Development Of A Quiz Taker Using Software Engineering Tool

Supervisor

Professor Mohammad Zahidur Rahman
Chairperson
Department of Computer Science and Engineering
BRAC University

Members
Amit Kumar Dey (09101016)
Moloy Ranjan Biswas (09201001)
A.M.M. Shahriar Kabir (14341003)

School of Engineering And Computer Science
Department of Computer Science
BRAC UNIVERSITY

Declaration

This is to certify that this thesis report entitled "Development Of A Quiz Taker Using Software Engineering Tool" submitted by the mentioned authors for the degree of Bachelor of Science in Computer Science and Engineering to the Department of Computer Science and Engineering, School of Engineering and Computer Science, BRAC University. The contents of this thesis have not been submitted to any other institute or university. We hereby declare that this thesis is based on the results found by ourselves and materials of work found by other researcher mentioned by reference.

Signature of authors:

_____	_____	_____
Amit Kumar Dey	Moloy Ranjan Biswas	A.M.M. Shahriar Kabir

Date :

Signature of Supervisor :

_____
Professor Mohammad Zahidur Rahman

Acknowledgement

We are very grateful to Professor Mohammad Zahidur Rahman sir for encouragement, guidance and support from the beginning to end in spite of his very busy schedule and it is an honor for us to thank those who made this paper possible.

This paper would not be possible without help from fellow BRAC University students and extensive research in internet. We are very thankful to all our faculties, friends and well-wishers for their constant support specially Annajiat Alim Rasel sir for his support.

Last but not at least thanks to the Almighty for giving us the opportunity and helping us throughout the work.

Abstract

The primitive way of teaching in school and universities are still in progress in most of the institution. With the continuing evolution of modern handheld mobile devices this project is designed to enhance learning environment. It is observed that students are inattentive during class lectures and hence it degrades their performance. In a classroom of twenty to forty five students it is impractical to ask questions individually and deduce analysis of that student due to time constraint. So with the help of handheld devices a system must be developed which can take quizzes and evaluate it instantly. To develop this type of system our contribution is to use software engineering tools which may become a model for large scale application.

Table of Contents

Chapter 1: Introduction

Software engineering is the study and application of engineering to the design, development, and maintenance of software. It is the ideology behind developing software which meets user's requirements. In software engineering many approaches like waterfall, agile, prototyping etc. are applied for achieving a system that removes human involvement in most situations.

In Software engineering, Software Development is a complex and often a difficult process. From modeling and design to code generation, project management, testing, deployment, change management and beyond, a software engineering tool like readyset.tigris.org helps to guide the process and manage that complexity.

Software development lifecycle is a process of planning, creating, testing and implementing a system that reduces the human related tasks. It involves different work phases: analysis, design, building, testing, implement and maintenance of the new system. The whole work is divided into deadlines and is met according to timelines.

Classroom Environment is an idea where a teacher influences students to learn and develop their growth of mind and intellectual. The learning environment inside a classroom is the fundamental requirement of an education system. Students are influenced by the activities that are performed inside the class. To ensure an active participation of students a classroom have characteristics based on teaching style, level of study and the following[1]:

- Inquiry based learning
- Peer to Peer
- eLearning

This list does not end since there can be numerous alternative methods for making the student understand what are taught inside the class.

1.1 Literature Review

The emergence and prospect of "technology enhanced classrooms" has brought a new dimension in teaching style, understandability, perception of students. It enriches the classroom with new ideas which eases the human perception of knowledge. The use of modern handheld devices in people's everyday life has become an important part. It is estimated that use of handheld devices reached 6.66 billion in 2013 compared to 5.86 billion in 2011 according to International Telecommunication Union[2]. So there is an opportunity to further exploit the use of these devices with an integration of education system.

Many researchers worked with the cognitive approach of students' understanding in classrooms but few worked with the advancement of learning environment with the help of contemporary devices. An Interactive Learning Network (ILN) system was made in Cañada College by Dr. Enriquez where students' progress was monitored and instant feedback, assessment was given using Tablet PCs and a software application[3]. This resulted a significant change in students' performance based on their final grades.

With a vast majority of student having access to handheld devices[4] which comprises with good hardware making it more effective in many scenarios. So this devices can be utilized in teaching environments where greater performance, results can be achieved with proper monitoring and maintenance.

1.2 Motivation

While visiting several schools we noticed that students were almost inattentive due to lack of communication, interest in lectures. Since we are still students, while attending classes we feel the notion that education system is still primitive in remote schools, college and universities especially in developing countries. This method of teaching is still in progress which includes writing down lectures from the board, appearing quiz in an answer sheet, taking attendance manually.

We decided to introduce a system to enhance a part of learning environment. Teachers will feel free from the burden of managing course administrations. Students receive instant update, quizzes and results. Moreover students can actively participate in the class and improve their performance.

Chapter 2: Project Plan

This chapter focuses on the summary of the project where the reader will find the brief plan, the method going to use, the work plan, deliverables and its time and risk management of the project.

2.1 Survey

Before project planning we posted a set of questions online[5] targeting students who studies in different levels. There were 61 responses. Below are the results.

Q.1) Do you think current or old fashioned classroom environment is effective for you being a student?
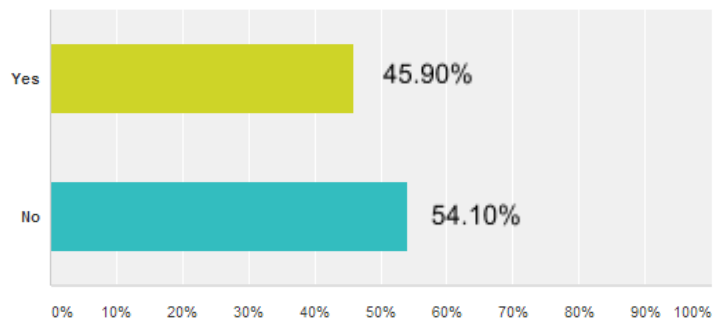
Q.2) Do you want a system that can enhance student's performance inside the classroom?
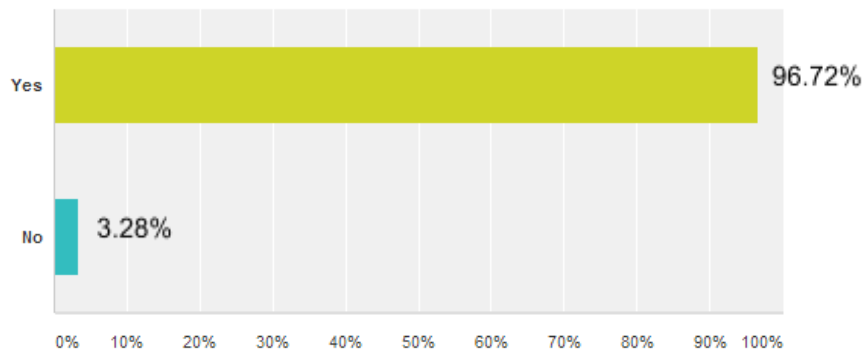
Q.3) If handheld devices were used for educational purpose, which device is comfortable for you inside the classroom?

Mobile Phones 42.62%
Tablets 55.74%

0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
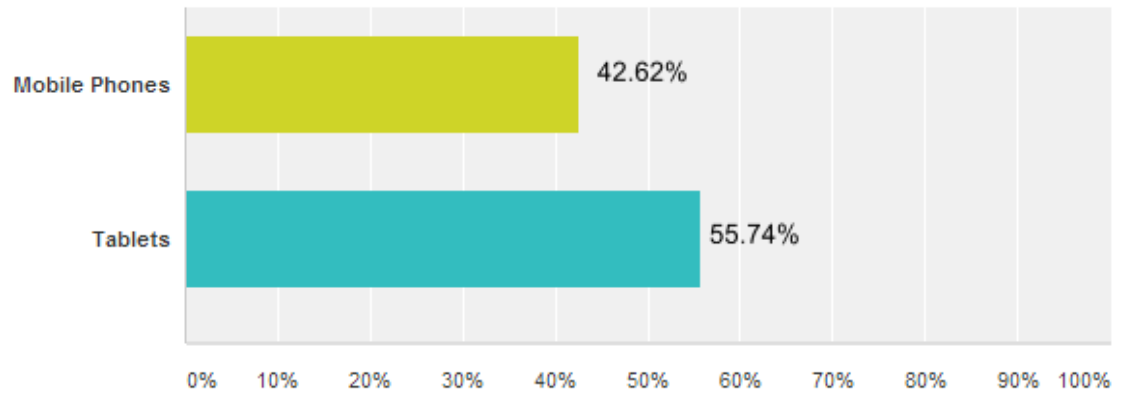
Q.4) Which mobile operating system you prefer most while using the device inside the classroom?

Android 77.05%
iOS 6.56%
Windows Mobile 16.39%

0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
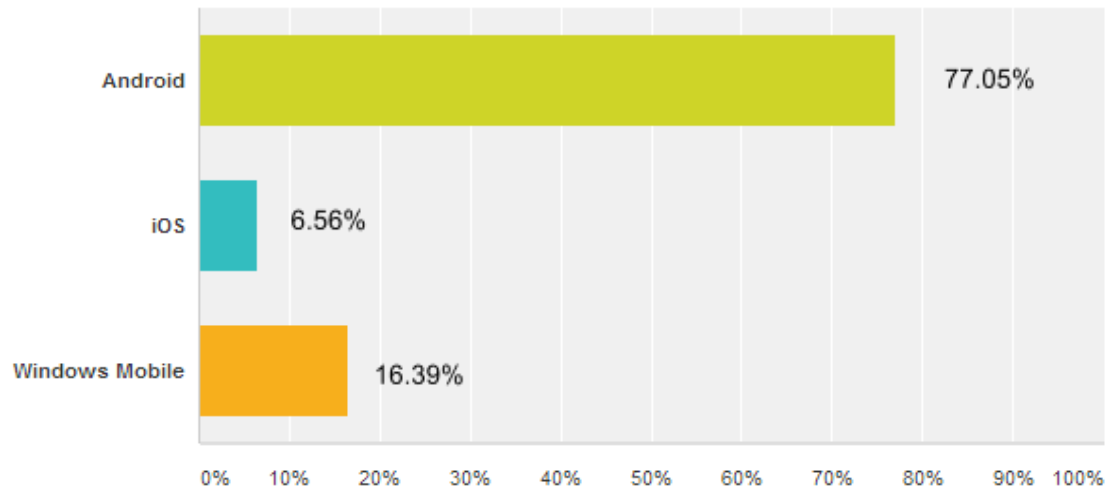
Q.5) Do you think using handheld devices will improve teacher-student communication inside the classroom for giving quiz, notes, lectures etc. ?
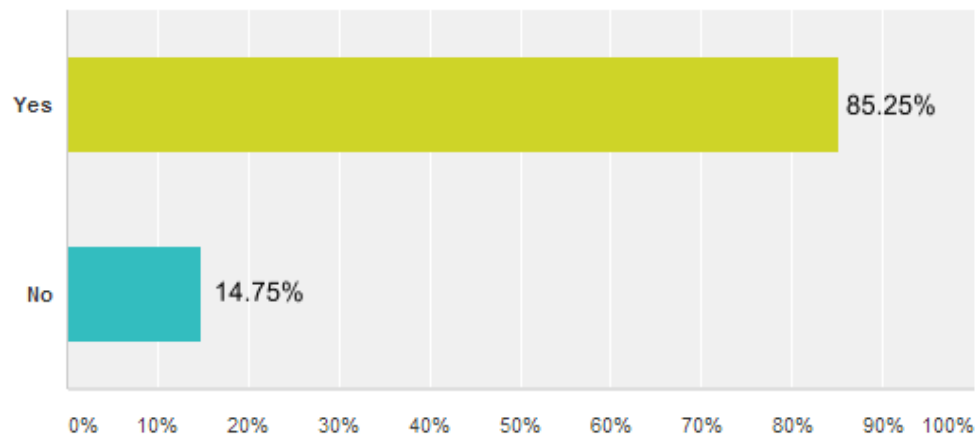
The results shows students want an environment designed for them which can enrich their capabilities to perform better in classrooms. They are more comfortable in tablets using Android operating system.

2.2 Summary of Project

This project is designed to enhance student's communication in a classroom environment using cutting-edge technology. Students and teachers can interact efficiently in the classroom by using software designed for them.

- Dynamic interface for the students will be designed for their easy access to course materials while being present in the class lectures.
- Real time pop quiz component and result analysis based on the students' performance.

2.3 Summary of Methodology

In this project prototyping methodology is going to be used. We will design a prototype and then modify it according to its requirement. Since advancement in technology is swelling rapidly, the system needs modifications meeting needs of its users.

All three members are in the development team and will collaborate in building the system. We would meet once in a week for discussion and review of previous tasks. One of us will act as a project manager and developer throughout the project.

Throughout the project we are going to use the following tools

- Google Drive: We will use cloud base service Google drive for sharing our idea with others and get their feedback. Important updates/releases will be notified there.
- Pair testing/ buddy testing: We will be doing pair testing method for each of the software feature through a quick sanity check of Quality Assurance.

An issue tracking system will be used where request for changes will be monitored and meet the requirements on time. After this issue has been solved, it will be saved with a specific ID and no more changes will be made on that version.

The plan will be updated as needed throughout the work. All the updates, instruction for accessing different versions of this project will be enlisted in the project website. Important issues will be discussed through the website which can lead to useful feedback from other experts.

2.4 Work Breakdown Structure and Estimates

Task	Description	Estimate
1.	Preparation	
1.1.	Developer Training	72h
2.	Inception	
2.1.	Requirements Gathering	30h
2.2.	Requirements Specification	20h
2.3.	Requirements Validation	10h
3.	Elaboration	
3.1.	High-Level Design	10h
3.2.	Low-Level Design	
3.2.A.	Interface Design	10h
3.2.B.	Database Design	10h
3.3.	Design Review and Evaluation	5h
4.	Construction	
4.1.A.	System Implementation	
4.1.A.1.	Implement Dynamic Interface	48h
4.1.A.2.	Implement Auto Presence	72h
4.1.A.3.	Implement Lecture Video Recording	48h
4.1.A.4.	Implement Centralized Server	48h
4.1.A.5.	Integrate Components	10h
4.1.B.	Technical Documentation	15h
4.1.C.	User Documentation	15h
4.1.D.	Testing	
4.1.D.1.	Test Planning	10h
4.1.D.2.	Test Code Implementation	30h
4.1.D.3.	Test Execution	10h
4.2.	Implementation Review and Evaluation	15h
5.	Transition	
5.A.	Release Packaging	3h
5.B.	Documents for other groups	3h
6.	Reflection	
6.1.	Postmortem Report	10h
	Total:	**504h**

2.5 Deliverables in this Release

Deliverable Name	Description	Delivery Date
Prototype 1	This will consist of the interface, connectivity with server, manipulating materials in the app.	15th April 2014
Final Version	Full functional system including pop quiz component and result analysis.	15th August 2014

2.6 Schedules for this Release

Task\Week	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	Task Total
1.	72h												01
2.		60h											03
3.			35h										04
4.1.A.1.				48h									01
4.1.A.2.					72h								01
4.1.A.3.						48h							01
4.1.A.4.							48h						01
4.1.A.5.							10h						01
4.1.B.							15h						01
4.1.C.							15h						01
4.1.D.1.								10h					01
4.1.D.2.									30h				01
4.1.D.3.										10h			01
4.2.										15h			01
5.											6h		02
6.												10h	01
Weekly Totals	72h	60h	35h	48h	72h	48h	88h	10h	30h	25h	6h	10h	

2.7 Risk Management

Following are the main risks of this project:

● One of the main risks is the affordability of gadgets available for good performance. Students will find difficult to buy and use these high tech product. We will plan to contact with universities, schools for mass order of these gadgets so that price becomes cheaper.

● The adaptability of this application is an issue here, both teachers and students might find difficult to use it at first glance. We would then organize a training session for them to overcome the problems.

● There is a significant difficulty in building a cloud server which maintains course materials. We have to learn and then implement as we are naive in this technology. We would contact with an expert in this field and seek help.

● There is an issue of finding the correct algorithm for face detection and meet our demands. Otherwise we would go for the most used tools for face detection.

● The quality of this product might be useful in Western regions but will lack behind in this Asian parts. Further updates will be made to meet this demand.

2.8 Project Planning Dependencies

We are working only on this project for time being, standalone project and does not depend on any other projects or vice versa. This will be the first release and we plan to release further updates if we find good offer and sufficient funds. Component of this project will not be used in any other projects. Few components will depend on availability of cloud systems.

Chapter 3: Software Requirements Specification

A use case diagram is a representation of a user's interaction with the system. It represent different types of users of a system and the various ways that they interact with the system. A functional requirement is the function of a system and its components. It defines a function as a set of inputs, the behavior, and outputs. On the other hand, non-functional requirement specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. This chapter includes the use cases, functional, non-functional requirements.

3.1 Use Cases

Students and instructors will use this software for participating in the quizzes from the server. On the other hand, instructors will use this software to teach students efficiently. They can arrange quiz very quickly and tracking the progress of the students will be very easy along with their grades. The most important part about this software is the instructors have a quick insight of the class and students can track their progress instantly. Thus the software will provide an opportunity for the educational institution to create a friendly educational environment for students and instructors.

3.2 Functional Requirements of the Software

Our system will provide some features which will ensure better educational environment. The system will work on mobile phone and tablet computers. Central server will ensure data storage, collection, manipulation and retrieval. We will develop following features for the software.

3.2.1 Features

- F-00: Dynamic Interface: This is the main interface where users will interact with the system, basically it will contain three parts- course, quiz, performance. Users will navigate through these parts.

- F-01: Server Connectivity: This server will hold all databases. It will also act as web server.

- F-02: Quiz Polling: In this feature instructors will set questions. He/she then approves the quiz just before it starts and then students' starts answering. Marks can be shown instantly. It will contain only multiple choice questions types.

3.3 Non-Functional Requirements

The user interface should be as familiar as possible to users who have used other handheld devices and desktop PCs e.g., we will follow UI guidelines for naming menus, buttons, and dialog boxes whenever possible. Moreover all the features will be as simple as possible for the users to understand and play around with.

For reliability of this software we will be using prototyping methodology. We will release prototype version every four month and we will randomly pick some students and instructors for testing this prototype. After every three month we will review our work for consistence. Every one of our group member will specially keep track of our developing along with the user specification in every month. For testing purpose we will select some students, instructors and a group of people representing educational institution. They will use the software and we will note down their feedback for better improvement.

The up time of this software depends upon proper requirement gathering and analysis and spends more time for constructing this software. As we are spending three months for gathering system requirements and doing analysis with our project supervisor it will be no problem. We are also doing the documentation part also which will be very handy for that software.

Access of the instructors and students will be controlled with unique usernames and passwords. Administrator will create the faculty's and student's account. Only administrator users will have access to administrative options, functions. Password of the users must be 6-14 characters long which must have at least one or more alphabetical letters. We will use encrypted password for connecting with the server for checking usernames and password.

Maintainability is our ability to make changes to the system over time. We need strong maintainability in order to retain our early users. We will address this by anticipating several types of change, and by carefully documenting our design

and implementation. Then we will ensure these new user needs in the software by releasing a new upgrade version.

Upgradability is our ability to cost-effectively deploy new versions of the product to customers with minimal downtime or disruption. A key feature supporting this goal is automatic download of patches and upgrade of the end-user's machine. But user also can choose whether he/she should upgrade or not.

Supportability is our ability to provide cost effective technical support. The product's automatic upgrade feature will help us easily deploy defect fixes to end-users. The user guide/manual will include a troubleshooting guide to have at hand before contacting technical support.

3.4 Environmental Requirements

For operating this software we need some system hardware. For example, we will have primary hardware in which the software will run smoothly. There are some other hardware issues for maintaining this software in the back end by the system administrators. A server is needed for saving the records of the users securely. Some personal computers are needed for the administrators for maintenance issues.

For developing this system we will few softwares. For writing code we will use Net Beans, Eclipse depending upon the personal choice of the developer. For database design we will use MySQL Workbench. For documentation we will use ReadySet. We, the developers will be working remotely, we will be using GitHub and Google Drive for sharing our work with each other very efficiently.

Our software will use a secure database for storing the records of the users (both instructors and students). The system will store all data in a standard SQL database. We can easily import or export files from the database.

3.5 Feature Set:

A feature set outlines what an application does, how users behave with the system.

3.6 All feature set are in Appendix A, page-40.

- ● Quiz Management and Grading
    - ▪ UC-31 Create Quiz
    - ▪ UC-32 Send quiz
    - ▪ UC-33 Receive quiz
    - ▪ UC-34 Auto grade quiz
    - ▪ UC-35 Send quiz report to Instructors
    - ▪ UC-36 Create a graph based on the quiz performance
    - ▪ UC-37 Generate a student performance chart

- ● Course Management
    - ▪ UC-41 Access course materials
    - ▪ UC-42 Create category of courses

3.7 Use cases are shown in Appendix B, page- 42

Chapter 4: Design

4.1 Introduction

This main page describes the system design in terms of packages, classes, relationships, and behavior. Several attached worksheets address specific aspects of the overall system design, such as user interface and database design.

The design is done in object-oriented approach with an idea of system specification. The developer should know the scope and different diagrams of this application. He should also know the risks and time allocation of this project.

4.2 Prioritized goals of this design

1. **Correctness:** This design should match the given system requirements.

2. **Feasibility:** This design can be implemented and tested with the planned amount of time and effort. The web application will be tested with proper amount of time.

3. **Understandability:** The application should be properly done and easily understandable by any developer.

4. **Implementation phase guidance:** This design divides the implementation into some small components or aspects that can correspond to reasonable implementation tasks.

5. **Modularity:** Concerns are clearly separated so that the impact of most

design changes would be limited to only one or a few modules.

6. **Extensibility:** New features or components can be easily added later. This means that we can add more features and also increase the scope of this web application.

7. **Testability:** It is easy to test components of this design independently, and information is available to help diagnose defects.

8. **Efficiency:** The design enables the system to perform functions with an acceptable amount of time, storage space, bandwidth, and other resources. So efficiency is considered while developing this web application.

9. **Ease of integration:** The components will work together as we develop some small features and then integrate the whole in a single web application.

10. **Capacity matching:** The architecture deploys components onto machines that provide needed resources with reasonable total expense.

11. **Expressiveness:** It allows for storage of all valid values and relationships.

12. **Ease of access:** Application code to access stored data is simple.

13. **Reliability:** Stored data cannot easily be corrupted by defective code, concurrent access, or unexpected process termination. So the whole

system code along with the web application is reliable.

14. **Data capacity:** The system can store the amount of data needed, which is mentioned in the system specification. Moreover we can also increase the data capacity as needed later on.

15. **Performance:** Data can be accessed quickly and efficiently.

16. **Interoperability:** The database or data files can be accessed and updated by other applications as needed.

4.3 UML Structural Design

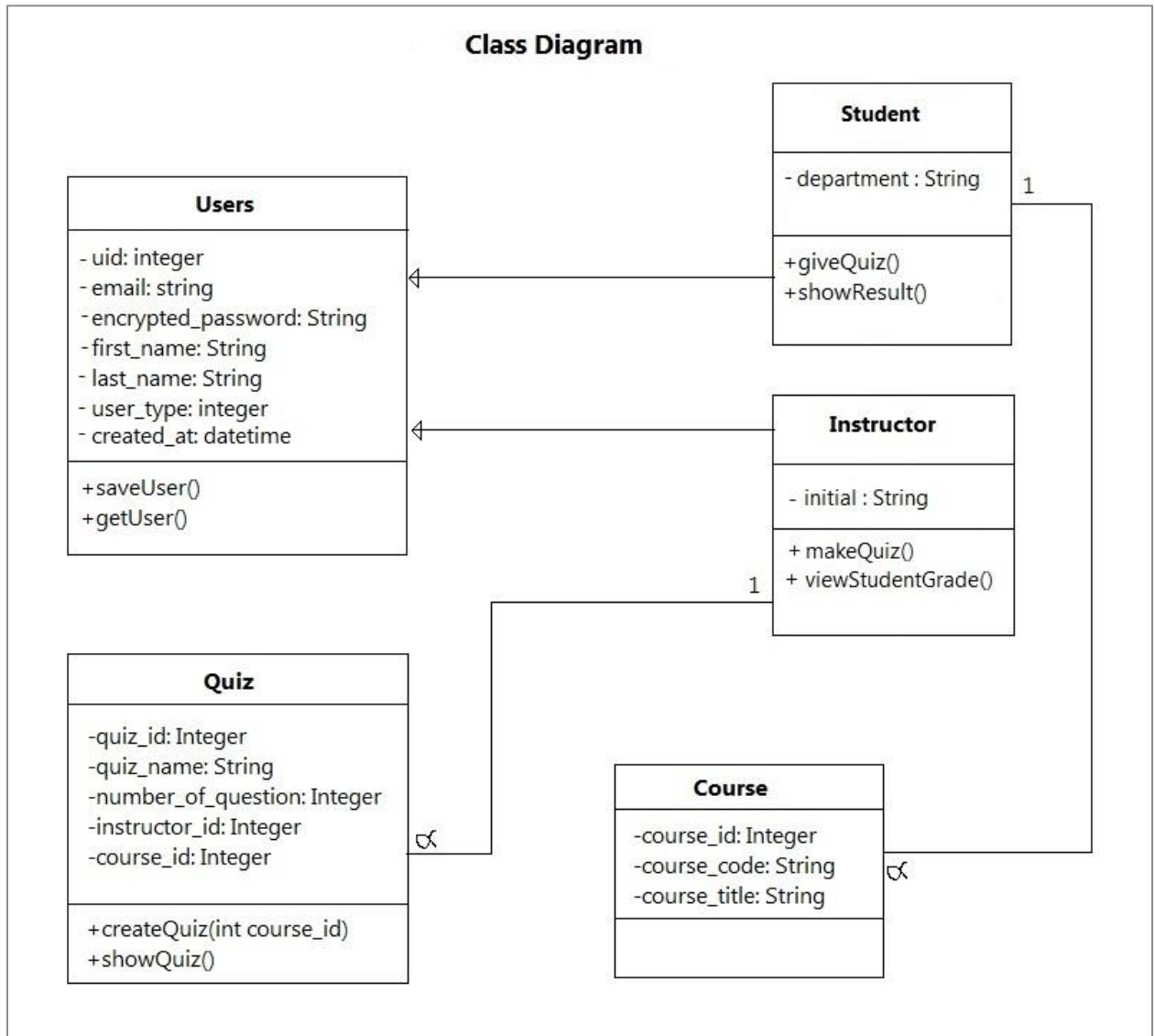**Class Diagram:** The system's class diagram is described in the following UML model.

Fig. 4.3.1 Class Diagram

Object Diagram: The following object diagram shows when a student is enrolled in a course and gives three quizzes.
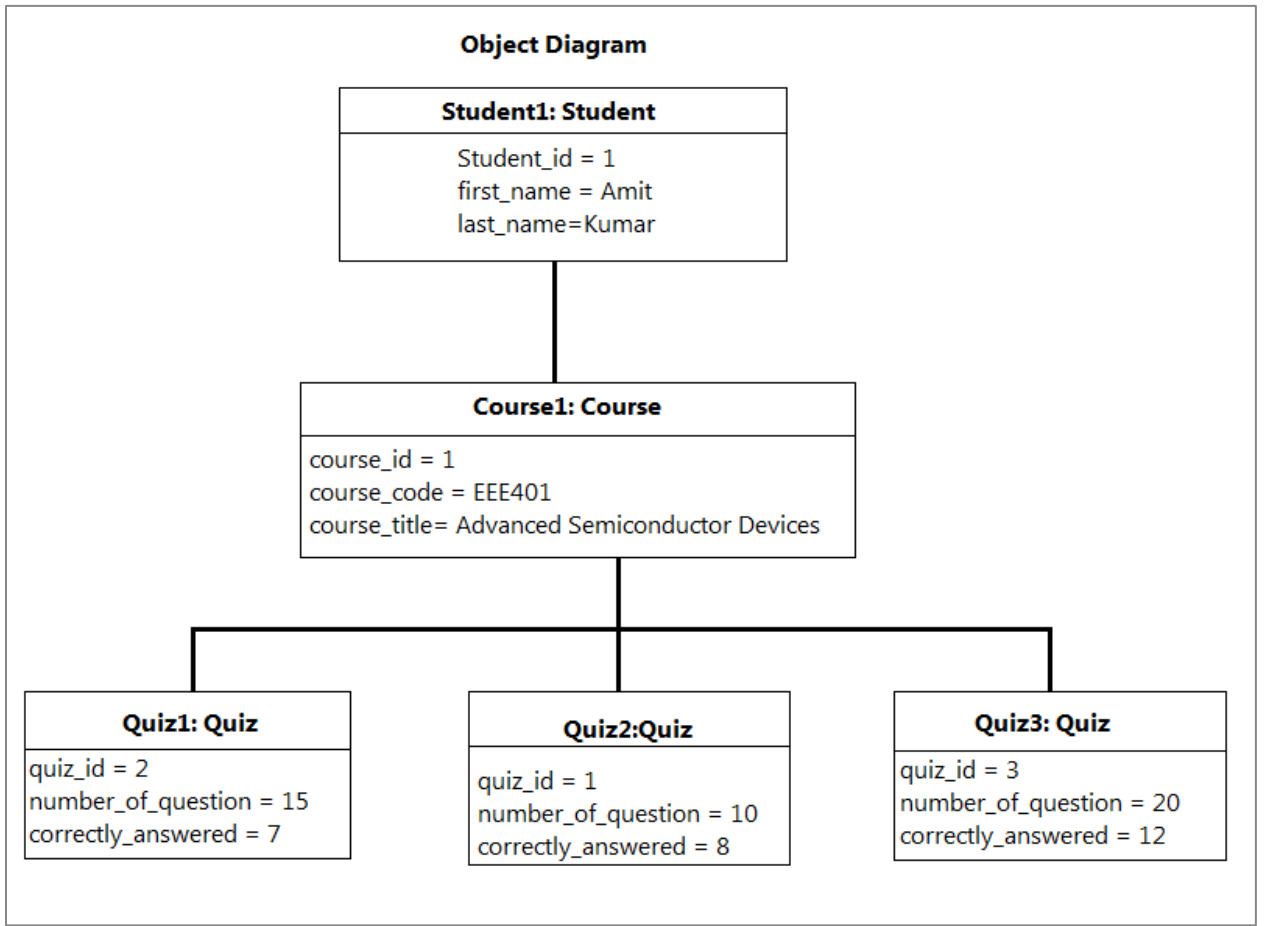
Object Diagram

Student1: Student
Student_id = 1
first_name = Amit
last_name=Kumar

Course1: Course
course_id = 1
course_code = EEE401
course_title= Advanced Semiconductor Devices

Quiz1: Quiz
quiz_id = 2
number_of_question = 15
correctly_answered = 7

Quiz2:Quiz
quiz_id = 1
number_of_question = 10
correctly_answered = 8

Quiz3: Quiz
quiz_id = 3
number_of_question = 20
correctly_answered = 12

Fig. 4.3.2 Object Diagram

4.4 UML Behavioral Design

Sequence Diagram: The following sequence diagram shows when a student logs in and give quiz.
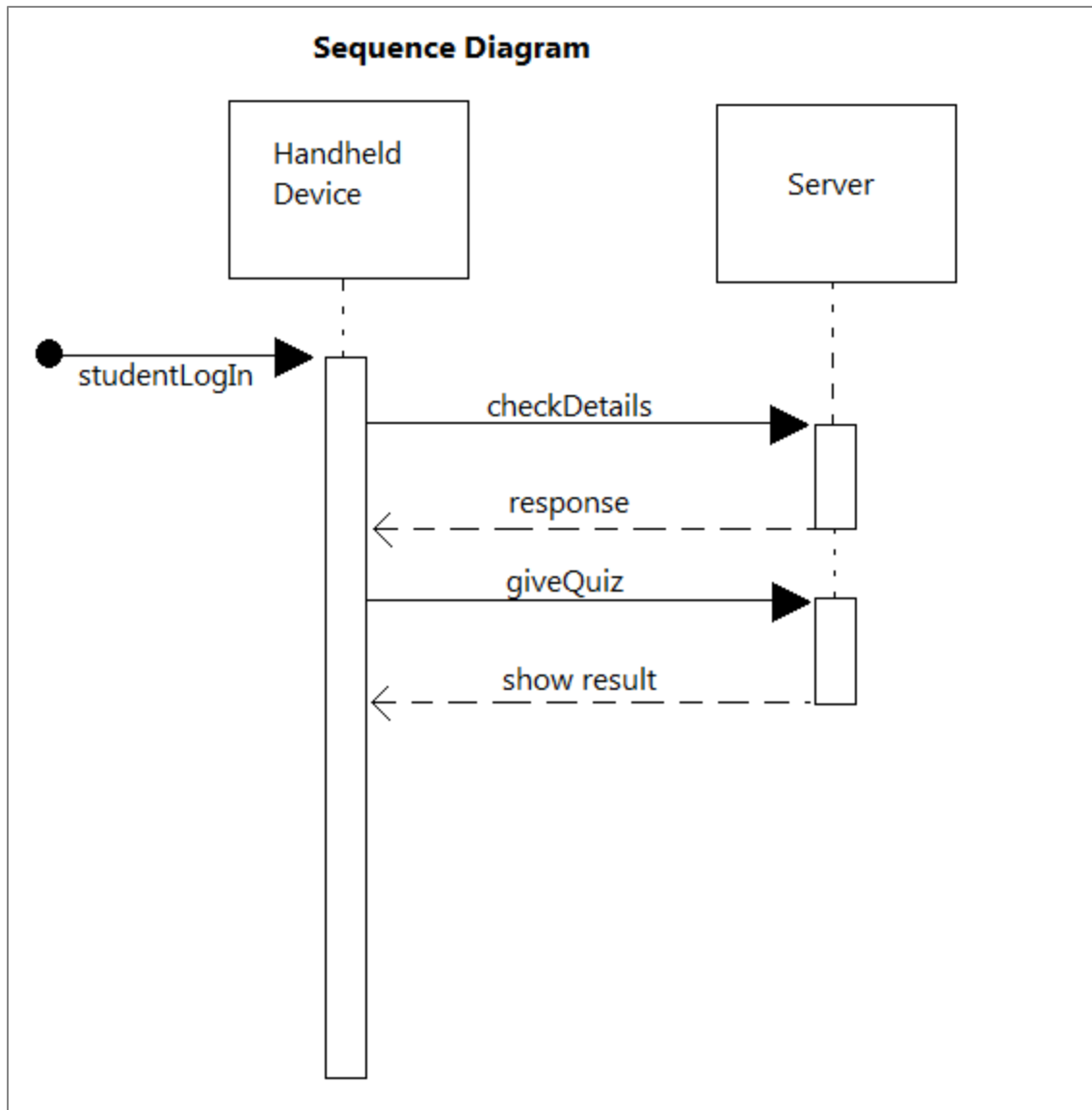

Fig. 4.4.1 Sequence Diagram

Activity Diagram: The following activity diagram shows when an instructor makes quiz and student logs in to give the quiz.
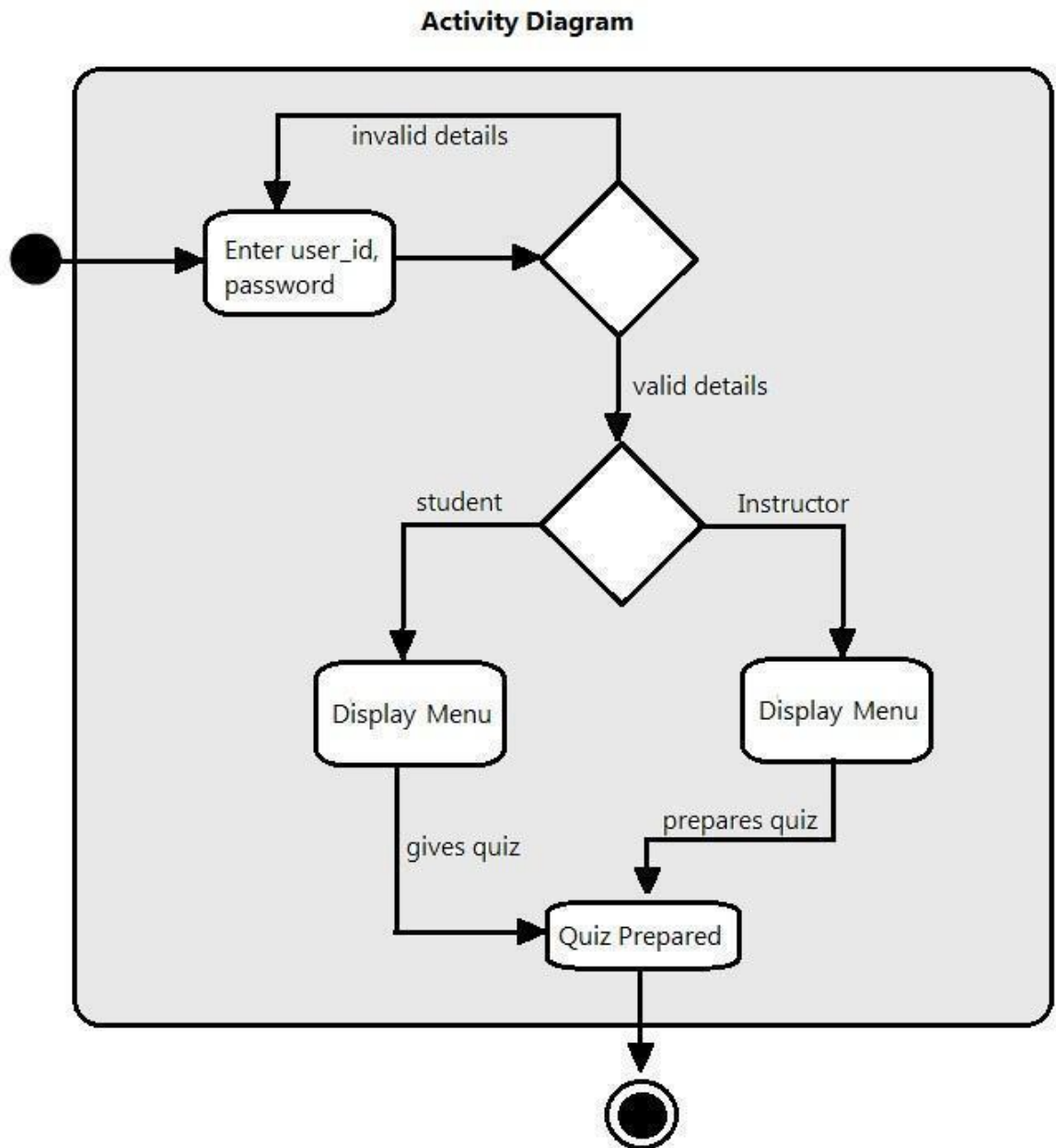

Fig. 4.4.2 Activity Diagram

4.5 Database Design

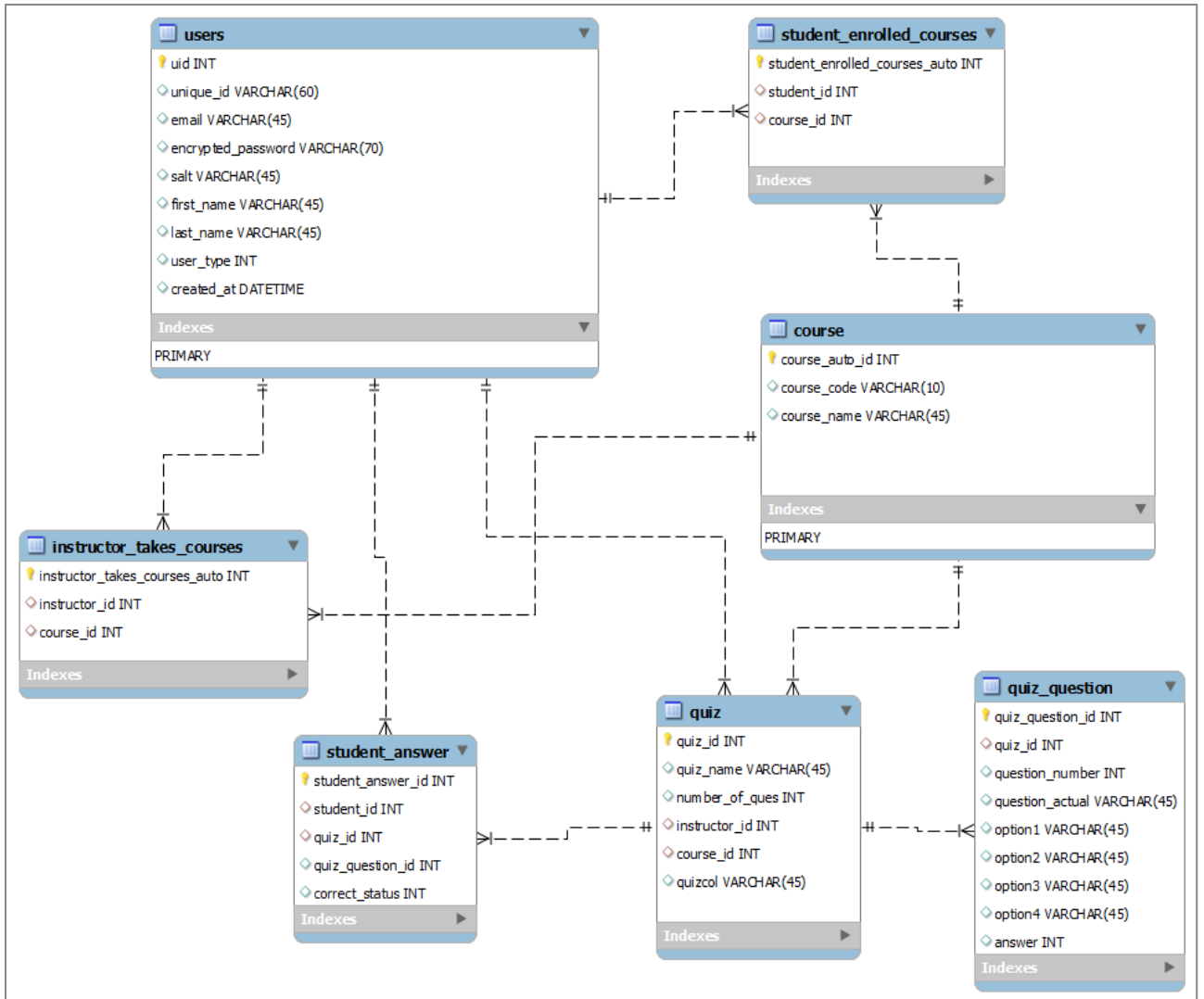The following figure shows our database design.

Fig. 4.5.1 Database Design

4.6 Project resource needs

Human Resource Needs

Need	Resource	Amount	Status	Comments/Responsibilities
Project Management	A.K. Dey, M.S. Kabir	20 hours	Pending	Follow up the process regularly.
Requirements	M.R. Biswas, M.S.Kabir, A.K. Dey	20 hours	Assigned	Collect Requirements from Student experience
Requirements	M.R. Biswas, M.S.Kabir	10 hours	Assigned	Collect Requirements from Faculty experience
Consultation with domain expert	M.S.Kabir, A.K. Dey	2 hours	Assigned	Consult with Faculty
Consultation with topic expert	A.K. Dey	4 hours	Assigned	Consult with Faculty and other researcher of the same topic
Training on use of component/technology	M.R. Biswas, M.S.Kabir, A.K. Dey	8 hours	Assigned	Video tutorial from Coursera and MiT edX
Overall Design	M.R. Biswas, M.S.Kabir,	A.K. Dey	20 hours	Assigned
Detailed UI Design	M.R. Biswas, M.S.Kabir,	5 hours	Assigned	After design review it to any UI expert.
Detailed Database Design	M.S.Kabir, A.K. Dey	5 hours	Assigned	After designing the DB review it to any DB expert.
Development	M.R. Biswas, M.S.Kabir, A.K. Dey	40 hours	Assigned	
Development	M.R. Biswas, M.S.Kabir, A.K. Dey	80 hours	Assigned	
Development	M.S. Kabir	20 hours	Assigned	
Technical writing	M.R. Biswas, M.S.Kabir, A.K. Dey	10 hours	Assigned	Take http://readyset.stage.tigris.org/ as a reference or guideline of the writing

Capital Needs

Need	Resource	Amount	Status	Comments
Training materials	Book/Course on specific technology	1	Allocated	Book ordered, Video Tutorial Collected
Development Workstations	Core i3, 2GB RAM	3	Satisfied	Dev group will use existing workstations
Development DB Server	Dual CPU 1GHz PC, 512MB RAM	1	Allocated	
Interactive Testing Workstations	Quad Core 1.2 GHz Cortex-A9, Android OS, v4.1 (Jelly Bean), 1 GB Ram	3	Allocated	
Testing DB Server	Dual CPU 1GHz PC, 512MB RAM	1	Rejected	Testing group will use development DB server
IDE Licenses	Standard development licenses	N/A	Satisfied	Will use open source tools
Testing Tool Licenses	Standard development licenses	N/A	Satisfied	Will use open source tools
DB Licenses	Standard development licenses	N/A	Satisfied	Will use open source BD
Development Tools	Android Developer Tools, Build v22, Eclipse IDE, NetBeans IDE 8.0, JRE 7, Notepad++	N/A	Satisfied	

Minimum Requirements for the system

- ▪ Hardware:
  - ● Processor: 2.13GHz
  - ● RAM: 2GB
  - ● Device which supports Android(min 4.1 JellyBean) with 1GB of RAM

4.7 Implemented Codes

Few implemented codes are in Appendix B, page 42

QA Plan

The term "Quality" refers to all the good things that we would like to see in our product. We build a quality product and assure its quality by keeping quality in mind all the time and performing the selected activities below. Testing is one QA activity, but it is not the best or only one, other QA activities include the use of style guides and checklists, review meetings, use of analysis tools, and careful quality measurements and estimates. A plan is needed to select and coordinate all the QA activities.

This is the first release. So after launching this release we are planning to test it and after the testing sessions we will decide to improve it in accordance to the needs of the valuable clients.

All components and aspects of the system will be evaluated in this release.
There are many quality goals and approaches to assuring them. Since we have limited time and resources for this release, we will focus on the following components and aspects:

- Uploading quiz by Instructor
- Quiz given by Students
- Show performance based on results

Quality Goals for this Release

- Essential:
    - Functionality > Correctness: For our application, the system will always provide correct output for specific sets of input from the users.

- Expected:
  - ▪ Functionality > Accuracy: In our application, the system always provides accurate result based on the given inputs. In our case the system gives accurate result of the quiz result obtained by the students. Another fact is, faculty can accurately set the quiz paper and the quiz answers for the students.

  - ▪ Functionality > Compatibility: Our application will render in most recent android versions. More specifically Android Jelly Bean 4.1 and Android KitKat 4.4.

Chapter 5: User Documentation

Students can answer multiple choice questions which were set by instructors. If you are an Instructor follow topic "5.3 User guide for Instructors" or if you are student follow topic "5.2 User guide for students". This chapter has elaborately shown all the steps from installing to using the application with step by step guideline including snapshots.

5.1 Installation of .apk file

Step 1: Transfer your APK to your Android Device and touch to install the APK into your device (Figure: 5.1.1).

Step 2: The Application is asking for the permission that will be needed to run the app, tap Install to forward (Figure: 5.1.2).

Step 3: This screen ensure the installation process are running smooth. Wait for a few seconds to continue (Figure: 5.1.3).
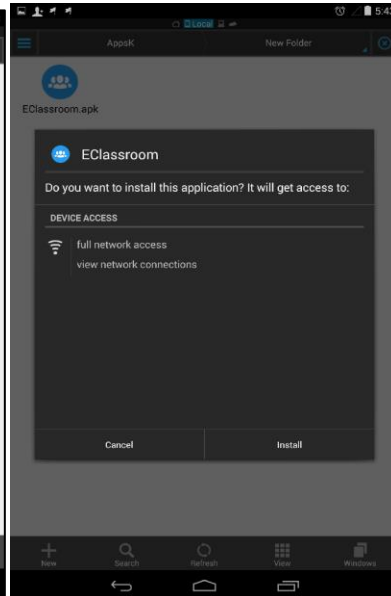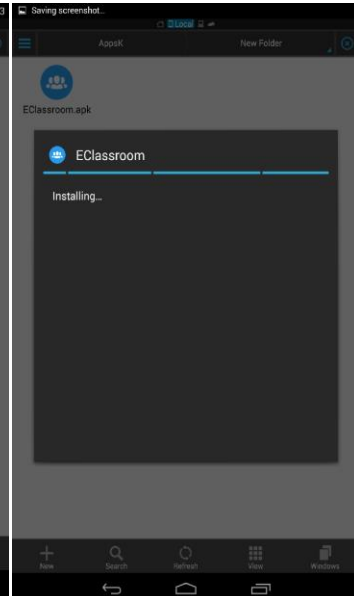

Figure: 5.1.1   Figure: 5.1.2   Figure: 5.1.3

Step 4: When the EClassroom.apk is done with the installation the screen will ask to tap "Done" or "Open". Tap "Done" to end the process here or you may tap "Open" to run the app now (Figure: 5.1.4).

5.2 User guide for students:

Step 1: After running the app enter your credentials and press "Login" (Figure: 5.2.1).

Step 2: After logging in, you will find three tabs for navigation, you have to touch the tabs or swipe right/left for navigation.

 "Course" tab will show what courses are you enlisted, "Quiz" tab will show quiz questions, and "Performance" tab will show results of quizzes (Figure: 5.2.2).
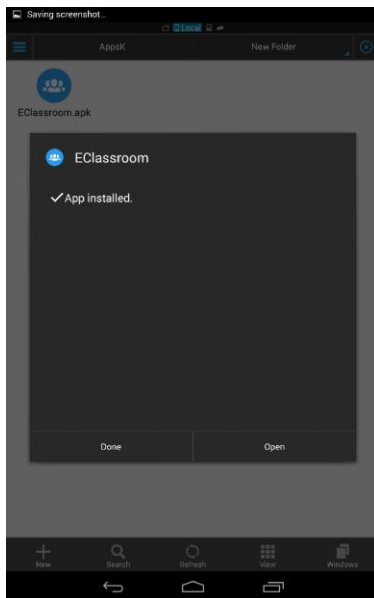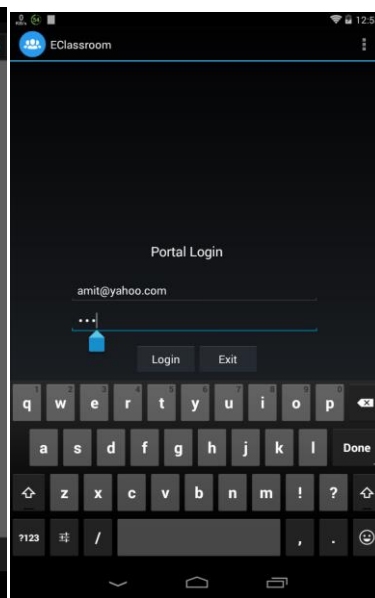

Figure: 5.1.4   Figure: 5.2.1   Figure: 5.2.2
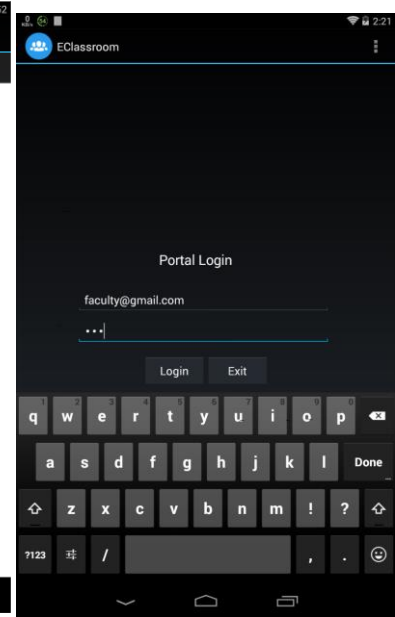
| Figure: 5.2.3 | Figure: 5.2.4 | Figure: 5.3.1 |

Step 3: You have to select correct answer for each question and press "Submit" button when finished (Figure: 5.2.3).

Step 4: Select your desired course from drop down list and press "Show" button to see quiz results (Figure: 5.2.4).

5.3 User guide for Instructors:

Step 1: After running the app enter your credentials and press "Login" (Figure: 5.3.1).

Step 2: After logging in, you will find three tabs for navigation, you have to touch the tabs or swipe right/left for navigation.

"Course" tab will show what courses are you take, "Quiz" tab will show to prepare questions, and "Performance" tab will show results of quizzes of students (Figure: 5.3.2).

Step 3: In the "Quiz" tab select the Course Code that you want to prepare quiz for. Then enter "Quiz Name" in the field. Lastly, enter the number of questions. Then press "Next" button(Figure: 5.3.3).

Step 4: Enter questions and options in the appropriate field and press "Submit" button (Figure: 5.3.4).
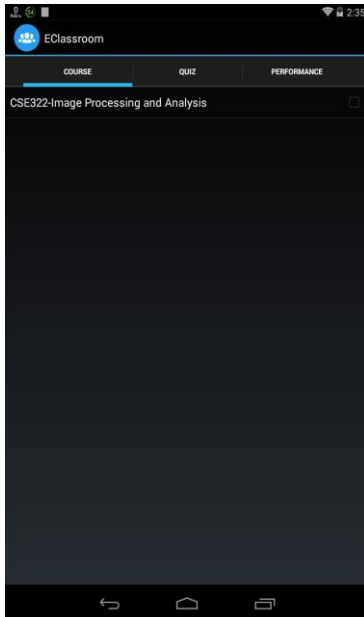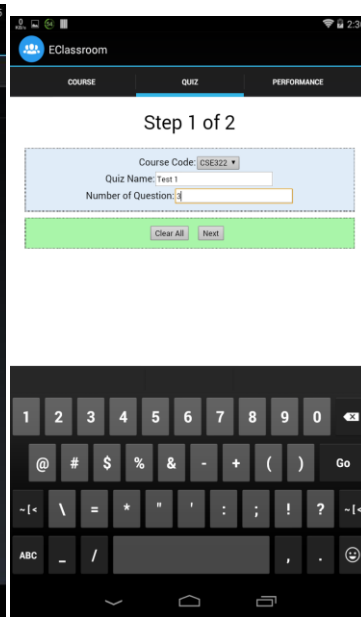

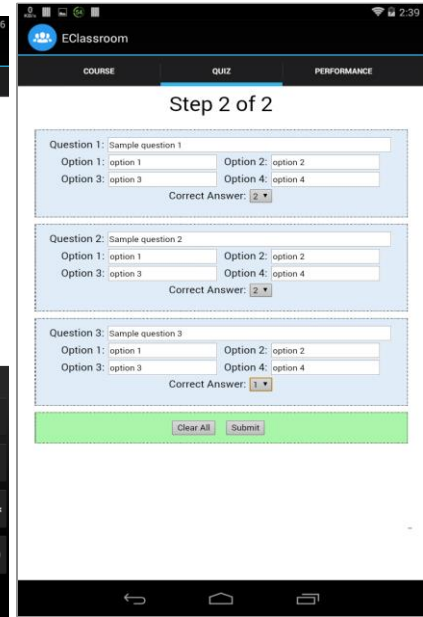Figure: 5.3.2   Figure: 5.3.3   Figure: 5.3.4

Step 5: Enter your desired course and press "Show" button to see performance of students in the quiz.

Figure: 5.3.5

Chapter 6: Conclusion & Future Works

6.1 Conclusion

This project was developed with prototyping methodologies thus we have a quick product which may have some short comings but ready as a beta version.

This project has gone through a software development life cycle; readyset.tigris.org was the guideline for the whole project from planning, work breakdown structure, quality assurance, to release date. This project has six major chapters where each chapter focused on four major components of the software development life cycle. The first chapter gives the idea of the full project, the problem specification, literature review and the motivation behind the project. The second chapter deals with the initial survey report, project planning, methodologies and work breakdown structure. The third chapter gives the total cutinizing on the project. This chapter talks about the functional and nonfunctional requirements, analysis the use cases and feature set for the project. The fourth chapter move forward with the necessary design for the project. Its deals with the UML structure design, object design, UML behavioral design, database design, and activity diagram and quality assurance phase. The fifth chapter emphasizes on technical documentation and the sixth chapter describes the user guide elaborately.

While following the ReadySet, we have been challenged many times by the vision of our project due to plan it properly at the very beginning. We have pivoted our few features keeping the goal constant. We have also gone through some technical challenges while developing the prototype too. One of them was to parsing data from database to Android using XML. We overcome this challenge through an extensive research on the web[9-13] and talking with the field expert.

6.2 Future Works

Since this system only manages grading of students based on quizzes they answer, it can be further updated with more interesting features. The classroom can be made more effective by

- Ubiquitous Computing where students attendance will be taken automatically.
- Recording lectures and making it available for future.
- Performance analysis can be sent to parents/guardians.
- Touch-pen system can be implemented for taking notes, diagrams.

Appendix

Appendix A

3.6.1 Feature Set

F-00: DYNAMIC INTERFACE

Priority: Essential

Effort: 4 Weeks

Risk: Safe

Functional area(s): Quiz Management, Course Management

Use case(s): UC-10, UC-11, UC-12, UC-13, UC-14

Description: This is the main interface where users interact with the system. Basically, it will contain 2 main parts and other 3 sub parts.

Notes and Questions:
- Need to include other sub details
- Should be more informative

F-01: SERVER CONNECTIVITY

Priority: Essential

Effort: 1 Week

Risk: Safe

Functional area(s): Quiz Management, Course Management

Use case(s): Not Applicable

Description: This server will hold all databases. It will also act as web server.

Notes and Questions: Do we need high configuration server?
For cloud services, how we would configure our server?

F-03: QUIZ POLLING

Priority:	Expected
Effort:	1 Week
Risk:	Safe
Functional area(s):	Quiz Management, Course Management
Use case(s):	UC-31, UC-32, UC-33, UC-34, UC-35, UC-36, UC-37
Description:	In this feature instructors will set questions. He/she then approves the quiz just before it starts and then students' starts answering. Marks can be shown instantly. It will contain only multiple choice questions types.
Notes and Questions:	Will it be more tedious for the students if answers are descriptive since typing is a hassle in android devices?

Appendix B

3.7.1 Use Cases

3.7.1.2 Use case diagram for dynamic interface

- ▪ UC-10 Dynamic Interface
- ▪ UC-11 Responsive menu system
- ▪ UC-12 Less lag time between transition
- ▪ UC-13 Improved layout performance
- ▪ UC-14 User friendly

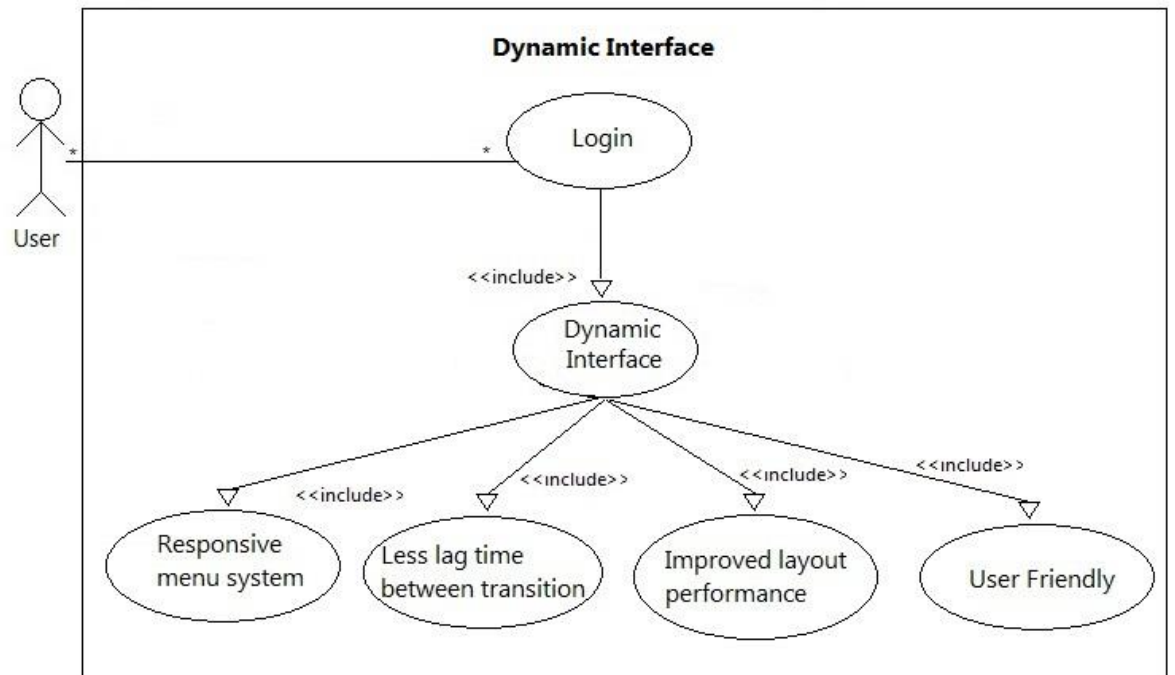Fig 3.7.1.2: Use case diagram for dynamic interface

3.7.1.3 Use case diagram for Quiz Management and Grading

- UC-31 Create Quiz
- UC-32 Send quiz
- UC-33 Receive quiz
- UC-34 Auto grade quiz
- UC-35 Send quiz report to Instructors
- UC-36 Create a graph based on the quiz performance
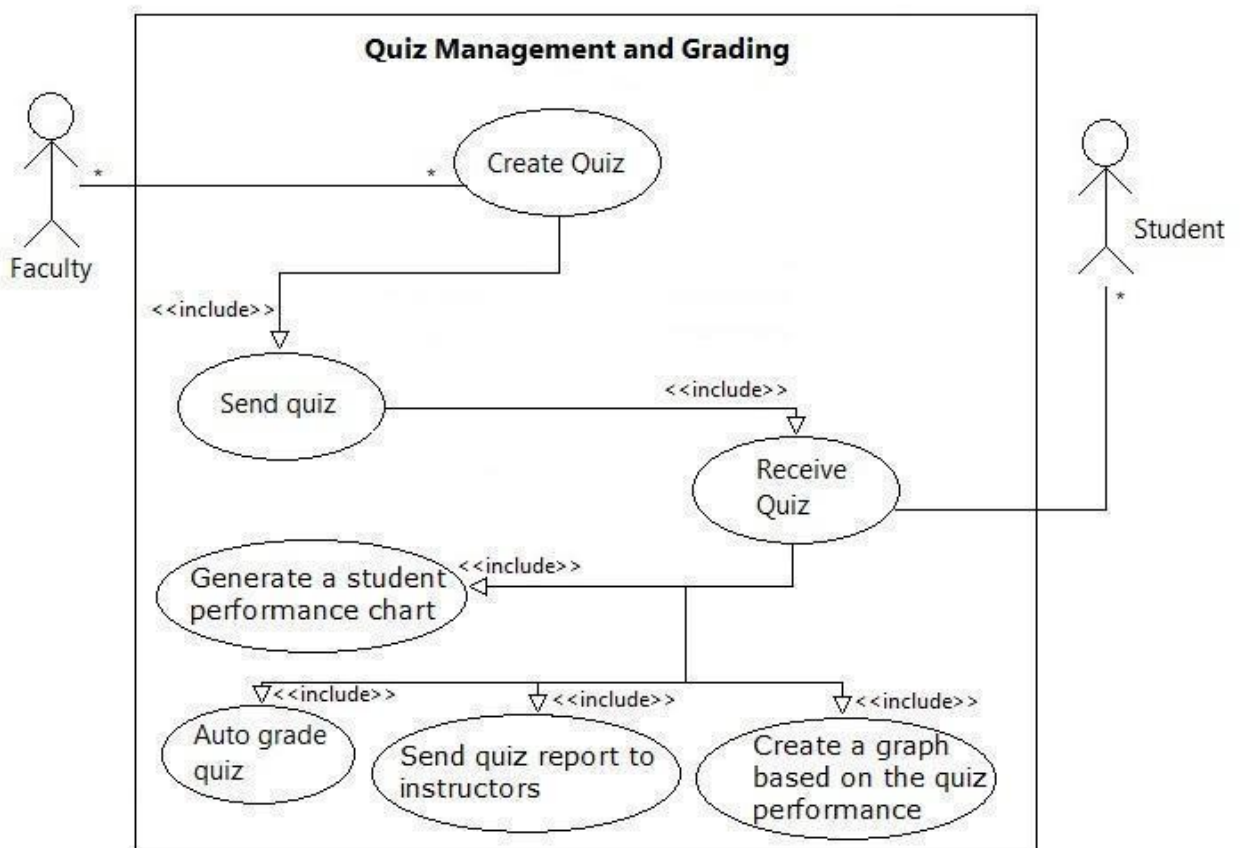- UC-37 Generate a student performance chart

Fig 3.7.1.3: Use case diagram for Quiz management and grading.

Implemented Codes:

UserFunctions.java

```java
package
com.example.eclassroom.library;


import java.util.ArrayList;

import java.util.List;

import
org.apache.http.NameValuePair;

import
org.apache.http.message.BasicNameVa
luePair;

import org.json.JSONObject;

import android.content.Context;


public class UserFunctions {


  private JSONParser jsonParser;


  // URL of the PHP API

  private static String loginURL =
"http://192.168.123.114/EClassroom/
";//192.168.0.102

  private static String courseURL =
"http://192.168.123.114/EClassroom/
show_courses.php";


  private static String login_tag =
"login";

  private static String register_tag
= "register";

  private static String forpass_tag
= "forpass";

  private static String chgpass_tag
= "chgpass";

  private static String
showCourses_tag = "loadCourse";

 // constructor

 public UserFunctions() {

  jsonParser = new JSONParser();

 }


 /**

  * Function to Login

  **/

 public JSONObject loginUser(String
email, String password) {

  // Building Parameters

  List<NameValuePair> params = new
ArrayList<NameValuePair>();

  params.add(new
BasicNameValuePair("tag",
login_tag));

  params.add(new
BasicNameValuePair("email",
email));

  params.add(new
BasicNameValuePair("password",
password));

  JSONObject json =
jsonParser.getJSONFromUrl(loginURL,
params);

  return json;

 }
```

```java
/**
 * Function to show course list
for individual student
 */

public JSONObject getCourseList(){

// Building Parameters

List<NameValuePair> params = new
ArrayList<NameValuePair>();

params.add(new
BasicNameValuePair("tag",
showCourses_tag));

JSONObject json =
jsonParser.getJSONFromUrl(courseURL
, params);

return json;

}
}
```

JSONParser.java

```java
package
com.example.eclassroom.library;


import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStream;

import java.io.InputStreamReader;

import
java.io.UnsupportedEncodingExceptio
n;

import java.util.List;

import org.apache.http.HttpEntity;

import
org.apache.http.HttpResponse;

import
org.apache.http.NameValuePair;

import
org.apache.http.client.ClientProtoc
olException;

import
org.apache.http.client.entity.UrlEn
codedFormEntity;

import
org.apache.http.client.methods.Http
Post;

import
org.apache.http.impl.client.Default
HttpClient;

import org.json.JSONException;

import org.json.JSONObject;


import android.util.Log;


public class JSONParser {


    static InputStream is = null;

    static JSONObject jObj = null;

    static String json = "";


    // constructor

    public JSONParser() {


    }
```

```java
public JSONObject
getJSONFromUrl(String url,
List<NameValuePair> params) {


        // Making HTTP request

        try {

                // defaultHttpClient

                DefaultHttpClient
httpClient = new
DefaultHttpClient();

                HttpPost httpPost = new
HttpPost(url);

                httpPost.setEntity(new
UrlEncodedFormEntity(params));


                HttpResponse
httpResponse =
httpClient.execute(httpPost);

                HttpEntity httpEntity =
httpResponse.getEntity();

                is =
httpEntity.getContent();


        } catch
(UnsupportedEncodingException e) {

                e.printStackTrace();

        } catch
(ClientProtocolException e) {

                e.printStackTrace();

        } catch (IOException e) {

                e.printStackTrace();

        }

        try {

                BufferedReader reader =
new BufferedReader(new
InputStreamReader(

                        is, "iso-8859-
1"), 8);

                StringBuilder sb = new
StringBuilder();

                String line = null;

                while ((line =
reader.readLine()) != null) {

                        sb.append(line +
"\n");

                }

                is.close();

                json = sb.toString();

                Log.e("JSON", json);

        } catch (Exception e) {

                Log.e("Buffer Error",
"Error converting result " +
e.toString());

        }


        // try parse the string to
a JSON object

        try {

                jObj = new
JSONObject(json);

        } catch (JSONException e) {

                Log.e("JSON Parser",
"Error parsing data " +
e.toString());

        }
```

```php
            return jObj;

    }

}
```

DB_Functions.php

```php
<?php

class DB_Functions {

    private $db;

    //put your code here
    // constructor
    function __construct() {
        require_once
'DB_Connect.php';
        // connecting to database
        $this->db = new
DB_Connect();
        $this->db->connect();
    }

    // destructor
    function __destruct() {

    }


    /**
     * Random string which is sent
by mail to reset password
     */

public function random_string()
{
    $character_set_array = array();
    $character_set_array[] =
array('count' => 7, 'characters' =>
'abcdefghijklmnopqrstuvwxyz');
    $character_set_array[] =
array('count' => 1, 'characters' =>
'0123456789');
    $temp_array = array();
    foreach ($character_set_array
as $character_set) {
        for ($i = 0; $i <
$character_set['count']; $i++) {
            $temp_array[] =
$character_set['characters'][rand(0
,
strlen($character_set['characters']
) - 1)];
        }
    }
    shuffle($temp_array);
    return implode('',
$temp_array);
}


public function
forgotPassword($forgotpassword,
$newpassword, $salt){
    $result = mysql_query("UPDATE
`users` SET `encrypted_password` =
'$newpassword',`salt` = '$salt'

WHERE `email` =
'$forgotpassword'");

if ($result) {

return true;

}
else
{
return false;
}

}
    /**
     * Verifies user by email and
password
     */
    public function
getUserByEmailAndPassword($email,
$password) {
        $result =
mysql_query("SELECT * FROM users
WHERE email = '$email'") or
die(mysql_error());
        // check for result
```

```php
        $no_of_rows =
mysql_num_rows($result);
        if ($no_of_rows > 0) {
            $result =
mysql_fetch_array($result);
            $salt =
$result['salt'];
            $encrypted_password =
$result['encrypted_password'];
            $hash = $this-
>checkhashSSHA($salt, $password);
            // check for password
equality
            if ($encrypted_password
== $hash) {
                // user
authentication details are correct
                return $result;
            }
        } else {
            // user not found
            return false;
        }
    }

 /**
    * Checks whether the email is
valid or fake
    */
public function validEmail($email)
{
   $isValid = true;
   $atIndex = strrpos($email, "@");
   if (is_bool($atIndex) &&
!$atIndex)
   {
      $isValid = false;
   }
   else
   {
      $domain = substr($email,
$atIndex+1);
      $local = substr($email, 0,
$atIndex);
      $localLen = strlen($local);
      $domainLen = strlen($domain);
      if ($localLen < 1 ||
$localLen > 64)
      {
         // local part length
exceeded
         $isValid = false;
      }
      else if ($domainLen < 1 ||
$domainLen > 255)
      {
         // domain part length
exceeded
         $isValid = false;
      }
      else if ($local[0] == '.' ||
$local[$localLen-1] == '.')
      {
         // local part starts or
ends with '.'
         $isValid = false;
      }
      else if
(preg_match('/\\.\\./', $local))
      {
         // local part has two
consecutive dots
         $isValid = false;
      }
      else if (!preg_match('/^[A-
Za-z0-9\\-\\.]+$/', $domain))
      {
         // character not valid in
domain part
         $isValid = false;
      }
      else if
(preg_match('/\\.\\./', $domain))
      {
         // domain part has two
consecutive dots
         $isValid = false;
      }
      else if
(!preg_match('/^(\\\\.|[A-Za-z0-
9!#%&`_=\\/$\'*+?^{}|~.-])+$/',

str_replace("\\\\","",$local)))
      {
         // character not valid in
local part unless
```

```php
        // local part is quoted
        if
(!preg_match('/^"(\\\\"|[^"])+"$/',

str_replace("\\\\","",$local)))
        {
            $isValid = false;
        }
    }
    if ($isValid &&
!(checkdnsrr($domain,"MX") ||
 â†ªcheckdnsrr($domain,"A")))
        {
            // domain not found in DNS
            $isValid = false;
        }
    }
    return $isValid;
}


    /**
     * Encrypting password
     * returns salt and encrypted
password
     */
    public function
hashSSHA($password) {

        $salt = sha1(rand());
        $salt = substr($salt, 0,
10);
        $encrypted =
base64_encode(sha1($password .
$salt, true) . $salt);
        //$hash = array("salt" =>
$salt, "encrypted" => $encrypted);
            $hash = array("salt"
=> $salt, "encrypted" =>
$password);
        return $hash;
    }

    /**
     * Decrypting password
     * returns hash string
     */
```

```php
    public function
checkhashSSHA($salt, $password) {

        $hash =
base64_encode(sha1($password .
$salt, true) . $salt);

        return $hash;
    }

    /*
     * Get course list
     * for individual students
     */
    public function courseList(){

            $result =
mysql_query("SELECT * FROM
courses") or die(mysql_error());

            return $result;

    }
}

?>
```

Index.php

```php
<?php


/**

 PHP API for Login, Register,
Changepassword, Resetpassword
Requests and for Email
Notifications.

 **/


if (isset($_POST['tag']) &&
$_POST['tag'] != '') {

    // Get tag

    $tag = $_POST['tag'];
```

```php
    // Include Database handler

    require_once
'include/DB_Functions.php';

    $db = new DB_Functions();

    // response Array

    $response = array("tag" =>
$tag, "success" => 0, "error" =>
0);


    // check for tag type

    if ($tag == 'login') {

        // Request type is check
Login

        $email = $_POST['email'];

        $password =
$_POST['password'];


        // check for user

        $user = $db-
>getUserByEmailAndPassword($email,
$password);

        if ($user != false) {

            // user found

            // echo json with
success = 1

            $response["success"] =
1;

$response["user"]["uid"] =
$user["uid"];//$user["unique_id"]

$response["user"]["email"] =
$user["email"];


$response["user"]["fname"] =
$user["first_name"];


$response["user"]["lname"] =
$user["last_name"];


$response["user"]["user_type"] =
$user["user_type"];

            //
$response["user"]["uname"] =
$user["username"];


$response["user"]["created_at"] =
$user["created_at"];


            echo
json_encode($response);
        } else {

            // user not found

            // echo json with error
= 1

            $response["error"] = 1;

            $response["error_msg"]
= "Incorrect email or password!";

            echo
json_encode($response);
        }

    }

    else {

        $response["error"] = 3;
```

```php
        $response["error_msg"] =                } else {
"JSON ERROR";
                                                    echo "EClassroom Login API";
        echo                                   }
json_encode($response);
                                            ?>
    }
```

References

References

[1] HEICK, T. 2014. 10 Characteristics Of A Highly Effective Learning Environment. *Teach Thought*. Retrieved September 01, 2014
http://www.teachthought.com/learning/10-characteristics-of-a-highly-effective-learning-environment/

[2] UNITED NATIONS. 2014. The World in 2014: ICT Facts and Figures.
http://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx

[3] ENRIQUEZ, A. G. Enhancing Student Performance Using Tablet Computers. *College Teaching*, 58, 3, 77-84. http://www.editlib.org/p/66053/

[4] MEERE, D., GANCHEV, I., O'DROMA, M., NOONAN, P. and HAYES, S. 2011. Using Mobile Phones As Tools For Enhanced Blended Learning. *Journal of International Scientific Publications: Educational Alternatives, 9* , 1,157-175.

[5] https://www.surveymonkey.com/s/BR5QN78 [Available]

[6] GOH, S. C., GALLIGAN, L.A. AND KU, H. Wireless tablet PCs for enhanced teaching at an Australian regional university teaching both campus and distance modes. In *120th american society for engineering education annual conference and exposition, ASEE 2013,* Atlanta, GA, 2013, Engineering Education: Frankly, We Do Give a D*mn

[7] LIU, T. C., WANG, H. Y., LIANG, J. K., CHAN, T. W., KO, H. W. and YANG, J. C. Wireless And Mobile Technologies To Enhance Teaching and Learning. *Journal of Computer Assisted Learning, 19,* 3, 371-382.

[8] STATE.TIGRIS.ORG..2009. Project home. Browse templates. Retrieved from http://readyset.tigris.org/nonav/templates/index.html, *[Last visited Sep 03,2014]*

[9] GOOGLE INC. 2008. Developers. Android Developers.
http://developer.android.com/index.html *[Last visited Sep 03,2014]*

[10] STACK EXCHANGE INC. 2014. Stack Overflow. http://stackoverflow.com/ *[Last visited Aug 23,2014]*

[11] APACHE FRIENDS. 2014. Xampp. https://www.apachefriends.org/community.html *[Last visited Aug 25,2014]*

[12] ANDROIDHIVE. 2014. http://www.androidhive.info/ *[Last visited Aug 22,2014]*

[13] EDX. 2014. https://www.edx.org/ *[Last visited Jul 15, 2014]*

[14] ERACK NETWORK.2008. php/mysql tutorial. http://www.tizag.com/ *[Last visited Sep 03,2014]*