Software Requirements Specification for 'Students Learning System"

Group Members

Nabeel Hussain Syed (122937)

Submitted to Sir Fahad Satti

TABLE OF CONTENTS

REVISION HISTORY

Name	Date	Reason For Changes	Version
SACS 1.0	14.11.2017	First Creation	1.0
SACS 2.0	16.11.2017	Added Introduction/Overall Description	2.0
SACS 2.1	18.11.2017	Added External Interface Requirements	2.1
SACS 2.2	24.11.2017	Added System Features	2.2
SACS 2.3	22.11.2017	Added Other Non-Functional Requirements	2.3
SACS 3.0	25.11.2017	Added Other Requirements	3.0

1 INTRODUCTION

Many Students, especially in schools and colleges, they take lectures by teachers but sometimes they don't understand the lecture completely so they suffer in exam. But as in every class there always some good students who understand the lecture so they can help those who did not. But sometimes it's possible that the good students might not understand the lecture so it becomes the problem for all students. This happens with most of students in schools and colleges then they look for academies to cater this problem and end up spending a lot of their parent's money. Even after joining the academy some students face the same problem. In that case they cannot prepare for exams properly and end up getting low grades that affect their further studies. Though some schools put their efforts to overcome this problem by arranging extra coaching classes but not every student can afford those coaching classes because it needs extra fee to attend those classes. Therefore it is still a big problem for many students and it affects their educational life on a large scale.

1.1 Purpose

The purpose of this report is mainly to explain in details the requirements of our project. This document is intended to be used by the members of the project team that will implement and verify the correct functioning of the system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints that it must operate. The parts for developers will be included in our report like data flows and which languages will be used. In addition, users can use product functions and use case diagrams to learn how the product can be used.

1.2 Document Conventions

The document is written using font "Times New Roman". Everything written in bold face text having font size of 14 and 16 are sub-headings and main headings respectively. Highlighting is to point out words in the glossary and italicized text is used to recognize diagrams/figures.

1.3 Intended Audience and Reading Suggestions

This project is a prototype designed for Student Learning System, and is based on guidance from college/university management and is intended for the use of school/college management for helping students who could not understand the lectures properly due to some reason. So they can easily cater the problems of students that they are facing in their studies.

If you intend to read this document as a developer you can start with reading the system features section of the document, reading all the functional and non-functional requirements and then heading towards the Interfaces section and if you intend to read this document as a user's perspective you should head on to the user documentation section of the document and then head towards the user interfaces section of the document to get to know the user interface.

1.4 Product Scope

The "Student Learning System" (SLS) is an Android application which makes sure that every student in class understands the lecture and preparing well for exams. There will be quizzes for students in this application from every topic based upon which it will decide which student needs help and which student can help the other student who could not understand the topic. Admin will be able to assign courses to instructors and also to the students based upon their enrollment. The instructor then can add relative topics to the respective course that will be visible to students. Instructor can also generate quizzes of different topics to know if students understood the topic or not so he can get the feedback and based upon that feedback he can help the students if any student is having issues. Student can view all the topics and there will be a quiz button with every topic which will be enabled if student has not attempted its quiz yet. After attempting the quiz he will get score and based upon that score different sessions will be created among the students who scored maximum marks and those who scored minimum marks in that quiz. In those sessions they can talk and discuss the quiz and the student with good score can help others who have problems in that topic and if the problem still remains the same then there will be a notification button that will notify the specific instructor of that topic about the student who is having issues in that topic in this he can help that student.

1.5 References

1. Example SRS reports from CENG 491 web page.
2. StarUML 5.0 User Guide. (2005). Retrieved from
    http://staruml.sourceforge.net/docs/userguide(en)/toc.html

2 OVERALL DESCRIPTION

The application will concentrate on quizzes attempted by students and based upon the grades of that quizzes the application will create sessions among the students with good score and the student with bad scores.

 Software product will be a mobile application at the client side in which instantaneous quizzes of different topic related to different courses will be seen on the screen by students and teacher can see the courses that are assigned to them by admin and the admin can see the instructor list that are available after a successful access to the system. At the server side, Operations such as creating quiz, adding instructor, assigning course to an instructor and maintaining the log of all actions. Furthermore, database operations will also be handled at the server side.

2.1 Product Perspectives

SLS (1.4) is a system that has parts in server and client part. The server performs internal functions of the system like adding instructor to database and adding quizzes modules. With this information coming from the server, clients (students) can attempt quizzes. The client program is a mobile based application that takes results from the server. There are many applications and systems in the market about helping students in studies. They are mostly used by university level students. These applications have a series of video lectures for students. However our product approaches helping students at school and college level where students usually suffer with this problem that if they could not understand the lecture there is no such app available to help them.
Our application will run on android smart phone which is available to every student nowadays.

2.2 Product Functions

a. **<u>Server functions of the product:</u>** These functions are used and run in the server of product.

- **Saving Quiz:** This function of the system is run in the server. The job of this module is taking quiz information from instructor user and save this information into database. The quiz information includes the quiz title, description, questions and their answers. This info is saved in database in the

form of different entity tables for example quiz entity and question entity. Both entities have their own attributes and connected through foreign key.

- **Database and File Management Module:** Database module will be used for storing date information coming from user interface. This will save the database info like student numbers, names and student's academic information. This module is used as a transition between different user types. This module is also used by client for retrieving information of students and courses.
- **User Authentication Module:** User authentication module is used for identifying students that are registered by the admin. When user enters the user name and password server checks it with the stored password and username in database which is stored by the admin.
- **Error Correction Module:** Error correction module is used by the system automatically for correcting any reported errors. These errors are wrong inputs entered by user, false login recognitions. Errors are reported by the user of the system.

b. **External and Client Based Functions of System:** These functions are the functions that are used by clients of the system and takes information from the server of system.
- **Login Function:** Login function is used by the user who has authentication to enter the system. Here is the actor is teacher who has the attendance control over the classroom. When the user logs he/she can the access of other related information for the system from the client computer. Figure 3 show the diagram of the function.

Figure 1: Login Function

- **Instructor View:** This function gives view to instructor type user. It takes information from database and display in instructor's view. For example this function displays the courses that are assigned to instructor by admin.

Figure 2: Instructor View Function

- **Student View:** This function gives view to Student type user. It takes information from database and display in student's view. For example this function displays the topics lectured by the instructor.

Figure 3: Student View Function

- **Admin View:** This function gives view to Admin type user. It takes information from database and display in admin's view. For example this function displays the courses that are already added to database by admin.

Figure 4: Admin View Function

- **Error Report Function:** In this function, in case of a system error such a mismatched student recognition, i.e. Student was in the class but system did not recognize him, teacher can report this to the system to prevent future mismatches. Figure 7 show the diagram of the function.

Figure 6: Error Report Function

2.3 User Classes and Characteristics

This system gives access to three different kinds of users.
1. Administrator: The school or college administrator will have administrator access means that he will be able to add, delete and modify information of the users which is stored in the database.

2. Instructor: Teaching staff will have access to only view the courses that are assigned to them and stored in the database, his/her profile and will also be able to add topics in those courses and can also create quizzes for that topics.

3. Student: Students will only have access to view the courses they are involved in and topics specific to those courses and they can also view and attempt the quizzes related to these topics.

2.4 Operating Environment

The operating environment of the product's main feature which is helping students in their preparation is based such that it should be used with the availability of internet for best performance, such as at home or where the students usually prepare for exams etc.

2.5 Design and Implementation Constraints

- The information about the students cannot be used for other purposes and cannot be distributed.
- When reporting, IEEE standards should be used and when drawing diagrams, UML standards will be used.
- We will use Android Studio, interfaces will be implemented in PHP, MySQL server will be used for database server and SQL will be used for data management. Java and XML, will be used for user interface.
- There will be only admin to control this application. The students and instructors have limited access to application.
- As this application android operating system so it can be installed and used to any android mobile device.

2.6 User Documentation

2.6.1 User manuals

The user manuals for how to use application as different roles mentioned in the application.

2.6.2 Maintenance of product

- The places where students usually study should have an internet connection to ensure performance durability

- The mobile device running the product's application for helping students should be kept virus free

2.7 Assumptions and Dependencies

The correct functioning of the system will partly be dependent on the correctness of the data stored and managed as part of the Student Learning system. Also, the application will be hosted by the server as one of many applications; the event of the server failing due to an error with one of these applications might result in service becoming temporarily unavailable. If primary database fails then backup database come into the service.

3 EXTERNAL INTERFACE REQUIREMENTS

3.1 User Interfaces

Our product is a mobile application. In our user interface, user will face to Login/Signup Screen. After successful login/signup operation there is another activity that represents some choices according to the user type. These are:

- Add Courses
    - — In this area, admin user can add different courses to the database.
- Assign Courses to instructor
    - — In this area, admin user can assign different courses to instructor.
- Add Topics
    - — In this area Instructor user can add different topics that he has covered in specific course to the database.
- Create Quiz
    - — In this area, Instructor user can create quizzes of different topics for students and save them in database for display to student users.
- View Topics
    - — In this area, Student user can view different topics related to the courses they are enrolled in.

- View/Attempt quiz
  - — In this area, Student user can view the attempted quizzes and can attempt the quizzes that are not attempted yet.

3.2 Hardware Interfaces

We will have Mobile device and server hardware components for our application. There will be a communication between mobile and the server via WIFI internet. With this WIFI connection Mobile app and server computer will be connected. Our server computer will have enough capabilities and properties.

Enough main memory requirements are the requirements for running our program seamlessly. There should be enough main memory to process related information, latest modern computers with average RAM is sufficient with small amount of data (i.e. Classes with 20 students). Enough space for storing database and information should be provided in the hard disk of server computer. This space information is estimated from the size of information that is in the server. The system saves databases to hard disk of the server computer. For a system of 20 students, the optimal database size is 40 MB. The size of the database must be bigger for bigger number of students.

3.3 Software Interfaces

- **<u>Server side:</u>** The software will be hosted on user's Mobile and will be connected to the server on the school/college. The server is listening on the web standard port, port 80. In order to access the data that are produced by the mobile application, the server must have MYSQL database system.
- **<u>Client side:</u>** The system is a Mobile application. In order to run the application properly, the user must use a smart phone with android operating system. The mobile must have an Internet connection in order to be able to access the system..

3.4 Communication Interfaces

The HTTP protocol will be used to connect server computer and client computer. Server will reply HTTP calls of client computers to send necessary information. Also there will be a USB data sending protocol for connection between camera and server computer.

4 SYSTEM FEATURES

4.1 Log in

4.1.1 Description and Priority:

The user enters his specific username and password and logs into the system. The priority for this system feature is low.

4.1.2 Stimulus/Response Sequences

Before this feature can be initiated, the user has already accessed the Login page of the system in the client side of the system.

1. The user selects the text input box for the username from the page and enters his authenticated user name to the field.

2. The user selects the text input box for the password from the page and enters his authenticated password to the field.

3. User clicks to 'Login' button after entering related information.

4. If the authentication is confirmed the user is redirected to his main page.

5. If the authentication fails, then the user takes an error message and login page comes again.

4.2 Adding Courses to database

4.2.1 Description and Priority:

The admin selects 'Add Course' screen for adding course to the database. The priority for this system feature is high.

4.2.2 Stimulus/Response Sequences

Before this use case can be initiated, the user has to login to system.

1. The user clicks to 'Add Course' link in the main page.

2. In the 'Add Course' page there are 2 options with related buttons. One option is for adding a course and second is for cancelling.

3. If the user clicks the 'Add' button, then the course will be added to the database.

4. If the user clicks to 'Cancel' button, the addition of course will be cancelled.

4.3 Generating Sessions

4.3.1 Description and Priority:

The Application uses this function to generate a discussion session between students who scored low grades and who scored good grades for helping purpose.

4.3.2 Stimulus/Response Sequences

Before this use case, the user must have attempted the quizzes so this function can decide who needs discussion session.

1. The user selects the 'Accept Session' notification from the interface.

2. A new interface opens where two students with low and high grades can discuss the topic to understand it completely.

3. After generating this session if problem still remains the same then this session will be refer to instructor of that respective topic.

4. There will be a button on top of the screen refer to instructor.

4.4 Marks Report

4.4.1 Description and Priority:

The user can view their marks report after attempting a quiz and can get feedback this way if they really have good understanding of that particular topic. The priority of this feature is high.

4.4.2 Stimulus/Response Sequences

Before this use case can be initiated, the user should login to system and he must have attempted the quiz.

1. The user clicks to Finish Quiz button in the interface.

2. With newly opened page the user can see marks report with remarks pass/fail.

4.5 Adding Topics

4.5.1 Description and Priority:

The user can add different topics to different courses. The priority for this system feature is high.

4.5.2 Stimulus/Response Sequences

Before this use case can be initiated, the user should login to system and some courses should be assigned to user.

1. The user clicks 'Add Topic' from the user interface.

2. In the new page the user can enter the information about this topic.

3. Then finally by clicking the add button the topic will be added to the respective course.

4.6 Assign Course

4.7.1 Description and Priority:

The Admin can assign different courses to different instructors. The priority for this system feature is high.

4.7.2 Stimulus/Response Sequences

Before this use case can be initiated, the user should login to system and there should be some instructor records generated by the system.

Before this use case can be initiated, the user should be logged in.

1. The user clicks to 'Assign Course' button in the interface.

2. There will be a new page where user can select the course and the instructor to which that course should be assigned.

3. After specifying properties of course defined above, the user clicks 'Assign course'

4. The course information is saved to database to run the system in exact times defined.

4.7 Functional Requirements

4.7.1 Saving Quiz Module:

This function should take the quiz information from the Instructor type user then after necessary operations saves this information in database.

The App should send quiz data to server and the server should have enough capabilities to store quiz data coming from the App in order to this module run correctly. Also necessary quiz object specification information (here the info related to topic and course) should be sent by the app.

1. The App takes quiz information from the instructor in the class.

2. The App sends the Quiz information of the Quiz class to server.

3. Server takes the quiz information and makes necessary data processing procedures like assigning auto generated ids etc.

4. After analyzing all of the quiz info the program returns the information that needs to be stored in database.

5. With the given information the data is stored to 'Database and File Management Module'.

4.13.3 Database and File Management Module:

This module handles the database and file operations that take place between and along the quiz, student and instructor' info saving modules.

1. After taking quiz info operation this module should take the related info results.

2. After taking all data information this module should do necessary data processing operations for a better data management.

3. Also this module controls database operations about client side of system.
4. After managing data the module saves newly found students and also saves courses records to database.

4.13.4 Error Correction Module:

This function handles the reported face recognition errors.

1. In some time intervals, errors that are reported by the user is checked by this module.

2. If there is a mismatch between records, the system will try to take last successful match record from the database.

3. With the last successful record retrieved, this error will be tried to be handled.

4. If there is no successful record for mismatching, then it will renew the record of mismatched student in next lectures of classrooms.

5 OTHER NON-FUNCTIONAL REQUIREMENTS

5.1 Performance requirements

Each of our servers should able to compute attendance of all classes in corresponding school. All attendance reports can be checked after the server machine finishes the computation of attendance. Cameras are able to send views of class that are have enough resolution to recognize students.

5.2 Safety requirements

Any damage to the database due to unforeseen and uncontrollable catastrophes such as disk failures, virus attack or some physical damage due to natural disasters can lead to great loss of data and the recovery process can be very long to register the face IDs of all the users who are in use of the product under a certain organization such as all the population of a college or university.

5.3 Security requirements

This system needs to implement database server for the storage of the database of the face IDs of the students/users so the security requirement is that the vendor must choose their database service/partner carefully.

5.4 Software Quality Attributes

- **Adaptability**
  The application must be adaptable to multiple environments such as android and other environments.
- **Availability**
  The Internet WIFI should be available/ON during the class/sessions only and the database of the data records should be available 24/7 for the all type users.
- **Maintainability**
  For the maintainability of this application the required updates should be installed time by time and admin should manage the application for best performance.
- **Correctness**
  The information entered by all type users should be valid and can easily be implemented without any possibilities of errors.
- **Flexibility**
  The Application should be flexible to be used in different settings such as colleges/schools and offices and government institutes as well.

5.5 Business Rules

The product will be designed such that only the administrators of the product who can be the college management such as the faculty or Principal can have special access to specific features such as adding student or instructor records or can register new users of the system while only the higher authorities can use the monitoring features for the databases.

6 OTHER REQUIREMENTS

Appendix A: Glossary

- SLS: Students Learning System

- DBMS: Database Management System

- SRS: Software Requirements Specification

- PHP: Personal Home Page

- HTTP: Hypertext Transfer Protocol

- RAM: Random Access Memory

- SQL: Structured Query Language

- UML: Unified Modeling Language

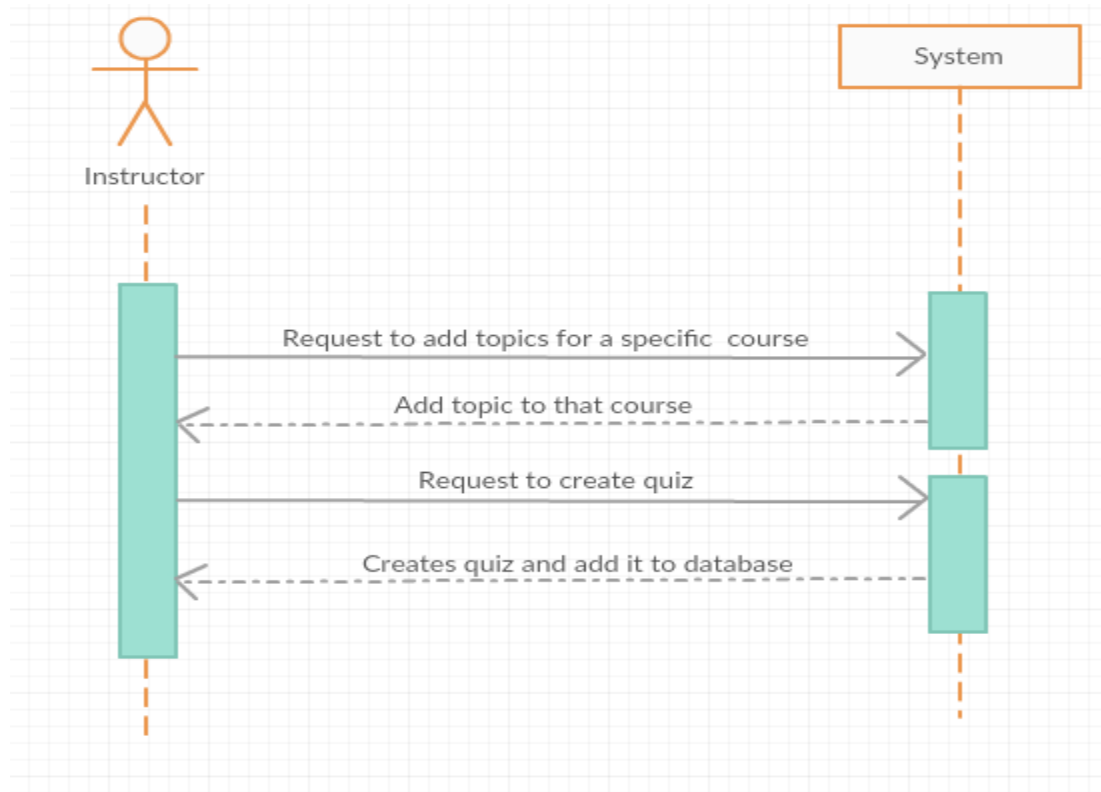- MYSQL: The world's second most widely used open-source relational database management system
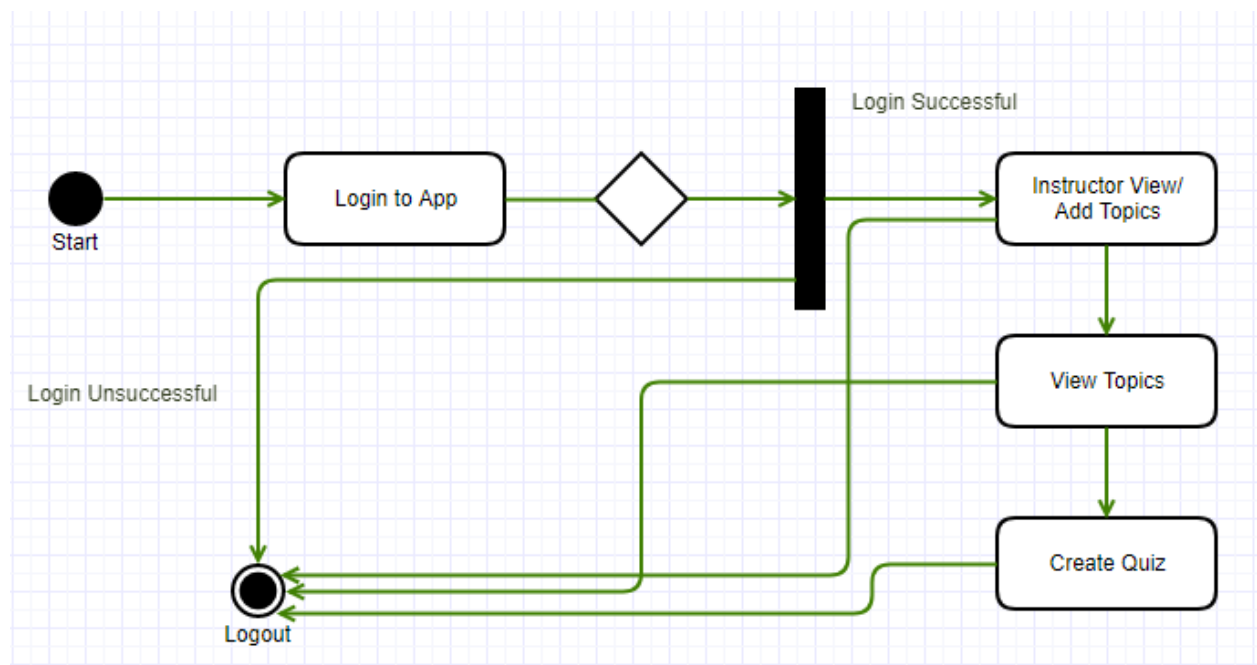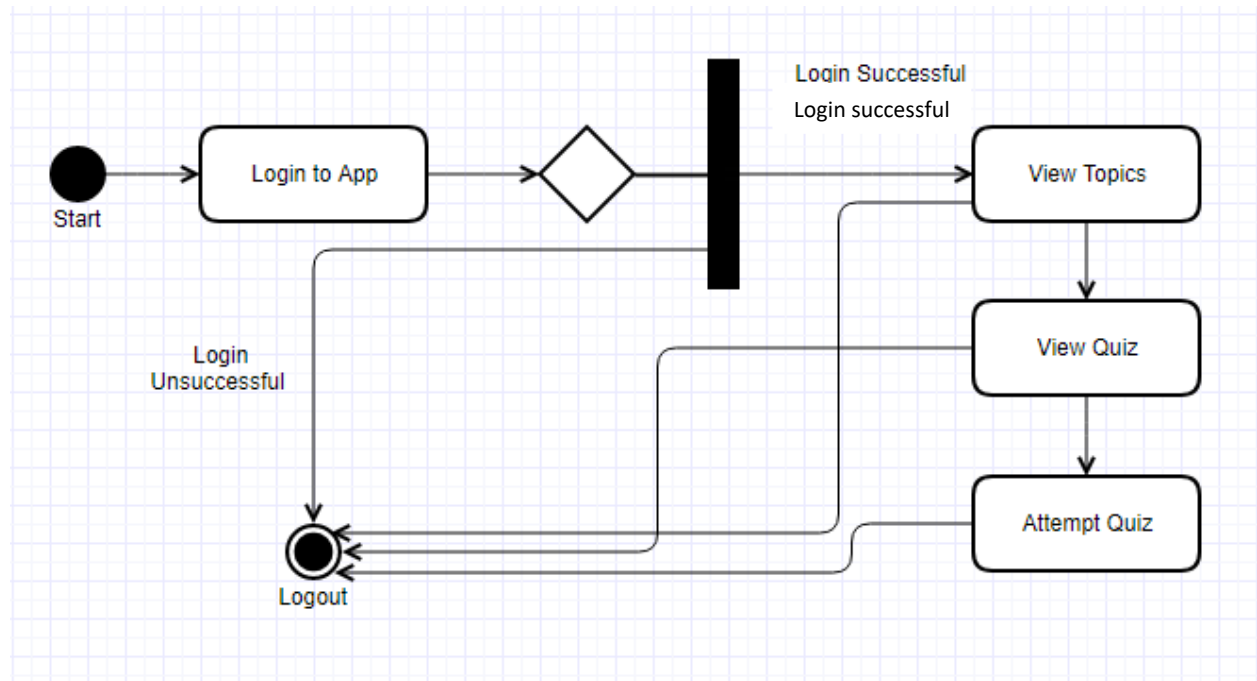
Appendix B: Analysis Models

Use Case Diagram

Sequence Diagrams

Instructor

System

Request to add topics for a specific course

Add topic to that course

Request to create quiz

Creates quiz and add it to database

Admin/Student/Instructor

Login page

Database

Admin/Student/Instructor home page

Press login button

Press login for account

ValidateUser()
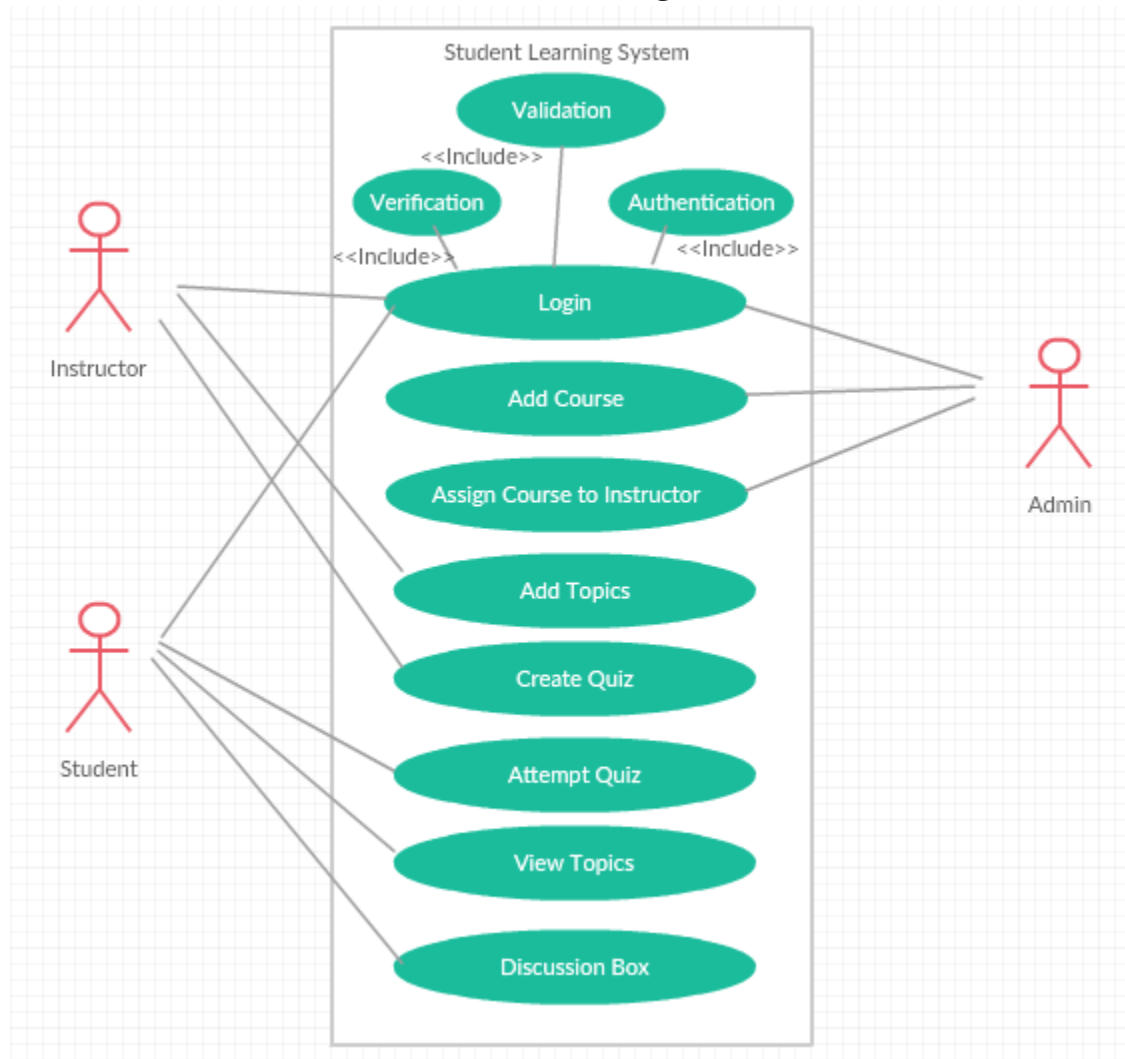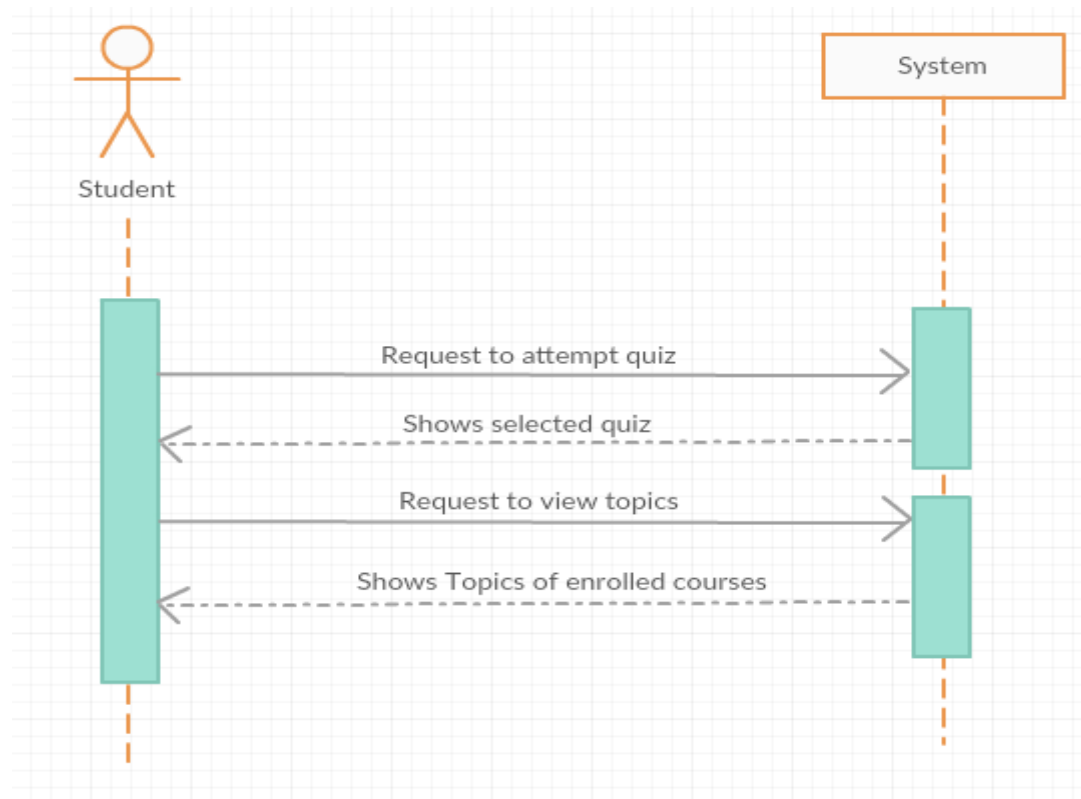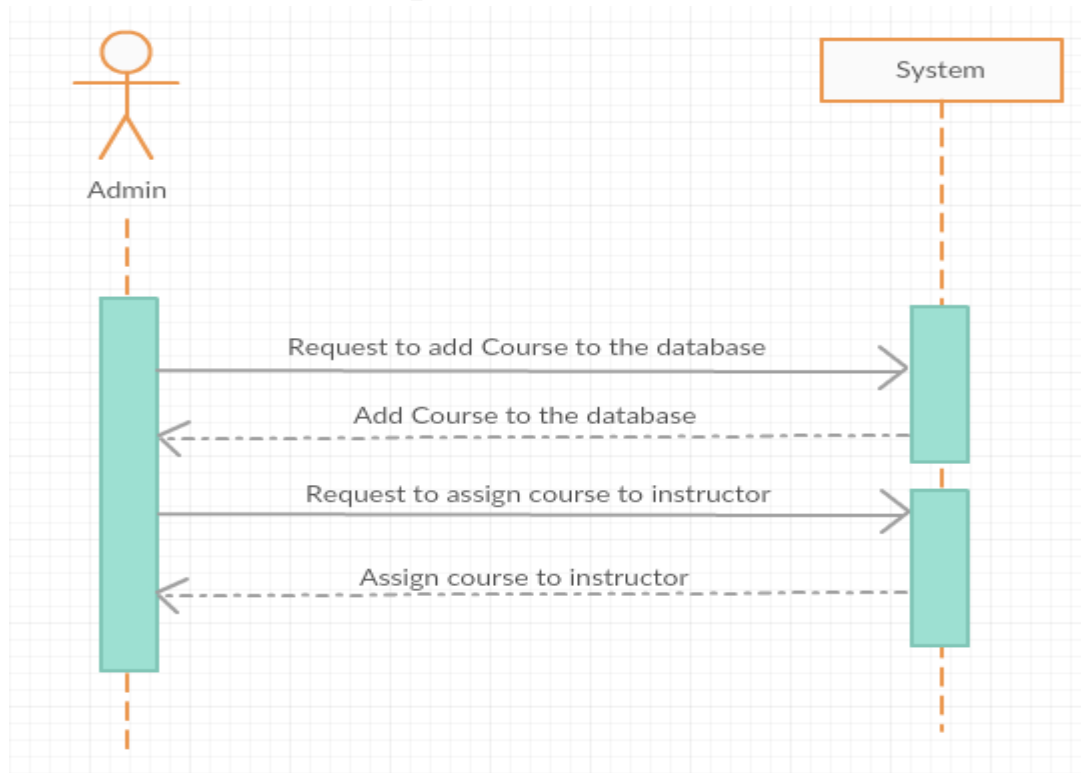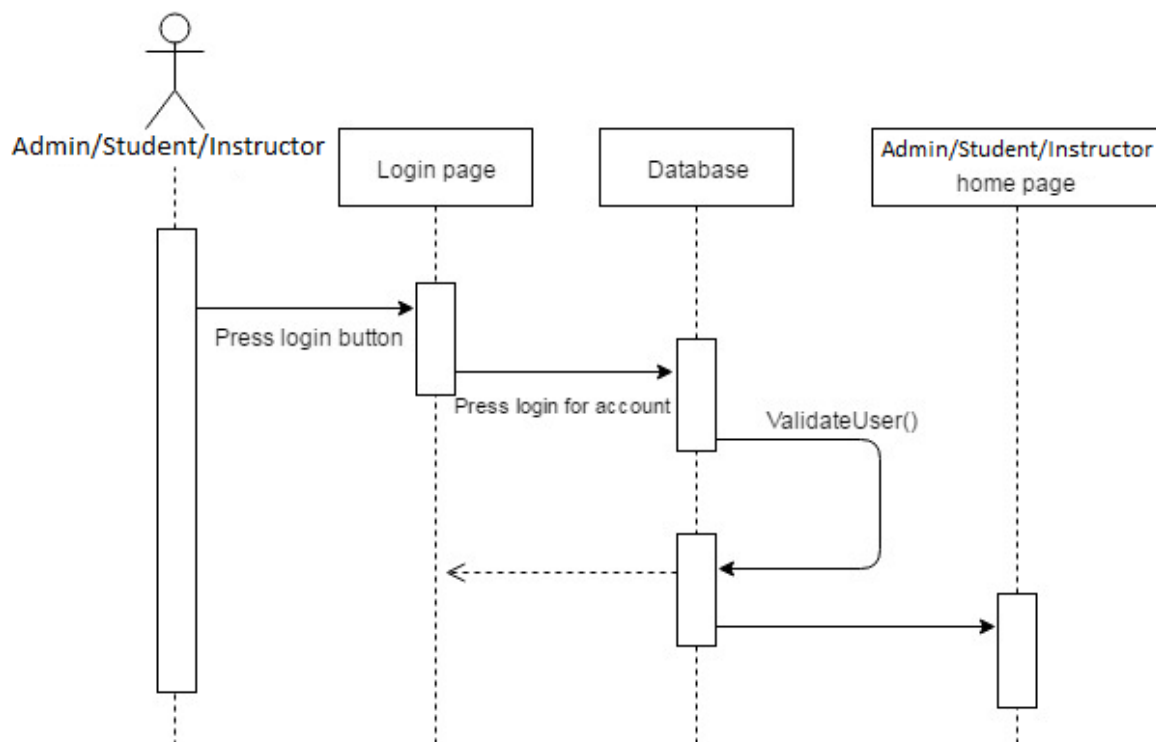
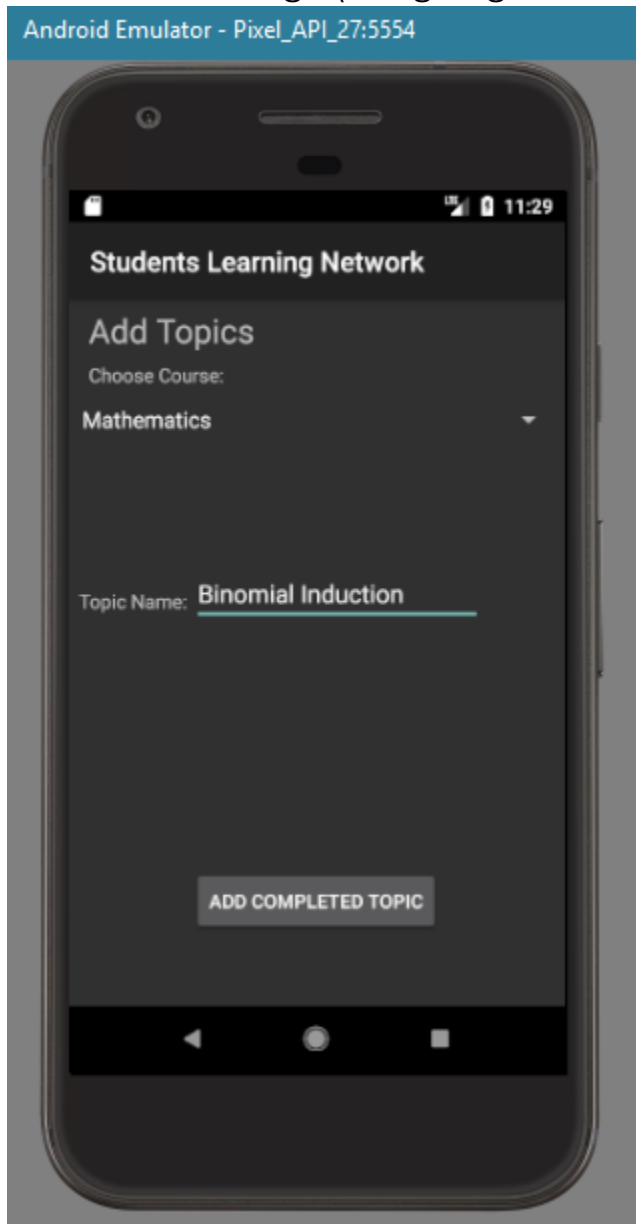## *Activity Diagrams*
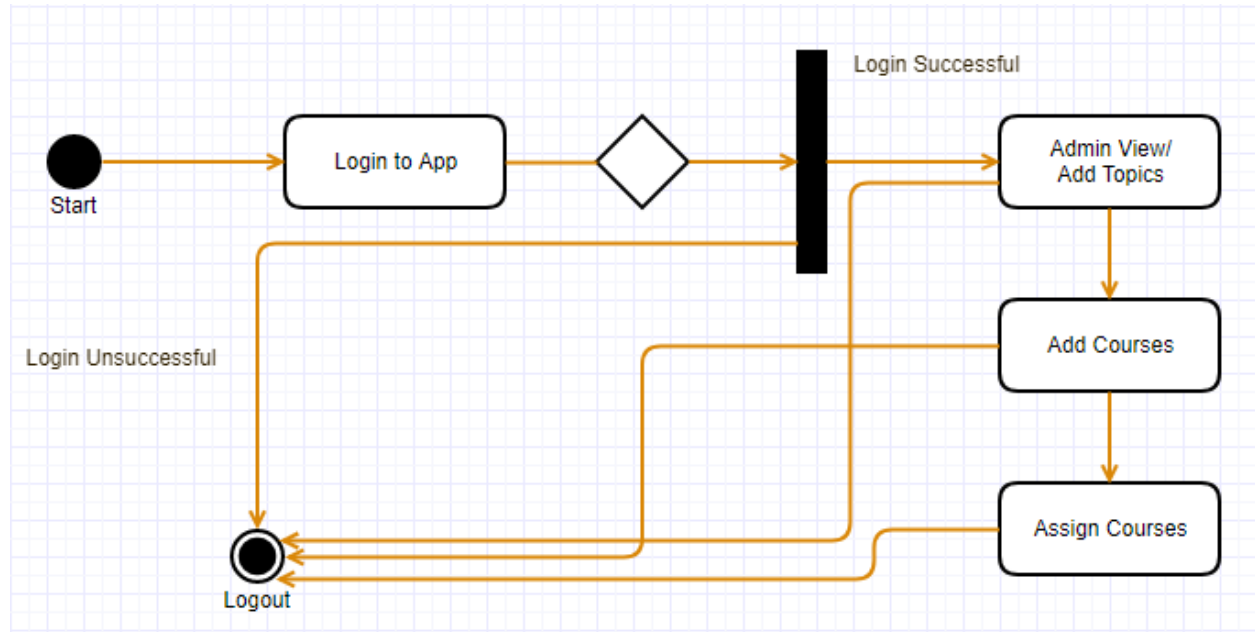
E-R Diagram
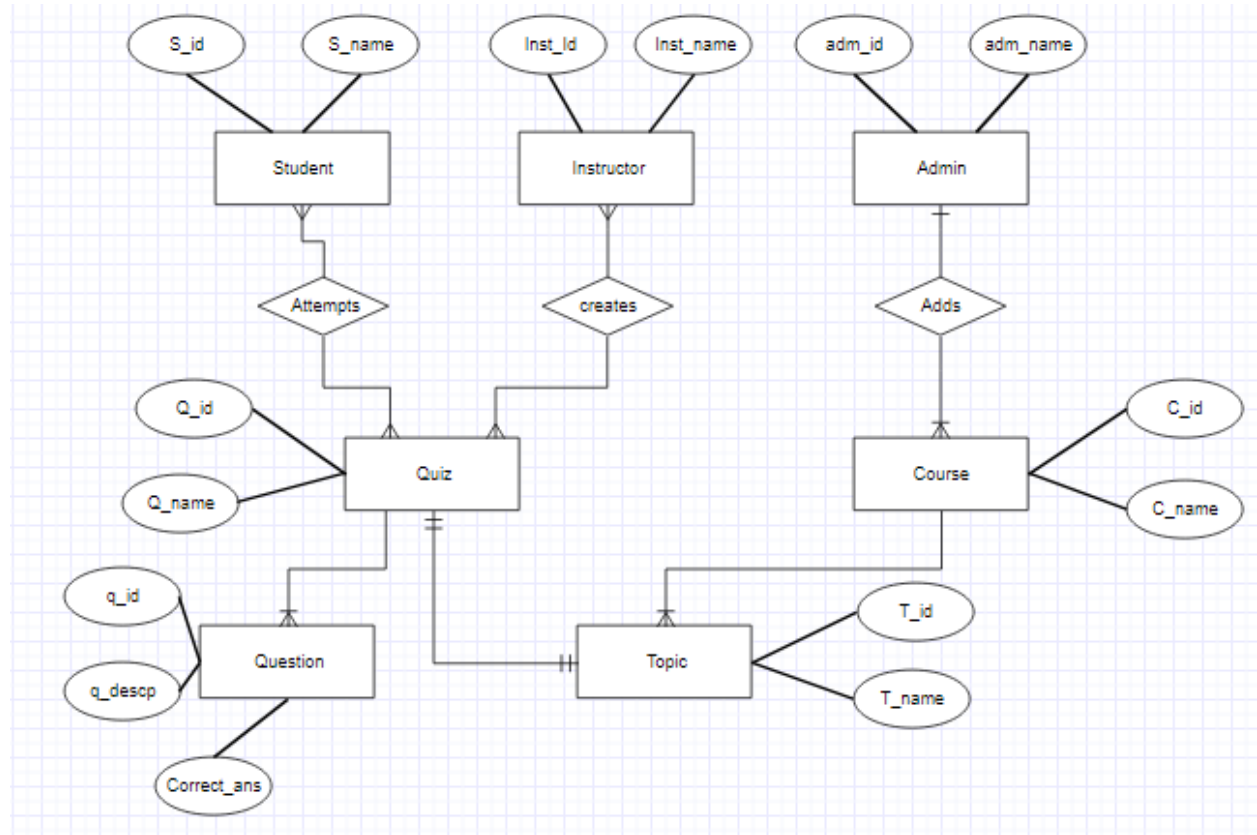
Class Diagram
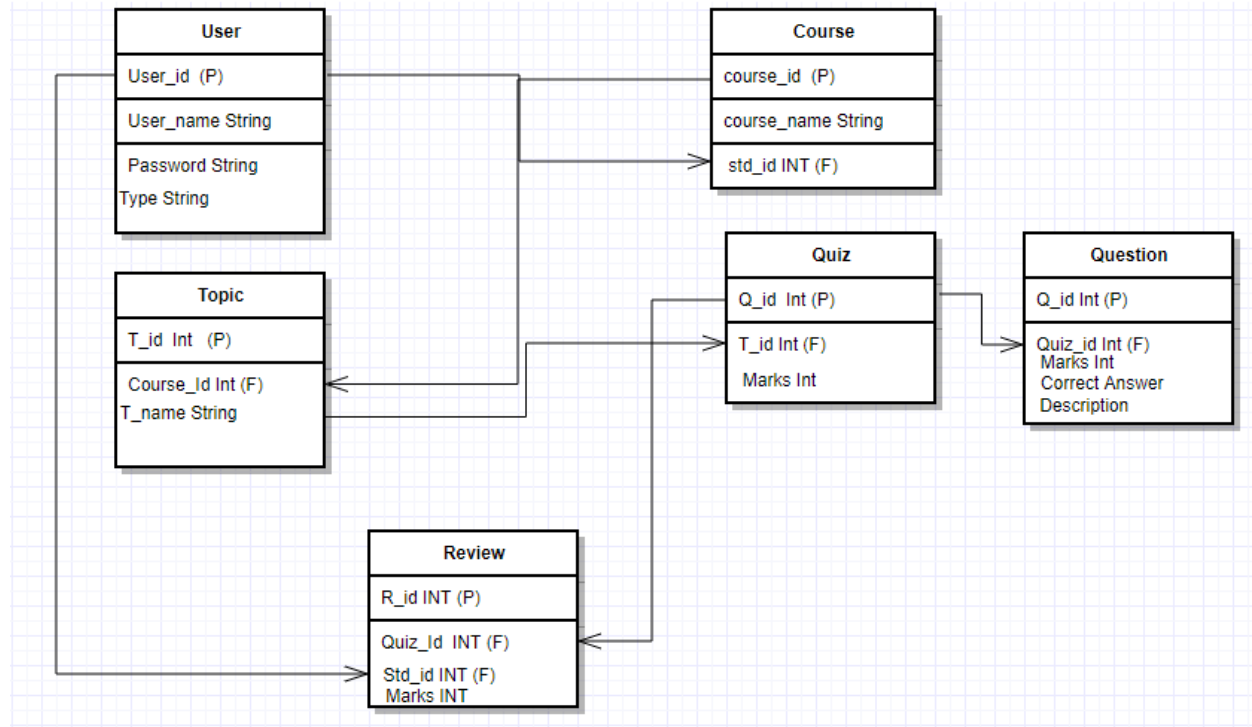
User Interfaces

Login Page:
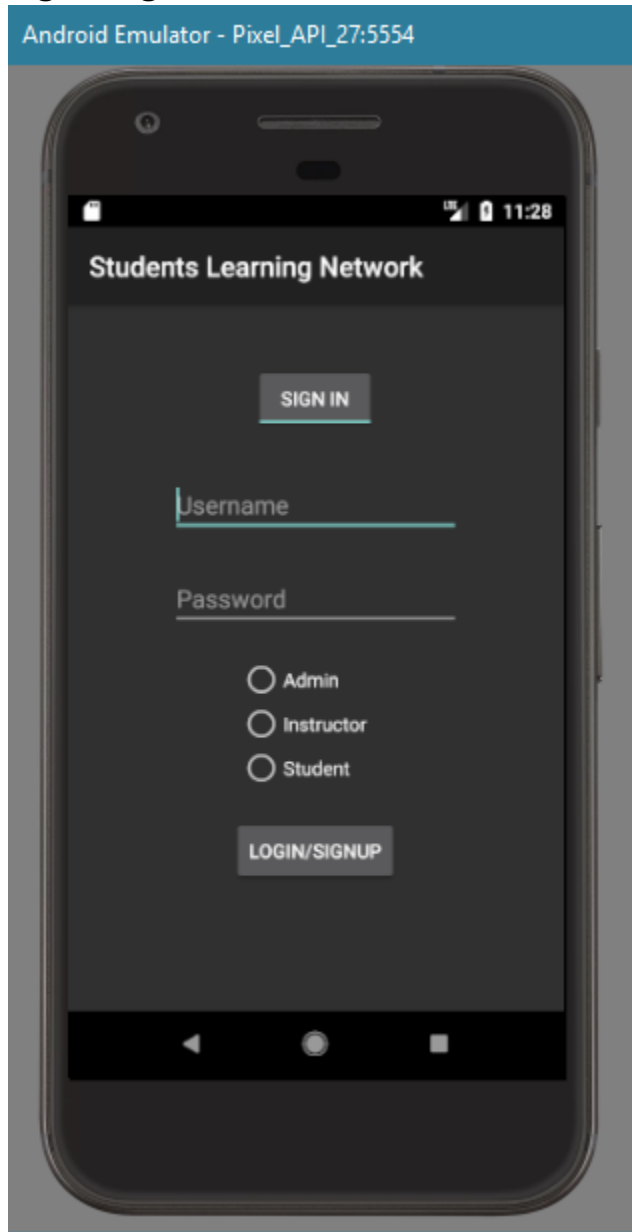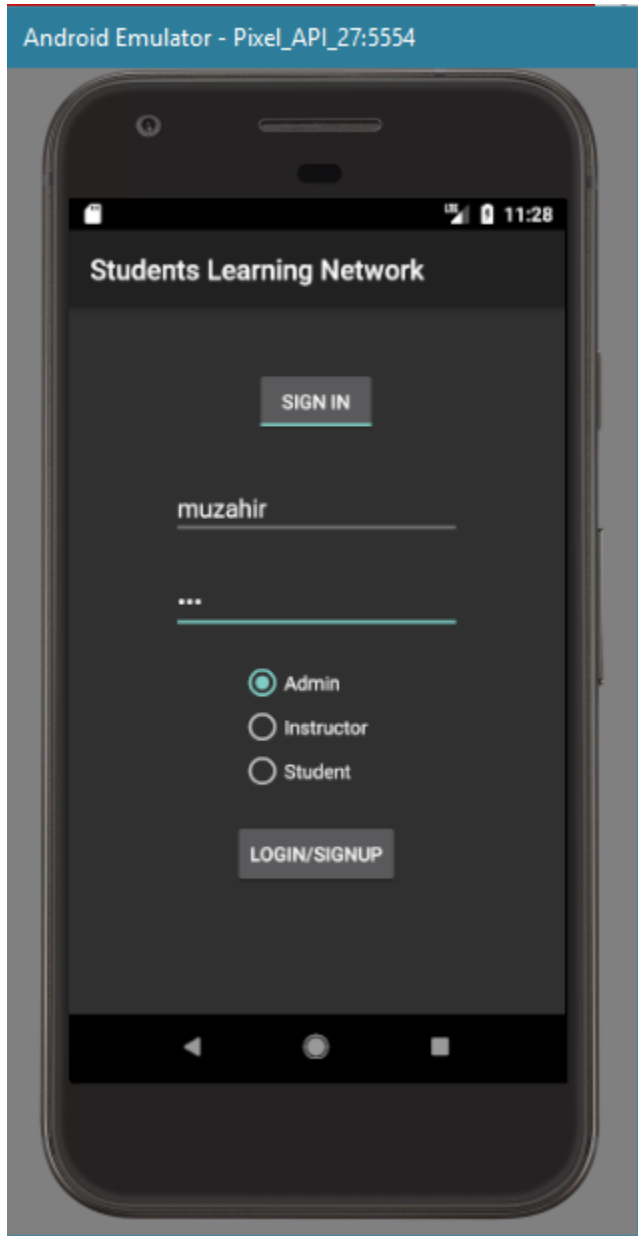
Admin Home Page (Assigning Courses to Instructor):
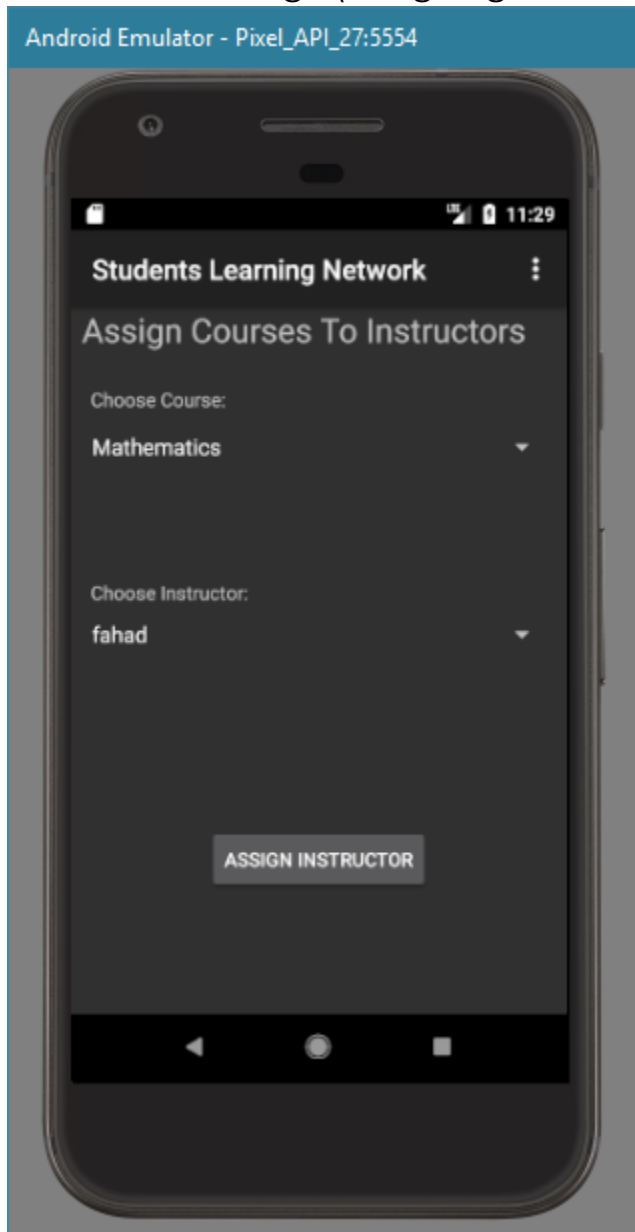
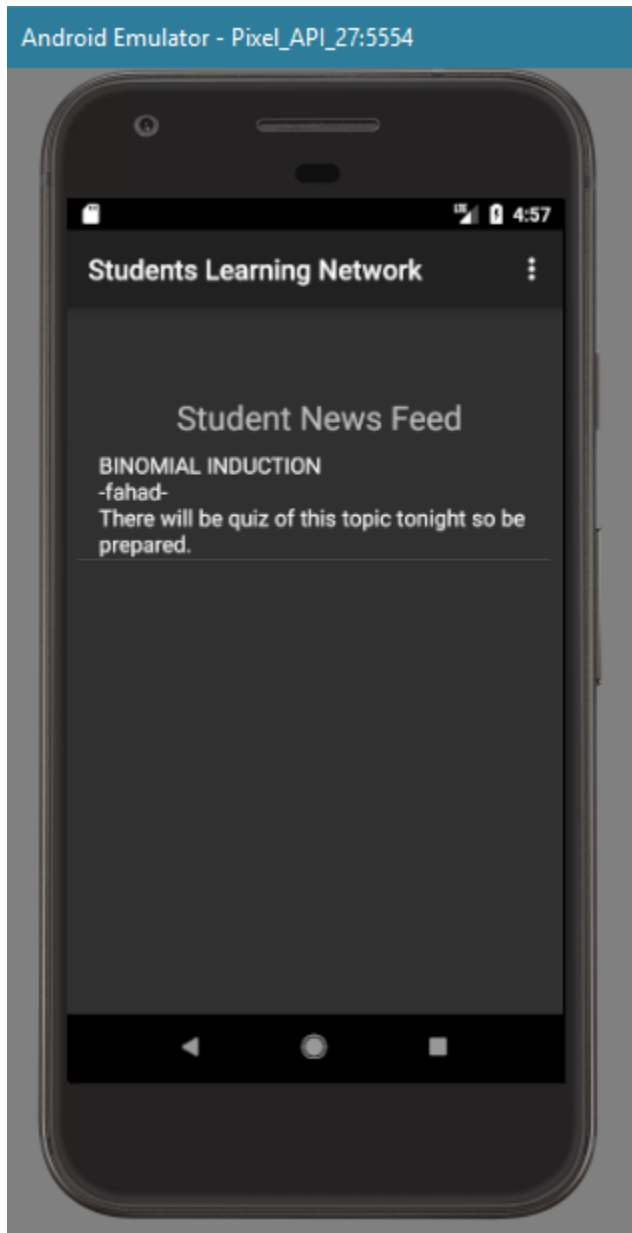## Admin Home Page (Assigning Courses to Instructor):
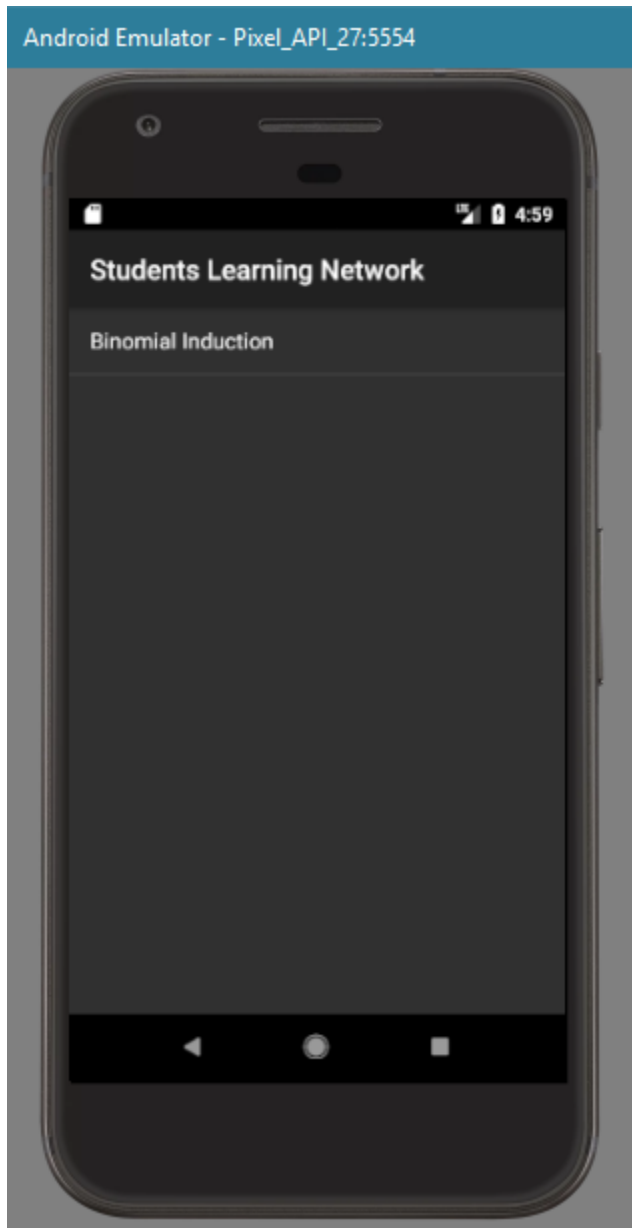
Instructor Checking his Newsfeed:
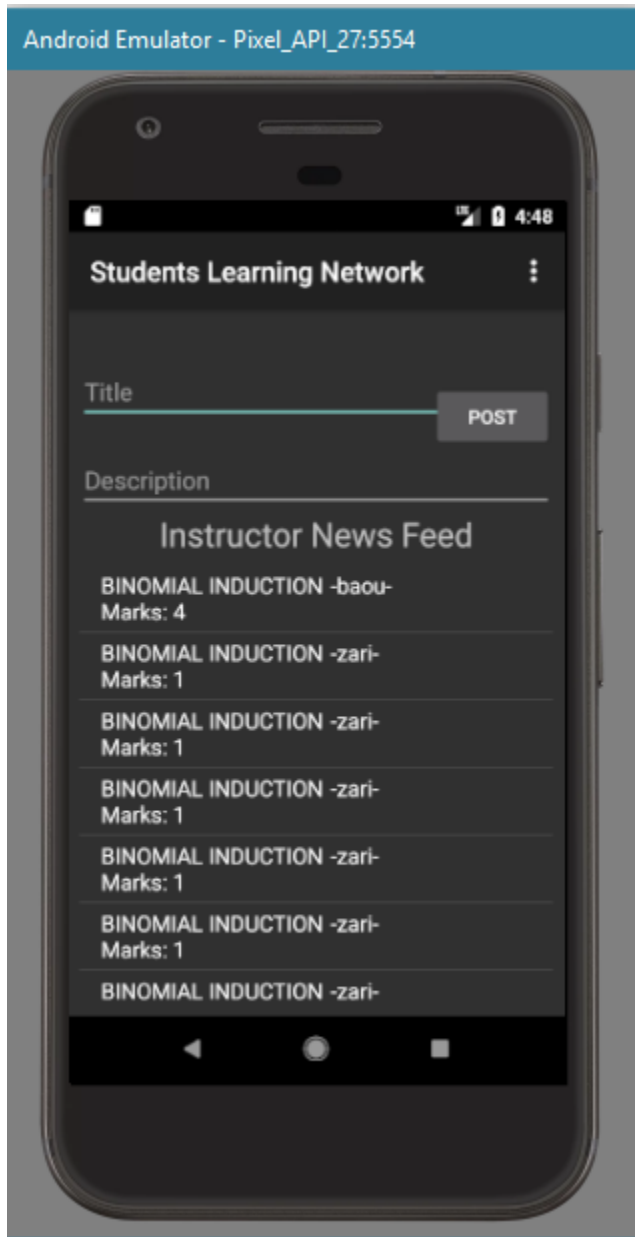
Instructor creating a quiz:
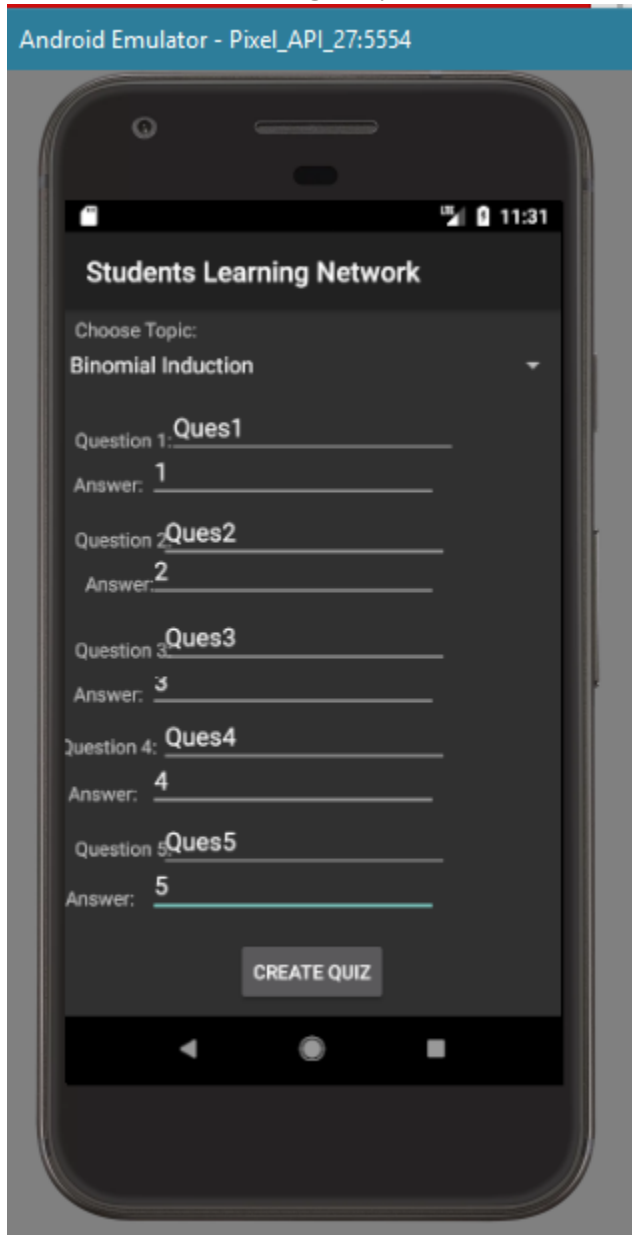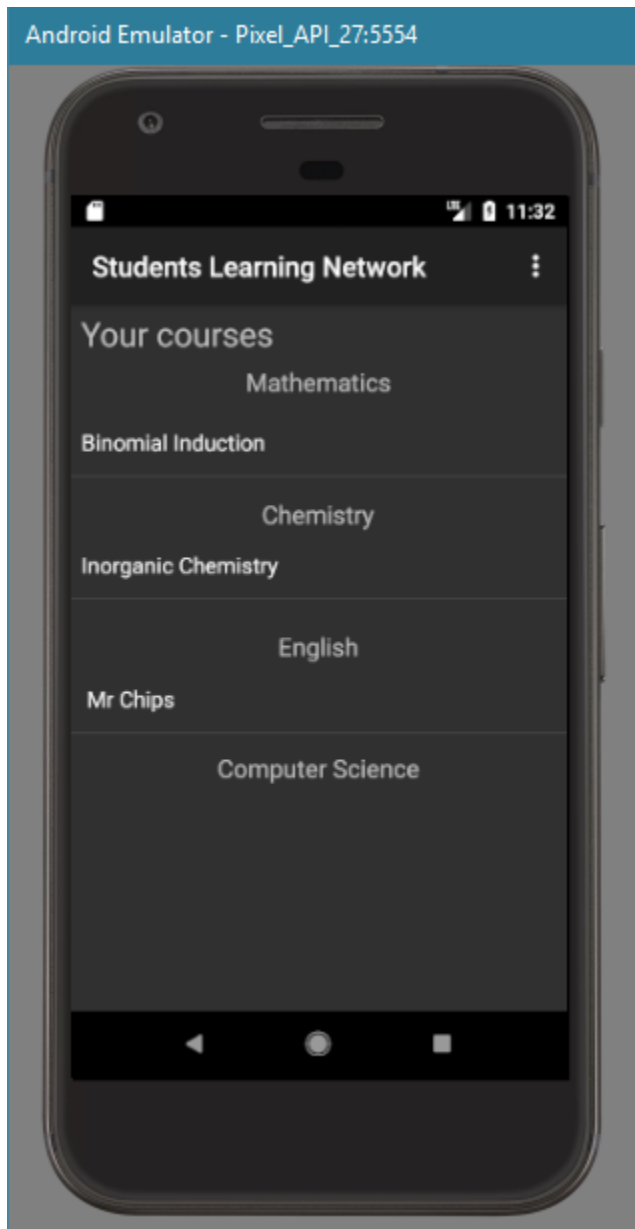
Student checking his Newsfeed:

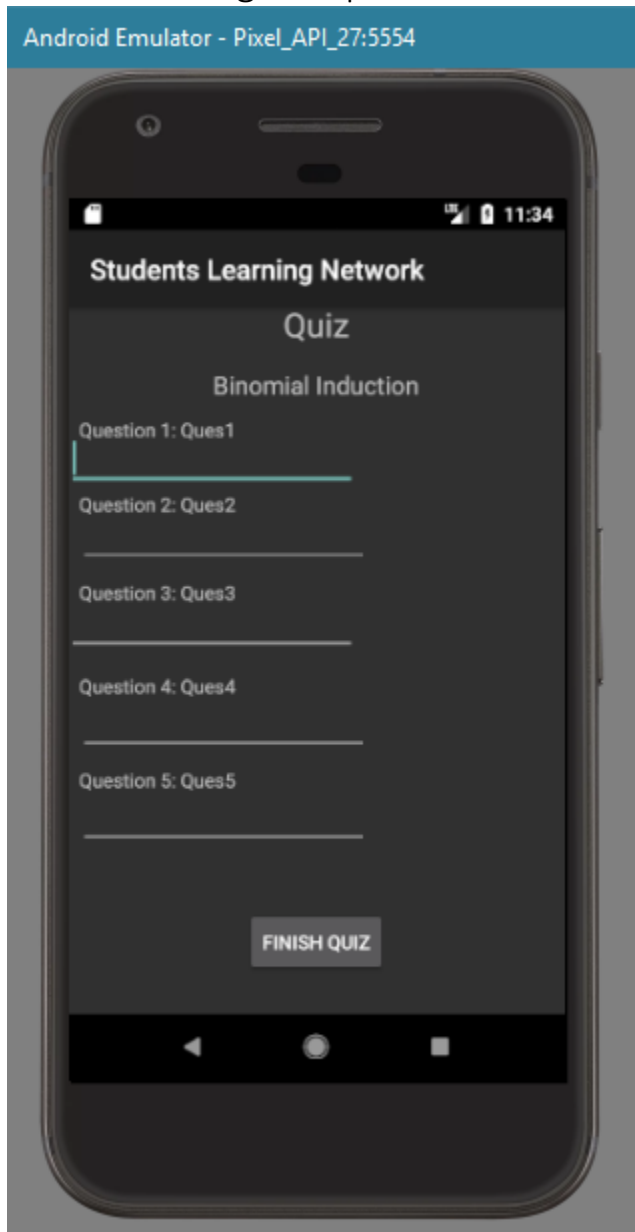## Student checking out his sessions topics:

Student messenger:

Student checking his courses and topic:
Can click on any topic to take a quiz.

Student taking the quiz:

Appendix C: To Be Determined List

Following is a to be determined list for our project

- Adding multiple students in a single session.
- Adding different types of quizzes for the instructor.
- Adding a time limit for the student to take a quiz.

7 CONCLUSION

In Conclusion of this document, we have defined how SLS project should operate in detail. These are the first plans about the project and during the implementation process; some minor details may be changed or added. These changes will be shown in updated documents.