

Project Report-318

QUIZ FUN



Submitted By:

SrNo	Student ID	Name of Student
1	1993645	Sukhmanpreet Kaur
2	1993480	Bindu Bindu
3	1995966	Darshan Joshi
4	1993770	Parwinder Kaur
5	1993696	Bipan Dadra
6	1993901	Kulwinder Kaur

Guided By:

Prof. Lucas Brantecula

Prof. Soumaya Chaffar

TABLE OF CONTENTS

1. Introduction

1.1 Quiz Fun

1.2 Purpose

1.3 Scope

2. General Description

2.1 Product Function

2.2 User Characteristics

3. Specific Requirement

3.1 Functional Requirements

3.2 Non-Functional Requirements.

4. Analysis Model

4.1 Use Case Diagram

4.2 Use Case Scenarios

4.3 Class Diagram

4.4 Sequence Diagrams

4.5 Activity Diagrams

5. User Stories

6. Database Schema

6.1 Queries

6.2 ER Diagram

7. Web Calls

8. Screen Flow Diagram:

8. Test case Scenarios

References

1. INTRODUCTION

Quiz Fun application provides different types of quiz tests related to different subjects and topics. This application contains different fields of knowledge like Computers, Mathematics, Learning French, and Science. Many applications are available for the quiz for fun, but our application provides knowledge along with fun. With the help of this application students can not only prepare themselves for different competitions, but they can also gain more and more knowledge even if they would be wrong, they can get right answers afterwards. Users can practice their knowledge using smart phones. This is an interesting and easy way to earn knowledge and learn new things.

1.2 Purpose

The main purpose of this application is learning, gaining and improving knowledge. Users can learn a different language with the help of this application. It also provides interesting quiz tests about computer hardware and software, science, mathematics etc. This is not it; this app will allow users to make their sets of Questions/ answers of their own interest and in this way, they can memorize more easily.

1.3 Scope

Scope of our application is to check and provide the knowledge. Sometimes, users use quiz applications as a pre-test or sometimes as a post-test, so it is easy for them to check their knowledge and increase it with a simple and easy way.

1. This can be used in Educational institutions as well as in the Entertainment world.
2. Can be used anywhere any time as it is a web-based application.
3. Many Categories are included in this Quiz Fun Application.
4. Easy way to test knowledge.
5. User-friendly app for easy understanding.

2. GENERAL DESCRIPTION

2.1 Product Function

Quiz Fun app is a digital platform which allows users to test their level of education by taking quiz tests.

2.2 User Characteristics

This application provides Quiz tests related to different subjects. This application allows users to search and choose quiz tests according to their topic of interest to check their level of knowledge. During the quiz test, a user can skip the question and exit the quiz if required. After choosing the answer, the user can get the answer if it is right or wrong. In case of choosing the wrong answer, the user can get the right answer too. This application will provide immediate results after submitting the test. Users will be allowed to give feedback about their test and application. This application provides a user facility to update personal information as well.

3. SPECIFIC REQUIREMENTS

3.1 FUNCTIONAL REQUIREMENTS

Requirement ID	Requirement Statement	Must/Should/Could/Would	Comments
FR001	A user must be logged into the application with authenticated username and password.	MUST	The system shall save the username and password within the database.
FR002	A logged in user shall be able to sign out of the application.	MUST	The system shall logout the user of the app.
FR003	A user shall be able to reset the password of the app.	MUST	The system shall update the user password in the database.
FR004	Verification email is sent to the user whenever he/she registers for the first time on some software system.	MUST	The system shall send the verification email for authentication of the user.
FR005	A user must be able to add questions (cards) in the database.	MUST	The system shall add the questions in the database..
FR006	The user must be notified whether their input is the correct answer to the question.	MUST	The system shall check the correct answer of the question
FR007	A user must be provided the number of total questions they correctly answered out of total questions given in the form of percentage.	MUST	The system shall show the result.
FR008	A user must be able to add the ratings.	MUST	The system shall add the rating in the database.

Project Report

FR009	A user can be able to provide the questions they correctly and incorrectly answered.	MUST	The system shall show the right and wrong answers.
FR010	A user must be able to manage the profile accordingly.	MUST	The system shall update the user profile in the database.

3.2 NON-FUNCTIONAL REQUIREMENTS

Requirement ID	Requirement Statement	Type
NFR001	A user shall not be able to add more than one account in the app.	Security
NFR002	The user can edit the profile at any time and it must be updated within 5 seconds.	Performance
NFR003	The application should be available to users to use at any time.	Availability
NFR004	The username and password should be authenticated within 5 seconds by the system.	Performance
NFR005	Every unsuccessful attempt by a user to access an item shall be recorded on an audit trail .Continuous 3 unsuccessful attempts leads to blockage timeout of 3 minutes.	Security
NFR006	Maximum 1000 students can use the app at the same time.	Scalability
NFR007	A user must be able to complete any operation without assistance 90% of the time.	Usability
NFR008	The database may crash at any time due to virus and any other operating system, so the database backup must be taken by the user at any time.	Safety
NFR009	The user must have a function to pop-up error messages in case of component failure.	Availability
NFR010	All the activities work smoothly one after another.	Efficiency

4. ANALYSIS MODEL

4.1 Use Case Diagram



4.2 USE CASES

Use Case Model for Quiz Fun software is represented at different levels of abstraction using UML use case diagrams:

List of Use Cases	
<ul style="list-style-type: none"><input type="checkbox"/> UC1- Register<input type="checkbox"/> UC2 –Login<input type="checkbox"/> UC3 - Reset password<input type="checkbox"/> UC4 –Manage profile<input type="checkbox"/> UC5-Search Quiz	<ul style="list-style-type: none"><input type="checkbox"/> UC6–Create Quiz<input type="checkbox"/> UC7 –Play Quiz<input type="checkbox"/> UC8 –Update Sets<input type="checkbox"/> UC9 –Track Performance

4.2.1 UCI- Register

System:	A Desktop Application – Quiz Fun
Identifier:	UC1
Author(s):	Team 4
Version:	None
Name:	Register
Pre-condition(s):	The user is non-registered and does not have an account on Quiz Fun software
Post-condition(s):	The user is directed to the Login Page of the web application after getting registered.
Trigger:	The user has clicked on the Register button.
Normal flow:	<ul style="list-style-type: none"> ➤ The user selects the Sign Up option from the main page. ➤ The system displays a Register Page. ➤ User clicks on the Register button to create the user profile. ➤ User enters the details in the form and sets up the details. ➤ The system validates the entry. ➤ The System saves the entry.
Alternate flow:	<ul style="list-style-type: none"> ➤ The entry is not valid. ➤ The system displays error messages and notifies the error sources. ➤ The user corrects the data entry.
Exceptional flow(s):	
Related Actor(s):	Primary -Registered User Secondary -System database
Related use case(s):	Login

4.2.2 UC2-Login

System:	A Desktop Application – Quiz Fun
Identifier:	UC2
Author(s):	Team 4
Version:	None
Name:	Login
Pre-condition(s):	The user is registered and has an account on Quiz Fun software.
Post-condition(s):	The user is directed to the Quiz Fun homepage and User is ready to play the quiz.
Trigger:	The user has clicked on the Login button and accepts the terms and conditions.
Normal flow:	<ul style="list-style-type: none"> ➤ User clicks on the Login button. ➤ User login with existing Username and password. ➤ User is directed to the homepage specific to the user. ➤ The user Accepts the Terms and conditions. ➤ The system displays the Play Quiz option.
Alternate flow:	<ul style="list-style-type: none"> ➤ User clicks on “forget password”. ➤ User enters the email used for registration. ➤ Email to change password is sent to the given email address. ➤ User changes the password and confirms the changes. ➤ User is directed to the Login page. ➤ User enters the existing user name and the password. ➤ User is directed to the homepage specific to the user.
Exceptional flow(s):	Exception: The user enters the wrong password or username. 2. The system will prompt the message “Wrong Username or password, Try again” on the web browser display.
Related Actor(s):	Primary- Registered User Secondary- System database
Related use case(s):	Reset Password

4.2.3 UC3-Reset password

System:	A Desktop Application – Quiz Fun
Identifier:	UC2
Author(s):	Team 4
Version:	None
Name:	Reset Password
Pre-condition(s):	The user is registered and has an account on Quiz Fun software.
Post-condition(s):	The user is directed to the Quiz Fun homepage and the user is ready to play a quiz.
Trigger:	The user has clicked on the Forgot Password/Reset password.
Normal flow:	<ul style="list-style-type: none"> ➤ After successful login the user will be directed to the homepage. ➤ The user goes to password settings. ➤ User clicks on Reset Password. ➤ The User will reset password for security purposes.
Alternate flow:	<ul style="list-style-type: none"> ➤ User clicks on “forget password”. ➤ User enters the email used for registration. ➤ Email to change password is sent to the given email address. ➤ User changes the password and confirms the changes. ➤ User is directed to the Login page. ➤ User enters the existing user name and the password. ➤ User is directed to the homepage specific to the user.
Exceptional flow(s):	Exception: The user enters the wrong password or username. 2. The system will prompt the message “Wrong Username or password, Try again” on the web browser display.
Related Actor(s):	Primary- Registered User Secondary- System database
Related use case(s):	Manage Profile

4.2.4 UC4-Manage Profile

System:	A Desktop Application – Quiz Fun
Identifier:	UC2
Author(s):	Team 4
Version:	None
Name:	Manage Profile
Pre-condition(s):	The user is registered and has an account on Quiz Fun software.
Post-condition(s):	The user is directed to the Quiz Fun homepage and the user is ready to play a quiz.
Trigger:	The user has clicked on the Submit button.
Normal flow:	<ul style="list-style-type: none"> ➤ User enters the details in the form and updates the information. ➤ The User can also upload the picture. ➤ User clicks on the Submit button to update the user profile.
Exceptional flow(s):	Exception: The user enters the wrong password or username. 2. The system will prompt the message “Wrong Username or password, Try again” on the web browser display.
Related Actor(s):	Primary -Registered User Secondary -System database
Related use case(s):	Search Quiz

4.2.5 UC5-Search Quiz

System:	A Desktop Application – Quiz Fun
Identifier:	UC2
Author(s):	Team 4
Version:	None
Name:	Search Quiz
Pre-condition(s):	The user is registered and has an account on Quiz Fun software.
Post-condition(s):	Searched queries will be displayed after each quiz.
Trigger:	The user has clicked on the Search button.
Normal flow:	<ul style="list-style-type: none"> ➤ Questions from the selected category will start displaying. ➤ one question with 4 options will be available to users. ➤ The user selects one of 4 options. ➤ The user will move to the next questions.
Exceptional flow(s):	<ul style="list-style-type: none"> ● If any execution error happens due to database connection which does not happen frequently our system handles that in a common page.
Related Actor(s):	Registered User
Related use case(s):	Play quiz

4.2.7 UC6-Create Quiz

System:	A Desktop Application – Quiz Fun
Identifier:	UC6
Author(s):	Team 4
Version:	None
Name:	Play Quiz
Pre-condition(s):	The user is registered and has an account on Quiz Fun software
Post-condition(s):	The User information is added accordingly.
Trigger:	Users will Select the option to create a quiz and start creating and adding data.
Normal flow:	<ul style="list-style-type: none"> ➤ The System displays create a Quiz option. ➤ The User will select on the option. ➤ The user starts creating categories of their own need. ➤ The system saves the data/quiz entry for a particular user. ➤ The system displays created data to User in the list.
Alternative flow:	<ul style="list-style-type: none"> ➤ The User will select another option. ➤ The user will perform the activity according to the chosen option.
Exceptional flow(s):	<ul style="list-style-type: none"> ● If any execution error happens due to database connection which does not happen frequently our system handles that in a common page.
Related Actor(s):	Primary -Registered User Secondary -System database, Customer Service Representative
Related use case(s):	Play Quiz

4.2.8 UC7-Play Quiz

System:	A Desktop Application – Quiz Fun
Identifier:	UC7
Author(s):	Team 4
Version:	None
Name:	Play Quiz
Pre-condition(s):	The user is registered and has an account on Quiz Fun software
Post-condition(s):	The User information is added.
Trigger:	User will Select one option from the given ones and the system will display the next question.
Normal flow:	<ul style="list-style-type: none"> ➤ The System displays Each Question with Four options. ➤ The User will select one option. ➤ The user click on next Button ➤ The system validates the entry. ➤ The system saves the entry and displays another Question to User.
Alternative flow:	<ul style="list-style-type: none"> ➤ The User is not selecting any option. ➤ The system displays another question after 30 Seconds. ➤ The user selects the one option.
Exceptional flow(s):	<ul style="list-style-type: none"> ● If any execution error happens due to database connection which does not happen frequently our system handles that in a common page.
Related Actor(s):	Primary -Registered User Secondary -System database
Related use case(s):	Update Sets

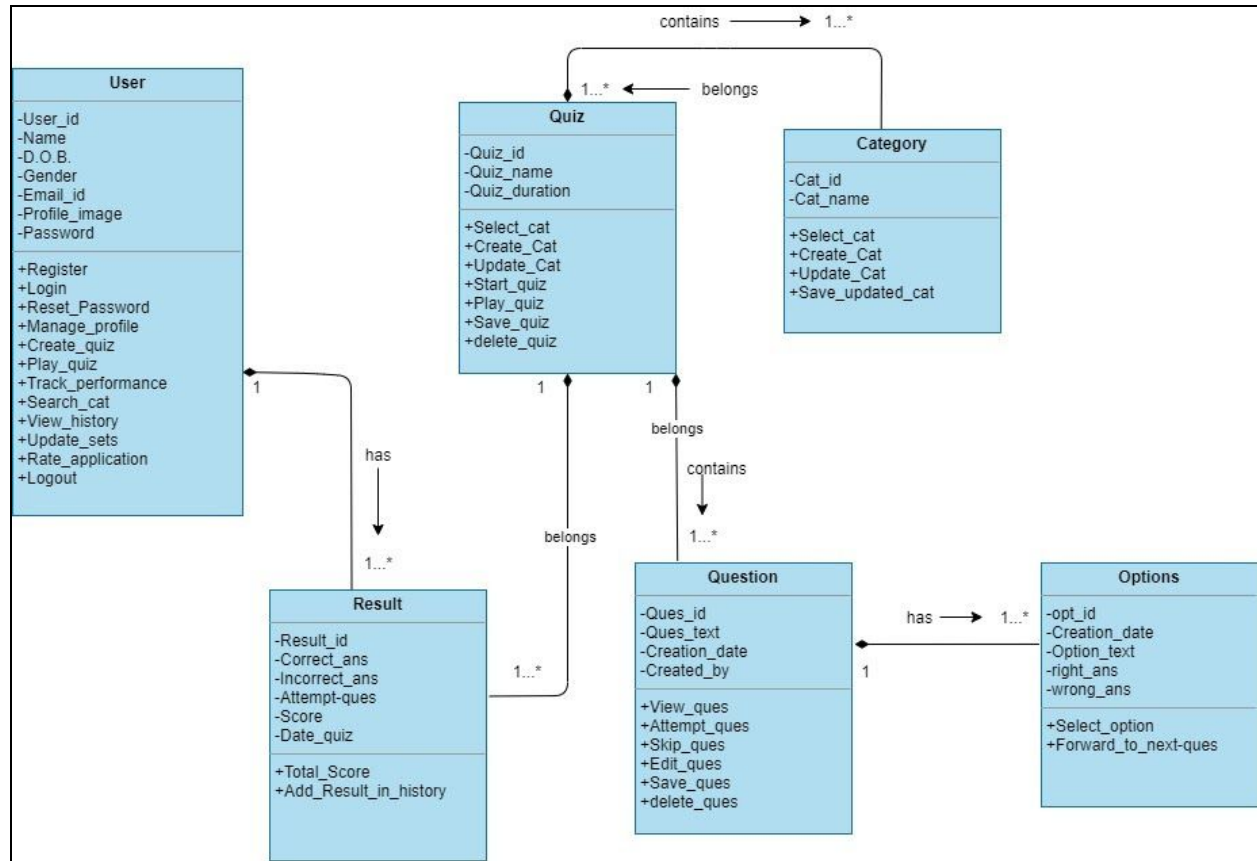
4.2.9 UC8-Update sets

System:	A Desktop Application – Quiz Fun
Identifier:	UC6
Author(s):	Team 4
Version:	None
Name:	Update Sets
Pre-condition(s):	The user is registered and has an account on Quiz Fun software
Post-condition(s):	The User will update the information as they need.
Trigger:	Users will Select the option to Update and start adding data.
Normal flow:	<ul style="list-style-type: none"> ➤ The System displays update option. ➤ The User will select on the option. ➤ The user starts adding or deleting data into categories. ➤ The user can add/delete questions,sets,categories. ➤ The system saves the data/quiz entry for a particular user. ➤ The system displays updated data to User.
Exceptional flow(s):	<ul style="list-style-type: none"> ● If any execution error happens due to database connection which does not happen frequently our system handles that in a common page.
Related Actor(s):	Primary -Registered User Secondary -System database
Related use case(s):	Track Performance

4.2.10 UC9-Track Performance

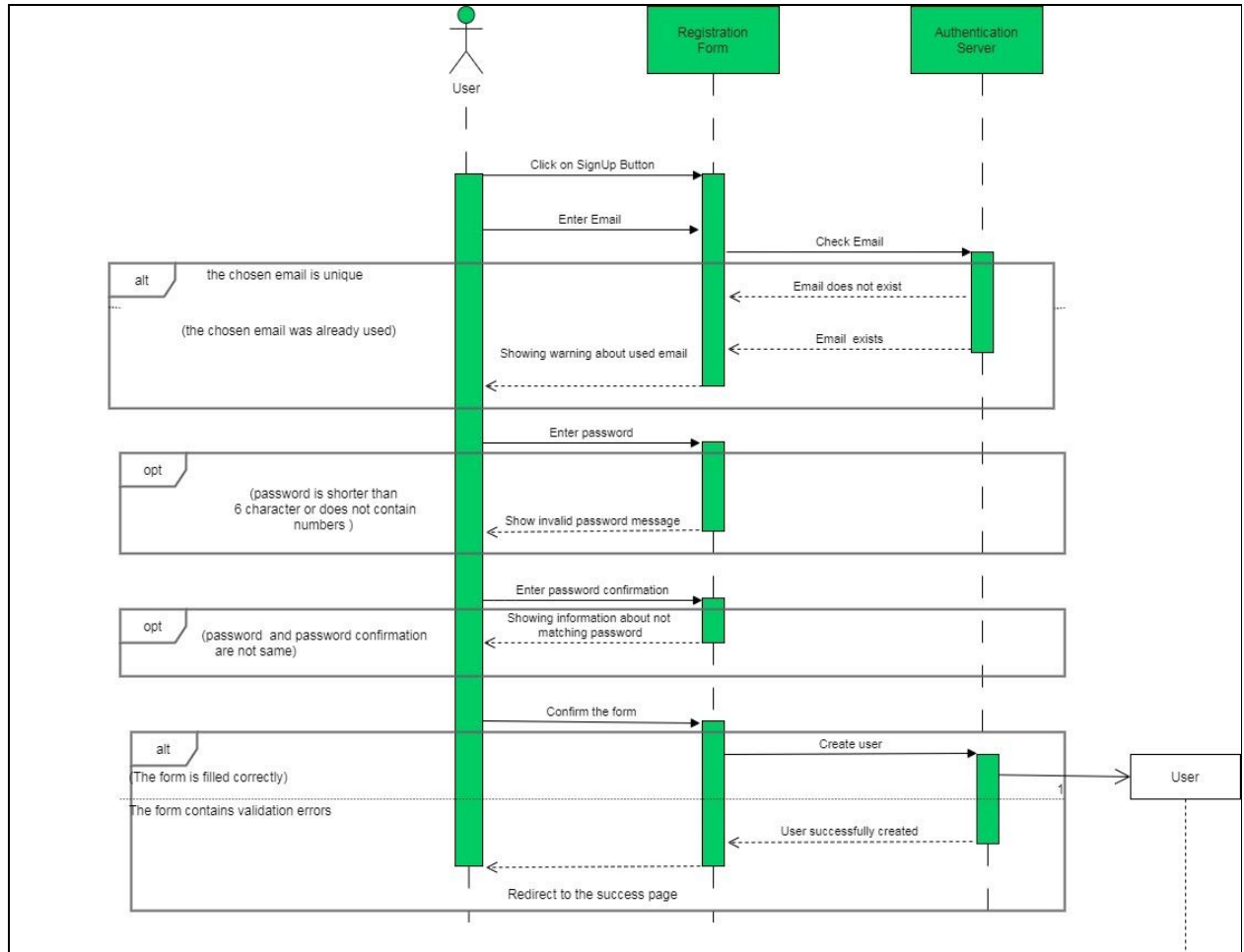
System:	A Desktop Application – Quiz Fun
Identifier:	UC8
Author(s):	Team 4
Version:	None
Name:	Track Performance
Pre-condition(s):	The user is registered and has an account on Quiz Fun software
Post-condition(s):	The user is directed to the Quiz Fun homepage specific to the user.
Trigger:	The user has clicked on the result/history.
Normal flow:	<ul style="list-style-type: none"> ➤ After submission of the quiz , the user can view the result. ➤ User will go to history ➤ Users can view the performance bar for all the played quizzes bi-weekly. ➤
Exceptional flow(s):	<ul style="list-style-type: none"> ● If any execution error happens due to database connection which does not happen frequently our system handles that in a common page.
Related Actor(s):	User
Related use case(s):	

4.3 CLASS DIAGRAM



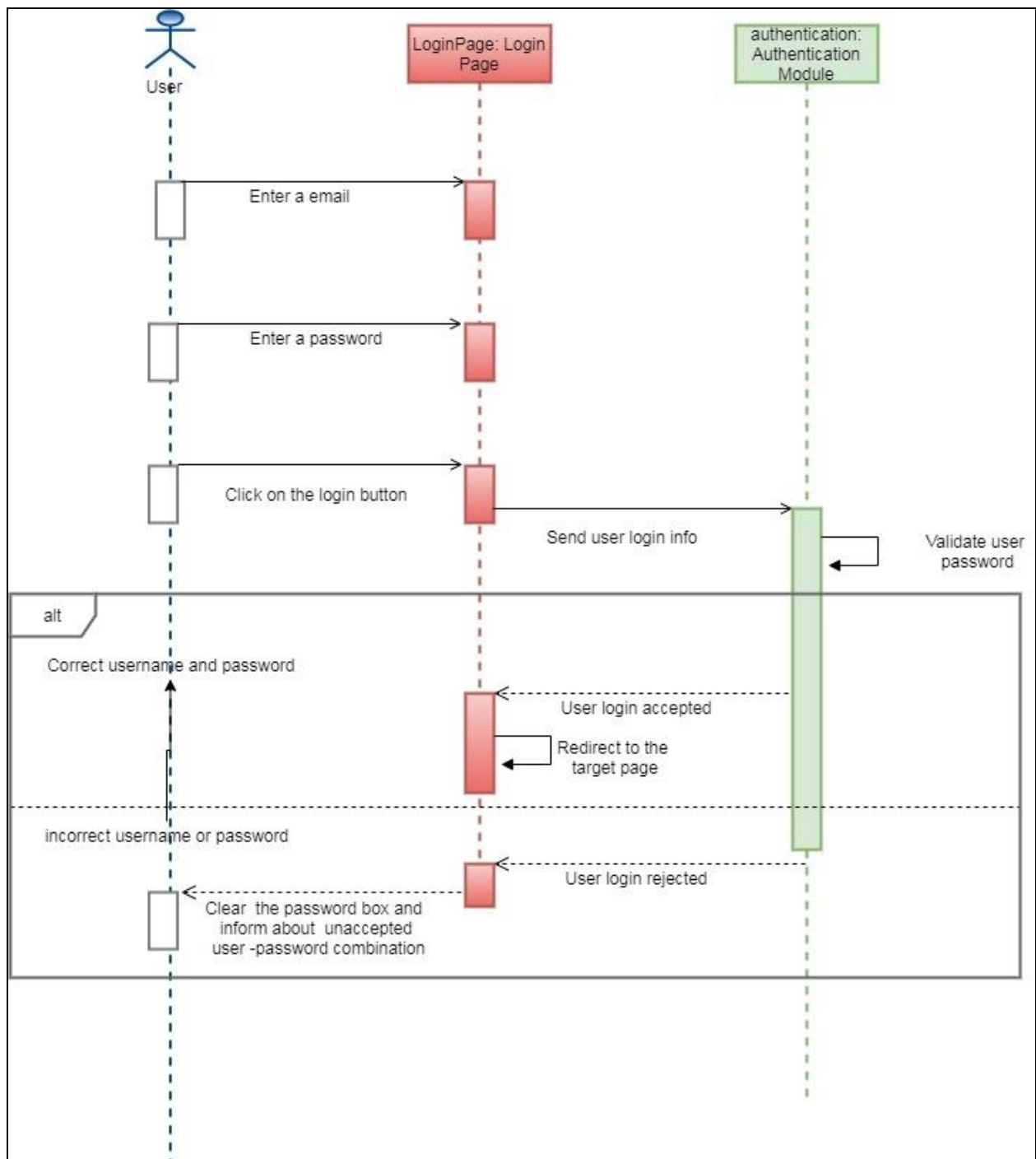
4.4 Sequence Diagram

1. Scenario: Registration to Quiz Fun (App)

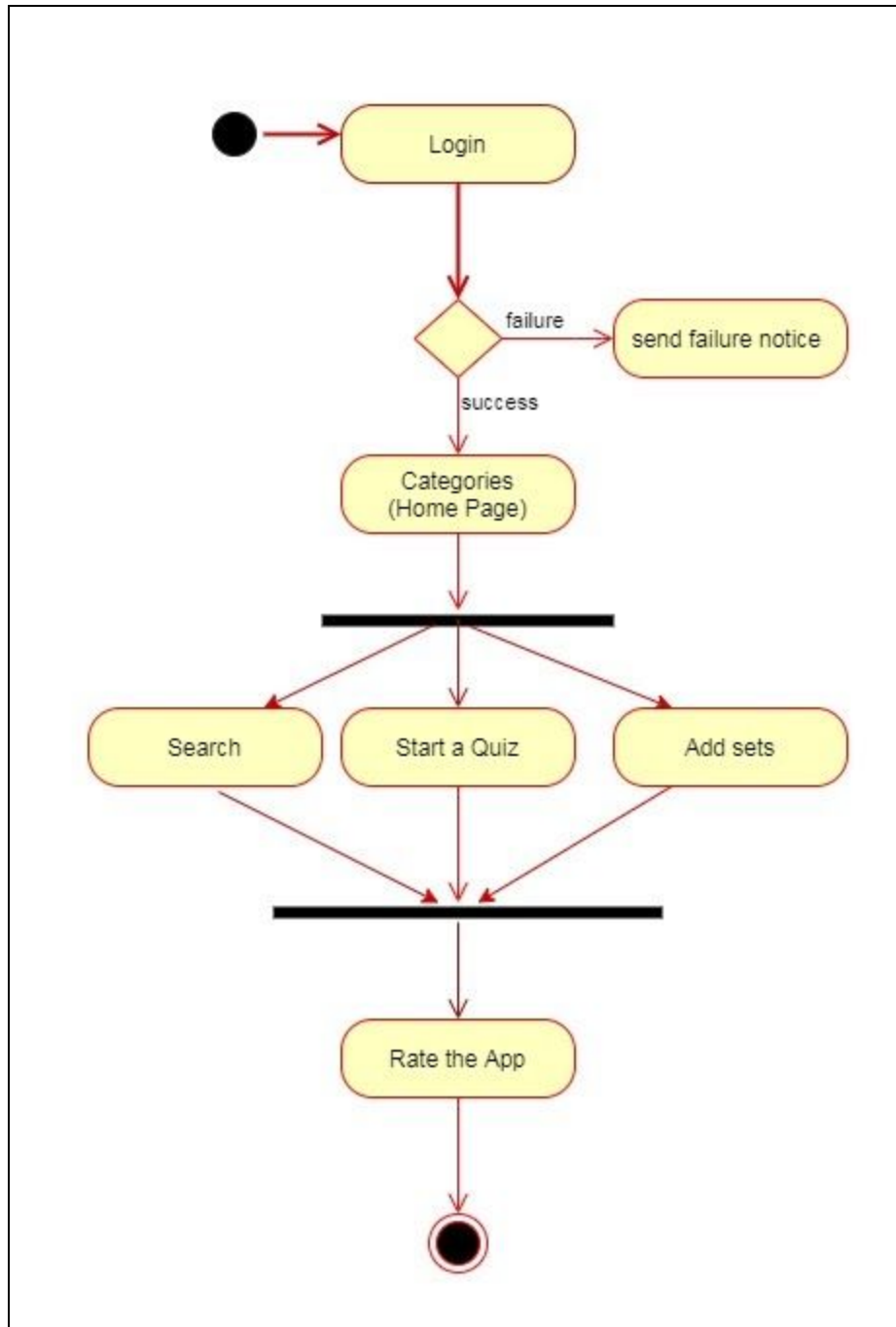


4.4 Sequence Diagram

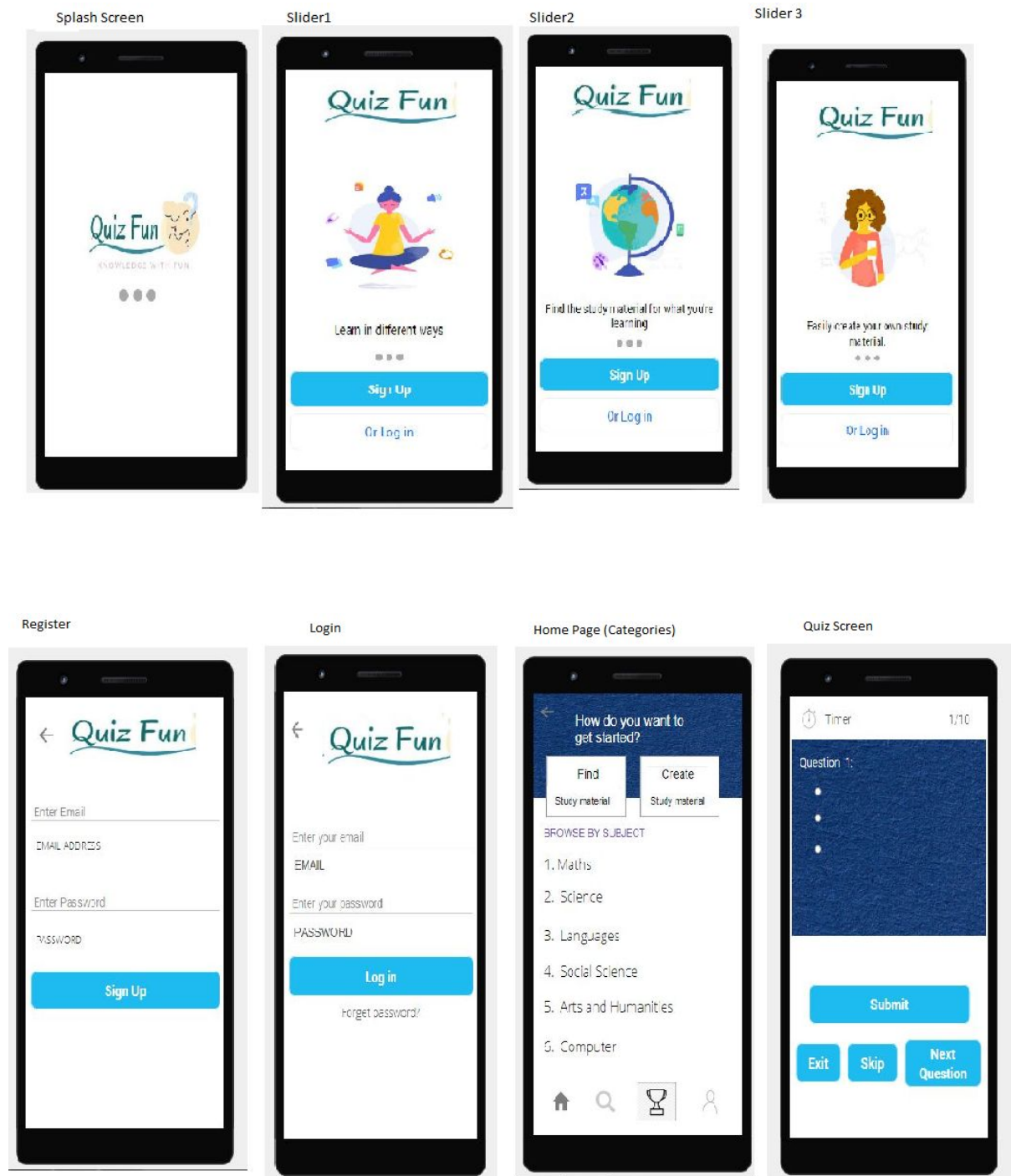
2. Scenario: Login to Quiz Fun (App)



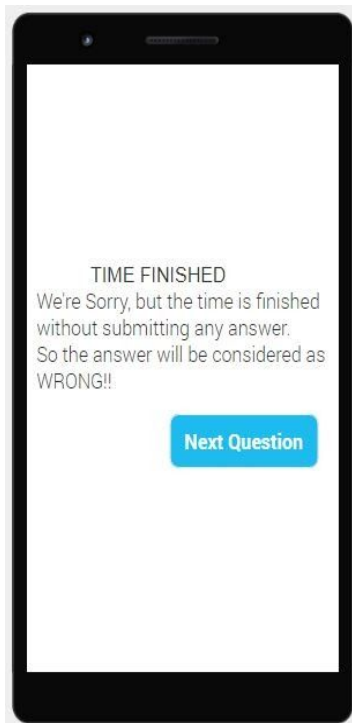
4.5 Activity Diagram



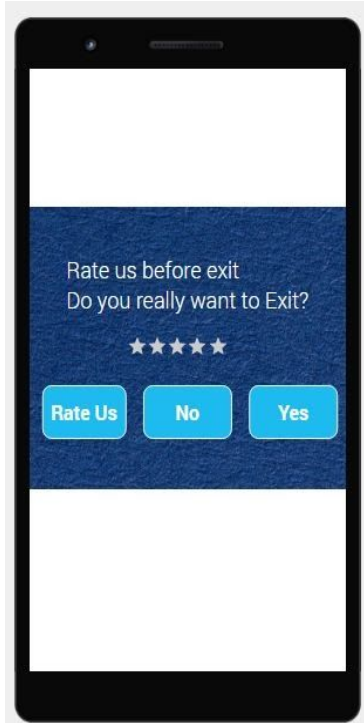
SCREENS FLOW



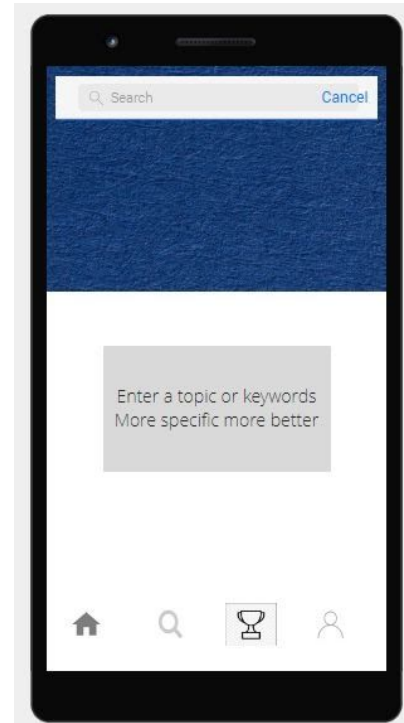
Timeout Screen



Rating Screen



Search Screen



Reset Password

A mobile app mockup for a 'Reset Password' form. The form has a white background. At the top, it says 'Reset your password' in bold. Below that, it says 'for dadrabipan@gmail.com'. Then, there is a label 'New password' followed by a password input field with a masked password '.....' and a toggle icon. At the bottom, there is a blue button with the text 'SAVE'.