SHAHEER FAROOQ AZIZ & HAFIZ MUHAMMAD AHMAD
01-134132-172 & 01-134132-048

Thesis Management Automation

Bachelor of Science in Computer Science

Supervisor: Dr. Muhammad Muzammal

Department of Computer Science
Bahria University, Islamabad

June 2017

© Shaheer Farooq Aziz & Hafiz Muhammad Ahmad,June 2017

Certificate

We accept the work contained in the report titled "Thesis Management Automation", written by Hafiz Muhammad Ahmad and Shaheer Farooq Aziz as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Computer Science.

Approved by . . . :

Supervisor: Dr. Muhammad Muzammal (Associate Lecturer)

_____

Internal Examiner: Dr. Muneeb Gohar (Assistant Professor)

_____

External Examiner: Dr. Salabat Khan

_____

Project Coordinator: Dr. Arif Ur Rahman (Assistant Professor)

_____

Head of the Department: Dr. Faisal Bashir (Associate Professor)

_____

June 2017

Abstract

Management system are going to be used in probably every field of this world which gives the user opportunity to manage their daily routine tasks in a better way. Most of the management systems are being used in organizations or universities. To manage every process through management system makes life easier and more secure.

In this project, we developed a Management system based portal known as Thesis Evaluation Automation that is going to be used to automate the manual thesis system and will be integrated with the university campus management system. Thesis management system allows interested students to submit proposals and suggest supervisor first and then that data is acknowledged in university database. Admin arrange defense for proposals submitted by students. Defense evaluation forms are submitted by examiners appointed by admin. Supervisors are appointed after successful defense and submit four bimonthly reports form. After four bimonthly reports, students can submit thesis. Admin arrange defense for thesis submitted by students. Defense Evaluation by examiners updates the student thesis status.

Two types of users must interact with system. Students and Admin/DD-PGP have access to this system. Examiner and supervisor can submit evaluation forms.

Acknowledgments

First of all, thanks to Almighty ALLAH who gave us opportunity and courage to study and get knowledge about domain of this project which helped us in completing the project. Throughout this project, we faced a lot of ups and downs which help us to learn more. We are really thankful to our supervisor Dr. Muhammad Muzammal, Director(PGP) Dr. Shehzad Khalid, Director(IT) Rizwan Aamir, friends and our parents because without their prays contribution and love, we could not be able to think about the success we achieved in the shape of this project. we special thank our supervisor Dr. Muhammad Muzammal for their great support during all phases of this project because he helped us a lot during the time of difficulties. We would also like to thank all our teachers who taught us in best manner during the degree. In the end, we offer our best regard and blessing to all those individuals who supported and encouraged us during the completion of the project.

DIRECTOR(PGP) DR. SHEHZAD KHALID
DIRECTOR(IT) RIZWAN AAMIR
SHAHEER FAROOQ AZIZ
HAFIZ MUHAMMAD AHMAD
Islamabad, Pakistan

June 2017

"Keep your face always towards the sunshine and shadows will fall behind you."

Walter White

Contents

1 Introduction **1**
- 1.1 Problem Description . . . 1
- 1.2 Project Objectives . . . 2
- 1.3 Project Scope . . . 2
- 1.4 Resource Requirement . . . 2

2 Literature Review **3**
- 2.1 Management Information System . . . 3
- 2.2 Educational Management Information System . . . 4
  - 2.2.1 Explosive growth of educational system . . . 4
  - 2.2.2 Low Cost Information Technology . . . 4
- 2.3 BU - Campus Management System . . . 4

3 Requirement Specifications **5**
- 3.1 Manual Thesis Process . . . 5
- 3.2 Thesis Management Automation (TMA) . . . 6
- 3.3 Requirement Specifications . . . 6
  - 3.3.1 Functional Requirement . . . 7
  - 3.3.2 Non-functional Requirement . . . 7
- 3.4 Software Requirements . . . 8
- 3.5 Hardware Requirements . . . 8
  - 3.5.1 Client side hardware requirements . . . 8
  - 3.5.2 Server side hardware requirements . . . 8
- 3.6 Use Cases . . . 9
  - 3.6.1 Usecase 1 - User Authentication . . . 9
  - 3.6.2 Usecase 2 - Proposal submission and defense setup . . . 10
  - 3.6.3 Usecase 3 - Admin Setup proposal/thesis Defense . . . 11
  - 3.6.4 Usecase 4 - Examiner submit evaluation form . . . 12
  - 3.6.5 Usecase 5 - Supervisor submit bimonthly report . . . 13

4 Design **15**
- 4.1 System Architecture . . . 15
- 4.2 Design Constraints . . . 16
- 4.3 Design Methodology . . . 17

4.4 High Level Design . . . 17
4.5 Process . . . 17
4.6 Database Design . . . 18
4.7 GUI Design . . . 22

5 System Implementation **25**
5.1 System Architecture . . . 26
5.2 System Internal Components . . . 26
5.2.1 Student Panel . . . 27
5.2.2 Admin Panel . . . 27
5.2.3 Sql Server Database . . . 27
5.2.4 Examiner Evaluation Form . . . 27
5.2.5 Supervisor Bimonthly Evaluation Report . . . 27
5.3 Tools and Technology Used . . . 27
5.3.1 Microsoft Visual Studio . . . 27
5.3.2 SQL Server Management Studio . . . 28
5.4 Development Environment/Languages Used . . . 28
5.5 Processing Logic Flow . . . 29
5.6 Database Security Measures . . . 30

6 System Testing and Evaluation **31**
6.1 Graphical user interface testing . . . 31
6.2 Usablility Testing . . . 41
6.2.1 User friendly Interface . . . 41
6.2.2 Easy to Use . . . 41
6.2.3 Easy to Learn . . . 41
6.3 Compatibility Testing . . . 41
6.3.1 Device Features . . . 41
6.3.2 Operating System . . . 42
6.3.3 Browser Compatibility . . . 42
6.4 Platform Version . . . 42
6.5 Security Testing . . . 43
6.6 Installation Testing . . . 43
6.6.1 Testcase-1 (Login Verification Testing) . . . 43
6.6.2 Testcase-2 (Profile Information Testing) . . . 44
6.6.3 Testcase-3 (Upload Proposal Validation Testing) . . . 44
6.6.4 Testcase-4 (Proposal Defense Detail Testing) . . . 44
6.6.5 Testcase-5 (Update Proposal Testing) . . . 45
6.6.6 Testcase-6 (Admin Panel-Student info Testing) . . . 45
6.6.7 Testcase-7 (Setup Proposal Defense Validation Testing) . . . 45
6.6.8 Testcase-8 (Proposal Defense Details Testing) . . . 46
6.6.9 Testcase-9 (Examiner-Email Verification Testing) . . . 46
6.6.10 Testcase-10 (Student-Email Verification Testing) . . . 46
6.6.11 Testcase-11 (Proposal Status Testing) . . . 47
6.6.12 Testcase-12 (Upload Thesis Testing) . . . 47

6.6.13 Testcase-13 (Thesis Details Verification Testing) 47

7 Conclusions **49**

References **51**

List of Figures

3.1 Usecase Diagram . . . 9
3.2 User Authentication . . . 9
3.3 Proposal submission and defense setup . . . 10
3.4 Admin setup Proposal/Thesis Defense . . . 11
3.5 Examiner submit evaluation form . . . 12
3.6 Supervisor bimonthly report form . . . 13

4.1 System Architecture Diagram of the application . . . 16
4.2 Methodology Diagram of The Application . . . 17
4.3 Sequence Diagram of The Application . . . 18
4.4 Database Design First Diagram . . . 19
4.5 Database Design Second Diagram . . . 20
4.6 Database Design Third Diagram . . . 21
4.7 Student Panel Interface . . . 22
4.8 Admin Panel Interface . . . 23

5.1 System Architecture Diagram of Application . . . 25
5.2 System Internal Diagram of Application . . . 26
5.3 Processing Logic Flow Diagram of Application . . . 30

6.1 Student Login Panel . . . 31
6.2 Student Profile Panel . . . 32
6.3 Student Submit Proposal . . . 32
6.4 Student Proposal Details . . . 33
6.5 Admin Login Panel . . . 33
6.6 Admin Profile . . . 34
6.7 Admin Setup Proposal Defense . . . 34
6.8 Admin View Proposal Defense Details . . . 35
6.9 Student View Proposal Defense Details . . . 35
6.10 Student Upload Thesis . . . 36
6.11 Student Panel Thesis Details . . . 36
6.12 Thesis Outcome . . . 37
6.13 Proposal Defense Evaluation Form For Examiner . . . 38
6.14 Supervisor Bimonthly Evaluation Report Form . . . 39
6.15 Thesis Defense Evaluation Form For Examiner . . . 40

List of Tables

3.1 Software Requirements 8
3.2 Hardware Requirements-Client Side 8
3.3 Hardware Requirements-Server Side 8
3.4 UseCase2 10
3.5 UseCase3 11
3.6 UseCase4 12
3.7 UseCase5 12
3.8 UseCase6 13

6.1 Verification Testing 43
6.2 Profile Information Testing 44
6.3 Upload Proposal Validation Testing 44
6.4 Proposal Defense Detail Testing 44
6.5 Update Proposal Testing 45
6.6 Admin Panel-Student info Testing 45
6.7 Setup Proposal Defense Validation Testing 45
6.8 Proposal Defense Details Testing 46
6.9 Examiner-Email Verification Testing 46
6.10 Student-Email Verification Testing 46
6.11 Proposal Status Testing 47
6.12 Upload Thesis Testing 47
6.13 Thesis Details Verification Testing 47

Acronyms and Abbreviations

TMA	Thesis Management Automation
GUI	Graphical User Interface
IDE	Integrated Development Environment
XML	Extensible Markup Language
CSS	Cascading Style Sheets
HTML	Hypertext Markup Language
EMIS	Educational Management System

Chapter 1

Introduction

Thesis management automation is a tool aimed to assist students, administration, supervisors and examiners in submission, evaluation and management of thesis respectively. It allows students to submit proposals and thesis. There are two panels in our system i.e. admin and student panel, admin panel is used to arrange two types of defense (proposal and thesis defense). Whereas, student panel is used by students to submit proposals/thesis and keep record of their evaluations which are submitted by examiners/supervisors [1].

Examiners and supervisors evaluate defenses and progress reports using web evaluation forms which are sent to them via email. Once evaluation forms are submitted, they can be viewed on admin panel. After submission of evaluation forms respective statuses are updated on student panel as well.

Bahria university manages thesis of students manually which includes a lot of paper work. The objective of this project is to automate the thesis management process which does not require any paper work and where students, supervisors and university administration can interact with the automated thesis management system with ease. This project was initiated on suggestion of Dr. Shehzad Khalid (Directorate Post Graduate Program) as a lot of human effort was required in handling thesis manually.

1.1 Problem Description

Major issue faced in using existing thesis management process is that it is very difficult for supervisors to keep record of students working on thesis. Manual process faces a lot of issues, one of them is misplacement of student thesis evaluation records. Manually handling student thesis is very time consuming process and is prone to errors since it involves a lot of human

effort. Apart from slow process, a lot of physical space is also required to store printed thesis which makes it less secure [2].

1.2 Project Objectives

This project automates the existing manual process of evaluating thesis at Bahria University. It is aimed to resolve problems being faced by university students, administration and faculty members who are responsible for evaluating student thesis. After the completion of project, it will be integrated with CMS (Campus Management System) of Bahria University.

1.3 Project Scope

Initially we're aiming to assist students, administration and faculty members at Bahria University Islamabad campus. Currently our project is automating thesis management process only, it does not perform plagiarism check on student thesis. With respect to time, we are hoping to incorporate this feature with our project as well.

1.4 Resource Requirement

Resources required in development process are given below

1. Microsoft Visual Studio (2010 or above).
2. Microsoft SQL Server Management Studio (2008 or above).
3. Dataset for testing purpose.
4. Format of Evaluation forms at Bahria University.

Chapter 2

Literature Review

In this chapter, all the concepts, theories and methods that are related to management information system will be discussed. The concept of EMIS (Educational management information system) in the world is thoroughly presented in this chapter with special focus of this concept in Pakistan. Detailed literature review will help us in developing our project as per concept of EMIS used worldwide.

There have been numerous kind of systems that have been developed over the past several years. These information systems helped to fulfil the needs and requirements of decision making not only at managerial but also at the operational level. Every organization develops its own management information system (MIS) which is totally dependent on the specific needs of organizations.

In the management information system, not only the system itself is important but to get the maximum advantages from the system it is important that the human intelligence, perception and judgement must be powerful and strong enough to get combined with the system information [3].

This combination will provide managers with the unique and valuable tool for the information management in any company.

Management system in this literature uses technique of having the directly accessible secure storage with well managed database.

2.1 Management Information System

Management Information system facilitates your needs. It is a system which may vary in every organization according to their needs or requirements [4]. A system where data is collected and routed in desired locations where it is supposed to be and manage the data with getting

desired results can be referred as management Information system. This gives us the idea to first make the database as per our desired need. Testing that database with sample data is called dry run. Database is most essential and important part in any management system. Second thing we need to focus on is getting appropriate requirements. For this purpose, we need to be in constant touch with administration.

2.2 Educational Management Information System

In most of the universities EMIS(educational management system) is needed because of the following reasons

2.2.1 Explosive growth of educational system

This problem mentions the point that with growing student strength, management system in developing countries are compulsory to handle that much burden. This gives us the idea to make our management system power enough to cater with the university growing student strength and make it as efficient to use as we can [5].

2.2.2 Low Cost Information Technology

As technology is advancing now a day and we have lot of opportunity to get hardware as per our requirements. So this literature gives us the idea of having good hardware at low cost.

2.3 BU - Campus Management System

Bahria university - campus management system is working on asp.net framework. Database used in this system is sql server. Bootstrap is used for front end interface to make it responsive and user friendly. Current system is working on layers which means that it can be accessible on different levels with having different confidential access.

Chapter 3

Requirement Specifications

In this chapter, description of the system being developed is presented. This chapter lays out functional and non-functional requirements and includes a set of use cases that describe user interactions that software must provide.

3.1 Manual Thesis Process

Manual thesis management process is entirely based on paper work, which may be vulnerable to human error. In existing system, students need to find out the supervisor for thesis and they have to communicate to supervisors by checking whether a supervisor is available or not. With a manual operating system, students rely on regular face to face contact with their supervisor which is basically very time consuming. The process of submitting proposal/thesis manually have a lot of paper work involved and also there is no proper place to save the students records. University management face difficulties with placement of student's thesis files. The main problems of existing system are following

1. A lot of paper work is required
2. Less Secure
3. Time consuming
4. No proper physical place to store student record
5. Human error chances
6. Very slow to operate

3.2 Thesis Management Automation (TMA)

The proposed system is being developed to overcome problems mentioned above i.e. To convert existing manual thesis management process to automated thesis management system, which involves no paper work. This objective can be achieved by making a user-friendly interface for admin and student so that both users can perform their tasks quickly and efficiently. In this project we are using the following technologies:

1. C#
2. AJAX
3. SQL Server Database
4. JavaScript
5. HTML
6. CSS

Thesis Evaluation Automation is designed to facilitate university administration, faculty and students. In this system, students can submit their proposals and proposal defenses are arranged by coordinator or university administration. Examiners submit defense evaluation forms after once defense has taken place. After successful completion of thesis, supervisors are assigned to respective students and proposal status is updated on student panel. Supervisors submit bimonthly reports using report forms which are sent to them via email. After all four bimonthly reports are submitted, student is allowed to submit thesis. Coordinator will arrange defense for thesis and again the defense evaluation forms are submitted by examiners appointed in defense. Thesis status on student panel is updated per evaluation.
The main objective of this application is to facilitate the students and faculty members of Bahria University. This application allows the Faculty members to save students records in database in an organized way and get rid of manual evaluation process. This results in less usage of human resources, and automated process is more secured and fast.

3.3 Requirement Specifications

Requirement specification is a description of all the functional and non-functional requirements of a system. In the sections below we have discussed both functional and non-functional requirements of our system.

3.3.1 Functional Requirement

Functional requirements describe the set of inputs, outputs and behavior of any system. It defines the main functions of a system and its core components. Given below are functional requirements of our system that are categorized as following.

1. Student Panel

   Student panel provides an option to upload proposal and thesis. Detailed description of proposal and thesis can be seen on student panel which includes supervisor's name, status whether supervisor has been assigned or not, proposal's name and proposal's status.

2. Admin Panel

   Admin/PGP Coordinator can view the list of students who have submitted their proposals/thesis.

   Coordinator has an option to arrange defense for proposals and thesis by appointing examiners.

3. Defense Evaluation

   Examiners appointed by Coordinator submit defense evaluation forms sent to them via email.

4. Bimonthly Reports

   Supervisor submits bimonthly reports of student progress.

3.3.2 Non-functional Requirement

These requirements are used to judge the operation of a system, rather than behavior of system. The application must satisfies the following non-functional requirements which are discussed below

1. Efficiency - Application needs to be efficient by using less resources of server and client systems.
2. Maintainability - Application needs to be maintainable.
3. Reliability - Applications should always be available.
4. Security - Data of students should be secured and should only be visible to concerned personnel.

5. Attractive layout - Application layout needs to be attractive and user friendly.

6. Usability- System needs to be usable

3.4 Software Requirements

The software requirements for this application is shown in Table 3.1.

Development Tool	**Operating System**	**Database System**
Visual Studio 2010 or above	Windows XP, 7, 8, 8.1, 10	Microsoft SQL Server
Microsoft Sql Server Management Studio	Windows XP, 7, 8, 8.1, 10	Microsoft SQL Server

Table 3.1: Software Requirements

3.5 Hardware Requirements

Two types of hardware is required i.e. one for client and other one for server.

3.5.1 Client side hardware requirements

Hardware required on client side is given below in table 3.2

Device	Desktop or Laptop
Processor	Dual Core 1.2 or above
Memory	1GB or above
Hard Drive Capacity	20GB or above

Table 3.2: Hardware Requirements-Client Side

3.5.2 Server side hardware requirements

Hardware required on server side is given below in table 3.3

Device	Desktop or Laptop
Processor	Dual Core 1.6 or above
Memory	2GB or above
Hard Drive Capacity	40GB or above

Table 3.3: Hardware Requirements-Server Side

3.6 Use Cases

A use case is a graphical representation of actions or events which define the interaction between the actors and system to achieve a goal. The main use case diagram of our system is shown in Figure 3.1. The actors in this use-cases are the people who will be interacting with the system

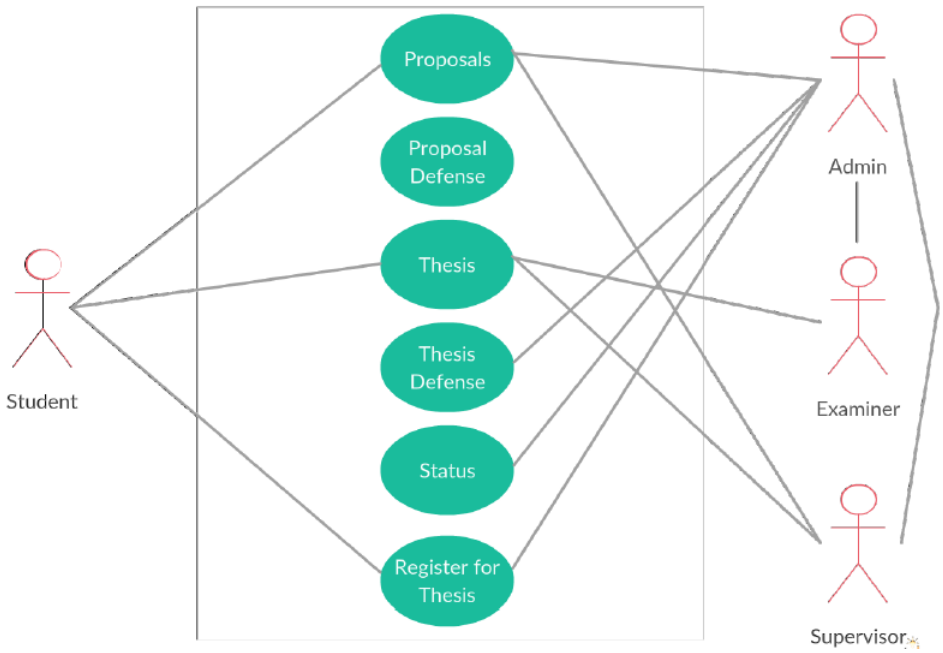


Figure 3.1: Usecase Diagram

3.6.1 Usecase 1 - User Authentication

User needs to login to system using valid user credentials

Figure 3.2: User Authentication

Use Case id	UC-1
Title	User Authentication
Description	User login to system using valid credentials
Primary Actor	Student/Admin
Pre Condition	Username and Password boxes are empty and internet connection is stable
Post Condition	User logged in
Success scenario	User view personal profile
Exceptions	User enter wrong credentials are avoided
Assumptions	User have valid username and password

Table 3.4: UseCase2

3.6.2 Usecase 2 - Proposal submission and defense setup

Student submit proposal and defense arranged by admin should visible on both student and admin panel.

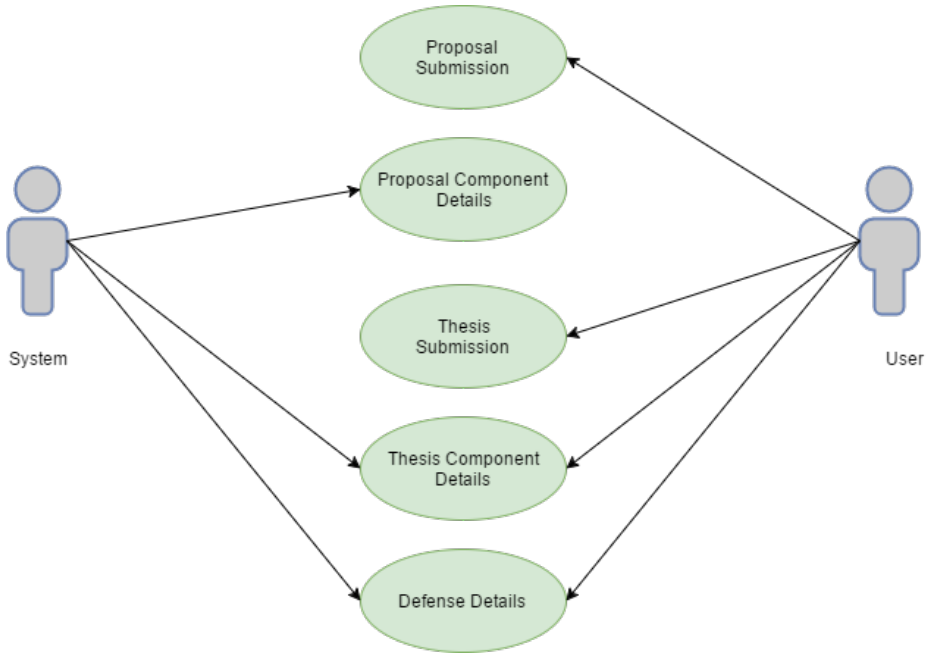


Figure 3.3: Proposal submission and defense setup

Use Case id	UC-2
Title	Proposal submission and defense setup
Description	Student submit proposal and admin setup defense
Primary Actor	Student and Admin
Pre Condition	No proposal submitted
Post Condition	Proposal submitted by student and defense details shown on both student/admin panels
Success scenario	Student/Admin view defense details panel
Exceptions	No setup details updated after defense setup
Assumptions	User have valid proposal file to submit

Table 3.5: UseCase3

Pre condition describes that no proposal is submitted and proposal details panel is hidden on both student and client side.

3.6.3 Usecase 3 - Admin Setup proposal/thesis Defense

After successful defense setup, defense details panel must visible with accurate defense details.

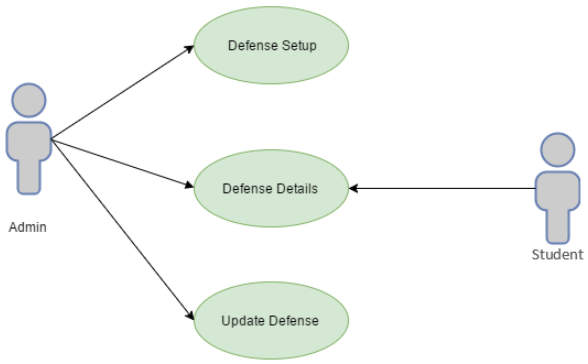


Figure 3.4: Admin setup Proposal/Thesis Defense

Use Case id	UC-3
Title	Admin setup Proposal/Thesis Defense
Description	After submission of proposal or thesis admin setup defense
Primary Actor	Admin
Pre Condition	No Defense Setup
Post Condition	Defense details panel updated
Success scenario	Defense setup done and defense details panel updated
Exceptions	No detail panel updated after defense
Assumptions	Admin setup defense for proposal/thesis

Table 3.6: UseCase4

3.6.4 Usecase 4 - Examiner submit evaluation form

When examiner submits evaluation form, records are submitted and link is destroyed.

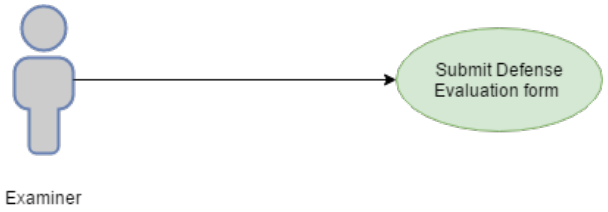


Figure 3.5: Examiner submit evaluation form

Use Case id	UC-4
Title	Examiner submit evaluation form
Description	Examiner evaluates the proposal/thesis in defense and submit evaluation form
Primary Actor	Examiner
Pre Condition	All fields are empty
Post Condition	Evaluation form submitted and link is destroyed
Success scenario	Record submitted and Link destroyed
Exceptions	No form submitted
Assumptions	Examiner took defense before evaluation

Table 3.7: UseCase5

3.6.5 Usecase 5 - Supervisor submit bimonthly report

Supervisor submit bimonthly report and records are submitted successfully.

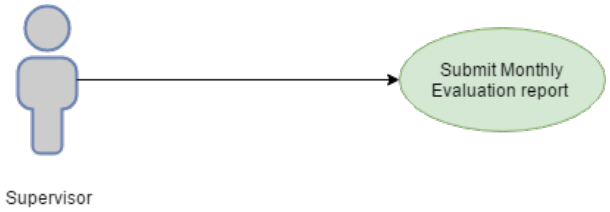


Figure 3.6: Supervisor bimonthly report form

Use Case id	UC-5
Title	Supervisor bimonthly report form
Description	Supervisor submit bimonthly reports of student
Primary Actor	Supervisor
Pre Condition	All fields are empty
Post Condition	Report form submitted and link is destroyed
Success scenario	Record submitted and Link destroyed
Exceptions	No form submitted
Assumptions	Supervisor assigned before report evaluation

Table 3.8: UseCase6

Chapter 4

Design

In this chapter, system design and diagrams (sequence, methodology, architecture and database) related to the design of our system will be discussed. The diagrams are used to explain the working and architecture of subsystems and their role with main system. Architecture, components, modules and interfaces will also be discussed in this chapter.

4.1 System Architecture

System Architecture is a high level logical representation of application which shows what components a system will have and how they are connected to each other. Figure 4.1 show that the system is web based application.

System architecture describes the functionality of the system and high level logical implementation of the system components. It also describes the way in which these components interact with each other. System architecture also explains the environment for the system in which system will be implemented. Our Web Application provides multiuser interface to student, admin directly and supervisor, examiner indirectly. System Architecture diagram shows at first step student submits the research proposal and in return status is updated once examiners has submitted the evaluation forms. Admin/coordinator at second step generate research proposal defense by appointing the examiners. All these tasks performed are recorded in the SQL Server database. System notify the student and examiners about the defense details and route the evaluation forms among the examiners. Supervisor enters in the system after successful proposal defense. When supervisor logs in the system, supervisor must submit four bimonthly forms to proceed further. After successful submission of all four bimonthly

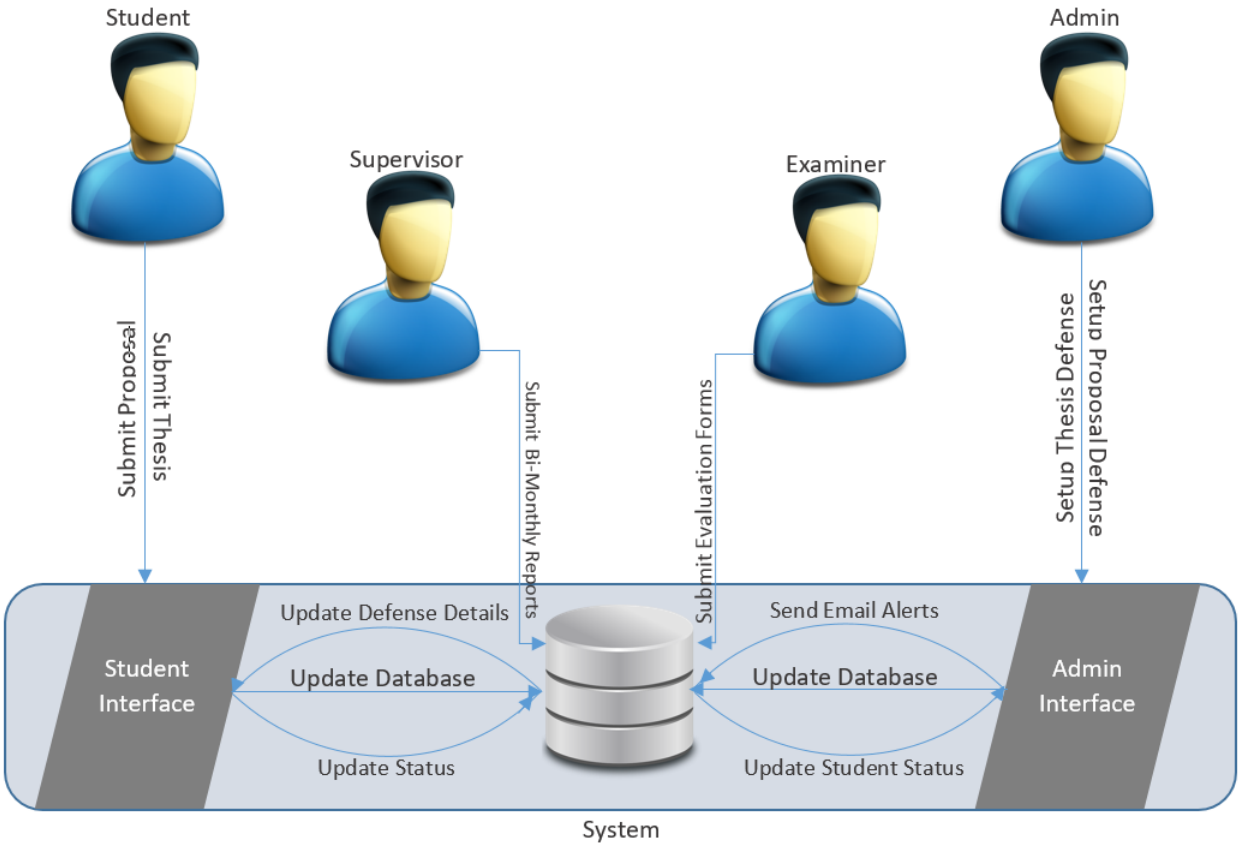


Figure 4.1: System Architecture Diagram of the application

evaluation forms, student can upload thesis before due date. System again ask admin to setup defense, but this time for thesis. System waits for examiners to submit defense evaluation forms. After successful submission, relevant users are notified by updating the thesis status.

4.2 Design Constraints

The main emphasis of developing this application is to overcome the issues faced in manual thesis management process. Students submit their proposal/thesis through Thesis Management Automation and their results can be viewed on this portal as well. This application is being developed using C# interface thus making certain changes in interface methodology is one of the major constraints. Providentially, using ASP.NET and SQL server database can achieve the desired objective.

4.3 Design Methodology

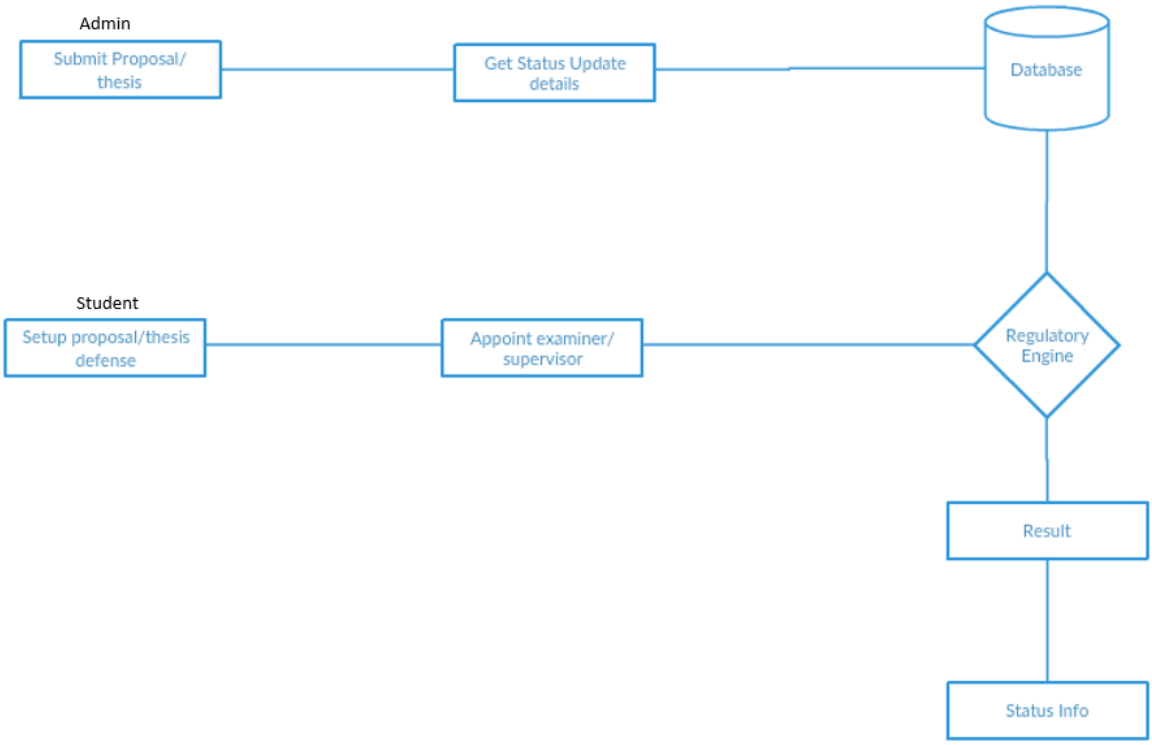


Figure 4.2: Methodology Diagram of The Application

Our system's methodology is expressed in Figure 4.2 where modules are working separately. First student submits the proposal or thesis and get proposal/thesis details panel on their screen. Secondly, admin setup proposal defense for proposal/thesis.

4.4 High Level Design

This section describes in further detail elements discussed in the previous section. Following are the different views of high level design.

4.5 Process

This is basic work flow of the system. It describes the run time view of the system. It shows the interaction of user with the system and inter-system interaction of different subsystems. Figure 4.3 shows the detailed process of interaction between the different components of the application.

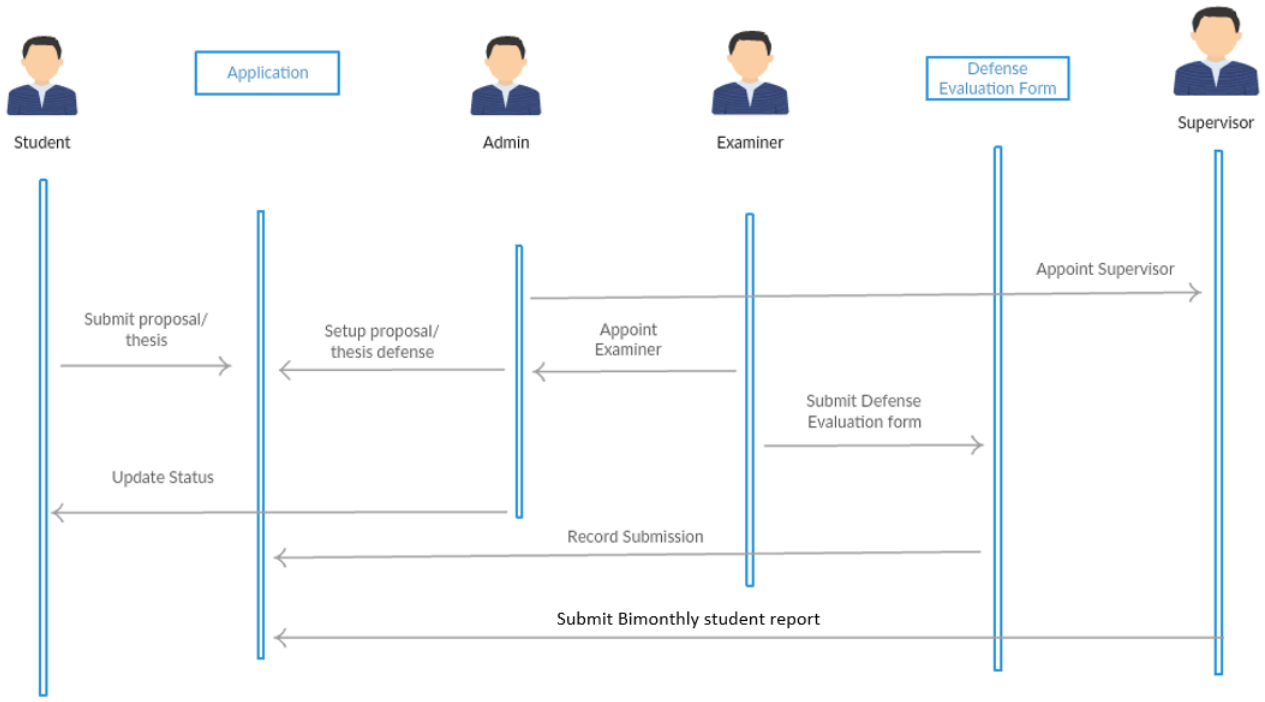


Figure 4.3: Sequence Diagram of The Application

4.6 Database Design

Process of producing a detailed data model of database is known as database design. This data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. Database design for our project is given in figures form 4.4 to 4.6. Figure 4.4 describes when student enters in our system, student's relation with thesis system is made by entering a new record using student enrollment. Proposal, thesis, frc approval and supervisor id's are added into the proposal thesis info table. Figure 4.5 shows defense generated by admin has examiner's id and each defense has specific type, whether it is proposal defense or thesis defense.Figure 4.6 describes the evaluation done by examiners is saved in separate table and id of evaluation result is inserted into defense evaluation details table. Progress reports submitted by supervisors are stored in table named progress report details. Each progress report submitted has unique supervisor id to identify supervisor.

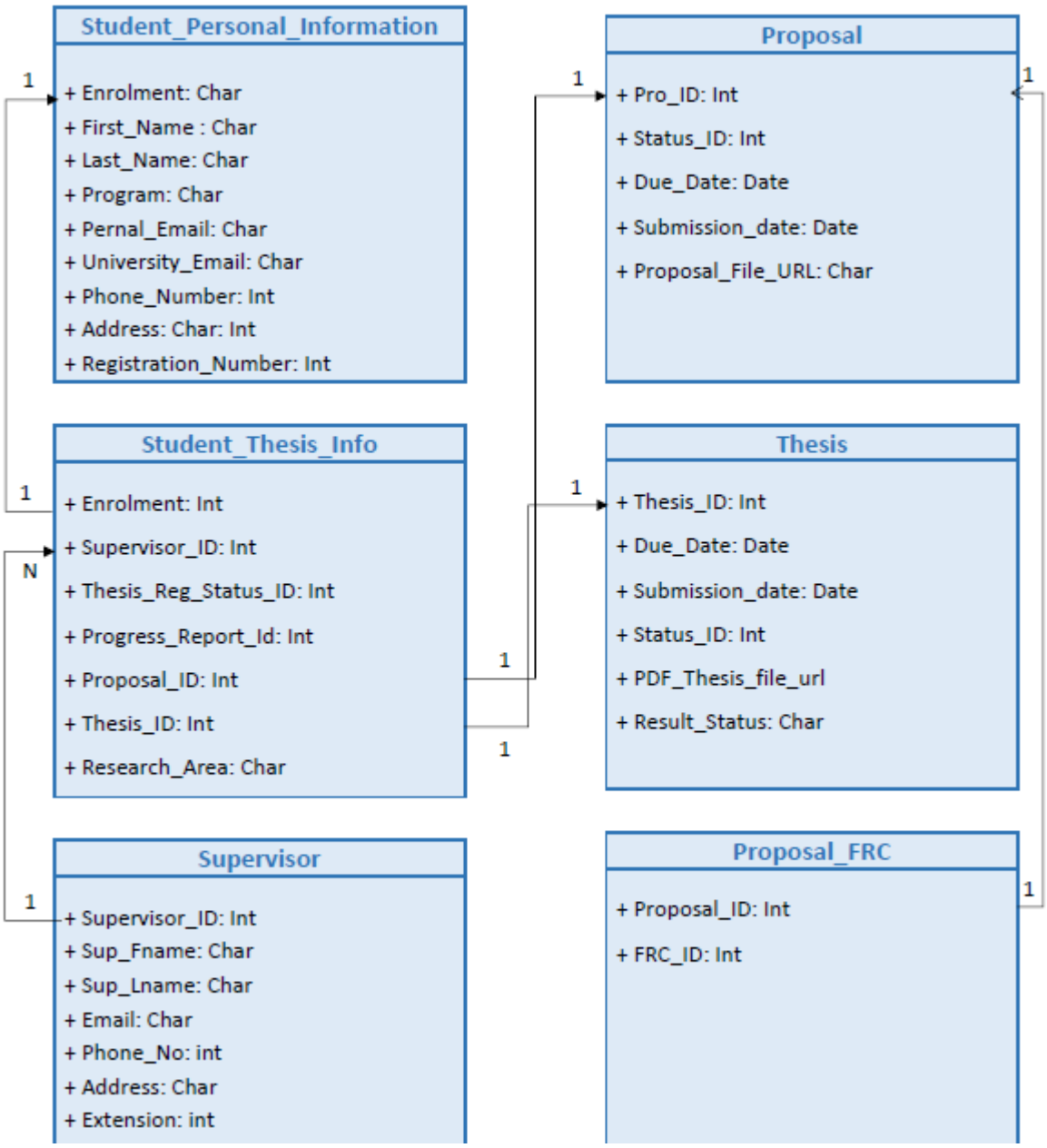


Figure 4.4: Database Design First Diagram

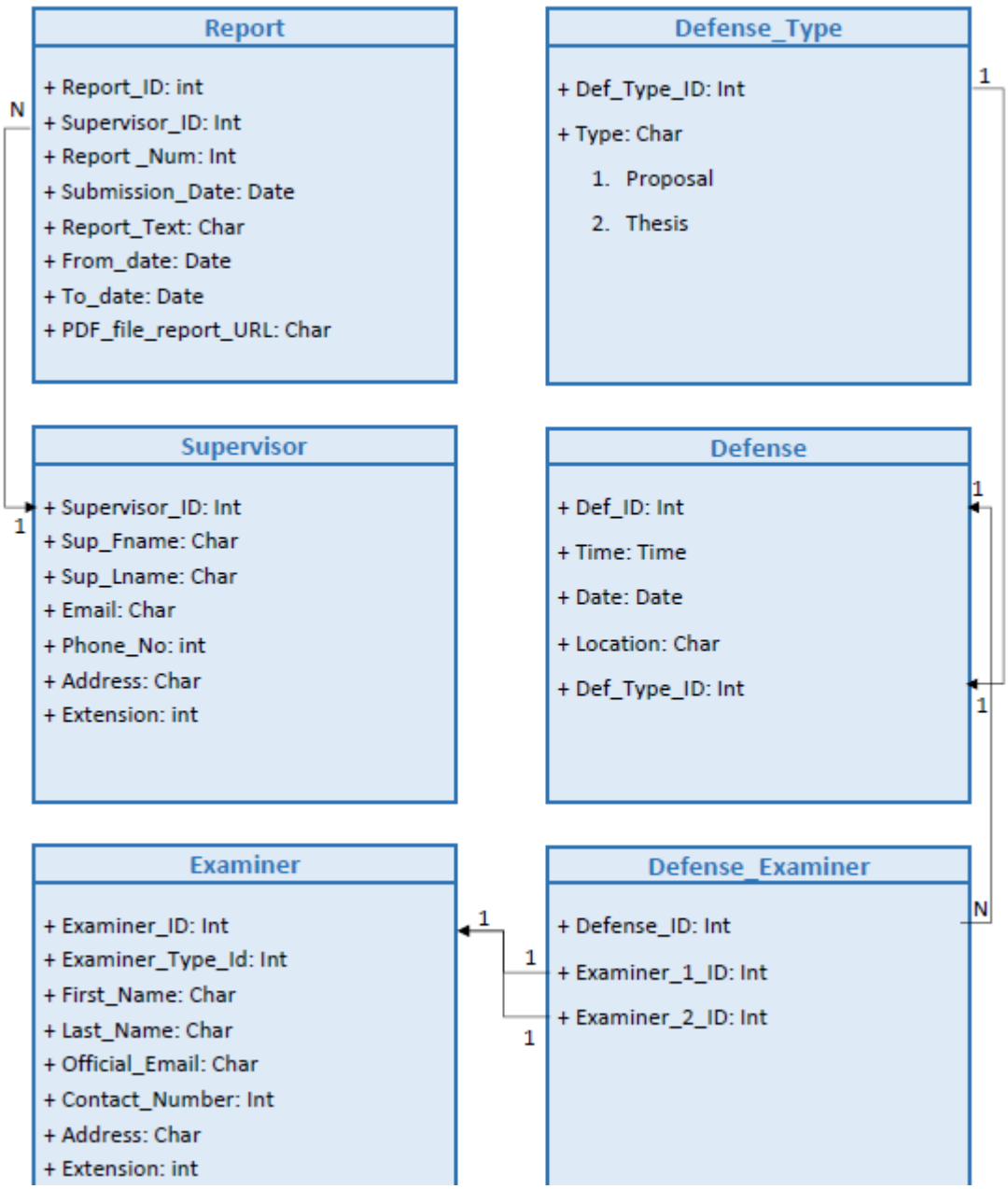


Figure 4.5: Database Design Second Diagram

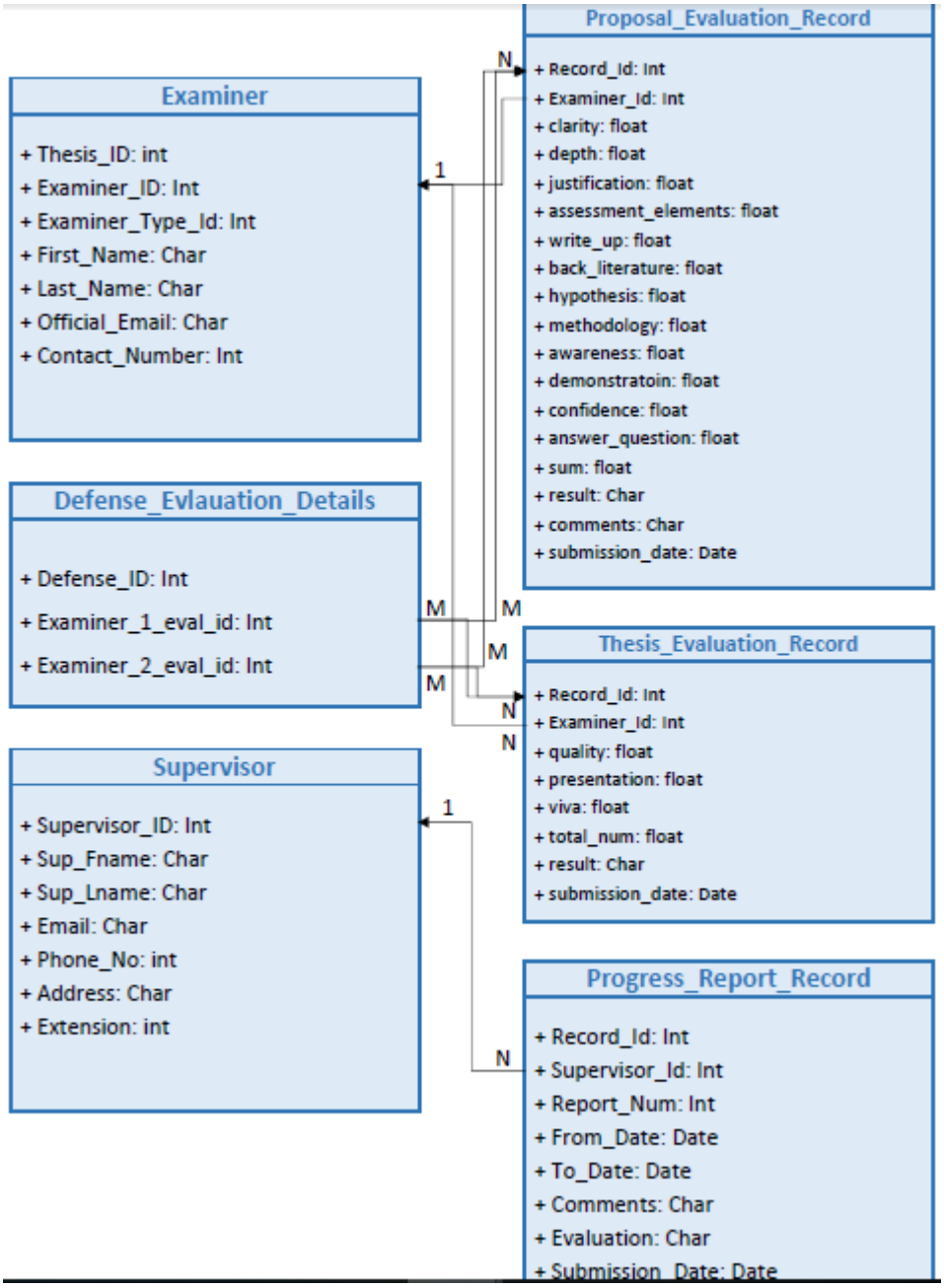


Figure 4.6: Database Design Third Diagram

4.7 GUI Design

The GUI (Graphical User Interface) of a system requires being simple and user friendly. Since our designed system will be used by students and faculty members instead of technical people, therefore the GUI has to be very user friendly but extremely comprehensive. Our GUI design guides the user through the system step by step by going from one activity to another activity. This chapter will explain the graphical user interface of the system we're developing. By taking advantage of an individual suggestions impression, we are designing graphical user interface quite simple yet appealing. We used linear page layout for most layouts to make it consistent.

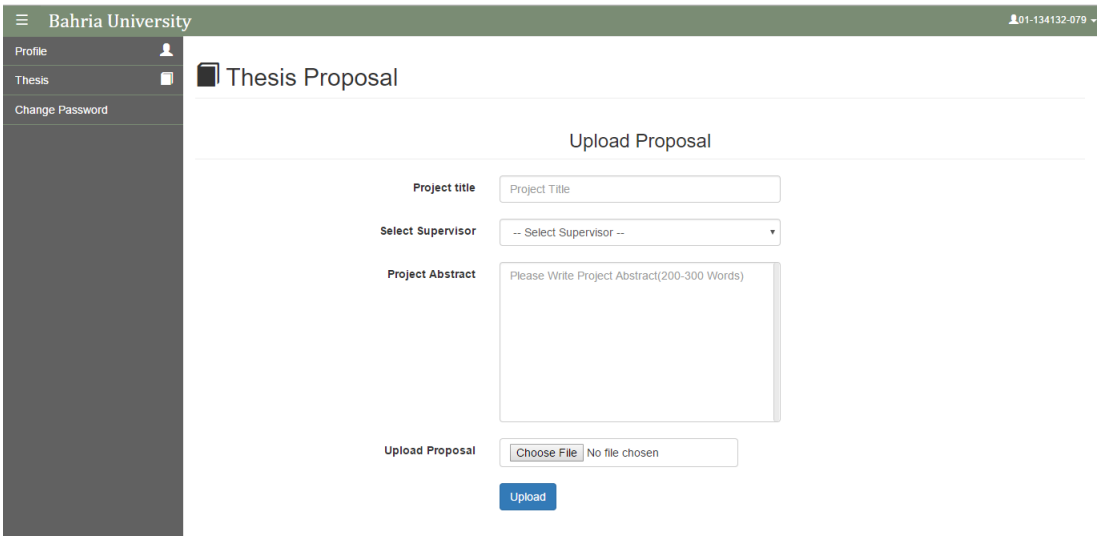


Figure 4.7: Student Panel Interface

Student panel interface with upload proposal option is shown in Figure 4.7

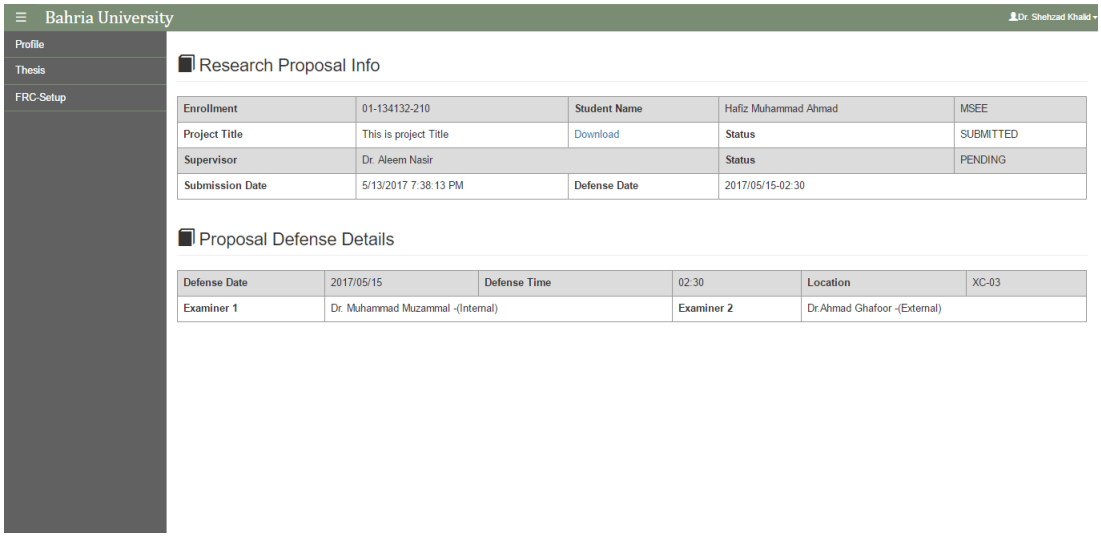


Figure 4.8: Admin Panel Interface

Admin panel interface with setup proposal defense is shown in Figure 4.8

Chapter 5

System Implementation

System Implementation is the process of defining system architecture and moving an idea form concept into reality. The system implementation is a discernment of algorithm and technical specifications. The application is web based for automating the thesis system of Bahria University. Our web application consists of database source based upon SQL Server relational database for data storage and retrieval. Figure 5.1 shows the architecture of our system.

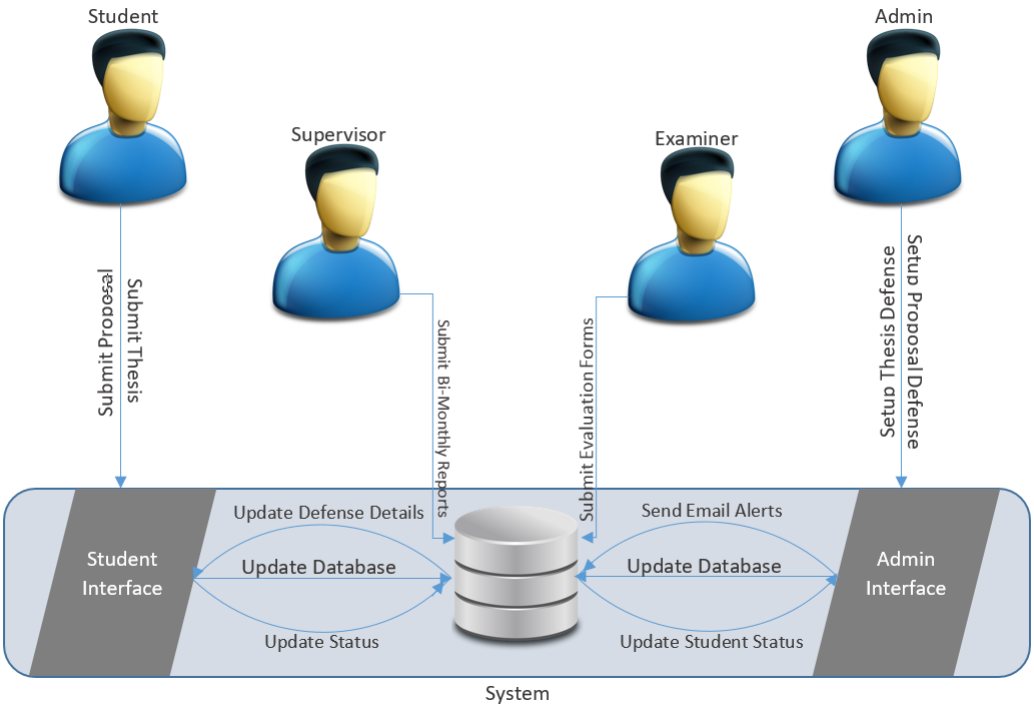


Figure 5.1: System Architecture Diagram of Application

5.1 System Architecture

System architecture describes the functionality of the system and high level logical implementation of the system components. It also describes the way in which these components interact with each other. System architecture also explain the environment for the system in which system will be implemented. Our application provides multiuser interface to student and admin directly and supervisor, examiner indirectly. System Architecture shows clearly that at first step student submits the Research Proposal and get status updated by the system. Admin at second step generate research proposal defense by appointing the examiners. All these tasks performed are recorded in the SQL Server database. System notifies the student and examiners about the defense details and sends the evaluation forms among the examiners. Supervisor enters in the system after successful proposal defense. When supervisor enters in the system, supervisor submits four bimonthly report forms to proceed further. After successful submission of all four bimonthly evaluation forms, student is allowed to upload thesis before due date. System again asks admin to setup defense, but this time for thesis. System waits for examiners to submit defense evaluation forms. After successful submission, relevant users are notified by updating the thesis status.

5.2 System Internal Components

Our application is divided into two major and two minor components which are linked with each other. Two major internal components are student panel and admin panel. Two minor internal components are examiner evaluation forms and supervisor's bimonthly evaluation reports. How these components are the linked and what are the processes in it are all shown in Figure 5.2 which displays the system Internal components.

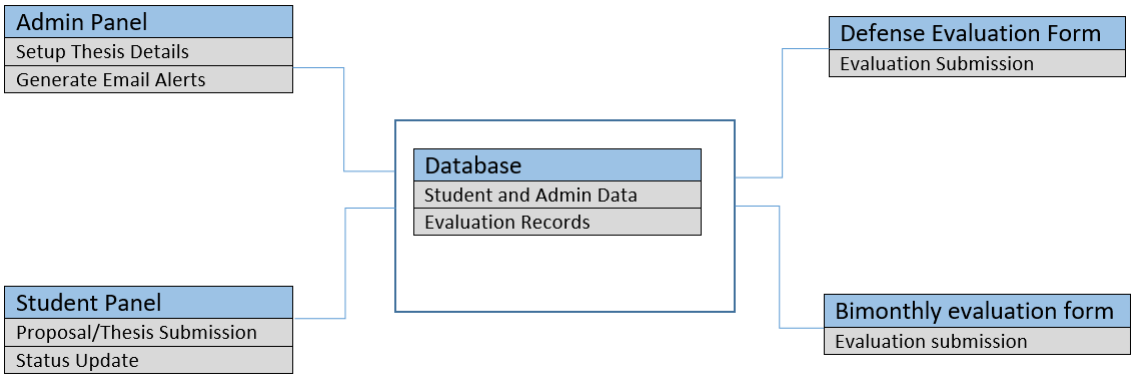


Figure 5.2: System Internal Diagram of Application

5.2.1 Student Panel

For Student Panel Asp.net functionalities are used to manipulate the data on single page. Student uploads proposal and thesis on student panel and gets details accordingly using Asp panel controls. Multiple Asp panel controls are used to show data which appears in appropriate panel. When data expires, panel's visible property is set to false.

5.2.2 Admin Panel

For admin panel web services are also used with asp.net controls to access data from database in the form of lists. JavaScript calendar property is used to select date time for defense purposes. Admin panel also used SQL server email services to send email notifications to concerned examiners/supeprvisors.

5.2.3 Sql Server Database

Sequential query language is used to insert and retrieve data from database. SQL server email services are used to send emails to notify the concerned persons. It is accessed by both the student and admin panel but managed by admin only.

5.2.4 Examiner Evaluation Form

Examiner evaluation forms used URL-rewriting to access the specific student records to be uploaded in the form. Ajax is used to restrict the text boxes to enter float values only.

5.2.5 Supervisor Bimonthly Evaluation Report

Examiner evaluation forms also using URL-rewriting to access the specific student records to be uploaded in the form. Evaluation records are submitted into database using sequential query language.

5.3 Tools and Technology Used

Two types of tools are used in developing this system which are given below.

5.3.1 Microsoft Visual Studio

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web

applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silver light. Thesis system application is implemented in version 2015.

5.3.2 SQL Server Management Studio

SQL Server Management Studio is an integrated environment for managing any SQL infrastructure, from SQL Server to SQL Database. It also provides tools to configure, monitor, and administer instances of SQL from wherever you deploy it. In our application 2008 version of this tool is used.

5.4 Development Environment/Languages Used

The development environment used for this application is Visual Studio, its version is Visual Studio 2015 64 bit.
This project has been developed using ASP.NET framework with multiple programming languages.

1. **C-sharp**: C-sharp is an object-oriented programming language from Microsoft that aims to combine the computing power of C++ with the programming ease of Visual Basic. C-sharp is based on C++ and contains features like those of Java. C-sharp is designed to work with Microsoft's .Net platform.

2. **Javascript**: It is a prototype-based, multi-paradigm scripting language that is dynamic, and supports object-oriented, imperative, and functional programming styles. JavaScript runs on the client side of the web, which can be used to design / program how the web pages behave on the occurrence of an event.

3. **Bootstrap**: It is a combination of HTML, CSS, and JavaScript code designed to help build user interface components. Bootstrap was also programmed to support both HTML5 and CSS3. Also, it is called Front-end-framework. Bootstrap is a free collection of tools for creating a websites and web applications.

4. **Latex**: Latex is a powerfully captured language engineered for technical documentation. LATEX is a portable document formatting system based on TEX, a typesetting language initially designed particularly for math and science. It is also a programming language, which means you can generate your own instructions to simplify and modify it. TEX and LATEX use by default a font family 21 called Computer Modern, which comprises

a variability of styles such as serif, sans serif, typewriter, and a mainly rich set of mathematical symbols.

5. **AJAX**: AJAX stands for Asynchronous JavaScript and XML. In a nutshell, it is the use of the XMLHttpRequest object to communicate with servers. It can send and receive information in various formats, including JSON, XML, HTML, and text files. AJAX's most appealing characteristic is its "asynchronous" nature, which means it can communicate with the server, exchange data, and update the page without having to refresh the page. The two major features of AJAX allow you to do the following:

   - Make requests to the server without reloading the page.
   - Receive and work with data from the server.

5.5 Processing Logic Flow

In our application logical flow starts from the process of submitting proposal by student which indicates admin to setup proposal defense for the student who submitted proposal. Examiners submits evaluation forms mailed by admin. Proposal status at Student panel is updated according to the evaluation reports. Supervisor is assigned by admin and asked to submit four bimonthly progress reports. After all bimonthly reports submission, student is allowed to upload thesis before due date. Again, admin arrange defense for thesis and assign examiners. Examiner evaluates final submission and updates the student's thesis result accordingly.

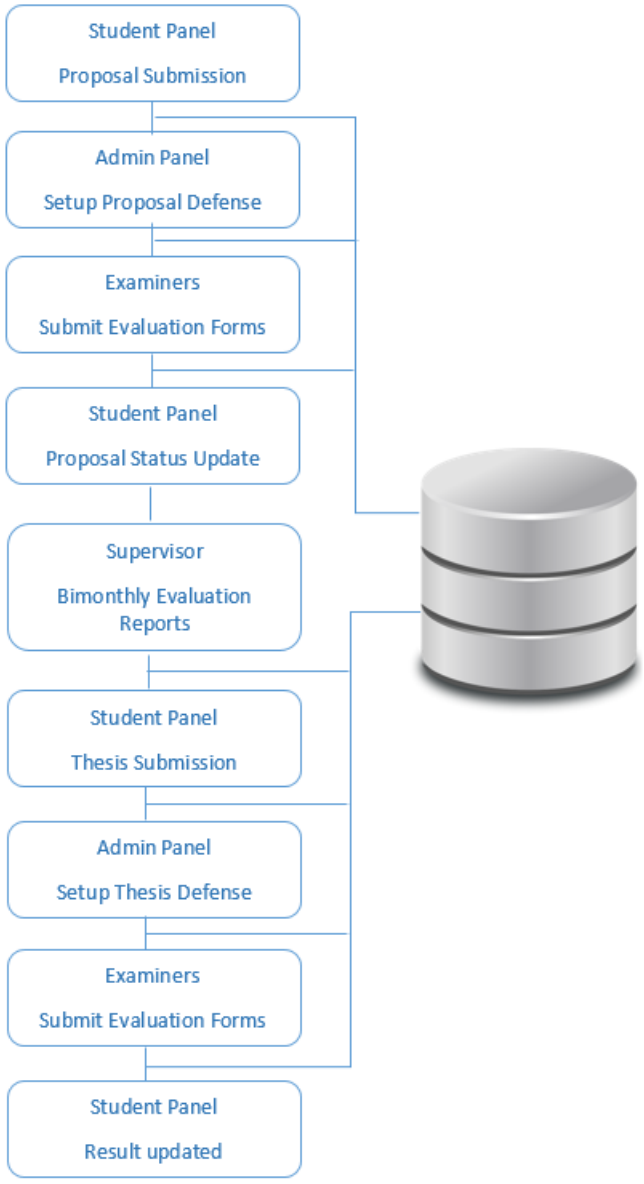


Figure 5.3: Processing Logic Flow Diagram of Application

5.6 Database Security Measures

In our system SQL server database is used for data storage. SQL server is not directly installed in project but separately on the server. SQL server is accessed by making connection with the database and after performing actions connection is closed. No unauthorized personnel can access the database without making connection using fake username and password.

Chapter 6

System Testing and Evaluation

In this chapter application performance and evaluation is going to be discussed. Evaluation of the system being developed is done by using various tools and techniques. This application is consists of four modules and each module is validated according to the working of system and requirements of the system.

6.1 Graphical user interface testing

One of the main four modules is student panel, where student needs to provide valid enrollment and password in order to log into the system as shown in Figure 6.1

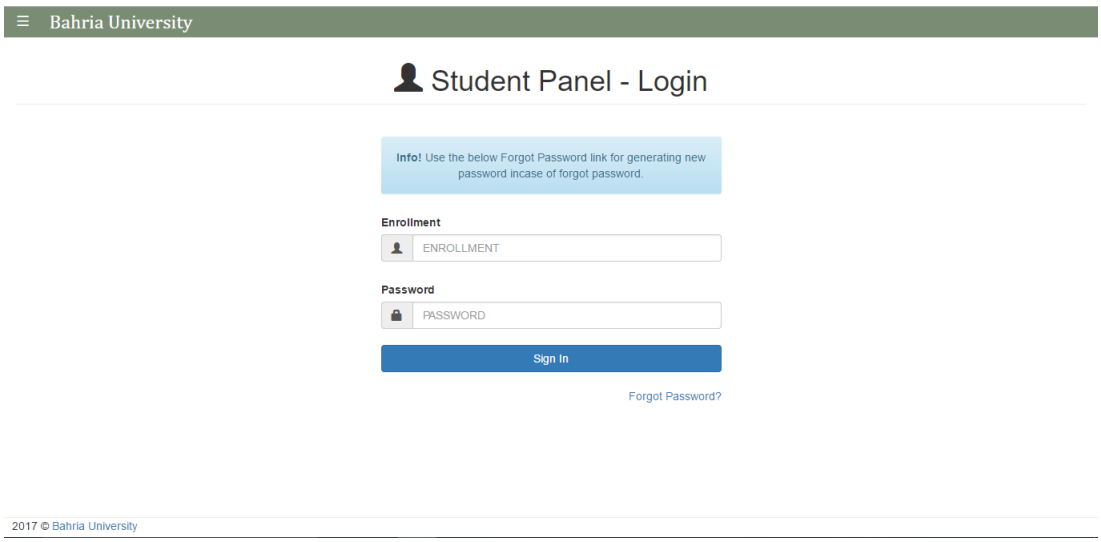


Figure 6.1: Student Login Panel

After successful login, student enters into profile page as shown in Figure 6.2.

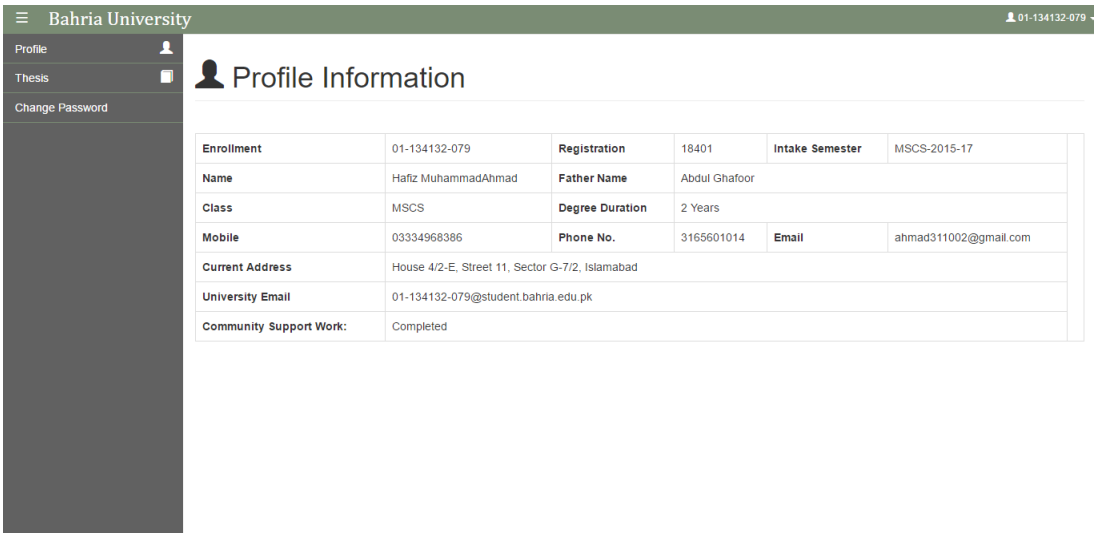


Figure 6.2: Student Profile Panel

Student Proposal panel is shown in Figure 6.3 where student submits the proposal details by entering valid information.

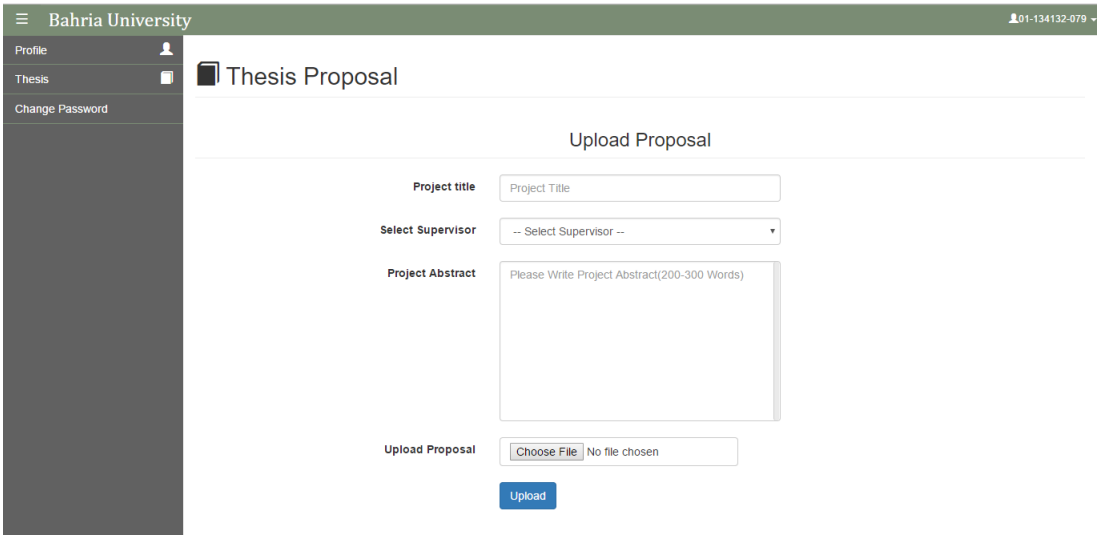


Figure 6.3: Student Submit Proposal

After submitting the proposal student is able to view the proposal details whether details are entered correctly or not. Details can also be updated later on by using update proposal panel as shown in Figure 6.4.

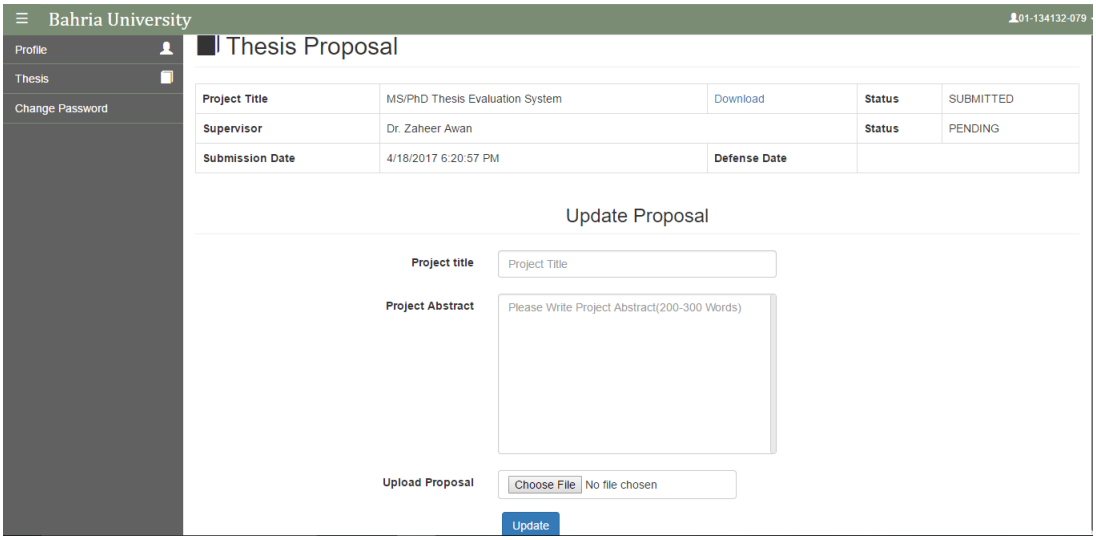


Figure 6.4: Student Proposal Details

For the testing of next module, there is an admin login panel as shown in Figure 6.5 where admin can only login using valid username and password.

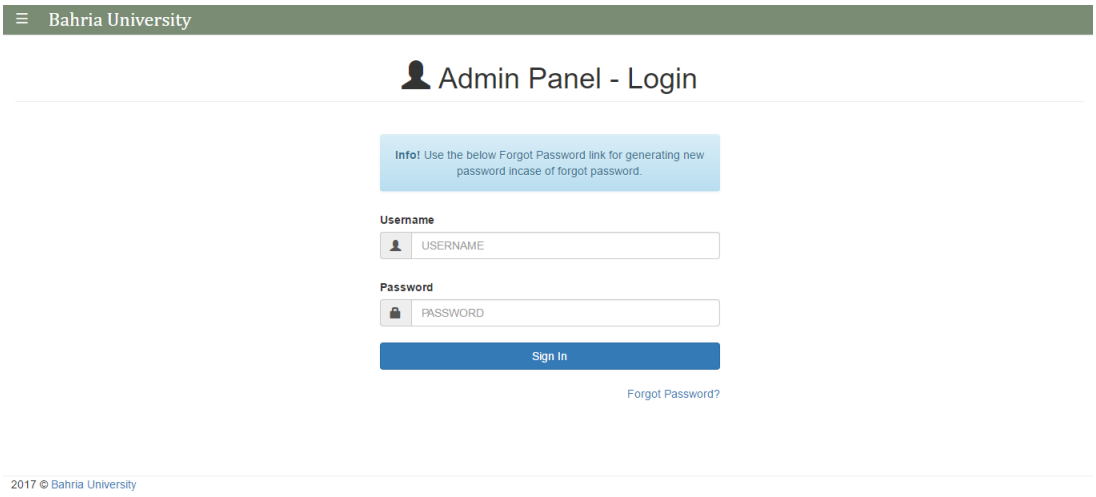


Figure 6.5: Admin Login Panel

After successful login admin view profile page as shown in Figure 6.6.

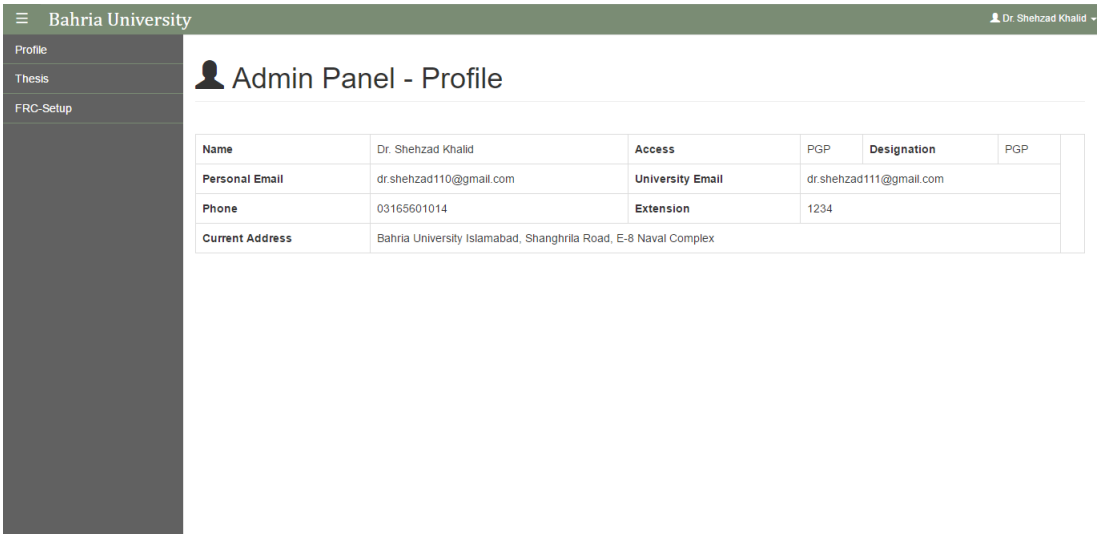


Figure 6.6: Admin Profile

For testing of the module given below there is an Admin setup proposal defense of student as shown in Figure 6.7 where admin setups defense for proposal submitted by the student and appoints examiners.

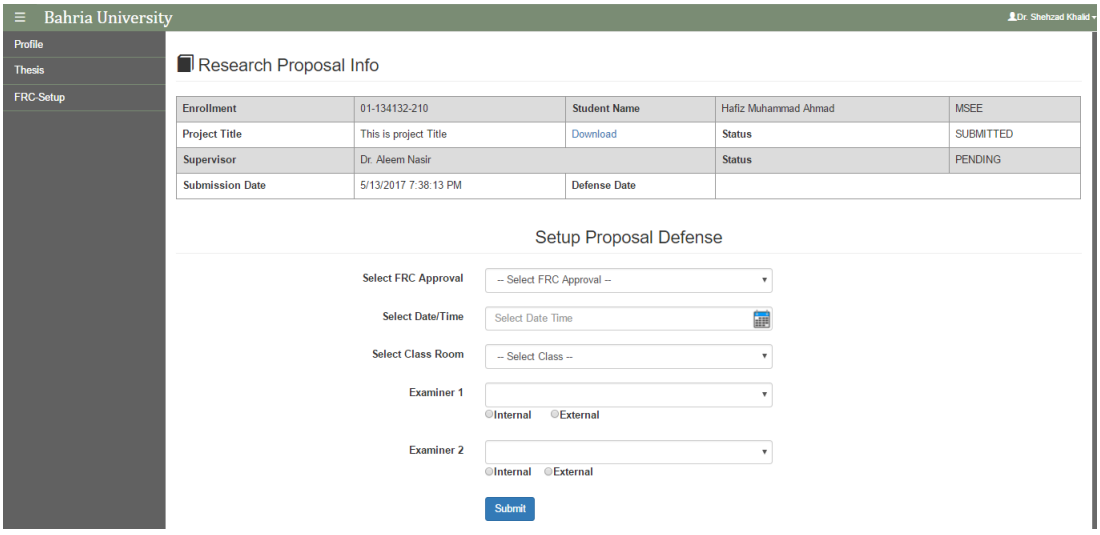


Figure 6.7: Admin Setup Proposal Defense

After setting up proposal defense, admin can view the defense details with the name of examiners appointed as shown in Figure 6.8.

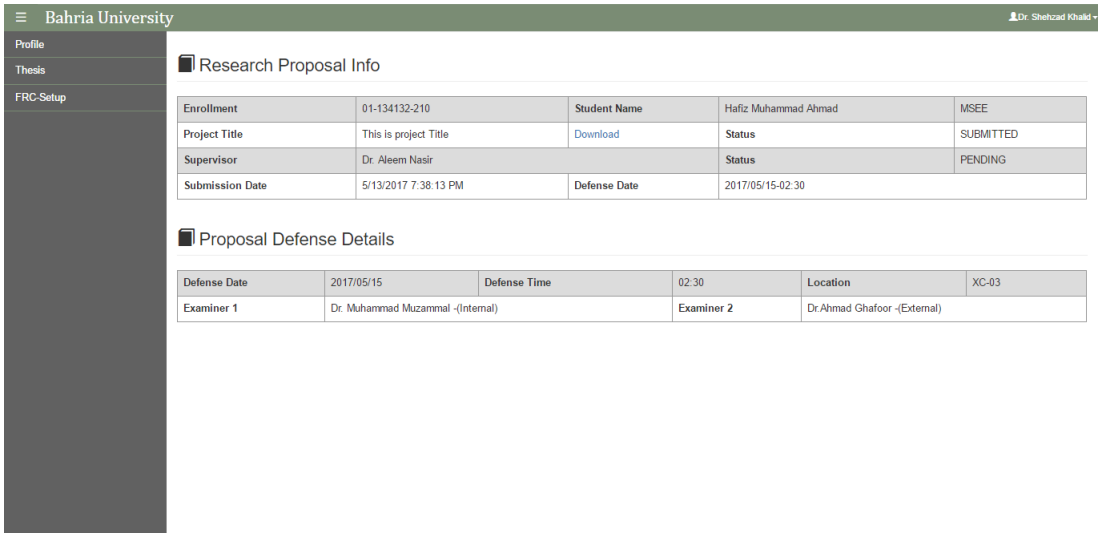


Figure 6.8: Admin View Proposal Defense Details

This test case shows that student also view the defense details on student panel as shown in Figure 6.9.

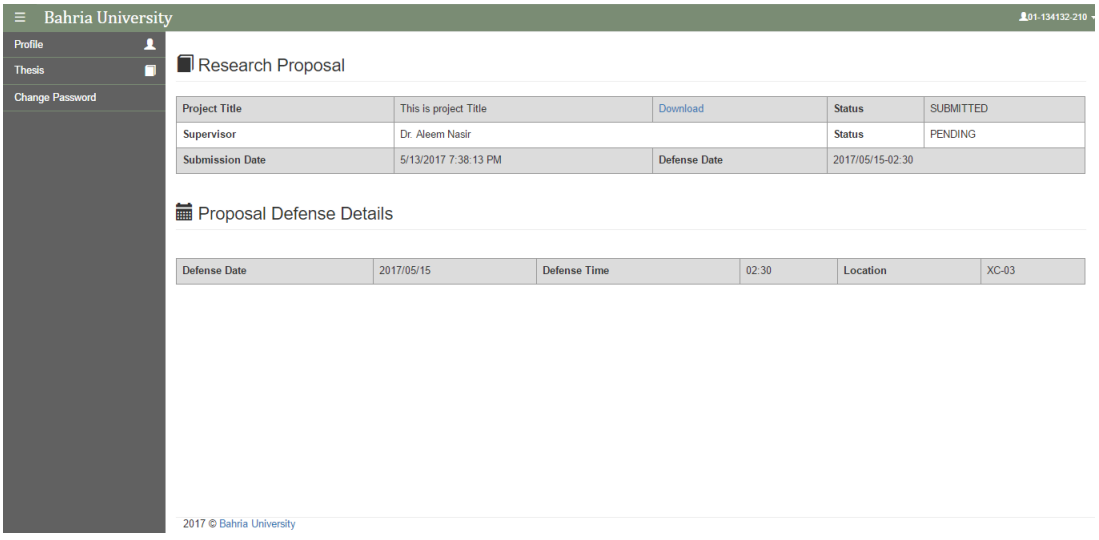


Figure 6.9: Student View Proposal Defense Details

After submission of all four bimonthly reports, student is able to upload thesis using thesis panel which is available on his/her student panel as shown in Figure 6.10

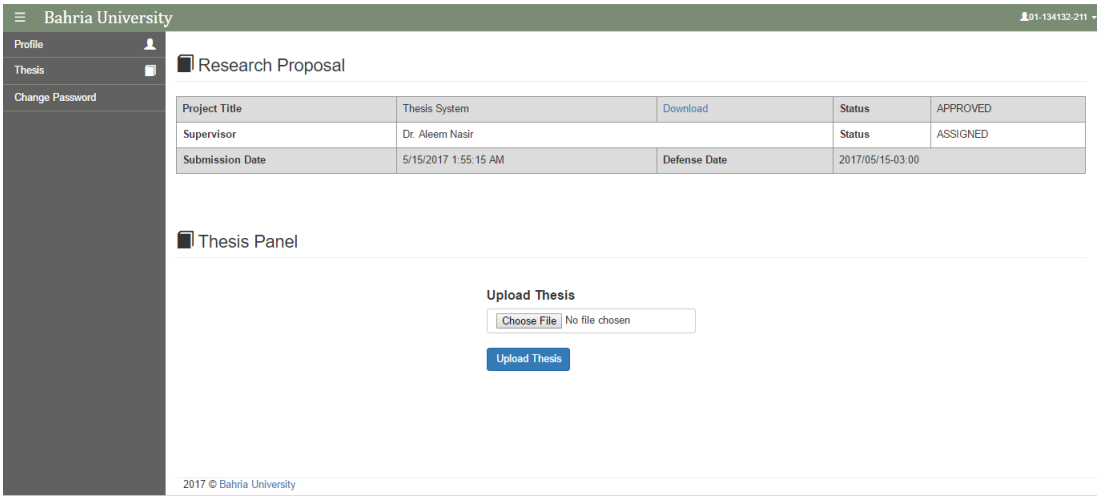


Figure 6.10: Student Upload Thesis

After uploading thesis, student can view thesis submission details as shown in figure 6.11.

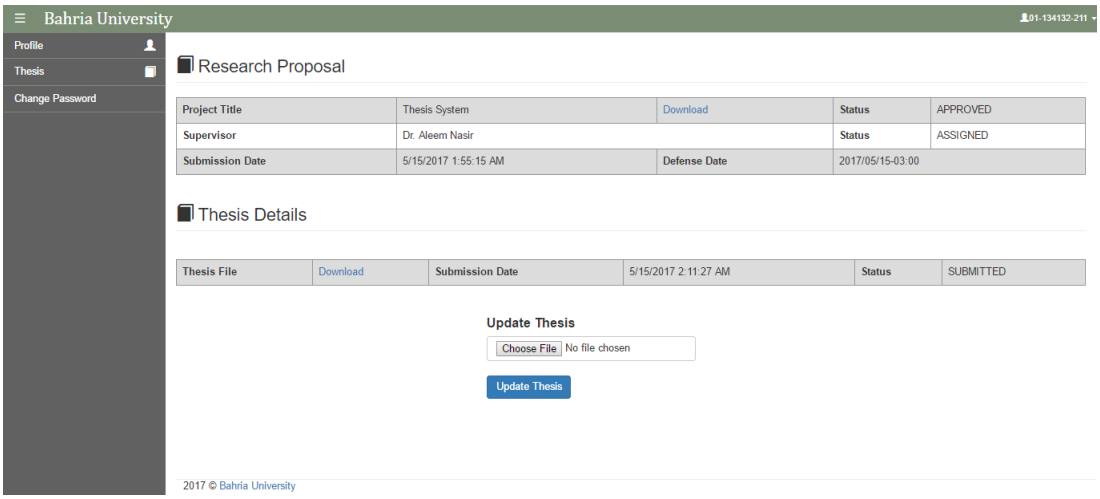


Figure 6.11: Student Panel Thesis Details

This test case show the final outcome of thesis after successful thesis defense as shown in Figure 6.11.

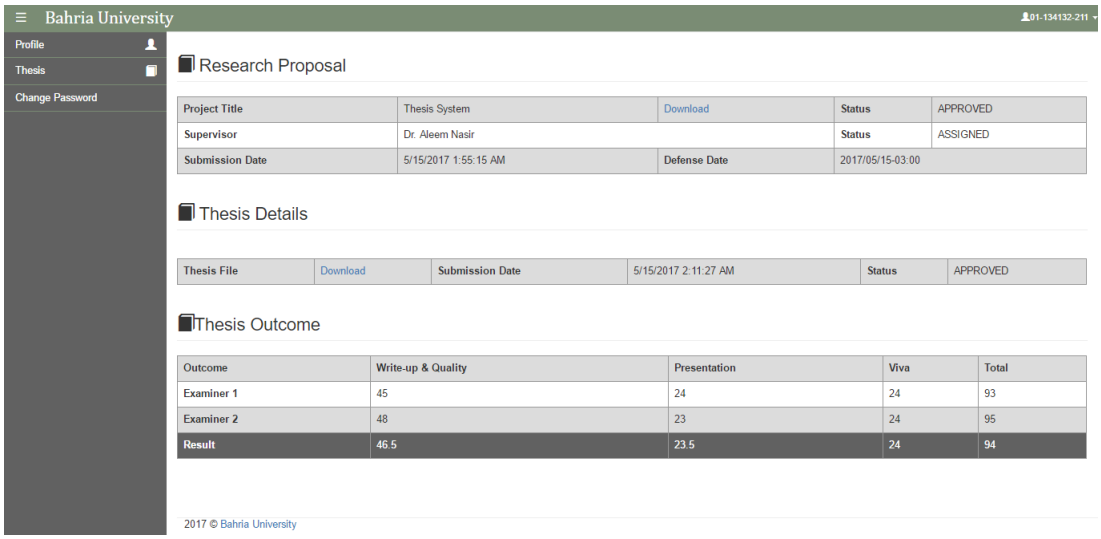


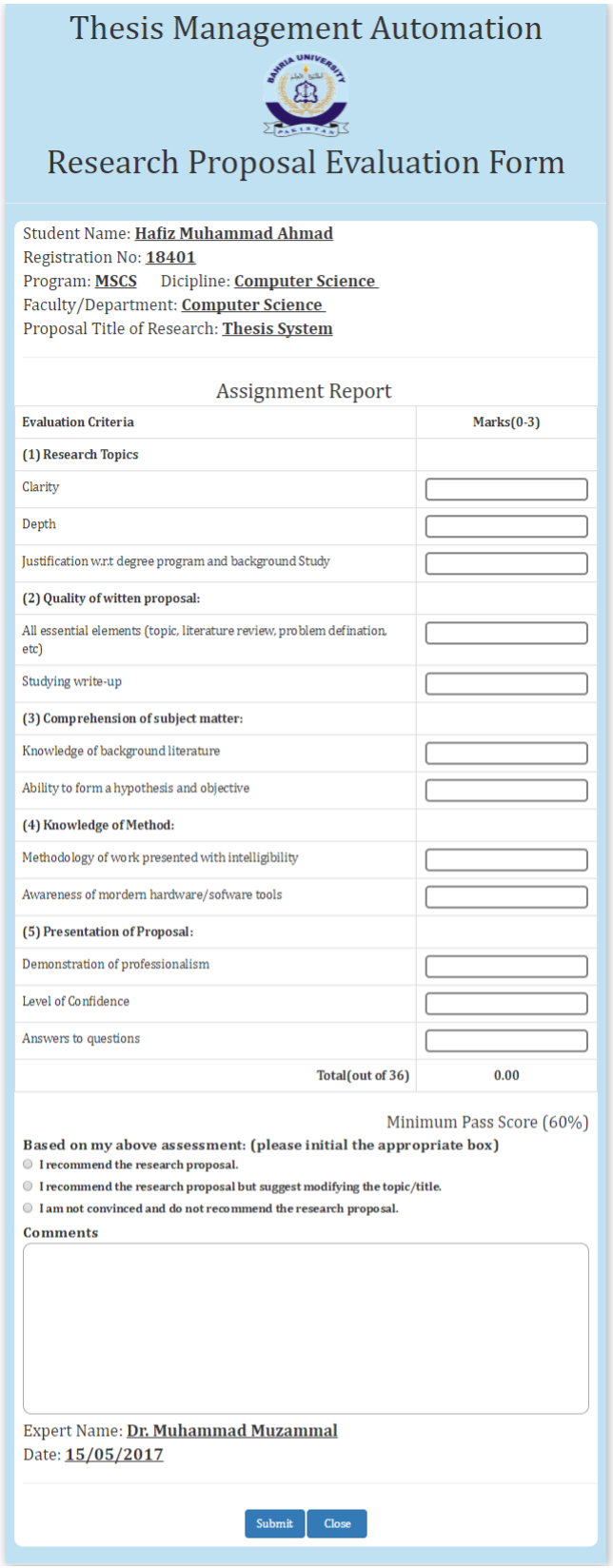
Figure 6.12: Thesis Outcome

Next three test cases view evaluation forms and progress report form shown in figure from 6.13 to 6.15.

Test case in Figure 6.13 show that examiners appointed by admin must submit this evaluation form after proposal defense.

Test case in Figure 6.14 demonstrate that after successful submission of examiners, assigned supervisor submits four bimonthly reports in the form of evaluation form.

Test case in Figure 6.15 shows that thesis defense evaluation form to be submitted by examiners appointed in defense.



Thesis Management Automation

Research Proposal Evaluation Form

Student Name: **Hafiz Muhammad Ahmad**
Registration No: **18401**
Program: **MSCS** Dicipline: **Computer Science**
Faculty/Department: **Computer Science**
Proposal Title of Research: **Thesis System**

Assignment Report

Evaluation Criteria	Marks(0-3)
(1) Research Topics	
Clarity	
Depth	
Justification w.r.t degree program and background Study	
(2) Quality of witten proposal:	
All essential elements (topic, literature review, problem defination, etc)	
Studying write-up	
(3) Comprehension of subject matter:	
Knowledge of background literature	
Ability to form a hypothesis and objective	
(4) Knowledge of Method:	
Methodology of work presented with intelligibility	
Awareness of mordern hardware/sofware tools	
(5) Presentation of Proposal:	
Demonstration of professionalism	
Level of Confidence	
Answers to questions	
Total(out of 36)	**0.00**

Minimum Pass Score (60%)

Based on my above assessment: (please initial the appropriate box)

- **I recommend the research proposal.**
- **I recommend the research proposal but suggest modifying the topic/title.**
- **I am not convinced and do not recommend the research proposal.**

Comments

Expert Name: **Dr. Muhammad Muzammal**
Date: **15/05/2017**

Submit Close

Figure 6.13: Proposal Defense Evaluation Form For Examiner

Figure 6.14: Supervisor Bimonthly Evaluation Report Form

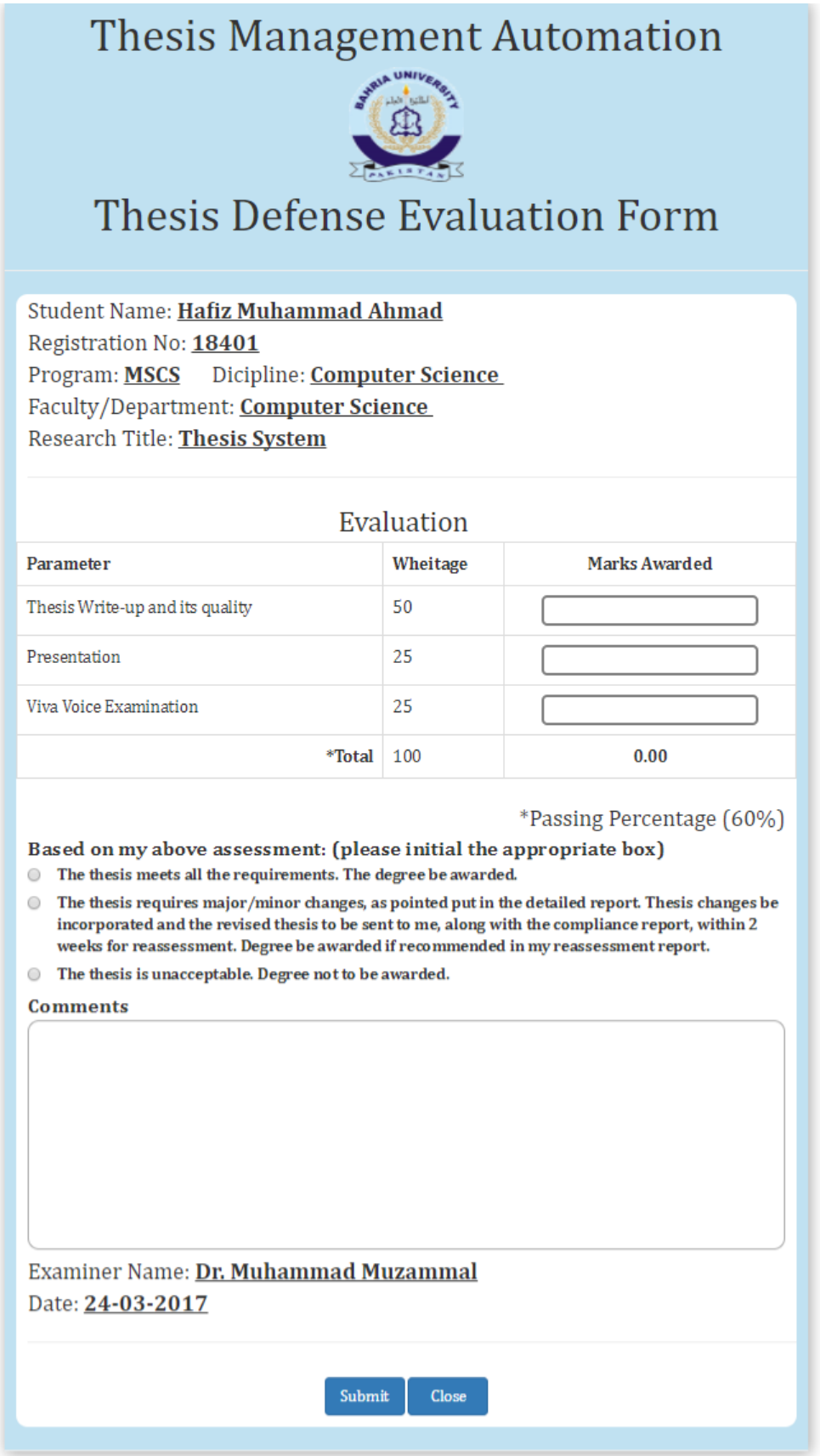


Figure 6.15: Thesis Defense Evaluation Form For Examiner

6.2 Usablility Testing

Usability testing is test by the audience. This testing gives the feedback to the developer that how much time user needs to perform the desired task in the application i.e. what is the response time of the system. It shows whether the user is satisfied with the performance of the system or not. This application has been used by various types of users, but it is most helpful for students who can learn where to submit the proposal and thesis and get details related to it. This is easily achievable because all the details area managed on single page.

6.2.1 User friendly Interface

Interface of this application is designed to be extremely user friendly. Users can easily understand the purpose of the buttons designed in the system. No need to learn or consult a user guide as it is very simple to use.

6.2.2 Easy to Use

Application is very easy to use. User can easily understand how to use the system to do desired task and get desired results from system.

6.2.3 Easy to Learn

Our application is designed in a very simple manner; fresh user can easily learn and understand how to use the system.

6.3 Compatibility Testing

Compatibility testing ensures the application's compatibility with the hardware and operating system. It tests the application for those hardware/software resources to check whether they are compatible for this application or not. It also checks operating system which is required to run this application. For compatibility testing we divide the resources according to category which is given below:

6.3.1 Device Features

- Processor Dual-core 1.2 GHz
- Ram 1GB

6.3.2 Operating System

This application is developed on Microsoft Visual Studio by using C sharp language which is Microsoft's Product. So this application runs only on Microsoft Windows operating system.

- Windows XP
- Windows 7
- Windows 8
- Windows 8.1
- Windows 10

6.3.3 Browser Compatibility

This application is compatible with following browsers.

- Google Chrome (Mobile and Desktop)
- Microsoft Edge
- Internet Explorer
- Mozilla Firefox (Mobile and Desktop)
- Opera (Mobile and Desktop)
- Safari (Mobile and Desktop)

6.4 Platform Version

This project can only run on the listed below versions:

- Microsoft Visual Studio 2010 and above
- .NET Framework 4.5.1
- .NET Framework 4.5.2
- .NET Framework 4.6.1
- .NET Framework 4.6.2

6.5 Security Testing

Thesis Management Automation system has SQL server database that needs to be connected by the authenticated username and password for successful connection. It's not possible for the unauthorized people to access because SQL server is running on another server. All the data manipulation is done on the back end. No single information is stored on client side so it provides security to data from any unauthorized access.

6.6 Installation Testing

This testing is done to check the installation of the application. To run this application, there must be the Microsoft Operating System installed on the system. This project is built on .NET Framework 4.5 so 4.5 or up to 4.5 must be installed on the system. This application will be available at windows store from where user can download and install it on their PCs. But right now .aspx and .aspx.cs files are needed to run on the system.

6.6.1 Testcase-1 (Login Verification Testing)

This test case verifies that users, only with registered credentials can login into the system.

Test Case id	TC-1
Unit Testing	Test to verify if the user has valid enrolment and password, what system will do
Steps to be executed	User must enter valid enrolment and password in order to login to the system
Expected Result	System should check the valid enrolment and password,and inform the user to check the enrolment or password, if incorrect
Actual Result	User logged by validating enrolment and password
Test Result	Success

Table 6.1: Verification Testing

6.6.2 Testcase-2 (Profile Information Testing)

This test case verifies that users views only the correct information on profile page.

Test Case id	TC-2
Unit Testing	Test to verify the user get the correct information on profile page after login
Steps to be executed	User must logged in
Expected Result	Correct information viewed on profile page
Actual Result	User gets correct information on student profile page
Test Result	Success

Table 6.2: Profile Information Testing

6.6.3 Testcase-3 (Upload Proposal Validation Testing)

This test case verifies that users views only the correct information on profile page.

Test Case id	TC-3
Unit Testing	Test to verify if input fields on the thesis page for proposal section are not empty.
Steps to be executed	User must fill all the input fields
Expected Result	No "input field required" error shown on the proposal section
Actual Result	No error shown
Test Result	Success

Table 6.3: Upload Proposal Validation Testing

6.6.4 Testcase-4 (Proposal Defense Detail Testing)

This test case verifies that student view only correct proposal details in proposal detail section.

Test Case id	TC-4
Unit Testing	Test to verify correct proposal information shown on proposal detail section after submitting proposal detail
Steps to be executed	User submits proposal details
Expected Result	System should check the proposal detail and execute on proposal detail section accurately
Actual Result	Accurate information shown on proposal detail section
Test Result	Success

Table 6.4: Proposal Defense Detail Testing

6.6.5 Testcase-5 (Update Proposal Testing)

This test case verifies that student view only correct proposal details in proposal detail section.

Test Case id	TC-5
Unit Testing	Test to verify update proposal panel shown after submit proposal details
Steps to be executed	User submitted proposal successfully
Expected Result	System should check the update panel shown after submitting proposal
Actual Result	Update panel shown on thesis page
Test Result	Success

Table 6.5: Update Proposal Testing

6.6.6 Testcase-6 (Admin Panel-Student info Testing)

This test case verifies that admin view accurate information of student selected.

Test Case id	TC-6
Unit Testing	Test to verify if the admin get the correct information of the student on setup proposal defense page
Steps to be executed	User select specific student to setup a proposal defense.
Expected Result	System should check the valid enrolment and password, and inform the user to check the enrolment or password, if incorrect
Actual Result	User logged by validating enrolment and password
Test Result	Success

Table 6.6: Admin Panel-Student info Testing

6.6.7 Testcase-7 (Setup Proposal Defense Validation Testing)

This test case verifies that admin gets error if any input field is required while defense setup.

Test Case id	TC-7
Unit Testing	Test to verify if input fields are empty on setup defense proposal panel
Steps to be executed	User must not enter in any input field
Expected Result	System should check if input fields are empty than throw error
Actual Result	System shown an error, input fields are required
Test Result	Success

Table 6.7: Setup Proposal Defense Validation Testing

6.6.8 Testcase-8 (Proposal Defense Details Testing)

This test case verifies that admin view accurate information of proposal defense after defense setup.

Test Case id	TC-8
Unit Testing	Test to verify accurate proposal defense details panel shown
Steps to be executed	Admin must setup a proposal defense
Expected Result	System should view accurate information of defense
Actual Result	Proposal defense details panel shown with correct information
Test Result	Success

Table 6.8: Proposal Defense Details Testing

6.6.9 Testcase-9 (Examiner-Email Verification Testing)

This test case verifies that examiners get email with accurate defense evaluation form and proposal file.

Test Case id	TC-9
Unit Testing	Test to verify examiners get emails with link of evaluation form after successful proposal defense setup
Steps to be executed	Admin setup a proposal defense
Expected Result	System should check the emails sent to both of examiners
Actual Result	Email sends to examiners
Test Result	Success

Table 6.9: Examiner-Email Verification Testing

6.6.10 Testcase-10 (Student-Email Verification Testing)

This test case verifies that student gets email with accurate proposal defense details.

Test Case id	TC-10
Unit Testing	Test to verify that student get email with proposal defense information and proposal defense detail panel is visible on student thesis page
Steps to be executed	Admin setup a proposal defense
Expected Result	System checks if email sent to student with accurate defense information
Actual Result	Email sent to student with correct information and proposal defense detail panel shown on student thesis page
Test Result	Success

Table 6.10: Student-Email Verification Testing

6.6.11 Testcase-11 (Proposal Status Testing)

This test case verifies that proposal status of student is updated after successful submission of proposal defense evaluation forms by examiners .

Test Case id	TC-11
Unit Testing	Test to verify if examiners submit evaluation form by recommending the proposal then student's proposal status updated to approve
Steps to be executed	Examiner submits the evaluation form by recommending proposal
Expected Result	Proposal status on student profile change to approve
Actual Result	Proposal defense changed to approved
Test Result	Success

Table 6.11: Proposal Status Testing

6.6.12 Testcase-12 (Upload Thesis Testing)

This test case verifies that student gets option to upload thesis after successful submission of all four bimonthly reports by supervisor assigned to student.

Test Case id	TC-12
Unit Testing	Test to verify if supervisor submits all progress report then upload thesis panel is visible on student thesis page
Steps to be executed	Supervisor submits all the progress report of students
Expected Result	Upload thesis panel on student thesis page should be visible
Actual Result	Upload thesis panel shown on student thesis page
Test Result	Success

Table 6.12: Upload Thesis Testing

6.6.13 Testcase-13 (Thesis Details Verification Testing)

This test case verifies that student view accurate thesis details after successful submission of thesis.

Test Case id	TC-13
Unit Testing	Test to verify if student upload thesis then thesis detail panel is shown on student thesis page
Steps to be executed	Student upload thesis with valid file format
Expected Result	Thesis detail panel is visible on student thesis page
Actual Result	Thesis detail panel is shown on student thesis page
Test Result	Success

Table 6.13: Thesis Details Verification Testing

Chapter 7

Conclusions

This project was the part of those attempts that were evolving over the years in Bahria University. We identified from the commencement that creating a whole outcome would be very difficult within the given time frame. We observed the project as a journey where we learned many lessons and increased our knowledge of the topic which we tried to share in this report. We tried to look at the problem from many points of view which produced some new concepts that could be discovered in the future.

During working on this project we learnt a lot of things. Web base application using Asp.Net was totally new as well as challenging environment and ever changing requirements of university administration. We accepted the challenge to make an automated thesis management process for Bahria university islamabad campus. We have learnt short term project planning and implementation skills during the implementation of this project which includes:

- Requirement gathering
- Research about the domain system design
- Implementation and basic functional and quality testing
- GUI Design
- Interaction with different tools and technologies
- Time management

References

[1] Philip J. Goldstein and Richard N. Katz. Academic analytics: The uses of management information and technology in higher education. *Vol. 8. Educause*, 2005. Cited on p. 1.

[2] Richard O. Mason and Ian I. Mitroff. A program for research on management information systems. *Management science 19.5*, pages 475–487, 1973. Cited on p. 2.

[3] Frank Levy and Richard J. Murnane. The skill content of recent technological change: An empirical exploration. *The Quarterly journal of economics*, pages 1279–1333, 2003. Cited on p. 3.

[4] Kenneth Kraemer Melville, Nigel and Vijay Gurbaxani. Information technology and organizational performance. *An integrative model of IT business value.*, pages 283–322, 2004. Cited on p. 3.

[5] Claudia Dale Goldin and Lawrence F. Katz. The race between education and technology. *Harvard University Press*, 2009. Cited on p. 4.