

# **DOSSIER-DOCUMENT ARCHIVING SYSTEM**

A PROJECT REPORT

Submitted by

**SREEKUTTY S NAIR**

**Reg.No:MGP16MCA-D43**

**to**

the APJ Abdul Kalam Technological University

in partial fulfillment of the requirements for

the award of the degree

*of*

**MASTER OF COMPUTER APPLICATIONS**



**Department of Computer Applications**  
**SAINTGITS COLLEGE OF ENGINEERING**  
**Kottukulam Hills, Pathamuttom P.O., Kottayam 686 532**

**April, 2018**

## **DECLARATION**

I undersigned hereby declare that the project report “**DOSSIER-DOCUMENT ARCHIVING SYSTEM**” , 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 **Asst.Prof. Jijo Varghese**. 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.

**Pathamuttom**  
**20 April, 2018**

**SREEKUTTY S NAIR**  
**Reg.No:MGP16MCA-D43**

**DEPARTMENT OF COMPUTER APPLICATIONS**  
**SAINTGITS COLLEGE OF ENGINEERING**  
Kottukulam Hills, Pathamuttom P.O., Kottayam 686 532.  
(Approved by AICTE and affiliated to APJ Abdul Kalam Technological University)



**CERTIFICATE**

This is to certify that the report entitled “**DOSSIER-DOCUMENT ARCHIVING SYSTEM**” submitted by “**SREEKUTTY S NAIR , Reg.No:MGP16MCA-D43**” to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master in Computer Applications is a bonafide record of the project work carried out by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

**Asst.Prof.Jijo Varghese**  
Internal Guide

**Asst.Prof. Libin M Joseph**  
Project Co-ordinator

**Dr. Rajesh K.S**  
Head of the Department

**Dr. M.D.Mathew**  
Principal

*Viva-voce held on:.....*

**External Examiner 1:**

**External Examiner 2:**

## **ABSTRACT**

This system has developed for managing all the old question paper in a repository system, where user can easily search the past question paper as their need. Each user of the system has role and permission according to their role. This system will help the students to get all question banks at a single place, there will be no need to search for question papers on different sites. This system also allow the students to submit their project and seminar reports, so that the faculty can approve or reject the document submission. Different books and sites are available for question papers. But searching for required questions in different books and on different sites is time consuming. And it needs lots of efforts. Also the system providing E-books for each subject and so, the student can easily access. Many times students do not get the questions on a particular topic.

This system reduces searching efforts for the Question Paper, Sample Solved Question papers are available, Marks and topic wise Questions are available, Updating is possible. Users can upload their own questions also. Students can get all the old question papers and questions related to a particular topic in a single system. And it will be easy for teachers to generate question paper, and can conduct examination at any instance of time. This application will be helpful to educational infrastructure.

The system consist of three modules:-

- Admin
- Staff
- Student

The system contains three users, administrator ,faculty and staff. Role of admin is to register the authorized staffs and add old question papers based on corre-

sponding subject. The staffs can upload questions and view the questions in the system. Admin can view the questions that are uploaded by the staffs and can delete those questions which are seen irrelevant. In the question paper module, staff and student can upload a pool of questions. Staff can also approve or reject the document submission by the students.

## ACKNOWLEDGEMENT

If words are considered as symbols of approval and tokens of acknowledgment, then let words play the heralding role in expressing my gratitude. To bring something into existence is truly a work of God. I would like to thank God for not letting me down and showing me the silver lining in the dark clouds.

I would like to thank **Dr. M D Mathew**, Principal, Saintgits College of Engineering for his support and encouragement. I convey my heartfelt thanks to **Dr. Rajesh K.S** (Head of the Department of Master of Computer Applications, Saintgits College of Engineering,) for providing an opportunity for the project presentation. It is my pleasure to express my gratitude to the project coordinator **Asst. Prof. Libin M Joseph**, Department of Computer Applications, Saintgits College of Engineering whose support and constructive criticism has led to the successful completion of the task.

With the biggest contribution to this report, I would like to thank **Asst.Prof.Jijo Varghese**, Department of Computer Applications who had given me full support in guiding me with stimulating suggestions and encouragement to go ahead in all the time of the this work.

I would also thank my institution and faculty, my family and friends without whom this project would have been a distant reality.

**Sreekuty S Nair**

# CONTENTS

| Contents   | Page No.   |
|--|------------|
| <b>ACKNOWLEDGEMENT</b>                             | <b>i</b>   |
| <b>ABSTRACT</b>                                    | <b>ii</b>  |
| <b>LIST OF TABLES</b>                              | <b>vi</b>  |
| <b>LIST OF FIGURES</b>                             | <b>vii</b> |
| <b>Chapter I: INTRODUCTION</b>                     | <b>1</b>   |
| I.1 PROBLEM DEFINITION . . . . .                   | 1          |
| I.2 About the Organization . . . . .               | 1          |
| I.3 Objective of the Project . . . . .             | 2          |
| <b>Chapter II: LITERATURE SURVEY</b>               | <b>3</b>   |
| II.1 Initial Investigation . . . . .               | 3          |
| II.2 Existing System . . . . .                     | 3          |
| II.3 PROPOSED SYSTEM . . . . .                     | 4          |
| II.3.1 Advantages of the Proposed System . . . . . | 4          |
| II.3.2 Features of the Proposed System . . . . .   | 4          |
| II.4 FEASIBILITY STUDY . . . . .                   | 5          |
| II.4.1 TECHNICAL FEASIBILITY . . . . .             | 6          |
| II.4.2 ECONOMIC FEASIBILITY . . . . .              | 6          |
| II.4.3 OPERATIONAL FEASIBILITY . . . . .           | 7          |
| II.4.4 BEHAVIORAL FEASIBILITY . . . . .            | 7          |
| <b>Chapter III: SYSTEM ANALYSIS AND DESIGN</b>     | <b>8</b>   |
| III.1 Software Requirement Specification . . . . . | 8          |
| III.1.1 HARDWARE REQUIREMENT . . . . .             | 8          |
| III.1.2 SOFTWARE REQUIREMENT . . . . .             | 8          |
| III.2 SYSTEM DESIGN . . . . .                      | 9          |
| III.2.1 NON-FUNCTIONAL REQUIREMENTS . . . . .      | 9          |
| III.3 UNIFIED MODELING LANGUAGE[UML] . . . . .     | 9          |
| III.3.1 Usecase Diagram . . . . .                  | 11         |
| III.3.2 Sequence Diagram . . . . .                 | 13         |
| III.4 System Design . . . . .                      | 16         |
| III.4.1 Input Design . . . . .                     | 20         |
| III.4.2 Output Design . . . . .                    | 21         |

|                    |  |           |
|--------------------|--|-----------|
| III.4.3            | TABLES . . . . .                               | 22        |
| III.5              | TOOLS AND PLATFORMS . . . . .                  | 26        |
| III.5.1            | Introduction to Microsoft.Net . . . . .        | 26        |
| III.5.2            | The .NET Platform . . . . .                    | 26        |
| III.5.3            | The .NET Products . . . . .                    | 27        |
| III.5.4            | The .NET Services . . . . .                    | 27        |
| III.5.5            | The .NET Framework . . . . .                   | 27        |
| III.5.6            | Common Language Runtime (clr) . . . . .        | 27        |
| III.5.7            | Web Services . . . . .                         | 27        |
| III.5.8            | ASP.NET . . . . .                              | 28        |
| III.5.9            | SQL SERVER 2014 . . . . .                      | 28        |
| III.5.10           | Normalization . . . . .                        | 28        |
| <b>Chapter IV:</b> | <b>SYSTEM TESTING</b>                          | <b>30</b> |
| IV.1               | TESTING METHODOLOGIES AND STRATEGIES . . . . . | 30        |
| IV.1.1             | Unit Testing . . . . .                         | 30        |
| IV.1.2             | Integration Testing . . . . .                  | 31        |
| IV.1.3             | system Testing . . . . .                       | 31        |
| IV.1.4             | User Acceptance Testing . . . . .              | 32        |
| <b>Chapter V:</b>  | <b>SYSTEM IMPLEMENTATION</b>                   | <b>33</b> |
| <b>Chapter VI:</b> | <b>CONCLUSION</b>                              | <b>34</b> |
| <b>REFERENCES</b>  |  | <b>35</b> |
| <b>Chapter A:</b>  | <b>APPENDICES</b>                              | <b>36</b> |
| A.1                | Screen Shots Input Form,Output Forms . . . . . | 36        |
| A.2                | Sample Code . . . . .                          | 48        |



## LIST OF TABLES

| No.    | Title                           | Page No. |
|--------|---------------------------------|----------|
| III.1  | Table Department . . . . .      | 22       |
| III.2  | Table Course . . . . .          | 22       |
| III.3  | Table Scheme . . . . .          | 22       |
| III.4  | Table Batch . . . . .           | 22       |
| III.5  | Table Semester . . . . .        | 23       |
| III.6  | Table Subject . . . . .         | 23       |
| III.7  | Table User Type . . . . .       | 23       |
| III.8  | Table Staff . . . . .           | 23       |
| III.9  | Table Student . . . . .         | 24       |
| III.10 | Table Question Type . . . . .   | 24       |
| III.11 | Table Question Upload . . . . . | 24       |
| III.12 | Table Document Type . . . . .   | 24       |
| III.13 | Table Document Upload . . . . . | 25       |
| III.14 | Table Report Type . . . . .     | 25       |
| III.15 | Table Report Upload . . . . .   | 26       |

## LIST OF FIGURES

| No.   | Title  | Page No. |
|-------|--|----------|
| III.1 | Actor . . . . .                                | 12       |
| III.2 | Use Case . . . . .                             | 13       |
| III.3 | UML DIAGRAM FOR ADMIN . . . . .                | 14       |
| III.4 | UML DIAGRAM FOR STAFF . . . . .                | 14       |
| III.5 | UML DIAGRAM FOR STUDENT . . . . .              | 15       |
| III.6 | LEVEL0 . . . . .                               | 20       |
| III.7 | LEVEL1-ADMIN . . . . .                         | 20       |
| III.8 | LEVEL2-STAFF . . . . .                         | 20       |
| III.9 | LEVEL1-STUDENT . . . . .                       | 21       |
| A.1   | Admin Login . . . . .                          | 36       |
| A.2   | Admin Home . . . . .                           | 36       |
| A.3   | Admin Add and Edit Department . . . . .        | 37       |
| A.4   | Add and Edit Course . . . . .                  | 37       |
| A.6   | Add and Edit Subject . . . . .                 | 38       |
| A.5   | Add and Edit Batch . . . . .                   | 38       |
| A.7   | Add and Edit Syllabus . . . . .                | 39       |
| A.8   | Add and Edit Staff . . . . .                   | 39       |
| A.9   | Add and Edit Student . . . . .                 | 40       |
| A.12  | Staff Login . . . . .                          | 40       |
| A.10  | Upload Previous Year Question Papers . . . . . | 41       |
| A.11  | Uploading E-books and E-notes . . . . .        | 41       |
| A.13  | Staff Home . . . . .                           | 42       |
| A.14  | Uploading Question Papers . . . . .            | 42       |
| A.15  | Uploading E-books and E-notes . . . . .        | 43       |
| A.16  | Report Approvel . . . . .                      | 43       |
| A.17  | Question Paper Search . . . . .                | 44       |
| A.18  | Studdent Login . . . . .                       | 44       |
| A.19  | Student Home . . . . .                         | 45       |
| A.20  | Search for Question Papers . . . . .           | 45       |
| A.22  | Search for Syllabus . . . . .                  | 46       |
| A.21  | Search For E-books . . . . .                   | 46       |
| A.23  | Search for reports . . . . .                   | 47       |
| A.24  | Report Submission . . . . .                    | 47       |

# **CHAPTER I**

## **INTRODUCTION**

### **I.1 PROBLEM DEFINITION**

This project is aimed to developing an Document Archieving System as responsive interface system.This system can be used to store the previous year question papers,E-books,E-notes,project reports,seminar reports,assignments,the details of the department, staff,students and updating the details.The existing Document Archieving System is the system which under the study for the project that which going through many types of above specified processes.Like storing question papers,department details,uploading seminar reports,project reports and the details of the department,staff etc all these are manually.

This system will help the students to get all question banks at a single place, there will be no need to search for question papers on different sites.This system also allow the students to submit their project and seminar reports,so that the faculty can approve or reject the document submission. Different books and sites are available for question papers. But searching for required questions in different books and on different sites is time consuming. And it needs lots of efforts.Also the system providing E-books for each subject and so, the student can easily access.

### **I.2 ABOUT THE ORGANIZATION**

The college was founded by a group of well known academics.They are pioneering educators,having unmatched experience in the field of education with a belief that the continuous search for knowledge is the sole path to success.The Primary focus of the institution is to expose the young minds to be world of technology,instilling in them confidence and fortitude to face new challenges that enables them to excel in their chosen field.The college inculcates the development of all facets of the mind culminating in an intellectual and balanced personality.Our team of dedicated and caring faculty strives to widen the students horizon of learning thereby achieving excellent results for every student.

SAINTGITS college of Engineering,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. The students of first batch of mca bagged the first two ranks in the university. The college is also the venue of national and state level seminars and symposiums and has emerged as the hub of technical education in the state. The placement scenario is also quite commendable, with several premier industries visiting SAINTGITS college of engineering for placement and recruitment.

### **I.3 OBJECTIVE OF THE PROJECT**

This system has developed for managing all the old question paper in a repository system, where user can easily search the previous year question papers as their need. Each user of the system has role and permission according to their role. This system will help the students to get all question banks at a single place, there will be no need to search for question papers on different sites. This system also allow the students to submit their project and seminar reports, so that the faculty can approve or reject the document submission. Different books and sites are available for question papers. But searching for required questions in different books and on different sites is time consuming. And it needs lots of efforts. Also the system providing E-books for each subject and so, the student can easily access.

## **CHAPTER II LITERATURE SURVEY**

### **II.1 INITIAL INVESTIGATION**

The purpose of this document is to give a clear picture of the module designs of the project Dossier. The website provides an easy way to upload and access previous year question papers, seminar reports, project reports, assignments etc. The registered staff can upload question papers, E-books, E-notes to the repository and also staff can approve or reject the reports submitted by the students and also the registered students can upload their reports, they can easily access all the university questions, model exam question papers and series exam question papers. 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.

### **II.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 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.

Limitation 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.

## **II.3 PROPOSED SYSTEM**

The proposed system computerization is developed using SQL server as back-end and ASP.net as 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 easy 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 extent and the data consistency can be maintained. The record keeping and searching process are easy.

### **II.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

### **II.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.

## **II.4 FEASIBILITY STUDY**

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.

The benefits of this site are users can easily interact and get the services without much complexity. It helps to make it possible that more users can interact with the site at a time. Feasibility study is to determine whether the proposed system is technically, economically and behaviorally feasible in all respects.

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,

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:-

#### **II.4.1 TECHNICAL FEASIBILITY**

According to feasibility analysis procedure the technical feasibility of the system is analyzed and the technical requirements such as 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 as back end technology.

#### **II.4.2 ECONOMIC FEASIBILITY**

Economic analysis is most frequent used for evaluating of the effectiveness of the candidate system. More commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that are expected from a candidate system and compare them with the existing system. Except for the initial capital amount and the amount after each financial year, no other huge amount is needed. The expenses can be handles by any participants. So, the system is economically feasible.

This feasibility involves some questions such as whether the firm can afford to build the system, whether its benefits should substantially exceed its costs, and whether the project has higher priority and profits than other projects that might use the same resources. Here there is no problem. This firm has fully equipped hard ware, and fully fledged software, so no need to spend money on these issues. And as the client and the developer are one, there is no further problem in economic issues.



### **II.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.

### **II.4.4 BEHAVIORAL FEASIBILITY**

In today's 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 employers, login facilities are provided. These are the main feasibility studies tested in this application.

## **CHAPTER III SYSTEM ANALYSIS AND DESIGN**

### **III.1 SOFTWARE 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 "DOSSIER".

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 found out then we have to determine whether a particular software package fits for those system requirements. This selection summarizes the application requirement.

#### **III.1.1 HARDWARE REQUIREMENT**

- CPU - INTEL(R) PENTIUM(R)
- HARD DISK SPACE -500 GB
- RAM -2GB
- DISPLAY -19 STANDARD RATIO LCD MONITOR
- KEYBOARD -99-104 KEYS
- CLOCK SPEED -1.99 GHZ

#### **III.1.2 SOFTWARE REQUIREMENT**

- OPERATING SYSTEM -WINDOWS 10
- WEB SERVER -IIS 7.5

- FRONT END -ASP.NET
- BACK END -MS SQL SERVER 2014

## **III.2 SYSTEM DESIGN**

Designing the system in an effective way leads to the smooth working of any software's. 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.

### **III.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.

## **III.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

### **III.3.1 Usecase 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
- System
- Package

#### **Actor**

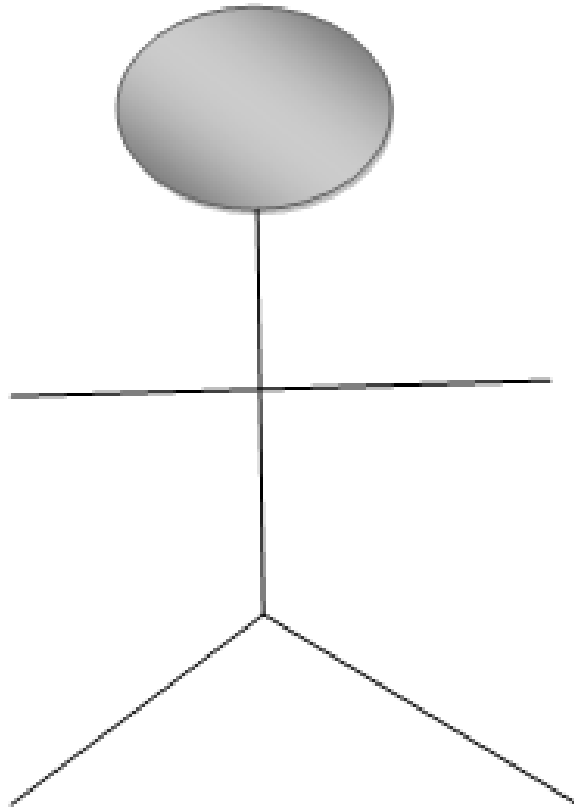


Figure III.1: Actor

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**



Figure III.2: Use Case

System is used to define the scope of the use case and drawn as a rectangle. This is an optional element but useful when your 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:-

### **III.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.

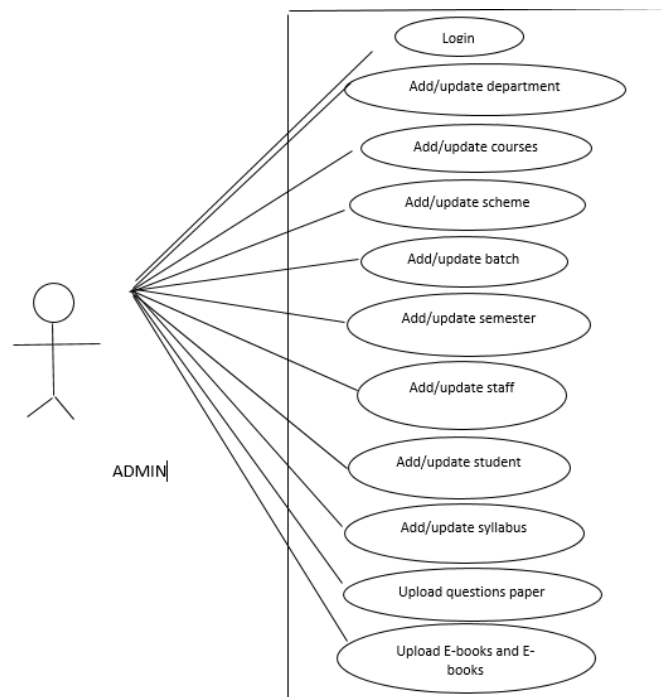


Figure III.3: UML DIAGRAM FOR ADMIN

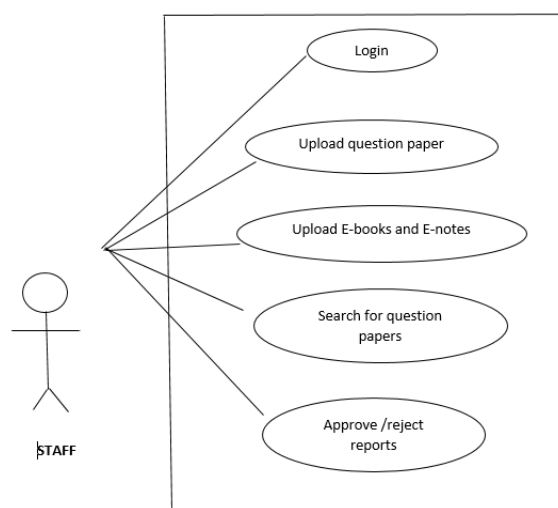


Figure III.4: UML DIAGRAM FOR STAFF



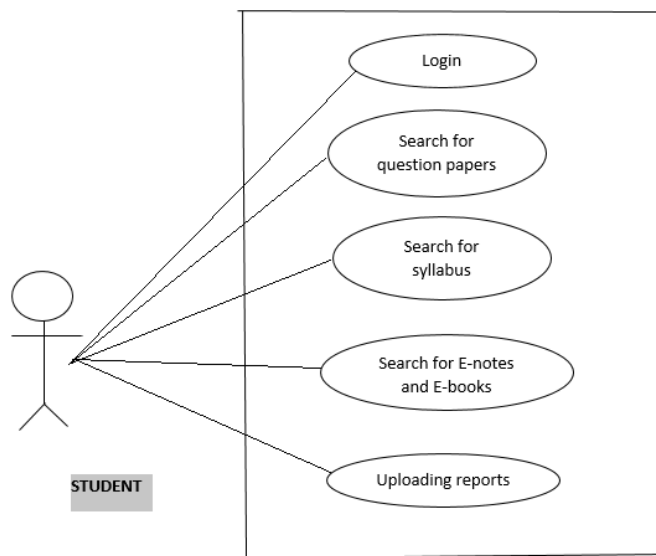


Figure III.5: UML DIAGRAM FOR STUDENT

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.

### **III.4 SYSTEM DESIGN**

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 customer's 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.

**Mainly this project consists of 3 Modules:**

- **Admin**
- **Faculty**
- **Student**

**Admin Module**

Administrator is the main actor in this system. He has the entire control of the system which includes adding all the details to generate the reports. 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 department.

- **Add Course/Delete Course**

Admin is responsible to add courses according to the each department and also responsible for delete the courses.

- **Add Batch/ Delete Batch**

A new batch can be added and deleted by the admin.

- **Add Scheme/ Delete Scheme**

Admin is responsible to add the scheme with which the university currently follows, and also responsible to delete the scheme.

- **Add Subject/ Delete Subject**

Admin is responsible to add the subjects according to the scheme and semester and also can delete the subjects.

- **Add Staff/ Delete Staff**

Admin is responsible to add Staff's details and also delete the staffs.

- **Add Students/Delete Students**

Students can be added by the admin using bulk uploading in excel.

- **Uploding Question Papers**

Admin can upload previous year university question papers,series exam and model exam question papers.

- **Upload E-books**

Admin can upload E-books according to the syllabus of each subject.

## **Faculty Module**

Faculty is important actor in the system.Faculty can upload previous year question papers,model exam question papers and series exam question papers,And also they can upload syllabus based textbooks pdf files and also upload notes prepared for subjects allotted to the corresponding staff.Add Approve or reject the report submission by the students.

- **Staff Login**

By using the user name and password, students can login to the system.

- **Question Paper Uploading**

Staff can upload previous year university question papers, series exam and model exam question papers.

- **E-books Uploading**

Staff can also upload the syllabus based textbook pdf files of corresponding subjects to the system.

- **E-notes Uploading**

Staff prepared ppts, pdf, word files of the notes of corresponding subjects can also be uploaded by the staff.

- **Searching option**

Staff can search subject based E-books, E-notes, and all the question papers for reference.

- **Approve/Reject Reports**

Corresponding class in-charge has the right to approve or reject the reports submitted by the students.

## **Student Module**

Student is an important module in the system. Student can refer all the previous year university question papers, series exam question papers, model exam question papers. He/She can view and download files of the referenced text books and notes prepared by the corresponding staff of each subject. They can also submit their project reports, seminar reports and assignments to their corresponding class in-charge.

- **Search for Question papers**

Students can search all previous year question papers for reference.

- **Search for E-books**

Students can search corresponding E-book of each subject.

- **Search for E-notes**

Students can search their semester based subject notes prepared by the corresponding staff.

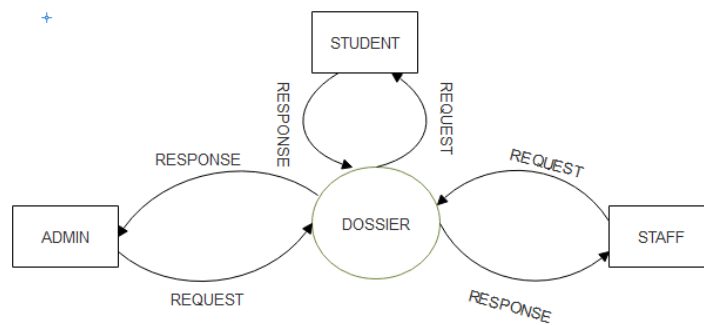


Figure III.6: LEVEL0

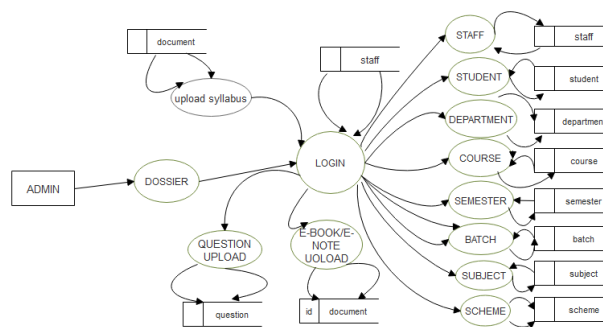


Figure III.7: LEVEL1-ADMIN

- **Search for reports**

Students can search various topics of the seminar projects for reference.

- **Upload reports**

Students can upload their project reports, seminar reports and that will be approve or reject by the coressponding staff.

### III.4.1 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

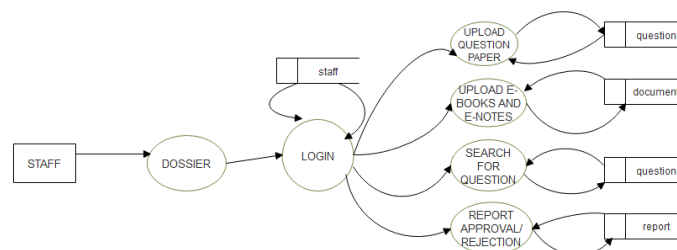


Figure III.8: LEVEL2-STAFF

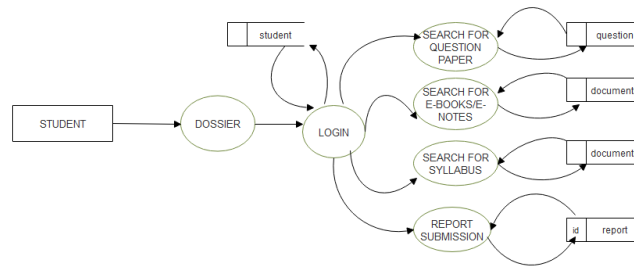


Figure III.9: LEVEL1-STUDENT

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 use 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 lift in confusion as to what is happening. Instead appropriate error messages and acknowledgments are displayed to the user.

#### III.4.2 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 system's relationship and helps user decision making.

It generally refers to the results and the information that are generated by the system. Effective, descriptive and useful design will improve the relationship with the user and the system because it is the direct source of information to the user. The objective of the output design is to convey the information of all the past activities, required status and to emphasize important events. Outputs from the computers are providing primarily to communicate the results of processing to the user. They also used to provide a permanent copy of these results for later consultation. The major outputs are system flow diagrams and data flow diagram.

### III.4.3 TABLES

Table III.1: Table Department

| FIELDS    | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|-----------|-------------|-------------|-----------------------------------|
| dept_id   | int         | PRIMARY KEY | It is used to store the unique id |
| dept_name | varchar(50) | Not Null    | Shoulde not be null               |

Table III.2: Table Course

| FIELDS    | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|-----------|-------------|-------------|-----------------------------------|
| cors_id   | int         | PRIMARY KEY | It is used to store the unique id |
| cors_name | Varchar(50) | Not Null    | Should not be null.               |
| dept_id   | int         | Foreign key | it refers to the table department |

Table III.3: Table Scheme

| FIELDS      | DATA TYPE   | CONSTRAINT  | DESCRIPTION                         |
|-------------|-------------|-------------|-------------------------------------|
| scheme_id   | int         | PRIMARY KEY | It is used to store the unique id   |
| scheme_name | Varchar(50) | Not Null    | Should not be null.                 |
| dept_id     | int         | Foreign key | it refers to the table depaertment. |

Table III.4: Table Batch

| FIELDS     | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|------------|-------------|-------------|-----------------------------------|
| batch_id   | int         | PRIMARY KEY | It is used to store the unique id |
| batch_name | Varchar(50) | Not Null    | Should not be null.               |
| cors_id    | int         | Foriegn key | it refers to the table course     |
| dept_id    | int         | Foriegn key | it refers to the table department |



Table III.5: Table Semester

| FIELDS   | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|----------|-------------|-------------|-----------------------------------|
| sem_id   | int         | PRIMARY KEY | It is used to store the unique id |
| sem_name | Varchar(50) | Not Null    | it should not be null             |
| dept_id  | int         | Foreign key | it refers to the table department |
| cors_id  | int         | Foreign key | it refers to the table department |

Table III.6: Table Subject

| FIELDS   | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|----------|-------------|-------------|-----------------------------------|
| sub_id   | int         | PRIMARY KEY | It is used to store the unique id |
| sub_name | Varchar(50) | Not Null    | Should not be null.               |
| sub_code | Varchar(50) | Not Null    | Should not be null.               |
| dept_id  | int         | Foreign key | it refers to the table department |
| cors_id  | int         | Foreign key | it refers to the table department |
| sem_id   | int         | Foreign key | it refers to the table department |

Table III.7: Table User Type

| FIELDS        | DATA TYPE   | CONSTRAINT  | DESCRIPTION                   |
|---------------|-------------|-------------|-------------------------------|
| usertype_id   | int         | Primary key | it is used to store unique id |
| usertype_name | varchar(50) | Not null    | it should not be null         |

Table III.8: Table Staff

| FIELDS      | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|-------------|-------------|-------------|-----------------------------------|
| staff_id    | int         | Primary key | it is used to store unique id     |
| staff_name  | varchar(50) | Not null    | it should not be null             |
| dept_id     | int         | Foriegn key | it refers to the table department |
| usertype_id | int         | Foriegn key | it refers to the table user type  |
| user_name   | varchar(50) | Not null    | it should not be null             |
| password    | varchar(50) | Not null    | it should not be null             |

Table III.9: Table Student

| FIELDS      | DATA TYPE   | CONSTRAINT  | DESCRIPTION                       |
|-------------|-------------|-------------|-----------------------------------|
| stud_id     | int         | Primary key | it is used to store unique id     |
| stud_name   | varchar(50) | Not null    | it should not be null             |
| cors_id     | int         | Foreign key | it refers to the table course     |
| dept_id     | int         | Foreign key | it refers to the table department |
| usertype_id | int         | Foreign key | it refers to the table user type  |
| batch_id    | int         | Foreign key | it refers to the table batch      |
| username    | varchar(50) | Not null    | it should not be null             |
| password    | varchar(50) | Not null    | it should not be null             |

Table III.10: Table Question Type

| FIELDS        | DATA TYPE   | CONSTRAINT  | DESCRIPTION                   |
|---------------|-------------|-------------|-------------------------------|
| questype_id   | int         | Primary key | it is used to store unique id |
| questype_name | varchar(50) | Not null    | it should not be null         |

Table III.11: Table Question Upload

| FIELDS       | DATA TYPE     | CONSTRAINT  | DESCRIPTION                       |
|--------------|---------------|-------------|-----------------------------------|
| ques_id      | int           | Primary key | it is used to store unique id     |
| content_type | nvarchar(200) | Not null    | it should not be null             |
| data         | varchar(max)  | Not null    | it should not be null             |
| file_name    | varchar(200)  | Not null    | it should not be null             |
| dept_id      | int           | Foreign key | it refers to the table department |
| batch_id     | int           | Foreign key | it refers to the table batch      |
| cors_id      | int           | Foreign key | it refers to the table course     |
| sub_id       | int           | Foreign key | it refers to the table subject    |
| batch_id     | int           | Foreign key | it refers to the table batch      |
| sem_id       | int           | Foreign key | it refers to the table semester   |
| date_time    | datetime2(7)  | Not null    | it should not be null             |
| uploaded_by  | varchar(50)   | Not null    | it should not be null             |

Table III.12: Table Document Type

| FIELDS       | DATA TYPE   | CONSTRAINT  | DESCRIPTION                   |
|--------------|-------------|-------------|-------------------------------|
| doctype_id   | int         | Primary key | it is used to store unique id |
| doctype_name | varchar(50) | Not null    | it should not be null         |

Table III.13: Table Document Upload

| FIELDS       | DATA TYPE     | CONSTRAINT  | DESCRIPTION                          |
|--------------|---------------|-------------|--------------------------------------|
| edocument_id | int           | Primary key | it is used to store unique id        |
| content_type | nvarchar(200) | Not null    | it should not be null                |
| data         | varchar(max)  | Not null    | it should not be null                |
| file_name    | varchar(200)  | Not null    | it should not be null                |
| dept_id      | int           | Foreign key | it refers to the table department    |
| batch_id     | int           | Foriegn key | it refers to the table batch         |
| cors_id      | int           | Foriegn key | it refers to the table course        |
| sub_id       | int           | Foriegn key | it refers to the table subject       |
| doctype_id   | int           | Foriegn key | it refers to the table document type |
| doc_name     | varchar(50)   | Not null    | it should not be null                |
| title        | varchar(50)   | Not null    | it should not be null                |
| author       | varchar(50)   | Not null    | it should not be null                |
| sem_id       | int           | Foriegn key | it refers to the table semester      |
| date_time    | datetime2(7)  | Not null    | it should not be null                |
| uploaded_by  | varchar(50)   | Not null    | it should not be null                |

Table III.14: Table Report Type

| FIELDS       | DATA TYPE   | CONSTRAINT  | DESCRIPTION                   |
|--------------|-------------|-------------|-------------------------------|
| reptype_id   | int         | Primary key | it is used to store unique id |
| reptype_name | varchar(50) | Not null    | it should not be null         |

Table III.15: Table Report Upload

| FIELDS        | DATA TYPE     | CONSTRAINT  | DESCRIPTION                        |
|---------------|---------------|-------------|------------------------------------|
| report_id     | int           | Primary key | it is used to store unique id      |
| content_type  | nvarchar(200) | Not null    | it should not be null              |
| data          | varchar(max)  | Not null    | it should not be null              |
| file_name     | varchar(200)  | Not null    | it should not be null              |
| dept_id       | int           | Foreign key | it refers to the table department  |
| batch_id      | int           | Foriegn key | it refers to the table batch       |
| cors_id       | int           | Foriegn key | it refers to the table course      |
| sub_id        | int           | Foriegn key | it refers to the table subject     |
| reporttype_id | int           | Foriegn key | it refers to the table report type |
| doc_name      | varchar(50)   | Not null    | it should not be null              |
| title         | varchar(50)   | Not null    | it should not be null              |
| guide_name    | varchar(50)   | Not null    | it should not be null              |
| sem_id        | int           | Foriegn key | it refers to the table semester    |
| staff_id      | int           | Foriegn key | it refers to the table staff       |
| active        | int           | Not null    | it should not be null              |
| date_time     | datetime2(7)  | Not null    | it should not be null              |
| uploaded_by   | varchar(50)   | Not null    | it should not be null              |

## III.5 TOOLS AND PLATFORMS

### III.5.1 Introduction to Microsoft.Net

Microsoft .Net is the umbrella term for the Microsoft's 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.

### III.5.2 The .NET Platform

The .Net platform is the developer's 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.

### **III.5.3 The .NET Products**

The .NET products will include a whole range of tools and servers that rely 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.

### **III.5.4 The .NET Services**

The .NET services will include all the web services and other corporate services provide 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.

### **III.5.5 The .NET Framework**

The .NET framework is an environment for building, deploying and running web services and other application. 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.

### **III.5.6 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 time's 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.

### **III.5.7 Web Services**

Web Services are the applications that delivered ad services that can be integrated with other web services using Internet Standards. In other words, it is an URL addressable resource that returns information to client who wants to use it and

component base programming techniques are being increasingly used to develop web application.

### **III.5.8 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 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.

### **III.5.9 SQL SERVER 2014**

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.

### **III.5.10 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 won't 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 it's 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 2014 is the most advanced, trusted, and scalable data platform released to date. Building on the success of the original SQL Server 2014 release, SQL Server 2014 has made 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.

## **CHAPTER IV SYSTEM TESTING**

### **IV.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.

In the project "DOSSIER" the testing has been successfully handled with the modules. The test data was given to each and every module in all respect and got the desired output. Each module that has been tested is found working properly.

#### **IV.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 "DOSSIER" 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 "DOSSIER" 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.

#### **IV.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.

#### **IV.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.

#### **IV.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 found 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.

## **CHAPTER V**

### **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 VI CONCLUSION**

The project titled "DOSSIER", this system has developed for managing all the old question paper in a repository system, where user can easily search the previous year question papers as their need. The project is developed as a Web Application by using ASP.NET as the front end and SQL Server 2014 as the back end. Each user of the system has role and permission according to their role. This system will help the students to get all question banks at a single place, there will be no need to search for question papers on different sites. This system also allow the students to submit their project and seminar reports, so that the faculty can approve or reject the document submission. Different books and sites are available for question papers. But searching for required questions in different books and on different sites is time consuming. And it needs lots of efforts. Also the system providing E-books for each subject and so, the student can easily access. Thus this system is very beneficial for the students and staff.

## **REFERENCES**

1. Marino Posadas,"Mastering C sharp and .NETt 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>

## APPENDIX A APPENDICES

### A.1 SCREEN SHOTS INPUT FORM,OUTPUT FORMS ADMIN FUNCTIONALITIES:-

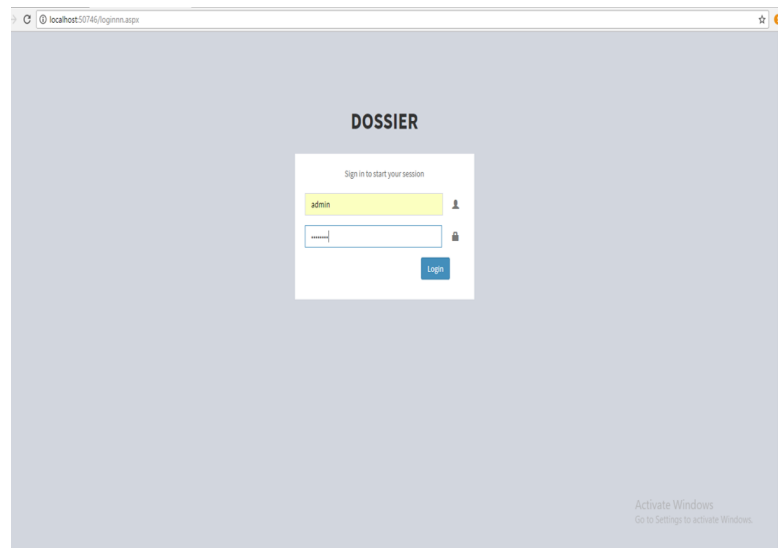


Figure A.1: Admin Login

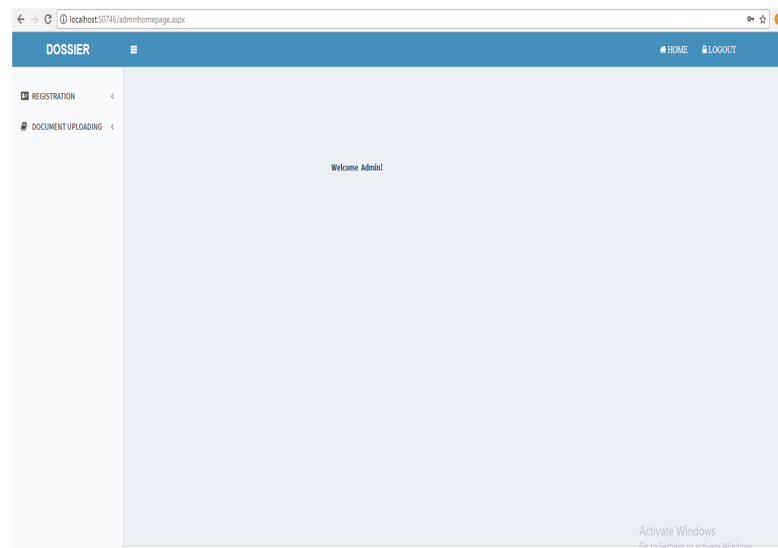


Figure A.2: Admin Home

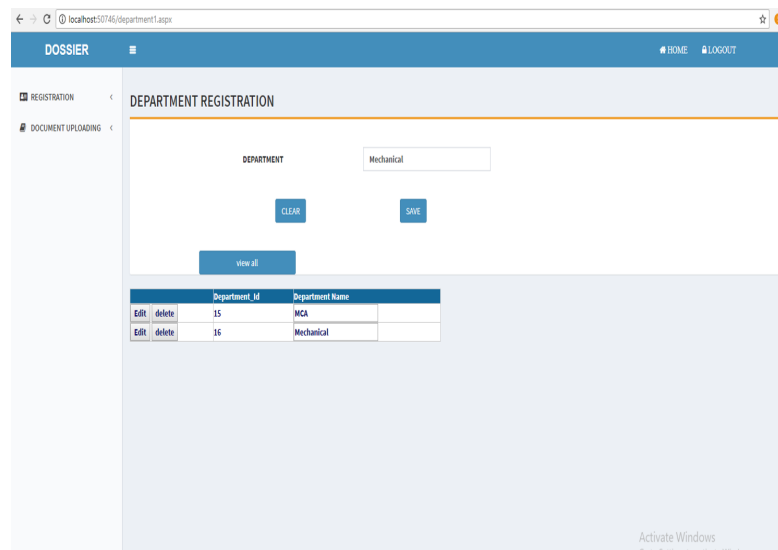


Figure A.3: Admin Add and Edit Department

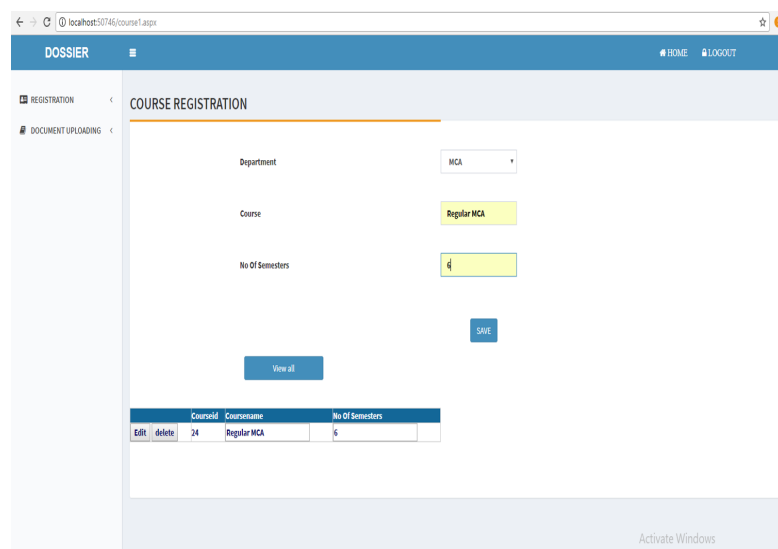


Figure A.4: Add and Edit Course

DOSSIER

HOME LOGOUT

REGISTRATION

DOCUMENT UPLOADING

SUBJECT REGISTRATION

Department

MCA

Course

Regular MCA

Scheme

kitu

Semester

semester1

Subject

DDMS

Subject Code

RLMCA201

SAVE

View all

|    | Subject_id/Subjectcode | Subjectname | Semester  | Course      |
|----|------------------------|-------------|-----------|-------------|
| 15 | RLMCA201               | DDMS        | semester1 | Regular MCA |

Activate Windows  
Go to Settings to activate Windows.

Figure A.6: Add and Edit Subject

DOSSIER

HOME LOGOUT

REGISTRATION

DOCUMENT UPLOADING

BATCH REGISTRATION

Department

MCA

Course

Regular MCA

Scheme

kitu

Batch

2016-2020

SAVE

View all

|    | BatchId/Batchname |
|----|-------------------|
| 13 | 2016-2020         |

Figure A.5: Add and Edit Batch



Department: MCA

Course: Regular MCA

Semester: semester1

Batch: 2016-2020

Choose File: topics.txt

SAVE

View

| delete | Sem_id | Course      | Semester  | View Syllabus |
|--------|--------|-------------|-----------|---------------|
|        | 38     | Regular MCA | semester1 | View          |

Figure A.7: Add and Edit Syllabus

Department: MCA

Staff Name: Libin

Designation: Assistant Professor

User Name: libin001

Password: .....

SAVE

View all

| Edit | delete | Staff_id | Staff Name    | Designation         | User Name | Password | Department |
|------|--------|----------|---------------|---------------------|-----------|----------|------------|
|      |        | 8        | RANI SARDITHA | Assistant Professor | rani001   | .....    | MCA        |
|      |        | 9        | Libin         | Assistant Professor | libin001  | .....    | MCA        |

Figure A.8: Add and Edit Staff

The screenshot shows a web browser window with the URL `localhost:50746/student.aspx`. The application has a blue header bar with the title "DOSSIER" and navigation links for "HOME" and "LOGOUT". A left sidebar contains "REGISTRATION" and "DOCUMENT UPLOADING" options. The main content area is titled "STUDENT REGISTRATION" and contains a form with the following fields: "Department" (dropdown menu showing "MCA"), "Course" (dropdown menu showing "Regular MCA"), "Batch" (dropdown menu showing "2016-2020"), and "Class Incharge" (dropdown menu showing "RANI SARITHA"). Below these fields is a "Choose File" button next to the text "student3.xlsx", and an "Upload" button. At the bottom right of the page, there is a watermark that says "Activate Windows" and "Go to Settings to activate Windows."

Figure A.9: Add and Edit Student

## STAFF FUNCTIONALITIES:-

The screenshot shows a web browser window with the URL `localhost:50746/loginn.aspx`. The application has a grey background with the title "DOSSIER" at the top. In the center, there is a white box containing a login form. The form has the text "Sign in to start your session" at the top. Below this are two input fields: the first is for the username, containing the text "mekha13", and the second is for the password, containing masked characters "\*\*\*\*\*". To the right of each input field is a small icon (a person for the username and a key for the password). At the bottom right of the form is a blue "Login" button.

Figure A.12: Staff Login

DOSSIER

From localhost:50746  
Successfully Inserted

REGISTRATION

DOCUMENT UPLOADING

### QUESTION PAPER UPLOADING

Department: MCA

Course: Regular MCA

Batch: 2015-2018

Semester: semester1

Subject: DBMS

Question Paper Type: university questio

Question Name: dbms university june

Question: Choose File: Database M...J 2017.pdf

view questions

UPLOAD

Activate Windows  
Go to Settings to activate Windows.

Figure A.10: Upload Previous Year Question Papers

DOSSIER

From localhost:50746  
Successfully Inserted

REGISTRATION

DOCUMENT UPLOADING

### E-BOOKS AND E-NOTES UPLOADING

Document Type: E-book

Subject: DBMS

Title: database system third editio

Author: Raghu Ramakrishnan

Choose File: Database M...J 2017.pdf

SAVE

Activate Windows  
Go to Settings to activate Windows.

Figure A.11: Uploading E-books and E-notes

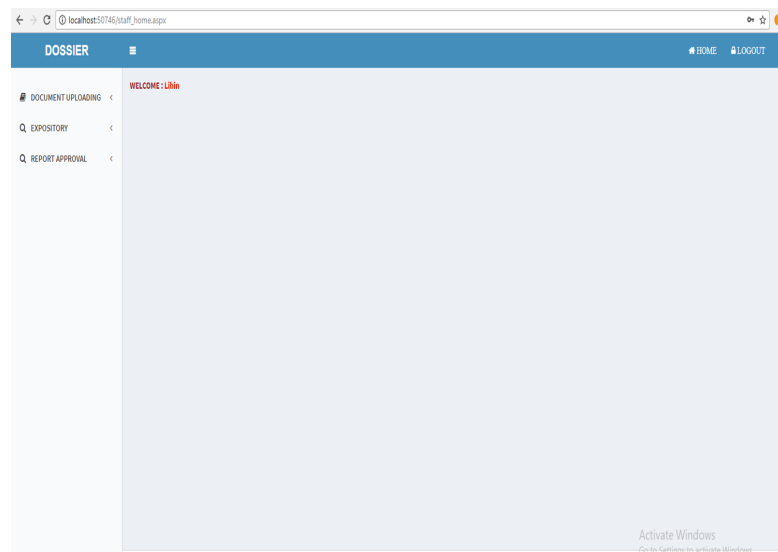


Figure A.13: Staff Home

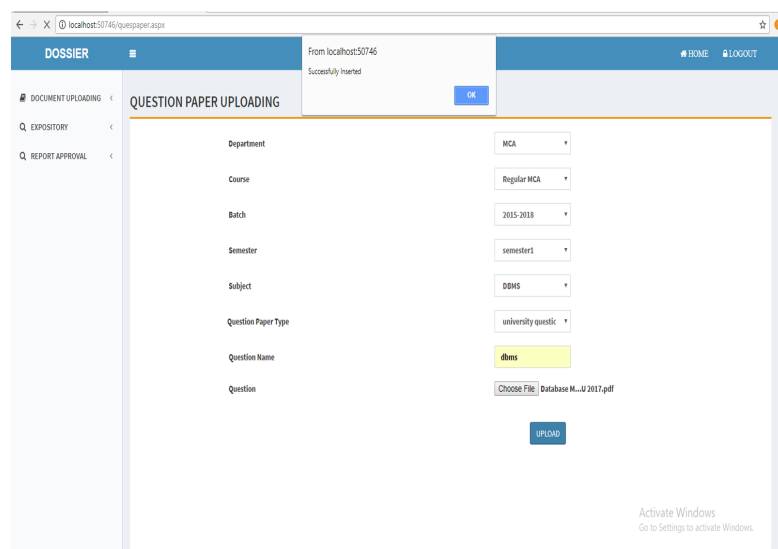


Figure A.14: Uploading Question Papers

DOSSIER

From localhost:50746  
Successfully Inserted

HOME LOGOUT

DOCUMENT UPLOADING

E-BOOKS AND E-NOTES UPLOADING

Department: MCA

Course: Regular MCA

Semester: semester1

Document Type: E-notes

Subject: DBMS

Document Title: dbms module1

Choose File Database M...U 2017.pdf

SAVE

Activate Windows  
Go to Settings to activate Windows.

Figure A.15: Uploading E-books and E-notes

localhost:50746/staff\_projectreport.aspx

DOSSIER

HOME

LOGOUT

DOCUMENT UPLOADING

EXPOSITORY

REPORT APPROVAL

PROJECT REPORT

View Reports

|         | View   | Roll No | Topic | Guided By    | Uploaded By  | Date   | Student Id            |    |
|---------|--------|---------|-------|--------------|--------------|--------|-----------------------|----|
| Approve | Reject | View    | 33    | digital bill | RANI SARITHA | meekha | 4/23/2018 11:25:08 AM | 22 |

Activate Windows  
Go to Settings to activate Windows.

Figure A.16: Report Approval

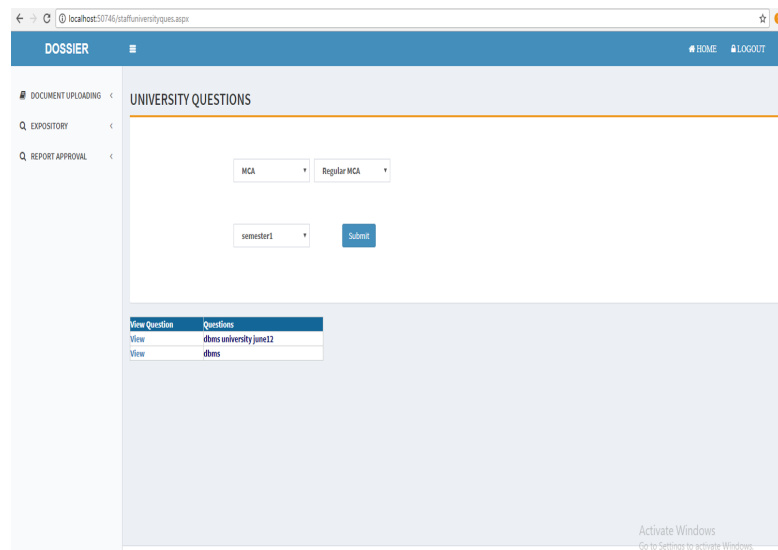


Figure A.17: Question Paper Search

## STUDENT FUNCTIONALITIES:-

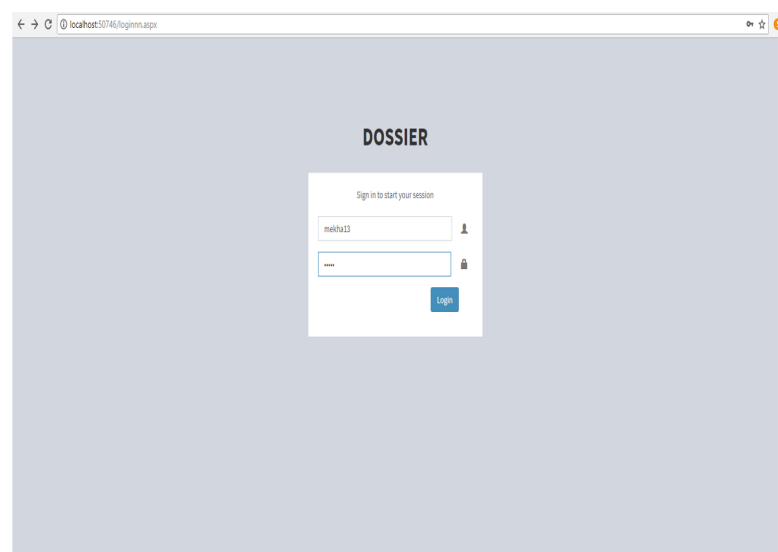


Figure A.18: Studdent Login

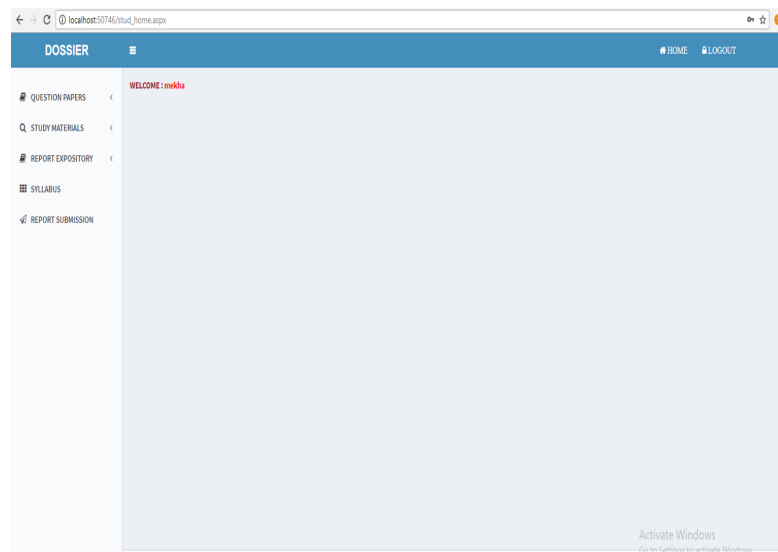


Figure A.19: Student Home

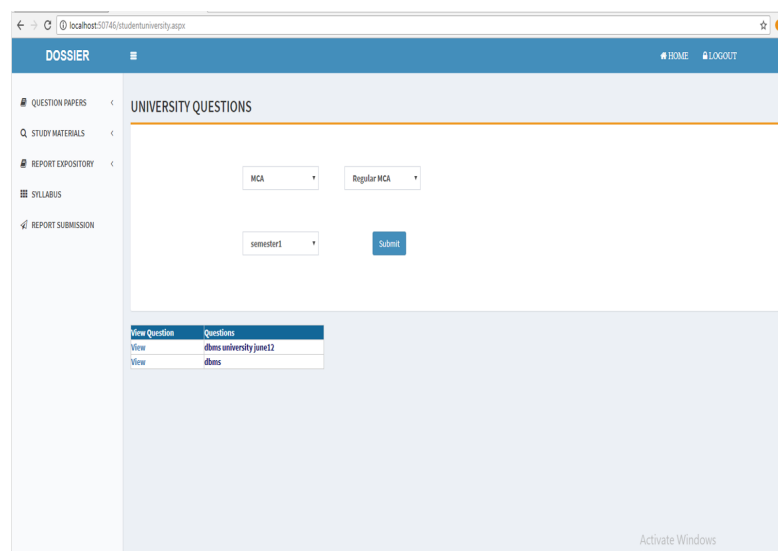


Figure A.20: Search for Question Papers

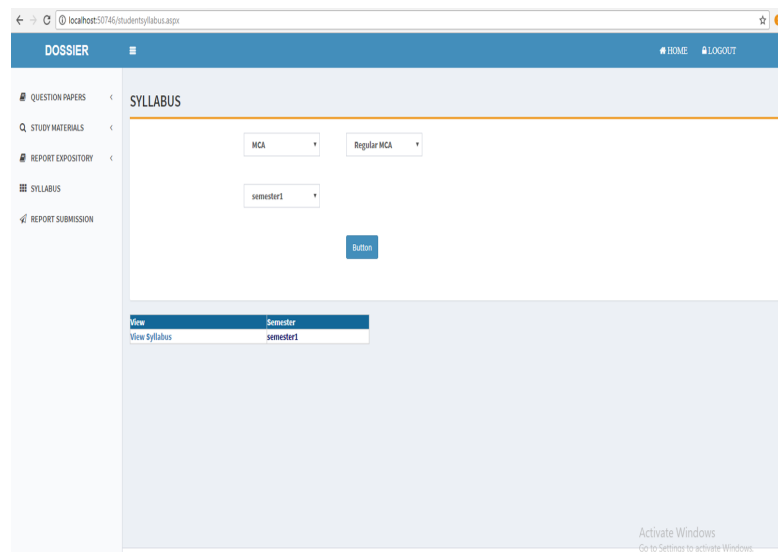


Figure A.22: Search for Syllabus

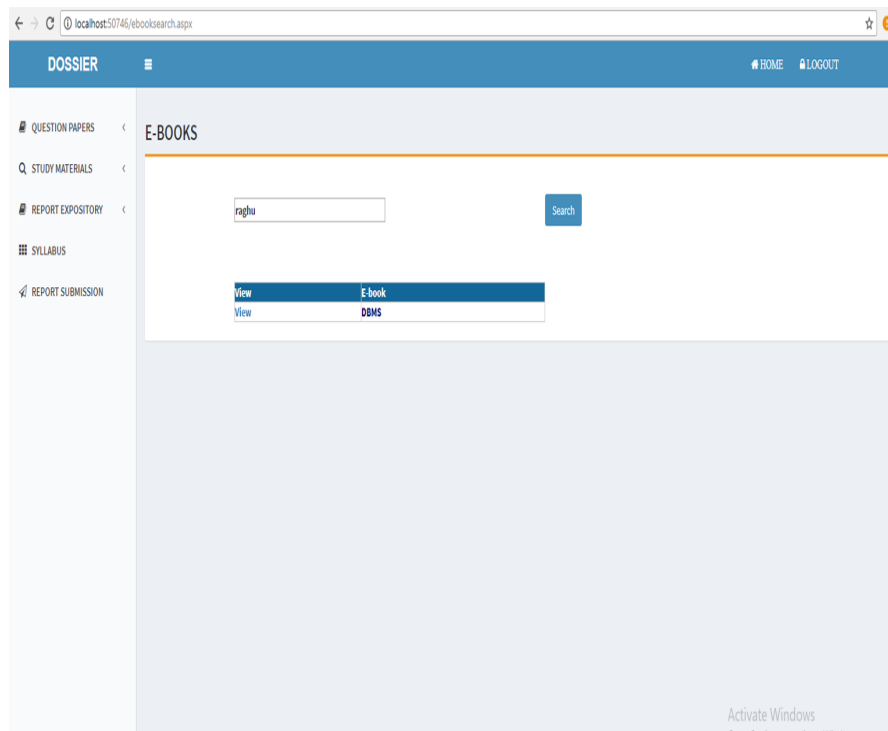


Figure A.21: Search For E-books



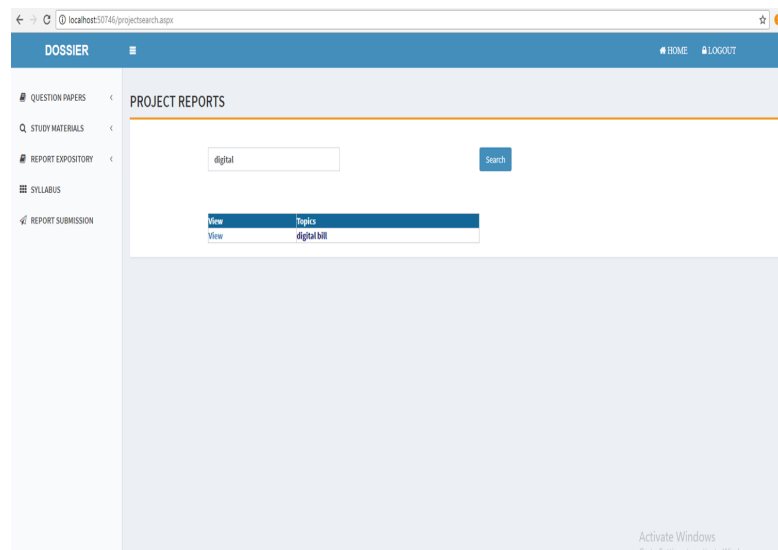


Figure A.23: Search for reports

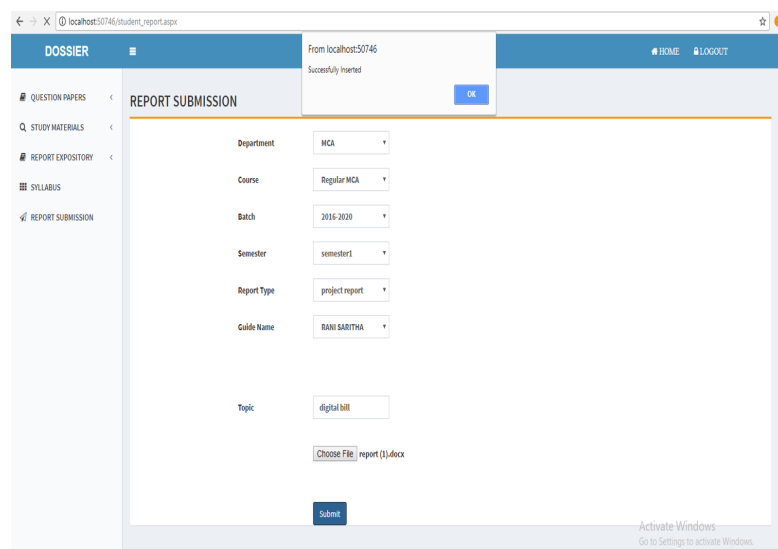


Figure A.24: Report Submission

## A.2 SAMPLE CODE

### Login.aspx.cs

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.Data.SqlClient;
8 using System.Data;
9 namespace Document_archieving_system
10 {
11     public partial class loginnn : System.Web.UI.Page
12     {
13         protected void Page_Load(object sender, EventArgs e)
14         {
15
16         }
17
18         protected void Button1_Click(object sender, EventArgs e)
19         {
20             SqlConnection con = new SqlConnection("Data Source=.;Initial
                Catalog=dossier;User ID=sa;Password=admin123");
21             con.Open();
22             SqlCommand cmd = new SqlCommand("select * from tbl_staff
                where username =@username and password=@password", con);
23
24             cmd.Parameters.AddWithValue("@username", txt_user.Text);
25
26             cmd.Parameters.AddWithValue("@password", txt_pass.Text);
27
28             SqlDataAdapter da = new SqlDataAdapter(cmd);
29
30             DataTable dt = new DataTable();
31
32             da.Fill(dt);
33
34             if (dt.Rows.Count > 0)
35             {
36                 int staff_id = Int32.Parse(dt.Rows[0]["staff_id"].ToString())
37                 ;
38                 int dept_id = Int32.Parse(dt.Rows[0]["dept_id"].ToString());
39                 string staff_name = dt.Rows[0]["staff_name"].ToString();
```

```

39 Session["staff_id"] = staff_id;
40 Session["staff_name"] = staff_name;
41 Session["dept_id"] = dept_id;
42 Response.Redirect("staff_home.aspx");
43
44
45
46
47 }
48 else if (txt_user.Text == "admin" && txt_pass.Text == "
    admin123")
49 {
50 Response.Redirect("adminhomepage.aspx");
51 }
52 else
53 {
54
55 SqlCommand cmd1 = new SqlCommand("select * from tbl_student
    where username=@username and password=@password", con);
56
57 cmd1.Parameters.AddWithValue("@username", txt_user.Text);
58
59 cmd1.Parameters.AddWithValue("@password", txt_pass.Text);
60
61 SqlDataAdapter dal = new SqlDataAdapter(cmd1);
62 DataTable dt1 = new DataTable();
63 dal.Fill(dt1);
64 if (dt1.Rows.Count > 0)
65 {
66 int stud_id = Int32.Parse(dt1.Rows[0]["stud_id"].ToString());
67 int dept_id = Int32.Parse(dt1.Rows[0]["dept_id"].ToString());
68 string stud_name = dt1.Rows[0]["stud_name"].ToString();
69 Session["stud_id"] = stud_id;
70 Session["stud_name"] = stud_name;
71 Session["dept_id"] = dept_id;
72 Response.Redirect("stud_home.aspx");
73
74
75 }
76 else
77 {
78
79 txt_user.Text = "";

```

```

80 txt_pass.Text = "";
81 Response.Write("<script language=javascript>alert('Invalid
    username and password')</script>");
82 }
83 }
84 }
85 }
86 }

```

---

### Department.aspx.cs

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.Data.SqlClient;
8 using System.Configuration;
9 using System.Data;
10 namespace Document_archieving_system
11 {
12     public partial class department1 : System.Web.UI.Page
13     {
14         protected void Page_Load(object sender, EventArgs e)
15         {
16
17         }
18
19         protected void btn_sav_Click(object sender, EventArgs e)
20         {
21
22             SqlConnection con = new SqlConnection("Data Source=.;Initial
                Catalog=dossier;User ID=sa;Password=admin123");
23             con.Open();
24             if (txt_dept.Text == "")
25             {
26                 Labell1.Visible = true;
27             }
28             SqlCommand cmd2 = new SqlCommand("select 8 from tbl_dept
                where dept_name='"+txt_dept.Text+"'", con);
29             cmd2.ExecuteNonQuery();
30             SqlDataAdapter da2 = new SqlDataAdapter(cmd2);
31             DataTable dt2 = new DataTable();

```

```

32 da2.Fill(dt2);
33 if (dt2.Rows.Count > 0)
34 {
35
36 Response.Write("<script  >alert('Already Exist')</script>");
37 txt_dept.Text = "";
38 }
39 else
40 {
41 if (txt_dept.Text == "")
42 {
43 Labell.Visible = true;
44 }
45 else
46 {
47 string sql = "insert into tbl_dept(dept_name,active)values('"
      + txt_dept.Text + "',1)";
48 SqlCommand cmd = new SqlCommand(sql, con);
49
50 cmd.ExecuteNonQuery();
51 txt_dept.Text = "";
52 Response.Write("<script>alert('Successfully Inserted')</
      script>");
53 }
54 }
55 }
56 protected void ShowData()
57 {
58 SqlConnection con = new SqlConnection("Data Source=.;Initial
      Catalog=dossier;User ID=sa;Password=admin123");
59 string a = "select dept_id,dept_name from tbl_dept where
      active=1 ";
60 SqlDataAdapter adptss = new SqlDataAdapter(a, con);
61 DataTable dtss = new DataTable();
62 adptss.Fill(dtss);
63 GridView1.DataSource = dtss;
64 GridView1.DataBind();
65 }
66 protected void GridView1_RowEditing(object sender, System.Web
      .UI.WebControls.GridViewEditEventArgs e)
67 {
68 ShowData();
69 }

```

```

70 protected void GridView1_RowUpdating(object sender, System.
    Web.UI.WebControls.GridViewUpdateEventArgs e)
71 {
72 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
73 con.Open();
74 Label dept_id = GridView1.Rows[e.RowIndex].FindControl("
    lbl_dept_id") as Label;
75 TextBox dept = GridView1.Rows[e.RowIndex].FindControl("
    txt_dept_name") as TextBox;
76 string ds = Convert.ToString(dept.Text);
77 SqlCommand cmd = new SqlCommand("Update tbl_dept set
    dept_name = '" + dept.Text + "' where dept_id =" + Convert
    .ToInt32(dept_id.Text), con);
78 cmd.ExecuteNonQuery();
79 con.Close();
80 ShowData();
81 }
82 protected void GridView1_RowCancelingEdit(object sender,
    System.Web.UI.WebControls.GridViewCancelEditEventArgs e)
83 {
84 ShowData();
85 }
86 protected void GridView1_RowDeleting(object sender,
    GridViewDeleteEventArgs e)
87 {
88 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;UserID=sa;Password=admin123");
89 con.Open();
90 Label dept_id = GridView1.Rows[e.RowIndex].FindControl("
    lbl_dep_id") as Label;
91 TextBox dept = GridView1.Rows[e.RowIndex].FindControl("
    txt_dept_name") as TextBox;
92 SqlCommand cmd = new SqlCommand("Update tbl_dept set active =
    0 where dept_id =" + Convert.ToInt32(dept_id.Text), con);
93 cmd.ExecuteNonQuery();
94 con.Close();
95 ShowData();
96 }
97 protected void GridView1_SelectedIndexChanged(object sender,
    EventArgs e)
98 {
99

```

```

100 }
101 protected void Button1_Click(object sender, EventArgs e)
102 {
103     ShowData();
104 }
105 protected void OnRowDataBound(object sender,
        GridViewRowEventArgs e)
106 {
107
108 }
109 protected void Button1_Click1(object sender, EventArgs e)
110 {
111     ShowData();
112 }
113 protected void Button2_Click(object sender, EventArgs e)
114 {
115     txt_dept.Text = "";
116     GridView1.Visible = false;
117     Label1.Text = "";
118 }
119 }
120 }

```

---

### **QuestionUpload.aspx.cs**

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.IO;
8 using System.Data;
9 using System.Data.SqlClient;
10 using System.Configuration;
11
12
13 namespace Document_archieving_system
14 {
15     public partial class questionupload : System.Web.UI.Page
16     {
17         protected void Page_Load(object sender, EventArgs e)
18         {
19             if (!IsPostBack)

```

```

20 {
21 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
22
23 con.Open();
24 string com = "Select dept_id,dept_name from tbl_dept where
    active=1";
25 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
26 DataTable dt = new DataTable();
27 adpt.Fill(dt);
28 ddl_dept.DataSource = dt;
29 ddl_dept.DataTextField = "dept_name";
30 ddl_dept.DataValueField = "dept_id";
31 ddl_dept.DataBind();
32 ddl_dept.Items.Insert(0, new ListItem("Choose Department", "0
    "));
33 con.Close();
34
35 }
36 }
37
38 protected void btn_sav_Click(object sender, EventArgs e)
39 {
40 string filename = Path.GetFileName(FileUpload1.PostedFile.
    FileName);
41 string contentType = FileUpload1.PostedFile.ContentType;
42 FileUpload1.SaveAs(Server.MapPath("~/questions/") + filename)
    ;
43
44 using (Stream fs = FileUpload1.PostedFile.InputStream)
45 {
46 using (BinaryReader br = new BinaryReader(fs))
47 {
48 byte[] bytes = br.ReadBytes((Int32)fs.Length);
49 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
50 con.Open();
51 string sql = "insert into tbl_upload(dept_id,dept_name,
    cors_id,cors_name,sem_id,sub_id,sub_name,content_type,data
    ,filename,batch_id,batch_name,qtype_id,date_time,q_name)
    values(' " + ddl_dept.SelectedItem.Value + "',' " + ddl_dept
    .SelectedItem.Value + "',' " + ddl_cors.SelectedItem.Value
    + "',' " + ddl_cors.SelectedItem.Text + "',' " + ddl_sem.

```



```

SelectedItem.Value + "','" + ddl_sub.SelectedItem.Value +
    "','" + ddl_sub.SelectedItem.Text + "','" + FileUpload1.
FileContent + "','" + FileUpload1.FileBytes + "','" +
FileUpload1.FileName + "','" + ddl_batch.SelectedItem.
Value + "','" + ddl_batch.SelectedItem.Text + "','" +
ddl_qtype.SelectedItem.Value + "','" + (GETDATE()),'" + txt_qname
.Text + "')";
52 SqlCommand cmd = new SqlCommand(sql, con);
53 cmd.ExecuteNonQuery();
54 ddl_cors.ClearSelection();
55 ddl_dept.ClearSelection();
56 ddl_batch.ClearSelection();
57 ddl_sem.ClearSelection();
58 ddl_sub.ClearSelection();
59 ddl_qtype.ClearSelection();
60 Response.Write("<script>alert('Successfully Inserted')</
    script>");
61
62
63 }
64 }
65 }
66
67 protected void ddl_dept_SelectedIndexChanged(object sender,
    EventArgs e)
68 {
69 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
70
71 con.Open();
72 string com = "Select cors_id,cors_name from tbl_cors where
    dept_id='" + ddl_dept.SelectedItem.Value + "'";
73 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
74 DataTable dt = new DataTable();
75 adpt.Fill(dt);
76 ddl_cors.DataSource = dt;
77 ddl_cors.DataTextField = "cors_name";
78 ddl_cors.DataValueField = "cors_id";
79 ddl_cors.DataBind();
80 ddl_cors.Items.Insert(0, new ListItem("select", "0"));
81 con.Close();
82 }
83

```

```

84 protected void ddl_cors_SelectedIndexChanged(object sender,
    EventArgs e)
85 {
86     SqlConnection con = new SqlConnection("Data Source=.;Initial
        Catalog=dossier;User ID=sa;Password=admin123");
87
88     con.Open();
89     string com = "Select batch_id,batch_name from tbl_batch";
90     SqlDataAdapter adpt = new SqlDataAdapter(com, con);
91     DataTable dt = new DataTable();
92     adpt.Fill(dt);
93     ddl_batch.DataSource = dt;
94     ddl_batch.DataTextField = "batch_name";
95     ddl_batch.DataValueField = "batch_id";
96     ddl_batch.DataBind();
97     ddl_batch.Items.Insert(0, new ListItem("select", "0"));
98     con.Close();
99 }
100
101 protected void ddl_batch_SelectedIndexChanged(object sender,
    EventArgs e)
102 {
103     SqlConnection con = new SqlConnection("Data Source=.;Initial
        Catalog=dossier;User ID=sa;Password=admin123");
104
105     con.Open();
106     string com = "Select sem_id,sem_name from tbl_sem where
        cors_id='" + ddl_cors.SelectedItem.Value + "'";
107     SqlDataAdapter adpt = new SqlDataAdapter(com, con);
108     DataTable dt = new DataTable();
109     adpt.Fill(dt);
110     ddl_sem.DataSource = dt;
111     ddl_sem.DataTextField = "sem_name";
112     ddl_sem.DataValueField = "sem_id";
113
114     ddl_sem.DataBind();
115     ddl_sem.Items.Insert(0, new ListItem("select", "0"));
116     con.Close();
117 }
118
119 protected void ddl_sem_SelectedIndexChanged(object sender,
    EventArgs e)
120 {

```

```

121 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
122
123 con.Open();
124 string com = "Select sub_id,sub_name from tbl_sub where
    dept_id='"+ddl_dept.SelectedItem.Value+"' and cors_id='"+
    ddl_cors.SelectedItem.Value+"'";
125 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
126 DataTable dt = new DataTable();
127 adpt.Fill(dt);
128 ddl_sub.DataSource = dt;
129 ddl_sub.DataTextField = "sub_name";
130 ddl_sub.DataValueField = "sub_id";
131 ddl_sub.DataBind();
132 ddl_sub.Items.Insert(0, new ListItem("select", "0"));
133 con.Close();
134 }
135
136 protected void ddl_sub_SelectedIndexChanged(object sender,
    EventArgs e)
137 {
138 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
139
140 con.Open();
141 string com = "Select qtype_id,qtype_name from tbl_qtype";
142 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
143 DataTable dt = new DataTable();
144 adpt.Fill(dt);
145 ddl_qtype.DataSource = dt;
146 ddl_qtype.DataTextField = "qtype_name";
147 ddl_qtype.DataValueField = "qtype_id";
148 ddl_qtype.DataBind();
149 ddl_qtype.Items.Insert(0, new ListItem("select", "0"));
150 con.Close();
151 }
152 public void viewgrid() //course grid function
153 {
154 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
155 con.Open();
156 SqlCommand cmd = new SqlCommand("select dept_name,cors_name,
    batch_name,sub_name,date_time,filename from tbl_upload",

```

```

        con);
157 SqlDataAdapter da = new SqlDataAdapter(cmd);
158 DataTable dt = new DataTable();
159 da.Fill(dt);
160 cmd.ExecuteNonQuery();
161 GridView1.DataSource = dt;
162
163 GridView1.DataBind();
164 }
165
166 protected void ddl_qtype_SelectedIndexChanged(object sender,
        EventArgs e)
167 {
168
169 }
170
171 protected void Button1_Click(object sender, EventArgs e)
172 {
173 viewgrid();
174 }
175
176 }
177 }

```

---

### Search.aspx.cs

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.Data.SqlClient;
8 using System.Configuration;
9 using System.Drawing;
10 using System.Data;
11
12 namespace Document_archieving_system
13 {
14 public partial class ebooksearch : System.Web.UI.Page
15 {
16 protected void Page_Load(object sender, EventArgs e)
17 {
18

```

```

19 }
20
21 protected void Button1_Click(object sender, EventArgs e)
22 {
23
24 }
25 protected void TextBox1_TextChanged1(object sender, EventArgs
    e)
26 {
27 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
28
29 con.Open();
30
31 SqlDataAdapter da = new SqlDataAdapter("Select sub_name,
    filename from tbl_edoc where sub_name like'" + TextBox1.
    Text + "%' or title like'" + TextBox1.Text + "%' or author
    like'" + TextBox1.Text + "%' and doc_id=1 ", con);
32
33 string text = ((TextBox)sender).Text;
34
35 DataSet ds = new DataSet();
36
37
38 da.Fill(ds);
39
40 if (ds.Tables[0].Rows.Count > 0)
41 {
42
43 GridView1.DataSource = ds.Tables[0];
44
45 GridView1.DataBind();
46
47 Label1.Visible = false;
48
49
50
51 }
52
53 else
54 {
55
56 Label1.Visible = true;

```

```

57
58 Label1.Text = "No Record Found";
59
60 this.Label1.ForeColor = Color.Red;
61 GridView1.Visible = false;
62 }
63 }
64
65 protected void GridView1_SelectedIndexChanged(object sender,
        EventArgs e)
66 {
67
68 }
69 }
70 }

```

---

### **ReportApprove.aspx.cs**

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.IO;
8 using System.Data;
9 using System.Data.SqlClient;
10 using System.Configuration;
11
12 namespace Document_archieving_system
13 {
14     public partial class student_project : System.Web.UI.Page
15     {
16         protected void Page_Load(object sender, EventArgs e)
17         {
18
19         }
20
21         protected void Button1_Click(object sender, EventArgs e)
22         {
23             ShowData();
24         }
25
26         protected void ShowData()

```

```

27 {
28 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
29 int staff_id = Convert.ToInt32(Session["staff_id"].ToString()
    );
30 string a = "select t1.topic as Topic,t1.status,t1.
    uploaded_by,t1.uploadedby_name ,t1.staff_name as [Guided
    By],t1.date_time as Date,t1.filename,t1.uploaded_by,
    tbl_student.stud_name,tbl_student.roll_no from
    tbl_report_sub t1 ,tbl_student inner join tbl_report_sub
    t2 on t2.uploaded_by = tbl_student.stud_id inner join
    tbl_staff on tbl_staff.staff_id='"+staff_id+"' and t2.
    active=0 and report_id=1";
31 SqlDataAdapter adptss = new SqlDataAdapter(a, con);
32 DataTable dtss = new DataTable();
33 adptss.Fill(dtss);
34 GridView1.DataSource = dtss;
35 GridView1.DataBind();
36 }
37
38 protected void GridView1_RowDeleting(object sender,
    GridViewDeleteEventArgs e)
39 {
40 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
41
42 con.Open();
43 Label uploaded = GridView1.Rows[e.RowIndex].FindControl("
    lbl_upload") as Label;
44 Label roll = GridView1.Rows[e.RowIndex].FindControl("lbl_roll
    ") as Label;
45 Label topic = GridView1.Rows[e.RowIndex].FindControl("
    lbl_topic") as Label;
46 Label staff = GridView1.Rows[e.RowIndex].FindControl("
    lbl_staffname") as Label;
47 Label date = GridView1.Rows[e.RowIndex].FindControl("lbl_date
    ") as Label;
48 Label uploaded1 = GridView1.Rows[e.RowIndex].FindControl("
    lbl_upload1") as Label;
49 Label status = GridView1.Rows[e.RowIndex].FindControl("
    lbl_status") as Label;
50 SqlCommand cmd = new SqlCommand("delete from tbl_report_sub
    where uploaded_by =" + Convert.ToInt32(uploaded1.Text),

```

```

        con);
51 cmd.ExecuteNonQuery();
52 con.Close();
53 ShowData();
54
55 }
56
57
58 protected void GridView1_SelectedIndexChanged(object sender,
        EventArgs e)
59 {
60
61 }
62 protected void OnRowDataBound(object sender,
        GridViewRowEventArgs e)
63 {
64
65 }
66 protected void GridView1_RowUpdating(object sender, System.
        Web.UI.WebControls.GridViewUpdateEventArgs e)
67 {
68 }
69 protected void GridView1_RowEditing(object sender, System.Web
        .UI.WebControls.GridViewEditEventArgs e)
70 {
71 GridView1.EditIndex = e.NewEditIndex;
72 ShowData();
73 }
74
75 protected void GridView1_RowCommand(object sender,
        GridViewCommandEventArgs e)
76 {
77 SqlConnection con = new SqlConnection("Data Source=.;Initial
        Catalog=dossier;User ID=sa;Password=admin123");
78 con.Open();
79
80 string sql = "update tbl_report_sub set active=1 where
        report_id=1";
81 SqlCommand cmd = new SqlCommand(sql, con);
82
83 cmd.ExecuteNonQuery();
84
85 }

```



```

86
87 protected void GridView1_SelectedIndexChanging(object sender,
    GridViewSelectEventArgs e)
88 {
89
90 }
91 }
92 }

```

---

### **Enotesearch.aspx.cs**

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.IO;
8 using System.Data;
9 using System.Data.SqlClient;
10 using System.Configuration;
11 namespace Document_archieving_system
12 {
13 public partial class studentenotes : System.Web.UI.Page
14 {
15 protected void Page_Load(object sender, EventArgs e)
16 {
17 if (!IsPostBack)
18 {
19 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
20
21 con.Open();
22 string com = "Select dept_id,dept_name from tbl_dept";
23 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
24 DataTable dt = new DataTable();
25 adpt.Fill(dt);
26
27 ddl_dept.DataSource = dt;
28 ddl_dept.DataTextField = "dept_name";
29 ddl_dept.DataValueField = "dept_id";
30 ddl_dept.DataBind();
31 ddl_dept.Items.Insert(0, new ListItem("Choose Department", "0
    "));

```

```

32
33 con.Close();
34 }
35 }
36
37 protected void Button1_Click(object sender, EventArgs e)
38 {
39 viewgrid();
40 }
41
42 protected void ddl_dept_SelectedIndexChanged(object sender,
    EventArgs e)
43 {
44 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
45
46 con.Open();
47 string com = "Select cors_id,cors_name from tbl_cors where
    dept_id='" + ddl_dept.SelectedItem.Value + "'";
48 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
49 DataTable dt = new DataTable();
50 adpt.Fill(dt);
51 ddl_cors.DataSource = dt;
52 ddl_cors.DataTextField = "cors_name";
53 ddl_cors.DataValueField = "cors_id";
54 ddl_cors.DataBind();
55 ddl_cors.Items.Insert(0, new ListItem("Choose Program", "0"))
    ;
56 con.Close();
57 }
58
59 protected void ddl_cors_SelectedIndexChanged(object sender,
    EventArgs e)
60 {
61 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
62
63 con.Open();
64 string com = "Select sem_id,sem_name from tbl_sem where
    cors_id='" + ddl_cors.SelectedItem.Value + "'";
65 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
66 DataTable dt = new DataTable();
67 adpt.Fill(dt);

```

```

68 ddl_sem.DataSource = dt;
69 ddl_sem.DataTextField = "sem_name";
70 ddl_sem.DataValueField = "sem_id";
71 ddl_sem.DataBind();
72 ddl_sem.Items.Insert(0, new ListItem("Choose Semester", "0"))
    ;
73 con.Close();
74 }
75
76 protected void ddl_sem_SelectedIndexChanged(object sender,
    EventArgs e)
77 {
78
79 }
80 public void viewgrid()
81 {
82 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
83 con.Open();
84 SqlCommand cmd = new SqlCommand("select edoc_title as [E-
    notes], filename from tbl_edoc where cors_id='" + ddl_cors.
    SelectedItem.Value + "' and sem_id='" + ddl_sem.
    SelectedItem.Value + "' and doc_id=2", con);
85
86 SqlDataAdapter da = new SqlDataAdapter(cmd);
87 DataTable dt = new DataTable();
88 da.Fill(dt);
89 cmd.ExecuteNonQuery();
90
91 GridView1.DataSource = dt;
92
93 GridView1.DataBind();
94
95 }
96
97 protected void GridView1_SelectedIndexChanged(object sender,
    EventArgs e)
98 {
99
100 }
101 }
102 }

```

---

## ReportSubmit.aspx.cs

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.UI;
6 using System.Web.UI.WebControls;
7 using System.IO;
8 using System.Data;
9 using System.Data.SqlClient;
10 using System.Configuration;
11
12
13
14 namespace Document_archieving_system
15 {
16     public partial class student_report : System.Web.UI.Page
17     {
18         protected void Page_Load(object sender, EventArgs e)
19         {
20
21             if (!IsPostBack)
22             {
23                 SqlConnection con = new SqlConnection("Data Source=.;Initial
                    Catalog=dossier;User ID=sa;Password=admin123");
24                 con.Open();
25                 string com = "Select dept_id,dept_name from tbl_dept where
                    active=1";
26                 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
27                 DataTable dt = new DataTable();
28                 adpt.Fill(dt);
29                 ddl_dept.DataSource = dt;
30                 ddl_dept.DataTextField = "dept_name";
31                 ddl_dept.DataValueField = "dept_id";
32                 ddl_dept.DataBind();
33                 ddl_dept.Items.Insert(0, new ListItem("Choose Department", "0
                    "));
34                 con.Close();
35             }
36         }
37
38         protected void ddl_cors_SelectedIndexChanged(object sender,
            EventArgs e)
```

```

39 {
40 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
41 con.Open();
42 string com = "Select batch_id,batch_name from tbl_batch where
    cors_id='" + ddl_cors.SelectedItem.Value + "'";
43 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
44 DataTable dt = new DataTable();
45 adpt.Fill(dt);
46 ddl_batch.DataSource = dt;
47 ddl_batch.DataTextField = "batch_name";
48 ddl_batch.DataValueField = "batch_id";
49 ddl_batch.DataBind();
50 ddl_batch.Items.Insert(0, new ListItem("select", "0"));
51 con.Close();
52 }
53
54 protected void ddl_batch_SelectedIndexChanged(object sender,
    EventArgs e)
55 {
56 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
57
58 con.Open();
59 string com = "Select sem_id,sem_name from tbl_sem where
    cors_id='" + ddl_cors.SelectedItem.Value + "'";
60 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
61 DataTable dt = new DataTable();
62 adpt.Fill(dt);
63 ddl_sem.DataSource = dt;
64 ddl_sem.DataTextField = "sem_name";
65 ddl_sem.DataValueField = "sem_id";
66
67 ddl_sem.DataBind();
68 ddl_sem.Items.Insert(0, new ListItem("select", "0"));
69 con.Close();
70 }
71
72 protected void ddl_sem_SelectedIndexChanged(object sender,
    EventArgs e)
73 {
74 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");

```

```

75
76 con.Open();
77 string com = "Select report_id,report_name from tbl_report ";
78 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
79 DataTable dt = new DataTable();
80 adpt.Fill(dt);
81 ddl_rept.DataSource = dt;
82 ddl_rept.DataTextField = "report_name";
83 ddl_rept.DataValueField = "report_id";
84 ddl_rept.DataBind();
85 ddl_rept.Items.Insert(0, new ListItem("select", "0"));
86 con.Close();
87 }
88
89 protected void ddl_rept_SelectedIndexChanged(object sender,
    EventArgs e)
90 {
91
92 if (ddl_rept.SelectedItem.Text == "project report" ||
    ddl_rept.SelectedItem.Text == "seminar report")
93 {
94 ddl_guide.Visible = true;
95 lbl_guide.Visible = true;
96 lbl_submit.Visible = false;
97 ddl_submitted.Visible = false;
98 lbl_sub.Visible = false;
99 ddl_sub.Visible = false;
100 ddl_sub.Visible = false;
101
102 }
103 else if (ddl_rept.SelectedItem.Text == "Assignment")
104 {
105 ddl_guide.Visible = false;
106 lbl_guide.Visible = false;
107 lbl_submit.Visible = true;
108 lbl_sub.Visible = true;
109 ddl_sub.Visible = true;
110 ddl_sub.Visible = true;
111 ddl_submitted.Visible = true;
112 }
113 else
114 {
115 ddl_guide.Visible = true;

```

```

116 lbl_guide.Visible = true;
117 lbl_submit.Visible = false;
118 ddl_submitted.Visible = false;
119 lbl_sub.Visible = false;
120 ddl_sub.Visible = false;
121 ddl_sub.Visible = false;
122 }
123 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
124
125 con.Open();
126 string com = "Select staff_id,staff_name from tbl_staff where
    dept_id='"+ddl_dept.SelectedItem.Value+"' ";
127 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
128 DataTable dt = new DataTable();
129 adpt.Fill(dt);
130 ddl_guide.DataSource = dt;
131 ddl_guide.DataTextField = "staff_name";
132 ddl_guide.DataValueField = "staff_id";
133 ddl_guide.DataBind();
134 ddl_guide.Items.Insert(0, new ListItem("select", "0"));
135 con.Close();
136
137 SqlConnection con1 = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
138
139 con.Open();
140
141 string com1 = "Select staff_id,staff_name from tbl_staff
    where dept_id='"+ddl_dept.SelectedItem.Value+"' and active
    =1 ";
142 SqlDataAdapter adpt1 = new SqlDataAdapter(com1, con1);
143 DataTable dt1 = new DataTable();
144 adpt.Fill(dt1);
145 ddl_submitted.DataSource = dt1;
146 ddl_submitted.DataTextField = "staff_name";
147 ddl_submitted.DataValueField = "staff_id";
148 ddl_submitted.DataBind();
149 ddl_submitted.Items.Insert(0, new ListItem("select", "0"));
150 con.Close();
151
152 }
153

```

```

154 protected void ddl_guide_SelectedIndexChanged(object sender,
        EventArgs e)
155 {
156
157 }
158
159 protected void ddl_submitted_SelectedIndexChanged(object
        sender, EventArgs e)
160 {
161 SqlConnection con = new SqlConnection("Data Source=.;Initial
        Catalog=dossier;User ID=sa;Password=admin123");
162
163 con.Open();
164 string com = "Select sub_id,sub_name from tbl_sub where
        cors_id='"+ddl_cors.SelectedItem.Value+"' and sem_name='"+
        ddl_sem.SelectedItem.Text+"' ";
165 SqlDataAdapter adpt = new SqlDataAdapter(com, con);
166 DataTable dt = new DataTable();
167 adpt.Fill(dt);
168 ddl_sub.DataSource = dt;
169 ddl_sub.DataTextField = "sub_name";
170 ddl_sub.DataValueField = "sub_id";
171 ddl_sub.DataBind();
172 ddl_sub.Items.Insert(0, new ListItem("select subject", "0"));
173 con.Close();
174 }
175
176 protected void ddl_sub_SelectedIndexChanged(object sender,
        EventArgs e)
177 {
178
179 }
180
181 protected void Submit_Click(object sender, EventArgs e)
182 {
183 string filename = Path.GetFileName(FileUpload1.PostedFile.
        FileName);
184 string contentType = FileUpload1.PostedFile.ContentType;
185 FileUpload1.SaveAs(Server.MapPath("~/Edocuments/") + filename
        );
186
187 using (Stream fs = FileUpload1.PostedFile.InputStream)
188 {

```



```

189 using (BinaryReader br = new BinaryReader(fs))
190 {
191 byte[] bytes = br.ReadBytes((Int32)fs.Length);
192 SqlConnection con = new SqlConnection("Data Source=.;Initial
    Catalog=dossier;User ID=sa;Password=admin123");
193 con.Open();
194
195 if (ddl_rept.SelectedItem.Text == "project report" ||
    ddl_rept.SelectedItem.Text == "seminar report" || ddl_rept
    .SelectedItem.Text=="mini project")
196 {
197 int stud_id = Convert.ToInt32(Session["stud_id"].ToString());
198 string stud_name =(Session["stud_name"].ToString());
199 string sql = "insert into tbl_report_sub(dept_id,dept_name,
    cors_id,cors_name,sem_id,sem_name,batch_id,batch_name,
    report_id,report_name,staff_id,topic,c_typ,data,filename,
    date_time,active,uploaded_by,staff_name,uploadedby_name)
    values('" + ddl_dept.SelectedItem.Value + "','" + ddl_dept
    .SelectedItem.Value + "','" + ddl_cors.SelectedItem.Value
    + "','" + ddl_cors.SelectedItem.Text + "','" + ddl_sem.
    SelectedItem.Value + "','" + ddl_sem.SelectedItem.Text + "
    ','" + ddl_batch.SelectedItem.Value + "','" + ddl_batch.
    SelectedItem.Text + "','" + ddl_rept.SelectedItem.Value +
    "','" +ddl_rept.SelectedItem.Text+"','" + ddl_guide.
    SelectedItem.Value + "','" + txt_topic.Text + "','" +
    FileUpload1.FileContent + "','" + FileUpload1.FileBytes +
    "','" + FileUpload1.FileName + "','" + (GETDATE()),0,'" +stud_id
    + "','" +ddl_guide.SelectedItem.Text+"','" +stud_name+"')";
200 SqlCommand cmd = new SqlCommand(sql, con);
201 cmd.ExecuteNonQuery();
202 ddl_dept.ClearSelection();
203 ddl_cors.ClearSelection();
204 ddl_batch.ClearSelection();
205 ddl_sem.ClearSelection();
206 ddl_sub.ClearSelection();
207 ddl_rept.ClearSelection();
208 ddl_guide.ClearSelection();
209 txt_topic.Text= "";
210 Response.Write("<script>alert('Successfully Inserted')</
    script>");
211
212 }
213 else

```

```

214 {
215 int stud_id = Convert.ToInt32(Session["stud_id"].ToString());
216 string stud_name = (Session["stud_name"].ToString());
217 string sql = "insert into tbl_report_sub(dept_id,dept_name,
        cors_id,cors_name,sem_id,sem_name,batch_id,batch_name,
        report_id,report_name,staff_id,topic,c_typ,data,filename,
        date_time,sub_id,sub_name,active,uploaded_by,staff_name,
        uploadedby_name)values('" + ddl_dept.SelectedItem.Value +
        "',' + ddl_dept.SelectedItem.Value + "',' + ddl_cors.
        SelectedItem.Value + "',' + ddl_cors.SelectedItem.Text +
        "',' + ddl_sem.SelectedItem.Value + "',' + ddl_sem.
        SelectedItem.Text + "',' + ddl_batch.SelectedItem.Value +
        "',' + ddl_batch.SelectedItem.Text + "',' + ddl_rept.
        SelectedItem.Value + "',' + ddl_rept.SelectedItem.Text +
        "',' + ddl_submitted.SelectedItem.Value + "',' +
        txt_topic.Text + "',' + FileUpload1.FileContent + "',' +
        FileUpload1.FileBytes + "',' + FileUpload1.FileName + "
        ',(GETDATE()),' + ddl_sub.SelectedItem.Value + "',' +
        ddl_sub.SelectedItem.Text + "','0,' + stud_id + "',' +
        ddl_submitted.SelectedItem.Value + "',' + stud_name + '"
        ";
218 SqlCommand cmd = new SqlCommand(sql, con);
219 cmd.ExecuteNonQuery();
220
221 Response.Write("<script>alert('Successfully Inserted')</
        script>");
222 ddl_dept.ClearSelection();
223 ddl_cors.ClearSelection();
224 ddl_batch.ClearSelection();
225 ddl_sem.ClearSelection();
226 ddl_sub.ClearSelection();
227 ddl_rept.ClearSelection();
228 ddl_guide.ClearSelection();
229 txt_topic.Text = "";
230 ddl_submitted.ClearSelection();
231 }
232
233
234 }
235 }
236 }
237
238 protected void ddl_dept_SelectedIndexChanged1(object sender,

```

```

        EventArgs e)
239 {
240     SqlConnection con = new SqlConnection("Data Source=.;Initial
        Catalog=dossier;User ID=sa;Password=admin123");
241
242     con.Open();
243     string com = "Select cors_id,cors_name from tbl_cors where
        dept_id='" + ddl_dept.SelectedItem.Value + "'";
244     SqlDataAdapter adpt = new SqlDataAdapter(com, con);
245     DataTable dt = new DataTable();
246     adpt.Fill(dt);
247     ddl_cors.DataSource = dt;
248     ddl_cors.DataTextField = "cors_name";
249     ddl_cors.DataValueField = "cors_id";
250     ddl_cors.DataBind();
251     ddl_cors.Items.Insert(0, new ListItem("select", "0"));
252     con.Close();
253 }
254 }
255 }

```

---