

E-LEARNING PORTAL
PROJECT REPORT

Submitted by

ASWATHY SATHYAN

Reg.No. SJC16MCA9

to

the APJ Abdul Kalam Technological University
in partial fulfillment of the requirement for the award of the degree
of

Master of Computer Applications



Department of Computer Science and Applications

St. Josephs College of Engineering and Technology

Palai

MAY 2019

DECLARATION

I undersigned hereby declare that the project report **E-Learning Portal**, submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala, is a bonafide work done by me under supervision of **Mrs.Rinu Rachel Varughese**. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Choondacherry

29 May, 2019

Aswathy Sathyan

**DEPARTMENT OF COMPUTER SCIENCE AND APPLICATIONS
ST. JOSEPHS COLLEGE OF ENGINEERING AND TECHNOLOGY**

Palai, Kottayam 686 579

(Approved by AICTE and affiliated to APJ Abdul Kalam Technological University)



CERTIFICATE

This is to certify that the report entitled **E-Learning Portal** submitted by **ASWATHY SATHYAN** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications is a bonafide record of the project work carried out by her under my guidance and supervision.

Mrs. Rinu Rachel Varughese
Asst. Professor, MCA
Internal Guide

Mrs. Rinu Rachel Varughese
Asst. Professor, MCA
Project Co-ordinator

Dr. T.D Jainendrakumar
Head of the Department

Viva-voce held on.....

External Examiner 1:

External Examiner 2:

ACKNOWLEDGEMENT

First and foremost, we give all glory, honour and praise to God Almighty who gave us wisdom and enabled us to complete the project successfully. I also express sincere thanks, from the bottom of our heart, to our parents for their encouragement and support in all our endeavors and especially in this project.

I respect and thank the management of St. Josephs College of Engineering and Technology for providing me an opportunity and platform for doing the project work.

I am extremely thankful to our beloved Principal, **Dr. David J** for providing such a nice support and facilities to carry out this project.

I am extremely indebted to **Dr. Jainendrakumar T D**, Professor Head, department of Computer Science and Applications, who has been a constant source of inspiration and without his tremendous help and support this project would not have been materialized.

I owe my deep gratitude to my project guide **Mrs. Rinu Rachel Varughese**, Asst. Professor, department of Computer Science and Applications who took keen interest on my project and guided us all along, till the completion of my project work by providing all the necessary information for developing a good system.

I am thankful to and fortunate enough to get constant encouragement, support and guidance from all the faculty members of department of Computer Science and Applications, which helped us in successfully completing my project work. Also, I would like to extend my sincere esteem to all staff in laboratory for their timely support.

Aswathy Sathyan

Abstract

The proposes of this project is the establishment of a web based e-learning portal and defines components that make up an E-learning system and the objects that must be moved among these components. E-learning is essentially the computer and network-enabled transfer of skills and knowledge. E-learning applications and processes include Web-based learning, computer-based learning, virtual education opportunities and digital collaboration. This system helps teachers and students in a college to share all study materials more easily at a single place. So that students can easily get all the notes uploaded by the faculties in previous years. There is a forum where all users can discuss about there doubts. This system consist of an internal mailing system where students and faculties can send messages to any of the registered users.

This system consists of three modules.

- Student Module
- Faulty Module
- Administration Module

This system helps students to improve there aptitude skill by attending the online quiz. Students can view and download notes, reports or video tutorials that are uploaded by the faculties. They can also see their Previous Quiz Results and can thus evaluate the improvements in their aptitude skill. Students can ask queries regarding the documents uploaded by the faculties. Faculties can upload notes, reports and video tutorials which are then verified by the admin. Faculties can also add Quiz Questions, Categories etc. They can set Quiz for the students in the concerned department. Faculties can give replies to the queries posted by the students. They can also view student details and their quiz results. Admin is the one who manage all the users, can add new users and delete existing users. Admin manages the quiz questions by adding or deleting questions and admin verify the documents uploaded by the faculties and documents are available to students after this verification. Admin can add course details, department details etc.

Contents

1	INTRODUCTION	1
1.1	PROBLEM DEFINITION	1
1.2	ABOUT THE ORGANIZATION	2
1.3	OBJECTIVE OF THE PROJECT	2
2	SYSTEM ANALYSIS	3
2.1	INITIAL INVESTIGATION	3
2.2	EXISTING SYSTEM	4
2.3	PROPOSED SYSTEM	4
2.3.1	Advantages of the Proposed System	5
2.3.2	Features of the Proposed System	5
2.4	FEASIBILITY STUDY	6
2.4.1	Technical Feasibility	7
2.4.2	Economic Feasibility	7
2.4.3	Operational Feasibility	8
2.4.4	Behavioral Feasibility	9
3	SYSTEM DESIGN	10
3.1	SOFTWARE AND HARDWARE REQUIREMENT SPECIFICATION	10
3.1.1	Hardware Requirement	10
3.1.2	Software Requirement	11
3.2	SYSTEM DESIGN	11
3.2.1	Non-Functional Requirements	12
3.2.2	Modular Design	14
3.2.3	Input Design	17
3.2.4	Output Design	18
3.2.5	Database Design	19

3.2.6	Data Flow diagram	27
3.3	UNIFIED MODELING LANGUAGE[UML]	31
3.3.1	Use case Diagram	32
3.3.2	Sequence Diagram	37
3.3.3	Activity Diagram	41
3.4	TOOLS AND PLATFORMS	45
3.4.1	Introduction to Microsoft.Net	45
3.4.2	The .NET Platform	45
3.4.3	The .NET Products	45
3.4.4	The .NET Services	46
3.4.5	The .NET Framework	46
3.4.6	Common Language Runtime (clr)	46
3.4.7	ASP.NET	46
3.4.8	SQL SERVER 2012	47
3.4.9	Normalization	47
3.5	GITHUB	48
4	SYSTEM TESTING	50
4.1	TESTING METHODOLOGIES AND STRATEGIES	50
4.1.1	Unit Testing	51
4.1.2	Integration Testing	51
4.1.3	System Testing	52
4.1.4	User Acceptance Testing	52
4.2	TEST CASES	53
5	SYSTEM IMPLEMENTATION	56
6	CONCLUSION	57
	REFERENCES	58

Appendix A APPENDICES	59
A.1 SCREEN SHOTS INPUT FORM,OUTPUT FORMS	59
A.2 SAMPLE CODE	75

List of Tables

3.1	User Details	20
3.2	Qualification Details	21
3.3	Experience Details	21
3.4	Uploaded Document Details	22
3.5	Program Details	22
3.6	District Details	22
3.7	Department Details	22
3.8	Course Details	23
3.9	Subject Details	23
3.10	Quiz-Category Details	23
3.11	Quiz-SubCategory Details	24
3.12	Query Details	24
3.13	Query-Reply	24
3.14	Mail-Message	25
3.15	Quiz-Questions	25
3.16	Quiz-Report	26
3.17	Temporary-Questions	26

List of Figures

3.1	LEVEL-0 DFD	27
3.2	LEVEL-1 DFD for Student	28
3.3	LEVEL-1 DFD for Faculty	29
3.4	LEVEL-1 DFD for Admin	30
3.5	Usecase Diagram for Student	34
3.6	Usecase Diagram for Faculty	35
3.7	Usecase Diagram for Admin	36
3.8	Sequence Diagram for Student	38
3.9	Sequence Diagram for Faculty	39
3.10	Sequence Diagram for Admin	40
3.11	Activity Diagram for Student	42
3.12	Activity Diagram for Faculty	43
3.13	Activity Diagram for Admin	44
3.14	Git History	49
4.1	Validation Testing	54
4.2	Unit Testing	55
A.1	Home Page	59
A.2	Login Page	60
A.3	Admin Home Page	61
A.4	Document Verification by Admin	62
A.5	View/Delete Staff Details	62
A.6	User Registration	63
A.7	Add Category	63
A.8	Add Questions	64
A.9	View Student Quiz Results	64

A.10 View/Delete Quiz Questions	65
A.11 Student Home Page	66
A.12 Quiz Category Selection	67
A.13 Attend Online Quiz	67
A.14 Ask Doubts	68
A.15 View/Download Notes	68
A.16 View/Edit Profile	69
A.17 View/Download Reports	69
A.18 View/Download Video Tutorials	70
A.19 Faculty Home Page	71
A.20 View Queries Posted by Students	72
A.21 Add Replies to the Queries	72
A.22 Upload E-Notes	73
A.23 Upload Video Tutorials	73
A.24 View/Sent Mails to Other Users	74

Chapter 1

INTRODUCTION

1.1 PROBLEM DEFINITION

The paper proposes the establishment of a web based e-learning portal and defines components that make up an E-learning system and the objects that must be moved among these components. E-learning is essentially the computer and network-enabled transfer of skills and knowledge. E-learning applications and processes include Web-based learning, computer-based learning, virtual education opportunities and digital collaboration. This system consists of three modules:- Student, Faculty, Admin. This system helps teachers and students in an institution to share all study materials more easily at a single place. So that students can easily get all the notes uploaded by the faculties in previous years. There is a forum where all users can discuss about their doubts. This system consists of an internal mailing system where students and faculties can send messages to any of the registered users.

This system helps students to improve their aptitude skill by attending the online quiz. Students can view and download notes, reports or video tutorials that are uploaded by the faculties. They can also see their Previous Quiz Results and can thus evaluate the improvements in their aptitude skill. Students can ask queries regarding the documents uploaded by the faculties. Faculties can upload notes, reports and video tutorials which are then verified by the admin. Faculties can also add Quiz Questions, Categories etc. They can set Quiz for the students in the concerned department. Faculties can give replies to the queries posted by the students. They can also view student details and their quiz results. Admin is the one who manages all the users, can add new users and delete existing users. Admin manages the quiz questions by adding or deleting questions and admin verifies the documents uploaded by the faculties and documents are available to students after this verification. Admin can add course details, department details etc.

1.2 ABOUT THE ORGANIZATION

St. Josephs College of Engineering and Technology, Palai was instituted with the objective of developing a center of professional learning with a distinct identity and character. The college aims to provide the kind of education that helps to achieve academic excellence and thereby ensures a challenging and satisfying career for the students on the successful completion of the programme. With this perspective, training is organized on a regular basis for the development of personality, learning and communication skills as well as employability skills. Discipline, hard work, positive thinking, commitment to excellence and abiding faith in the Almighty are the guiding principles that propel the college to its vision of emerging as a Centre of Excellence in technical education in the country.

SJCET right from inception, has been maintaining high levels of standard in academic and extra curricular realms of activities. We offer BTECH degree courses in 6 engineering disciplines, and Masters Degree courses in Engineering, Computer Application and Business Administration. In the short span of a decade of its existence and among the six batches of students that have graduated, the college bagged several university ranks and has a remarkably high percentage of pass.

1.3 OBJECTIVE OF THE PROJECT

The main objective of the project is to create a website that helps students and teachers in a college to share the study materials like notes, reports and video tutorials. This system helps the faculties to share the notes with students by uploading all documents in this system. Students can evaluate their aptitude skills by attending Quiz on various topics. Admin manage all the users, quiz questions, course details and subject details.

Chapter 2

SYSTEM ANALYSIS

System analysis is a management technique which helps us in designing a new system or improving an existing system. System analysis is a process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvement of the system. During these places, the analyst and the user come to a detailed agreement on what function the proposed system has to perform. This one contains:

- Inputs to be supplied.
- Output to be supplied.
- Procedures to get the outputs from the given inputs.
- Data to be retained.

2.1 INITIAL INVESTIGATION

The purpose of this document is to give a clear picture of the module designs of the project E-Learning Portal. The website provide an easy way to upload and access previous year notes, seminar/assignment reports and video tutorials. The registered Faculty can upload notes, assignment/seminar reports and video tutorials to the repository and admin can approve or reject these documents submitted by the faculties. The students and staff registration is done by the admin and the registered students can attend quiz on various topics. This website provides an easy way to clear the doubts of students by discussing it with other students and faculties in the college. This document is developed after a number of consultations with the staff and the HOD and considering the complete requirement specifications of the given project.

This project also helps to understand various functionalities of the modules in the project as well as it gives a pictorial design of how the website will look like with its functionalities working together various to achieve the requirements.

2.2 EXISTING SYSTEM

The study of the existing system is a pre-requisite for developing any software system. The study of the system reveals many features of the existing system. This gives analyst an insight into the working of the system and helps the developer to design an appropriate system, which will eliminate the many limitations present in the existing system.

Limitations of the Existing System are :-

- It is a manual system
- Process is by means of paper work
- Difficult to keep all the paper records
- The file manipulating method was not done in a centralized manner
- Document storing, accessing them takes more time
- Searching process is mainly done manually and it is difficult
- Chances of loss in document containing important details
- Difficult to find out accurate data in minimal time
- Time consuming and miss handling of reports

2.3 PROPOSED SYSTEM

The proposed system computerization is developed using SQL Server as back-end and ASP.Net ad front-end. The Asp.net framework is managed, type safe environment for application, development and execution. The software is developed as a simulated system and the complex procedures are avoided to make the system east to use. The proposed system is user friendly and has simplicity and security. In the proposed system the data redundancy can be avoided to certain extend and the data consistency can be maintained. The record keeping and searching process are easy.

2.3.1 Advantages of the Proposed System

- Give solution to the current system problems
- Less time consuming and more efficient
- Result will be very precise and accurate
- Easy to use and fast
- Simple user interface to reduce processing time
- Easy searching and storing documents
- Eliminate chances for errors and reduce effort

2.3.2 Features of the Proposed System

The various features of proposed system are as follows :-

- Access to the system and database as per user identification
- The maximum security ensured
- Integrity reliability and integrity of data
- User-friendly and flexible in all aspects
- Data entry updates is quite easy
- Effective table manipulation as facilitated by the rich SQL
- Good validation checking
- Easy maintenance
- Removes chances of leakage of information
- Provides a better record keeping system

All these form the major aspects and advantages of the proposed system. Provision is made for effective improvements of maintenance are needed at any stage.

2.4 FEASIBILITY STUDY

An important outcome of the preliminary investigation is the determination that the system requested is feasible. Feasibility study is carried out to select the best system that meets the performance requirements. During system analysis, a feasibility study of the proposed system was carried out to see whether it was beneficial to the organization. The main aim of the feasibility study is to determine whether it would be financially and technically feasible to develop the product. While evaluating the existing system, many advantages and disadvantages raised. Analyzing the problem thoroughly forms the vital part of the system buddy. Problematic areas are identified and information is collected.

Feasibility study is both necessary and prudent to evaluate the feasibility of the project at the earliest possible time. It involves preliminary investigation of the project and examines whether the designed system will be useful to the organization. Months or years of effort, thousand for millions of money and untold professional embarrassment can be averted if an in-conceived system is recognized early in the definition phase.

Eight steps involved in the feasibility analysis are :-

- Form a project team and appoint a project leader
- Prepare system flowcharts
- Enumerate potential proposed system
- Define and identify characteristics of proposed system
- Determine and evaluate performance and cost effective of each proposed system
- Weight system performance and cost data
- Select the best Proposed system

The main aim of feasibility study is to evaluate alternative site and propose the most feasible and desirable site for development. If there is no loss for the organization then the proposed system

is considered financially feasible. A feasibility study is carried out to select the best system that meets performance requirements.

The feasibility study activity involves the analysis of the problem and collection of all relevant information relating to the product such as the different data items which would be input to the system, the processing required to be carried out on these data, 5 the output data required to be produced by the system as well as various constraints on the behavior of the system.

In this scenario, problems are identified. Essential data are being gathered for the existing problems. It is necessary that this analysis familiarizes the designer with objectives, activities, and the function of the organization in which the system is to be implemented. The feasibility study was divided into four:- Technical, Economical, Operational and Behavioral. It is summarized below :-

2.4.1 Technical Feasibility

According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such ad software facilities, procedure, inputs, are identified. While considering the problems of existing system, it is sufficient to implement the new system. The proposed system can be implemented to solve issues in the existing system. It includes the evaluation of and how it meets the proposed system. This system use ASP.NET as front-end technology and SQL Server back-end technology.

2.4.2 Economic Feasibility

Economic feasibility deals about the economical impact faced by the organization to implement a new system. Financial benefits must equal or exceed the costs. The cost of conducting a full system, including software and hardware cost for the class of application being considered should be evaluated. Economic Feasibility in this project:

- The cost to conduct a full system investigation is possible.
- There is no additional manpower requirement.
- There is no additional cost involved in maintaining the proposed system.

Economic justification is generally the Bottom Line consideration for most systems. Economic justification includes a broad range of concerns that includes cost benefit analysis. In this we weight the cost and the benefits associated with the candidate system and if it suits the basic purpose of the organization i.e. profit making, the project is making to the analysis and design phase. The financial and the economic questions during the preliminary investigation are verified to estimate the followings:

- The cost to conduct a full system investigation.
- The cost of hardware and software for the class of application being considered.
- The benefits in the form of reduced cost.

The proposed system will give the minute information; as a result the performance is improved. This feasibility checks whether the system can be developed with the available funds. The E-Learning Portal does not require enormous amount of money to be developed. This can be done economically if planned judicially, so it is economically feasible.

2.4.3 Operational Feasibility

Methods of processing and presentation are all according to the needs of clients since they can meet all user requirements here. The proposed system will not cause any problem under any circumstances and will work according to the specifications mentioned. Hence the proposed system is operationally feasible.

People are inherently resistant to change and computer has been known to facilitate changes. The system operation is the longest phase in the development life cycle of a system. So, Operational Feasibility should be given much importance. This system has a user-friendly interface. Thus it is easy to handle.

2.4.4 Behavioral Feasibility

In todays world, computer is an inevitable entity. As per the definition of behavior design, many valid points are recognized in this study. This system behavior changes according to different environment. In order to ensure proper authentication and authorization and security of sensitive data of the admin or users, login facilities are provided. These are the main feasibility studies tested in this application.

Chapter 3

SYSTEM DESIGN

3.1 SOFTWARE AND HARDWARE REQUIREMENT SPECIFICATION

The primary goal of the system analyst is to improve the efficiency of the existing system. For that study of specification of the requirement is very essential. For the development of the new system, a preliminary survey of the existing system will be conducted. An investigation is done whether the up gradation of the system into an application program could solve the problems and eradicate the inefficiency of the existing system. This gives an idea about the system specifications required to develop and install the project E-LEARNING PORTAL.

The System Requirements Specification is based on the System Definition. The requirement specifications are primarily concerned with functional and performance aspect of a software product and emphasis are placed on specifying product characteristics implying how the product will provide those characteristics. One of the most difficult tasks is selecting software, once the system requirement is find out then we have to determine whether a particular software package fits for those system requirements. This selection summarizes the application requirement.

3.1.1 Hardware Requirement

- CPU - INTEL(R) Core(TM) i5-8250U
- HARD DISK SPACE -931.51 GB
- RAM - 8 GB
- DISPLAY - 19 STANDARD RATIO LCD MONITOR
- KEYBOARD - 101- or 102- KEYS
- CLOCK SPEED - 1.60 GHZ

3.1.2 Software Requirement

- OPERATING SYSTEM - WINDOWS 10(R)
- WEB SERVER - IIS 7.5
- FRONT END - ASP.NET
- BACK END - MS SQL SERVER 2012

3.2 SYSTEM DESIGN

Designing the system in an effective way leads to the smooth working of any softwares. System design is the process of developing specification for a candidate system that meet the criteria established in the system analysis. Major step in the system design is the preparation of the input forms and output reports in a form applicable to the user. The main objective of the system design is to use the package easily by any computer operator. System design is the creative act of invention, developing new inputs, and database, off-line files, method, procedure and output for processing business to meet an organization objective. System design builds information gathered during the system analysis. This system is designed neatly so that user will never get ambiguity while using the system.

The System Requirements Specification is based on the System Definition. The requirement specifications are primarily concerned with functional and performance aspect of a software product and emphasis are placed on specifying product characteristics implying how the product will provide those characteristics. One of the most difficult tasks is selecting software, once the system requirement is find out then we have to determine whether a particular software package fits for those system requirements. This selection summarizes the application requirement.

3.2.1 Non-Functional Requirements

Performance Requirements

For the efficient performance of the application, network must have high bandwidth so that the task of centralized management does not lead to network jam. Also the hard disk capability must be high so that data can be effectively stored and retrieved.

Security Requirements

Security requirements of this application involves authentication using user name and password so that invalid users are restricted from data access. For the security of data, periodic database backups must be performed so that we can recover data in the case of data loss.

The most creative and challenging phase of the system life cycle is the system design. The term design describes a final system and the process by which it is developed. It refers to the technical specification that will be applied in implementing the candidate system. In system design, we move from the logical to the physical aspects of the life cycle.

The first step is to determine how the output is to be produced and in what format. Then input data and master files have to be designed as the next step and finally the impact of the candidate system on the user and organization are documented and evaluated by the management. After identifying the problem and the limitation of the existing system, a detailed design of the proposed system is conducted. Free flow personnel interview and reference to previous records prepared manually were the only methods taken to collect necessary information. At present, all organizations are on the path of computerization process.

Design is the phase that indicates the final system. It is the solution, the translation of requirements into ways of meeting them. In this phase the following elements were designed namely, data flow, data stores, processes, procedures was formulated in a manner that meet the project requirements. After logical design physical construction of the system is done.

The database tables, input screens, output screens, output reports are designed. After analyzing the various functions involved in the system the database, labels as dictionaries designed. Care is taken for the field name to be in self-explanatory form. Unnecessary fields are avoiding so as not affecting the storage system. Care must be taken to design the input screen in the most user-friendly way so as to help even the novice users to make entries approximately in the right place. This is being accomplished by the use of giving online help messages, which are brief and cleanly prompts users for appropriate action.

Design is the only way that we can accurately translate a customers requirements into a finished software product or system. Without design, risk of building an unstable system exist one that will fail when small changes are made, one that will be difficult to test.

All input screens in the system are user friendly and are designed in such a way that even a layman can operate. The sizes of all screens are standardized. Reports generated in this software give the finer accepts of the required information, which helps in taking vital decision. The importance of the software design can be stated with a single word quality. Design is a place where quality is fostered in software development. Design is the only way where requirements are actually translated into a finished software product or system.

3.2.2 Modular Design

Mainly this project consists of 4 Modules:

- Student Module
- Faulty Module
- Administration Module

Administration Module

Administrator is the main actor in this system. He has the entire control of the system which includes adding all the details to the system. Brief description about the functionalities performed by the admin is given below. After the admin successfully login to this website the admin can perform the functionalities including:

- **Admin Login**

By the Username and password admin can login to the system.

- **Add Department/Delete Department**

Admin is responsible to add the departments of the college and also responsible for delete the departments.

- **Add Course/Delete Course**

Admin is responsible to add or delete courses according to each department in the college.

- **Add Subject/Delete Subject**

Admin is responsible to add the subjects according to each Program and also can delete the subjects.

- **Add/ Delete Faculties, Students and other Admins**

Admin is responsible to add or delete Students, Faculties and other Admin Details.

- **Add/ Delete Quiz Categories**

Admin is responsible to add quiz category details and can also delete category details.

- **Add/ Delete Quiz Questions**

Admin is responsible to add or delete Quiz Questions corresponding to different quiz categories.

- **Document Verification**

Admin is responsible to verify the documents uploaded by faculties and thus approve or reject the document.

- **Delete Queries Replies**

Admin is responsible to delete the queries and replies posted by students and faculties.

Faculty Module

Faculty is important actor in the system. Faculty can upload Notes, Previous year Assignment/Seminar Reports, Video Tutorials prepared for subjects allotted to corresponding staff. After the faculty successfully login to this website the faculty can perform the functionalities including:

- **Faculty Login**

By the Username and password faculties can login to the system.

- **Upload Documents**

Faulty is responsible to add documents like notes, assignment/seminar reports and video tutorials prepared for subjects allotted to corresponding staff.

- **Add Quiz Categories**

Admin is responsible to add quiz category and subcategory details .

- **Add Quiz Questions**

Admin is responsible to add Quiz Questions corresponding to different quiz categories.

- **Search Student Details**

Faculties can view details of the students in the concerned department.

- **Set Quiz**

Faculties can set department wise online quiz on desired topics.

- **View Quiz Reports**

Faculties can verify the quiz results of students in their department.

- **View/Reply to Queries**

Faculties can view and post replies to the queries posted by the students regarding the documents uploaded by the corresponding faculty.

- **Mailing System**

Faculties can sent mails to other students and faculties who have registered in this system.

Student Module

Student is an important actor in this system. Students can refer all the notes, reports and video tutorials uploaded by the faculties. After the student successfully login to this website the faculty can perform the functionalities including:

- **Student Login**

By the Username and password students can login to the system.

- **Search E-Notes**

Students can search E-Notes uploaded by the faculties corresponding to each subjects.

- **Search Reports**

Students can search assignment/seminar reports uploaded by the faculties corresponding to each subjects.

- **Search Video Tutorials**

Students can search Video Tutorials uploaded by the faculties corresponding to each subjects.

- **Ask Doubts**

Students can ask their doubts regarding any topics and view the replies provided by the faculties corresponding to their queries.

- **Attend Online Quiz**

Students can attend online quiz on various categories and can improve their aptitude skills.

- **Mailing System**

Students can sent mails to other students and faculties who have registered in this system.

3.2.3 Input Design

Input design is the process of converting user-oriented input into a computer based format. The goal of the designing input is to make data entry as easy and free from error. In .NET, input to the system is entered through forms. A form is any surface on which information is to be entered, the nature of which is determined by what is already on that surface. If the data going into the system is incorrect, then processing and output will magnify these errors. So designer should ensure that form is accessible and understandable by the user.

End-users are people who communicate to the system frequently through the user interface , the design of the input screen should be according to their recommendations.

The data is validated wherever it requires in the project. This ensures only correct data is entered to the system. GUI is the interface used in input design. All the input data are validated in the order and if any data violates any condition the user is warned by a message and asks to re-enter data. If the data satisfies all the conditions then it is transferred to the appropriate tables in the database. This project uses text boxes and drop down to accept user input. If user enters wrong format then it shows a message to the user. User is never left in confusion as to what is happening. Instead appropriate error messages and acknowledgments are displayed to the user.

3.2.4 Output Design

A quality output is one, which meets the requirement of the end user and presents the information clearly. In any system results of processing are communicated to the user and to the other systems through outputs. In the output design it is determined how the information is to be displayed for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the systems relationship and helps user decision making.

A quality output is one, which meets the requirements of the end user and presents the information clearly. The objective of output design is to convey information about past activities, current status or projections of the future, signal important events, opportunities, problems, or warnings, trigger an action, confirm an action etc. Efficient, intelligible output design should improve the systems relationship with the user and helps in decisions making. In output design the emphasis is on displaying the output on a CRT screen in a predefined format. The primary consideration in design of output is the information requirement and objectives of the end users. The major formation of the output is to convey the information and so its layout and design need a careful consideration. Two phases of the output design are:-

1. Output definition.
2. Output specification.

3.2.5 Database Design

A database is an organized mechanism that has the capability of storing Information through which a user can retrieve stored information in an effective and efficient Manner. The data is the purpose of any database and must be protected. The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual Database Management System (DBMS). In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design.

Tables

Table 3.1: User Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
User-Id	Varchar(20)	Primary Key	It stores unique User Id of users
User-Type	Varchar(50)	Not Null	Should not be null
Name	Varchar(100)	Not Null	Should not be null
Gender	Varchar(10)	Not Null	Should not be null
Mobile-No	numeric(12,0)	Not Null	Should not be null
Dob	date	Not Null	Should not be null
Email-Id	Varchar(100)	Not Null	Should not be null
Religion	Varchar(50)	Null	Can be null
Category	Varchar(50)	Null	Can be null
Nationality	Varchar(50)	Null	Can be null
Blood-Group	Varchar(10)	Null	Can be null
Photo	Varchaar(MAX)	Not Null	Should not be null
Address	Varchar(MAX)	Not Null	Should not be null
District	Varchar(100)	Not Null	Should not be null
State	Varchar(100)	Not Null	Should not be null
Pincode	numeric(7,0)	Not Null	Should not be null
Father-Name	Varchar(50)	Null	Can be null
Occupation	Varchar(50)	Null	Can be null
F-Mob-No	numeric(12,0)	Null	Can be null
Joining-Date	date	Not Null	Should not be null
Admitted-Program	Varchar(MAX)	Null	Can be null
Branch	Varchar(MAX)	Null	Can be null
Department	Varchar(50)	Not Null	Should not be null
Admission-No	Varchar(50)	Not Null	Should not be null
Semester	Varchar(10)	Null	Can be null
Username	Varchar(50)	Null	Can be null
Password	Varchar(50)	Null	Can be null

Table 3.2: Qualification Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Qual-Id	Varchar(20)	Primary Key	It stores unique Qualification Id for each qualification of users
User-Id	varchar(20)	Foreign Key	It refers to the table User-Details
Qualification	varchar(100)	Not Null	Should not be null
Specialization	varchar(100)	Not Null	Should not be null
College	varchar(MAX)	Not Null	Should not be null
University	varchar(MAX)	Not Null	Should not be null
CGPA	float	Null	Can be null
Percentage	float	Not Null	Should not be null

Table 3.3: Experience Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Exp-Id	Varchar(20)	Primary Key	It stores unique Experience Id of users
User-Id	varchar(20)	Foreign Key	It refers to the table User-Details
Organization-Worked	varchar(MAX)	Not Null	Should not be null
Designation	varchar(MAX)	Not	Can be null
From-Date	date	Not	Can be null
To-Date	date	Not	Can be null
Duartion	varchar(50)	Null	Can be null

Table 3.4: Uploaded Document Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Doc-Id	Varchar(50)	Primary Key	It stores unique Document Id
User-Id	varchar(20)	Foreign Key	It refers to the table User-Details
Doc-Type	varchar(50)	Not Null	Should not be null
Subject	varchar(MAX)	Null	Can be null
Topic	varchar(MAX)	Null	Can be null
Document-File	varbinary(MAX)	Not Null	Should not be null
Description	varchar(MAX)	Null	Can be null
Status	varchar(50)	Null	Can be null

Table 3.5: Program Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Prgm-Id	Int	Primary Key	It stores unique Program Id
Program	varchar(MAX)	Not Null	Should not be null

Table 3.6: District Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
D-Id	Int	Primary Key	It stores unique District Id
District	varchar(100)	Not Null	Should not be null
S-Id	int	Foreign Key	It refers to State table

Table 3.7: Department Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Dept-Id	Int	Primary Key	It stores unique Department Id
Department	varchar(MAX)	Not Null	Should not be null

Table 3.8: Course Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
B-Id	Int	Primary Key	It stores unique Course Id
Branch-Name	varchar(MAX)	Not Null	Should not be null
P-Id	int	Foreign Key	It refers to table Program-Details
Dept-Id	int	Foreign Key	It refers to table Department-Details

Table 3.9: Subject Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Sub-Id	Int	Primary Key	It stores unique Subject Id
Subject	varchar(MAX)	Not Null	Should not be null
B-Id	int	Foreign Key	It refers to table Course-Details

Table 3.10: Quiz-Category Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
C-Id	varchar(20)	Primary Key	It stores unique Category Id
Category	varchar(100)	Not Null	Should not be null
Image	varchar(MAX)	Not Null	Should not be null
Created-By	varchar(100)	Null	Can be null
Created-On	date	null	Can be null

Table 3.11: Quiz-SubCategory Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
SubCat-Id	varchar(20)	Primary Key	It stores unique Sub-Category Id
SubCategory	varchar(MAX)	Not Null	Should not be null
Image	varchar(MAX)	Not Null	Should not be null
C-Id	varchar(20)	Foreign Key	It refers to the table Quiz-Category Details
Created-By	varchar(100)	Null	Can be null
Created-On	date	null	Can be null

Table 3.12: Query Details

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Query-Id	int	Primary Key	It stores unique Query Id
QUser-Id	varchar(50)	Foreign Key	It refers to the table User-Details
Subject	varchar(50)	Not Null	Should not be null
Query	varchar(MAX)	Not Null	Should not be null
Date	date	null	Can be null

Table 3.13: Query-Reply

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Reply-Id	int	Primary Key	It stores unique Reply Id
Query-Id	int	Foreign Key	It refers to the table Query Details
RUser-Id	varchar(50)	Foreign Key	It refers to the table User-Details
Reply	varchar(MAX)	Not Null	Should not be null
Date	date	null	Can be null

Table 3.14: Mail-Message

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
m-Id	int	Primary Key	It stores unique Mail Id
from-id	varchar(50)	Not Null	Should not be null
to-id	varchar(50)	Not Null	Should not be null
subject	varchar(MAX)	Not Null	Should not be null
message	varchar(MAX)	Not Null	Should not be null
date	date	Not Null	Should not be null
flag	varchar(50)	Not Null	Should not be null

Table 3.15: Quiz-Questions

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Q-Id	varchar(20)	Primary Key	It stores unique Question Id
Category-Id	varchar(20)	Foreign Key	It refers to the table Sub-Category Details
Question	varchar(MAX)	Not Null	Should not be null
Option-A	varchar(500)	Not Null	Should not be null
Option-B	varchar(500)	Not Null	Should not be null
Option-C	varchar(500)	Not Null	Should not be null
Option-D	varchar(500)	Not Null	Should not be null
Answer	varchar(500)	Not Null	Should not be null
Description	varchar(MAX)	Null	Can be null
Created-By	varchar(100)	Null	Can be null
Created-On	date	Null	Can be null

Table 3.16: Quiz-Report

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Quiz-Id	varchar(20)	Primary Key	It stores unique Quiz Id
User-Id	varchar(20)	Foreign Key	It refers to the table User-Details
Date	date	Null	Can be null
SubCategory	varchar(MAX)	Foreign Key	It refers to the table SubCategory-Details
Start-Time	varchar(30)	Not Null	Should not be null
End-Time	varchar(30)	Not Null	Should not be null
Correct-Answer	int	Not Null	Should not be null
Incorrect-Answer	int	Not Null	Should not be null
Attended-Qstns	int	Not Null	Should not be null
Total-Score	int	Not Null	Should not be null
Percentage	numeric(2,2)	Null	Can be null

Table 3.17: Temporary-Questions

FIELDS	DATATYPE(SIZE)	CONSTRAINTS	DESCRIPTION
Qstn-No	int	Primary Key	It stores unique Question No
Qstn-Id	varchar(20)	Foreign Key	It refers to the table Quiz-Questions
Question	varchar(MAX)	Not Null	Should not be null
OptionA	varchar(MAX)	Not Null	Should not be null
OptionB	varchar(MAX)	Not Null	Should not be null
OptionC	varchar(MAX)	Not Null	Should not be null
OptionD	varchar(MAX)	Not Null	Should not be null
Answer	varchar(MAX)	Not Null	Should not be null
Selected-Option	varchar(MAX)	Not Null	Should not be null
Status	varchar(MAX)	Null	Can be null

3.2.6 Data Flow diagram

Data flow diagram is the graphical representation of the system. It is a network that uses special symbols to describe the flow of data and process that transforms data throughout the system. Data flow diagram is a way of representing system requirements in a graphic form. A DFD also known as Bubble Chart has the purpose of clarifying system requirements and identifies major transformations that will become program in system design. So it is the starting point of design phase that functionally decomposes the requirements specifications down to the lowest level of details. A DFD consist of series of bubbles joined by lines. The bubbles represent data transformation and the lines represent data flow in the system.

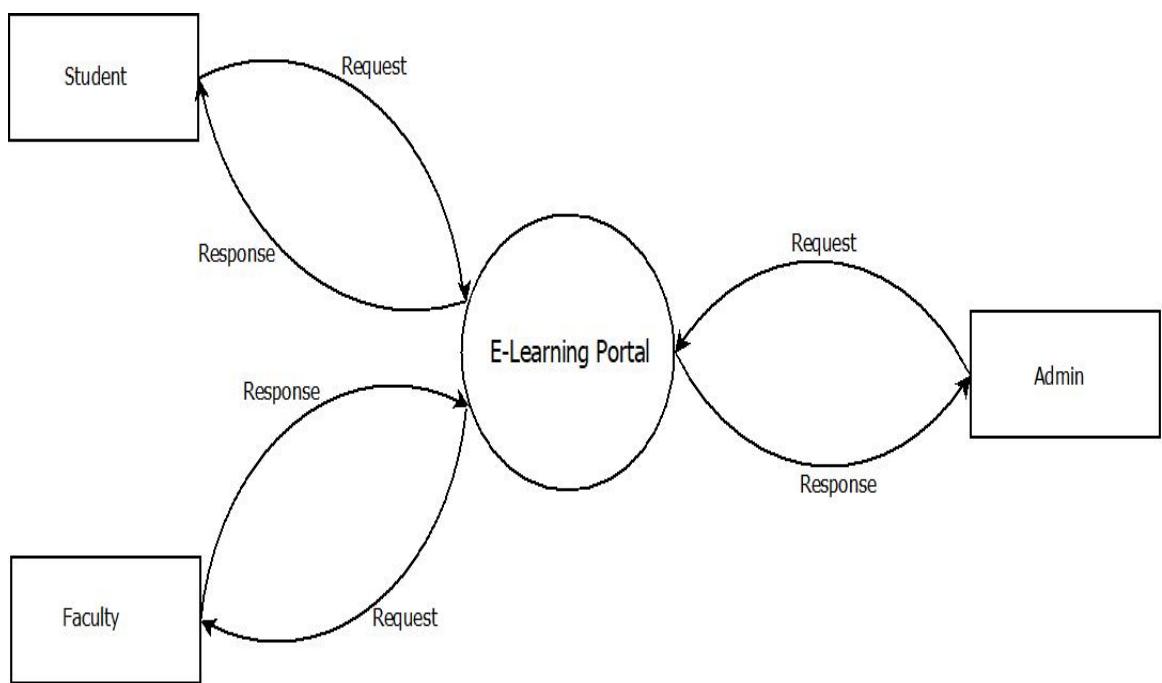


Fig. 3.1: LEVEL-0 DFD

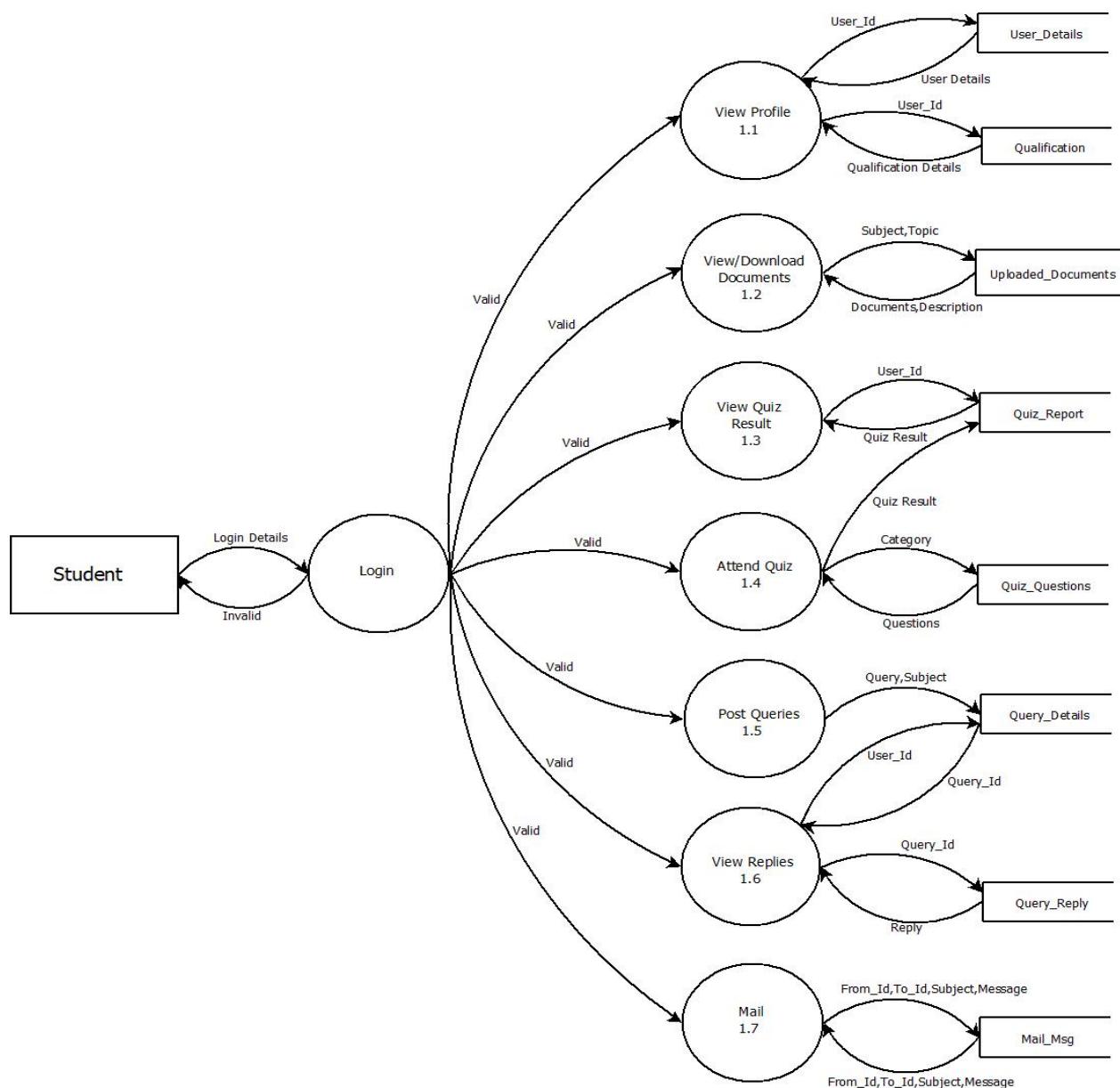


Fig. 3.2: LEVEL-1 DFD for Student

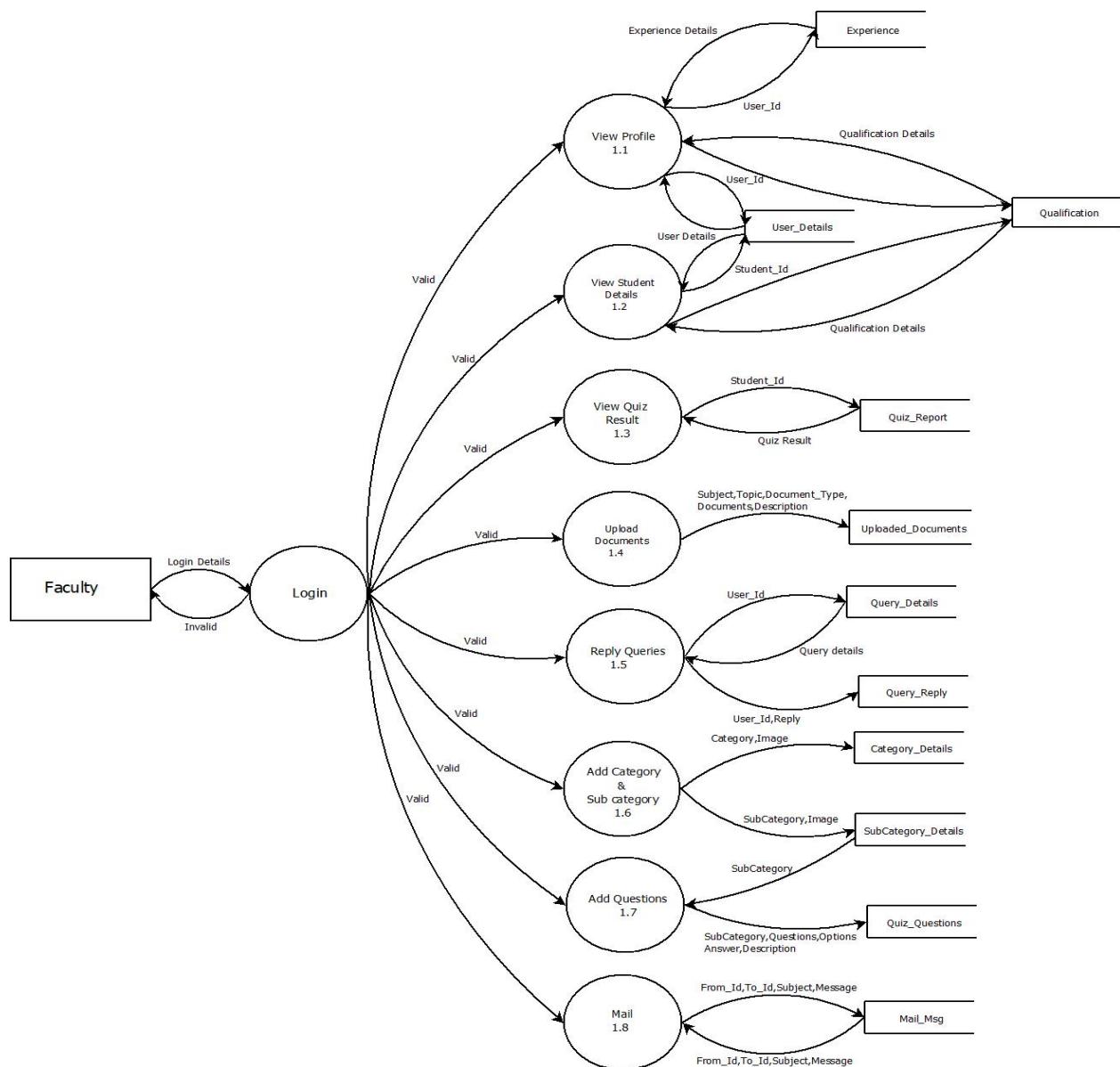


Fig. 3.3: LEVEL-1 DFD for Faculty

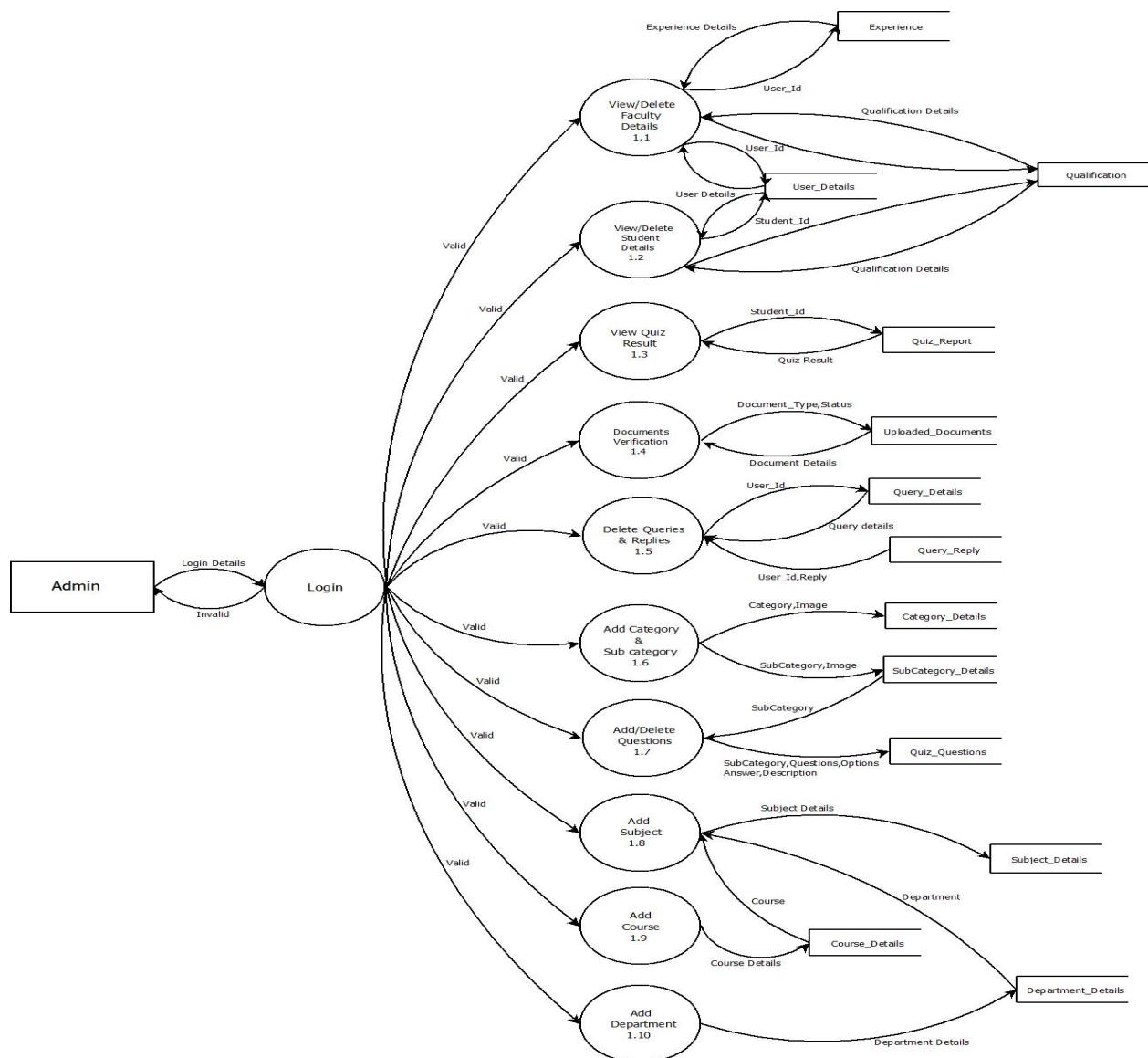


Fig. 3.4: LEVEL-1 DFD for Admin

3.3 UNIFIED MODELING LANGUAGE[UML]

UML is a way of visualizing a software program using a collection of diagrams. The notation has evolved from the work of Grady Booch, James Rumbaugh, Ivar Jacobson and the Rational Software Corporation to be used for object-oriented design, but it has since been extended to cover a wider variety of software engineering projects. Today, UML is accepted by the Object Management Group(OMG) as the standard for modelling software development.

UML stands for Unified Modeling Language. UML 2.0 helps extend the original UML specification to cover a wider portion of software development efforts including agile practices.

Improved integration between structural models like class diagrams and behavior models like activity diagrams. The original UML specified nine diagrams; UML 2.x brings that number up to 13.

The four new diagrams are called: communication diagram, composite diagram, interaction overview diagram and timing diagram. It also renamed state chart diagrams to state machine diagrams , also known as state diagrams.

Types of UML diagrams

The current UML standards call for 13 different types of diagrams: class, activity, object, use case, sequence, package, state, component, communication, composite structure, interaction overview, timing and deployment. These diagrams are organized into two distinct groups: structural diagrams and behavioral or interaction diagrams.

Structural UML diagrams

- Class diagram
- Package diagram
- Object diagram
- Component diagram
- Composite structure diagram
- Deployment diagram

Behavioral UML diagrams

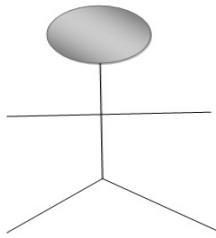
- Activity diagram
- Sequence diagram
- Use case diagram
- State diagram
- Communication diagram
- Interaction overview diagram
- Timing diagram

3.3.1 Use case Diagram

To model a system the most important aspect is capture the dynamic behaviour. To modify a bit in details, dynamic behaviour of the system when it is running or operating. So only behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. So use case diagram consists of actors, use case and their relationships. The diagram is used to model the system of an application. A single use case diagram captures a particular functionality of a system.

Use case Diagram objects:

- Actor
- Use case
- Use case diagram
- System
- Package

Actor

Actor is a use case diagram in an entity that performs a role in one given system. This could be a person, organization or an external system usually drawn like skeleton.

Use case

A use case represents a function or an action within the system. Its drawn as an oval and named with the function.

System

System is used to define the scope of the use case and drawn as a rectangle. This is an optional element but useful when you're visualizing large systems. For example you can create all the use cases and then use the system object to define the scope covered by your project. Or you can even use it to show the different areas covered in different releases.

Package

Package is another optional element that is extremely useful in complex diagrams. Similar to use class diagrams, packages are used to group together use cases.

The following is the UML diagram of this system :-

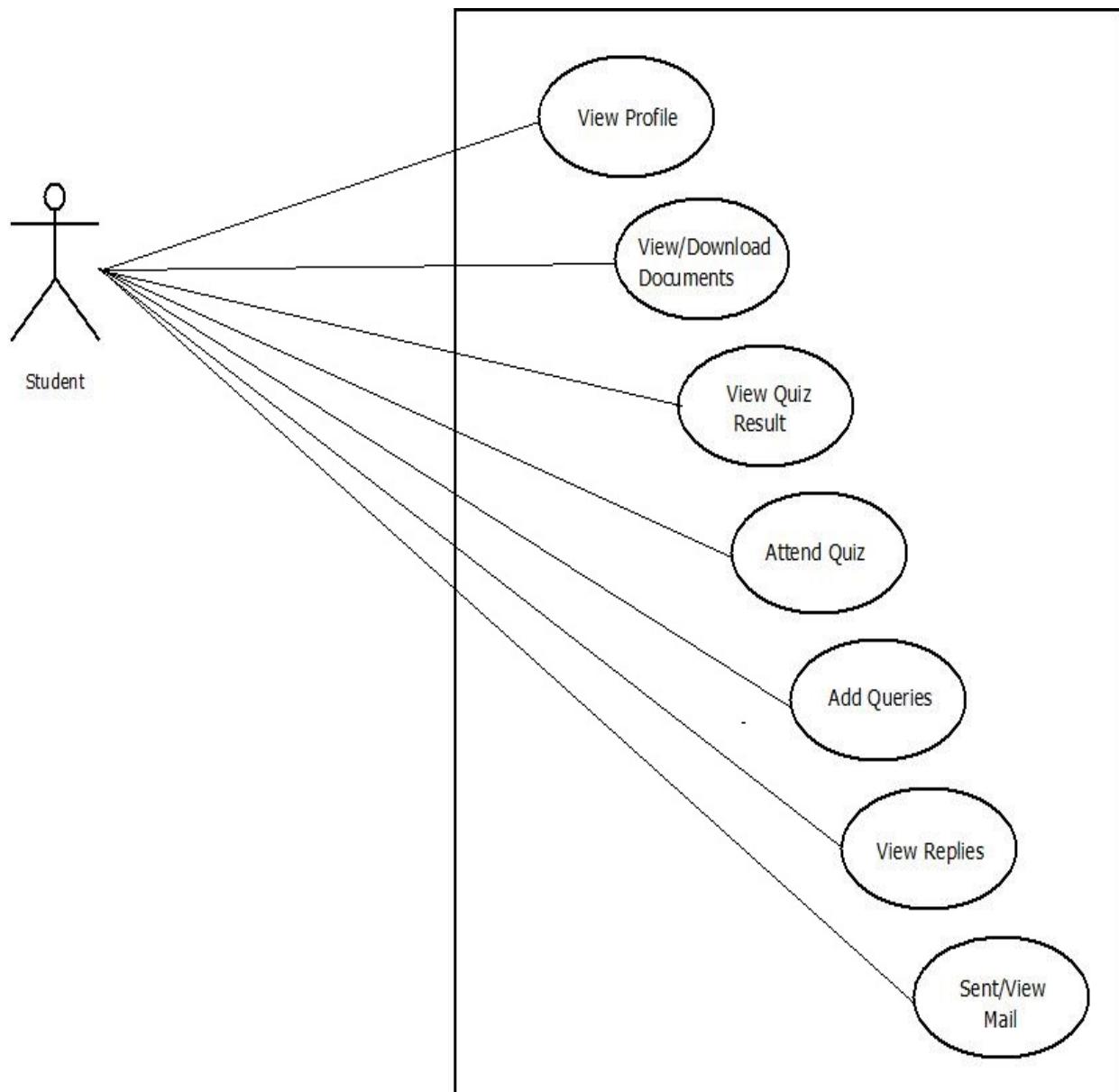


Fig. 3.5: Usecase Diagram for Student

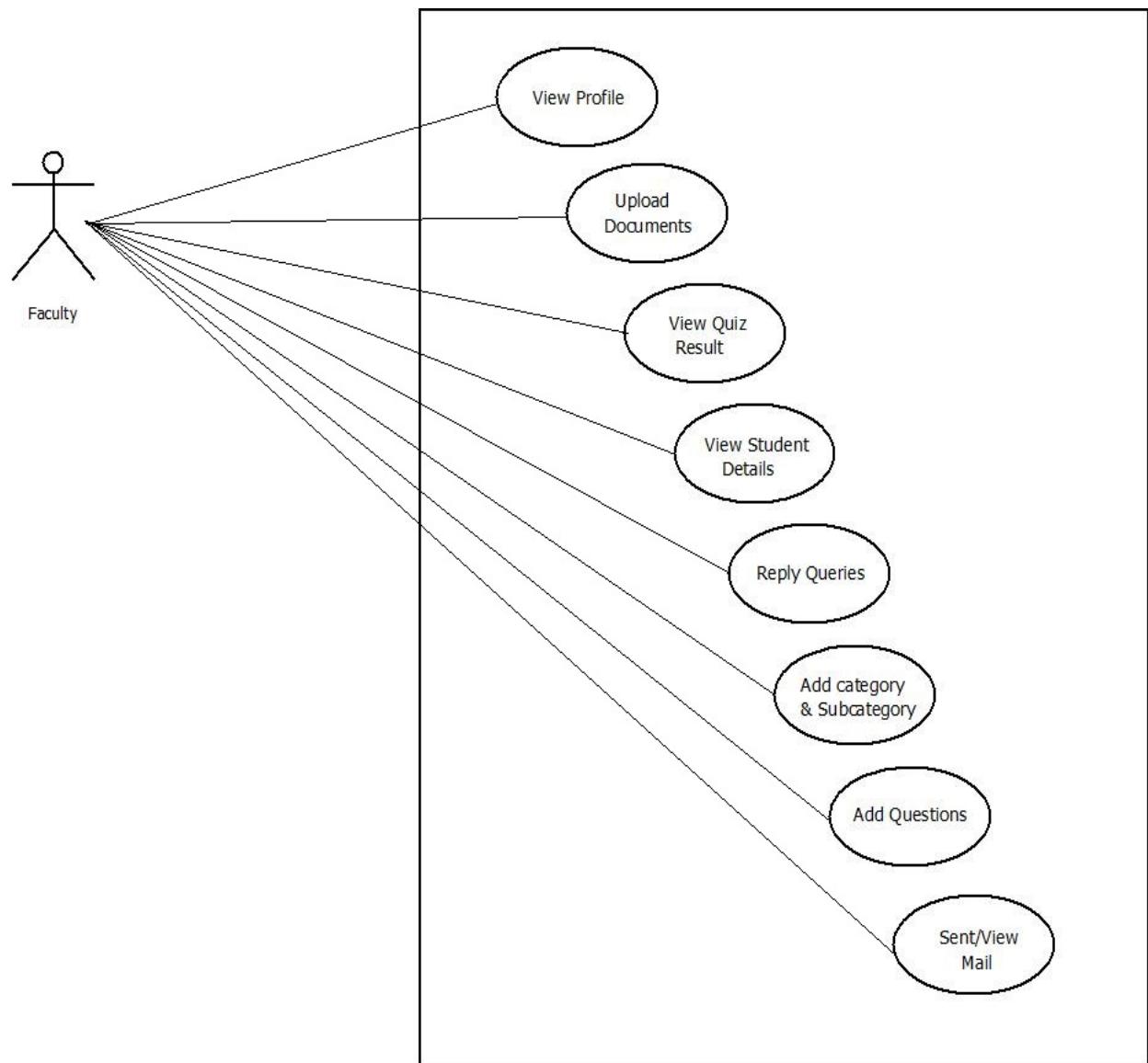


Fig. 3.6: Usecase Diagram for Faculty

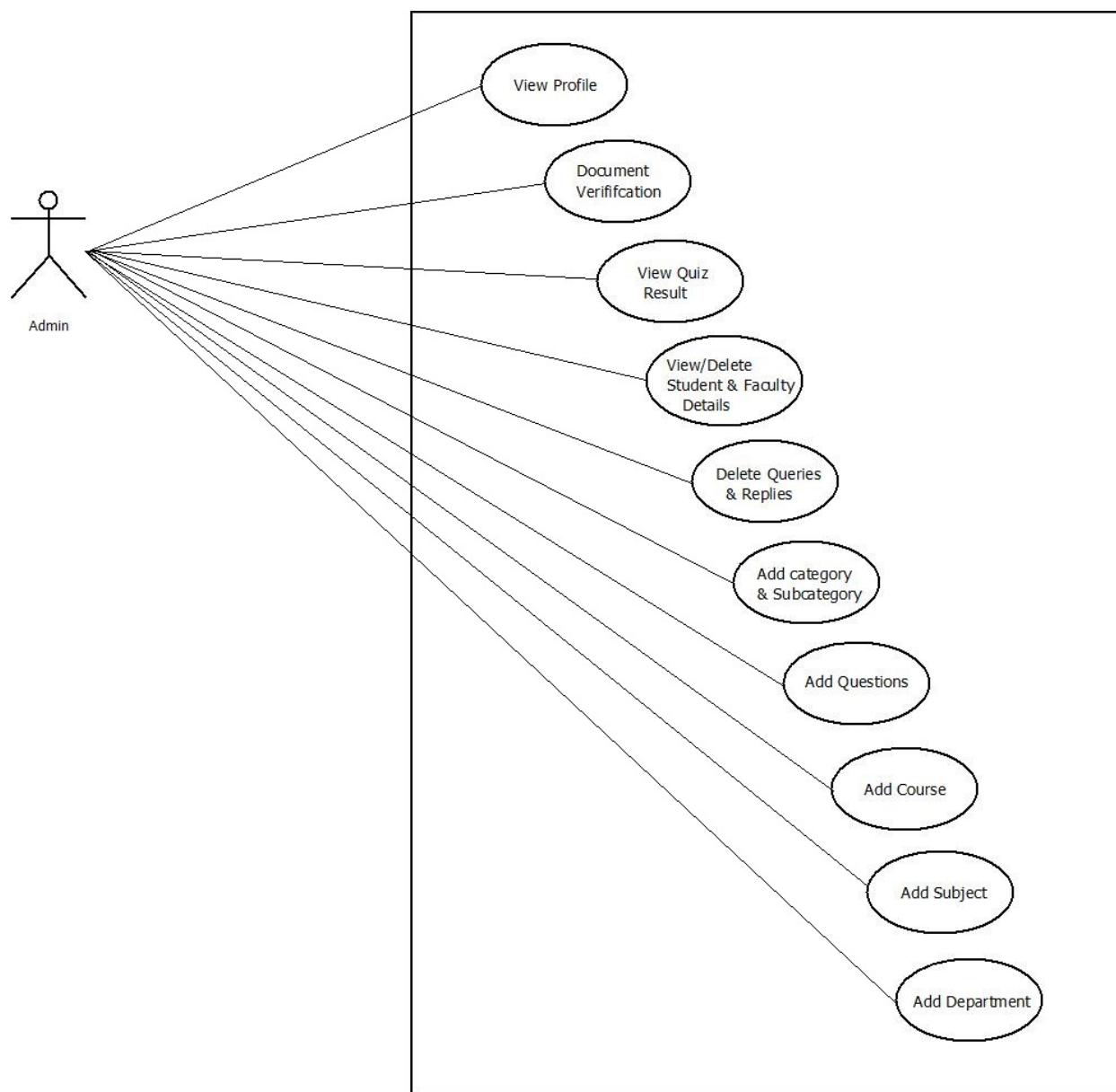


Fig. 3.7: Usecase Diagram for Admin

3.3.2 Sequence Diagram

UML sequence diagrams are used to represent or model the flow of messages, events and actions between the objects or components of a system. Time is represented in the vertical direction showing the sequence of interaction of the header elements.

Sequence Diagrams are used primarily to design, document and validate the architecture, interfaces and logic of the system by describing the sequence of actions that need to be performed to complete a task. UML sequence diagrams are useful design tools because they provide a dynamic view of the system behavior which can be difficult to extract from static diagrams or specifications.

Although UML sequence diagrams are typically used to describe object-oriented software systems, they are also extremely useful as system engineering tools to design system architectures in business process, as message sequence charts and call flows for telecoms or wireless system design, and for protocol stack design and analysis.

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence Diagrams are typically associated with use case realizations in the logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

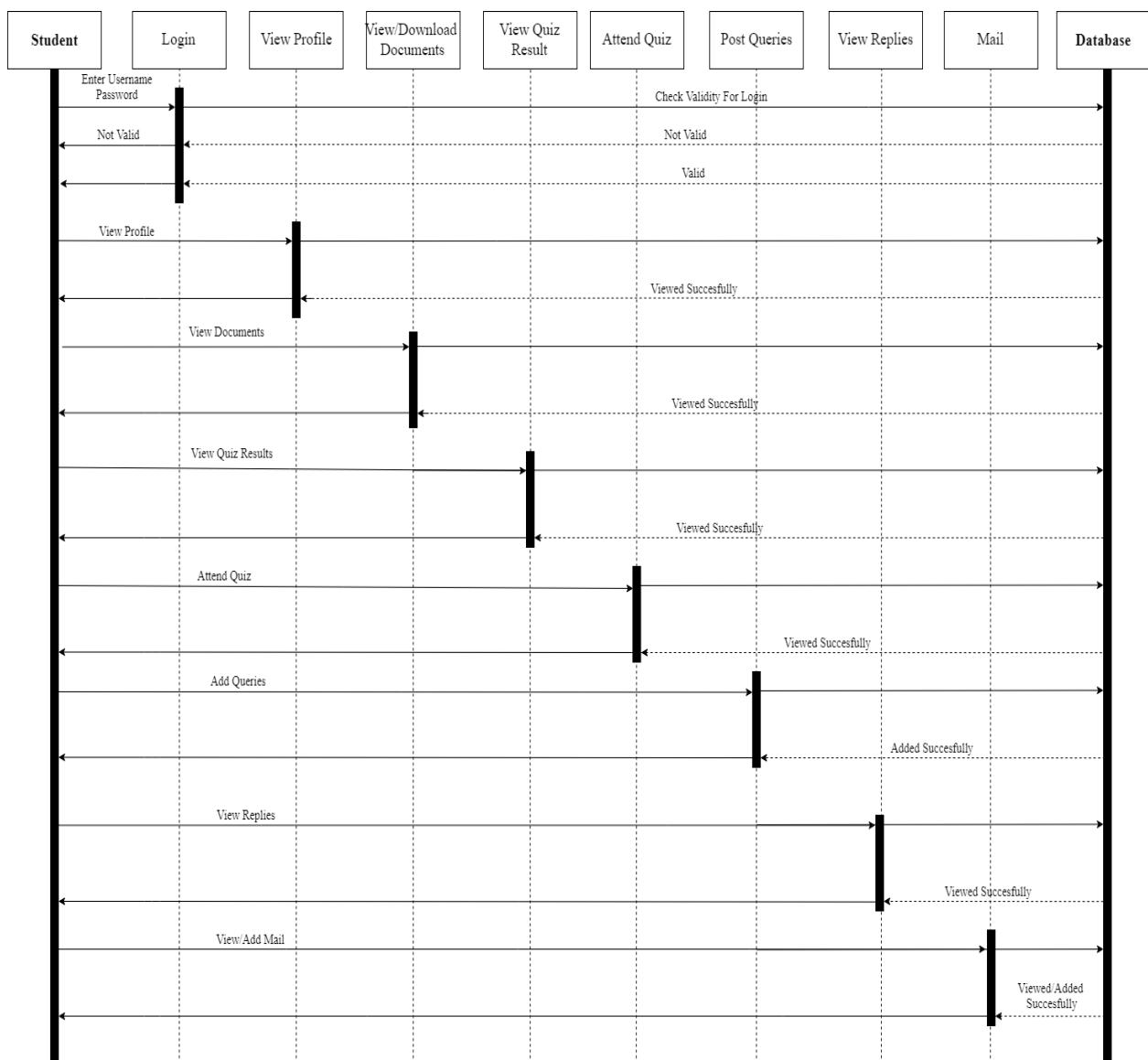


Fig. 3.8: Sequence Diagram for Student

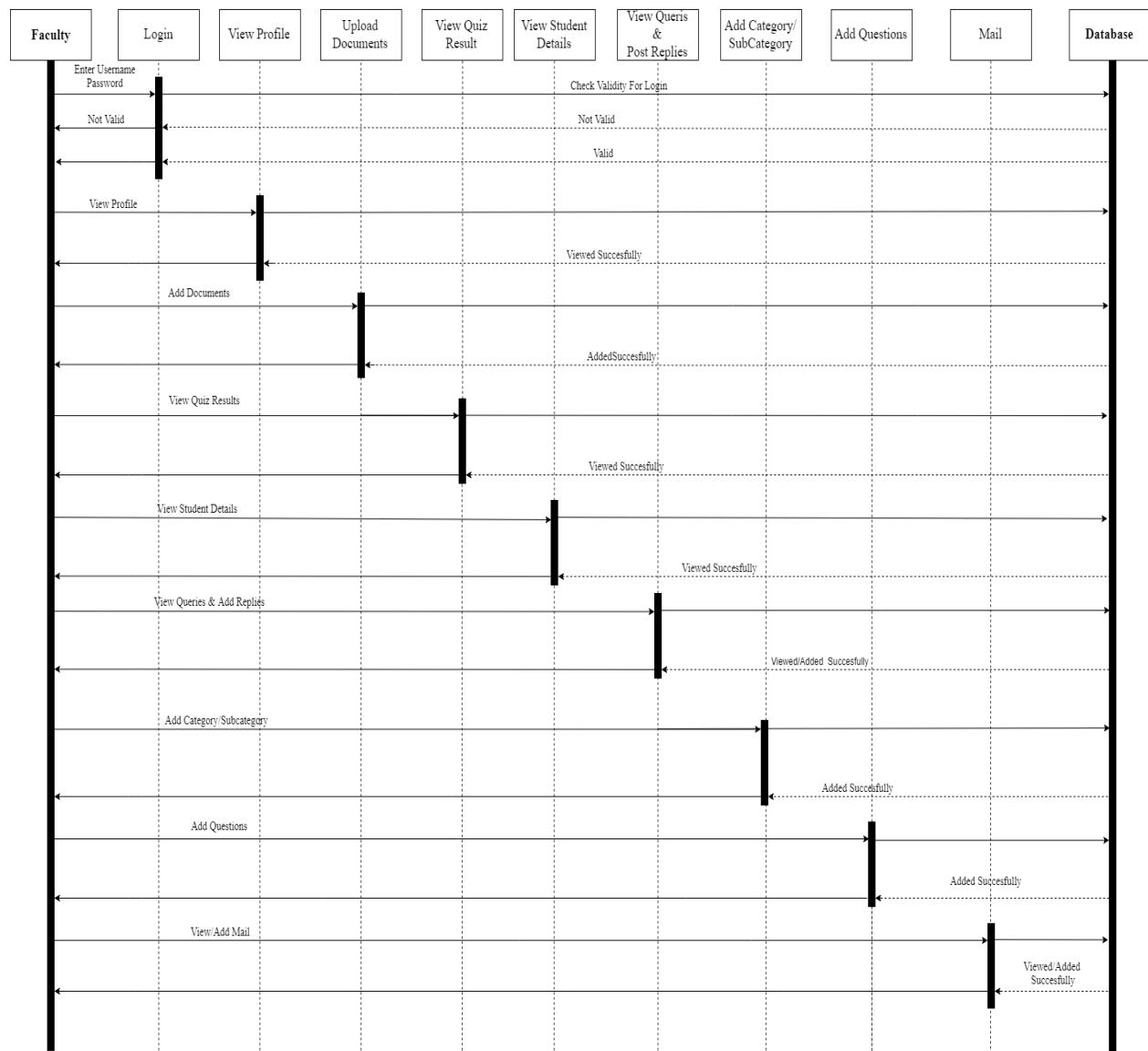


Fig. 3.9: Sequence Diagram for Faculty

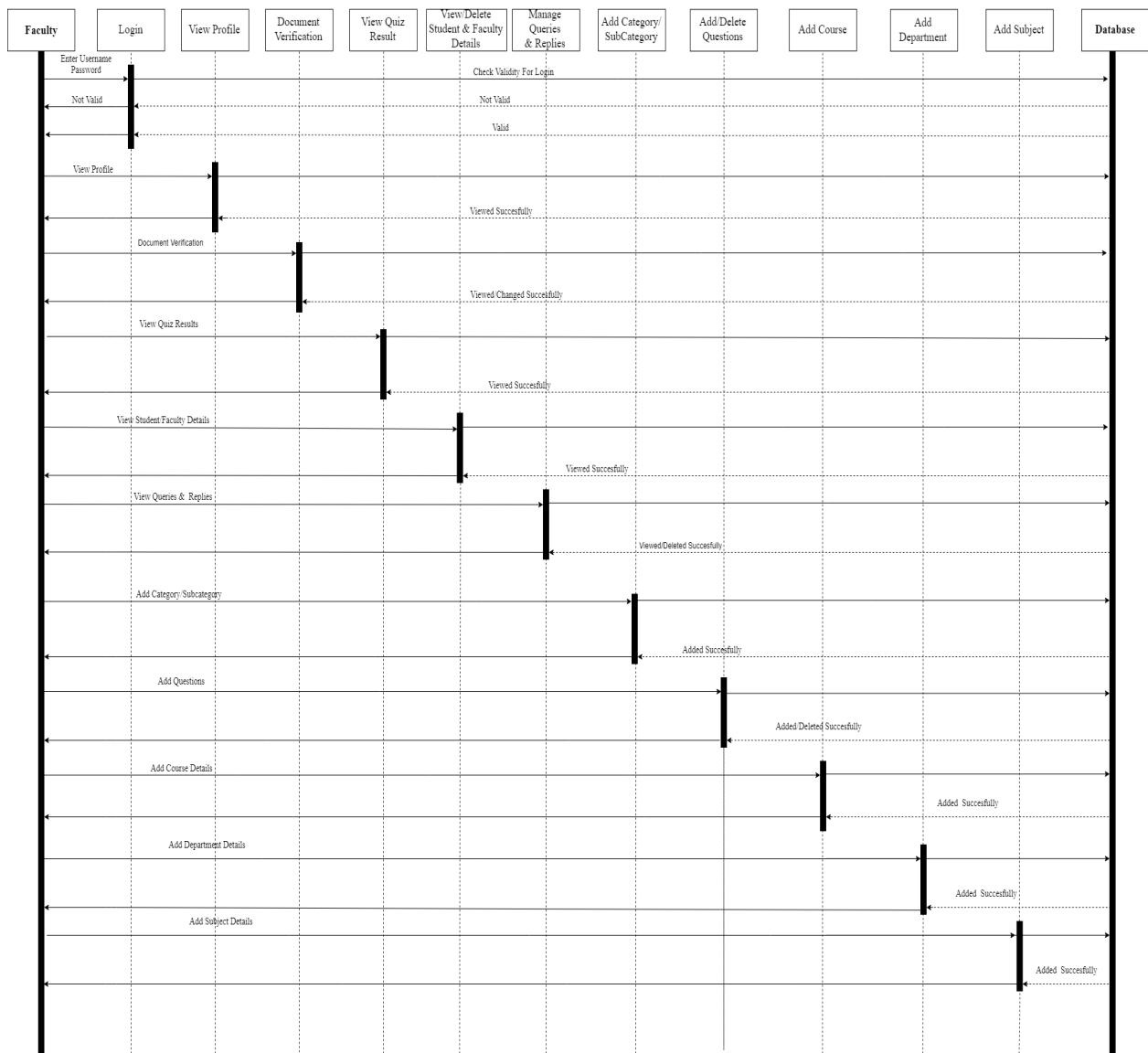


Fig. 3.10: Sequence Diagram for Admin

3.3.3 Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

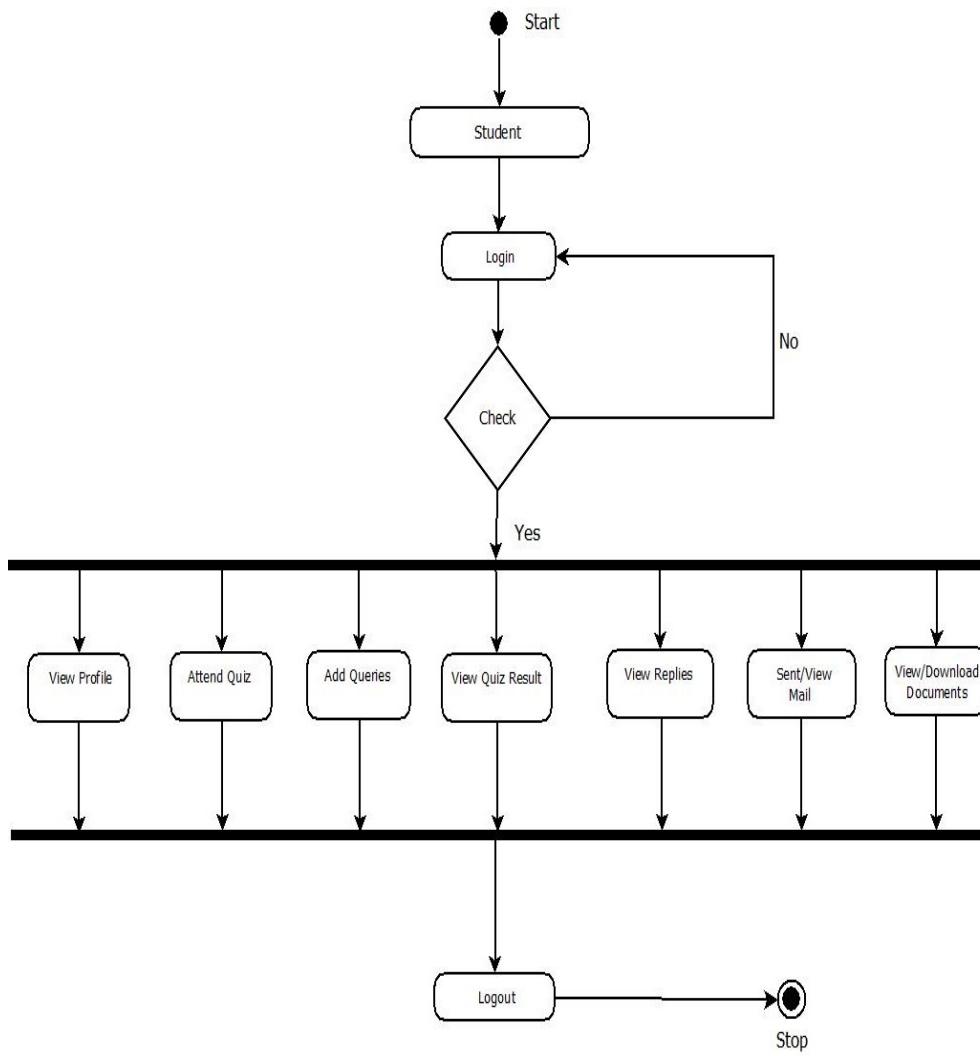


Fig. 3.11: Activity Diagram for Student

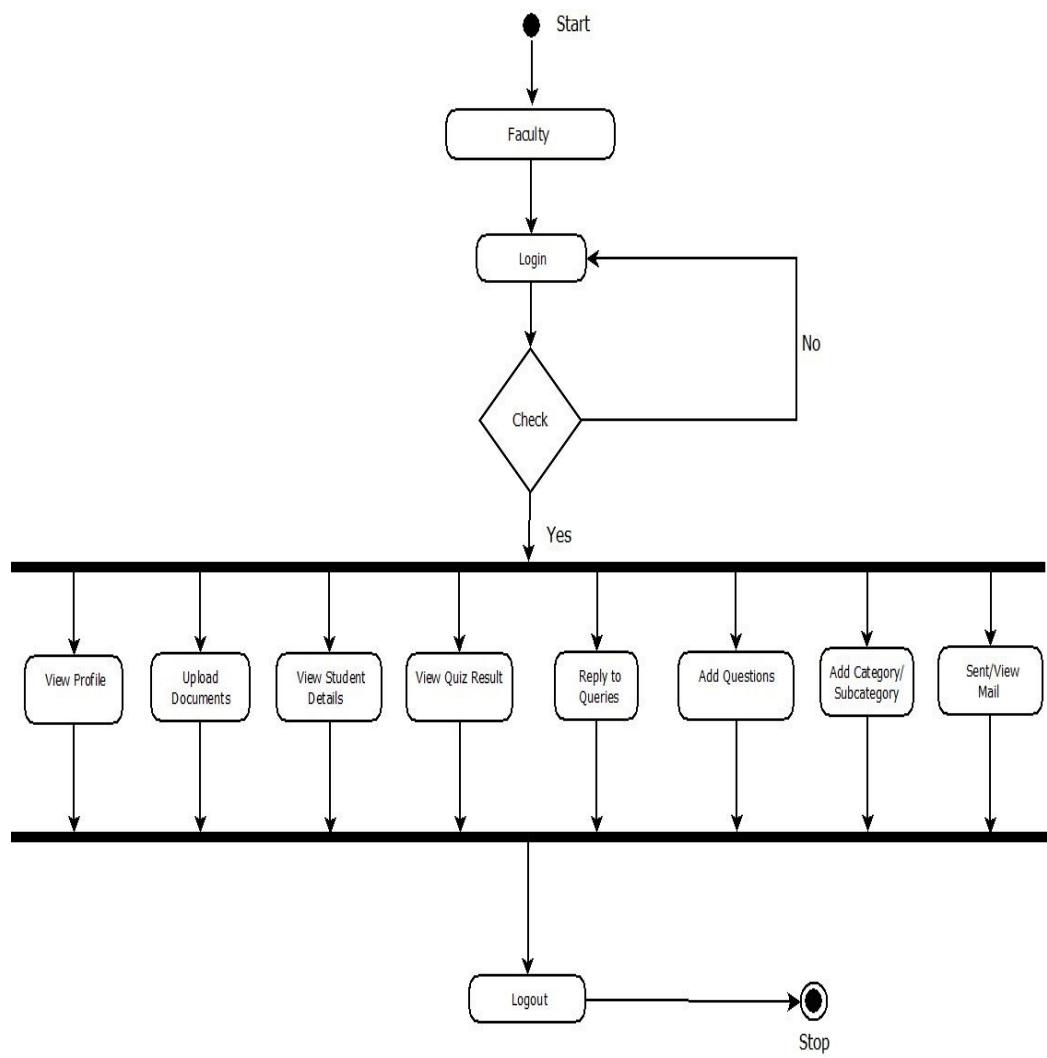


Fig. 3.12: Activity Diagram for Faculty

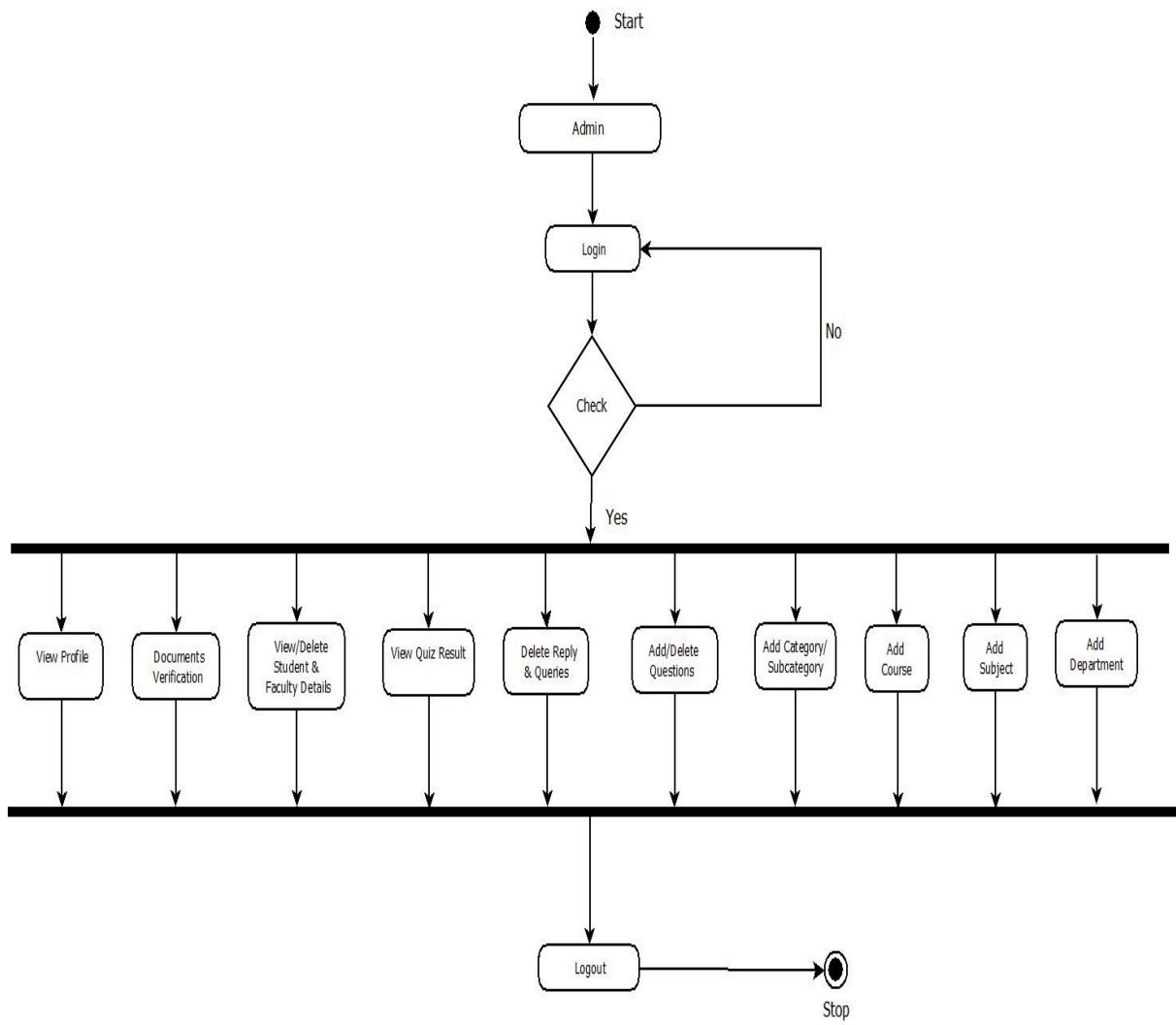


Fig. 3.13: Activity Diagram for Admin

3.4 TOOLS AND PLATFORMS

3.4.1 Introduction to Microsoft.Net

Microsoft .Net is the umbrella term for the Microsofts Strategy of to move from a client centric model to a network centric model. It can be best described as the initiative that will allow the Internet to the basic of a new operating system. It is free from the constraints of hardware by making user data available from the Internet. It is important to developer because it will change the way. They develop applications by allowing them to hook on web services. The vision of .NET is globally distributed system that use XML as the universal glue to allow functions running on different computers across the world to come together in a single application. In this vision, systems from servers to wireless palmtops, will share the same general platform, with versions of.NET available for all of them, and with each of them able to collaborate with others.

3.4.2 The .NET Platform

The .Net platform is the developers perspective in the views.NET as an amalgam of a set of services, specification, guidelines and tools for incorporating the .NET vision. It includes the .NET infrastructure and tools to build and operate a new generation of smart Internet devices. As of the developer .NET platform is something that helps to put the .NET vision into a reality and hence helps the developers in providing the user with the .NET experience.

3.4.3 The .NET Products

The .NET products will include a whole range of tools and servers that rely of on XML as a language to describe data and SOAP (Simple Object Access Protocols) as protocol for transmission of data between products. This includes Microsoft Windows.NET, MSN.NET, Personal subscription services, Microsoft visual studio.NET and Microsoft bcentric for .NET.

3.4.4 The .NET Services

The .NET services will include all the web services and other corporate services provided by the third party vendors. A vast range of partners and developers will have the opportunity to produce corporate and vertical services built on the .NET platform.

3.4.5 The .NET Framework

The .NET framework is an environment for building, deploying and running web services and other applications. Microsoft.NET framework is a standard that aims at integrating web application and services development to enable deployment and maintenance of HTTP and XML. It goes beyond development to enable deployment and maintenance of application and services along with handling their scalability and reliability.

3.4.6 Common Language Runtime (clr)

The .NET framework provides a runtime environment called the Common Language Runtime, Which manages the execution of code and provides services that make the deployment process easier. Compilers and tools expose the run times functionality and enable you to write code that benefits from this managed execution environment. The Common Language Runtime makes it easy to design components and applications whose objects interact across language. Objects written in different language can communicate with each other and their behaviors can be tightly integrated.

3.4.7 ASP.NET

Active Server Page is the server side technology for creating dynamic web pages. The technology basically used scripting interspersed with in HTML to generate information on the fly, whenever a client requested, an ASP page ,the script embedded in the HTML pages was interpreted by scripting engine on the web server and execute to generate HTML.

This HTML tags and displayed the output on the browser section. ASP.NET is server side web technology that can be used to develop dynamic and scalable web applications. These applications

cations can connect to database, interact with user and provide asynchronous services. It is used on a server to build powerful web applications. ASP.NET offers several important advantages over previous web development model.

3.4.8 SQL SERVER 2012

SQL Server is Database Management System (DBMS) developed and marketed by Microsoft. This system is the most important part of Microsoft back office an enterprise suite of client server application. The most important aspects of SQL SERVER are: - SQL Server is easy to use, SQL Server scales from a laptop to symmetric multiprocessor systems, SQL Server provides data warehousing feature that until now have only been available in oracle and other more expensive DBMS. It is relatively easy to manage through the use of a graphical computing environment for almost every task of system and database administration.

3.4.9 Normalization

Normalization refers how to implement the relationships and storage of data in the database tables. Keys are used to uniquely define a relationship to another instance or set of information. In first normal form each value in the database table is atomic or represented only once. In second form each instance or raw in the database table must be uniquely identifiable. The table in the third normal form wont have redundant non key information.

The SQL Server provides the ability to create roles so that security permission granted to all members is the same. Roles are defined on a database which means that when a role is created in database its not available in other. In standard role user are assigned to the role and the role is granted. Permission on database objects. This can be significantly reducing the number of needed logins to the server to only those users who need direct access to tables.

Microsoft SQL Server 2012 is the most advanced, trusted, and scalable data platform released to date. Building on the success of the original SQL Server 2012 release, SQL Server 2012 has made an impact an impact on organizations worldwide with its groundbreaking capabilities, empowering end users through self-services business intelligent(BI), bolstering efficiency and collaboration between database administrators(DBAs) and application developers, and scaling to accommodate

the most demanding data workloads.

3.5 GITHUB

Version control is a system that manages changes to a file or files. These changes are kept as logs in a history, with detailed information on what file(s) was changed, what was changed within the file, who changed it, and a message on why the change was made. This is extremely useful, especially when working in teams. To understand how incredibly powerful version control is How many files of different versions of a manuscript or thesis do you have laying around after getting feedback from your supervisor or co-authors?

Have you ever wanted to experiment with your code or your manuscript and need to make a new file so that the original is not touched ? Have you ever deleted something and wish you hadn't ? Have you ever forgotten what you were doing on a project ? All these problems are fixed by using version control (git)!

Git Hub History

In this project "**E-LEARNING PORTAL**", maintained a Git Hub Repository to store the whole project details to know about the changes made. <https://github.com/AswathySathyan95/E-Learning-System>

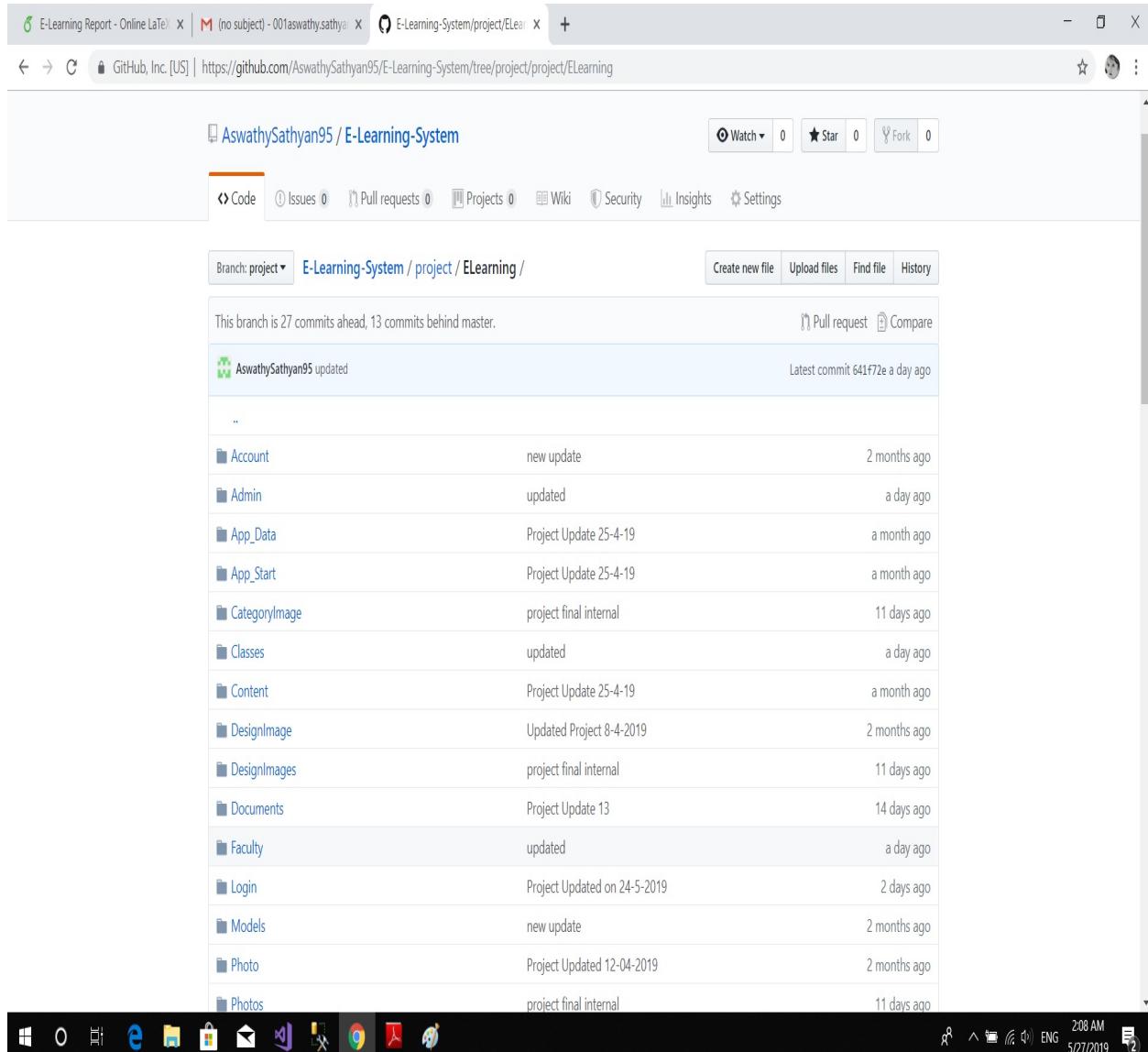


Fig. 3.14: Git History

Chapter 4

SYSTEM TESTING

4.1 TESTING METHODOLOGIES AND STRATEGIES

Software testing is an integral part of to ensure software quality, some software organizations are reluctant to include testing in their software cycle, because they are afraid of the high cost associated with the software testing .There are several factors that attribute the cost of software testing. Creating and maintaining large number of test cases is a time consuming process. Furthermore, it requires skilled and experienced testers to develop great quality test cases.

Even with the wide availability of automation tools for testing, the degree of automation mostly remains at the automated test script level and generally significant amount of human intervention is required in testing. In addition data collected, as testing is conducted provides a good indication of software quality as a while. The debugging process is the most unpredictable part of testing process. Testing begins at the module level and work towards the integration of entire computer based system. No testing is completed without verification and validation part.

The goal of verification and validation activities are to access and improve the quality of work products generated during the development and modification of the software. Testing plays a vital role in determining the reliability and efficiency of the software and hence is very important stage in software development. Tests are to be conducted on the software to evaluate its performance under a number of conditions. Ideally, it should do so at the level of each module and also when all of them are integrated to form the completed system.

4.1.1 Unit Testing

Here we test each module individually and integrated the overall system. Unit testing focuses verification efforts even in the smallest unit of software design in each module. This is known as module testing. The modules of the E-Learning Portal are tested separately. This testing is carried out in the programming style itself. In this testing each module is focused to work satisfactorily as regard to expected output from the module. There are some validation checks for the fields. Unit testing gives stress on the modules of E-Learning Portal independently of one another, to find errors. Different modules are tested against the specifications produced during the design of the modules. Unit testing is done to test the working of individual modules with test servers. Program unit is usually small enough that the programmer who developed it can test it in a great detail. Unit testing focuses first on that the modules to locate errors. These error are verified and corrected and so that the unit perfectly fits to the project.

4.1.2 Integration Testing

Data can be lost across an interface, one module can have an adverse effect on the other sub-functions, when combined they may not perform the desired functions. Integrated testing is the systematic testing to uncover the errors within the interface. This testing is done with simple data and the developed system has run successfully with this simple data. The need for integrated system is to find the overall system performance. The Modules of this project are connected and tested.

After splitting the programs into units, the units were tested together to see the defects between each module and function. It is testing to one or more modules or functions together with the intent of finding interface defects between the modules or functions. Testing completed at as part of unit or functional testing, integration testing can involve putting together of groups of modules and functions with the goal of completing and verifying meets the system requirements.

4.1.3 System Testing

System testing focuses on testing the system as a whole. System Testing is a crucial step in Quality Management Process. In the Software Development Life Cycle, System Testing is the first level where the System is tested as a whole. The System is tested to verify whether it meets the functional and technical requirements. The application/System is tested in an environment that closely resembles the production environment where the application will be finally deployed.

The perquisites for System Testing are:-

- All the components should have been successfully Unit Tested.
- All the components should have been successfully integrated.
- Testing should be completed in an environment closely resembling the production environment. When necessary iterations of System Testing are done in multiple environments.

4.1.4 User Acceptance Testing

The system was tested by a small client community to see if the program met the requirements defined the analysis stage. It was fond to be satisfactory. In this phase, the system is fully tested by the client community against the requirements defined in the analysis and design stages, corrections are made as required, and the production system is built. User acceptance of the system is key factor for success of the system.

4.2 TEST CASES

S.No	Page	Test Step	Expected Result	Actual Result	Status	Remarks (if any)
1	Login page	Click on the Login button without entering user name or User Name" and "Please Enter Password" should appear.	Messages like "Please enter without entering user name or User Name" and "Please Enter Password" should appear.	Enter Password" appear.	Pass	None
		Enter a non-existing user name password and click on the Login button	Message like "Invalid User Name" should appear	A message "Invalid User Name" appears	Pass	None
		Enter a valid user name but wrong password and click on the Login button	Message like "Wrong Password" should appear	A message "Wrong Password" appears	Pass	None
		Enter a valid user name and password and click on the Login button	The page should be navigated to the home page	The page is navigated to the home page	Pass	None
2 Registration	Registration	Click on Register button without entering the mandatory fields.	Messages like "Please enter the details" or "Mandatory fields" should appear.	Messages like "Please enter the details" or "Mandatory fields" appear.	Pass	None
		Enter an invalid name.	Message "Please enter a valid name" should appear.	Message shown.	Pass	None
		Enter an invalid Email id.	Message like "Please enter a valid email id" should appear.	Message shown.	Pass	None
		Enter a invalid Mobile No.	Message "Please enter a valid mobile no" should appear.	Message shown.	Pass	None
		Upload a photo which is not in the specified format or size.	Message "Please upload an image of specified size" should appear.	Message shown.	Pass	None
		When the religion is selected	Corresponding to the selected religion category should be loaded in the drop down list.	Category corresponding to the selected religion is loaded in the drop down list.	Pass	None
		When state is selected	Districts corresponding to the selected state should be loaded in the drop down list.	Districts corresponding to the selected state is loaded in the drop down list.	Pass	None
		Enter an invalid pin code.	Message "Please enter a valid pin code(Only digits are allowed)" should appear.	Message shown.	Pass	None
		When an admitted program is selected	Branches corresponding to the selected program should appear in the drop down list.	Message shown.	Pass	None
		Enter an invalid mark	Message "Please enter a valid mark(Only digits and dot is allowed)" should appear.	Message shown.	Pass	None

		Enter all valid details and click on Register button.	Should generate a username and password which must be send to the user's mobile no. And save all details to the database and reload the page.	Generates a username and password which is sended to the user's mobile no and all datas are saved to database and the page is reloaded.	Pass	None
		Click on Clear button	Should clear all the fields	Clear all the fields	Pass	None
3 Forgot password	Enter an invalid email id	Message "Please enter a valid email id" should appear. Message "This email id has not been registered. Please enter correct email id" should appear	Message appears.	Pass	None	
	Enter an email id which is not already registered.	Message "Please check your email" should appear and navigate to the login page.	Message appears and navigated to the login page.	Pass	None	
	Enter valid email id and click on submit button	Message "Incorrect Password!Please enter correct password" should appear.	Message appears and navigated to the login page.	Pass	None	
4 Change Password	Enter wrong password	Message shown	Pass	None		
5 Update quiz questions	Enter an invalid new password	Message "Please enter a valid password. Password must contain minimum 8 characters,atleast one uppercase character,one digit and any of the special character(@,_/,#) should appear.	Message shown	Pass	None	
	Enter password in the confirm password field which is not same as the password in the field new password.	Message "Password doesn't match..Please enter correct password.." should appear.	Message shown	Pass	None	
	Enter all valid details	Message "Your password has been successfully changed" and navigate to login page.	Message shown and navigated to login page	Pass	None	
5 Update quiz questions	When correct answer is entered	Should check whether the correct answer is there in the options. If it is not message "Please enter correct answer" should appear Message "Question added successfully" should appear	Message shown	Pass	None	
	Enter all valid details	Message shown	Pass	None		

Fig. 4.1: Validation Testing

S.No	Page	Test Step	Expected Result	Actual Result	Status	Remarks (if any)
1	Login page	Enter valid username and password	Should validate the user and provide access to user accounts	Got the entry to user accounts	Pass	None
	Login page	Enter invalid username or password	Should flash an error message	Error message shown	Pass	None
	Login page	Enter invalid category	Should flash an error message	Error message shown	Pass	None
2	Registration	Submit registration request without entering Name.	Message "Please Enter Name" should appear.	Error message shown	Pass	None
		Submit registration request without entering Email ID.	Message "Please Enter Email ID*" should appear.	Error message shown	Pass	None
	Registration	Submit registration request without entering Mobile No.	Message "Please Enter your Mobile No" should appear.	Error message shown	Pass	None
		Submit registration request without entering Gender.	Message "Please Enter Gender" should appear.	Error message shown	Pass	None
	Registration	Submit registration request without entering Date of birth.	Message "Please Enter Your Date of birth" should appear.	Error message shown	Pass	None
		Submit registration request without entering Address.	Message "Please enter your address" should appear.	Error message shown	Pass	None
	Registration	Submit registration request without entering SSLC Details	Message "Please enter your SSLC details" should appear.	Error message shown	Pass	None
		Submit registration request without uploading photo	Message "Please upload your photo" should appear.	Error message shown	Pass	None
	Registration	Submit registration details without entering HSE Details	Message "Please enter your HSE details" should appear.	Error message shown	Pass	None
3	Upload Documents	Submit registration details will all valid details	Message "Registration Successfully Completed" should appear.	Message shown	Pass	None
		Click on save button without entering subject	Message "Please enter subject" should appear	Error message shown	Pass	None
		Click on save button without entering topic	Message "Please enter topic" should appear	Error message shown	Pass	None
		Click on save button without uploading document	Message "Please upload any documents" should appear	Error message shown	Pass	None
		Enter all valid details	Message "Documents Successfully uploaded" should appear	Error message shown	Pass	None
4	Add Quiz Category	Click on save button without entering category	Message "Please enter category" should appear	Error message shown	Pass	None
		Click on save button without entering Subcategory	Message "Please enter Subcategory" should appear	Error message shown	Pass	None
		Enter all valid details	Message "Details added successfully" should appear	Error message shown	Pass	None
5	Add Quiz Questions	Submit without entering the question	Message "Please enter Question" should appear	Error message shown	Pass	None
		Submit without filling any of the 4 options	Message "Please add Options" should appear	Error message shown	Pass	None
		Submit without selecting category	Message "Please select category" should appear	Error message shown	Pass	None
		Enter all valid details	Message "Successfully added" should appear	Error message shown	Pass	None
6	Forgot Password	Submit without entering email id	Message "Please enter email id" should appear	Error message shown	Pass	None
		Enter valid email id	Message "Please check your email id" should appear	Error message shown	Pass	None

Fig. 4.2: Unit Testing

Chapter 5

SYSTEM IMPLEMENTATION

The implementation is one phase of software development. Implementation is that stage in the project where theoretical design is turned into working system. Implementation involves placing the complete and tested software system into actual work environment. Implementation is concerned with translating design specification with source code. The primary goal of implementation is to write the source code to its specification that can be achieved by making the source code clear and straight forward as possible. Implementation means the process of converting a new or revised system design into operational one. The three types of implementation are:-implementation of a computerized system to replace a manual system, implementation of a new system to replace existing one and implementation of a modified system to replace an existing one.

The implementation is the final stage and it is an important phase. It involves the individual programming ; system testing, user training , and the operational running of developed proposed system that constitute the application subsystem. The implementation phase of the software development is concerned with translating design specification in the source code. The user tests the developed system and the changes are according to the needs. Before implementation, Several tests have been conducted to ensure no errors encountered during the operation. The implementation phase ends with an evaluation of the system after placing it into operation of time. The validity and proper functionality of all the modules of the developed application is assured during the process of implementation. Implementation is the process of assuring that the information system is operational and then allowing user to take over its operation for use and evaluation. Implementation is the stage in the project where the theoretical design is turned into a working system. The implementation phase constructs ,installs and operated the new system. The most crucial stage in achieving a new successful system is that it works effectively and efficiently.

Chapter 6

CONCLUSION

The project titled "**E-LEARNING PORTAL**", this system has developed for sharing study materials in a college. The project is developed as a Web Application by using ASP.NET as the front end and SQL Server 2012 as the back end. Each user of the system has role and permission according to their role. This system helps faculties to upload all the study materials like E-Notes, Reports and Video Tutorials. This system helps Students to get all the previous year documents at a single place.

E-resources serve as a platform for random access to multiple users at the same time and save plenty of time. E-learning has developed very fast due to internet. This project supports teaching and learning using a computer web technology by sharing documents like notes, reports or video tutorials. Students can post their queries and they can view replies provided by different Faculty. This project helps Students to improve their aptitude skills by attending online quiz.

REFERENCES

1. Marino Posadas,Mastering C sharp and .NET Framework ,Paperback,2016
2. William Penberthy,Beginning ASP.NET for Visual Studio,2015
3. Rob Miles,C sharp Programming Yellow Book,Cheese,2016
4. <https://www.w3schools.com/asp/default.asp>
5. <https://www.c-sharpcorner.com>
6. <http://www.wikipedia.org.>

Appendix A

APPENDICES

A.1 SCREEN SHOTS INPUT FORM,OUTPUT FORMS



Fig. A.1: Home Page

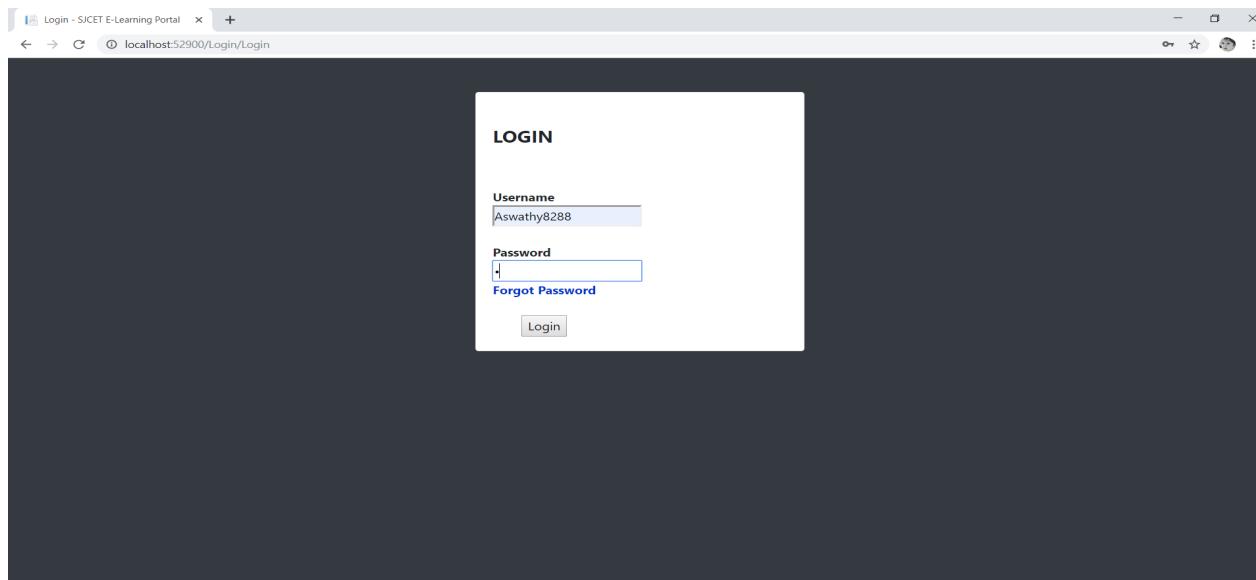


Fig. A.2: Login Page

Admin Functionalites :-

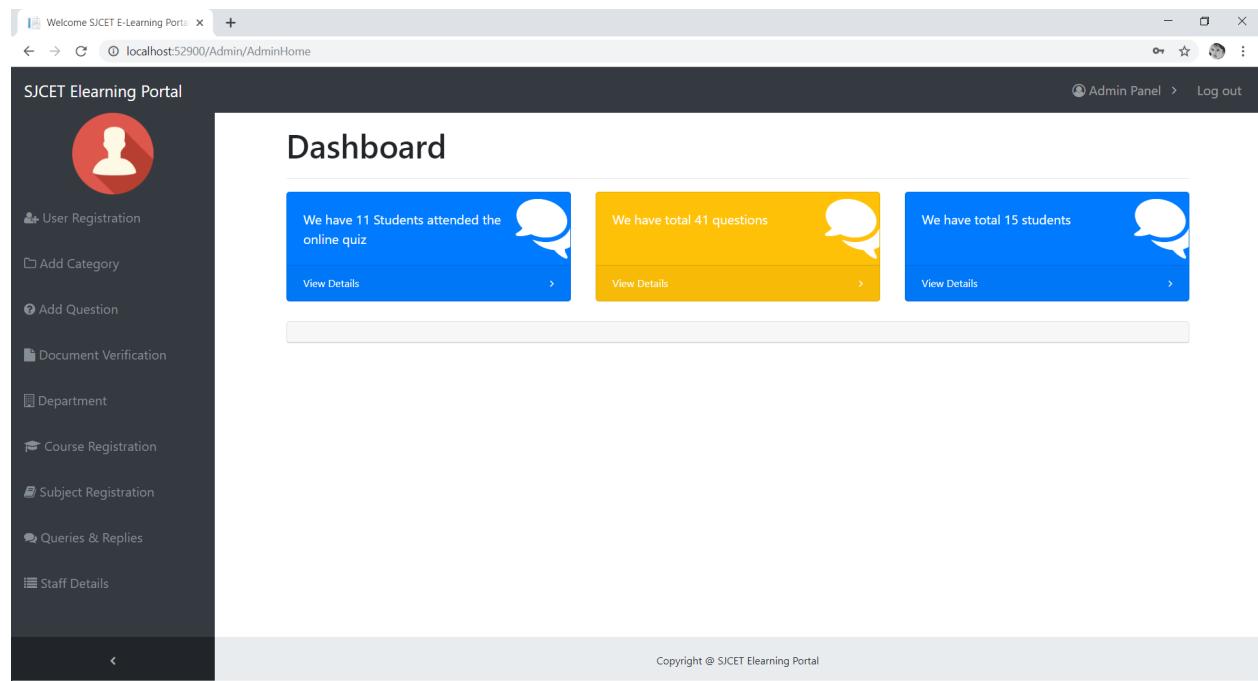


Fig. A.3: Admin Home Page

The screenshot shows the 'Document Verification' section of the admin portal. On the left is a sidebar with various management options like User Registration, Add Category, Add Question, etc. The main area displays a table of documents with columns for Subject, Topic, Document Description, View Document, and Verification status (all marked as 'NOT VERIFIED').

Subject	Topic	Document Description	View Document	Verification
Discrete Mathematics	fhf	cffcc	> View	NOT VERIFIED
Statistics	dfgh	zsxdcfg	> View	NOT VERIFIED
Statistics	xxx		> View	NOT VERIFIED

Fig. A.4: Document Verification by Admin

The screenshot shows the 'Staff Details' section. It includes a dropdown for 'Department' set to 'Master of Computer Application'. Below it is a table showing staff details with columns for Student Id, Name, and View Details.

Student Id	Name	View Details
Fcly4	ammu s	> View Details

Fig. A.5: View/Delete Staff Details

The screenshot shows the 'Staff Registration' form within the SJCET Elearning Portal. The left sidebar contains navigation links for User Registration, Add Category, Add Question, Document Verification, Department, Course Registration, Subject Registration, Queries & Replies, and Staff Details. The main content area has a title 'Staff Registration' and a note 'All Fields marked with * are Mandatory'. It includes sections for 'Personal Details' (First Name, Middle Name, Last Name, Gender, Date of Birth, Religion, Nationality, Upload Photo), 'Address' (House No./House Name, City, State, Street, District, Pin Code), and 'Admission Details' (Date of Joining, Department). All input fields marked with an asterisk (*) are mandatory.

Fig. A.6: User Registration

The screenshot shows the 'QUIZ CATEGORY' addition form. The left sidebar is identical to Fig. A.6. The main content area has a title 'QUIZ CATEGORY'. It includes fields for 'Category' (Aptitude) and 'Category Image' (Choose file: aptitudetest.png). There is a 'Save Details' button and a link 'Click To Add SubCategory'. Below this, there is a 'Sub Category' section with 'Category' (General Aptitude) and 'SubCategory' (Quantitative) dropdown menus. The footer includes a copyright notice: 'Copyright © SJCET Elearning Portal'.

Fig. A.7: Add Category

The screenshot shows the 'Add Questions' section of the admin panel. On the left sidebar, there are various navigation links: User Registration, Add Category, Add Question, Document Verification, Department, Course Registration, Subject Registration, Queries & Replies, and Staff Details. The main content area has a title 'QUIZ QUESTIONS'. It includes fields for 'Question Id' (Qstn42), 'Category' (Programming & Technical), 'Sub Category' (C Programming), 'Question' (First operating system designed using C programming language.), 'Options' (with four options: DOS, Option C, UNIX, and MAC), 'Correct Answer' (radio buttons for DOS, Windows, UNIX, and MAC), and a 'Description' box containing the text: 'UNIX. C actually invented to write an operation system called UNIX. By 1973 the entire UNIX OS is designed using C|'. At the top right of the main content area, there are 'Admin Panel' and 'Log out' links.

Fig. A.8: Add Questions

The screenshot shows the 'Quiz Reports' section of the admin panel. The left sidebar contains the same navigation links as Fig. A.8. The main content area has a title 'Quiz Reports'. It includes dropdown menus for 'Branch' (MCA_Regular) and 'Semester' (6). Below these are two tables. The first table shows student quiz results for 'Stud1' across five quizzes (Quiz1, Quiz10, Quiz11, Quiz2, Quiz3) with details like date, category, start/end time, and scores. The second table is a summary table with columns: Quiz_Id, Student_Id, Date, Quiz_Category, Start_Time, End_Time, No Of Correct Answer, No Of Incorrect Answer, No of Questions Attended, and Quiz Score. The summary table shows values such as 123 for the total number of questions attended and 11 for the total quiz score. At the bottom of the page, there is a copyright notice: 'Copyright © SJCET Elearning Portal'.

Quiz_Id	Student_Id	Date	Quiz_Category	Start_Time	End_Time	No Of Correct Answer	No Of Incorrect Answer	No of Questions Attended	Quiz Score
Quiz1	Stud1	5/11/2019	SCTgry8	May 11 2019 3:02PM	May 11 2019 3:03PM	1	3	10	1
Quiz10	Stud1	5/13/2019	SCTgry8	May 13 2019 10:11AM	May 13 2019 10:11AM	0	0	25	0
Quiz11	Stud1	5/21/2019	SCTgry8	May 21 2019 2:46PM	May 21 2019 2:47PM	3	6	25	3
Quiz2	Stud1	5/11/2019	SCTgry8	May 11 2019 3:07PM	May 11 2019 3:08PM	5	5	10	5
Quiz3	Stud1	5/11/2019	SCTgry8	May 11 2019 4:26PM	May 11 2019 4:28PM	11	8	25	11

Fig. A.9: View Student Quiz Results

The screenshot shows a web browser window titled "Welcome SJCET E-Learning Portal" with the URL "localhost:52900/Admin/Total_Questions". The main content area is titled "Quiz Questions" and displays a table of quiz questions categorized under "Verbal Ability". The table has columns for Question Id, Question, Option A, Option B, Option C, Option D, Answer, Description, and Delete. Three rows of data are visible:

Question Id	Question	Option A	Option B	Option C	Option D	Answer	Description	Delete
Qstn32	Choose the word which is different from the rest.	Cap	Turban	Helmet	Veil	Veil	All except Veil cover the head, while Veil covers the face	
Qstn34	Choose the word which is different from the rest.	Curd	Butter	Oil	Cheese	Oil	All except Oil are products obtained from milk.	
Qstn38	Choose the word which is most nearly the SAME in meaning as the word given ARDUOUS	Hazardous	Difficult	Different	Pleasurable	Difficult	The opening sentence introduces us to the	

The left sidebar contains links for User Registration, Add Category, Add Question, Document Verification, Department, Course Registration, Subject Registration, Queries & Replies, and Staff Details. The top right corner shows "Admin Panel" and "Log out". The footer of the page includes a copyright notice: "Copyright © SJCET Elearning Portal".

Fig. A.10: View/Delete Quiz Questions

Students Functionalites :-

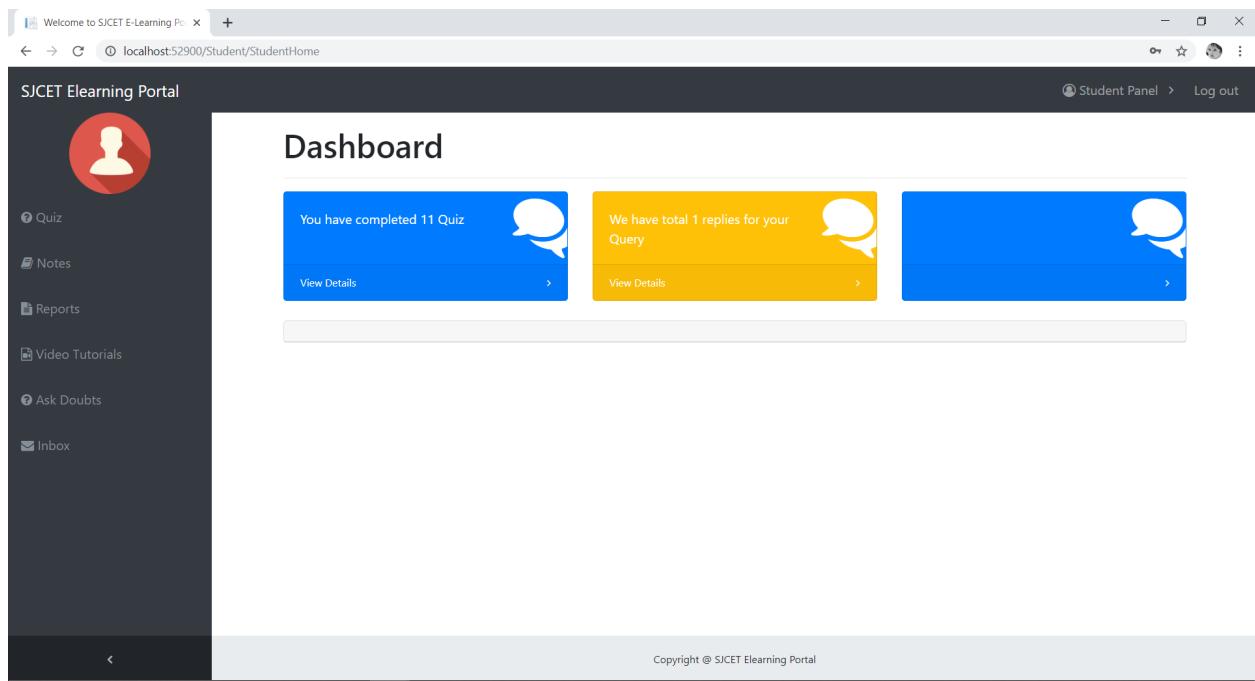


Fig. A.11: Student Home Page

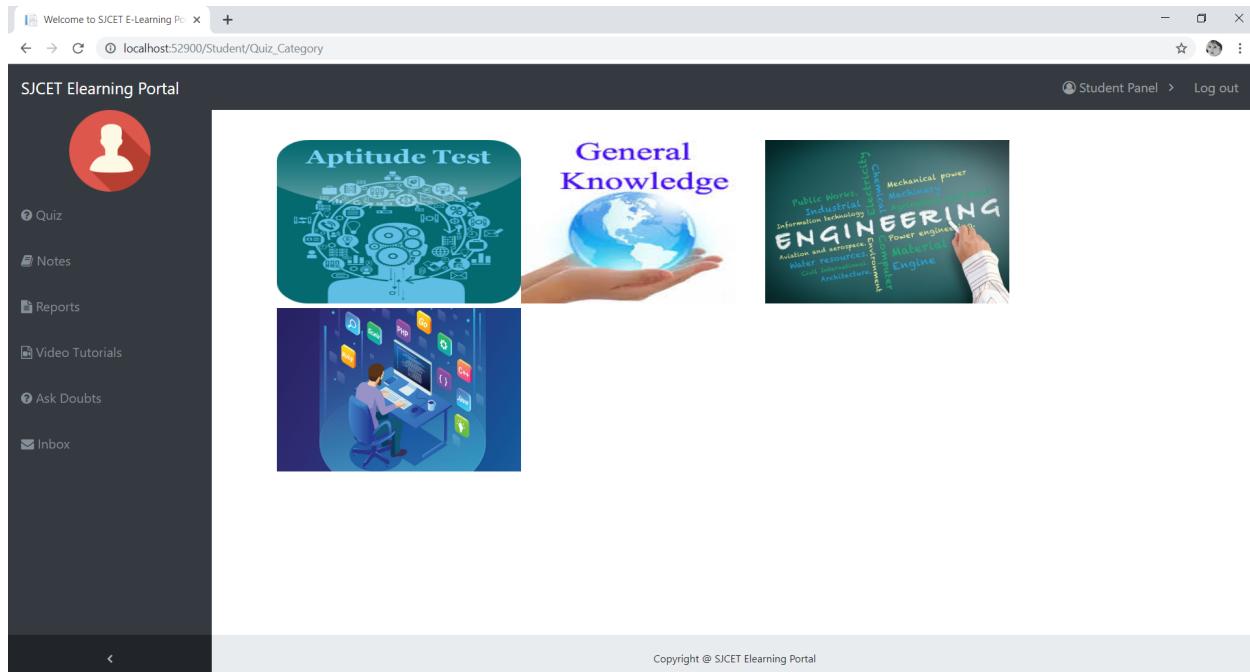


Fig. A.12: Quiz Category Selection

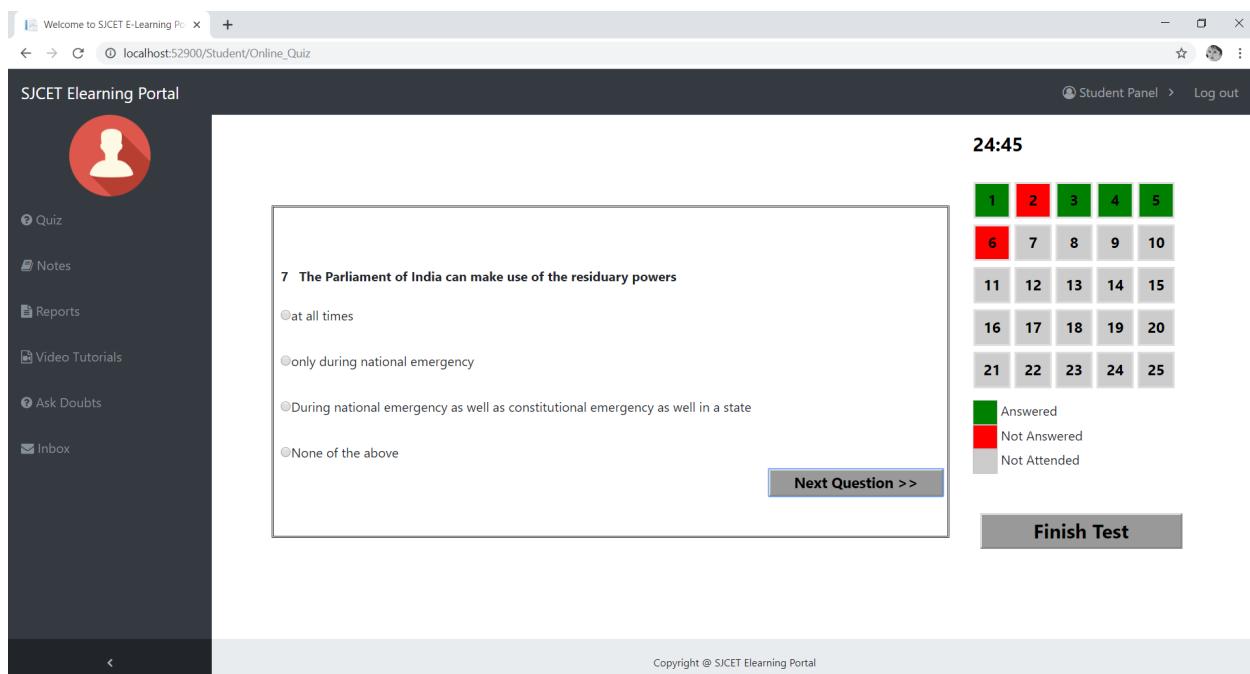


Fig. A.13: Attend Online Quiz

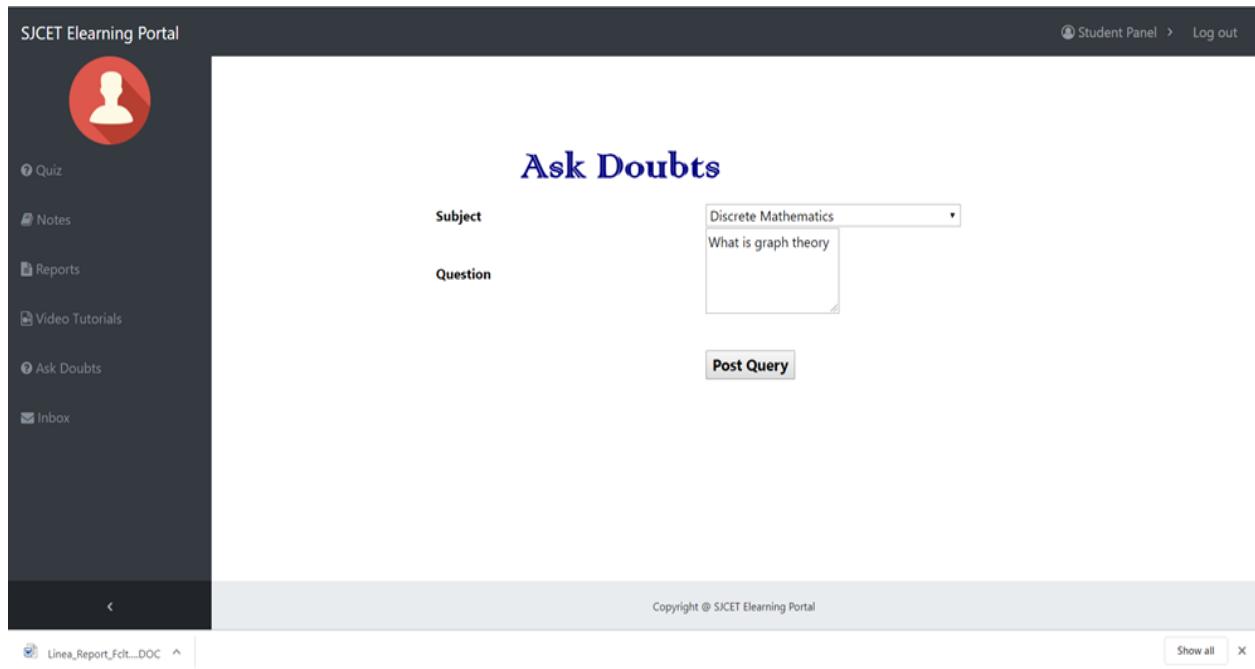


Fig. A.14: Ask Doubts

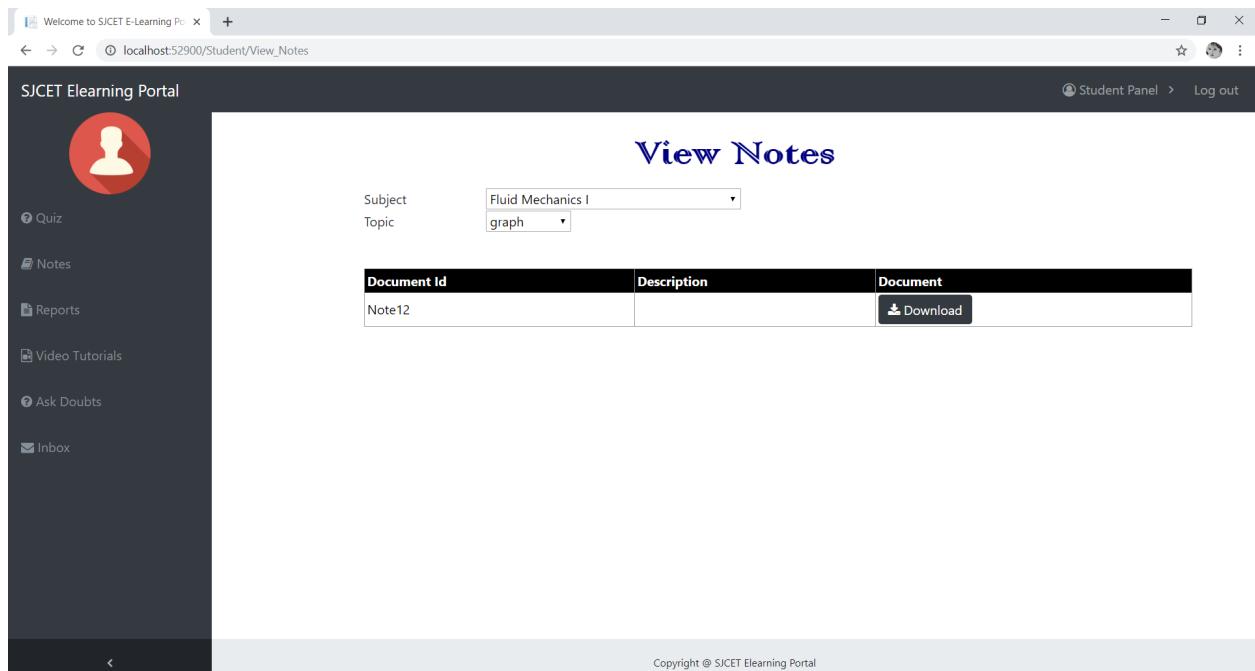


Fig. A.15: View/Download Notes

Personal Details			
Name	Aswathy Sathyan	Gender	Female
Date Of Birth	12/30/1995 12:00:00 AM	Mobile No	9747628288
Email Id	001aswathy.sathyan@gmail.com	Nationality	Indian
Religion	Hindu	Category	Vishwakarma
Blood Group	AB+		

Address & Guardian Details			
Address	Aswathy bhavan, Thittamel, Chengannur		
State	Kerala	District	Alappuzha
Father Name	Sathyan	Pincode	689121
Mobile Number	9605376919	Occupation	Mechanic

Admission Details			
Date Of Admission	8/8/2016 12:00:00 AM	Department	Master of Computer Application
Admitted Program	MCA	Admitted Branch	MCA_Regular
Admission No	16mca011	Semester	6

Qualification Details			
-----------------------	--	--	--

Copyright @ SJCET Elearning Portal

Fig. A.16: View/Edit Profile

Subject	Topic	Document
Linear Algebra & Complex Analysis	qwer	Download
Report1	ZX	

Copyright @ SJCET Elearning Portal

Linea_Report_Flt....DOC ^ 144/144 kB Show all X

Fig. A.17: View/Download Reports

The screenshot shows a web browser window for the SJCET Elearning Portal. The URL in the address bar is `localhost:52900/Student/View_Video`. The page title is "Welcome to SJCET E-Learning Po...". The main content area is titled "View Video Tutorials". On the left, there is a sidebar with icons for Quiz, Notes, Reports, Video Tutorials (selected), Ask Doubts, and Inbox. The main content area has dropdown menus for Subject (Statistics) and Topic (xcvb). A table lists a single video entry: Document Id: Video3, Description: zsxvcvb, and a Download button. At the bottom, it says "Copyright © SJCET Elearning Portal".

Fig. A.18: View/Download Video Tutorials

Faculty Functionalites :-

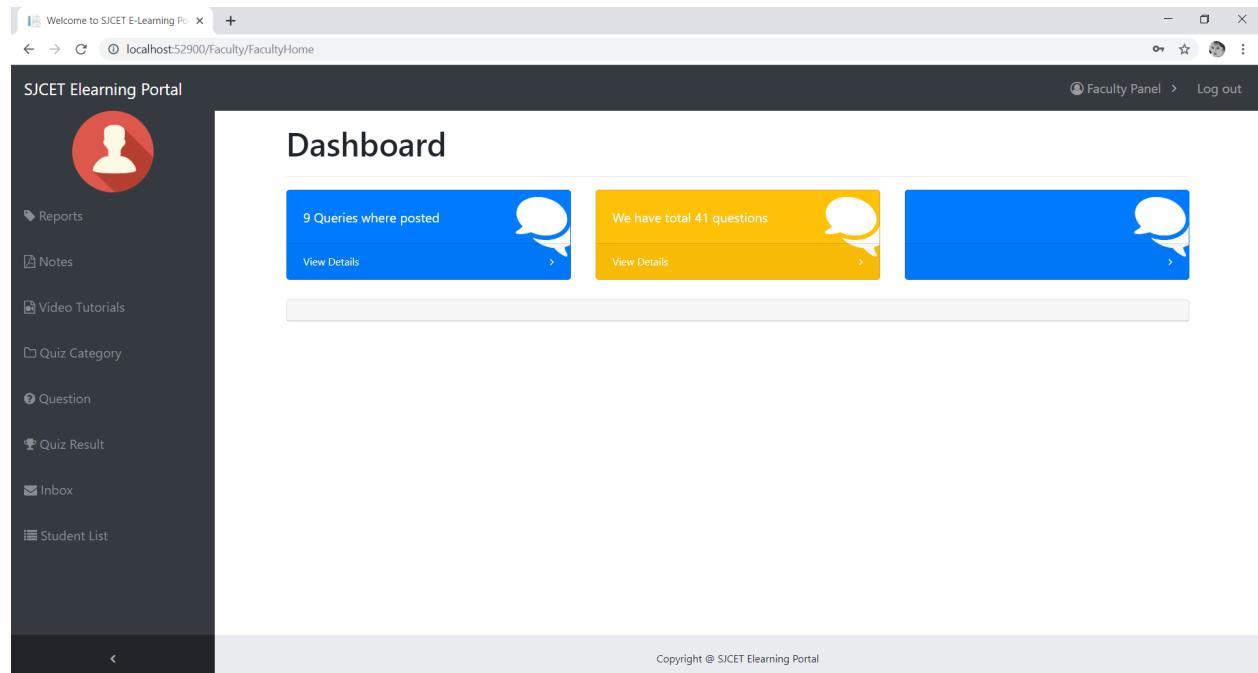


Fig. A.19: Faculty Home Page

Subject	Posted On	Query	Reply
Discrete Mathematics	5/13/2019 12:00:00 AM	zsxdcfgbh	> Read & Reply
	5/13/2019 12:00:00 AM	zxvc	> Read & Reply

Fig. A.20: View Queries Posted by Students

Fig. A.21: Add Replies to the Queries

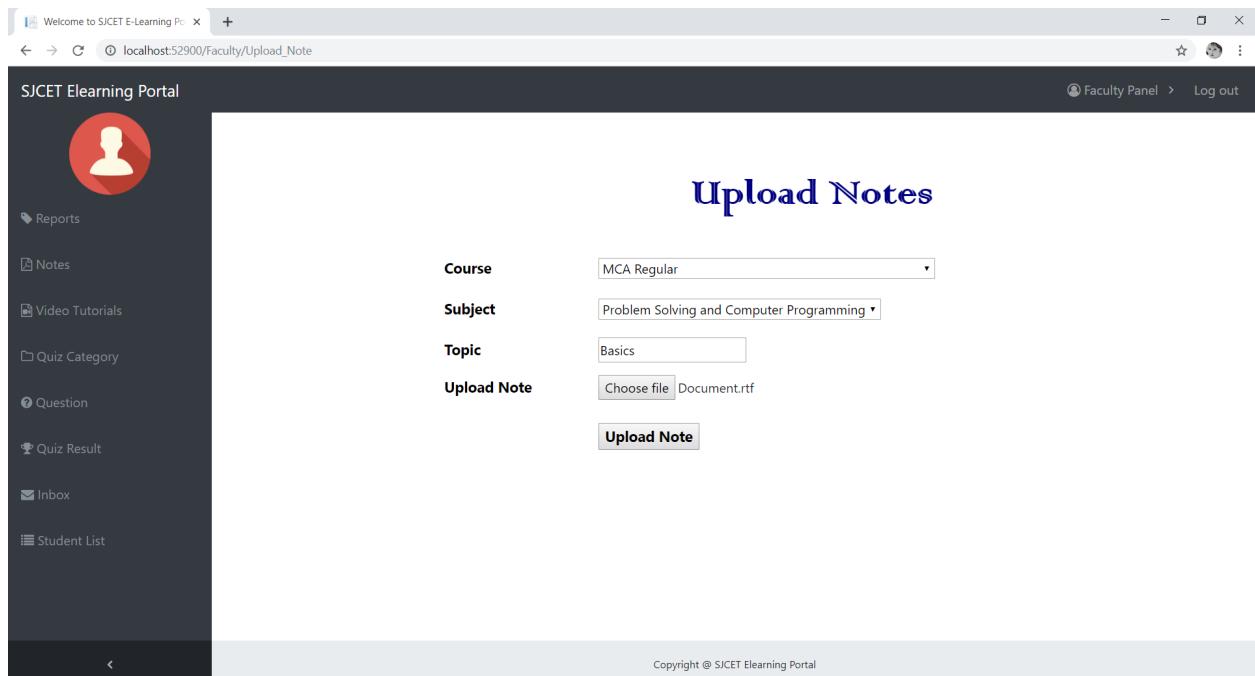


Fig. A.22: Upload E-Notes

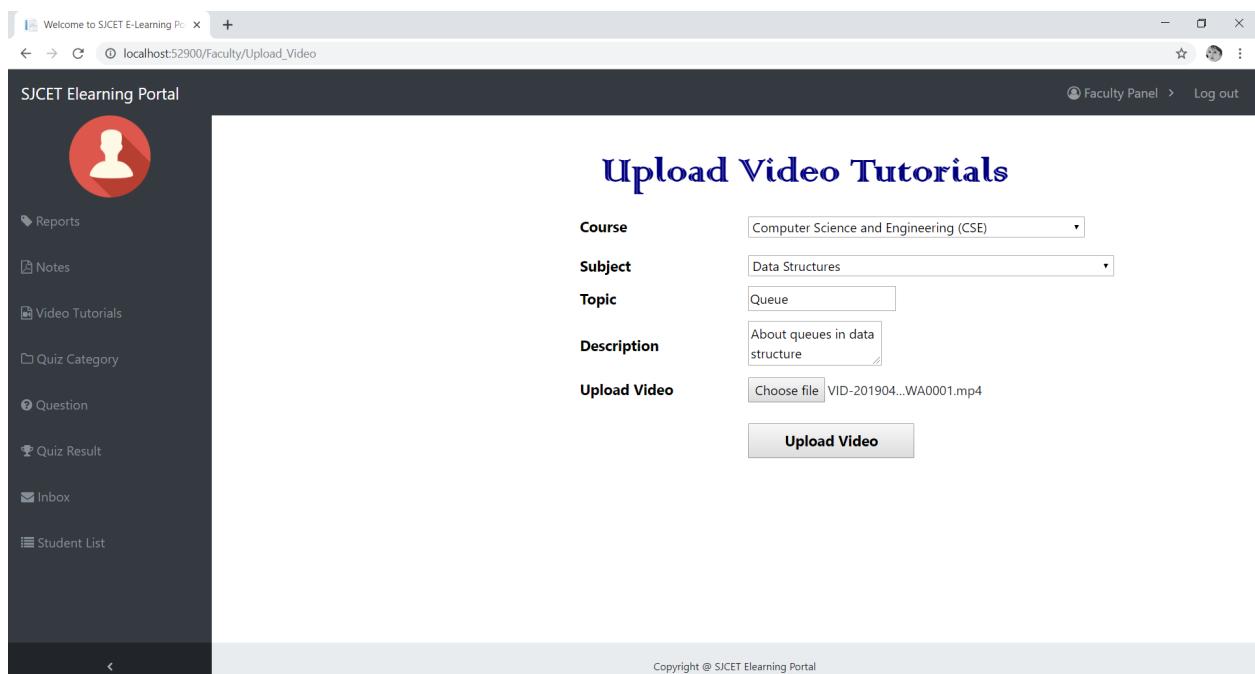


Fig. A.23: Upload Video Tutorials

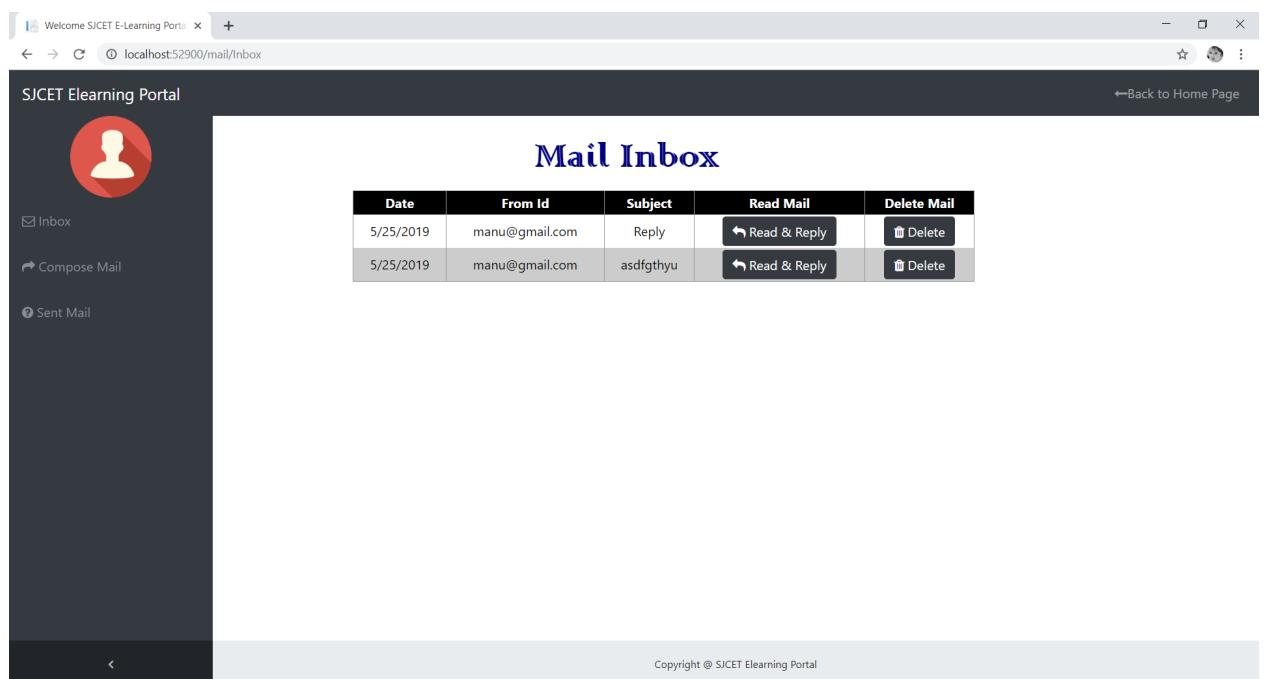


Fig. A.24: View/Sent Mails to Other Users

A.2 SAMPLE CODE

AdminClass.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data;
using System.Data.SqlClient;
namespace ELearning.Classes
{
    public class AdminNewClass
    {
        string ConnectionString =
            System.Configuration.ConfigurationManager.
            ConnectionStrings["myConn"].ConnectionString;
        SqlConnection con;
        public void OpenConnection()
        {
            con = new SqlConnection(ConnectionString);
            con.Open();
        }
        public void CloseConnection()
        {
            con.Close();
        }
        public void ExecuteQueries(string Query_)
        {
```

```
SqlCommand cmd = new SqlCommand(Query_, con);
cmd.ExecuteNonQuery();

}

//total quiz

private int quizcount;
private int qstn_count;
private string ctgry;
private string id;
private string subject;
private string rqueryid;

//view reply

private string qryid;
private string branch;
private string semester;
private string userid;
private int studcount;
private string dept;
private string fuserid;
private string usertype;

//Department

private string department;
private string prgmid;
private string subjectDetail;
private int bid;

public int Quizcount{get => quizcount; set => quizcount =
value;}
public int Qstn_count{get => qstn_count; set => qstn_count
= value;}
```

```
public string Ctgry{get => ctgry; set => ctgry = value; }
public string Id { get => id; set => id = value; }
public string Subject{ get => subject; set => subject =
    value; }
public string Rqueryid{get => rqueryid; set => rqueryid =
    value; }
public string Qryid{get => qryid; set => qryid = value; }
public string Branch{get => branch; set => branch = value; }
public string Semester{ get => semester; set => semester =
    value; }
public string Userid { get => userid; set => userid =
    value; }
public int Studcount { get => studcount; set => studcount
    = value; }
public string Dept { get => dept; set => dept = value; }
public string Fuserid { get => fuserid; set => fuserid =
    value; }
public string Usertype { get => usertype; set => usertype
    = value; }
public string Department{ get => department; set =>
    department = value; }
public string Prgmid { get => prgmid; set => prgmid =
    value; }
public string SubjectDetail{get => subjectDetail; set
=>subjectDetail=value;}
public int Bid { get => bid; set => bid = value; }

//Get Quiz Count
public int getCountQuiz()
```

```
{  
    OpenConnection();  
    SqlCommand command = new SqlCommand("select  
        count(Quiz_Id) from Quiz_Report", con);  
    object cnt = command.ExecuteScalar();  
    if (cnt != DBNull.Value)  
    {  
        quizcount = (int)cnt;  
    }  
    return quizcount;  
}  
  
//Get Question count  
public int getQuestionCount()  
{  
    OpenConnection();  
    SqlCommand command = new SqlCommand("select count(Q_Id)  
        from Quiz_Questions", con);  
    object cnt = command.ExecuteScalar();  
    if (cnt != DBNull.Value)  
    {  
        qstn_count = (int)cnt;  
    }  
    return qstn_count;  
}  
  
//Fetching category from the table Quiz_Category  
public DataTable FetchCategory()  
{  
    OpenConnection();  
    DataTable dtCategory = new DataTable();
```

```
SqlCommand command = new SqlCommand("Select
    SubCat_Id,SubCategory from Quiz_Subcategory", con);
SqlDataAdapter da = new SqlDataAdapter(command);
da.Fill(dtCategory);
CloseConnection();
return dtCategory;
}

//Fetch Questions
public DataTable FetchQuestions()
{
    OpenConnection();
    DataTable dtQuestions = new DataTable();
    SqlCommand command = new SqlCommand("Select
        Q_Id,Question,
        Option_A,Option_B,Option_C,Option_D,Answer,Description
        from Quiz_Questions where Category_Id=@Subid", con);
    command.Parameters.AddWithValue("@Subid", ctgry);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtQuestions);
    CloseConnection();
    return dtQuestions;
}

//Delete Question
public void DeleteQuestion()
{
    OpenConnection();
    string qry = "delete from Quiz_Questions where
        Q_Id=@qqid ";
```

```
SqlCommand cmd = new SqlCommand(qry, con);
cmd.Parameters.AddWithValue("@qqid", id);
cmd.ExecuteNonQuery();
CloseConnection();
}

public DataTable querysubject()
{
    OpenConnection();
    DataTable dtsub = new DataTable();
    SqlCommand command = new SqlCommand("Select distinct
        Subject from Query_Details", con);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtsub);
    CloseConnection();
    return dtsub;
}

public DataTable FetchQuery()
{
    OpenConnection();
    DataTable dtquery = new DataTable();
    SqlCommand command = new SqlCommand("Select
        QUser_Id, Date, Query,
        Query_Id from Query_Details where Subject=@sub", con);
    command.Parameters.AddWithValue("@sub", subject);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtquery);
    CloseConnection();
    return dtquery;
}
```

```
//Delete Query
public void DeleteQuery()
{
    OpenConnection();
    string qry = "delete from Query_Details where
        Query_Id=@qqid ";
    SqlCommand cmd = new SqlCommand(qry, con);
    cmd.Parameters.AddWithValue("@qqid", id);
    cmd.ExecuteNonQuery();
    CloseConnection();
}

//Delete Reply
public void DeleteReply()
{
    OpenConnection();
    string qry = "delete from Query_Reply where
        Reply_id=@qqid ";
    SqlCommand cmd = new SqlCommand(qry, con);
    cmd.Parameters.AddWithValue("@qqid", id);
    cmd.ExecuteNonQuery();
    CloseConnection();
}

//fetch replies
public DataTable FetchReply()
{
    OpenConnection();
    DataTable dtreply = new DataTable();
```

```
SqlCommand command = new SqlCommand("select
    RUser_Id,Date,Reply,
    Reply_id from Query_Reply where Query_Id=@qid ", con);
command.Parameters.AddWithValue("@qid", qryid);
SqlDataAdapter da = new SqlDataAdapter(command);
da.Fill(dtreply);
CloseConnection();
return dtreply;
}

//fetch branch
public DataTable BranchDetails()
{
    OpenConnection();
    DataTable dtbranch = new DataTable();
    SqlCommand command = new SqlCommand("Select distinct
        Branch from User_Details", con);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtbranch);
    CloseConnection();
    return dtbranch;
}

//Fetch name
public DataTable NameDetails()
{
    OpenConnection();
    DataTable dtName = new DataTable();
    SqlCommand command = new SqlCommand("Select
        User_Id,Name from User_Details where Branch=@brnch
        and Semester=@sem", con);
```

```
        command.Parameters.AddWithValue("@brnch", Branch);
        command.Parameters.AddWithValue("@sem", semester);
        SqlDataAdapter da = new SqlDataAdapter(command);
        da.Fill(dtName);
        CloseConnection();
        return dtName;
    }

    //fetch student details
    public DataTable StudentDetails()
    {
        OpenConnection();
        DataTable dtDetails = new DataTable();
        SqlCommand command = new SqlCommand("Select * from
            User_Details where User_Id=@userid", con);
        command.Parameters.AddWithValue("@userid", userid);
        SqlDataAdapter da = new SqlDataAdapter(command);
        da.Fill(dtDetails);
        CloseConnection();
        return dtDetails;
    }

    //fetch qualification details
    public DataTable QualificationDetails()
    {
        OpenConnection();
        DataTable dtDetails = new DataTable();
        SqlCommand command = new SqlCommand("Select * from
            Qualification where User_Id=@userid", con);
        command.Parameters.AddWithValue("@userid", userid);
        SqlDataAdapter da = new SqlDataAdapter(command);
```

```
da.Fill(dtDetails);
CloseConnection();
return dtDetails;
}

public int getStudentCount()
{
    OpenConnection();
    SqlCommand command = new SqlCommand("select
        count(User_Id) from User_Details where
        User_Type='Student'", con);
    object cnt = command.ExecuteScalar();
    if (cnt != DBNull.Value)
    {
        studcount = (int)cnt;
    }
    return studcount;
}

//Fetch faculty name
public DataTable FacultyNameDetails()
{
    OpenConnection();
    DataTable dtName = new DataTable();
    SqlCommand command = new SqlCommand("Select
        User_Id,Name from User_Details where Department=@det
        and User_Type=@type", con);
    command.Parameters.AddWithValue("@det", dept );
    command.Parameters.AddWithValue("@type", usertype);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtName);
```

```
        CloseConnection();
        return dtName;
    }

    //fetch Faculty details
    public DataTable FacultyDetails()
    {
        OpenConnection();
        DataTable dtDetails = new DataTable();
        SqlCommand command = new SqlCommand("Select * from
            User_Details where User_Id=@userid", con);
        command.Parameters.AddWithValue("@userid", userid);
        SqlDataAdapter da = new SqlDataAdapter(command);
        da.Fill(dtDetails);
        CloseConnection();
        return dtDetails;
    }

    //fetch Experience details
    public DataTable ExperienceDetails()
    {
        OpenConnection();
        DataTable dtExp = new DataTable();
        SqlCommand command = new SqlCommand("Select * from
            Experience where User_Id=@userid", con);
        command.Parameters.AddWithValue("@userid", userid);
        SqlDataAdapter da = new SqlDataAdapter(command);
        da.Fill(dtExp);
        CloseConnection();
        return dtExp;
    }
}
```

```
public DataTable QuizReport()
{
    OpenConnection();
    DataTable dtQuiz = new DataTable();
    SqlCommand command = new SqlCommand("Select
        r.Quiz_Id, r.User_Id, r.Date, r.SubCategory, r.Start_Time,
        r.End_Time, r.Correct_Answer, r.Incorrect_Answer,
        r.Attended_Qusers, r.Total_Score from Quiz_Report r
        left join User_Details u on u.User_Id=r.User_Id
        where u.Branch=@sbranch and u.Semester=@sem", con);
    command.Parameters.AddWithValue("@sbranch", branch);
    command.Parameters.AddWithValue("@sem", semester);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtQuiz);
    CloseConnection();
    return dtQuiz;
}

//Add Department

public void DepartmentDetails()
{
    OpenConnection();
    string qry = "insert into Department_Details
        values(@dept)";
    SqlCommand cmd = new SqlCommand(qry, con);
    cmd.Parameters.AddWithValue("@dept", department);
    // cmd.Parameters.AddWithValue("@prgm", prgmid);
    cmd.ExecuteNonQuery();
    CloseConnection();
}
```

```
//Check whether department already exists or not
public DataTable CheckDepartment()
{
    OpenConnection();
    DataTable dtDept = new DataTable();
    SqlCommand command = new SqlCommand("Select Dept_Id
        from Department_Details where Department=@sdept",
        con);
    command.Parameters.AddWithValue("@sdept", department);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtDept);
    CloseConnection();
    return dtDept;
}

//fetch Course
public DataTable FetchCourse()
{
    OpenConnection();
    DataTable dtCourse = new DataTable();
    SqlCommand command = new SqlCommand("select * from
        Branch_Details", con);
    SqlDataAdapter da = new SqlDataAdapter(command);
    da.Fill(dtCourse);
    CloseConnection();
    return dtCourse;
}

//fetch Course
public DataTable CheckSubject()
{
```

```
OpenConnection();

DataTable dtSubject = new DataTable();
SqlCommand command = new SqlCommand("select * from
Subject_Details
where Subject=@sub and B_Id=@bid", con);
command.Parameters.AddWithValue("@sub", subjectDetail);
command.Parameters.AddWithValue("@bid", bid);
SqlDataAdapter da = new SqlDataAdapter(command);
da.Fill(dtSubject);

CloseConnection();
return dtSubject;
}

//Insert Subject Details
public void InsertSubject()
{
    OpenConnection();
    string qry = "insert into Subject_Details
values(@subdet,@sbid)";
    SqlCommand cmd = new SqlCommand(qry, con);
    cmd.Parameters.AddWithValue("@sbid", bid);
    cmd.Parameters.AddWithValue("@subdet", subjectDetail);
    cmd.ExecuteNonQuery();
    CloseConnection();
}
}
```

Registration.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.IO;
using System.Data.SqlClient;
using ELearning.Classes;

namespace ELearning.Admin
{
    public partial class Registration : System.Web.UI.Page
    {
        AdminClass objAdmReg = new AdminClass();
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                DataTable dtDeptmnt = new DataTable();
                dtDeptmnt = objAdmReg.DepartmentDetails();
                if (dtDeptmnt.Rows.Count > 0)
                {
                    ddlDept.DataSource = dtDeptmnt;
                    ddlDept.DataTextField = "Department";
                    ddlDept.DataValueField = "Dept_Id";
                    ddlDept.DataBind();
                }
            }
        }
    }
}
```

```
        }

        ddlDept.Items.Insert(0, "----Select---");

    }

}

protected void btnRegister_Click(object sender, EventArgs e)
{
    if (rdfemale.Checked)
    {
        objAdmReg.Gender = rdfemale.Text.ToString();
    }
    else if (rdmale.Checked)
    {
        objAdmReg.Gender = rdmale.Text.ToString();
    }

    objAdmReg.User_type = Session["usertype"].ToString();
    string lastmob = txtMobNo.Text.ToString();
    lastmob = lastmob.Substring(lastmob.Length - 4);
    Session["username"] = txtfname.Text.ToString() +
        lastmob;
    Session["password"] = txtfname.Text.ToString() +
        lastmob;
    objAdmReg.Name = txtfname.Text.ToString() + " " +
        txtmname.Text.ToString() + " " +
        txtlname.Text.ToString();
    Session["name"] = objAdmReg.Name;
    objAdmReg.Mob_no = Convert.ToDouble(txtMobNo.Text);
    Session["mobno"] = objAdmReg.Mob_no;
```

```
objAdmReg.Dob =
    Convert.ToDateTime(txtDob.Text.ToString());
objAdmReg.Email = txtEmailId.Text.ToString();
objAdmReg.Religion =
    ddlReligion.SelectedItem.Text.ToString();
objAdmReg.Category = txtCategory.Text.ToString();
objAdmReg.Nationality =
    ddlNationality.SelectedItem.Text.ToString();
objAdmReg.Blood =
    ddlBloodgroup.SelectedItem.Text.ToString();
objAdmReg.Address = txtAddress.Text.ToString() + "," +
    txtStreet.Text.ToString() + "," +
    txtCity.Text.ToString();
objAdmReg.District =
    ddlDistrict.SelectedItem.Text.ToString();
objAdmReg.State = ddlState.SelectedItem.Text.ToString();
objAdmReg.Pincode = Convert.ToInt32(txtPin.Text);
objAdmReg.Doa =
    Convert.ToDateTime(txtDoa.Text.ToString());
objAdmReg.Admsn_no = txtAdmsnNo.Text.ToString();
objAdmReg.Dept = ddlDept.SelectedItem.Text.ToString();
objAdmReg.F_name = txtFathername.Text.ToString();
objAdmReg.Occupation = txtOccupation.Text.ToString();
objAdmReg.Photo = Session["photopath"].ToString();
objAdmReg.F_mobno =
    Convert.ToDouble(txtFMob.Text.ToString());
objAdmReg.RegInsertDetails();
Session["userid"] = objAdmReg.User_id;
Session["utype"] = objAdmReg.User_type;
```

```
Response.Redirect("~/Admin/Reg_Qualification.aspx");  
}  
  
protected void btnUpload_Click(object sender, EventArgs e)  
{  
    DateTime day =  
        Convert.ToDateTime(txtDob.Text.ToString());  
    string pname = txtfname.Text.ToString() +  
        day.Day.ToString();  
    string filename =  
        Path.GetFileName(fuPhoto.PostedFile.FileName);  
    string ext = Path.GetExtension(filename);  
    if (ext.ToLower() == ".jpg" || ext.ToLower() == ".bmp"  
        || ext.ToLower() == ".png" || ext.ToLower() ==  
        ".jpeg")  
    {  
        string src = Server.MapPath("~/Photos") + "/" +  
            pname + ".JPG";  
        fuPhoto.PostedFile.SaveAs(src);  
        string picpath = "~/Photos/" + pname + ".JPG";  
        imgPhoto.Visible = true;  
        imgPhoto.ImageUrl = picpath;  
        Session["photopath"] = picpath;  
    }  
    else  
    {  
        Response.Write("Please Select An Image File....");  
    }  
}
```

```
protected void ddlState_SelectedIndexChanged(object
    sender, EventArgs e)
{
    objAdmReg.S_id =
        Convert.ToInt16(ddlState.SelectedValue.ToString());
    DataTable dtADistrict = new DataTable();
    dtADistrict = objAdmReg.AddDistrict();
    if (dtADistrict.Rows.Count > 0)
    {
        for (int i=0;i<dtADistrict.Rows.Count;i++)
        {
            ddlDistrict.Items.Add(dtADistrict.Rows[i]["District"] .
                ToString());
        }
    }
}

protected void Button1_Click(object sender, EventArgs e)
{
    Response.Redirect("~/Admin/Registration.aspx");
}

}
```

Document-Verification.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using System.IO;
using ELearning.Classes;

namespace ELearning.Admin
{
    public partial class Document_Verification : System.Web.UI.Page
    {
        AdminClass objAdm = new AdminClass();
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btnApprove_Click(object sender, EventArgs e)
        {
            var closeLink = (Control)sender;
            GridViewRow row =
                (GridViewRow)closeLink.NamingContainer;
```

```
        string firstCellText = row.Cells[0].Text;
        objAdm.Doc_id = firstCellText;
        objAdm.UpdateTable();
        BindDocument();
    }

    private void BindDocument()
    {
        DataTable dtDoc = new DataTable();
        objAdm.Doc_type =
            DdlDocument.SelectedItem.Text.ToString();
        dtDoc = objAdm.FectchDocument();
        if (dtDoc.Rows.Count > 0)
        {
            GvDocument.DataSource = dtDoc;
            GvDocument.DataBind();
        }
    }

    protected void GvDocument_RowCommand(object sender,
        GridViewEventArgs e)
    {
        if (e.Row.RowType == DataControlRowType.DataRow)
        {
            if (((e.Row.FindControl("hdnApp") as
                HiddenField).Value == "0"))

            {
                (e.Row.FindControl("Button1") as Button).Enabled
                = true;
            }
            else

```

```
{  
    (e.Row.FindControl("Button1") as Button).Enabled  
    = false;  
}  
}  
  
}  
  
protected void GvDocument_RowCommand(object sender,  
GridViewCommandEventArgs e)  
{  
    if (e.CommandName == "downloadDoc")  
    {  
        string filename = e.CommandArgument.ToString();  
        Response.Redirect(filename);  
    }  
}  
  
protected void GvDocument_PageIndexChanging(object sender,  
GridViewPageEventArgs e)  
{  
    GvDocumentPageIndex = e.NewPageIndex;  
    BindDocument();  
}  
  
  
protected void DdlDocument_SelectedIndexChanged(object  
sender, EventArgs e)  
{  
    BindDocument();  
}  
}  
}
```

View-Reports.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.IO;
using ELearning.Classes;

namespace ELearning.Student
{
    public partial class View_Reports : System.Web.UI.Page
    {
        StudentNewClass objstud = new StudentNewClass();
        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                objstud.Doctype = "Reports";
                DataTable dtSubject = new DataTable();
                dtSubject = objstud.FetchSubject();
                if (dtSubject.Rows.Count > 0)
                {
                    DdlSubject.DataSource = dtSubject;
                    DdlSubject.DataTextField = "Subject";
                    DdlSubject.DataValueField = "Subject";
                }
            }
        }
    }
}
```

```
DdlSubject.DataBind();  
}  
DdlSubject.Items.Insert(0, "----Select---");  
}  
  
}  
  
protected void DdlSubject_SelectedIndexChanged(object  
sender, EventArgs e)  
{  
objstud.Subject =  
    DdlSubject.SelectedItem.Text.ToString();  
objstud.Doctype = "Reports";  
DataTable dtTopic = new DataTable();  
dtTopic = objstud.FetchTopic();  
if (dtTopic.Rows.Count > 0)  
{  
    DdlTopic.DataSource = dtTopic;  
    DdlTopic.DataTextField = "Topic";  
    DdlTopic.DataValueField = "Topic";  
    DdlTopic.DataBind();  
}  
DdlTopic.Items.Insert(0, "----Select---");  
}  
  
protected void DdlTopic_SelectedIndexChanged(object  
sender, EventArgs e)  
{  
BindDocument();  
}
```

```
public void BindDocument()
{
    objstud.Subject =
        DdlSubject.SelectedItem.Text.ToString();
    objstud.Topic = DdlTopic.SelectedItem.Text.ToString();
    objstud.Doctype = "Reports";
    DataTable dtDocument = new DataTable();
    dtDocument = objstud.FetchDocument();
    if (dtDocument.Rows.Count > 0)
    {
        GvNotes.DataSource = dtDocument;
        GvNotes.DataBind();
    }
}

protected void GvNotes_PageIndexChanging(object sender,
    GridViewEventArgs e)
{
    GvNotesPageIndex = e.NewPageIndex;
    BindDocument();
}

protected void GvNotes_RowCommand(object sender,
    GridViewCommandEventArgs e)
{
    if (e.CommandName == "downloadDoc")
    {
        string filename = e.CommandArgument.ToString();
```

```
        Response.Redirect(filename);  
    }  
}  
}  
}
```

Upload-Note.aspx.cs

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Web;  
using System.Web.UI;  
using System.Web.UI.WebControls;  
using System.Data;  
using System.Data.SqlClient;  
using System.IO;  
using ELearning.Classes;  
  
namespace ELearning.Faculty  
{  
    public partial class Upload_Note : System.Web.UI.Page  
    {  
        FacultyClass objFcly = new FacultyClass();  
        protected void Page_Load(object sender, EventArgs e)  
        {  
            if (!IsPostBack)  
            {  
                // Page Initialization  
            }  
        }  
    }  
}
```

```
DataTable dtDept = new DataTable();
dtDept = objFcly.FetchDept();
if (dtDept.Rows.Count > 0)
{
    ddlDepartment.DataSource = dtDept;
    ddlDepartment.DataTextField = "Branch_Name";
    ddlDepartment.DataValueField = "B_Id";
    ddlDepartment.DataBind();
}
ddlDepartment.Items.Insert(0, "---Select---");
}

protected void ddlDepartment_SelectedIndexChanged(object
sender, EventArgs e)
{
    objFcly.NDept = ddlDepartment.SelectedValue.ToString();
    DataTable dtSubject = new DataTable();
    dtSubject = objFcly.FetchSubject();
    if (dtSubject.Rows.Count > 0)
    {
        ddlSubject.DataSource = dtSubject;
        ddlSubject.DataTextField = "Subject";
        ddlSubject.DataValueField = "Sub_Id";
        ddlSubject.DataBind();
    }
    ddlSubject.Items.Insert(0, " ---Select---");
}
```

```
protected void btnNoteUpload_Click(object sender,
EventArgs e)
{
    //Setting document path
    string pname = ddlSubject.SelectedItem.Text.ToString() .
    Substring(0, 5)+"_Note" + "_" +
    Session["u_id"].ToString();
    string filename =
        Path.GetFileName(fuNotes.PostedFile.FileName);
    string ext = Path.GetExtension(filename);
    if (ext.ToLower() == ".doc" || ext.ToLower() == ".docx" ||
        ext.ToLower() == ".html" || ext.ToLower() ==
        ".htm" || ext.ToLower() == ".pdf" || ext.ToLower() ==
        ".xls" || ext.ToLower() == ".ppt" ||
        ext.ToLower() == ".pptx" || ext.ToLower() == ".txt")
    {
        string src = Server.MapPath("~/Documents") + "/" +
        pname + ".DOC";
        fuNotes.PostedFile.SaveAs(src);
        string doc = "~/Documents/" + pname + ".DOC";
        ViewState["DocPath"] = doc;
        //Inserting values to database
        objFclty.UserId = Session["u_id"].ToString();
        objFclty.NSubject =
            ddlSubject.SelectedItem.Text.ToString();
        objFclty.NTopic = txtTopic.Text.ToString();
        objFclty.Notes =
            Convert.ToString(ViewState["DocPath"]);
        objFclty.UploadNotes();
    }
}
```

```
Response.Write("<script LANGUAGE='JavaScript'  
    >alert('Successfully uploaded the  
    document...')</script>");  
txtTopic.Text = "";  
}  
  
else  
{  
    Response.Write("<script LANGUAGE='JavaScript'  
        >alert('Please select a document  
        file...')</script>");  
}  
}  
}  
}
```