Software Requirements Specification

for

GalaxSE

Team SpaceGo

Version 1.0 approved

School of Computer Science and Engineering (SCSE)
Nanyang Technological University, Singapore

Table of Contents

Revision History

S/N	Date	Reason for Changes	Version
1	24/08/2020	Initial Identified Functional Requirements	1.0
2	03/09/2020	Added Use Case Diagram and Description	1.1
3	30/09/2020	Architecture Diagram and Subsystem	1.2
4	10/10/2020	Updated Functional requirements	1.3
5	12/10/2020	Communication Diagram	1.4
6	20/10/2020	Component Diagram	1.5
7	25/10/2020	Unit Testing, Load Testing, Performance Testing	1.6
8	06/11/2020	Updated Use Cases	1.7
9	08/11/2020	Updated Diagrams	1.8

1. Introduction

1.1 Purpose

This document outlines the plan for the development of GalaxSE, a game application by Team SpaceGo. The intended readers of this document are current and future developers of GalaxSE. The SRS serves to describe the scope, functional and non-functional software requirements of GalaxSE consists, but is not limited to, a summary of the system functionality, the functional and non-functional requirements of the system, use-case models, User Interface model and other data-flow diagrams.

1.2 Document Conventions

The following describes the style and format used for this document.

- Main Section Heading:
  - Font: Times New Roman
  - Face: Bold
  - Size: 18
- Subsection Heading:
  - Font: Times New Roman
  - Face: Bold
  - Size: 14
- Main Body Content:
  - Font: Times New Roman
  - Face: Normal
  - Size: 11

1.3 Intended Audience and Reading Suggestions

This project is a prototype for the GalaxSE system, which consists of the GalaxSE Game application and Teachers WebApp web application. It is made available to Teachers and Students who are enrolled in Nanyang Technological University. This system aims to gamify and socialize teaching and learning of software engineering courses.

This document is intended for the users, developers and testers.

Users of the GalaxSE System include Teachers and Students. This document serves to inform users on the main functionality of the system and give a clearer overview of the software requirements. External Interface Requirements (Section 3) and System Features (section 4) are most useful to the users.

Developers and Project Managers that are involved in the maintenance or enhancement of the application will find this document useful to review the capabilities and understand the individual subsystems of the game. This document allows developers to determine which section of the system should be focused for

improvements or modification. It is recommended that developers begin reading from the introduction (Section 1).
Testers of the application, which includes Quality Assurance (QA) Engineers, should use this document as a guideline to create relevant test cases. This document serves as a starting point for testers to understand the entire system and find important areas to conduct tests. Testers should focus on the System Features (Section 4) and Non-Functional Requirements (Section 5) for system testing and requirements verification.

1.4 Product Scope

GalaxSE System consists of a desktop game application and a web application. The GalaxSE Game application will facilitate the teaching and learning of Software Engineering courses through a gamified and socialized approach. It aims to stimulate the learning interests of Students by providing a fun and interactive learning platform that Students can use outside of the classroom. Furthermore, GalaxSE Game comes with analysis features that use machine learning to enhance Students' learning experience by adjusting the difficulty of the game to ensure Students are progressing at their own pace.

The Teachers WebApp provides convenience to Teachers through its analysis tool and assignments feature. It allows Teachers to manage and test their Students through custom-made assignments. It provides an efficient analytical tool that allows Teachers to understand the Students' game progress and assess their overall mastery of the course. This information can be used for adjusting teaching content and key points during classroom teaching. Lastly, it allows the Teachers to make changes to the GalaxSE Game application if any modification of game content is necessary.

Currently, this project is being developed for Students and Teachers of the School of Computer Science and Engineering (SCSE) of Nanyang Technological University (NTU). The larger goal for this project would be to incorporate gamified learning in Computer Science schools across the globe.

1.5 References

The following are web addresses on how to implement a suitable format and standard.

- Unity development: https://docs.unity3d.com/Manual/index.html
- MongoDB Development: https://docs.mongodb.com/
- Reddit API: https://www.reddit.com/dev/api/
- Twitter API: https://developer.twitter.com/en/docs
- Auth0 API: https://auth0.com/docs/api

2. Overall Description

2.1 Product Perspective

Our product aims to gamify and socialize teaching and learning of software engineering courses for users and provide a convenient platform for Teachers to send assignments and be updated on Students' progress. The GalaxSE system consists of 2 applications, GalaxSE Game application and Teachers WebApp application. GalaxSE uses a space theme concept, with the Galaxies representing the four stages of SDLC (Software Design Life Cycle), Planets representing the topics for each stage and Monsters contains a group of questions. The questions will be chosen using machine learning based on the Student's past performance, and subsequent questions will be chosen through adaptive learning. The Teachers WebApp serves as an analysis tool for Teachers to analyze the Students or lab groups progress, send assignments to them and edit the question bank.

Adaptive learning is implemented by generating a new level question using prediction output by our Machine Learning Classifier. As the user plays the game, whether the question is hard or wrong determines the difficulty of the next questions.

Student accounts will be created by the Teacher to access the system. Students have access to the GalaxSE Game application and the 2 game modes, “Explore Solo Mission” and “Create Challenge Monster”. Explore Solo Mission allows Students to learn by completing questions and defeating Monsters, allowing them to explore the various phases in the software development life cycle. Create Challenge Monster allows Students to design challenges that will be sent to other Students via social media, promoting learning through socializing. Challenges created by other Students and Assignments created by Teachers are to be completed in the GalaxSE Game.

Teachers have access to the Teachers WebApp. They will be able to create assignments and send them to Students via social media platforms Reddit or Twitter. Teachers are also able to view an analytics report of the Students' progress and the summary report for each Galaxy, enabling Teachers to have better understanding of the topics that Students are facing difficulties with and focusing on these topics during their teachings in school. Teachers are able to edit the questions used for the GalaxSE Game. Teachers are also in charge of account management for the Students.

2.2 Product Functions

The major functions are as follows:

- Students: Play the game in 3 different modes:
    - Explore Solo Mission: a progressive learning experience
    - Challenges: create and complete challenges posted by other Students
    - Assignments: complete assignments posted by Teachers
- Students: Create and design Astronaut
- Students: Send challenges to other Students through social media
- Students: Leaderboard shows the overall ranking of Students for Explore Solo Mission mode.
- Teachers: Create and manage the questions in Explore Solo Mission. This includes modification or addition of questions and options.
- Teachers: Create and manage the questions in Assignments
- Teachers: Send Assignments to Students through social media
- Teachers: Compute and display statistics from quiz responses
    - View individual questions responses
    - View overall question responses
- Teachers: Accounts management
    - Create Student account
    - Delete Student account

2.3 User Classes and Characteristics

The GalaxSE System's purpose is to improve the learning and teaching experience of software engineering topics for NTU Students. The system is mainly targeted at Students and Teachers who are keen on learning and teaching software engineering. There are two users involved in the GalaxSE system, Students and Teachers.

Students will be able to play the GalaxSE game in three different modes (Explore Solo Mission, Create Challenge Monster and Assignment Mode) to better understand each software engineering concept. The game serves as a platform for Students to practice questions, complete assignments and improve their knowledge. It also encourages learning through socializing as Students will be able to send challenges to each other and compare their score. Students will be using the system regularly to keep up with their learning schedule.

Teachers will be able to create different questions to test different topics of software engineering. They can make changes to the questions due to changing syllabus or increasing difficulty, which improves the teaching process. Teachers are able to send assignments to Students and access Students' results to understand their learning progress. They will be using the system regularly. Teachers are given privileges to manage the accounts of the system and create the Student accounts. They will only do so occasionally when necessary.

2.4 Operating Environment

1. Operating Platform for GalaxSE game
   - Development platform: Unity 2019.3
   - Operating system platform: Windows and Mac OS
2. Web Browser for Teachers WebApp
   - Google Chrome
   - Mozilla Firefox
   - Safari
   - Microsoft Edge
3. Database Server
   - MongoDB Server
4. Social Media API for sharing assignment and challenges
   - Reddit API
   - Twitter API

2.5 Design and Implementation Constraints

Some design considerations were made to ensure the project's success and completion within the deadline. However, these impose some implementation constraints for the users. The constraints include:

1. Concurrent Users: The system might not be able to accommodate a huge number of concurrent users as our system needs to communicate online to our database.
2. Lack of Assets: As this is a new system, the current assets for the GalaxSE Game application might be inadequate and lacking (Monster design, music and effects). Additional resources to improve the design of the game needs to be outsourced or created.
3. PC Only Application: The game application is running only on windows and Mac OS only. As such, users using devices that run on other operating systems (linux) will not be able to use the game application.
4. Language Support: The system is only available in English and not in any other languages.
5. Security Design: The GalaxSE Game application uses a simple authentication for login and might not be the most secure method for protecting Student's information. Additional security measures need to be outsourced.

2.6 User Documentation

User documentation components will be delivered along with the software. Listed below are the components that will be delivered:

- Teacher WebApp user manual
- Student GalaxSE Game user manual

Separate user manuals will be given to the Teachers and Students. The user manual consists of an overview of all GalaxSE's features as well as the step-by-step guide on the complete configuration of the required software and hardware for it.

2.7 Assumptions and Dependencies

2.7.1 Project Schedule

Due to the tight deadline required by the client, the team must complete and deliver GalaxSE in 13 weeks. After 13 weeks, the application must be available. The application must also fulfill all functional and non-functional requirements

2.7.2 Authentication and Connectivity

All users are required to have Internet access in order to use any of the functionalities provided by our application. This is because our application is required to constantly perform authentication of the user's account and retrieval or modification of existing application data such as questions which reside on the remote database.

2.7.3 Users Hardware Capability

Users should have a desktop or laptop capable of running the application.

2.7.4 Software Support

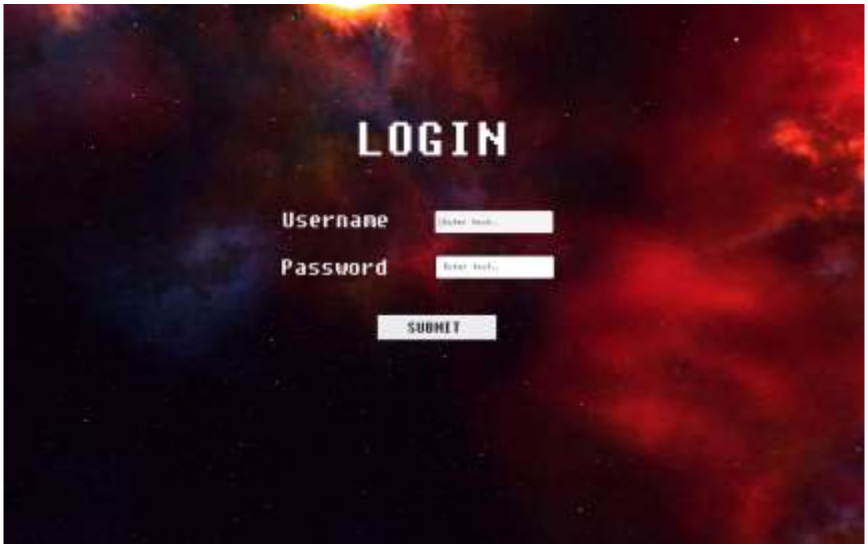
Unity will continue the support for game development so further development of GalaxSE can be implemented in the future.

3. External Interface Requirements

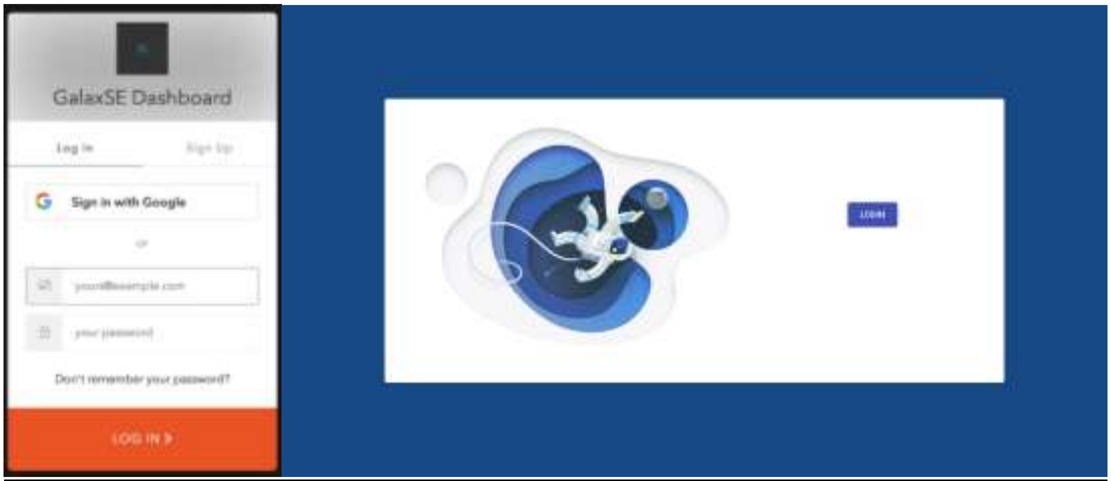
3.1 User Interfaces

3.1.1 Login Page

Allow users to log in using his/her username and password. Login credentials will be validated against what was stored in the database. Upon successful login, the user will be redirected to the respective interface depending on the system they are login into.



GalaxSE Game Login Page

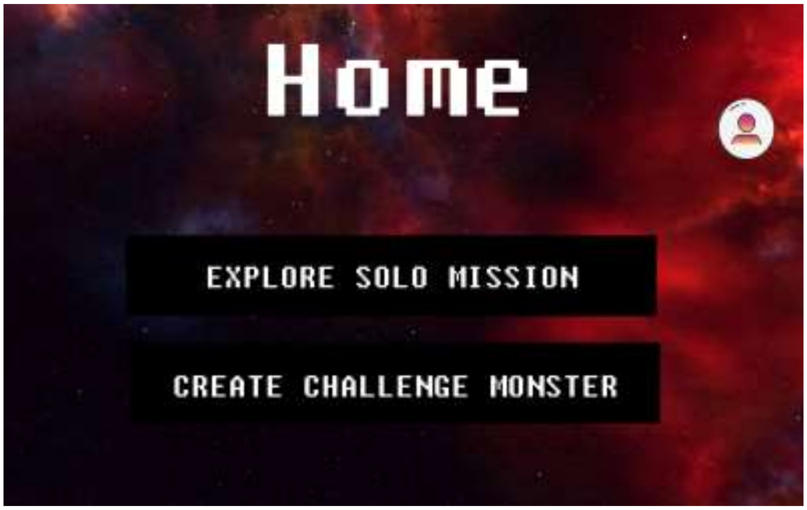


Teachers WebApp Login Page

3.1.2 GalaxSE Game Application: Home Page

The Home Page allows the Student to choose 3 different pages. (Explore Solo Mission, Create Challenge Monster, User Status Icon)

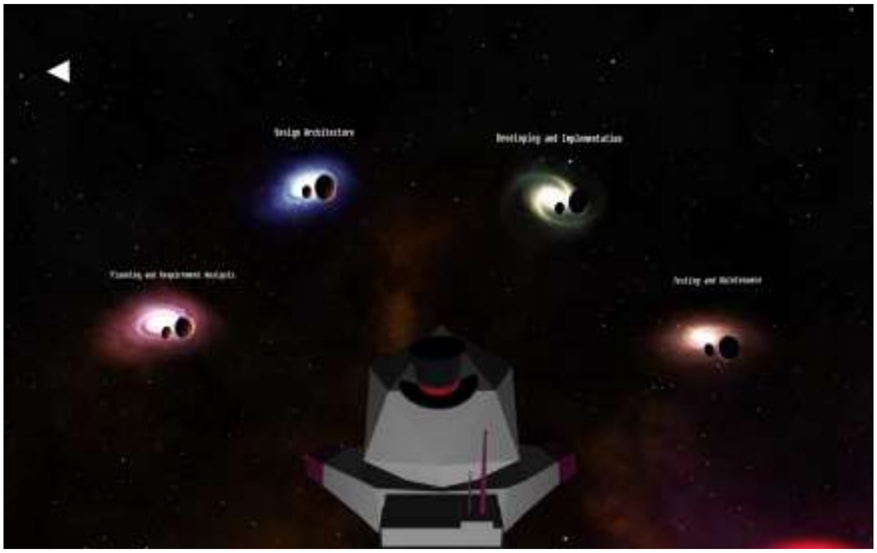
- Explore Solo Missions allows Students to select one Galaxy and Planet to play in order to learn about the development cycle and compete with other Students in the leaderboard.
- Create Challenge Monster allows Students to create a Monster which will contain 8 questions to challenge their friends through social media.
- User Status Icon allows Students to view profile, view leaderboard or log out.



GalaxSE Homepage

3.1.3 GalaxSE Game Application: Galaxy Selection Page

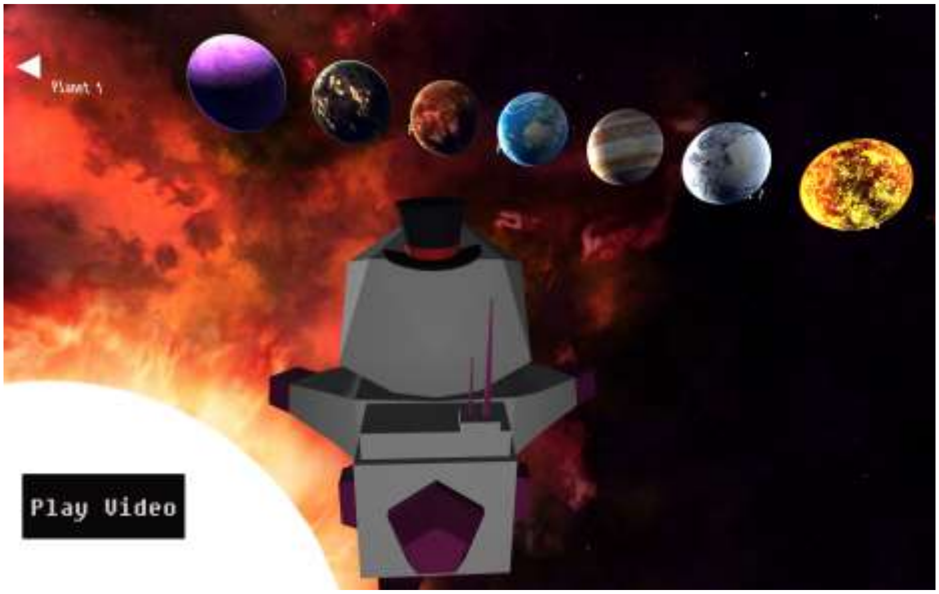
Students have to choose one of the four Galaxies (Planning and requirement analysis, design architecture, developing and implementation, testing and maintenance) in the Explore Solo Mission


Design Architecture
Developing and Implementation
Planning and Requirement Analysis
Testing and Maintenance

GalaxSE Galaxy Selection Page

3.1.4 GalaxSE Game Application: Planet Selection Page

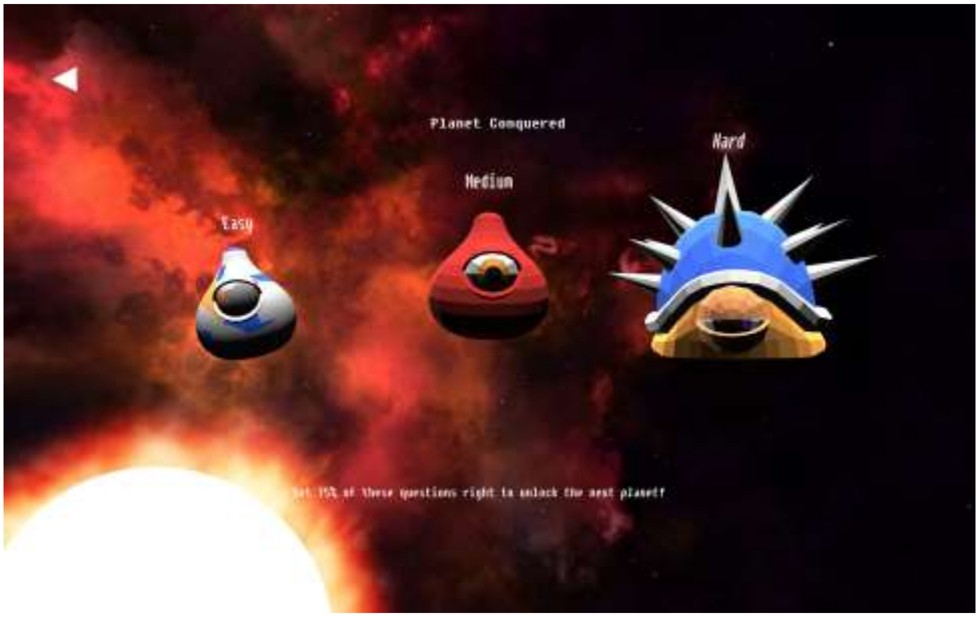
Students are allowed to choose one of the 8 Planets within each Galaxy.



GalaxSE Planet Selection Page

3.1.5 GalaxSE Game Application: Monster Selection Page

Students are allowed to choose one of the 3 Monsters within each Planet.

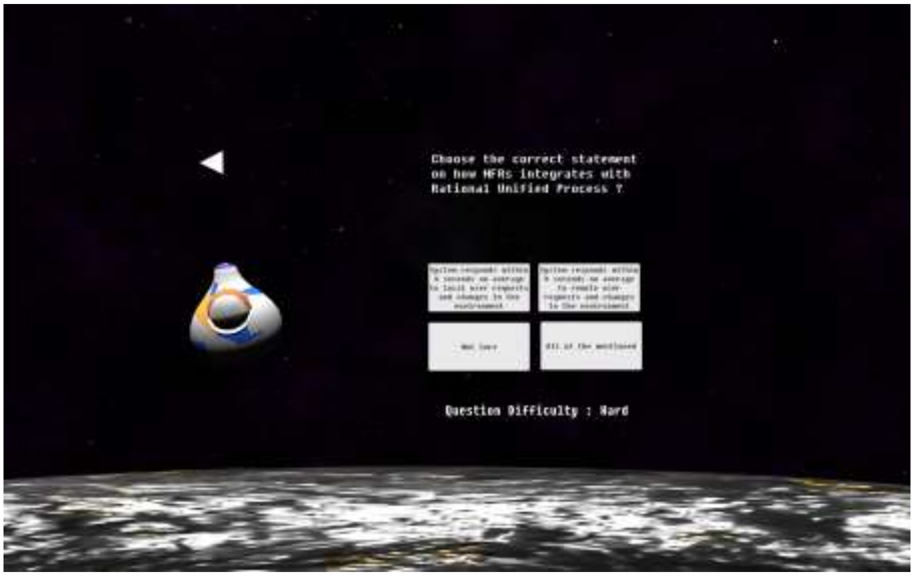


Get 75% of these questions right to unlock the next planet!

GalaxSE Monster Selection Page

3.1.6 GalaxSE Game Application: Monster Battle Page

Students must answer the question correctly to attack the Monster. Students who failed to answer the question correctly will be attacked by the Monster. The difficulty level of the question will be scaled according to Machine learning implemented in the GalaxSE Game.



Choose the correct statement on how NFRs integrates with Rational Unified Process ?

System responds within [illegible] seconds on average to local user requests and changes in the environment

System responds within [illegible] seconds on average to remote user requests and changes in the environment

Not sure

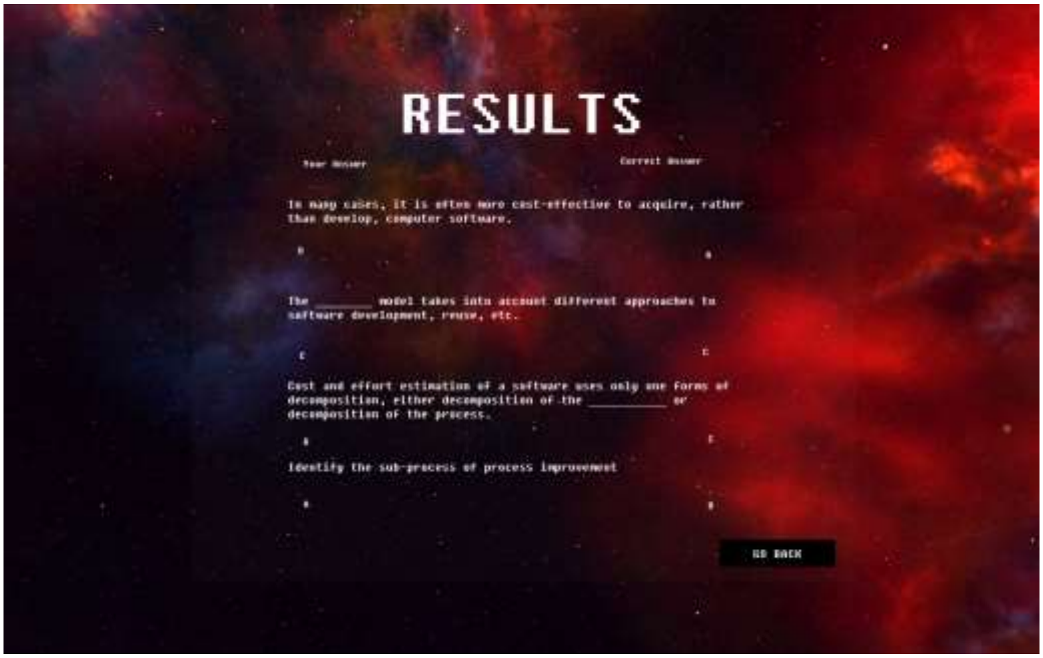
All of the mentioned

Question Difficulty : Hard

GalaxSE Monster Battle Page

3.1.7 GalaxSE Game Application: Results Page

Students will be able to review the questions and answers after battling with the Monster.



RESULTS

Your Answer Correct Answer

In many cases, it is often more cost-effective to acquire, rather than develop, computer software.

B A

The ________ model takes into account different approaches to software development, reuse, etc.

C C

Cost and effort estimation of a software uses only one forms of decomposition, either decomposition of the __________ or decomposition of the process.

B C

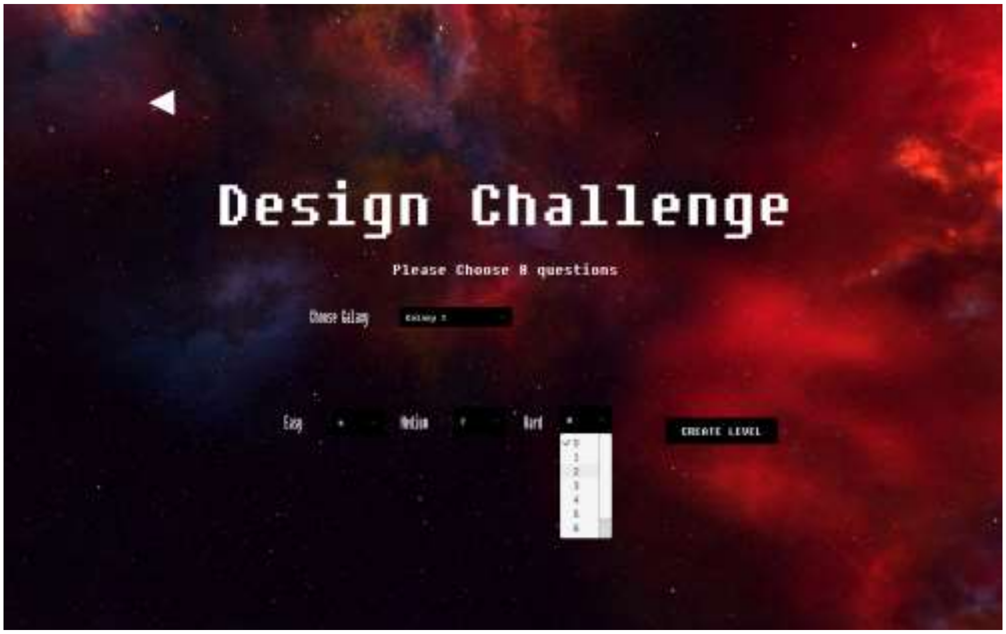
Identify the sub-process of process improvement

B B

GalaxSE Results Page

3.1.8 GalaxSE Game Application: Create Challenge Monster Page

Students will be able to customise the Challenge Monster. They will be able to choose a total of 8 questions of varying difficulties (easy medium hard). Next, the Student will have to answer the questions and receive a score. After which he/she can share the Challenge with other students on Twitter or Reddit.



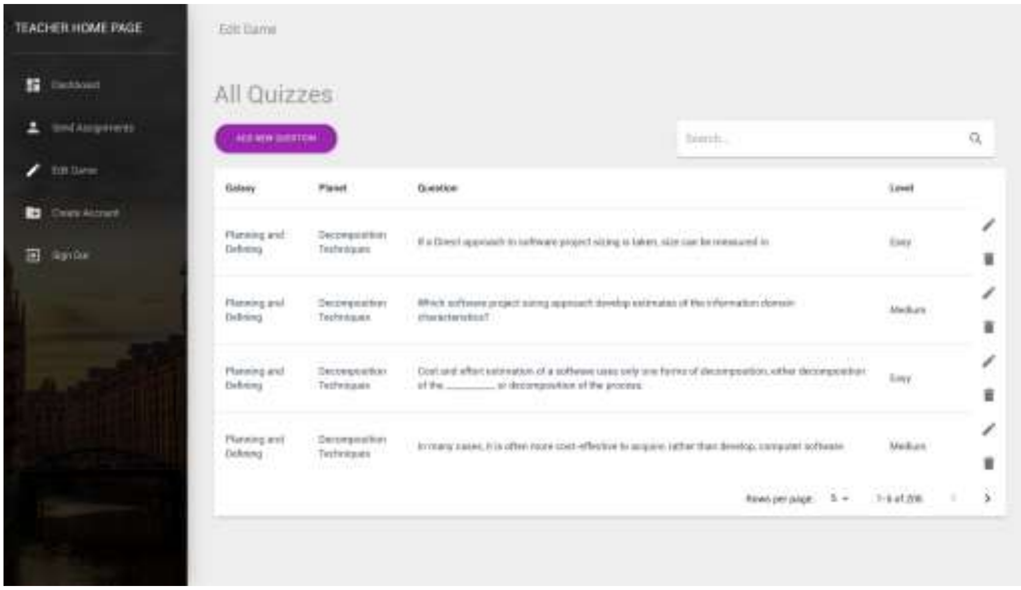
Design Challenge

Please Choose 8 questions

GalaxSE Create Challenge Monster Page

3.1.9 Teachers WebApp: Edit Game

Teachers will be able to edit the questions in the question bank. The teacher will be able to edit the description of the questions, options, answers, galaxy and planet. The teacher will be able to create new questions and delete existing questions.



All Quizzes

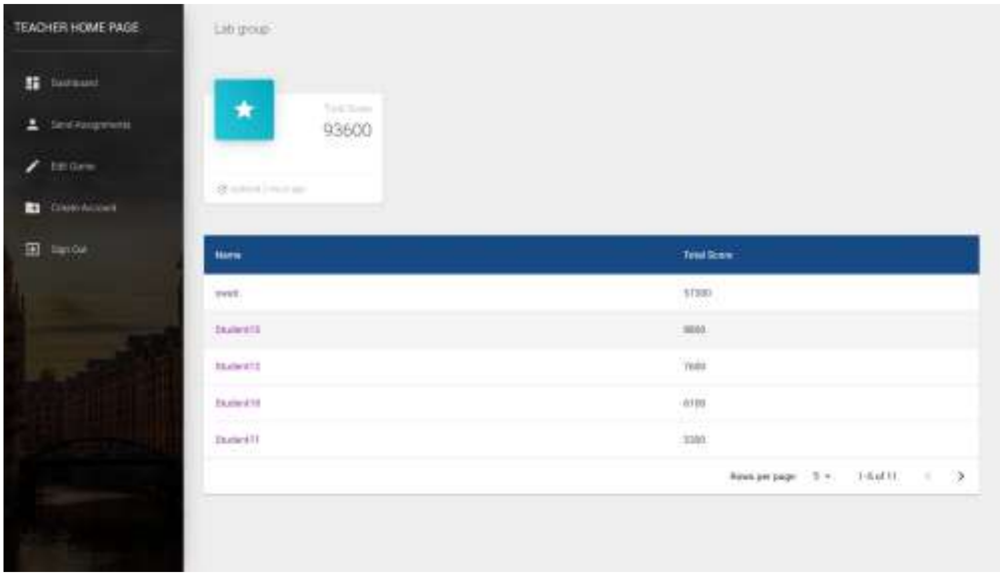
Galaxy	Planet	Question	Level
Planning and Defining	Decomposition Techniques	If a Direct approach to software project sizing is taken, size can be measured in	Easy
Planning and Defining	Decomposition Techniques	Which software project sizing approach develop estimates of the information domain characteristics?	Medium
Planning and Defining	Decomposition Techniques	Cost and effort estimation of a software uses only one forms of decomposition, either decomposition of the ________ or decomposition of the process.	Easy
Planning and Defining	Decomposition Techniques	In many cases, it is often more cost-effective to acquire, rather than develop, computer software	Medium

Rows per page: 5 ▾ 1-5 of [illegible] ‹ ›

Teachers WebApp Edit Game Page

3.1.10 Teachers WebApp: View Lab Group

The teacher will be able to view any or all lab groups. The teacher can view the statistics of the lab group.



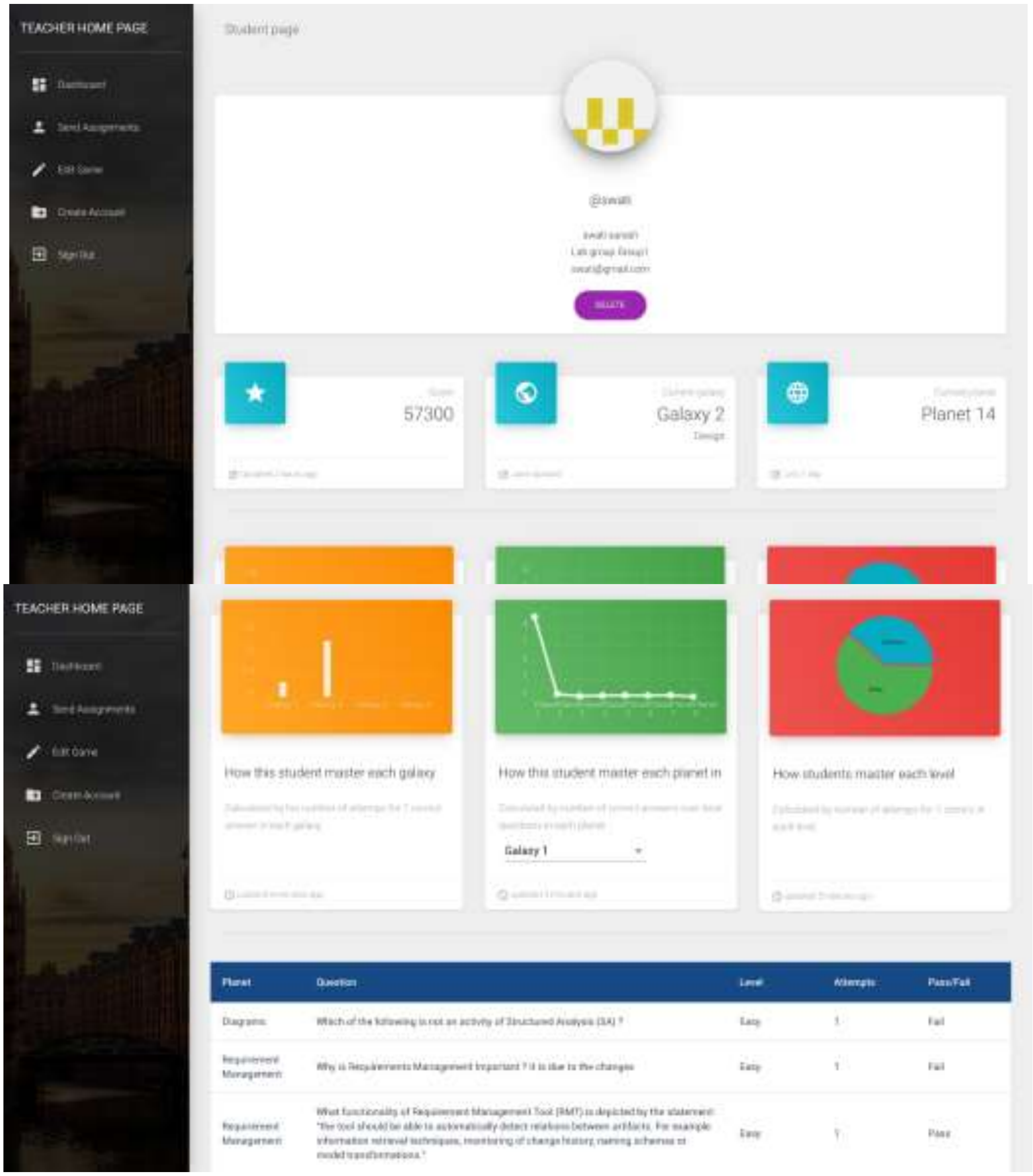
Name	Total Score
[illegible]	57300
Student13	9800
Student12	7600
Student10	6100
Student11	3300

Rows per page: 5 ▾ 1-5 of 11 ‹ ›

Teachers WebApp View Lab Group Page

3.1.11 Teachers WebApp: View Student Details

The teacher can view the student details and the performance of the particular student.

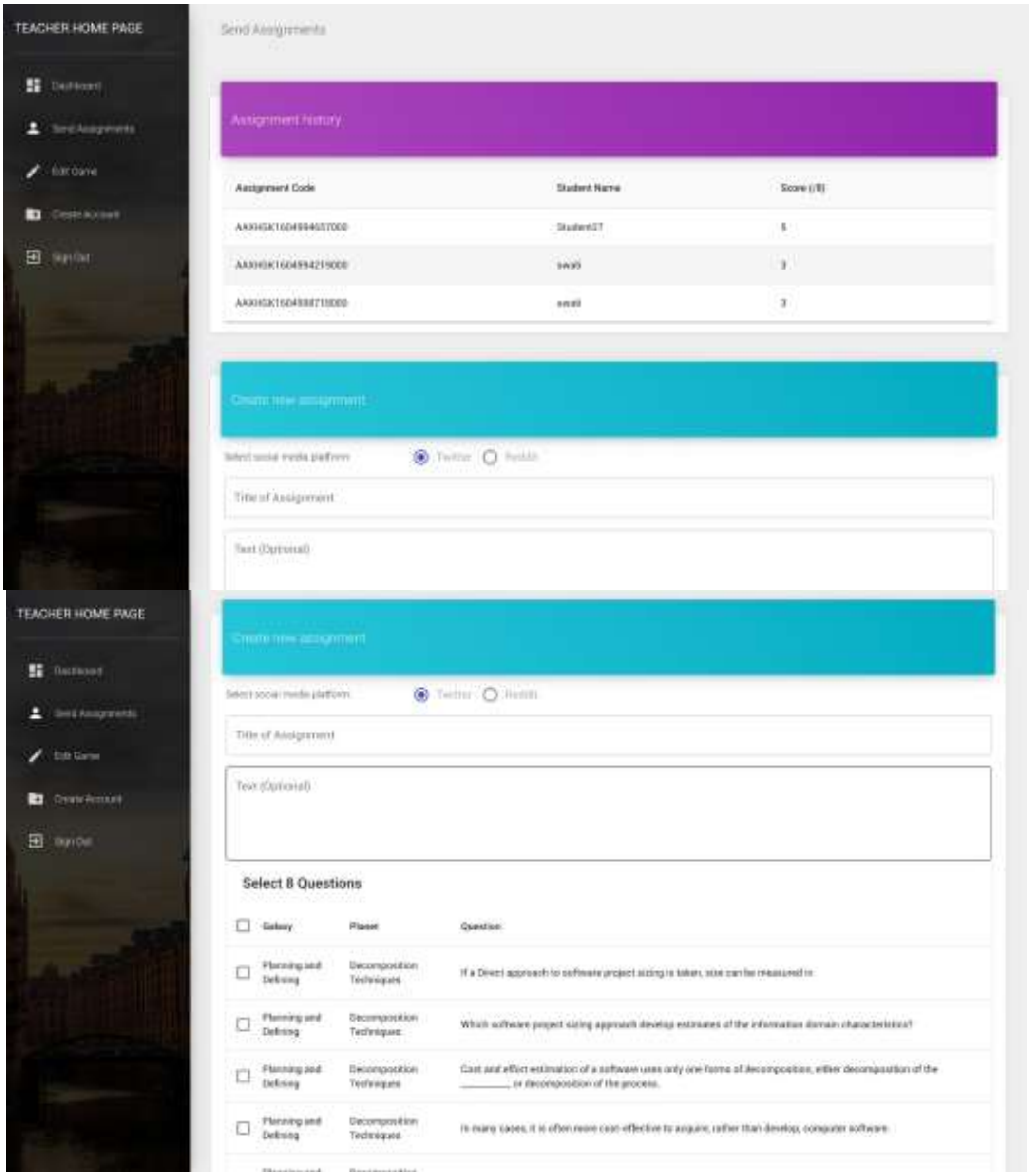


Planet	Question	Level	Attempts	Pass/Fail
Diagrams	Which of the following is not an activity of Structured Analysis (SA) ?	Easy	1	Fail
Requirement Management	Why is Requirements Management Important ? It is due to the changes	Easy	1	Fail
Requirement Management	What functionality of Requirement Management Tool (RMT) is depicted by the statement: "the tool should be able to automatically detect relations between artifacts. For example information retrieval techniques, monitoring of change history, naming schemas or model transformations."	Easy	1	Pass

Teachers WebApp View Student Details Page

3.1.12 Teachers WebApp: Create Assignments

Teacher will be able to pick 8 questions for the assignment. The teacher will be able to add descriptions for the assignments



Assignment Code	Student Name	Score (/8)
AAXHSK1604994657000	Student27	5
AAXHSK1604994219000	swa9	3
AAXHSK1604988718000	swati	3

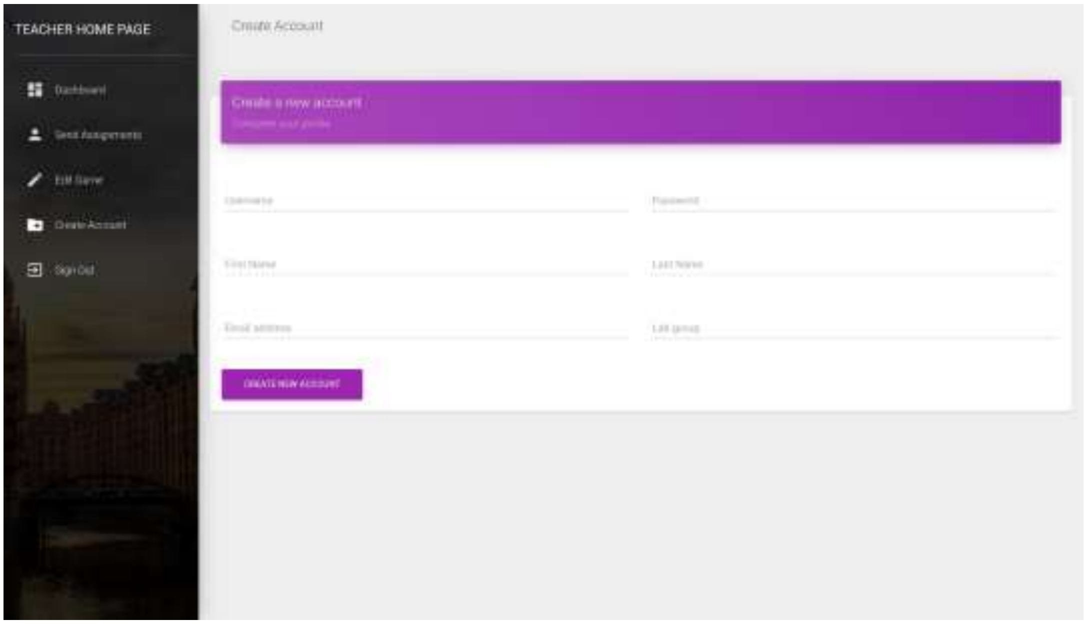
Select 8 Questions

□	Galaxy	Planet	Question
□	Planning and Defining	Decomposition Techniques	If a Direct approach to software project sizing is taken, size can be measured in:
□	Planning and Defining	Decomposition Techniques	Which software project sizing approach develop estimates of the information domain characteristics?
□	Planning and Defining	Decomposition Techniques	Cost and effort estimation of a software uses only one forms of decomposition, either decomposition of the ________ or decomposition of the process.
□	Planning and Defining	Decomposition Techniques	In many cases, it is often more cost-effective to acquire, rather than develop, computer software.

Teachers WebApp Create Assignments Page

3.1.13 Teachers WebApp: Create Student Account

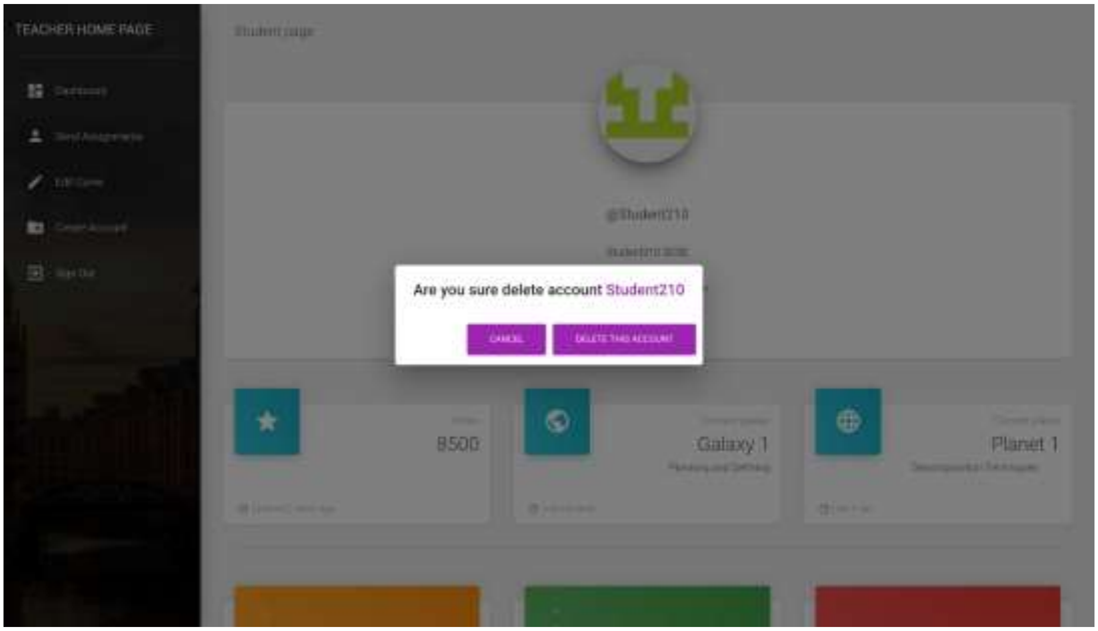
The teacher will be able to create a student account by inputting all the fields in the form.



Teachers WebApp Create Student Account Page

3.1.14 Teachers WebApp: Delete Student Account

The teacher will be able to delete the existing student account.


Are you sure delete account Student210

Teachers WebApp Delete Student Account Page

3.2 Hardware Interfaces

3.2.1 GalaxSE Game Application

The GalaxSE Game application requires a compatible laptop or desktop that supports the Windows platform with Unity. The device must have Internet connection capability.

3.2.2 Teachers WebApp Application

The Teachers WebApp is designed to run on any frequently used browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, etc. This includes the hardware which is able to run these browsers, some examples are Windows 8, Windows 10 and MacOS. The system must have internet connection capability. The application does not write information directly to the user's computer but instead uses a database which is located on a network server. The user's computer transfers and receives data from the server which uses basic networking protocols. All system's information is stored in the server's database which stores the data on the server's disk.

3.3 Software Interfaces

3.3.1 GalaxSE Game Application

- The PC must have a windows or Mac operating system installed. The operating system must meet these specific requirements: Windows 7 SP1+ and macOS 10.12+
- The platform will be collecting the results of the quizzes and uploading them to the database.

3.3.2 Teachers WebApp Application

- The web interface of the PlayToLearn application is required to run on all frequently used browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge and Internet Explorer.
- The platform will be handling the major features of the application including; account management, quiz management, crafting of questions, viewing statistics, etc.

3.4 Communications Interfaces

HTTPS Communication

- HTTP protocol encrypted by TLS which provides secure communication over the network and authentication of the website as well as the associated web server the client is communicating with thus helping to guard against the man-in-the-middle attack.
- In addition, it also provides bidirectional encryption of communications between the client and server and helps guard against any eavesdropping attempt or tampering of data

4. Functional Requirements

4.1. Account Management

4.1.1. The game has 2 types of accounts: Student and Teacher.

4.1.1.1. Student accounts are used for the GalaxSE Game

4.1.1.2. Teachers accounts are used for the Teachers WebApp

4.1.1.3. Account management is done on the Teachers WebApp by the Teachers

4.1.1.4. The GalaxSE Game application does not allow any account registration.

4.1.2. The Teachers must be able to create Student accounts.

4.1.2.1. The system must display a Create Account Form for the Teachers to input.

4.1.2.1.1. The form must consist of the account's username.

4.1.2.1.2. The form must consist of the account’s password.

4.1.2.1.3. The form must consist of the account’s first name.

4.1.2.1.4. The form must consist of the account’s last name.

4.1.2.1.5. The form must consist of the account’s lab group.

4.1.2.1.6. The form must consist of the account’s email.

4.1.2.2. The Teachers must fill in all the fields in the form.

4.1.2.3. The System must perform validation on the data entered in the form.

4.1.2.3.1. The email address must be valid.

4.1.3. The Teachers must be able to delete Student accounts.

4.1.4. The system must be able to store the account data into the database.

4.2 Login GalaxSE Game

4.2.1. Only Student accounts are able to login to GalaxSE Game

4.2.2. Students must enter their username and password and click on the “Submit” button.

4.2.2.1. The game must validate the username and password with the database.

4.2.2.2. The game rejects the user if username or password are invalid.

4.2.3. If login is successful, the game will validate redirect Student to the Home page

4.2.4. If login is unsuccessful, the game will deny access and indicate that the password and username are incorrect.

4.3 GalaxSE Gameplay

4.3.1 Students must be able to edit their profile

4.3.1.1. Student must be able to customize their Astronaut

4.3.1.1.1. Student must choose a color from a set of colors

4.3.1.1.2. Student are able to choose a hat from a set of hars

4.3.1.2. Student must be able to change their name

4.3.1.3. Student must be able to save changes

4.3.2. Homepage must have "Explore Solo Mission" , "Create Challenge Monster" and User Status Icon.

4.3.2.1. Explore Solo Mission:

4.3.2.1.1. The game must have a video tutorial in the bottom left corner of each Galaxy.

4.3.2.1.2. The game must have 4 Galaxies.

4.3.2.1.3 The game must 8 Planets within each Galaxy

4.3.2.1.4. The game must have 3 Monsters within each Planet

4.3.2.1.5. The game must have 4 questions in each Monster.

4.3.2.1.6. If the Students are to attempt the Monster again, 4 random questions will be generated.

4.3.2.1.7. All Monsters are unlocked for the Planet.

4.3.2.1.8. The Student must obtain a combined school of at least 75% for all Monsters in a Planet to unlock the next Planet.

4.3.2.1.9. The Student must unlock all Planets in a Galaxy to unlock the next Galaxy.

4.3.2.1.10. The game analyses the Student's previous performance to generate adaptive questions

4.3.2.1.11. The game must display all unlocked and locked objects (Galaxy, Planet, Monster)

4.3.2.1.12. The game must allow the first time players to start from the first Galaxy.

4.3.2.1.13. The game must keep all other Galaxies locked initially.

4.3.2.2. Create Challenge Monster:

4.3.2.2.1. Students must be able to choose Galaxy

4.3.2.2.2. Students must be able to choose the number of questions for each difficulty.

4.3.2.2.3. Total questions from easy medium hard must be equal to 8.

4.3.2.2.4. Student must be able to play the challenge and receive their score

4.3.2.2.5. Student must be able to share challenge via Twitter or Reddit

4.3.2.3. User Status:

4.3.2.3.1. Clicking on the user status icon must display the following: Leaderboard, Profile and Log out.

4.3.3. The game must record Student’s progress in database

4.3.3.1. Students must be able to continue gameplay from where they left off.

4.4. Leaderboard

4.4.1. The game must compute and display the overall leaderboard for “Explore Solo Mission”.

4.4.2. Student must be able to view the Leaderboard

4.4.3. The leaderboard must display the ranking, username and points of the Students.

4.4.4. The ranking must be from 1 to 10.

4.5 Complete Assignment

4.5.1. The game must allow Students to do Assignment in GalaxSE from the link given by the Teacher.

4.5.2. The game must display the correct Assignment to the Student

4.5.2.1. The game must display the correct Monster to the Student

4.5.2.2. The game must display the correct Questions to the Student

4.5.3. The game must record the completion and results of the Assignment.

4.5.4. Students must be able to view their results and the other Student's results.

4.5.5. The Assignment will send the results to the database.

4.5.6. Students must be able to share the results on Twitter or Reddit.

4.6 Complete Challenge

4.6.1. The game must allow Students to do the Challenge in GalaxSE from the link given by another Student.

4.6.2. The game must display the correct Challenge to the Student

4.6.2.1. The game must display the correct Monster to the Student

4.6.2.2. The game must display the correct Questions to the Student

4.6.3. The game must record the completion and results of the Challenge.

4.6.4. Students must be able to view their results and the other Student's results.

4.6.5. The game will send the results to the database.

4.6.6. Students must be able to share the results on Twitter or Reddit.

4.7 Teachers WebApp

4.7.1. Teachers can select to view any or all lab groups

4.7.1.1. Teachers can view a list of Students enrolled in each selected lab group

4.7.1.2. Teachers can view a leaderboard table in each selected lab group

4.7.1.2.1. The leaderboard table must include a Name column

4.7.1.2.2. The leaderboard table must include a Total Score column

4.7.1.3. Teachers can view the total score for the selected lab group

4.7.2. Teachers can select an individual Student to view his/her details

4.7.2.1. The System displays the information of that Student

4.7.2.1.1 System displays Student’s username

4.7.2.1.2 System displays Student’s full name

4.7.2.1.3 System displays Student’s lab group

4.7.2.1.4 System displays Student’s email

4.7.2.2. The System displays the progress of that Student

4.7.2.2.1 System displays Student’s Total score

4.7.2.2.2 System displays Student’s Current Galaxy

4.7.2.2.3 System displays Student’s Current Planet

4.7.2.2.4 System displays Student’s Analytics for Galaxy

4.7.2.2.4.1. Analytics show how many attempts for a correct question in each Galaxy

4.7.2.2.4.2. Analytics show the number of correct answer over total questions for each Planet in each Galaxy

4.7.2.2.4.3. Analytics show the number of attempts for 1 correct question in each difficulty level

4.7.2.3. The System displays all the questions attempted.

4.7.2.3.1 System displays Student’s Planet

4.7.2.3.2 System displays Student’s Question

4.7.2.3.3 System displays Student’s Level

4.7.2.3.4 System displays Student’s Attempts

4.7.2.3.5 System displays Student’s pass/fail

4.7.3. Teachers can edit the question bank for the GalaxSE Game

4.7.3.1. Teachers can add new questions into the question bank

4.7.3.1.1. Teachers can select the Galaxy using numbers 1 to 4

4.7.3.1.2. Teachers can select the Planet using numbers 1 to 32

4.7.3.1.3. Teachers can input the Quiz question

4.7.3.1.4. Teachers can input the difficulty of the question

4.7.3.1.5. Teachers can input the options

4.7.3.1.6. Teachers can select the correct option by clicking on the check box

4.7.3.1.7. Teachers can add the question into the question bank

4.7.3.2. Teachers can edit existing questions from the question bank

4.7.3.2.1. Teachers can edit the Galaxy using numbers 1 to 4

4.7.3.2.2. Teachers can edit the Planet using numbers 1 to 32

4.7.3.2.3. Teachers can edit the Quiz question

4.7.3.2.4. Teachers can edit the difficulty of the question

4.7.3.2.5. Teachers can edit the options

4.7.3.2.6. Teachers can edit the correct option by clicking on the check box

4.7.3.2.7. Teachers can update the question into the question bank

4.7.3.3. Teachers can delete questions from the question bank

4.7.3.3.4. Teachers can select the question to delete

4.7.3.3.5. System will delete the question from the database

4.7.4. Teachers can create assignment and send it to Students via Twitter or Reddit

4.7.4.1. Teachers must select 8 questions from the question bank

4.7.4.2. Teachers can input the assignment’s description

4.7.4.2.1. Teachers can input the Title of the assignment

4.7.4.2.2. Teachers can input the Description of the assignment

4.7.4.3. Teachers can choose which social media platform to share the assignment

4.7.4.3.1. Teachers can select Twitter

4.7.4.3.2. Teachers can select Reddit

4.7.4.4. The system will display a preview of the assignment

4.7.4.5. Teachers can share the assignment on the social media platform selected

4.7.5. Teachers can view Summary Report

4.7.5.1. Teachers can view Report for each Galaxy

4.7.5.1.1. Report shows Students mastery of each Planet

4.7.5.1.2. Report shows how Students answer questions

4.7.5.1.3. Report shows Students mastery of each difficulty level

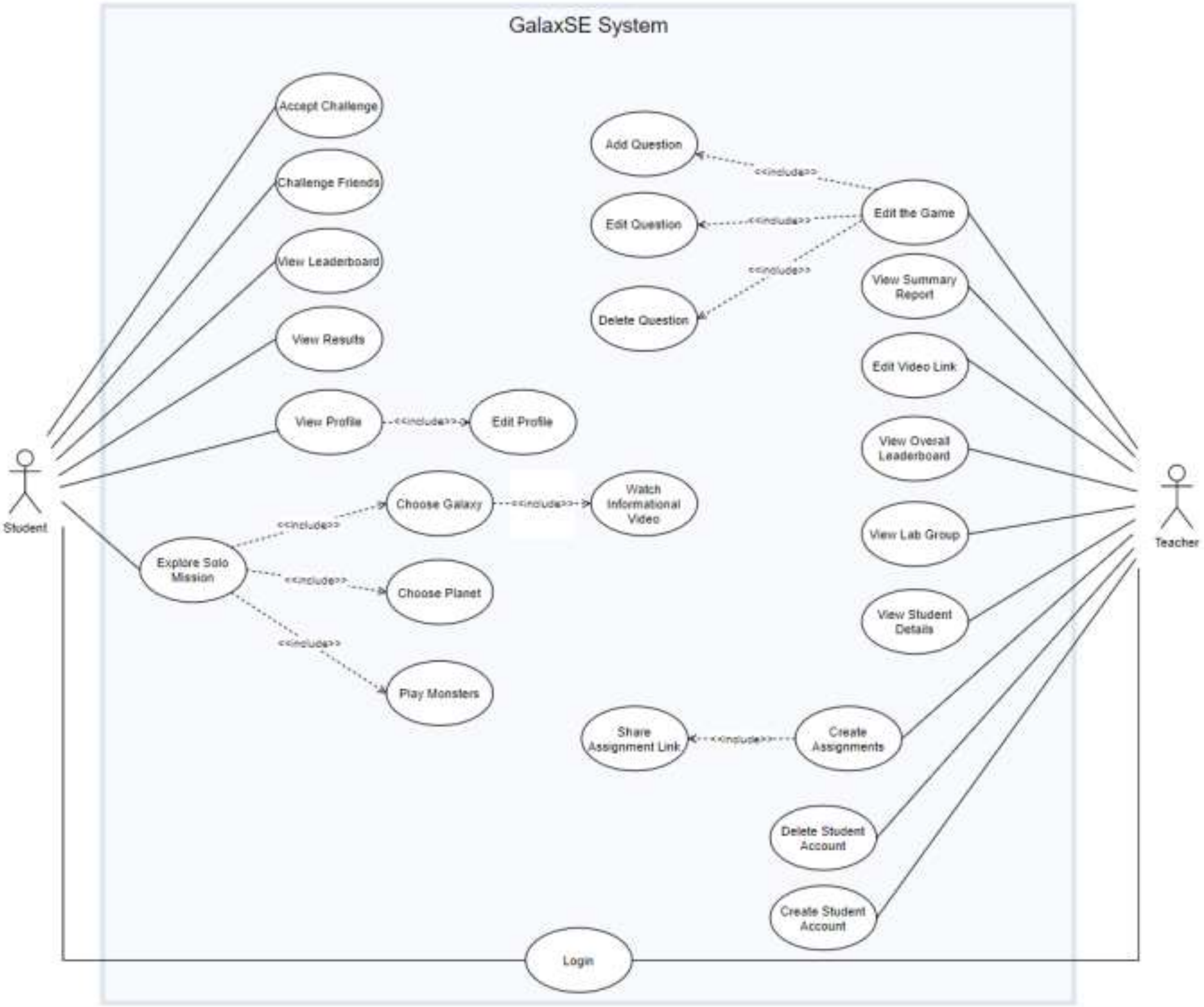
4.7.5.1.4. Report displays all questions in the Galaxy

4.7.6. Teachers can change the learning video for each Galaxy

4.7.6.1. Teachers can update learning video URL

4.7.6.2. System will update URL into database

4.1 Use Case Diagram



4.2 Use Case Descriptions

4.2.1 Use Cases (Student)

Use Case ID:	100		
Use Case Name:	Login		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initiating Actor)
Description	The user is able to login with their valid username and password at the login page to gain access to the application.
Preconditions	1. The system should be active 2. The user must have a valid account.
Postconditions	System must grant access to the user who has successful login or deny access to the user who has an unsuccessful login. Successful login brings the user to the homepage.
Priority	High
Frequency of Use	Frequently
Flow of Events	1. Student enters his/her username and password. 2. System retrieves the Student Data using the entered username. 3. System validates the username and password. 4. System checks if the account is validated. 5. System redirects Student to the home page.
Alternative Flows	If the user enters incorrect username or password 1. The application informs the user that “Username or Password Invalid”. 2. The application returns to step 1.
Exceptions	N/A

Includes	N/A
Special Requirements	The API should be reliable.
Assumptions	N/A
Notes and Issues	

Use Case ID:	101		
Use Case Name:	Explore Solo Mission		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initiating Actor)
Description	The user plays the game. He/she is shown a view of all the worlds (Galaxies) and is able to select one.
Preconditions	1. The user must be connected to the Internet. 2. The user must be logged into his/her account. 3. The user is in the home page
Postconditions	The system displays a map of the locked and unlocked Galaxies to the user.
Priority	High
Frequency of Use	High
Flow of Events	1. The user chooses Explore Solo Mission. 2. The application displays all unlocked and locked Galaxies to the user.
Alternative Flows	N/A
Exceptions	N/A

Includes	Choose Galaxy, Choose Planet, Play Monsters, View Results
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	102		
Use Case Name:	Choose Galaxy		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initiating Actor)
Description	The user selects an unlocked Galaxy for gameplay.
Preconditions	1. The user must be connected to the Internet 2. The user must be logged in 3. The selected Galaxy must be unlocked
Postconditions	The system opens the selected Galaxy and a map of all the Planets (levels) is displayed.
Priority	High
Frequency of Use	High
Flow of Events	1. The user chooses a Galaxy. 2. The system checks whether the Galaxy is unlocked 3. If unlocked, the system opens the selected Galaxy 4. The system displays the map of Planets
Alternative Flows	If the user selects a locked Galaxy

	1. The system informs the user that they have tried to access a locked Galaxy. 2. The system does not allow access into the locked Galaxy
Exceptions	N/A
Includes	Watch Informational Video
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	The first Galaxy of the game is automatically unlocked.

Use Case ID:	103		
Use Case Name:	Watch Informational Video		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initiating Actor)
Description	The user watches an informational video on the Galaxy's topic to prepare for the upcoming questions.
Preconditions	1. The user is connected to the Internet. 2. The user is logged in. 3. An instructional video has been uploaded or linked for the Galaxy by the Teacher.
Postconditions	The system plays the informational video in a pop-up.
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. The user clicks on the video. 2. The system redirects the user to the website to play the

	instructional video
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	1. The video link is not broken.
Notes and Issues	N/A

Exceptions	EX-1: If the user fails the same level five times 1. The system instructs the user to review the relevant concepts and contact their instructor if they have any doubts. 2. The system locks the level until the user plays the informational video. EX-2: If the user decides to stop attempting the Monster
Includes	N/A
Special Requirements	Questions are chosen randomly from the Question Bank. Difficulty increases with each sequential question.
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	104		
Use Case Name:	Choose Planet		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initiating Actor)
Description	The user selects a Planet (topic) to play.
Preconditions	1. The user is connected to the Internet. 2. The user is logged in. 3. The user clicks on an unlocked Planet.
Postconditions	The user enters the Planet to choose Monsters to fight
Priority	High
Frequency of Use	High
Flow of Events	1. The user clicks on a Planet. 2. The user enters the Planet if the Planet is unlocked.
Alternative Flows	If the Planet is not unlocked 1. The system informs the user that the Planet they are trying to access is locked. 2. The system instructs the user to play a different Planet.
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	The first Planet of every unlocked Galaxy is automatically unlocked.

Use Case ID:	105		
Use Case Name:	Play Monsters		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initiating Actor), Question Bank

Description	The user enters the gameplay, where he/she must battle and defeat an alien by answering a set of multiple choice questions based on the Galaxy's topic.
Preconditions	1. The user is connected to the Internet 2. The user is logged in. 3. The level chosen is unlocked.
Postconditions	1. The system generates a report for the user with answers and explanations. 2. The next Planet is unlocked
Priority	High
Frequency of Use	High
Flow of Events	1. User choose 1 of the 3 Monsters (Easy, Medium, Hard) available 2. The system randomly selects questions based on the required difficulty from the Question Bank. 3. The system displays the Monster and the questions to be answered. 4. The user must answer correctly all the questions to defeat the Monster. 5. Repeat till all Monsters are attempted. 6. If the score is equal to 75% and above and the next Planet will be unlocked
Alternative Flows	If the user is unable to pass the minimum 75% 1. The user must re-attempt the same Monsters until he/she successfully achieving 75% or more
Exceptions	If the user decides to stop attempting the Monsters
Includes	N/A
Special Requirements	Questions are chosen randomly from the Question Bank. Difficulty increases with each sequential question.
Assumptions	All Monsters are unlocked
Notes and Issues	N/A

Use Case ID:	106

Use Case Name:	View Results		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initiating Actor)
Description	The user views a system-generated report of his/her results. The report includes the Student's score as well as the correct answers and explanations.
Preconditions	1. The user is logged in. 2. The user is connected to the Internet. 3. The user has answered all the questions for a Monster
Postconditions	1. The user's position in the leaderboard is updated.
Priority	High
Frequency of Use	High
Flow of Events	1. The system generates the Student's score and displays a report for the level. 2. The leaderboard is updated. 3. The user is taken to the next Monster.
Alternative Flows	If the current Planet is the final Planet of the Galaxy 1. The next Galaxy is unlocked and the user is taken to the next Galaxy, where he/she can watch the informational video and proceed.
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	107		
Use Case Name:	View Leaderboard		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initiating Actor)
Description	The user views the system-generated leaderboard based on the performances of all Students across different levels and topics
Preconditions	1. The user is logged in 2. The user is connected to the Internet 3. The user is the homepage
Postconditions	The system displays the overall leaderboard which can be filtered by the user
Priority	High
Frequency of Use	High
Flow of Events	1. The user clicks on User status icon 2. The user clicks on leaderboard 3. The overall leaderboard is shown
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	Even if a Student has not yet participated in gameplay, the leaderboard must be displayed.

Use Case ID:	108		
Use Case Name:	Challenge Friends		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initiating Actor)
Description	The Student can invite other Students to a face-off for a particular topic/level
Preconditions	1. The user is connected to the Internet 2. The user is logged in
Postconditions	The user can challenge other friends through Twitter and Reddit by sending them a Challenge Monster
Priority	High
Frequency of Use	High
Flow of Events	1. The user will click on "Create Challenge Monster" 2. The user will be able to choose the Galaxy they want to challenge 3. The user will be able to choose the number of questions for each difficulty. 4. The total number of questions chosen must be equal to 8. 5. The user will proceed to attempt the challenge and receive their score 6. The user can share their challenge and score through Twitter or Reddit.
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A

Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	109		
Use Case Name:	Accepting the Challenges		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initiating Actor)
Description	The user chooses between viewing the pending requests and pending games
Preconditions	1. The user is connected to the Internet 2. The user has the game in his/her device 3. The user is has login into the game 4. The user is on the social media platform used to share the challenge
Postconditions	The user will complete the challenge and display the score
Priority	High
Frequency of Use	Medium
Flow of Events	1. The user will click on the link (The link is in the post the friend has created on Twitter or Reddit) 2. The user will be redirected to the Game 3. The challenge will start immediately 4. The user will complete the challenge and will be able to post their scores on Twitter of Reddit
Alternative Flows	User has not login to game 1. The user will be prompted to log in 2. After successful login, the challenge will start 3. The user will complete the challenge and will be able to post their

	scores on Twitter of Reddit
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	110		
Use Case Name:	View Profile		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initiating Actor)
Description	The user can view his/her name and Astronaut
Preconditions	1. The user must be connected to the internet 2. The user must be logged in 3. The user is in the homepage
Postconditions	The user information must be displayed.
Priority	Low
Frequency of Use	Low
Flow of Events	1. The user clicks on the User Status icon 2. The user clicks on profile 3. Student name and Astronaut is displayed
Alternative Flows	N/A
Exceptions	N/A

Includes	Edit Profile
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	111		
Use Case Name:	Edit Profile		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Student (Initializing Actor)
Description	The user can edit his/her profile
Preconditions	1. The user must be logged into his/her account 2. The user must be connected to the internet 3. The user is in the view profile page
Postconditions	The user's profile must be updated.
Priority	Low
Frequency of Use	Low
Flow of Events	1. The user selects what he/she wants to modify (username, color of Astronaut, accessory Astronaut is wearing) 2. If a user chooses to change their username, they must enter an alphanumeric string between 0-15 characters. 3. If the user chooses to customise their Astronaut, they may choose from the different colours and accessories available. 4. The user must save the changes. 5. The changes are stored in the user database system

Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

4.2.2 Uses Cases (Teacher)

Use Case ID:	200		
Use Case Name:	Login		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	The user must log into an existing account before proceeding to the content of the application. The user will log in through their Google account or email.
Preconditions	1. The system is active 2. The user must have a Google account or Email 3. The user must be connected to the Internet.
Postconditions	The user has been granted access to the application and is able to view the content.
Priority	High
Frequency of Use	High
Flow of Events	1. System prompts the user to login with an email or Google account. 2. The user is prompted to enter their email and password. 3. User credentials are validated through Auth0 API
Alternative Flows	If the user enters invalid information 1. The application informs the user that he/she has “Wrong email or password”. 2. The application returns to step 1.
Exceptions	N/A
Includes	N/A
Special Requirements	The API must be reliable.
Assumptions	N/A

Notes and Issues	N/A

Use Case ID:	201		
Use Case Name:	View Lab group		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	10/10/2020

Actor	Teacher (Initiating Actor)
Description	The user chooses one out of the available groups of Students.
Preconditions	1. The user is connected to the internet 2. The user is logged in as a Teacher 3. The Student groups have been created
Postconditions	User views group page with access to group data and content/questions.
Priority	Medium
Frequency of Use	High
Flow of Events	1. User views list of Lab groups 2. User selects one of the Lab groups 3. User can view all Students in the Lab group 4. User can view leaderboard table in the Lab group 5. User can view the total score of the Lab group.
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	There are Students in the Lab Groups which have already been created.
Notes and Issues	N/A

Use Case ID:	202		
Use Case Name:	View Student details		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	The user chooses one out of the available groups of Students.
Preconditions	1. The user is connected to the internet 2. The user is logged in as a Teacher 3. The Student groups have been created
Postconditions	System displays Student details.
Priority	Medium
Frequency of Use	High
Flow of Events	1. User clicks on the selected Students 2. Details of the Students are being shown (Username, Full name, Lab group, Email, Total score, Current Galaxy and Planets, Analytics for Galaxy and all attempted questions)
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	203		
Use Case Name:	View Overall Leaderboard		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	Users view the current ranking of Students on the leaderboard.
Preconditions	1. The user needs to be connected to the internet 2. The user has successfully logged in 3. Students are registered in the database 4. Students are collecting points by playing in solo mission mode/challenging their friends
Postconditions	Student names, rankings and points are displayed.
Priority	High
Frequency of Use	High
Flow of Events	1. System displays ranked list of Students 2. Student are ranked according to their total score
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	1. Students are participating actively in game play to accumulate

	points and appear on the leaderboard.
Notes and Issues	At the very beginning of the application, a small number of users who have participated in gameplay may result in a sparsely populated ranking list.

Use Case ID:	204		
Use Case Name:	Edit Game		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	User can add new questions, remove existing questions and edit questions in the question bank
Preconditions	User is logged in as a Teacher
Postconditions	Changes made by the user are saved in the Question Bank
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. User clicks Edit the Game 2. System displays the edit question bank page
Alternative Flows	N/A
Exceptions	N/A
Includes	Add questions, Edit questions and Delete questions
Special Requirements	N/A
Assumptions	1. User has knowledge about topics taught in the application 2. Questions added/edited by a user are relevant to the topic at hand

	3. User has adequate reasoning behind addition and deletion of questions
Notes and Issues	N/A

Use Case ID:	205		
Use Case Name:	Add Question		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	User can add new questions
Preconditions	User is logged in as a Teacher
Postconditions	Changes made by the user are saved in the Question Bank
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. User chooses to add a new question 2. User adds the options for the questions. 3. User will add the correct answers for the questions 4. User save their changes.
Alternative Flows	1. User chooses to add a new question 2. User adds the options for the questions. 3. User will add the correct answers for the questions 4. User choose to cancel changes
Exceptions	N/A

Includes	N/A
Special Requirements	N/A
Assumptions	1. User has knowledge about topics taught in the application 2. Questions added by a user are relevant to the topic at hand 3. User has adequate reasoning behind addition and deletion of questions
Notes and Issues	N/A

Use Case ID:	206		
Use Case Name:	Edit Question		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	User can edit questions in the question bank
Preconditions	User is logged in as a Teacher User has chosen a question they wish to edit
Postconditions	Changes made by the user are saved in the Question Bank
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. User select a question to edit 2. User edits the question content (title, options, correct answers, level, etc.) 3. User save the changes

Alternative Flows	1. User select a question to edit 2. User edits the question content (title, options, correct answers, level, etc.) 3. User cancel the changes
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	1. User has knowledge about topics taught in the application 2. Questions edited by a user are relevant to the topic at hand
Notes and Issues	N/A

Use Case ID:	207		
Use Case Name:	Delete Question		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	User can edit questions in the question bank
Preconditions	User is logged in as a Teacher User has chosen a question they wish to edit
Postconditions	Changes made by the user are saved in the Question Bank
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. User select a question to delete 2. System delete question from database
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	1. User has knowledge about topics taught in the application 2. User has adequate reasoning behind addition and deletion of questions
Notes and Issues	N/A

Use Case ID:	208		
Use Case Name:	Edit Video Link		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	Users can choose to upload a new video or to update the existing tutorial video.
Preconditions	1. User is logged in as a Teacher. 2. User is connected to the Internet. 3. User has chosen a Galaxy whose tutorial video they wish to edit.
Postconditions	Changes made by the user are saved and the database is updated
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. User chooses whether to add a new video (by uploading a link or file) or to replace an existing video. 2. User saves their changes
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	1. User has knowledge about topics taught in the application 2. Video uploaded by a user is relevant to the topic at hand
Notes and Issues	N/A

Use Case ID:	209		
Use Case Name:	View Summary Report		
Created By:	Teo W ee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	Teacher (Initiating Actor)
Description	User can view Students’ progress.
Preconditions	1. User is logged in as a Teacher. 2. User is connected to the Internet. 3. User has chosen a Galaxy
Postconditions	Analysis of Galaxy performance data is shown to the user.
Priority	High
Frequency of Use	Medium
Flow of Events	1. The user clicks on an individual Galaxy. 2. System displays analytics of that chosen Galaxy
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	1. Students in selected group have been participating actively in gameplay 2. Students have attempted questions from the Galaxy.
Notes and Issues	N/A

Use Case ID:	210		
Use Case Name:	Create Assignments		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initiating Actor), Question Bank
Description	The User can create new assignments
Preconditions	1. The user needs to be connected to the internet 2. The user needs to be logged in
Postconditions	A new level is created in the relevant Galaxy.
Priority	High
Frequency of Use	Medium
Flow of Events	1. The user creates a new level for the assignment. 2. The questions are added to the Question Bank along with a due date. 3. The system generates a link to the assignment.
Alternative Flows	N/A
Exceptions	N/A
Includes	Share Assignment Link
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

Use Case ID:	211		
Use Case Name:	Share Assignment Link		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initializing Actor), Twitter API, Reddit API
Description	The user shares the Assignment link through social media.
Preconditions	1. The user is connected to the internet 2. The user is logged in as Teacher
Postconditions	The Assignment link is shared through one or more social media.
Priority	High
Frequency of Use	Medium
Flow of Events	1. The Assignment link that is generated by the system is shared using social media
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

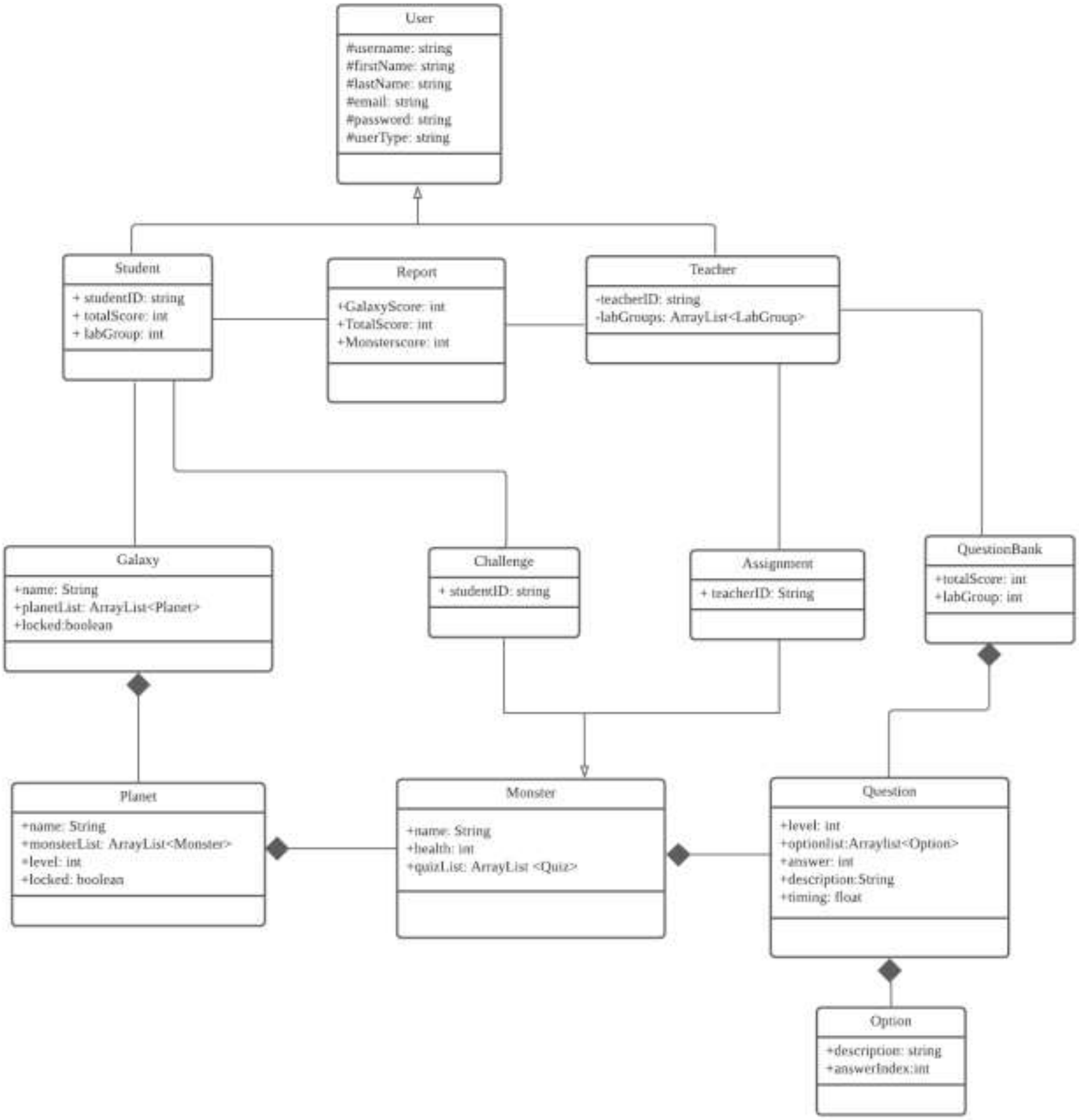
Use Case ID:	212		
Use Case Name:	Create Student Account		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initializing Actor), Database
Description	The user creates an account.
Preconditions	1. The user is connected to the internet 2. The user is logged in as Teachers
Postconditions	Student account is being created
Priority	High
Frequency of Use	Medium
Flow of Events	1. User click on create account 2. User fill in the form 3. Once details are validated, account will be created 4. Details will be stored in the database
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

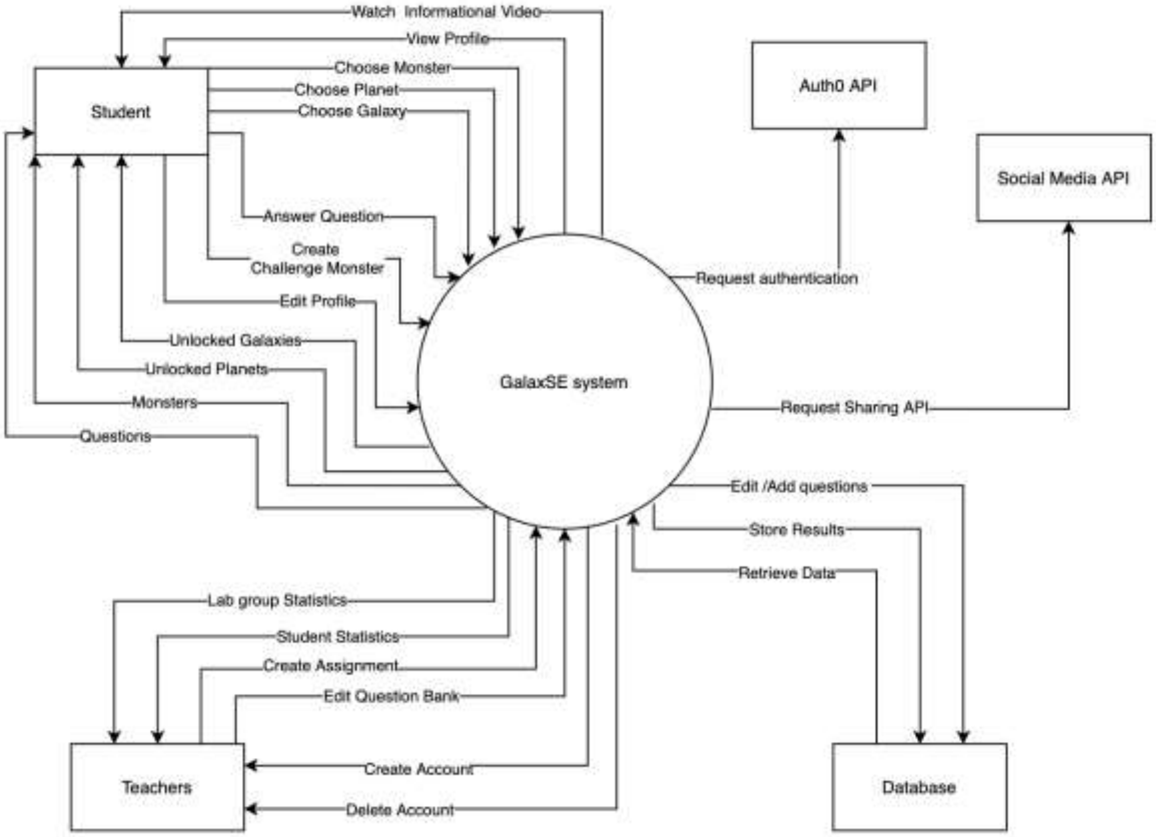
Use Case ID:	213		
Use Case Name:	Delete Student Account		
Created By:	Teo Wee Ren	**Last Updated By:**	Tan Yi Heng
Date Created:	15/09/2020	**Date Last Updated:**	20/11/2020

Actor	User (Initializing Actor), Database
Description	The user deactivates a Student account.
Preconditions	1. The user is connected to the internet 2. The user is logged in as Teachers
Postconditions	Student account is deleted and database is updated
Priority	High
Frequency of Use	Medium
Flow of Events	1. User click on account to be deleted 2. User clicks on delete 3. User confirms account to be delete 4. Account deleted 5. Database will be updated
Alternative Flows	N/A
Exceptions	N/A
Includes	N/A
Special Requirements	N/A
Assumptions	N/A
Notes and Issues	N/A

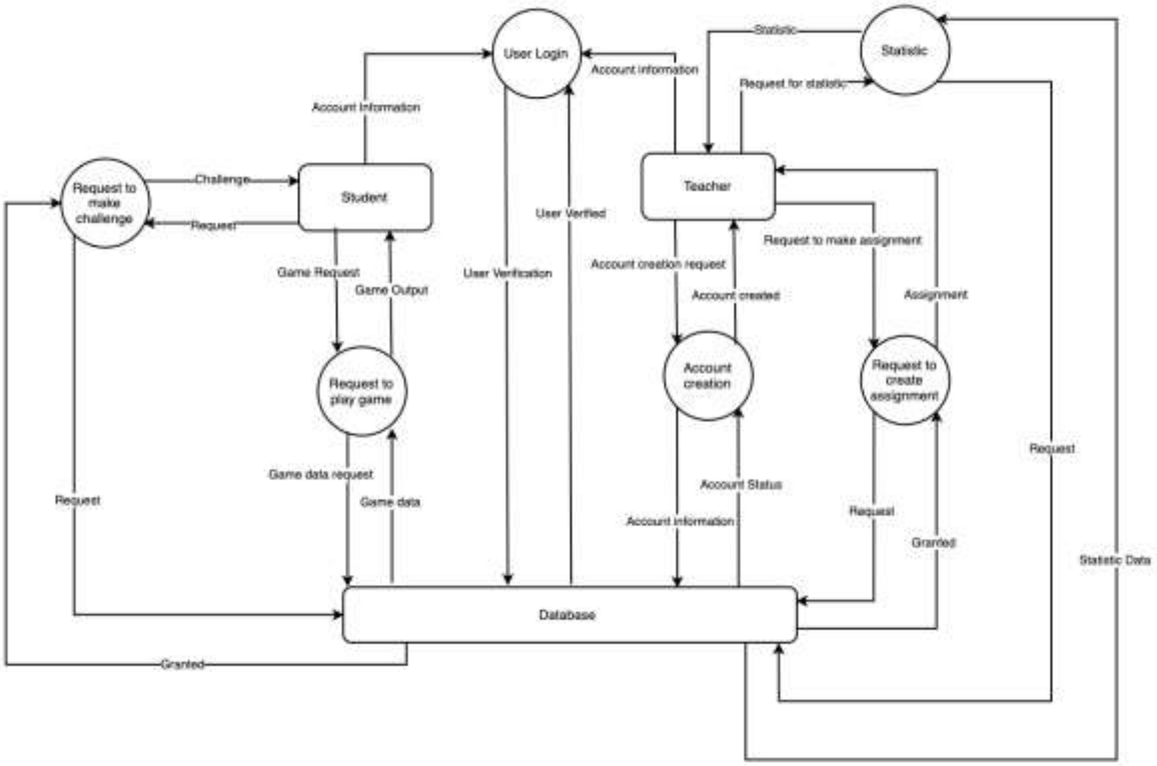
4.3 Class Diagram



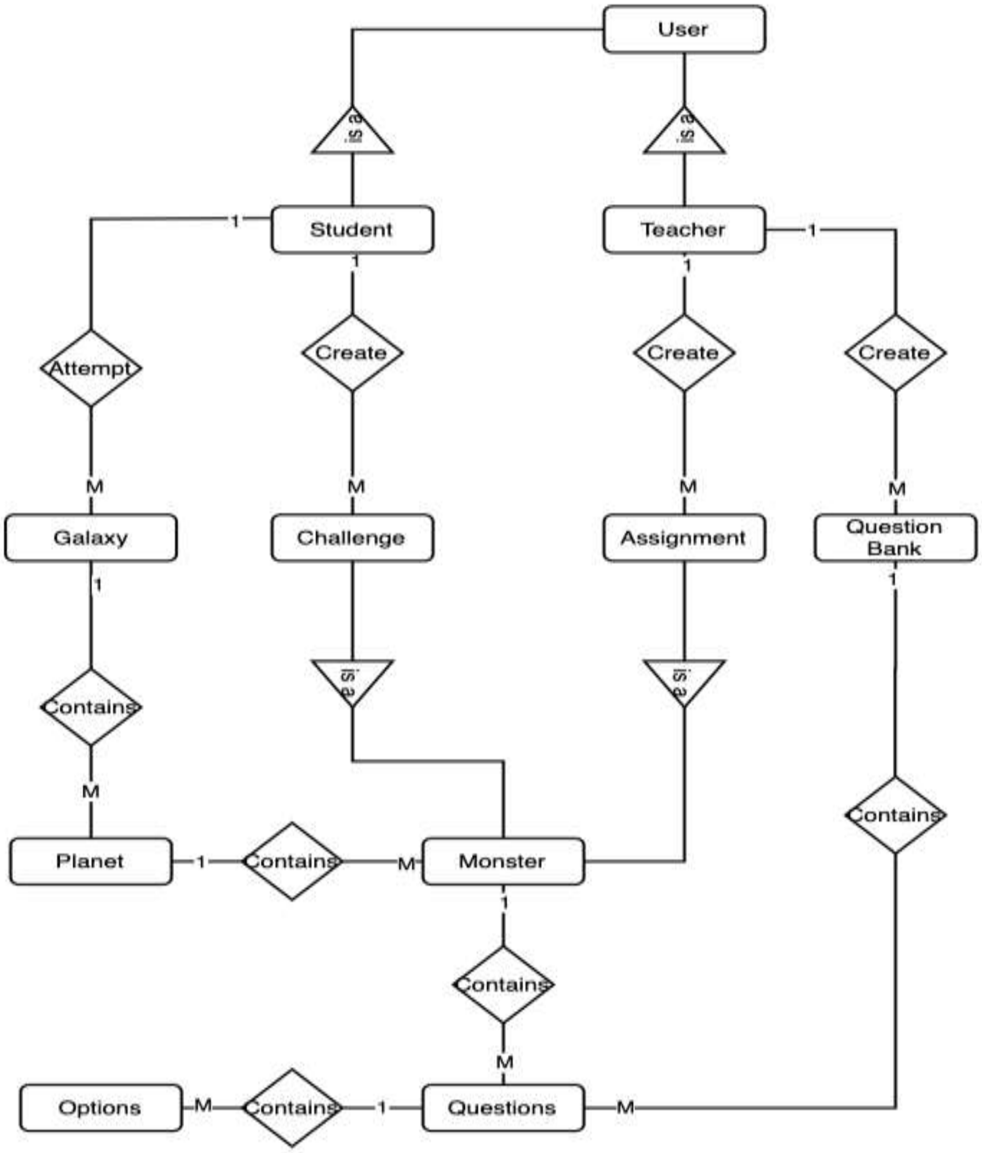
4.4 Context Diagram



4.5 Data Flow Diagram



4.6 Entity-relationship Diagram



4.7 Decision Tables

4.7.1 Proceed to a new Galaxy

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

4.7.2 Proceed to a new Planet

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

4.7.3 Student Send a Challenge

Condition	**1**	**2**	**3**	**4**	**5**	**6**
User authorized	F	T	T	T	T	T
The chosen Galaxy is unlocked	-	F	T	T	T	T
Total number of questions selected is equal to 8	-	-	F	T	T	T
Student completes the challenge and receives the score	-	-	-	F	T	T
Student has login Reddit or Twitter	-	-	-	-	F	T
Action						

Accept Request						X
Reject Request	X	X	X	X	X	

4.7.4 Teacher Send an Assignment

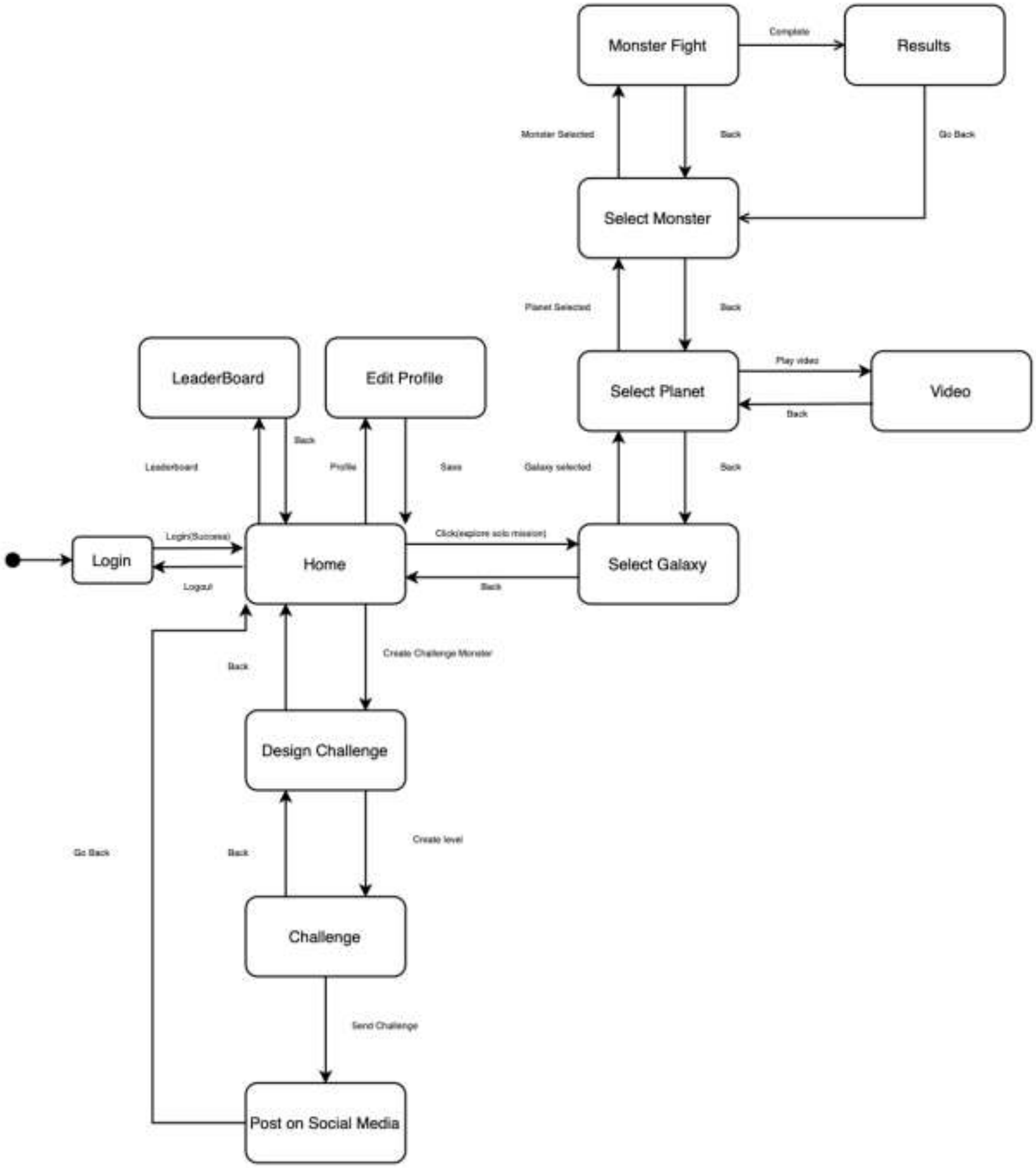
Condition	**1**	**2**	**3**	**4**
User authorized	F	T	T	T
The selected questions are available	-	F	T	T
Teacher has logged into Reddit or Twitter	-	-	F	T
Action				
Accept Request				X
Reject Request	X	X	X	

4.7.5 Teachers Deactivates Account

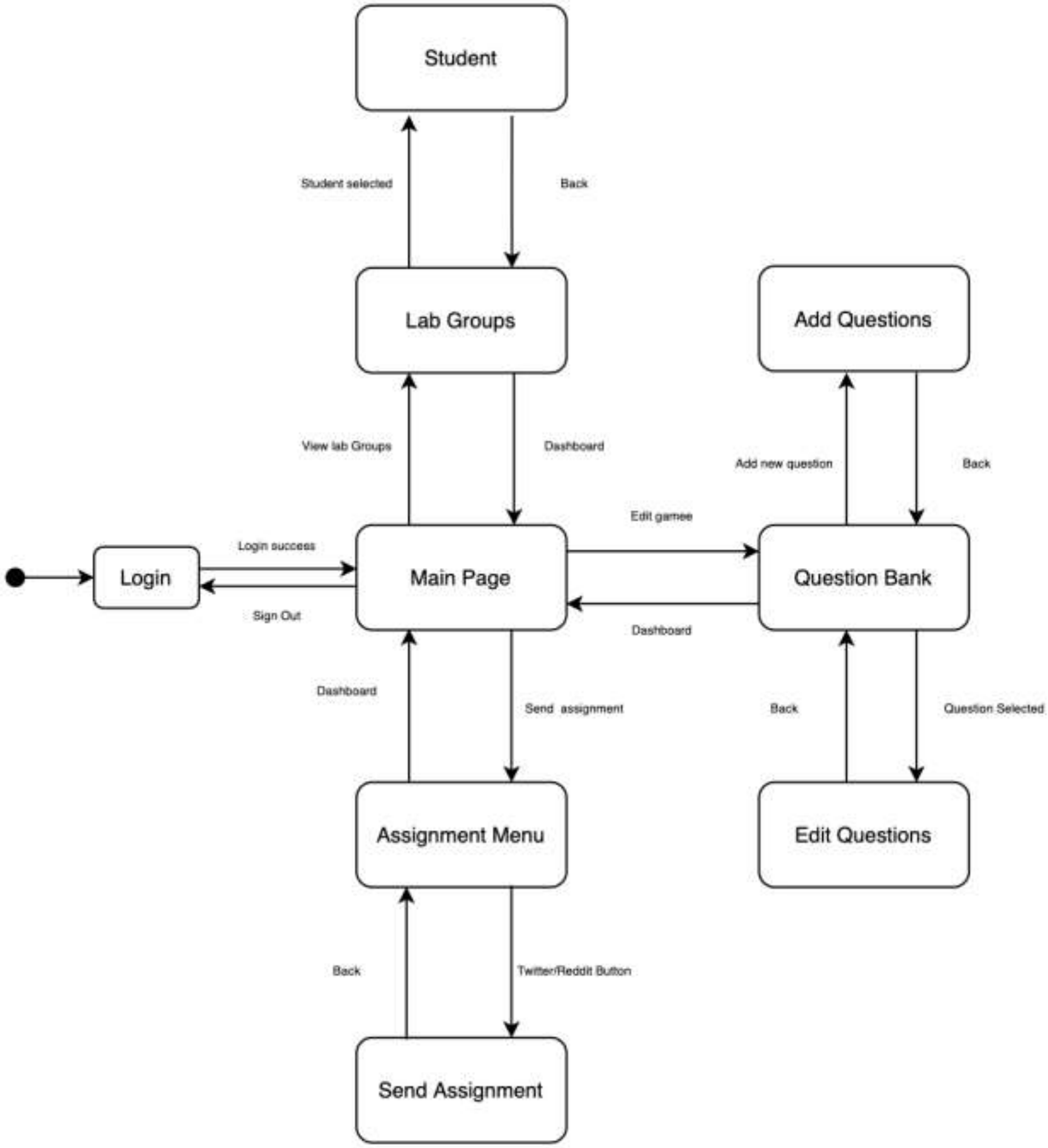
Condition	**1**	**2**	**3**
User authorized	F	T	T
The selected account exists	-	F	T
Action			
Accept Request			X
Reject Request	X	X	

4.8 Dialog Map

4.8.1 Student GalaxSE Game

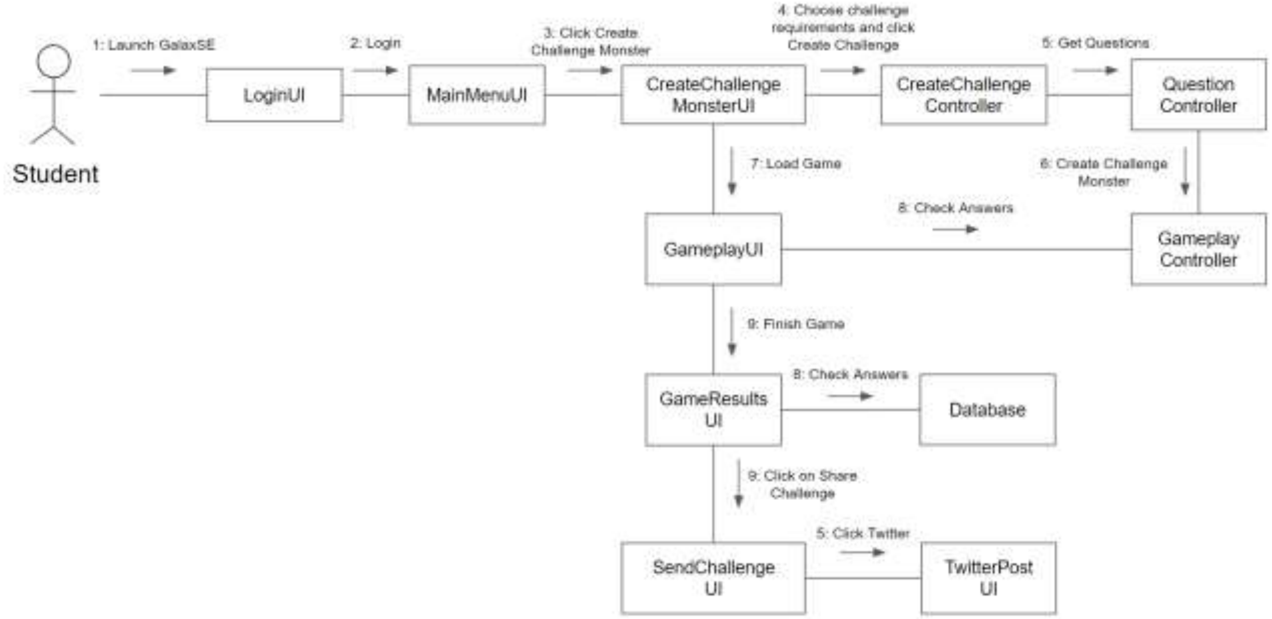


4.8.2 Teachers WebApp

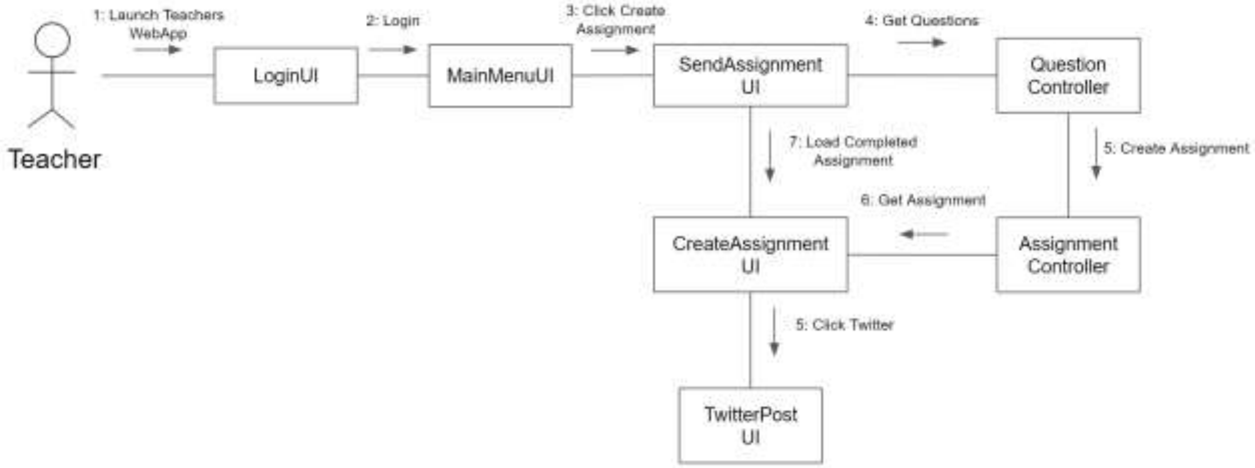


4.9 Communication Diagram

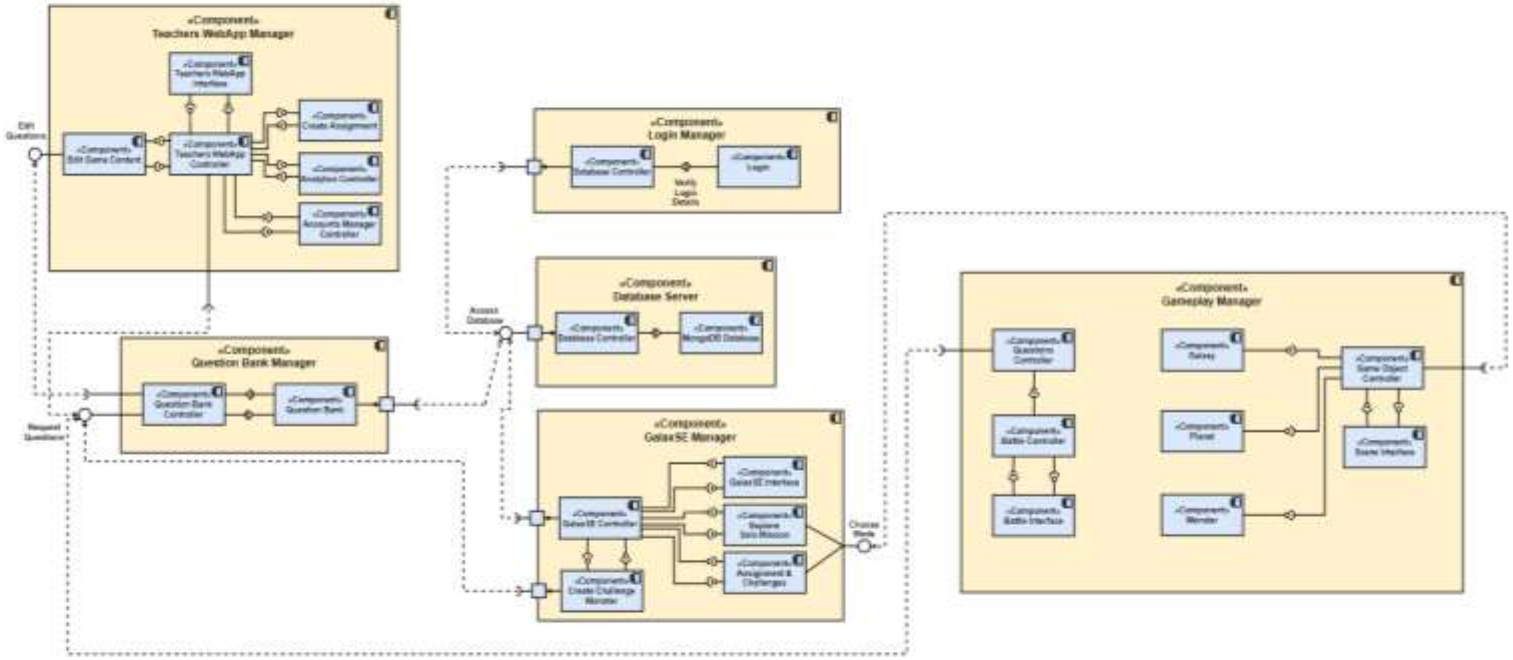
4.9.1 Student Create and Sends Challenge



4.9.2 Teacher Create and Send Assignment



4.10 Component Diagram

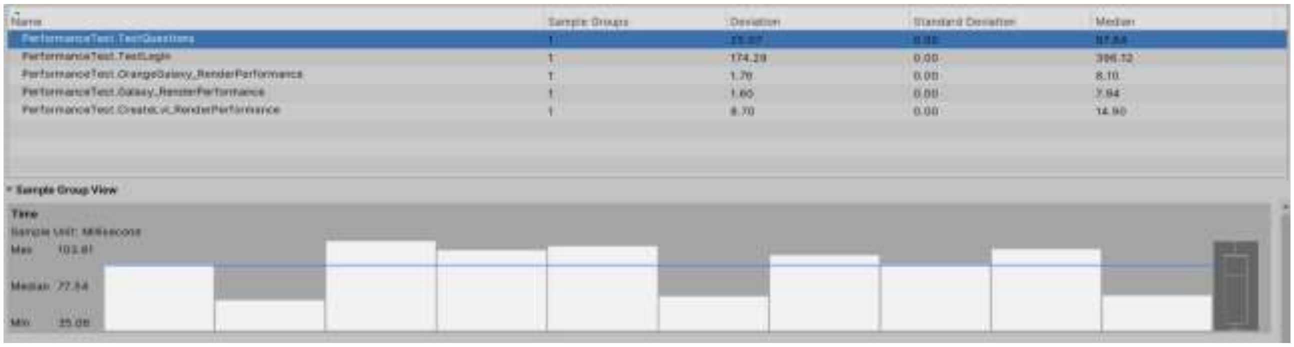


5. Non-functional Requirements

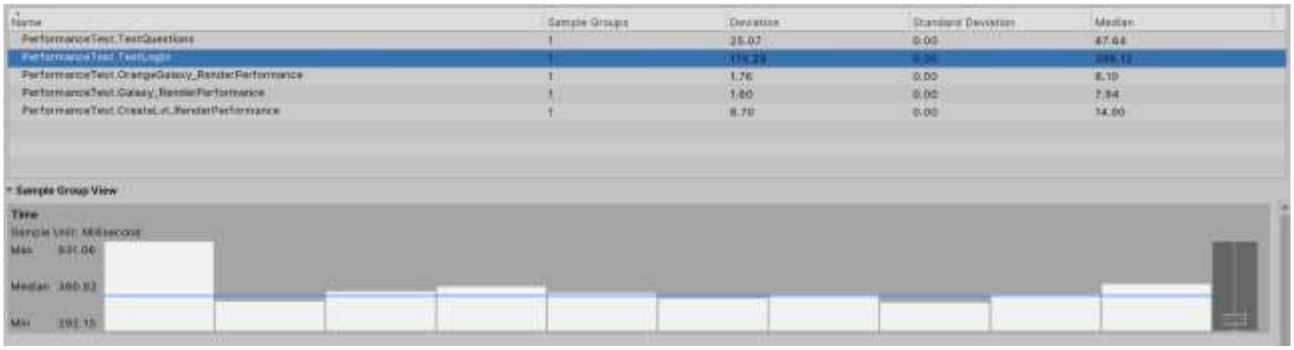
5.1 Performance Requirements

1. The GalaxSE Game should obtain the questions from database within 0.5 seconds
2. Logging into GalaxSE Game should take less than 1 seconds
3. Rendering of scenes should take less than 0.1 seconds

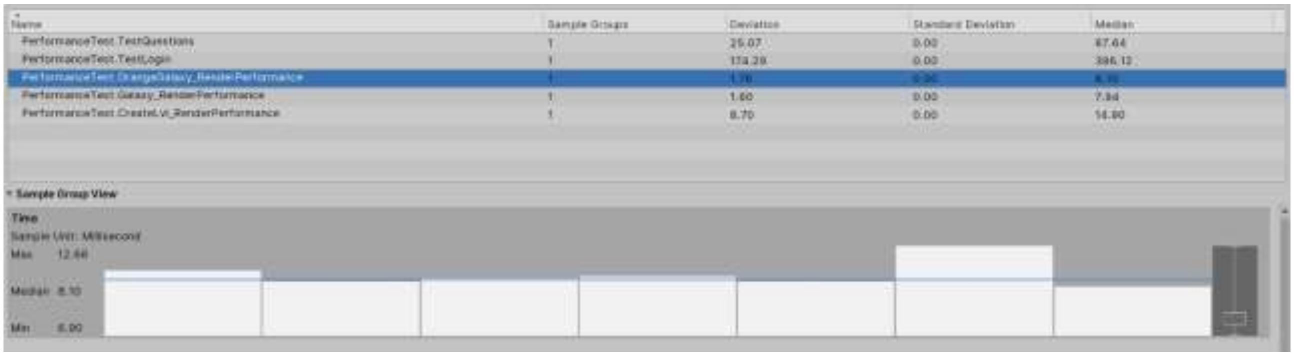
To ensure that our application meets the performance requirements, we will be using the Unity performance testing extension to test the performance of the system. We did 2 load testings, TestQuestions and Testlogin. Testquestions will simulate how fast the system will work when 150 users are concurrently playing the game and require the game to obtain the questions from the database to display in the game. Testlogin will simulate 50 users trying to login concurrently and we will measure how fast the users will take to login. The other 3 tests will test how fast the game can render the scenes.



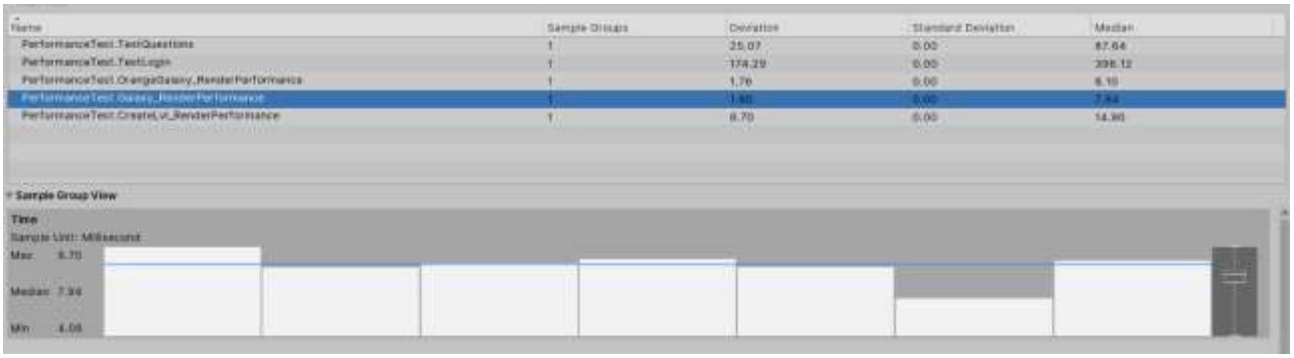
Name	Sample Groups	Deviation	Standard Deviation	Median
PerformanceTest.TestQuestions	1	25.07	0.00	87.64
PerformanceTest.TestLogin	1	174.29	0.00	386.12
PerformanceTest.OrangeGalaxy_RenderPerformance	1	1.76	0.00	8.10
PerformanceTest.Galaxy_RenderPerformance	1	1.60	0.00	7.84
PerformanceTest.CreateLvl_RenderPerformance	1	8.70	0.00	14.90



Name	Sample Groups	Deviation	Standard Deviation	Median
PerformanceTest.TestQuestions	1	25.07	0.00	87.64
PerformanceTest.TestLogin	1	174.29	0.00	386.12
PerformanceTest.OrangeGalaxy_RenderPerformance	1	1.76	0.00	8.10
PerformanceTest.Galaxy_RenderPerformance	1	1.60	0.00	7.84
PerformanceTest.CreateLvl_RenderPerformance	1	8.70	0.00	14.00



Name	Sample Groups	Deviation	Standard Deviation	Median
PerformanceTest.TestQuestions	1	25.07	0.00	87.64
PerformanceTest.TestLogin	1	174.29	0.00	386.12
PerformanceTest.OrangeGalaxy_RenderPerformance	1	1.76	0.00	8.10
PerformanceTest.Galaxy_RenderPerformance	1	1.60	0.00	7.84
PerformanceTest.CreateLvl_RenderPerformance	1	8.70	0.00	14.90



Name	Sample Groups	Deviation	Standard Deviation	Median
PerformanceTest.TestQuestions	1	25.07	0.00	87.64
PerformanceTest.TestLogin	1	174.29	0.00	398.12
PerformanceTest.OrangeGalaxy_RenderPerformance	1	1.76	0.00	8.10
PerformanceTest.Galaxy_RenderPerformance	1	1.60	0.00	7.94
PerformanceTest.CreateLvl_RenderPerformance	1	8.70	0.00	14.90

Sample Group View

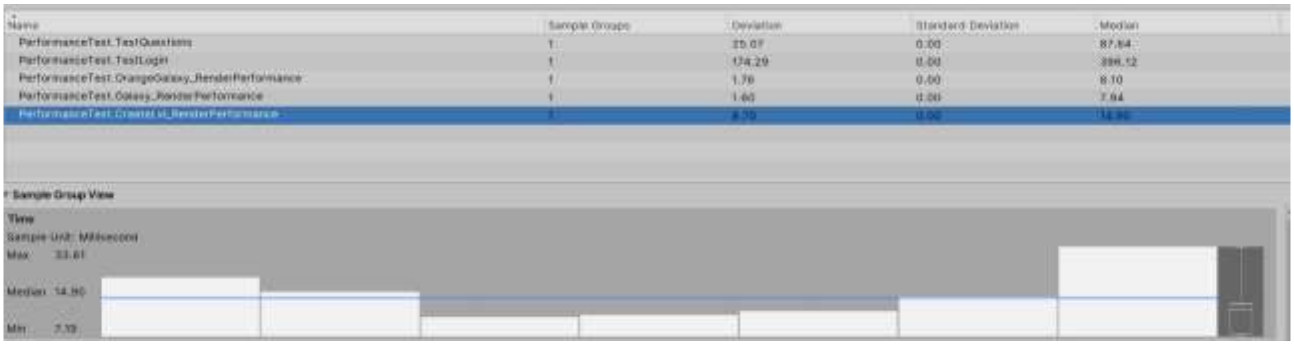
Time

Sample Unit: Millisecond

Max 8.70

Median 7.94

Min 4.08



Name	Sample Groups	Deviation	Standard Deviation	Median
PerformanceTest.TestQuestions	1	25.07	0.00	87.64
PerformanceTest.TestLogin	1	174.29	0.00	398.12
PerformanceTest.OrangeGalaxy_RenderPerformance	1	1.76	0.00	8.10
PerformanceTest.Galaxy_RenderPerformance	1	1.60	0.00	7.94
PerformanceTest.CreateLvl_RenderPerformance	1	8.70	0.00	14.90

Sample Group View

Time

Sample Unit: Millisecond

Max 33.61

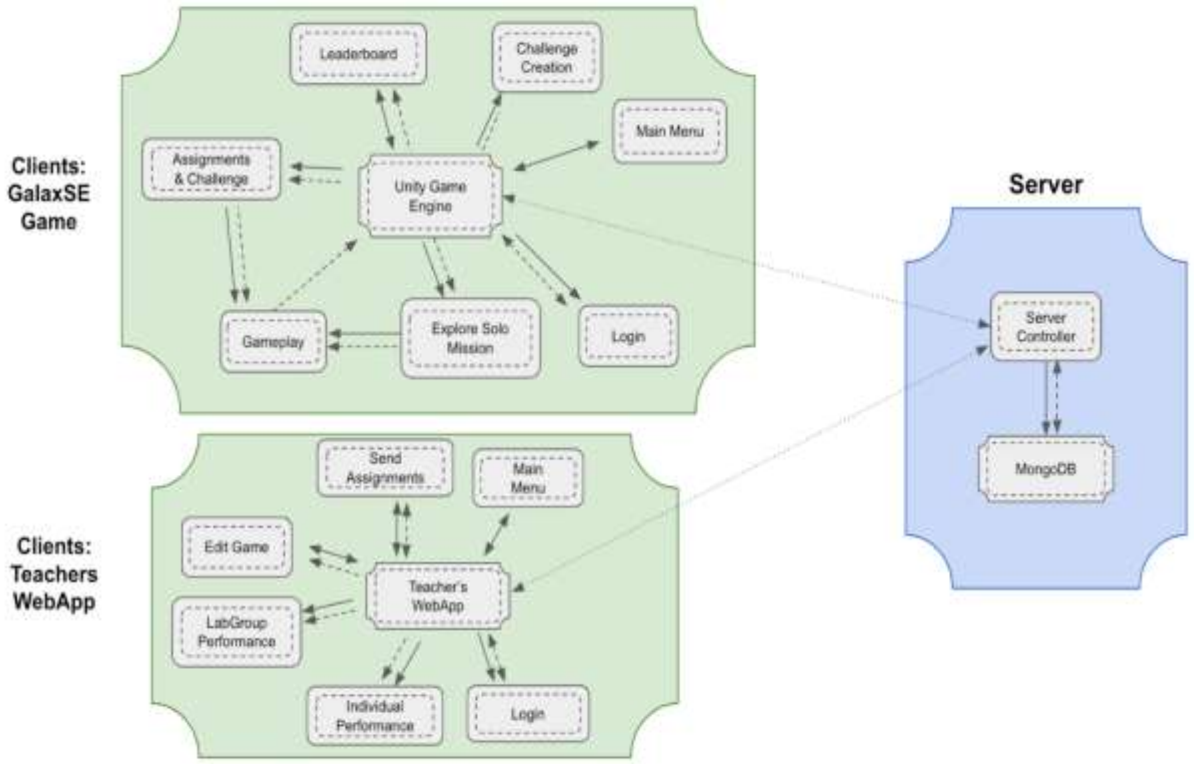
Median 14.90

Min 7.10

6. System Architecture

6.1 Primary Candidate Architecture:

6.1.1 Client & Server Architecture



Our main architecture is the client-server architecture. There will be 2 clients, the GalaxSE Game and Teachers WebApp, and they will interact with our server to access the database.

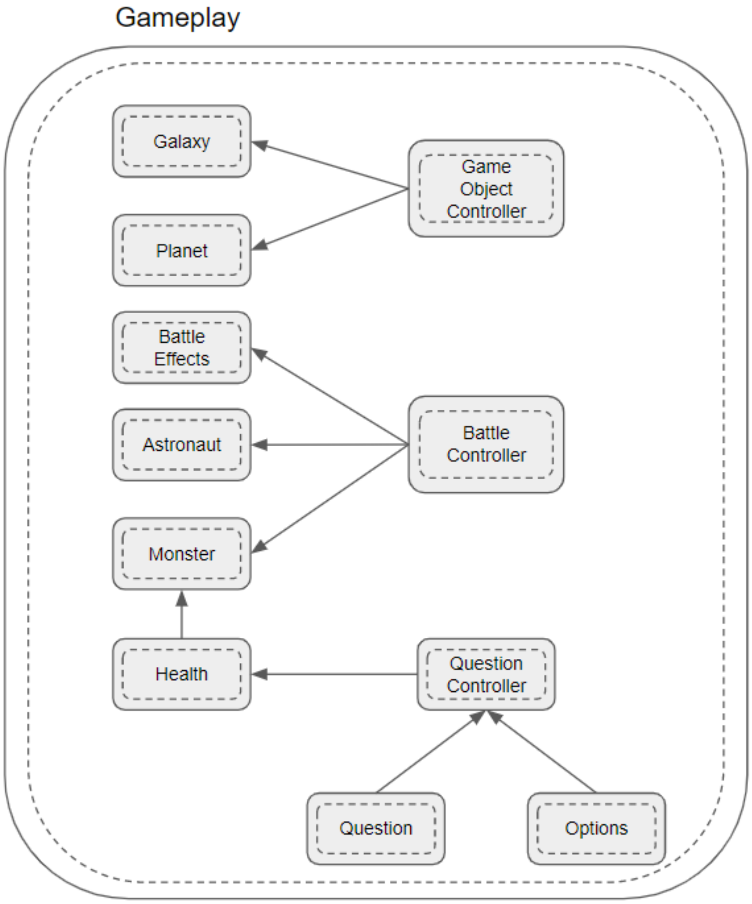
For the GalaxSE game, the Unity Game Engine receives and sends information from the server and will control all the components in the GalaxSE game. This includes the main gameplay "Explore Solo Mission", Assignments and Challenges, and Challenge Creation. For the main gameplay, the Assignment & Challenges and the Explore Solo Mission components will interact with the Gameplay component.

One example is when the Student decides to complete an assignment/challenge. When the Student clicked on the link provided, the link will bring them to the Gameplay and the user will play the game. Once the Student completes the game his results will be sent to the database through the unity game engine and Server.

We chose this architecture as our full system consists of 2 applications using 1 database. This architecture style ensures modularity and clearly shows the connection between our applications and the database. Furthermore, it allows our team to easily split the development process between the clients and server thus cutting short the time needed for development. We are able to split the development team into 3: Game developer, Teacher Developer and the Server Developer. We develop each component independently and combine them once they are ready. It is also easier to maintain and fix any errors on the system.

Within our main architecture, we implemented the entity component architecture style for our Gameplay subsystem. Entity component architecture is commonly used in games as it allows us to write reusable scripts and attach them as components to multiple entities.

6.1.2 Gameplay Subsystem: Entity Component Architecture

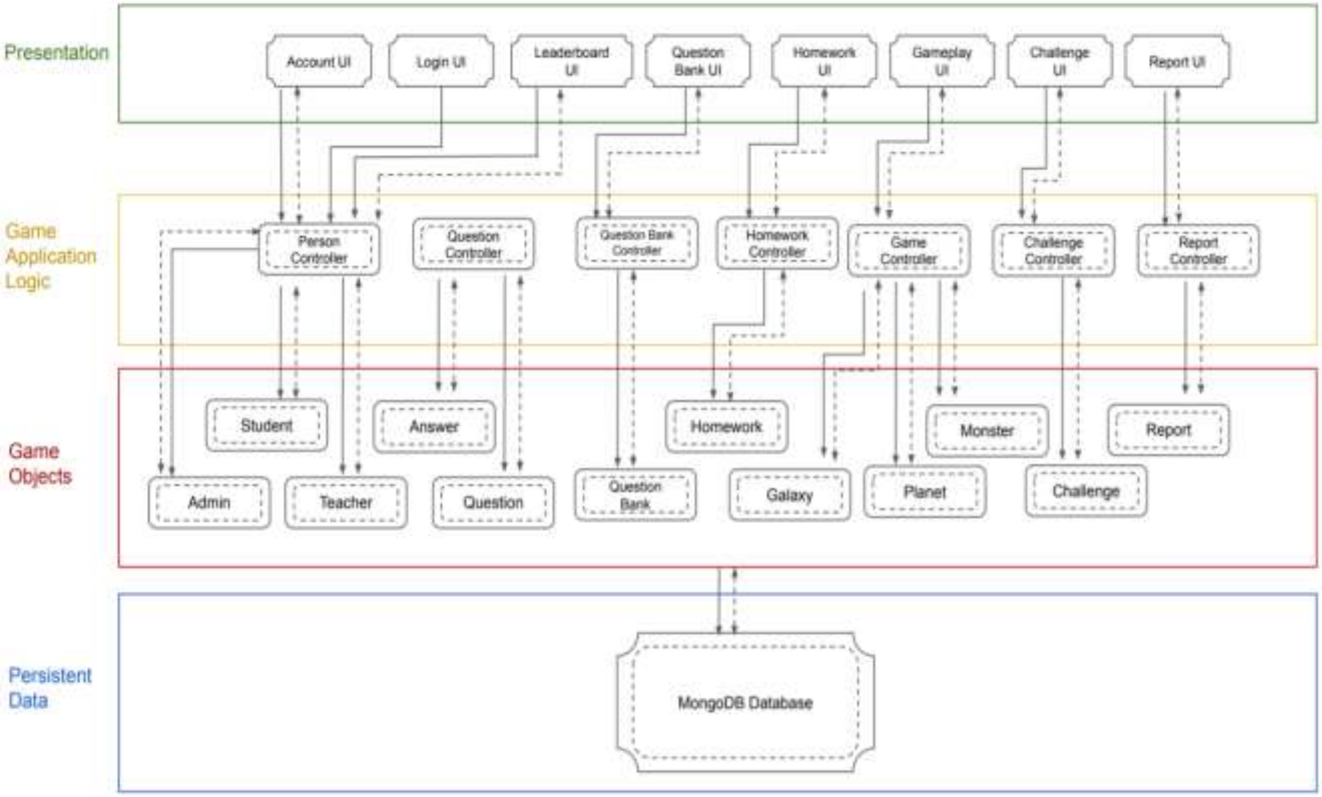


For the actual gameplay implementation of our project, we decided to use an entity-component architecture, which is commonly used in games. This allowed us to write reusable scripts and attach them as components to multiple entities. The key quality attribute our team is focusing on is maintainability, hence we require an architecture style that allows us to make modifications easily. This architecture style allows us to build a loosely coupled system so that we can make modifications to a particular component without affecting the entire game. One example is that we are able to add additional battle effects through the Battle Effects Component. We can also add new Monsters or questions easily.

This architecture style enhances our system's functionality through the introduction of new components and controllers allowing us to add additional features to the gameplay easily.

6.2 Alternative Architecture

6.2.1 4-Layered Systems Architecture



An alternative architecture we can use will be the 4 Layered Systems architecture. This architecture is useful as it increases maintainability by separating the user interfaces from the Game Application Logic, and the Game Application Logic from the Game Objects, and the Game Objects from the database. This allows different components within each layer to be easily replaced or reused.

As such, the separated layer system enables us to develop loosely coupled systems and allows the team to work on different parts of the application parallelly with minimal dependencies on each other. It also makes unit testing easier by allowing us to test the components independently of each other.

6.3 Rationale for Candidate Architecture Selection

Our team has decided that the primary architecture for our system will be a Hierarchically Heterogeneous architecture style that uses Client & Server as our primary candidate architecture and Entity Component as our subsystem architecture. We believe this combination of architecture styles allows us to split the development process easily among our team while achieving Maintainability.

Other architecture styles are not feasible for our system because of their limitations and lack of alignment with our key quality attribute. One example will be the Pipe and Filter architecture. Although Pipe and filter style is easy to understand and support reusability, it is unable to be implemented in our design as it is not good in handling interactive applications. Since we are developing a game, having an interactive gaming platform is essential.

The alternative architecture, 4 Layered Systems architecture is not as feasible because we need components to pass control to one another instead of having control strictly passed from the top layer to the bottom layer.

7. Testing

To ensure that our software fulfils the functional requirements defined in the System Requirement Specifications and to assure to the customer that the software system will be stable for consumer usage upon releasing it to the market, Software Testing will be conducted as part of our Software Development Process. All testing on the GalaxSE Game was done using Unity's testing framework and all scripts can be run through the Test Runner window.

Several levels of tests will be conducted, as follows:

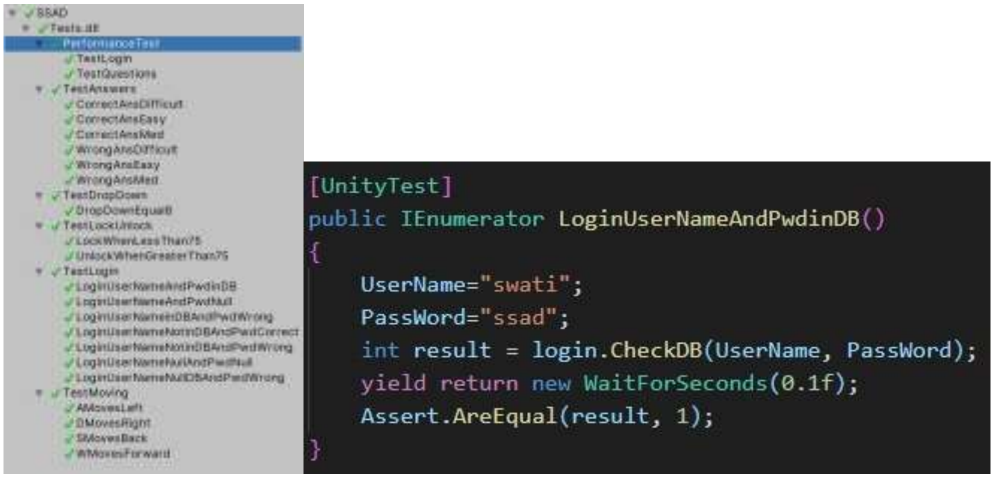
7.1 Unit Testing

Unit testing, also known as component testing, is meant to ensure that each component or function will operate accurately as it is intended to. Individual classes will be tested to ensure reliability and functionality within a unit-level. White box testing is conducted at this phase. Since the team has developed the codes for the software, we have full knowledge of the fields necessary fields for each system. By studying the implemented code, we will determine all legal (valid and invalid) and illegal inputs and verify the outputs against the expected outcomes.

For the GalaxSE game, we have used the Unity testing framework to implement unit testing. We have written scripts within Unity to test the inner workings of the application such as login, drop down, answering of questions etc. This type of testing has been carried out to ensure that each unit of our software is working as expected.

The scripts are divided such that each script is a test suite which tests a particular functionality that sits at the core of our application. Within each test suite script, there are multiple unit tests which cover different scenarios which that functionality has to correctly navigate in order to pass all tests. If any single unit test within a test suite fails, the entire test suite fails.

Shown below is the list of unit test cases carried out as well as an example of a unit test for login functionality which tests whether the application allows a user to log in when they have entered valid credentials which are stored in the database..



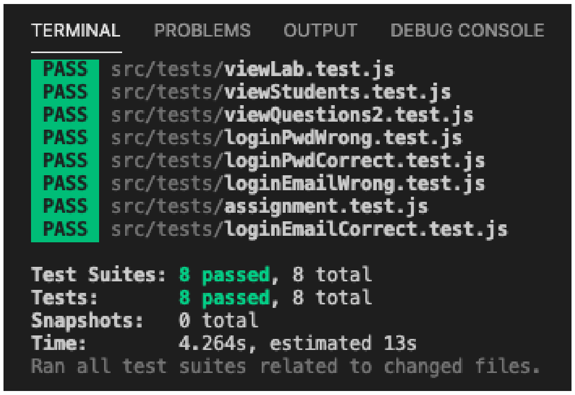
```
▼ ✓ SSAD
  ▼ ✓ Tests.dll
    ▼ PerformanceTest
        ✓ TestLogin
        ✓ TestQuestions
    ▼ ✓ TestAnswers
        ✓ CorrectAnsDifficult
        ✓ CorrectAnsEasy
        ✓ CorrectAnsMed
        ✓ WrongAnsDifficult
        ✓ WrongAnsEasy
        ✓ WrongAnsMed
    ▼ ✓ TestDropDown
        ✓ DropDownEqual8
    ▼ ✓ TestLockUnlock
        ✓ LockWhenLessThan75
        ✓ UnlockWhenGreaterThan75
    ▼ ✓ TestLogin
        ✓ LoginUserNameAndPwdinDB
        ✓ LoginUserNameAndPwdNull
        ✓ LoginUserNameinDBAndPwdWrong
        ✓ LoginUserNameNotinDBAndPwdCorrect
        ✓ LoginUserNameNotinDBAndPwdWrong
        ✓ LoginUserNameNullAndPwdNull
        ✓ LoginUserNameNullDBAndPwdWrong
    ▼ ✓ TestMoving
        ✓ AMovesLeft
        ✓ DMovesRight
        ✓ SMovesBack
        ✓ WMovesForward
```

```
[UnityTest]
public IEnumerator LoginUserNameAndPwdinDB()
{
    UserName="swati";
    PassWord="ssad";
    int result = login.CheckDB(UserName, PassWord);
    yield return new WaitForSeconds(0.1f);
    Assert.AreEqual(result, 1);
}
```

Testing Scripts for GalaxSE Game

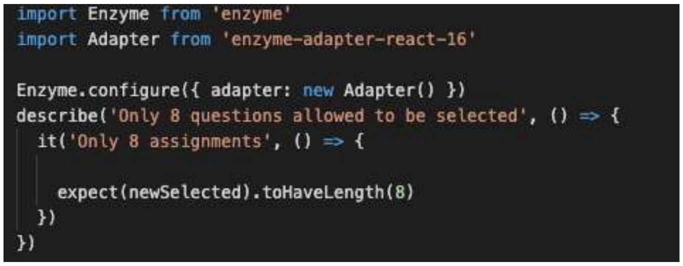
In Unity Unit testing, every unit test we write makes up part of a test suite. The test suite houses all unit tests related to a logical grouping of functionality. If any individual test in a test suite fails, the entire test suite fails. For example in TestLogin, if the unit testing called ‘LoginUserNameAndPwdinDB’ fails, the entire test suite will fail.

We also perform Unit testing for our Teacher’s web application using jest and enzyme packages. Shown below is the list of unit test cases carried out as well as an example of a unit test for selecting questions for the assignment which tests whether the application allows maximum 8 questions for the assignment.



```
PASS  src/tests/viewLab.test.js
PASS  src/tests/viewStudents.test.js
PASS  src/tests/viewQuestions2.test.js
PASS  src/tests/loginPwdWrong.test.js
PASS  src/tests/loginPwdCorrect.test.js
PASS  src/tests/loginEmailWrong.test.js
PASS  src/tests/assignment.test.js
PASS  src/tests/loginEmailCorrect.test.js

Test Suites: 8 passed, 8 total
Tests:       8 passed, 8 total
Snapshots:   0 total
Time:        4.264s, estimated 13s
Ran all test suites related to changed files.
```



```
import Enzyme from 'enzyme'
import Adapter from 'enzyme-adapter-react-16'

Enzyme.configure({ adapter: new Adapter() })
describe('Only 8 questions allowed to be selected', () => {
  it('Only 8 assignments', () => {

    expect(newSelected).toHaveLength(8)
  })
})
```

Testing Scripts for Teachers Webapp

7.2 Integration Testing

Integration testing is performed to expose defects in the interfaces and in the interactions between integrated components or systems. We will be checking whether each unit is interacting correctly and message passing between two units are functioning properly. The integration test will be done after unit testing is being done on each sub system.

7.3 Sub System Testing

At this stage, all integrations between components will be done and the system would be completely functional. A validation test will be done to ensure that requirements of the system meets the specifications of the SRS in terms of its functional and non-functional requirements. The system testing will also surface limitations that the software will face.

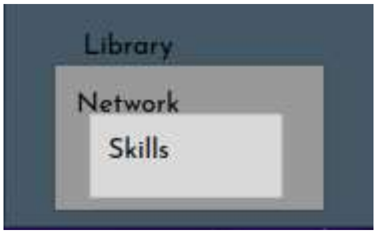
We will be doing so manually using test cases. (refer to test cases document for more information). It would be more focused on UI and the User based. We would be focusing on transitions between scenes and ensuring that the application displays the correct information and scene when certain actions are being executed.

8. Innovative Solution

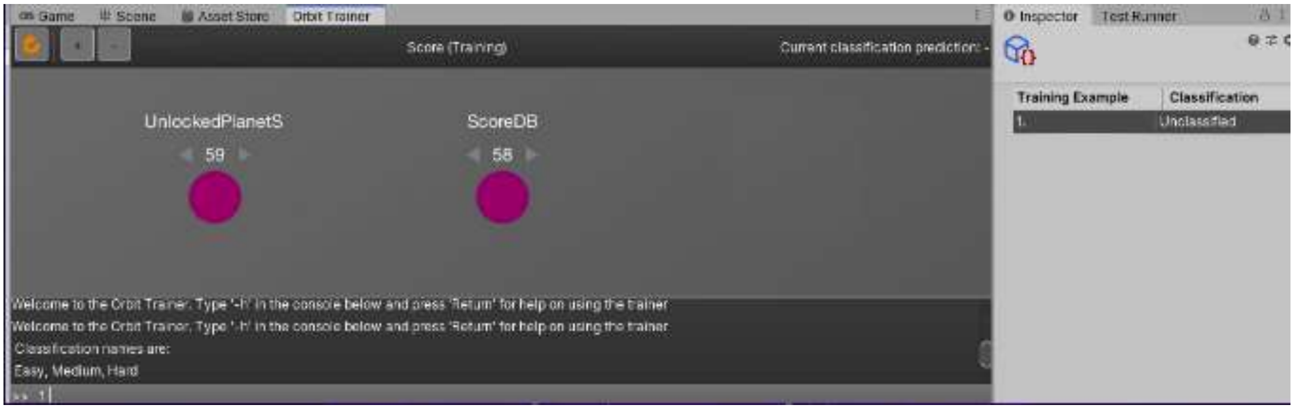
8.1 Machine Learning

Different students have different learning abilities. Therefore, we wanted the game to be scaled according to the abilities of the students. To do this, we have used a combination of machine learning and adaptive questioning in real-time. That is, the first question for a monster is generated based on the user's past history and the questions thereafter are based on the user's ability to answer the generated questions.

This functionality was implemented using the *coAdjoint Orbi*t Unity plugin. In this plugin, three nested data structures- libraries, networks and skills - must be created. Libraries contain networks and networks contain skills.

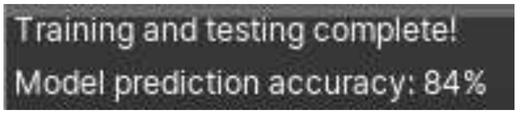


For our game, the skills used to train the inbuilt Support Vector Machine (SVM) classifier were the user's current unlocked level and total score so far both of which were retrieved from the database.
During training, normalised values for these two skills were randomly generated and one of the three possible labels (easy-1, medium-2 and hard-3) were assigned to each training example as shown below.



Training Example	Classification
1.	Unclassified

```
Welcome to the Orbit Trainer. Type '-h' in the console below and press 'Return' for help on using the trainer
Welcome to the Orbit Trainer. Type '-h' in the console below and press 'Return' for help on using the trainer
Classification names are:
Easy, Medium, Hard
>> 1
```



```
Training and testing complete!
Model prediction accuracy: 84%
```

Once the model was trained, it was tested on 50 more examples and an accuracy of 84% was reported.

Thus, during gameplay, questions are chosen according to the prediction of the SVM classifier. In real-time, difficulty is adjusted according to an adaptive questioning algorithm written by us. A combination of these have been used to maximise the ability of the game to adapt to both the user's ability as well as to their knowledge about the topic at hand.