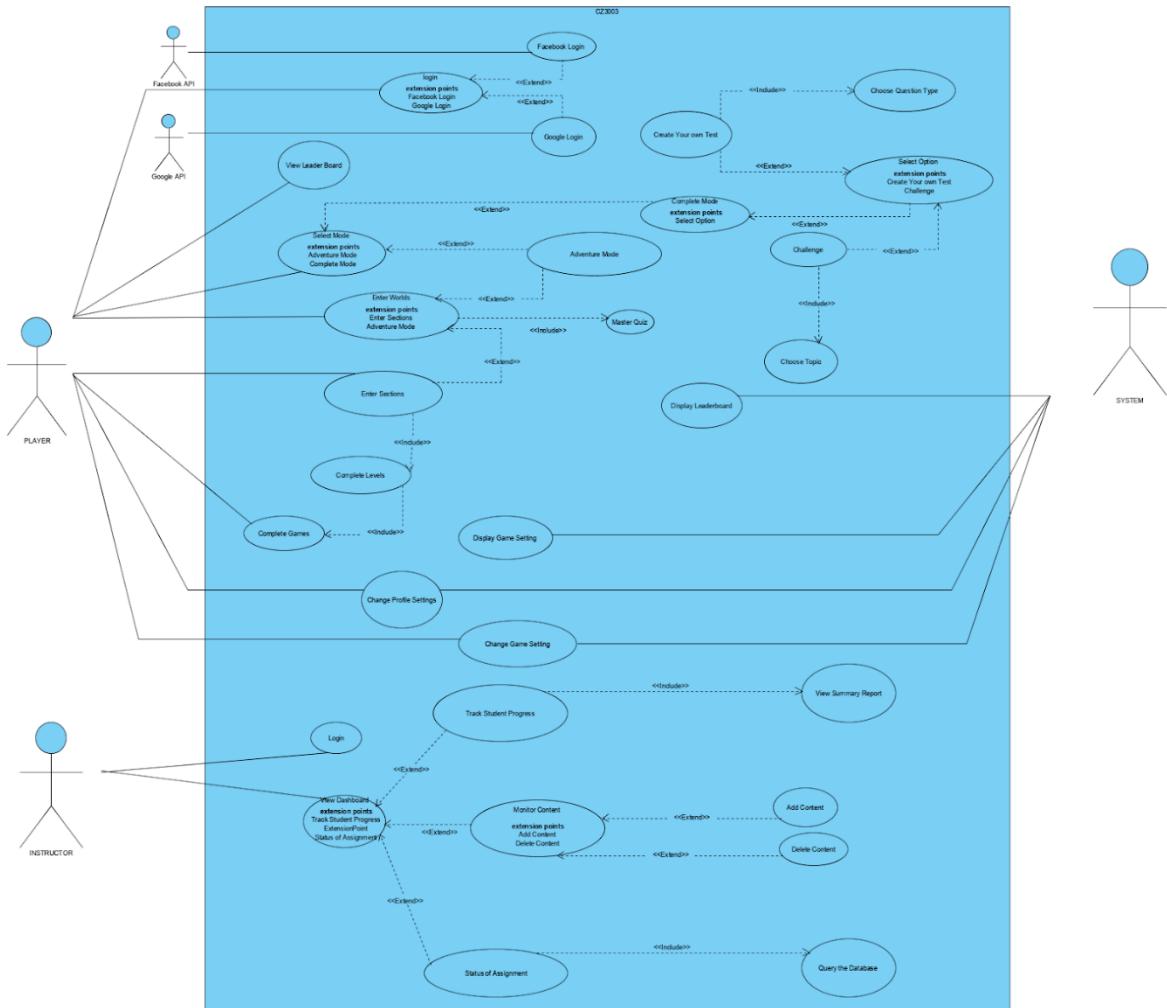
1. The system must provide a separate login entry for admins (tutors/instructors).
 - 1.1. There will be a checkbox on the login screen which says 'Are you an instructor?'
 - 1.2. The system shall allow the admin to login in the same way as students.
2. The system shall allow the instructor to create groups of students.
 - 2.1. The system shall allow the instructor to view a list of all students enrolled.
 - 2.2. On clicking 'Create Student Group' option, the system shall allow the user to select a maximum of 25 students.
 - 2.3. The system shall allow the user to name the group.
3. The system shall create a dashboard for instructors with three options.
 - 3.1. The system shall have a '**Send Assignment**' option.

- 3.1.1. The system shall allow the user to create assignments.
 - 3.1.1.1. The system must allow the user to select number of questions.
 - 3.1.1.2. The system shall allow the user to choose questions from the database of questions.
 - 3.1.1.3. The system shall allow the user to set an assignment with MCQ type questions only.
- 3.1.2. The system shall allow the user to send assignments.
 - 3.1.2.1. The system shall allow the user to send the assignment to a particular group of students.
 - 3.1.2.1.1. The system shall display the list of all student groups.
 - 3.1.2.1.2. The system shall allow the user to select one group minimum.
 - 3.1.2.1.3. The system shall also allow the user to select multiple groups.
- 3.1.3. The system shall allow the user to set a deadline for the assignment.
- 3.2. The system shall have a '**Track progress**' option.
 - 3.2.1. The system shall display the list of student groups.
 - 3.2.2. The system shall allow the user to select a particular student in a group.
 - 3.2.3. The system shall display the progress of the student.
 - 3.2.3.1. The system shall calculate progress in terms of number of assignments completed and its accuracy.
 - 3.2.3.2. The system shall display a graph with statistics.
 - 3.2.3.2.1. The system shall show the strengths and weaknesses of the student.
- 3.3. The system shall have an '**Assessment History**' option.
 - 3.3.1. The system shall display all the past assignments sent to students.
 - 3.3.2. The system shall show the completion status of each assignment.
 - 3.3.2.1. The system shall mark an assignment if all students have completed the assignment.
 - 3.3.2.2. If incomplete, the system shall allow the user to send reminder notifications.

Game Settings

1. The system shall display a '**Settings**' menu on the main menu.
 - 1.1. The system shall allow the player to adjust the volume of the music.
 - 1.2. The system shall allow the player to adjust the volume of the sounds.
2. The system shall allow the player to view their profile information through the Profile icon.
 - 2.1. The system shall display the username, total points, and account connection status (Facebook and Google).
 - 2.2. The system shall allow the player to connect their Facebook or Google accounts.
 - 2.3. The system shall allow the player to disconnect from their Facebook or Google accounts.

6.2. Use-Case Diagram



6.3. Use-Case Descriptions

1. Login

Use Case ID	1		
Use Case Name	Login		
Created By	Jay Yap	Last Updated By	Jay Yap
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User/Instructor (Initiating Actor), Facebook API, Google API
Description	Before accessing the content in this application, the user must log into their account. The user will either log in through their Facebook account or Google account.
Preconditions	<ol style="list-style-type: none"> 1. The User must have an existing Facebook or Google account. 2. The user must be connected to the internet. 3. The system must be active.
Postconditions	Access must be granted and the user shall access the content in the application.
Priority	High
Frequency of Use	Once per lifetime
Flow of Events	<ol style="list-style-type: none"> 1. System prompts the user to log in using a Facebook account or a Google account. 2. The User must be prompted to enter email and password for verification through the Social Media website. <ol style="list-style-type: none"> 2.1 The System validates the account through the Facebook API or the Google API based on the User's selection.

	3. Account is verified and the user is granted access to the application.
Alternative Flows	<p>AF-S3: If the User types in a wrong account.</p> <ol style="list-style-type: none"> 1. The application must display an error message “Invalid Email/Password. Please try again.” 2. The application returns to Step 1.
Exceptions	<p>EX1: If the user types in a wrong account more than three times.</p> <ol style="list-style-type: none"> 1. The application displays an error message “Invalid Email/Password. Please try again after 5 minutes”
Includes	Retrieve User information through Google or Facebook API
Special Requirements	The APIs must be reliable. It must grant access to the requested account within 5 seconds.
Assumptions	<ol style="list-style-type: none"> 1. The user is connected to the internet.
Notes and Issues	TBD

2. View Leader board

Use Case ID	2		
Use Case Name	View Leaderboard		
Created By	Jay Yap	Last Updated By	Jay Yap
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor)
Description	There will be a leaderboard recording the performance of different players. When the user wants to check the leaderboard, the user must select the “View Leaderboard” option and the leaderboard will be displayed.
Preconditions	<ol style="list-style-type: none"> 1. The user must be logged into his/her account. 2. The user must be connected to the internet. 3. The system must be active.
Postconditions	The leaderboard must be displayed.
Priority	Low
Frequency of Use	Once per Lifetime.
Flow of Events	<ol style="list-style-type: none"> 1. After successful login, user must select “View Leaderboard” from the main menu options. 2. Application retrieves leaderboard data from the leaderboard database of the application. 3. The Leaderboard data will be displayed.
Alternative Flows	N/A
Exceptions	N/A
Includes	Retrieve leaderboard data from Leaderboard Database

Special Requirements	The leaderboard consisting of correct data must be shown within 5 second upon request.
Assumptions	1. The user is connected to the internet.
Notes and Issues	TBD

3. Change Profile Settings

Use Case ID	3		
Use Case Name	Change Profile Settings		
Created By	Jay Yap	Last Updated By	Jay Yap
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor), System
Description	The settings of a user's profile can be adjusted by the user. The settings include Username and Profile Picture. When the user wants to change the settings of their profile, they must select the "Profile Setting" option to do so.
Preconditions	<ol style="list-style-type: none"> 1. The user must be logged into his/her account. 2. The user must be connected to the internet. 3. The system must be active.
Postconditions	Settings of their profile must be changed.
Priority	Low
Frequency of Use	Once per Lifetime
Flow of Events	<ol style="list-style-type: none"> 1. The User must select the setting option from the main menu. 2. The User must select profile settings. <ol style="list-style-type: none"> 2.1 The User makes changes to the profile settings. 2.2 The User must save the altered settings. 3. Settings of the profile will be saved in the system.
Alternative Flows	AF-S3 if the settings are not saved: <ol style="list-style-type: none"> 1. Setting changes made would not be applied.
Exceptions	TBD

Includes	Change Profile Settings Use-Case
Special Requirements	Profile settings must be applied within 2 seconds after saving.
Assumptions	1. The user is connected to the internet.
Notes and Issues	TBD

4. Change Game Settings

Use Case ID	4		
Use Case Name	Change Game Settings		
Created By	Jay Yap	Last Updated By	Jay Yap
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor), System
Description	The settings of the game can be adjusted by the user. The settings include: Volume Adjustment and Brightness Adjustment. When the user wants to change the settings of the game, they can select the “Game Setting” option to do so.
Preconditions	<ol style="list-style-type: none"> 1. The user has already logged into their account. 2. The user must be connected to the internet. 3. The system must be active.
Postconditions	Settings of the game must be changed.
Priority	Low
Frequency of Use	Once per Lifetime
Flow of Events	<ol style="list-style-type: none"> 1. The User must select the setting option from the main menu. 2. Application displays the types of settings that can be changed. <ol style="list-style-type: none"> 2.1 The User must make changes to the game settings. 2.2 The User must save the altered settings. 3. Settings of the game will be saved in the system.
Alternative Flows	<p>AF-S3: If the settings are not saved:</p> <ol style="list-style-type: none"> 1. Settings changed will not be saved.

Exceptions	TBD
Includes	Alter Game settings
Special Requirements	Game settings must be applied within 2 seconds after saving.
Assumptions	<ol style="list-style-type: none">1. Database server must be active and accessible all the time.2. Users' connections must be stable.3. System's connections must be stable.
Notes and Issues	TBD

5. Select Mode

Use Case ID	5		
Use Case Name	Select Mode		
Created By	Koh Zhuang Chean	Last Updated By	Jay Yap
Date Created	16/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor), System
Description	The user is required to select a mode (either Adventure or Compete Mode) to play along prior to entering a world.
Preconditions	<ol style="list-style-type: none"> 1. The user must have an account. 2. The user must be connected to the internet. 3. System must be active. 4. The user must have logged in successfully.
Postconditions	User shall enter the selected mode.
Priority	High
Frequency of use	Twice per lifetime
Flow of events	<ol style="list-style-type: none"> 1. On successful login, the system must prompt two modes for the user to choose. 2. The user must choose Adventure mode. 3. The system shows the worlds of the Adventure Mode according to the user's progress. 4. The user starts learning.
Alternative flows	<p>AF-S2: If the user chooses Compete mode</p> <ol style="list-style-type: none"> 1. The system directs the user to the Compete Mode. 2. The system shows the worlds of the Compete Mode according to the user's progress. 3. The user is asked to select one option between "Challenge" and "Create Your Own Test".

	<p>4. The user must choose according to their preference.</p>
Exceptions	<p>EX1: The user decides to exit the game prior to choosing the modes.</p> <ol style="list-style-type: none"> 1. The system must ask for confirmation from the user. 2. If Yes is chosen, the system must close the game.
Includes	<ol style="list-style-type: none"> 1. Adventure Mode 2. Compete Mode
Special Requirements	<ol style="list-style-type: none"> 1. Application system must be well connected to the database of questions and well linked to the user's social media account. 2. Worlds must be displayed within 2 seconds. 3. Users can go back to selecting the modes by clicking a "Back Button".
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible all the time. 2. Users' connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

6. Enter World

Use Case ID	6		
Use Case Name	Enter World		
Created By	Koh Zhuang Chean	Last Updated By	Jay Yap
Date Created	16/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating factor), System
Description	The user can select a world that is unlocked to enter.
Preconditions	<ol style="list-style-type: none"> 1. The user must have a valid account. 2. The user must be connected to the internet. 3. The system must be active. 4. The user must have logged in successfully. 5. The user must have selected the Adventure Mode.
Postconditions	User shall enter the selected world.
Priority	High
Frequency of use	Twice per lifetime.
Flow of events	<ol style="list-style-type: none"> 1. After the user has chosen Adventure Mode, the system shows all the worlds. 2. The user must only select the unlocked worlds. 3. The user enters the selected world.
Alternative flows	N/A
Exceptions	<p>EX2: The user decides to exit the game prior to choosing the worlds.</p> <ol style="list-style-type: none"> 1. The system must ask for confirmation from the user. 2. If Yes is chosen, the system closes the game.
Includes	N/A

Special Requirements	<ol style="list-style-type: none">1. Application system must be well connected to the database of questions and well linked to the user's social media account.2. Sections and mini games of the world that the user selected must be displayed within 2 seconds.3. Users can go back to selecting the worlds by simply clicking "Back Button".
Assumptions	<ol style="list-style-type: none">1. Database server must be active and accessible all the time.2. Users' connections must be stable.3. System's connections must be stable.
Notes and issues	TBD

7. Master Quiz

Use Case ID	7		
Use Case Name	Master Quiz		
Created By	Koh Zhuang Chean	Last Updated By	Koh Zhuang Chean
Date Created	16/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor), System
Description	The user can answer the quiz if the user deems, they are knowledgeable to pass the quiz.
Preconditions	<ol style="list-style-type: none"> 1. The user must have a valid account. 2. The user must be connected to the internet. 3. The system must be active. 4. The user must have logged in successfully. 5. The user must have selected the Adventure Mode. 6. The user must have entered one of the worlds.
Postconditions	<ol style="list-style-type: none"> 1. If the user has passed the quiz, the next world is unlocked. 2. Else, the user stays in the same world.
Priority	Low
Frequency of use	Once or twice per lifetime
Flow of events	<ol style="list-style-type: none"> 1. The user must select Master Quiz. 2. Relevant questions must be prompted for the user to answer. 3. The user answers all the questions on the quiz. 4. The user passes the quiz, the next world is unlocked.
Alternative flows	<p>AF-S4: if the user doesn't pass the quiz</p> <ol style="list-style-type: none"> 1. The user must stay in the same world and no new world is unlocked.

	<ol style="list-style-type: none"> 2. The system must recommend the user to revise the topics that the user failed to answer correctly.
Exceptions	<p>EX2: The user decides to exit the game prior to answering all the questions in the Master Quiz.</p> <ol style="list-style-type: none"> 1. The system must ask for confirmation from the user. 2. If Yes is chosen, the system must close the game. 3. The user is considered to fail the quiz.
Includes	N/A
Special Requirements	<ol style="list-style-type: none"> 1. Application system must be well connected to the database of questions and well linked to the user's social media account. 2. The questions must be displayed to the user within 1 seconds after the user enters the quiz. 3. The application must be able to determine whether the user passes the quiz or not within 2 seconds. 4. The user can go back to the world with sections and mini games by simply clicking the "Back Button".
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible all the time. 2. Users' connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

8. Enter Sections

Use Case ID	8		
Use Case Name	Enter Sections		
Created By	Koh Zhuang Chean	Last Updated By	Koh Zhuang Chean
Date Created	17/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor), System
Description	The user can choose to enter each section in the world and select a level/mini-game in the section.
Preconditions	The user must have entered one of the worlds.
Postconditions	<ol style="list-style-type: none"> 1. The user must have a valid account. 2. The user must be connected to the internet. 3. The system must be active. 4. The user must have logged in successfully. 5. The user must have selected the Adventure Mode. 6. The user must have entered one of the worlds.
Priority	High
Frequency of use	3 times per Lifetime
Flow of events	<ol style="list-style-type: none"> 1. After the user has entered a world, the system must display the list of sections. <ol style="list-style-type: none"> 1.1 The user must select one of the sections. 1.2 The levels and mini-games of that particular section must be displayed to the user. 1.3 The user shall choose either one of the levels or one of the mini games.
Alternative flows	N/A
Exceptions	EX1: The user decides to exit the game. <ol style="list-style-type: none"> 1. The system must ask for confirmation from the User.

	<p>2. If confirmed by the User, the system must keep save the User's progress and close the game.</p>
Includes	Complete Levels
Special Requirements	<p>1. Application system is well connected to the database of questions and well linked to the User's account.</p> <p>2. The content of the section must be displayed within 2 seconds after selecting that section.</p> <p>3. Users must be able to go back to the sections list by simply clicking the "Back Button".</p>
Assumptions	<p>1. Database server must be active and accessible all the time.</p> <p>2. Users' connections must be stable.</p> <p>3. System's connections must be stable.</p>
Notes and issues	TBD

9. Complete Levels

Use Case ID	9		
Use Case Name	Complete Levels		
Created By	Koh Zhuang Chean	Last Updated By	Koh Zhuang Chean
Date Created	17/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor), System
Description	The user will eventually complete a level or mini game bound with that section. Once all the levels and mini games are completed, the user is considered to have completed the section. Subsequently, the next Level is unlocked.
Preconditions	<ol style="list-style-type: none"> 1. The user must have a valid account. 2. The user must be connected to the internet. 3. The system must be active. 4. The user must have logged in successfully. 5. The user must have selected the Adventure Mode. 6. The user must have entered one of the worlds. 7. The user must have entered one of the sections of the world.
Postconditions	The user shall go to the next level of the section.
Priority	High
Frequency of use	5-10 times per Lifetime
Flow of events	<ol style="list-style-type: none"> 1. After the user has entered a section, they must choose one of the levels within that section. 2. At every level, the User must be prompted to answer the questions of the particular level. <ol style="list-style-type: none"> 2.1 Once more than 75% of the questions are answered correctly, the user is considered to have passed the level of that section.

	2.2 The subsequent Level must be unlocked by the System.
Alternative flows	<p>AF-S2.1: Less than 75% of the questions are answered correctly.</p> <ol style="list-style-type: none"> 1. The user must stay at the same level and no new level is unlocked. 2. The system must recommend the user to revise the topics that the user failed to answer correctly.
Exceptions	<p>EX1: The user decides to exit the game.</p> <ol style="list-style-type: none"> 1. The system must ask for confirmation from the User. 2. If confirmed by the User, the system must keep save the User's progress and close the game.
Includes	Complete Games
Special Requirements	<ol style="list-style-type: none"> 1. Application system is well connected to the database of questions and well linked to the User's account. 2. The content of the level must be displayed within 2 seconds after selecting that section. 3. Users must be able to go back to the level's list by simply clicking the "Back Button".
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible all the time. 2. Users' connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

10. Monitor Content

Use Case ID	10		
Use Case Name	Monitor Content		
Created By	Koh Zhuang Chean	Last Updated By	Koh Zhuang Chean
Date Created	17/02/2020	Date Last Updated	10/04/2020

Actor	Instructor (Initiating Actor), System
Description	Other than viewing the dashboard, the instructor can monitor the dashboard with ease. He/She can do three actions namely create groups of students, send assignments, set deadlines for the assignments.
Preconditions	<ol style="list-style-type: none"> 1. The instructor must have a valid account. 2. The instructor must be connected to the internet. 3. The system must be active. 4. The instructor must have logged in successfully. 5. The instructor must be at the View Dashboard page.
Postconditions	The instructor shall monitor the content of the dashboard.
Priority	High
Frequency of use	2 times per Lifetime
Flow of events	<ol style="list-style-type: none"> 1. Instructor must be at the page of viewing the dashboard. 2. The instructor then enters the monitor content page. 3. Three actions are available for the instructor namely create groups of students, send assignments, set deadlines for the assignments. 4. The instructor shall choose to implement one of them.
Alternative flows	N/A

Exceptions	<p>EX2: The instructor force quits the program before the page is shown.</p> <ol style="list-style-type: none"> 1. The system logs the instructor out and closes the entire application.
Includes	<p>AF-S2: If the instructor decides not to monitor the content.</p> <ol style="list-style-type: none"> 1. The instructor clicks “Back Button” to return to the “View Dashboard” page.
Special Requirements	<ol style="list-style-type: none"> 1. Application system is well connected to the database of questions and well linked to the user's social media account. 2. Contents of the dashboard must be displayed after the instructor has entered the page within 2 seconds. 3. Instructors can switch accounts by clicking “Logout Button”.
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible all the time. 2. Instructors' connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

11. Add Content

Use Case ID	11		
Use Case Name	Add Content		
Created By	Koh Zhuang Chean	Last Updated By	Koh Zhuang Chean
Date Created	17/02/2020	Date Last Updated	10/04/2020

Actor	Instructor (Initiating Actor), System
Description	The instructor can add what he/she feels is lacking such as assignments, deadlines of assignments, creations of groups, invitations of new members to the groups and so on.
Preconditions	<ol style="list-style-type: none"> 1. The instructor must have a valid account. 2. The instructor must be connected to the internet. 3. The system must be active. 4. The instructor must have entered the monitoring page. 5. The instructor must have entered the View Dashboard page. 6. The instructor must have entered the Monitor Page.
Postconditions	The instructor shall add something to the dashboard.
Priority	Low
Frequency of use	2 times per lifetime
Flow of events	<ol style="list-style-type: none"> 1. The instructor must have entered the monitor content page. 2. The instructor must click on “Add Content” at the monitoring page. 3. The instructor shall choose to implement one of the actions available. 4. The system acknowledges the instructor what he/she has added and directs him/her back to the monitoring page.

Alternative flows	<p>AF-S3: If the instructor decides not to add anything</p> <ol style="list-style-type: none"> 1. The instructor clicks “Back Button” to return to the monitoring page.
Exceptions	<p>EX2: The instructor force quit the program before something was added to the dashboard.</p> <ol style="list-style-type: none"> 1. The system deems nothing is added, logs the instructor out and closes the entire application.
Includes	N/A
Special Requirements	<ol style="list-style-type: none"> 1. Application system is well connected to the database of questions and well linked to the user's social media account. 2. Contents of the monitoring page must be displayed after the instructor has entered the page within 1 seconds. 3. Instructors can switch accounts by clicking “Logout Button”.
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible all the time. 2. Instructors' connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

12. Modify Content

Use Case ID:	12		
Use Case Name:	Modify Content		
Created By:	Koh Zhuang Chean	Last Updated By:	Koh Zhuang Chean
Date Created:	19/02/2020	Date Last Updated:	10/04/2020

Actor	Instructor (Initiating Actor), System
Description	The instructor can modify what they feel is inappropriate during the monitoring of the contents or with response to the users' feedback.
Preconditions	<ol style="list-style-type: none"> 1. The instructor must have a valid account. 2. The instructor must be connected to the internet. 3. The system must be active. 4. The instructor must have entered the View Dashboard page. 5. The instructor must have entered the Monitor Page.
Postconditions	The instructor has modified existing content of the dashboard.
Priority	Low
Frequency of use	2 times per lifetime
Flow of events	<ol style="list-style-type: none"> 1. The instructor enters the monitor content page. 2. The instructor chooses to click the "Modify Content" at the monitoring page. 3. The instructor can then choose to modify one of the existing contents available. For example, the content of assignments, deadlines, groups' names, and etc. 4. The system notifies the instructor what he/she has modified and directs him/her back to the monitoring page.
Alternative flows	AF-S3: If the instructor decides not to modify anything

	<ol style="list-style-type: none"> 1. The instructor clicks “Back Button” to return to the monitoring page.
Exceptions	EX2: The program force quits before something is modified. <ol style="list-style-type: none"> 1. The system deems nothing is modified, logs the instructor out and closes the entire application.
Includes	N/A
Special Requirements	<ol style="list-style-type: none"> 1. Application system is well connected to the database of questions and well linked to the user's social media account. 2. Contents of the monitoring page must be displayed after the instructor has entered the page within 1 second. 3. Instructors can switch accounts by clicking “Logout Button”.
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible at all times. 2. Instructors connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

13. Delete Content

Use Case ID:		13	
Use Case Name:		Delete Content	
Created By:	Koh Zhuang Chean	Last Updated By:	Koh Zhuang Chean
Date Created:	19/02/2020	Date Last Updated:	10/04/2020

Actor	Instructor (Initiating Actor), System
Description	The instructor can delete what they grasp while monitoring the user's content or what they know regarding that content.
Preconditions	<ol style="list-style-type: none"> 1. The instructor must have a valid account. 2. The instructor must be connected to the internet. 3. The system must be active. 4. The instructor must have entered the monitoring page. 5. The instructor must have entered the View Dashboard page. 6. The instructor must have entered the Monitor Page.
Postconditions	The instructor has deleted existing content of the dashboard.
Priority	Low
Frequency of use	2 times per lifetime
Flow of events	<ol style="list-style-type: none"> 1. The instructor enters the monitor content page. 2. The instructor chooses to click the “Delete Content” at the monitoring page. 3. The instructor can then choose to delete one of the existing contents available. For instance, the context of assignments, the deadlines of assignments, members of groups, etc. 4. The system notifies the instructor what he/she has deleted and directs him/her back to the monitoring page.

Alternative flows	<p>AF-S3: If the instructor decides not to delete anything</p> <ol style="list-style-type: none"> 1. The instructor clicks “Back Button” to return to the monitoring page.
Exceptions	<p>EX2: The program force quits before something is deleted.</p> <ol style="list-style-type: none"> 1. The system deems nothing is deleted, logs the instructor out and closes the entire application.
Includes	N/A
Special Requirements	<ol style="list-style-type: none"> 1. Application system is well connected to the database of questions and well linked to the user's social media account. 2. Contents of the monitoring page must be displayed after the instructor has entered the page within 1 seconds. 3. Instructors can switch accounts by clicking “Logout Button”.
Assumptions	<ol style="list-style-type: none"> 1. Database server must be active and accessible all the time. 2. Instructor's connections must be stable. 3. System's connections must be stable.
Notes and issues	TBD

14. Compete Mode

Use Case ID	14		
Use Case Name	Compete Mode		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor)
Description	The user can enter the ‘Compete’ mode, to either ‘Create your own Test’ or select ‘Challenge’ Mode.
Preconditions	<ol style="list-style-type: none"> 1. The User must be eligible to participate in the particular World, i.e. the World must be unlocked. 2. User must have a valid account 3. User must be logged in successfully 4. User must be connected to the internet 5. The system must be active
Postconditions	<ol style="list-style-type: none"> 1. User has selected one of the two options: <ol style="list-style-type: none"> 1. Create own Test 2. Select Challenge Mode
Priority	High
Frequency of Use	Once per Lifetime
Flow of Events	<ol style="list-style-type: none"> 1. The User chooses to enter the ‘Compete’ mode of the game and select one of the following options. <ol style="list-style-type: none"> 1. Challenge Mode: <ol style="list-style-type: none"> 1. User selects challenge mode 2. The System opens the challenge mode window

	<p>2. Create your Own Test:</p> <ol style="list-style-type: none"> 1. User selects “Create your Own Test” 2. The System opens the Create your “Own Test” window
Alternative Flows	<p>AF-S1: User decides to quit the application</p> <ol style="list-style-type: none"> 1. User is asked to confirm decision to quit the application 2. User is logged out 3. User session is terminated
Exceptions	N/A
Includes	Create your Own Test and Challenge use cases
Special Requirements	<ol style="list-style-type: none"> 1. Application system is well connected to the database of questions and well linked to the User’s account. 2. The content of the section must be displayed within 2 seconds after selecting that section. 3. Users must be able to go back to the sections list by clicking the “Back Button”.
Assumptions	<ol style="list-style-type: none"> 1. User’s connection must be stable. 2. System’s connection must be stable.
Notes and Issues	TBD

15. Compete Mode [Challenge Mode]

Use Case ID	15		
Use Case Name	Compete Mode [Challenge Mode]		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor)
Description	In the Challenge Mode, the System sets up a random game between two players.
Preconditions	<ol style="list-style-type: none"> 1. User must have a valid account 2. User must be logged in successfully 3. User must be connected to the internet 4. The system must be active 5. The User must select the “Challenge Mode” option under the “Compete Mode”. 6. The User must have at least one friend on their Social Media account to whom a challenge can be sent. 7. At least one friend on the user’s social media account must have installed the application. 8. Both the user and the selected friend must be eligible for the level/world. 9. Both the user and the friend must be willing for a challenge.
Postconditions	A random game must be set-up between the two players.
Priority	Medium
Frequency of Use	Once per Lifetime

Flow of Events	<ol style="list-style-type: none"> 1. When the User is in Challenge Mode, the System asks the initiating player to select the topic of the Challenge. 2. The System then extracts the list of the User's friends who are available in the application through their Social Media account. 3. The system creates game-style questions using the database of questions. 4. The system must calculate total points earned for a player using specific rules. <ol style="list-style-type: none"> 1. The player who completes the mini game earlier gets 15 points. 2. The player who completes the game later gets 5 points.
Alternative Flows	<p>AF-S2. The User has friends, but none of the friends are linked to the game software.</p> <ol style="list-style-type: none"> 1. The User is suggested to send an application download invite to friends in the user's social media friend list.
Exceptions	N/A
Includes	Challenge use case
Special Requirements	<ol style="list-style-type: none"> 1. The User should allow the application to access the user's friend list in the linked social media account. 2. The system is not allowed to post/update social media account without the permission of the user
Assumptions	<ol style="list-style-type: none"> 1. User's connection must be stable. 2. System's connection must be stable. 3. Database server must be active and accessible at all times.
Notes and Issues	TBD

16. Compete Mode [Choose Topic]

Use Case ID	16		
Use Case Name	Compete Mode [Choose Topic]		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	Feb 15, 2020	Date Last Updated	Feb 15, 2020

Actor	User (Initiating Actor), Friend's Application system
Description	User selects the topic on which the questions are to be set for the user friend. Once the friend accepts the challenge the challenge game begins. When a person (the challenger or the challenged) completes the test, the score is calculated by (No. of correct answers x Points per correct Answer) ÷ (Time Taken).
Preconditions	<ol style="list-style-type: none"> 1. User must have a valid account 2. User must be logged in successfully 3. User must be connected to the internet 4. The system must be active 5. User has selected a friend, to whom the user wishes to challenge 6. The friend is eligible for this challenge
Postconditions	<ol style="list-style-type: none"> 1. A random test from a list of premade tests from the developers to be presented to the user 2. A Challenge request is sent to the friend's application system
Priority	Medium
Frequency of Use	Once in a lifetime

Flow of Events	<ol style="list-style-type: none"> 1. System sends request access to user's social media friend list 2. On granting access, the system displays the friend list to the user 3. User selects a friend, to whom the user wishes to challenge 4. User selects the topic on which the questions are to be set for the user friend 5. System sends challenge request to the friend 6. Friend accepts the challenge request 7. The challenge begins
Alternative Flows	<p>AF-S6: Friend has declined the challenge</p> <ol style="list-style-type: none"> 1. User is notified that the selected friend has declined the challenge offer
Exceptions	<p>EX1: User internet connectivity is weak</p> <ol style="list-style-type: none"> 1. User is notified that internet connectivity is "weak" and might affect the game performance
Includes	N/A
Special Requirements	<ol style="list-style-type: none"> 1. The user's and friend's system is able to actively interact with one another 2. Friend's system accepts challenges from other users 3. Both systems have access to good internet connectivity
Assumptions	<ol style="list-style-type: none"> 1. Friend is eligible for this challenge 2. Friend has internet access 3. The user has already logged into their account. 4. User's connections must be stable. 5. System's connections must be stable. 6. Database server must be active and accessible all the time.
Notes and Issues	TBD

17. Create Your Own Test

Use Case ID	17		
Use Case Name	Create Your own Test		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor)
Description	User is asked to select game type and a list of questions, to make a custom test
Preconditions	<ol style="list-style-type: none"> 1. User must select the “Create Your own test” option in the “Compete mode” 2. User must have a valid account 3. User must be logged in successfully 4. User must be connected to the internet 5. The system must be active
Postconditions	User is told to select the type of questions
Priority	Medium
Frequency of Use	Once a lifetime
Flow of Events	<ol style="list-style-type: none"> 1. User selects the “Create Your own test” option in the “Compete mode” 2. User is asked to select the question type.
Alternative Flows	<p>AF-S1. User wishes to return to the ‘Compete Mode’ window</p> <ol style="list-style-type: none"> 1. User is taken back to the ‘Compete Mode’ window
Exceptions	N/A

Includes	N/A
Special Requirements	1. System has access to the database of questions
Assumptions	<ol style="list-style-type: none">1. User has a free time to complete the test<ol style="list-style-type: none">1. The test cannot be aborted once the test is started2. The user has already logged into their account.3. User's connection must be stable.4. System's connection must be stable.5. Database server must be active and accessible all the time.
Notes and Issues	TBD

18. Choose Question Type

Use Case ID	18		
Use Case Name	Choose Question Type		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	User (Initiating Actor)
Description	User selects the type of question to be used in the 'Create Your Own Test' option of the Compete Mode
Preconditions	<ol style="list-style-type: none"> 1. User must have selected the "Create your own test" mode 2. User must have a valid account 3. User must be logged in successfully 4. User must be connected to the internet 5. The system must be active
Postconditions	<ol style="list-style-type: none"> 1. The user has already logged into their account. 2. User is given a list of questions to answer. 3. The questions are in the user selected questions format
Priority	Low
Frequency of Use	Once a lifetime
Flow of Events	<ol style="list-style-type: none"> 1. User is asked to select the question type in the "Create your own" window 2. User is asked to choose from one of the 4 question types (Match, Hangman, MCQ, or Unscramble) 3. System sends request to access questions database 4. System extracts and displays the questions 5. User is asked to select questions from the list given 6. User selects at least 4 questions and at most 8 questions

	7. Custom test is made
Alternative Flows	AF-S1. The user return to compete mode window 1. User is taken back to the 'Compete Mode' window
Exceptions	EX 1. Question Database does not have sufficient questions in the use selected question format 1. User is suggested to choose a different question type
Includes	N/A
Special Requirements	1. System must have sufficient questions in the topic selected by the user 2. System must have fast and live time access to the question database 3. System must be able to fetch the required question with ease to minimise user waiting time 4. User can select maximum of 8 questions from the list of questions 5. User has selected a minimum of 4 questions from the question list
Assumptions	1. User has a good understanding of the concepts discussed in the test 2. User's connections must be stable. 3. System's connections must be stable. 4. Database server must be active and accessible all the time.
Notes and Issues	TBD

19. View Dashboard

Use Case ID	19		
Use Case Name	View Dashboard		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	Instructor (Initiating Actor)
Description	Dashboard for the instructor allows the instructor to use the application to either “Send Assignment”, “Track Progress”, or “View assessment history”
Preconditions	<ol style="list-style-type: none"> 1. Instructor must have a valid account 2. Instructor must be logged in successfully 3. Instructor must be connected to the internet 4. The system must be active 5. Instructor has entered the dashboard section
Postconditions	<ol style="list-style-type: none"> 1. The Instructor is able to view the dashboard 2. Instructor enters the “Send Assignment” section 3. Else instructor enters the “Track progress” section 4. Else instructor enters the “View assessment history” 5. Else instructor quits application or exits
Priority	High
Frequency of Use	3 times in a lifetime
Flow of Events	<ol style="list-style-type: none"> 1. Instructor enters the dashboard 2. Instructor clicks on “Send Assignment” <ol style="list-style-type: none"> 1. System allows the user to create assignment 3. Else Instructor clicks on “Track progress”

	<ol style="list-style-type: none"> 1. System performs the relevant functionalities to display progress status 4. Else Instructor click on “View assessment history” <ol style="list-style-type: none"> 1. System will display all the past assignments sent to students and the completion status of each assignment
Alternative Flows	N/A
Exceptions	<p>EX1. The instructor decides to exit the game before selecting any option</p> <ol style="list-style-type: none"> 1. User is asked for a confirmation before exiting 2. System exits <p>EX2. Instructor logs out of the system</p> <ol style="list-style-type: none"> 1. System terminates the current session of the user <p>EX3 Lack of activity causes auto logout</p> <ol style="list-style-type: none"> 1. Instructor must again log into the application to resume
Includes	None
Special Requirements	<ol style="list-style-type: none"> 1. Application must be well connected to the server and the database 2. Application server must be well functioning 3. Application server must be able to adapt to changes in dashboard functionality
Assumptions	<ol style="list-style-type: none"> 1. Instructor has logged into the system 2. Instructor system has access to internet 3. Application server is not down 4. Instructor's connections must be stable. 5. System's connections must be stable.
Notes and Issues	TBD

20. Track Student Progress

Use Case ID	20		
Use Case Name	Track Student Progress		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor	Instructor (Initiating Actor)
Description	The System shall display the list of student groups and allow the instructor to select a particular student in a group. Then the system shall display the progress of that student
Preconditions	<ol style="list-style-type: none"> 1. Instructor should have logged into the system 2. The application is connected to internet 3. System server is active 4. Instructor has entered the Track Student Progress
Postconditions	System must display the progress of the selected student of student group
Priority	Medium
Frequency of Use	Once a lifetime
Flow of Events	<ol style="list-style-type: none"> 1. Instructor chooses the “Track Student Progress” option from the instructor dashboard 2. System will fetch the student list from the database 3. System shall display the list of student groups to the instructor 4. Instructor shall select a student from a student group 5. System will fetch the progress status of the selected student

	<p>6. System will then display the progress status of the selected student</p>
Alternative Flows	<p>AF-S1: If the system is unable to fetch the student group list</p> <ol style="list-style-type: none"> 1. System informs the instructor that no student group is created 2. Instructor must create a student group and add students into it
Exceptions	<p>EX1. User has decided to exit the application</p> <ol style="list-style-type: none"> 1. User is asked to confirm decision to quit the application 2. Instructor session terminates
Includes	View Summary report
Special Requirements	<ol style="list-style-type: none"> 1. Application system must be able to access and retrieve student data from the database 2. System must be able to easily query through the database 3. System must minimise the time taken to retrieve the student data from the database 4. Application server must be functioning well.
Assumptions	<ol style="list-style-type: none"> 1. Instructor has successfully logged into the system 2. Instructor system has access to internet
Notes and Issues	TBD

21. View Summary Report

Use Case ID	21		
Use Case Name	View Summary Report		
Created By	Asok Kumar Gaurav	Last Updated By	Asok Kumar Gaurav
Date Created	15/02/2020	Date Last Updated	10/04/2020

Actor:	Instructor (Initiating Actor)
Description:	System will calculate the progress of a student in terms of the number of assignments completed and its accuracy. Then the system will display the corresponding student statistics and graphs to the instructor along with the strengths and weaknesses of the student.
Preconditions:	<ol style="list-style-type: none"> 1. Instructor must have logged into the system 2. The application must be connected to internet 3. System server must be active 4. Instructor must have chosen to view the summary report
Postconditions:	System will display the selected student's progress stats, strengths, and weaknesses
Priority:	Low
Frequency of Use:	1 time in a lifetime
Flow of Events:	<ol style="list-style-type: none"> 1. The application requests for access to the student database 2. Once access is granted, the system fetches and processes the student data 3. The system calculate the student progress in terms of the number of assignments completed and its accuracy

	<ol style="list-style-type: none"> 4. System will generate graphs and select the relevant statistics 5. Graphs and statistics are displayed to the instructor 6. The statistics are processed and analysed by the application to determine the student's strengths and weaknesses. 7. Student's strengths and weaknesses, are displayed to the instructor
Alternative Flows:	<p>AF-S1: If the system is unable to calculate process data</p> <ol style="list-style-type: none"> 1. System informs the instructor that there is not sufficient data to process and calculate strengths and weakness of the student 2. Student usage data is displayed
Exceptions:	<p>EX1. User has decided to exit the application</p> <ol style="list-style-type: none"> 1. User is asked to confirm decision to quit the application 2. Instructor session terminates
Includes:	N/A
Special Requirements:	<ol style="list-style-type: none"> 1. System must have access to the student database 2. System must be able to quickly retrieve the data 3. System must be able to accurately calculate the progress and completion stats 4. The system must be able to judge if there is enough data to calculate the strengths of the student
Assumptions:	<ol style="list-style-type: none"> 1. Instructor has successfully logged into the system 2. System has access to internet 3. System's connections must be stable. 4. Database server must be active and accessible.
Notes and Issues:	N/A

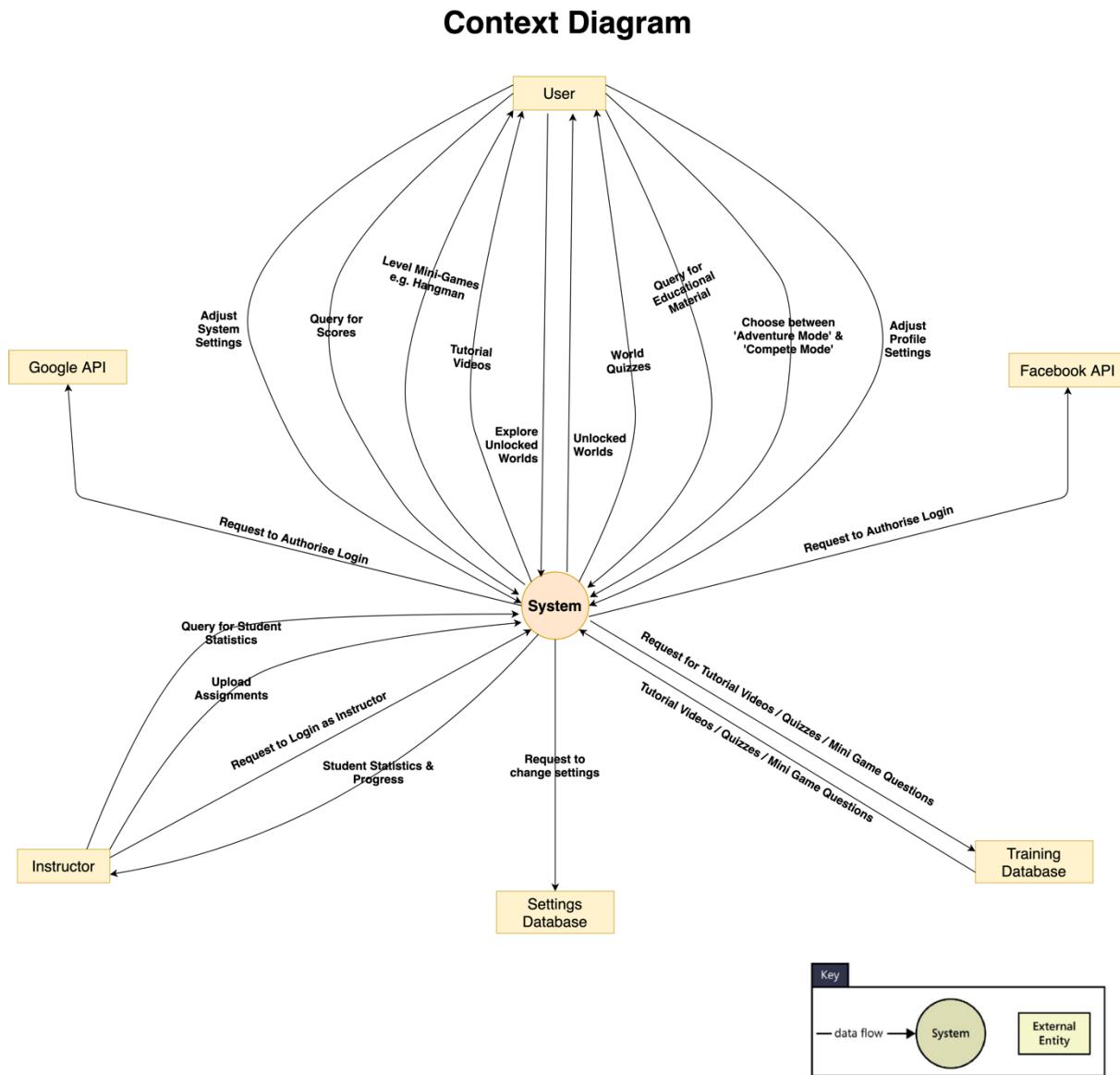
22. Check Status of Assignments

Use Case ID	22		
Use Case Name	Check Status of assignments		
Created By	Jay Yap	Last Updated By	Jay Yap
Date Created	19/02/2020	Date Last Updated	10/04/2020

Actor	Instructor (Initiating Actor), System
Description	The instructor can enquire about the assignments graded for a given student or a given world.
Preconditions	<ol style="list-style-type: none"> 1. The instructor must have a valid account. 2. The instructor must have entered the Status of Assignments page.
Postconditions	Status of the assignments enquired is shown to the instructor.
Priority	Medium
Frequency of Use	Twice per Lifetime
Flow of Events	<ol style="list-style-type: none"> 1. System will prompt the instructor to select a particular student or a world for enquiry. 2. The instructor makes a choice. <ol style="list-style-type: none"> 1. If the instructor wants to enquire on a student. <ol style="list-style-type: none"> 1. System will prompt the instructor to enter the matriculation number and name of the student. 2. The instructor inputs matriculation number and name of student 3. The details and performance of the student will be shown 2. If the instructor wants to enquire on a world

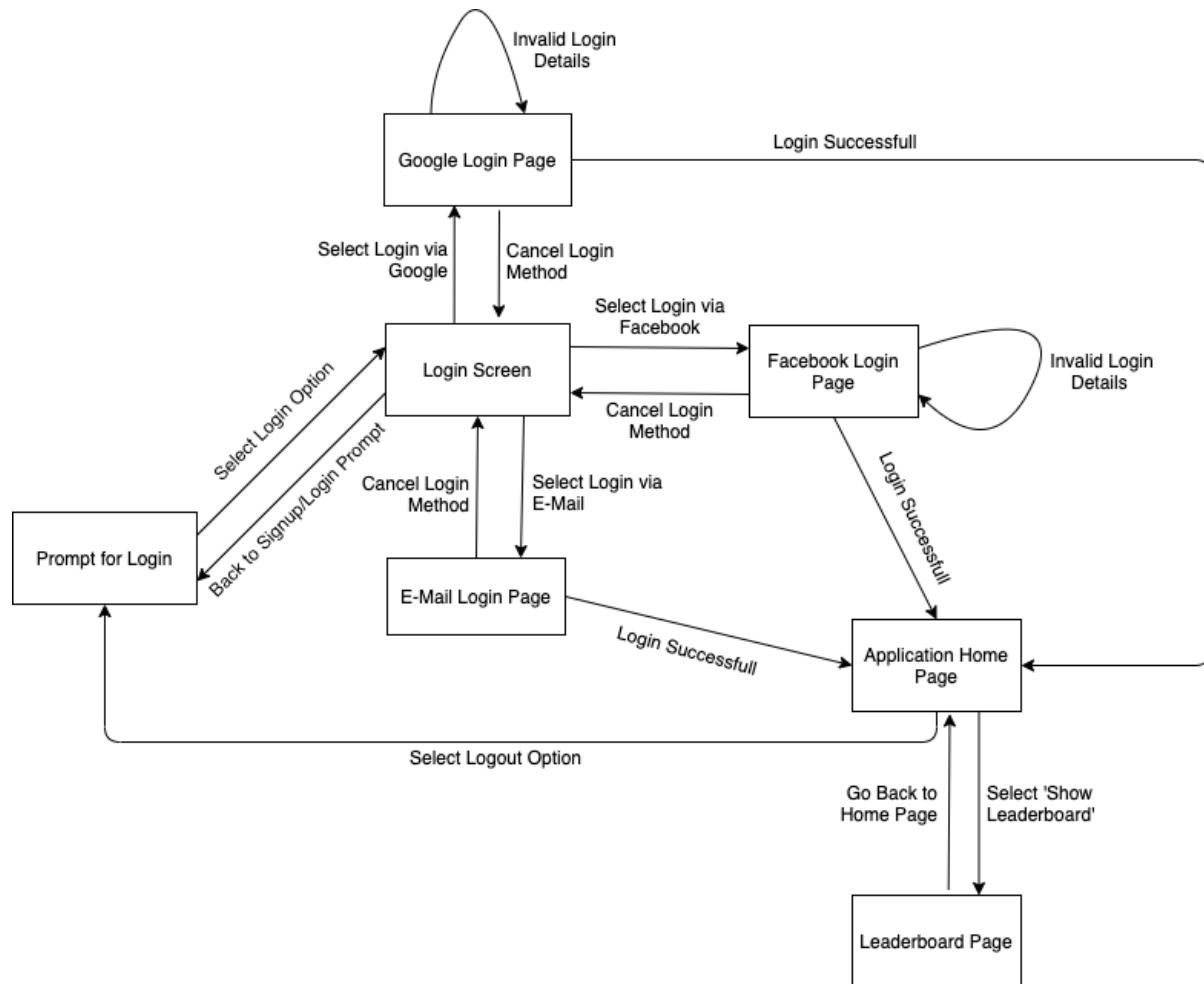
	<ol style="list-style-type: none"> 1. System will prompt the instructor to select a world from a dropdown menu 2. The instructor selects a world. 3. The number of students who have attempted the quiz for that world, and their scores for the selected world will be shown
Alternative Flows	N/A
Exceptions	N/A
Includes	Retrieve queried information from statistic database
Special Requirements	The queried data must be retrieved and displayed to the instructor within 1 second.
Assumptions	<ol style="list-style-type: none"> 1. The instructor is connected to the internet. 2. User's connections must be stable. 3. System's connections must be stable. 4. Database server must be active and accessible all the time.
Notes and Issues	TBD

6.4. Context Diagram

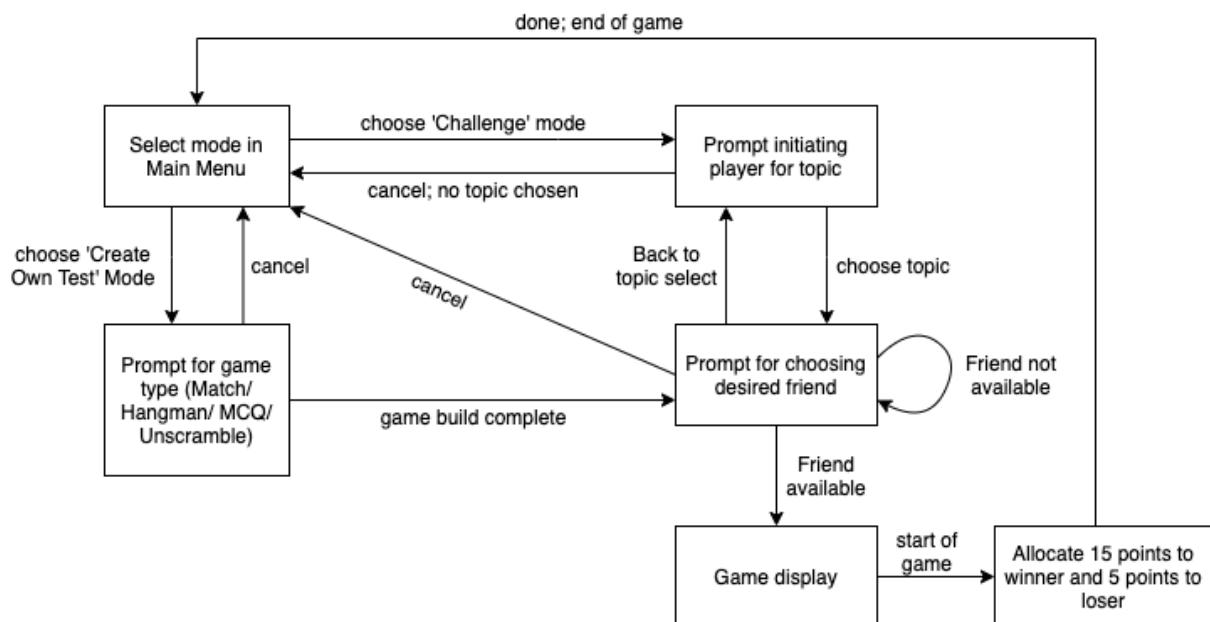


6.5. State Transition Diagram(s)

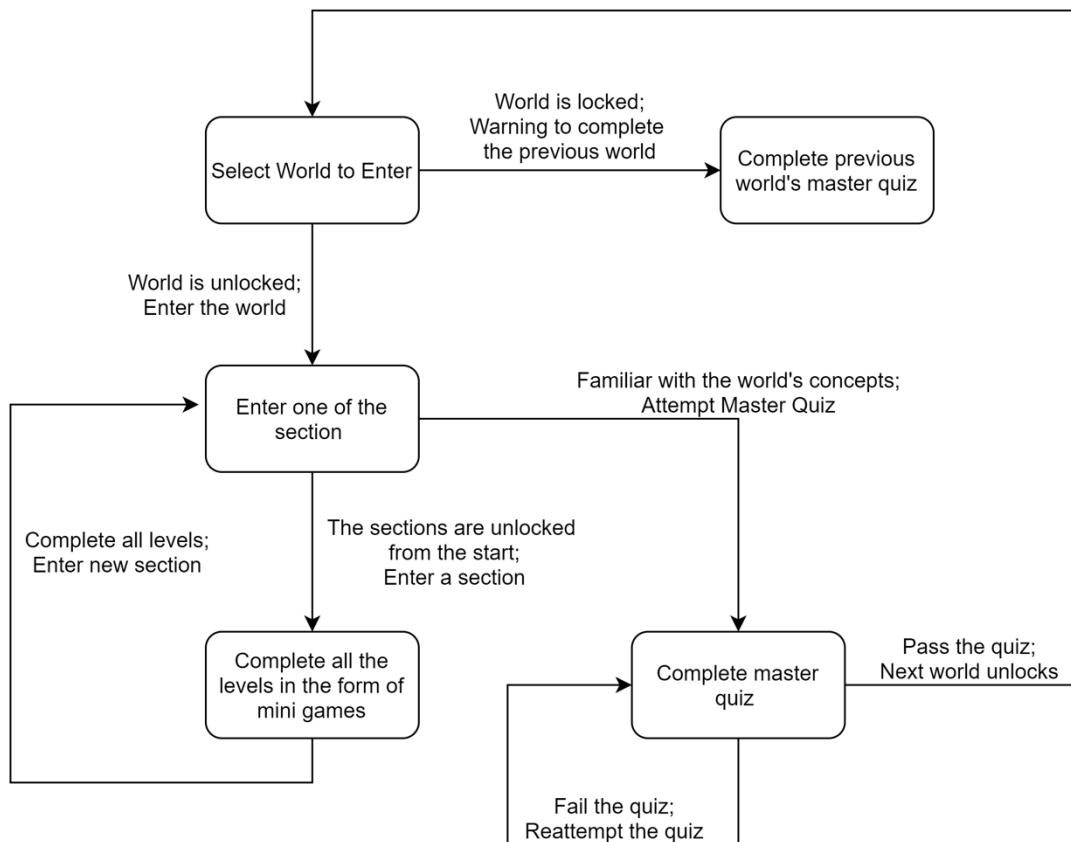
1. Login / Logout / Leaderboard



2. Compete Mode



3. Adventure Mode



6.6. CRUDL Matrix

Entity Use Case	Points per User	Database of Questions	Database having Login Details	Database of Learning Material	World Unlock Tracker	Game Settings
Create your Account	-	-	C, U	-	-	-
Login	-	-	R	-	-	-
Display Game Settings	-	-	-	-	-	R, L
Change Game Settings	-	-	-	-	-	U, L
Create your own Test	-	C	-	-	-	-
Delete your own Test	-	D	-	-	-	-
Update Questions of your own Test	-	U	-	-	-	-
Challenge your Friend	U	R, L	-	-	-	-
Answer Questions for each Level	U	R, L	-	-	-	-
Answer the Master Quiz	U	R, L	-	-	U, L	-
View Leaderboard	L	-	-	-	-	-

for User/Tutor						
Display the top 20 players in the Leaderboard	L	-	-	-	-	-
Users Learning	-	-	-	R, L	-	-
Add/Delete Content by Instructors	-	-	-	C, R, U, D, L	-	-
View reports of Students by Instructors	R, L	-	-	R	R	-

6.7. Decision Table(s)

A. Sending Verification Email

Condition	1	2	3
User has linked through a valid email	T	F	T
User has linked through Facebook/Google account	-	T	T
Action	X	X	
Send verification email			
Reject sending email			X

B. Proceed to a New World

Condition	1	2	3
User authorized	F	T	T
The previous world unlocked	-	F	T
Action			X
Accept Request			
Reject Request	X	X	

C. Carry out ‘Challenge’ Option

Condition	1	2	3	4
The user chooses the ‘Compete’ option	F	T	T	T
The user chooses the ‘Challenge’ option	-	F	T	T
The chosen friend is online too	-	-	F	T
Action				X
Conduct request				
Reject request	X	X	X	

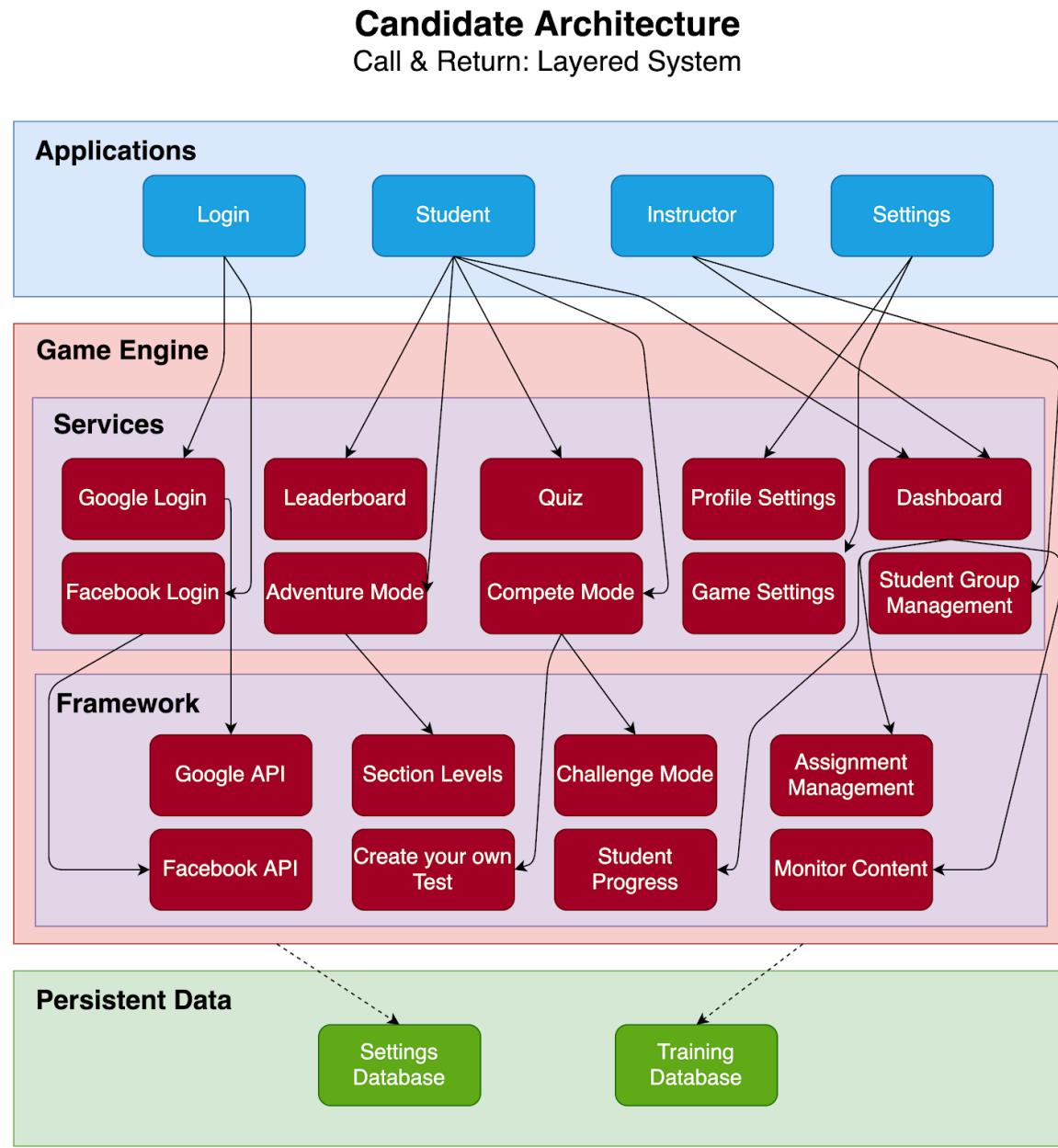
7. Non-Functional Requirements

7.1. Reliability

In case of the app being rebooted, the app should launch within 30 seconds.

There should be no case of the app crashing under normal usage.

8. Candidate Architecture



8.1. Rationale for Candidate Architecture

8.1.1. Our Approach

Here we have chosen the layered subtype of the call-and-return architectural style. This is useful for segregating our program into departments, so to speak, where each layer manages its own set of functions, processes and controls. This makes for efficient maintenance of the program and it is easy to assign separate roles to them.

We have decided to divide the application into two main layers, Services and Framework. The former has all the user-facing functions such as logging in, game navigation, settings and the student dashboard. The lower layer provides the underpinning architecture driving them, and it consists of login APIs, question creation frameworks, challenge systems and student monitoring tools. It is, however, our prerogative to ensure that the tasks handled by the Services layer are not mixed with those in the Framework layer or resolving the confusion may prove to be difficult.

There is a third, auxiliary layer called Persistent Data that sits below the Framework layer. It contains the databases used by the application, like the Training and Settings databases. This provides for a level of abstraction, since the Services layer faces the user, and the Framework and Persistent Data layers face away from him/her.

8.1.2. Compared with other Architectural Styles

Independent-component architectures (ICAs) do not serve our purpose since the components do not simply communicate by passing messages. Control and data need to flow from one to another constantly, and we require shared-state data and variables that can be easily used by all functions. Also, the components in ICAs cannot control each other, or rather cannot decide whether, if any, action is to be taken by the recipient of a message, and hence message-passing protocols required by ICAs are difficult to check.

Within ICAs, the two substyles learnt in the lectures (client-server and EBII or event-based implicit invocation) are discussed for their unsuitability as follows:

- The *client-server* substyle (also called the substyle of communicating processes) has only one server and multiple clients. We want the components to take data from multiple servers, such as the training database.
- In the EBII (event-based implicit invocation) architectural substyle, components announce that they wish to share data with their environment. This constitutes an event, and other components express their interest in receiving the data. In our application we want the request of data to happen without any announcement. If, for instance, the sign-in function (in the Services layer) requests a connection with the Google or Facebook sign-in API (in the Framework layer), it should happen at any time that the function pleases, not when the API says it is ready to do so.
- In **data flow** architectures, there is the flow of only data, and control flow and other interactions do not take place. This is clearly undesirable in our case, since we want the flow of control to take place amongst services and frameworks.
 - The *pipe-and-filter* substyle is built purposely for concurrent execution. The input data are transformed incrementally, and an output is fed again as an input. They force a lowest common denominator on data transmission. All this is to facilitate the concurrent execution of processes, which we do not really need. Because of these reasons, they are unsuitable for interactive applications like ours.
 - In the *batch sequential processing* substyle, a process runs to completion before the next starts. The performance and runtime are adversely affected, which we do not want.
- Finally, within the call-and-return architectural style that we have chosen, we have not chosen the **main-program-and-subroutine with shared data** substyle because there is not a single driver program running all the others, but a number of programs cooperating to achieve the outcome.

Specify the selected candidate architecture using notation that describes the static programme structure, as well as the dynamic programme behaviour. Use the notation discussed in the lectures.

As specified above, we have used the **Call and Return Architecture** for the overall system, and specifically the **Layered** subtype, the diagram for which has been included above as well.

→ : Solid arrow is used to depict the control flows via procedure/function calls that are made by the static components of the programme to each other, to achieve overall functionality.

⇒ : The dotted arrow is used to depict data flow, used between the static components in the ‘Framework’ layer to retrieve data from the databases in the ‘Persistent Data’ layer below it.

Together, the solid arrows and the dotted arrow form the dynamic component of the system, as they are responsible for the communication between the static components which in turn is used to achieve the functionalities of the overall systems

□: Rectangles with rounded corners are used to depict the Passive components of the system. These represent those components that do not do anything unless explicitly asked to do so (via procedure call/communication, etc.). These include functions/procedures without side effects and have also been used to show the databases in the ‘Persistent Data’ layer.

The rectangles hence form the static component of the system as they are not responsible for the dynamism in the program and will execute only when an explicit call is made.

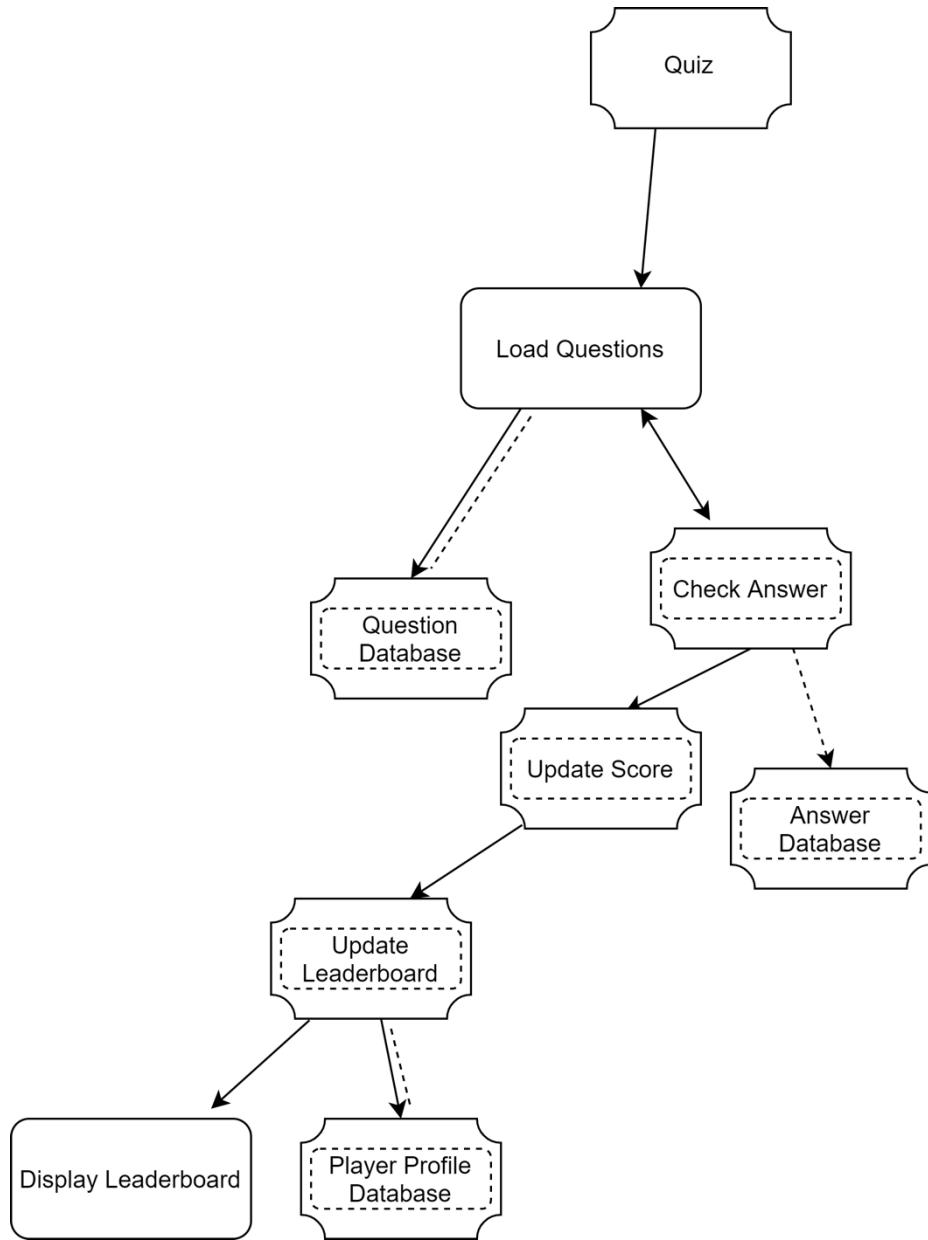
Each layer consists of multiple static components and the communication between the layers is facilitated with the help of the dynamic components, which are the arrows, as shown in the diagram.

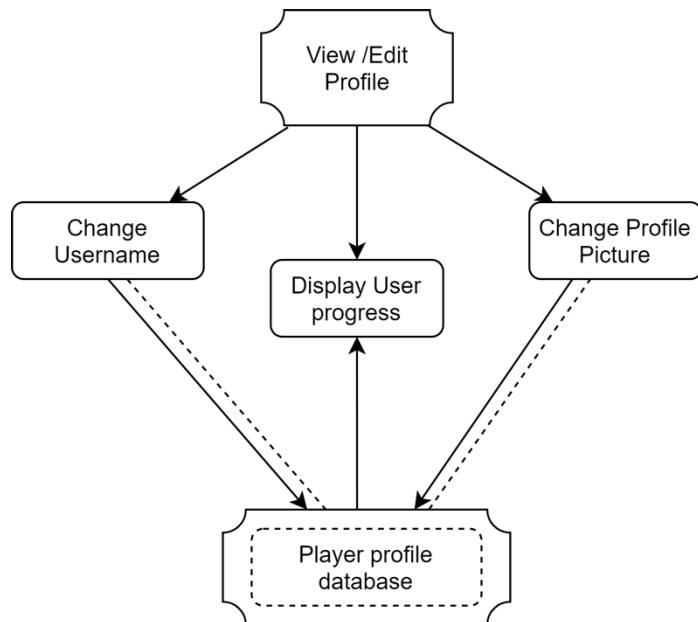
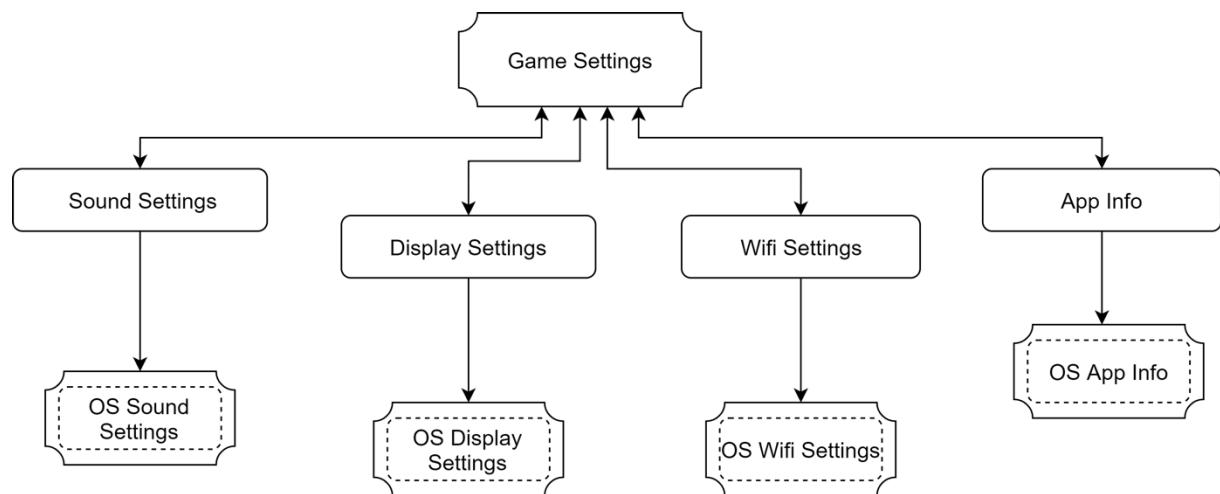
Such a division of the entire architecture provides a clear view of the components that make up the entire system and the control/data flow between such components.

9. Subsystem Interface Design (Game Engine)

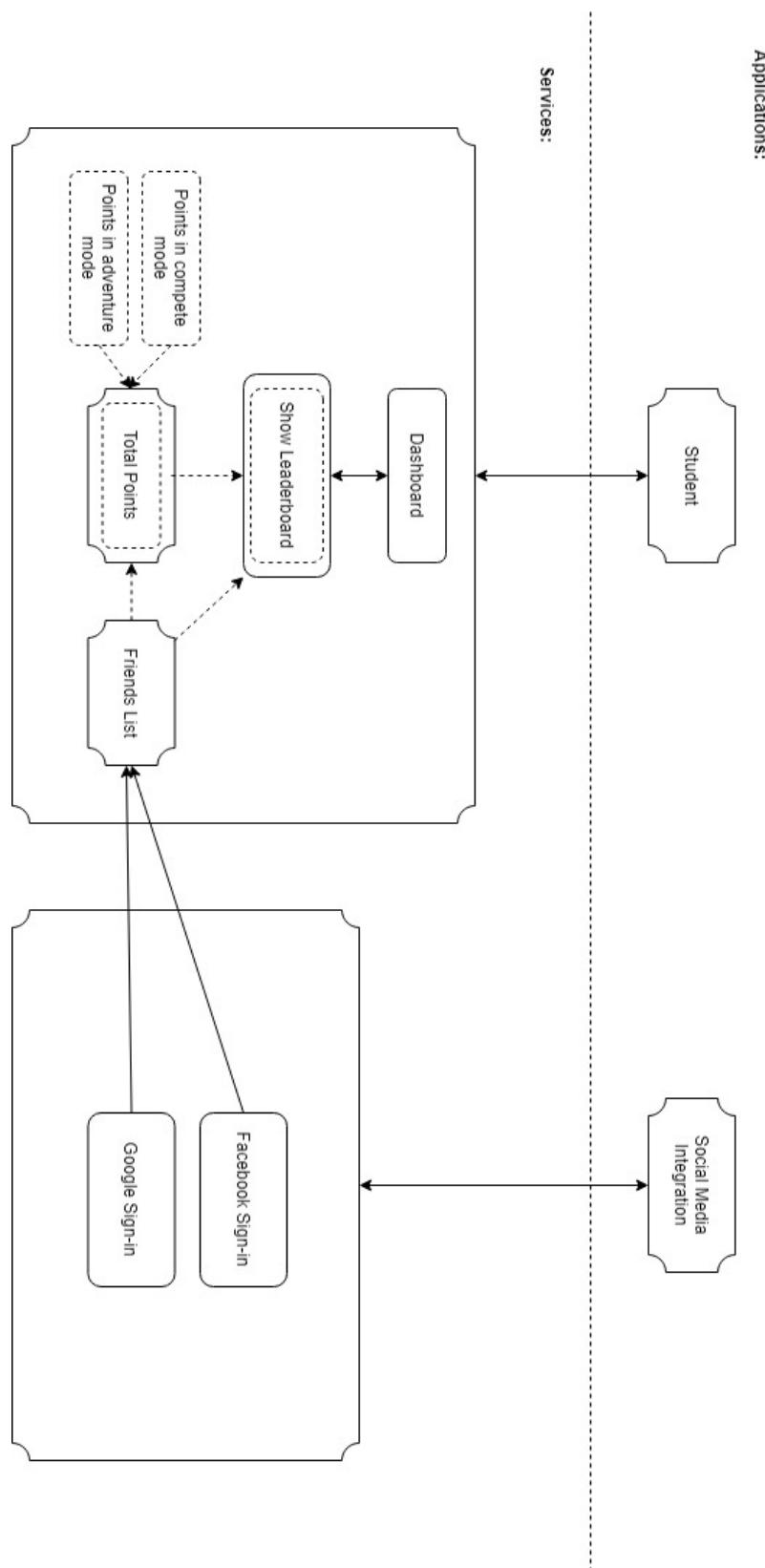
9.1. Services Layer

a. Quiz Architecture Diagram

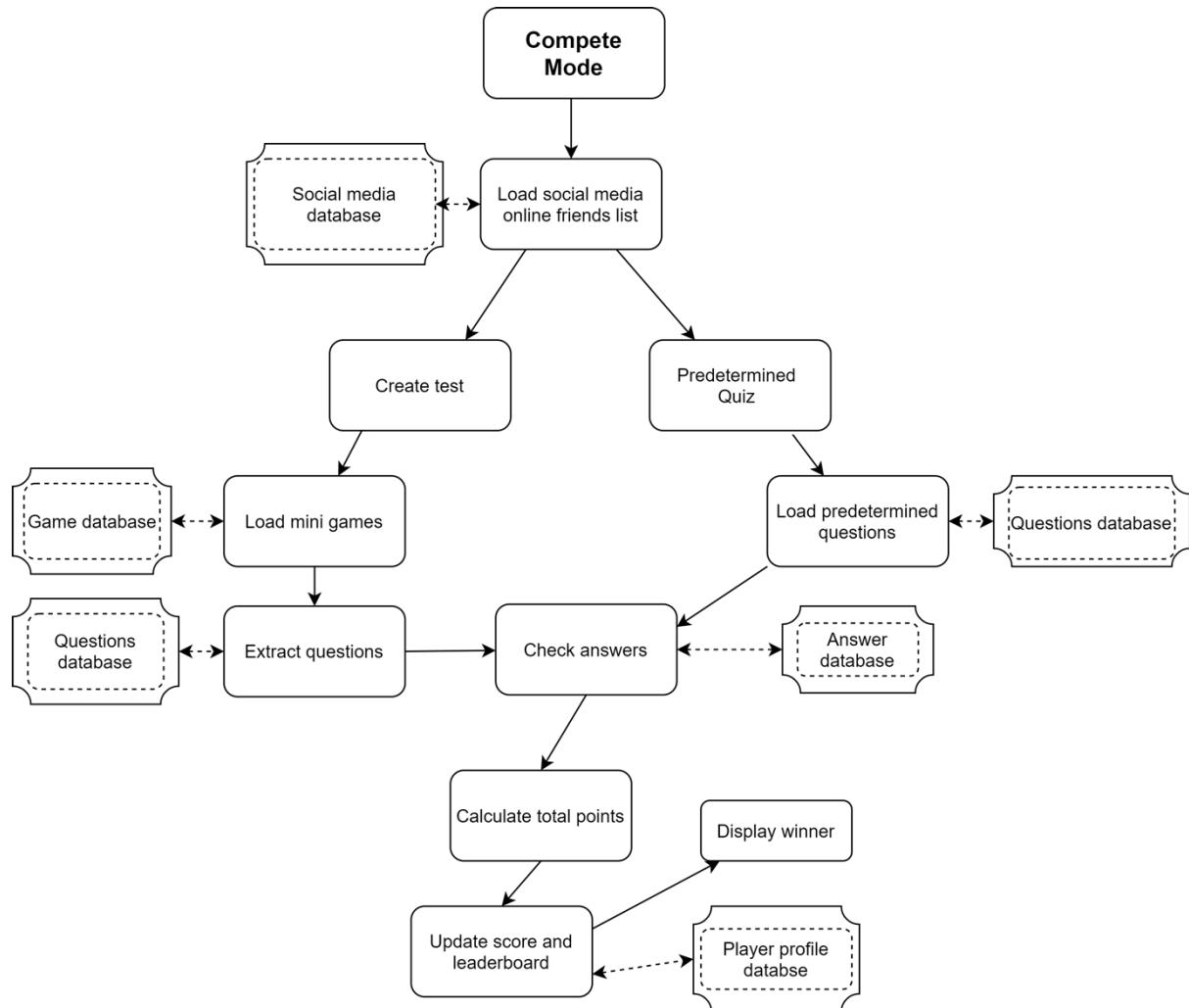


b. Profile Settings Diagram**c. Game Settings Diagram**

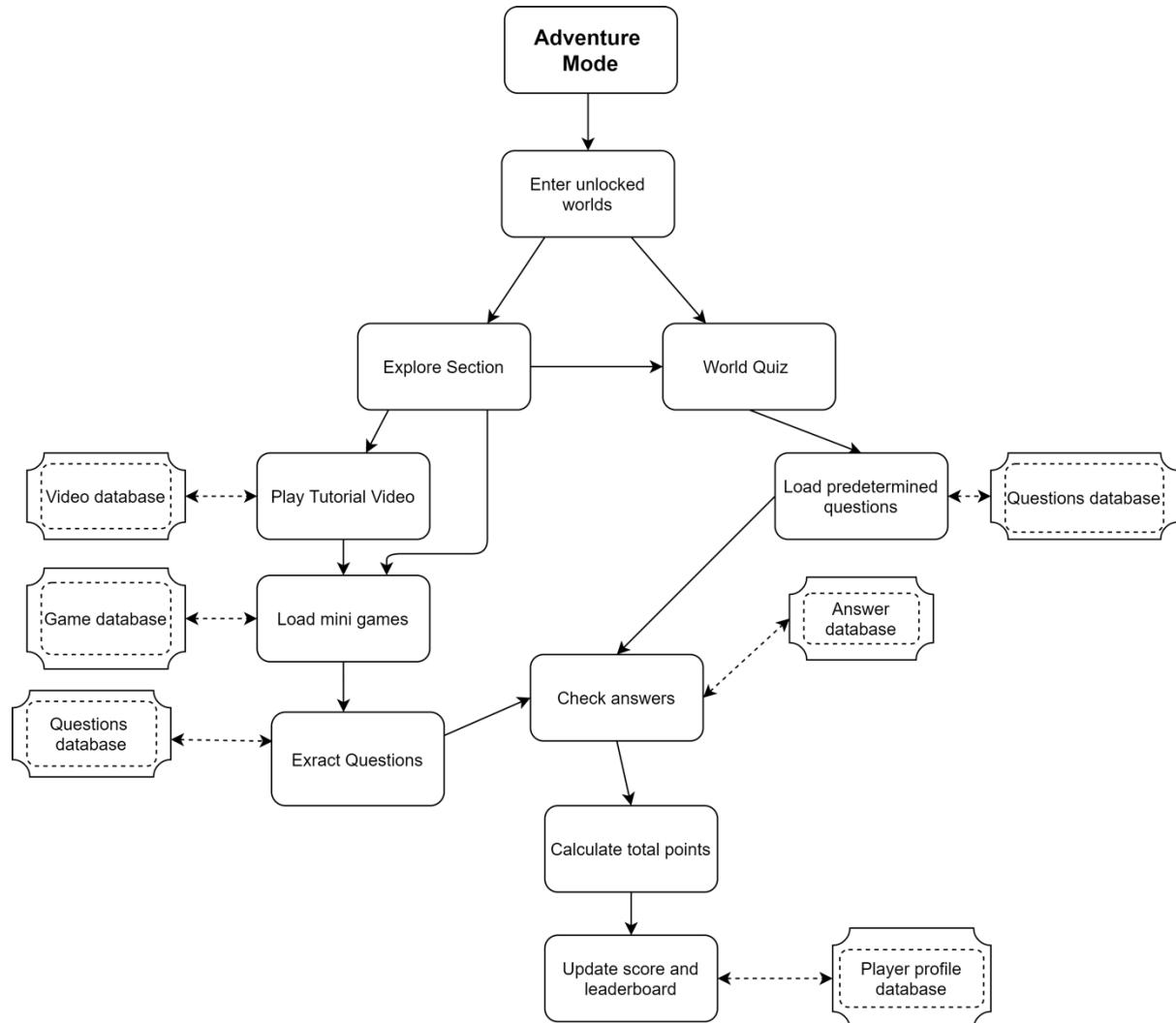
d. Leaderboard Diagram



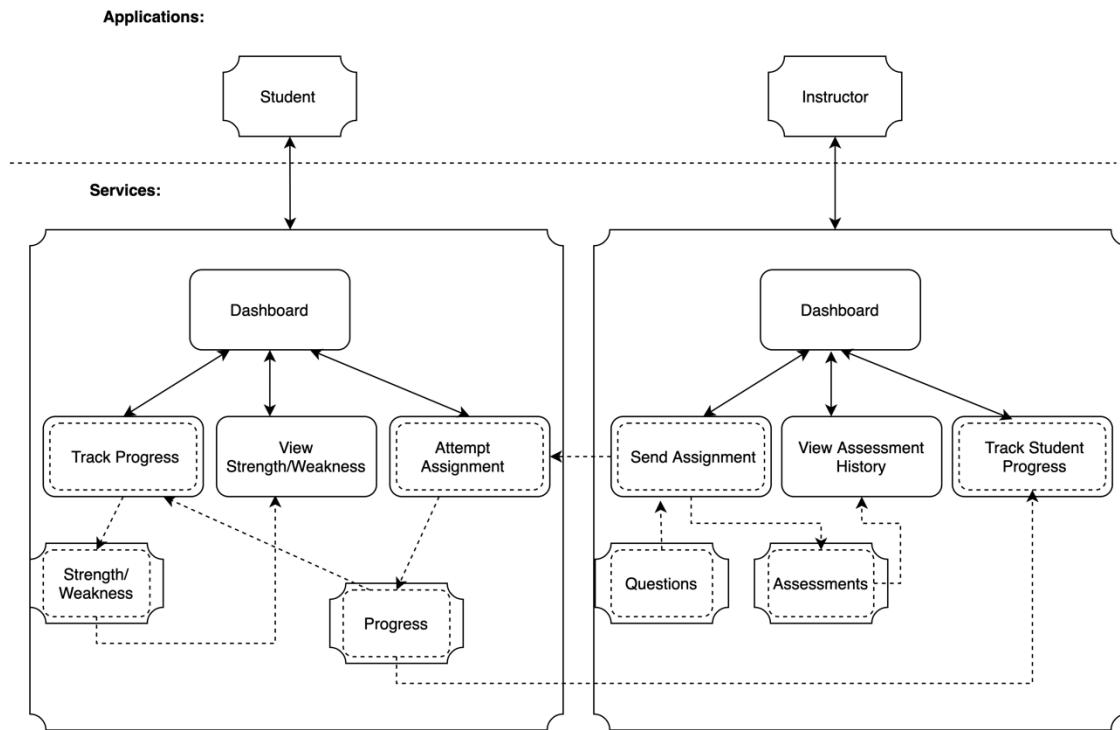
e. Compete Mode Diagram



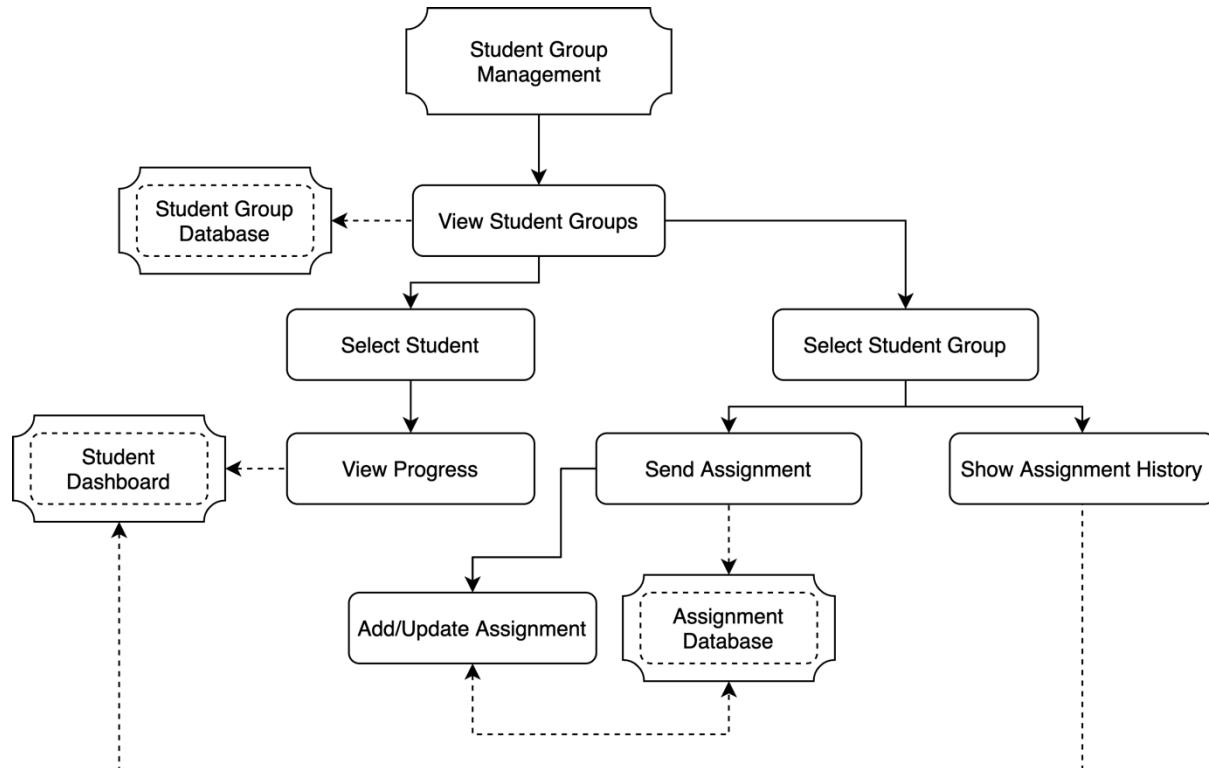
f. Adventure Mode



g. Dashboard Service

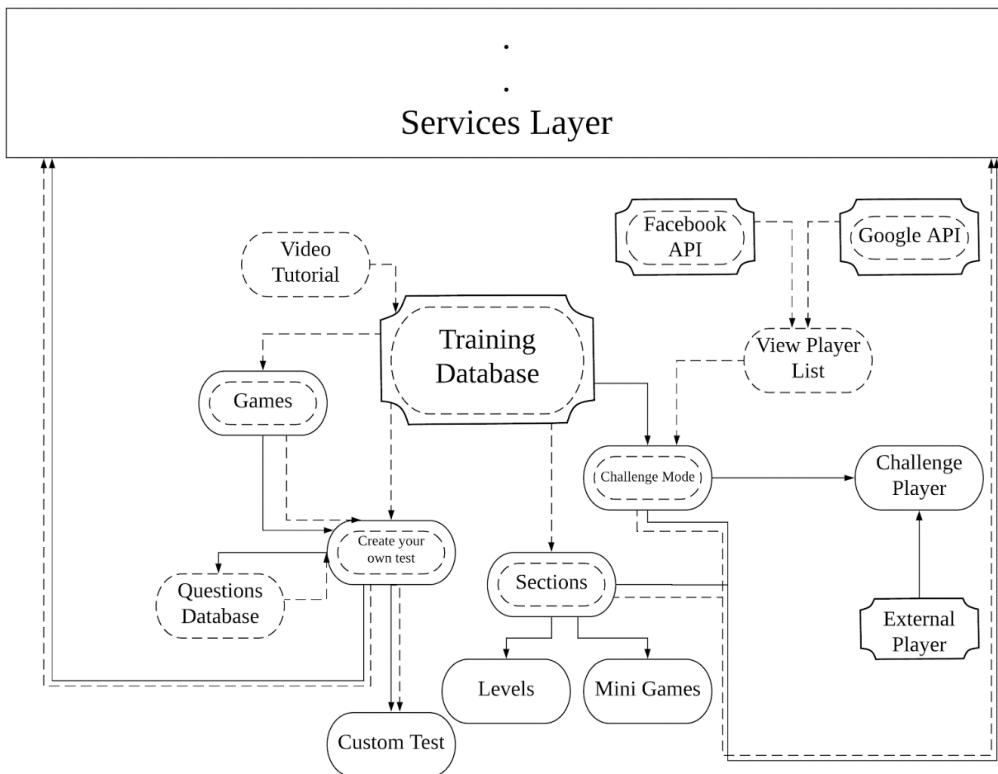


h. Student Group Management

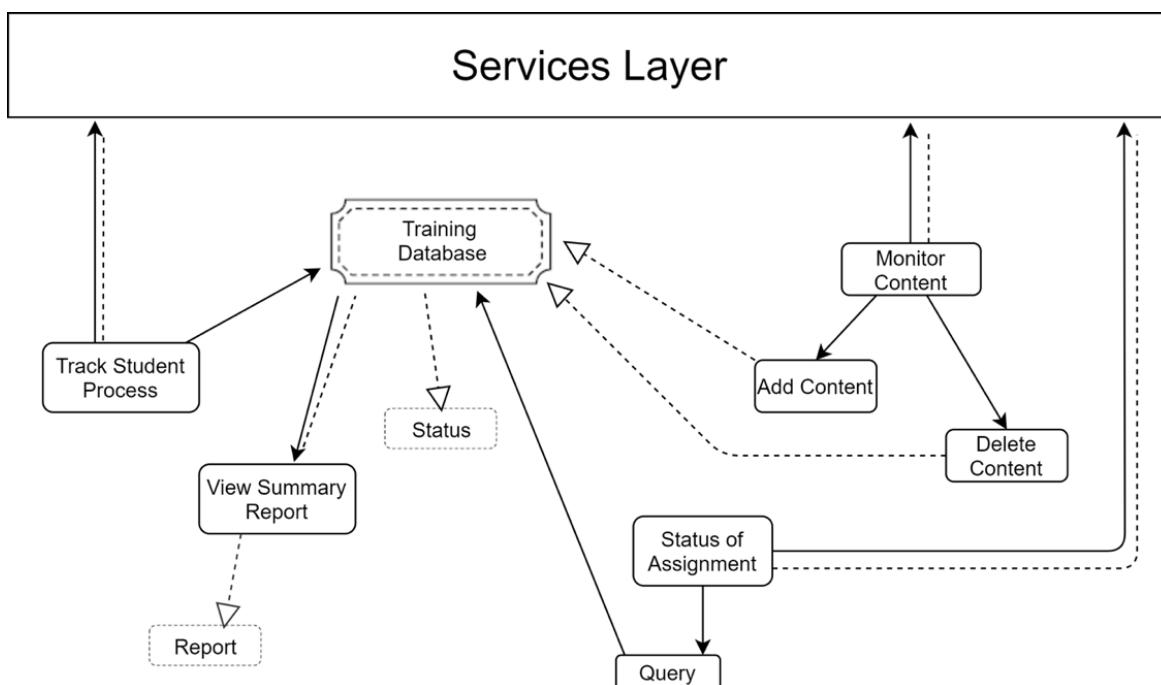


9.2. Framework Layer

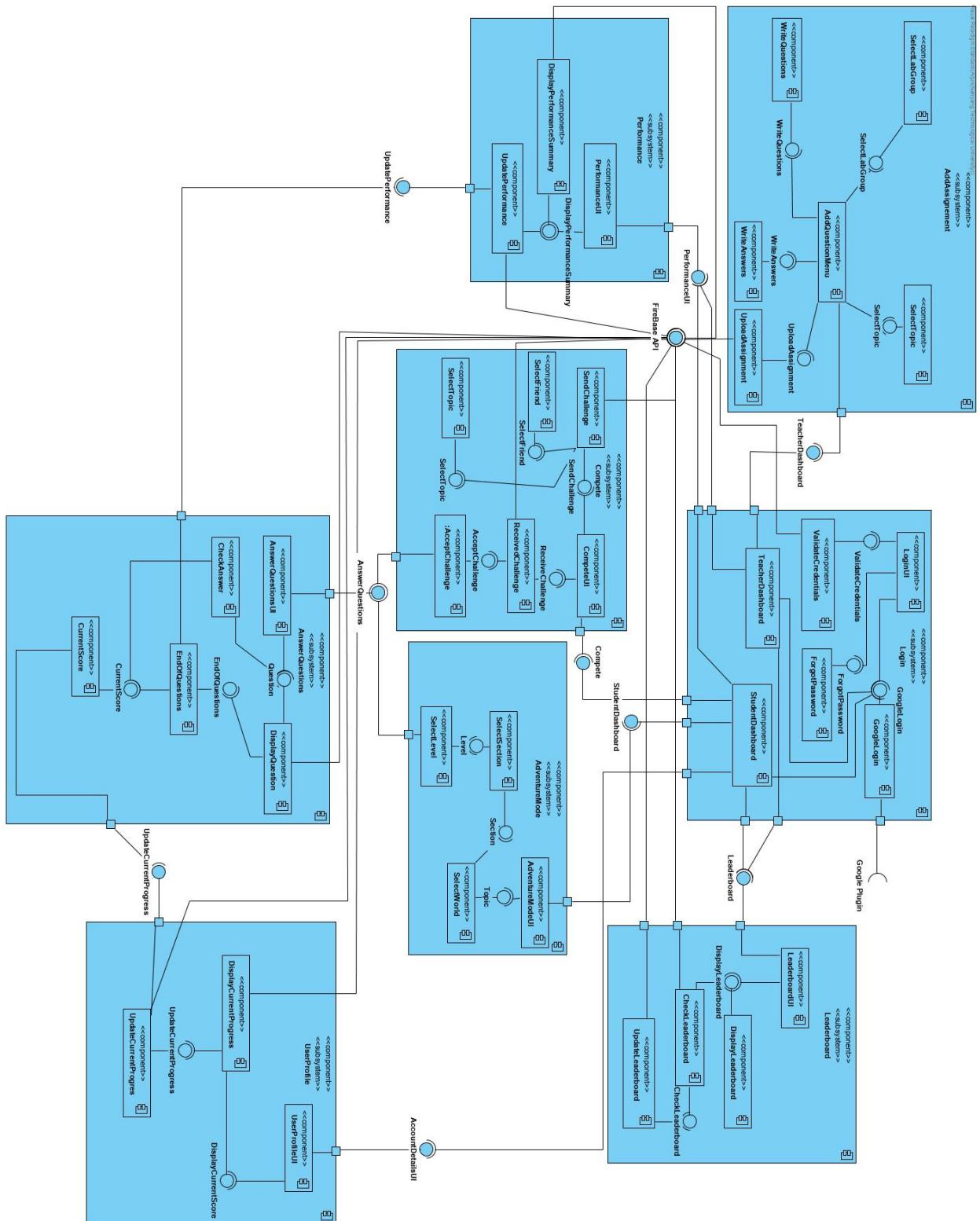
a. Sections, Levels, & Game Modes



b. Student Progress, Content & Assignments

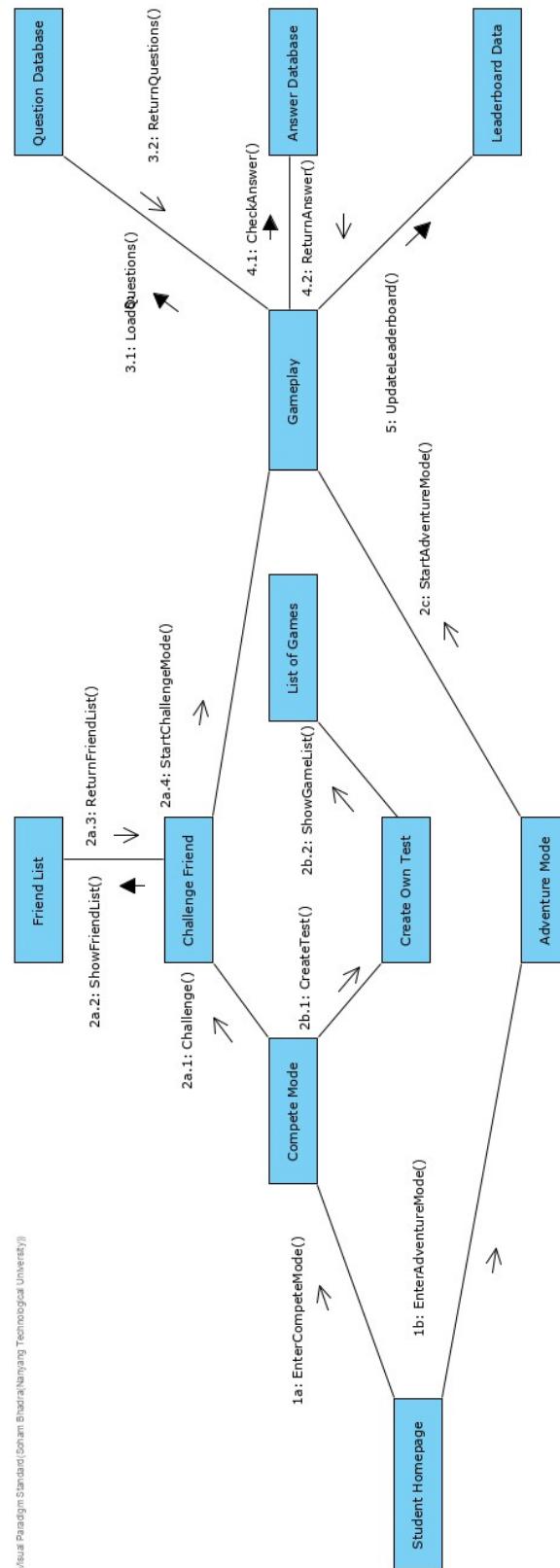


10. Component Diagram

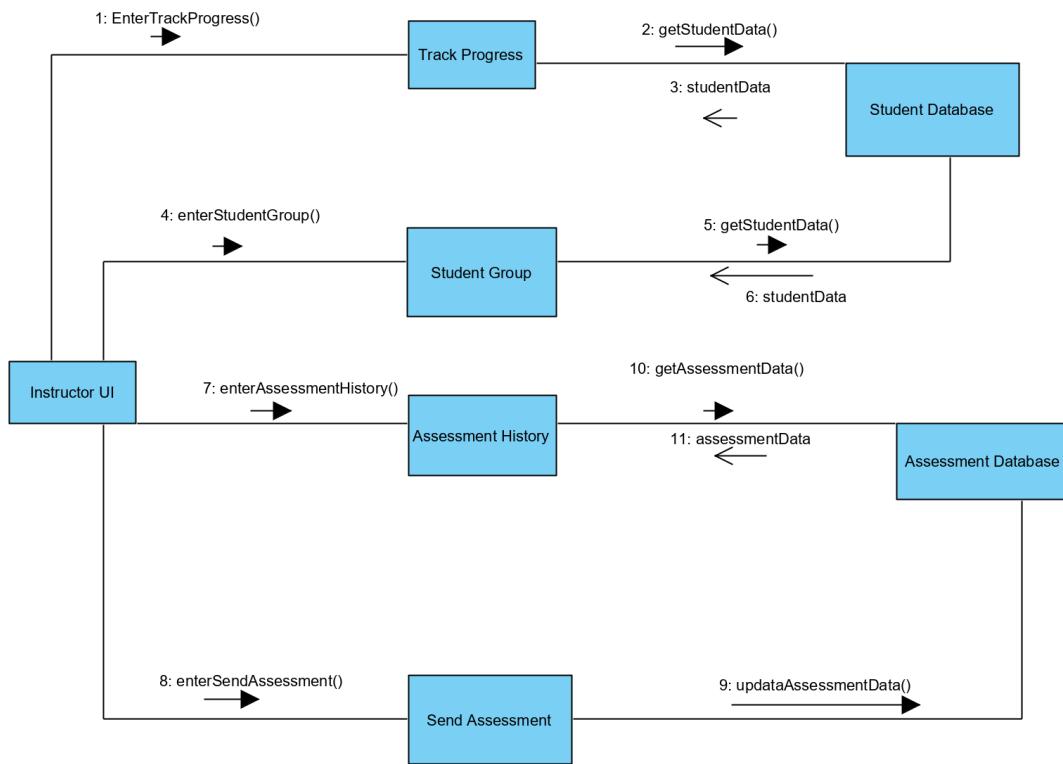


11. Communication Diagram

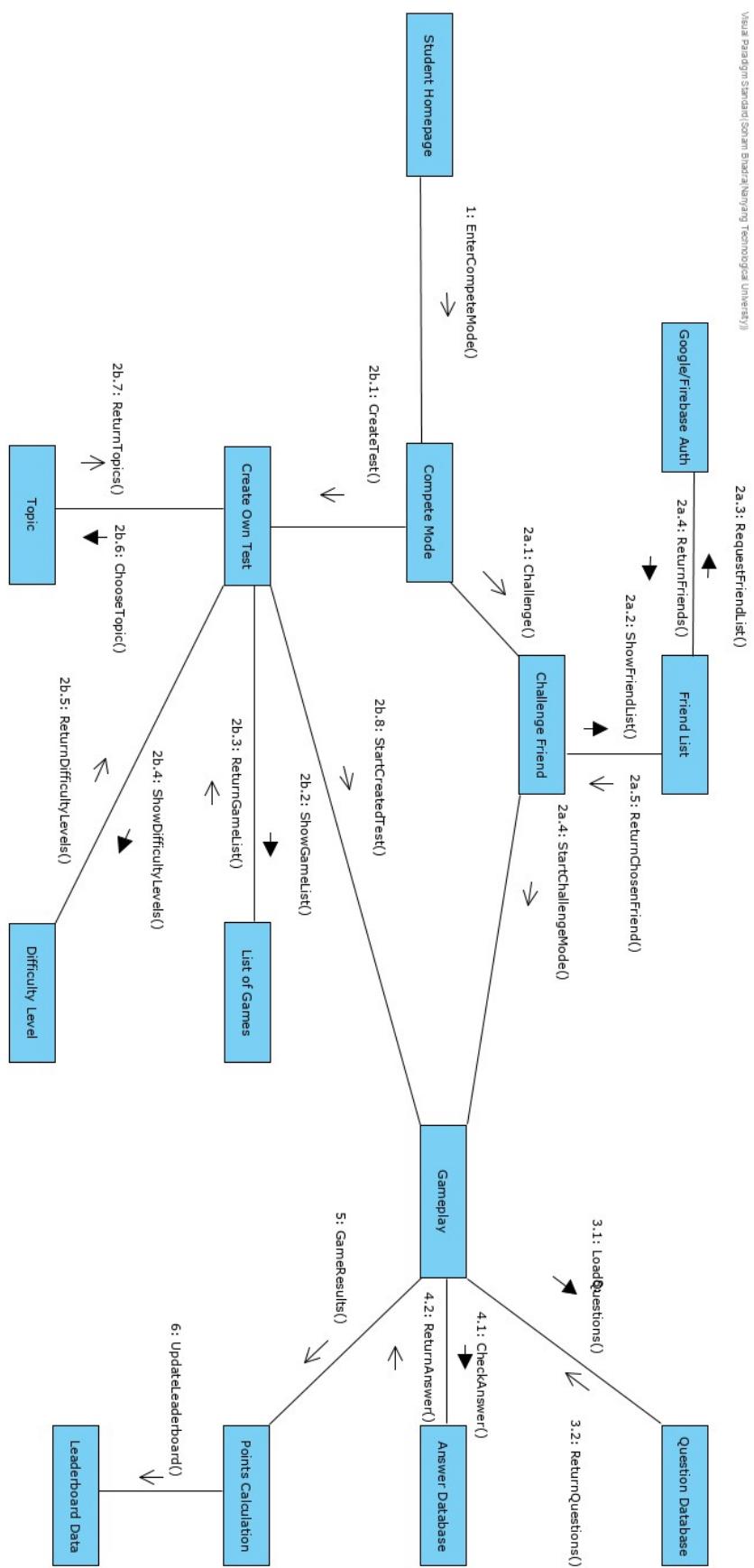
11.1 Student Management



11.2. Instructor Management



11.3. Compete Mode

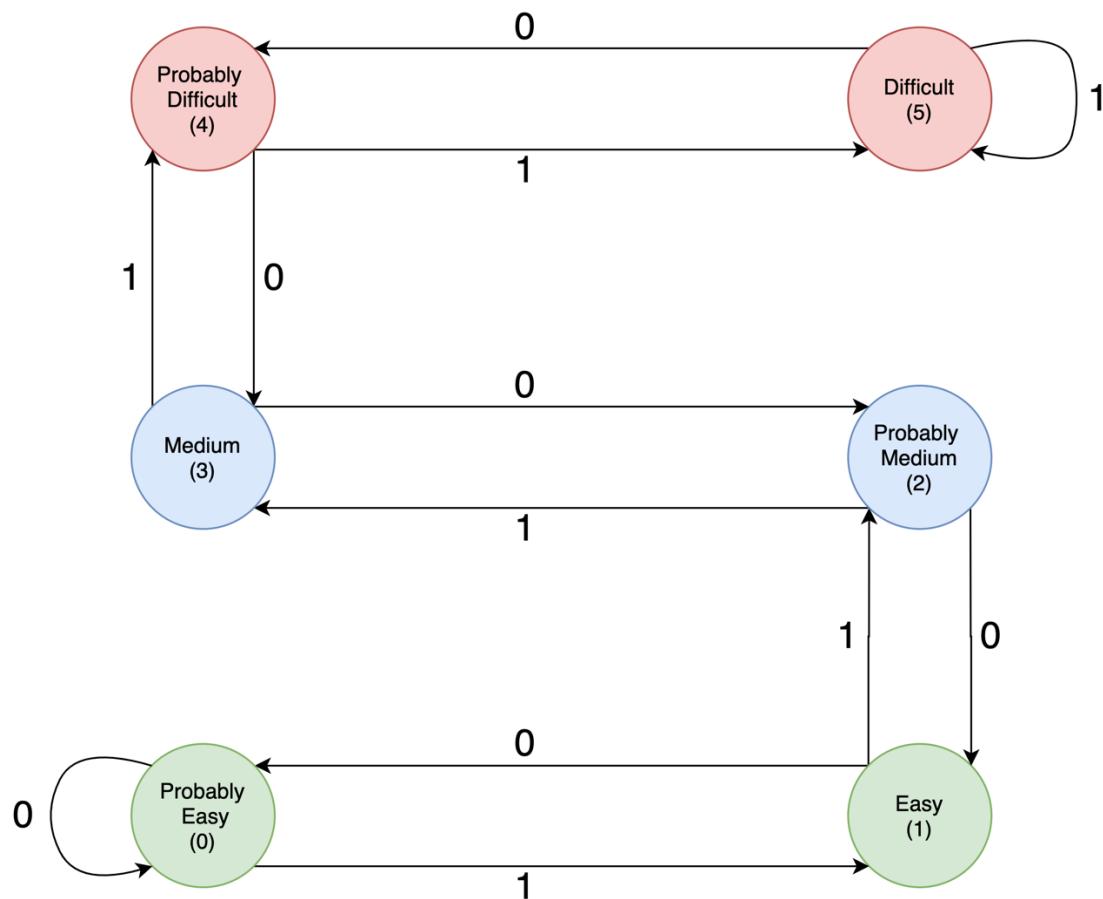


12. Real Time Data Analytics

12.1. Student Mode

The Adventure Mode of the application consists of different Worlds, where each World represents a phase of a Software Engineering Life Cycle. Every World consists of different Sections comprising Multiple Choice, Fill-in-the-Blanks and Match the Column type questions.

Based on this real-time analysis results, the questions of the levels are customized. For example, if a player has a good command of the course according to the playing history, the player shall deal with more difficult questions when playing the game. To track a student's progress in real-time, the difficulty of questions is dynamically predicted as the student progresses through the Section and provide instantaneous feedback. This is achieved with the use of a **Two Bit Dynamic Branch Predictor**.



- Each Circle represents a state with the name and the state number.
- Each arrow represents a transition between two states with the condition specified on the arrow.
- The Conditions are:
 - ‘1’ – The current Question was correctly answered in the first attempt.
 - ‘0’ – Otherwise
- The initial state is taken to be 0 (Probably Easy).

With such an implementation, the Dynamic Branch Predictor is easily able to determine the expertise of a student on a particular topic and the same feedback can be provided to the Instructors for easy monitoring. The game analysis all the players' playing history to continuously obtain the overall mastery of the course in real time, and produce a summary report for the teachers (including which parts are well-mastered by the students, and which parts are not).

Based on this report, teachers can logically adjust the teaching contents and key points during the classroom teaching.

12.2. Instructor Mode

The game analyses all the players' playing history to continuously obtaining the overall mastery of the course in real time, and produce a summary report for the teachers (including which parts are well-mastered by the students, and which parts are not). Based on this report, teachers can logically adjust the teaching contents and key points during the classroom teaching.

The Adventure Mode of the application consists of different Worlds, where each World represents a phase of a Software Engineering Life Cycle. Every World consists of different Sections comprising Multiple Choice, Fill-in-the-Blanks and Match the Column type questions.

As discussed in the Data Analytics section of Student Mode, the questions given to the students vary in difficulty. The questions are categorized along with their points as follows:

Question Type	Points
Easy	1
Medium	Points(Easy) + 1 = 2
Hard	Points(Medium) + 1 = 3

As a student progresses through different Sections of different Worlds, they accumulate points. With each Section representing a sub-topic of a phase of SDLC, the instructor can obtain a overview of a student's mastery in different Worlds and Sections.

All the statistics generated by the game can be viewed by logging into the application as an Instructor and tapping on 'Track Progress' card in the Instructor's Dashboard.

13. Testing

13.1. Student Mode

Test Case ID	Action	Description	Functionality to be tested	Expected output	Actual Output	P/F
TC_AM_001	The user chooses their avatar	User can choose which avatar represents them after logging in	Avatar choice	Save the avatar in the user's homepage and display homepage	Save the avatar in the user's homepage and display homepage	P
TC_AM_002	The user pressed the 'Adventure' button on the home page	Display Adventure Mode	Adventure Mode World Overview	Open the screen which shows all the worlds	Open the screen which shows all the worlds	P
TC_AM_003	The user pressed the 'Adventure' button on the home page	Display locked and unlocked worlds	Adventure Mode - Accessible worlds	Show unlocked worlds with unlocked icon and locked worlds with locked icon	Show unlocked worlds with unlocked icon and locked worlds with locked icon	P
TC_AM_004	The user presses an unlocked world to enter it	Enter the unlocked world	Adventure Mode - Accessible Worlds Unlocked	Allow user to enter the world - Open the world page, viewing the different sections	Allow user to enter the world - Open the world page, viewing the different sections	P

TC_AM_005	The user presses a locked world	Block user from entering a locked world	Adventure Mode - Locked World	Prompt the user that they are not allowed to enter	Prompt the user that they are not allowed to enter	Fail
TC_AM_006	The user presses a particular section button in the world	Enter the section to start the quiz for that particular topic	Adventure Mode Section	Open the section page, allowing the user to start the quiz	Open the section page, allowing the user to start the quiz	P
TC_Qui_z_001	The user has entered the section	Display random questions from pool of questions consisting of MCQ Type, fill in the blank and column matching	Adventure Mode Quiz questions	Random Questions displayed one by one	Random Questions displayed one by one	P
TC_Qui_z_002	The user chooses correct answer	For MCQ questions, four answer options will be provided and system has to check whether user has clicked on	Answer chosen by user correct/incorrect	The answer button turns green if user has chosen correct answer and proceeds to next question	The answer button turns green if user has chosen correct answer and proceeds to next question	P

		correct answer				
TC_Qui z_003	The user chooses incorrect answer	For MCQ questions, four answer options will be provided and system has to check whether user has clicked on correct answer	Answer chosen by user correct/incorrect	The answer button turns red if user has chosen incorrect answer and prompts user to try again	The answer button turns red if user has chosen incorrect answer and prompts user to try again	P
TC_Qui z_004	The user types an answer	For Fill in the blank questions, the user has to input their answer in words and system has to check whether their answer is correct	Answer typed by user correct/incorrect	If user has typed the correct answer, the next question is shown	If user has typed the correct answer, the next question is shown	P
TC_Qui z_005	The user types an answer	For Fill in the blank questions, the user has to input their answer in words and	Answer typed by user correct/incorrect	If user has typed the incorrect answer, the user is prompted to try again	If user has typed the incorrect answer, the user is prompted to try again	P

		system has to check whether their answer is correct				
TC_Qui_z_006	The user matches answers in two columns by dragging and dropping	For match the column type of questions, system has to check if user has matched all terms appropriately by dragging and dropping tiles	Answers matched by user correct/incorrect	If user has matched all tiles correctly, the next question is shown	If user has matched all tiles correctly, the next question is shown	P
TC_Qui_z_007	The user completes all the questions of a quiz	Once user completes all questions, they are done with a particular section	Section completion	If user has completed all questions, then system prompts that section is completed and the next section is unlocked	If user has completed all questions, then system prompts that section is completed and the next section is unlocked	P

TC_Qui z_008	The user completes all the sections in a world	Once user completes all questions of all sections, then a world is complete, and user can proceed to the next world	World completion	If user has completed all section, then system prompts that world is completed and the next world is unlocked	If user has completed all section, then system prompts that world is completed and the next world is unlocked	P
TC_Qui z_009	The user completes some sections and some worlds	Once user completes some worlds and sections, the progress tab is updated with number of worlds, sections and levels completed	Tracking Progress	The profile page of the user shows how many worlds, sections and levels the user has completed	The profile page of the user shows how many worlds, sections and levels the user has completed	P
TC_Co mpete_ 001	The user clicks on the 'Compete Mode' option	Check if user's friends can be viewed	View Friends List from Facebook/ Google	The screen should display a list of the user's friends with whom they can compete		P

TC_Co mpete0 02	The user clicks on the 'Compete Mode' option	Check if the user has received any challenge from another user	Challenging players	The screen should display that another user has challenged this user	The screen should display that another user has challenged this user	P
TC_Co mpete0 03	The user clicks on 'Accept Challenge'	The user must be able to attempt the quiz after clicking 'Accept Challenge'	Quiz attempt	The screen should display MCQ questions one by one	The screen should display MCQ questions one by one	P
TC_Co mplete_ 004	The user clicks on answer options	User must be able to choose their answers	Compete Mode Quiz	The system marks the answer box green if answer is correct and red if incorrect	The system marks the answer box green if answer is correct and red if incorrect	P
TC_Co mpete_ 005	User completes the quiz	Check if points have been calculated appropriately	Points Calculation	The system must display the total number of points gained after the user has completed the quiz	The system displayed the total number of points gained after the user completed the quiz	P

TC_Leaderboard_001	User presses the 'Leaderboard' button on the homepage	Check if leaderboard can be loaded	Leaderboard	Show the leaderboard with player names and their points in order of highest to lowest points	Show the leaderboard with player names and their points in order of highest to lowest points	P
TC_Assess_001	User presses the 'Assessment' button on the homepage	Check present and past assessments	Assessment	Show present assessments and details under 'Present' and past assessments under 'History'	Show present assessments and details under 'Present' and past assessments under 'History'	P

13.2. Instructor Mode

Test Case ID	Action	Description	Functionality to be tested	Expected output	Actual Output	P/F
TC_IM_01	User logs in with their Gmail ID	After logging in, check if system displays the instructor dashboard	View Instructor Dashboard	The instructor dashboard is displayed	The instructor dashboard is displayed	P
TC_IM_02	User taps on 'Create Student Group'	Check if system allows the user to create student groups	Create student groups	The screen displays a list of student groups and at the end has a button which says 'Add a group'	The screen displays a list of student groups and at the end has a button which says 'Add a group'	P
TC_IM_03	User taps on 'Add a group'	Check if system allows the user to add students to a group	Create student groups	The screen asks user for Course ID and group ID		P
TC_IM_04	User types in the course ID and the group ID	Check if system allows the user to add students to a group	Create student groups	The screen shows the option to add members for a particular group	The screen shows the option to add members for a particular group	P
TC_IM_05	User presses on icon to	Check if system allows the user to add	Create student groups	The screen shows a list of students and the user can	The screen shows a list of students and the user can	P

	add members	students to a group		can choose multiple students to add to the group	choose multiple students to add to the group	
TC_IM_06	User presses 'confirm' after selecting student names	Check if system allows the user to add students to a group	Create student groups	Group is created and added to the list of available groups	Group is created and added to the list of available groups	P
TC_IM_07	User presses on 'Send Assessment' in the dashboard	Check if system allows instructor to create and send assessment	Send Assessment	Screen is shown where user can input Course name/ID, topic, and due date	Screen is shown where user can input Course name/ID, topic, and due date	P
TC_IM_08	User inputs all details and clicks 'Next'	Check if system allows user to type in the questions and answers	Create Assessment	Screen is shown where user can input the questions and answers	Screen is shown where user can input the questions and answers	P
TC_IM_09	User inputs the questions and answers and then clicks 'Next'	Check if system allows instructor to create and send assignment	Choose student group to send assessment	Screen is shown where user can select which student group to send the assessment to	Screen is shown where user can select which student group to send the assessment to	P

TC_IM_0 10	User selects the student group to which assessment has to be sent	Check if system allows instructor to send assignment	Send Assessment	Screen prompts the user that the assessment has been sent successfully	Screen prompts the user that the assessment has been sent successfully	P
TC_IM_0 11	User presses on 'Track Progress' in the dashboard	Check if system shows the user progress of each student	Track Progress	Screen displays a list of all registered students	Screen displays a list of all registered students	P
TC_IM_0 12	User selects a student	Check if system shows the user progress of each student	Track Progress	Screen displays graphs of the student's progress in worlds, sections and levels	Screen displays graphs of the student's progress in worlds, sections and levels	P
TC_IM_0 13	User presses on 'Assessment History' in the dashboard	System allows the user to view past assessments and their details	Assessment History	Screen displays a list of present assessments under 'Present' and the past assessments under 'History'	Screen displays a list of present assessments under 'Present' and the past assessments under 'History'	P

				under 'History'		
TC_IM_0 14	User presses on 'Invite Students via Whatsapp' in the dashboard	Invite students through Whatsapp	Social Media function	On pressing the button, a slide up menu appears where users can send a message to invite students via Whatsapp	On pressing the button, a slide up menu appears where users can send a message to invite students via Whatsapp	P

13.3. Student Login

Test Case ID	Action	Description	Functionality to be tested	Expected output	Actual Output	P/F
TC_SL_001	A student clicks the icon 'Sign in with Google'	A student is signing in his/her account using google sign in by entering email id.	Student Login	Google's 'Sign In' page opens asking for the student email id as an input	Google Sign In page opens asking for the student email id as an input	P
TC_SL_002	The student clicks on the 'next' icon on google sign in page to enter the password	A student is signing in his/her account using google sign in by entering password.	Student Login	Google's 'Update Account' page opens	Google 'Update Account' page opens	P
TC_SL_003	The student clicks on the already registered account.	If the student has already registered the account, the student can directly select the avatar.	Student Login	Successful Login, the system gives the option to choose an avatar.	Successful Login, the system gives the option to choose an avatar.	P
TC_SL_004	The student clicks on the 'Yes I'm	A student is signing in his/her	Student Login	Google's 'Welcome' page opens	Google 'Welcome' page opens	P

	in' icon on google's 'Account Update' page	account using google sign in.				
TC_SL_005	The student clicks on the 'Agree' icon on google's 'Welcome' page	A student is signing in his/her account using google sign in.	Student Login	Google's 'Google Services' page opens	Google's 'Google Services' page opens	P
TC_SL_006	The student clicks on the 'Agree' icon on google's 'Google Services' page	A student is signing in his/her account using google sign in.	Student Login	Successful Login, the system gives the option to choose an avatar.	Successful Login, the system gives the option to choose an avatar.	P
TC_SL_007	The student clicks on the avatar he/her wants.	Avatar selection	Student Login	Student dashboard is shown.	Student dashboard is shown.	P

13.4. Instructor Login

Test Case ID	Action	Description	Functionality to be tested	Expected output	Actual Output	P/F
TC_IL_001	An Instructor clicks the icon 'Sign in with Google'	An instructor is signing in his/her account using google sign in by entering email id.	Instructor Login	Google's 'Sign In' page opens asking for the instructor email id as an input	Google Sign In page opens asking for the instructor email id as an input	P
TC_IL_002	The instructor clicks on the 'next' icon on google sign in page to enter the password	The instructor is signing in his/her account using google sign in by entering password.	Instructor Login	Google's 'Update Account' page opens	Google 'Update Account' page opens	P
TC_IL_003	The instructor clicks on the already registered account.	If the instructor has already registered the account, the instructor can see the instructor dashboard .	Instructor Login	Successful Login, the system displays the instructor dashboard.	Successful Login, the system displays the instructor dashboard.	P

TC_IL_004	The instructor clicks on the 'Yes I'm in' icon on google's 'Account Update' page	The instructor is signing in his/her account using google sign in.	Instructor Login	Google's 'Welcome' page opens	Google 'Welcome' page opens	P
TC_IL_005	The instructor clicks on the 'Agree' icon on google's 'Welcome' page	The instructor is signing in his/her account using google sign in.	Instructor Login	Google's 'Google Services' page opens	Google's 'Google Services' page opens	P
TC_IL_006	The instructor clicks on the 'Agree' icon on google's 'Google Services' page	The instructor is signing in his/her account using google sign in.	Instructor Login	Successful Login, the system displays the instructor dashboard.	Successful Login, the system displays the instructor dashboard.	P

13.5. Settings

Test Case ID	Action	Description	Functionality to be tested	Expected output	Actual Output	P/F
TC_S_001	The user clicks the icon 'Configuration'	The user tries to change the Wifi, Display, Sound, App Settings. The user may also want to view his/her profile.	Edit configuration	Systems displays the page giving the option of viewing profile or change settings	Systems displays the page giving the option of viewing profile or change settings	P
TC_S_002	The user clicks on the 'Profile'	The user views his/her profile	View profile	Profile page opens	Profile page opens	P
TC_S_003	The user clicks on the 'Settings' icon	The user wishes to change the Wifi, Display and Sound settings and view app information	Change settings	Settings page opens	Settings page opens	P
TC_S_004	The user clicks on the 'Wifi' icon	The user wishes to change the	Change Wifi settings	OS Wifi settings page open	OS Wifi settings page open	P

		Wifi settings				
TC_S_005	The user clicks on the 'Sound' icon	The user wishes to change the Sound settings	Change Sound settings	OS Sound settings page open	OS Sound settings page open	P
TC_S_006	The user clicks on the 'Display' icon	The user wishes to change the Display settings	Change Display settings	OS Display settings page open	OS Display settings page open	P
TC_S_007	The user clicks on the 'App Info' icon	The user wishes to view app information	View app information	OS App info page open	OS App info page open	P

14. Appendix

1. High resolution version of all the images and diagrams used in this SRS (Architecture Diagrams, Context Diagrams, etc.) can be found in the SVN submission, under the folder ‘images’ (Kindly note that NTU VPN is required to access the webpage outside of NTUSECURE network) – (<https://155.69.100.25:443/svn/CZ3003/SS5/TheChallengers>)
2. For the detailed code implementation of SSADPro, please refer our GitHub Repository at the following link – <https://github.com/guptajay/SSADPro>
3. For a detailed demo and walkthrough of SSADPro, please refer to the demo video uploaded in YouTube at the following link. – <https://youtu.be/QkArKSREvQo>
4. For a detailed explanation of the architecture of SSADPro, please refer to the PowerPoint presentation at the following link. – https://entuedu-my.sharepoint.com/:p/g/personal/jay002_e_ntu_edu_sg/ETWdJjZaDZtErecNhfb6GGoBWals_HhrU1Mq1u11oZB94A?e=Z8IYkQ
5. The contents of this Software Requirement Specification are also available at the following MediaWiki Link (Kindly note that NTUs VPN is required to access the webpage outside of NTUSECURE network) – http://155.69.100.27/3003s21920_TheChallengers/index.php/Main_Page
6. Visual Paradigm UML Software (<https://www.visual-paradigm.com/>), Draw.IO (<https://app.diagrams.net/>) and Lucidchart (<https://www.lucidchart.com/>) are used to generate all UML and other architectural diagrams.
7. Flutter (<https://flutter.dev/>) and Dart (<https://dart.dev/>) are used for the codebase of the application.
8. Google Firebase is used as the database system for the application – <https://firebase.google.com/>