

QUAID-I-AZAM UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Dated: Oct. 18, 2018


FINAL APPROVAL

This is to certify that we have read the project report submitted by Mr. Muhammad Imran Lashari and it is our judgment that this report is of sufficient standard to warrant its acceptance by the Quaid-i-Azam University, Islamabad for the degree of the Bachelors of Science in Computer Science.

COMMITTEE

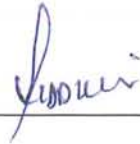
1. External Examiner

Dr. Rizwan Bin Faiz
Assistant Professor
Faculty of Computing
Ripha International University
Sector I-14,
Islamabad.



2. SUPERVISOR

Dr. Ghazanfar Farooq Siddique
Assistant Professor
Deptt. of Computer Sciences
Quaid-i-Azam University
Islamabad.



3. INCHARGE

Dr. Onaiza Maqbool
Assistant Professor
Deptt. of Computer Sciences
Quaid-i-Azam University
Islamabad.



Acknowledgement

First of all, I would like to express my sincere and humble gratitude to Almighty ALLAH whose blessings and guidance has been a real source of all achievements in my life. I am thankful to my parents, brothers and sister for their prayers and all kinds of support during my educational career. This project is the end of my journey in obtaining my BS degree. I have not traveled in a vacuum in this journey. At this moment of accomplishment, I pay homage to my supervisor Dr. Ghazanfar Farooq. This work would not have been possible without his guidance, support and encouragement. Under his guidance I successfully overcame many difficulties and learned a lot. I really appreciate their caring attitude and guidance.

Thanks Everyone...!

M IMRAN LASHARI

Abstract

Universities Website Evaluation System is a portal or web-based platform through which different types of users can post their reviews about websites of universities. This portal contains 3 users Admin, Registered User and Unregistered User. Admin can add new university information like URI, Name, Status. Admin can delete and modify university information. Registered User can post comment against any university name. Registered User can post only one comment at a time against one universities' website. Unregistered user cannot post comment, to post comment user must be registered and logged in. After posting comment ranking list will be updated. Admin, Registered User and Unregistered User can view the comments and ranking list

Table of Contents

1 SOFTWARE PROJECT MANAGEMENT PLAN... 10

1.1 Introduction.....11

1.2 Purpose.....11

1.3 Scope.....11

1.4 Project Deliverables11

1.5 Project Organization11

1.6 Software process model11

1.7 Roles and Responsibilities13

1.8 Tools and techniques.....13

1.9 PROJECT MANAGEMENT PLAN14

1.10 Tasks.....15

 1.10.1 Analysis Phase15

 1.10.2 Design Phase.....15

 1.10.3 Coding/Implementation Phase.....16

 1.10.4 Testing.....16

 1.10.5 Dependencies and Constraints16

2 SOFTWARE REQUIREMENT SPECIFICATION... 17

2.1 System Overview18

2.2 Project Overview18

 2.2.1 Laravel.....18

 2.2.2 Project Working18

 2.2.3 Calculating chance - the rules of Frequency.....19

 2.2.4 External Interface Requirements.....19

 2.2.5 User Interfaces20

 2.2.6 Hardware Interfaces20

 2.2.7 Software Interfaces20

 2.2.8 Communications Protocols20

2.2.9	Software Product Features	20
2.2.10	Software System Attributes	20
2.2.11	Reliability.....	20
2.2.12	Availability	21
2.2.13	Security	21
2.2.14	Maintainability.....	21
2.2.15	Portability.....	21
2.2.16	Performance	21
2.3	Product Functions:	22
2.3.1	Use Cases	22
2.3.2	Database Requirements.....	30
2.4	FIGURES	30
2.4.1	Use Case Diagram.....	30
3	SOFTWARE DESIGN DESCRIPTION	31
3.1	INTRODUCTION	32
3.2	Design Overview	32
3.3	Requirement Traceability Matrix.....	32
3.4	SYSTEM ARCHITECTURAL DESIGN.....	34
3.5	Chosen System Architecture.....	35
3.5.1	Domain Model	37
3.5.2	Component Diagram	38
3.5.3	Entity Relationship Diagram.....	39
3.6	DETAILED DESCRIPTION OF COMPONENTS	40
3.6.1	Login	40
3.6.2	Admin.....	40
3.6.3	Registered User	40
3.6.4	Unregistered User	40
3.6.5	Comment	40
3.6.6	Database	40
3.6.7	Ranking list	40

3.6.8	Registration	41
3.6.9	University Information.....	41
3.7	USER INTERFACE DESIGN.....	41
3.7.1	Description of User Interface.....	41
3.7.2	Sequence Diagram	42
3.7.3	Class Diagram	53
4	SOFTWARE IMPLEMENTATION DOCUMENT...55	
4.1	Introduction.....	56
4.2	Language Selection.....	56
4.3	Tools Selection.....	56
4.4	Resources	56
4.4.1	Data Tables.....	56
4.4.2	PHP Mailer Class	56
4.4.3	Bootstrap Library	56
4.5	Screen Images	56
4.5.1	Login Page	57
4.5.2	Registration	57
4.5.3	Home Page	58
4.5.4	Add Comments Page.....	58
4.5.5	Ranking list Page by Admin	59
4.5.6	Add University Information Page.....	59
4.5.7	Edit University information	60
4.5.8	Delete University information	60
4.5.9	Update Account.....	61
4.5.10	Delete Account.....	61

5	SOFTWARE TEST DOCUMENT.....	63
5.1	INTRODUCTION	64
5.2	Test Approach.....	64
5.3	TEST PLAN.....	65
5.4	Features to be tested.....	65
5.5	Features not to be tested.....	66
5.6	TEST CASES.....	66

List of Tables

Table 2. 1 Admin login / sign in	22
Table 2. 2 Add university Information	23
Table 2. 3 Reg User Login/Sign in	24
Table 2. 4 Add Comment(s).....	25
Table 2. 5 View Comment(s).....	25
Table 2. 6 View ranking list.....	25
Table 2. 7 Delete University Information.....	25
Table 2. 8 Edit University Information.....	25
Table 2. 9 Update Profile	25
Table 2. 10 Delete Account	25
Table 2. 11 Registration.....	25
Table 5. 1 Login.....	66
Table 5. 2 Add comment.....	67
Table 5. 3 Delete Comment	67
Table 5. 4 Add University Information	68
Table 5. 5 View comment.....	68
Table 5. 6 View Ranking List.....	69
Table 5. 7 Registration.....	69
Table 5. 8 Update Profile	70
Table 5. 9 Delete Account	70
Table 5. 10 Delete University Information.....	71
Table 5. 11 Edit University Information.....	71

List of Figures

Figure 1. 1 Water Fall Model.....	12
Figure 1. 2 Project Plan 1.....	14
Figure 1. 3 Project Plan 2.....	15
Figure 2. 1 Use case diagram.....	30
Figure 3. 1 Architectural Design.....	36
Figure 3. 2 Domain Model.....	37
Figure 3. 3 Component Diagram.....	38
Figure 3. 4 Entity Relationship Diagram.....	39
Figure 3. 5 Admin Login Sequence Diagram.....	43
Figure 3. 6 Registered User Login Sequence Diagram.....	44
Figure 3. 7 Add university info Sequence Diagram.....	45
Figure 3. 8 Edit university info Sequence Diagram.....	46
Figure 3. 9 Delete University Info sequence diagram.....	47
Figure 3. 10 Registered User add comment Sequence Diagram.....	48
Figure 3. 11 View comment Sequence Diagram.....	49
Figure 3. 12 View ranking list Sequence.....	50
Figure 3. 13 Registration Sequence Diagram.....	51
Figure 3. 14 Update Profile Sequence Diagram.....	52
Figure 3. 15 Delete Account Sequence Diagram.....	53
Figure 3. 16 Class Diagram.....	54
Figure 4. 1 Login screen image.....	57
Figure 4. 2 Registration page screen image.....	57
Figure 5. 1 Unit Test.....	65

CHAPTER NO 1

1 SOFTWARE PROJECT MANAGEMENT PLAN

1.1 Introduction

This project is about ranking of universities' website. In this project opinion mining technique will be used. Universities' websites will be evaluated and ranked according to user comment. Database of sentiments-based keywords will be used for evaluation. The factors through which a website can be evaluated listed below:

1.2 Purpose

The purpose of this project is to evaluate the websites of Pakistani Universities with the help of user's comment/reviews/opinions.

1.3 Scope

Only websites of Pakistani universities' will have evaluated. Any Registered user can give comment/review. Only Admin can add information of any Pakistani university. Any user can view the comments. Unregistered User can only view ranking list and comments. Furthermore, Language of comment/review will be in English and Rating will be done by a calculating frequency.

1.4 Project Deliverables

Web-based software or portal "University Website Evaluation System" which will evaluate universities' website using above mentioned factors and opinion mining.

1.5 Project Organization

In this project there is no team. As I am doing this project alone. The resources which I have used are internet, PHP Laravel documentation and stackoverflow.com

1.6 Software process model

It is an Abstract representation of a software process. It presents a description of a process from some particular perspective. The software process model followed is Agile.

What is Agile?

Agility has become today's buzzword when describing a modern software process. Everyone is agile. An agile team is a nimble team able to appropriately respond to changes. Change is what software development is very much about. Changes in the software being built, changes to the team members, changes because of new technology, changes of all kinds that may have an impact

on the product they build or the project that creates the product. Support for changes should be built-in everything we do in software, something we embrace because it is the heart and soul of software. An agile team recognizes that software is developed by individuals working in teams and that the skills of these people, their ability to collaborate is at the core for the success of the project.

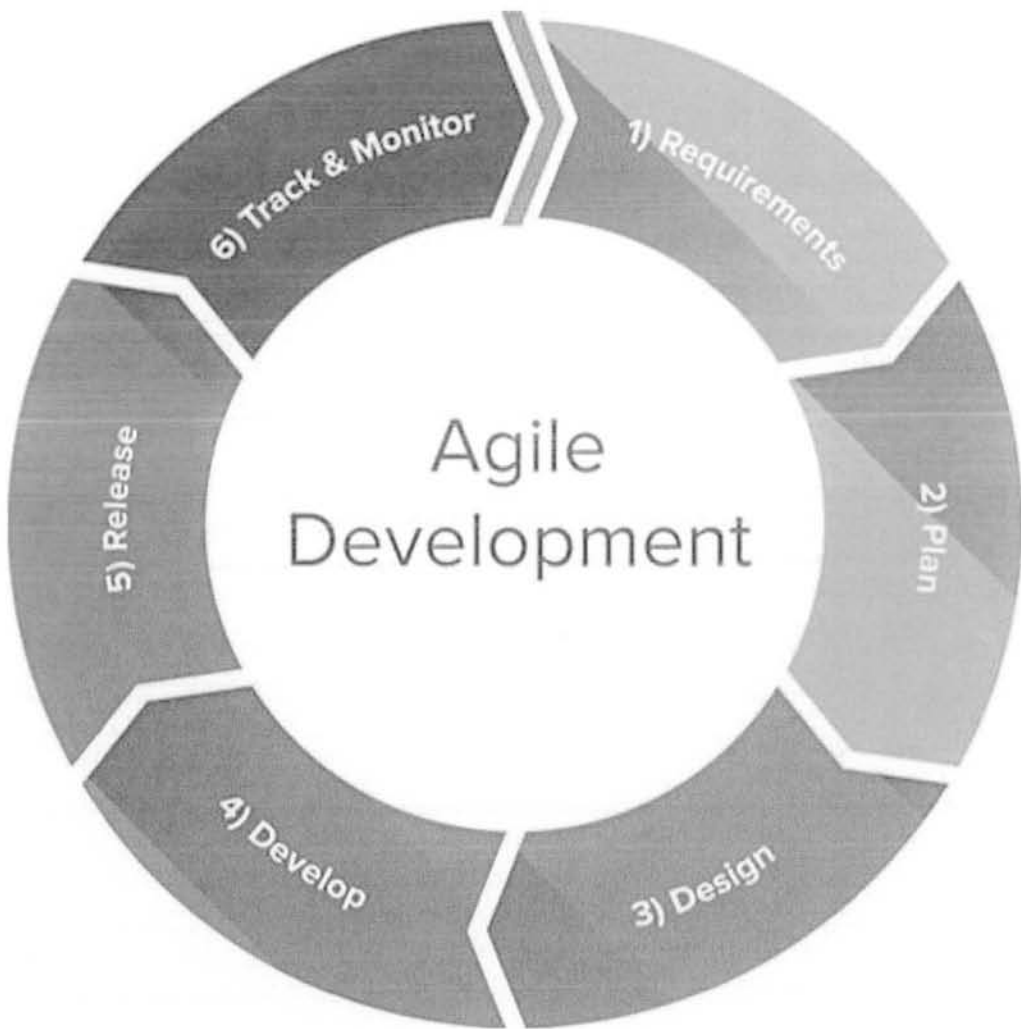


Figure 1. 1 Agile Process Model

1.7 Roles and Responsibilities

This project is solely allotted to me so all roles and responsibilities are on my side. The project contains different modules like Admin/User login, add university name and its website, give comment and review comment. Better communication with supervisor for a clear understanding of requirements. Meet the stakeholders for requirement gathering. After the complete understanding of system design a plan is made in order to meet the requirements. Make sure the availability of all stakeholders for refinement and for getting updates regarding system and requirements. The supervisor has very important role in refinements of requirement and testing of the system according to given requirements. Implementation of the project according to requirements and verify that system. Test the system and finally deployment of that system for the end users. The following are the role and responsibilities; project plan, requirements specification, analysis, architecture specifications, component or object specification, source code, test Plan and final deliverable.

1.8 Tools and techniques

Tools that we are using for implementation of this system are Laravel – PHP frame work, notepad and MySQL wamp server. Argo UML and Microsoft Visio for UML diagrams such as use case diagram, class diagram, activity diagram, domain model and Entity relationship diagram. In addition, for documentation Microsoft word is used. For designing a plan of the system, projectlibre is used. For CSS bootstrap is used as cascading style sheets to style the contents. Bootstrap, JavaScript, jQuery is also being used.

1.9 PROJECT MANAGEMENT PLAN

Software project planning is a task which is performed before the production of software actually starts. It is there for the software production but involves no concrete activity that has any direct connection with software production; rather it is a set of multiple processes, which facilitates software production.

	Views	Structure	Task
		Name	Duration Start Finish
1		Documentation	41 days? 9/25/17 8:00 AM 11/20/17 5:00 PM
2		SPMP	11 days? 9/25/17 8:00 AM 10/9/17 5:00 PM
3		Introduction	4 days? 9/25/17 8:00 AM 9/28/17 5:00 PM
4		Project Organization	1 day? 9/29/17 8:00 AM 9/29/17 5:00 PM
5		Meeting with supervisor	1 day? 10/2/17 9:00 AM 10/3/17 9:00 AM
6		Project Management Plan	3 days? 10/3/17 8:00 AM 10/5/17 5:00 PM
7		Additional Material	2 days? 10/5/17 8:00 AM 10/6/17 5:00 PM
8		Meeting with supervisor	1 day? 10/9/17 8:00 AM 10/9/17 5:00 PM
9		SRS	30 days? 10/10/17 8:00 AM 11/20/17 5:00 PM
10		Introduction	0 days? 10/10/17 8:00 AM 10/10/17 8:00 AM
11		Meeting with supervisor	1 day? 10/16/17 8:00 AM 10/16/17 5:00 PM
12		Specific Requirements	20 days? 10/17/17 8:00 AM 11/13/17 5:00 PM
13		External Interface Requirements	4 days? 10/17/17 8:00 AM 10/20/17 5:00 PM
14		Meeting with supervisor	1 day? 10/23/17 8:00 AM 10/23/17 5:00 PM
15		Software Product Features	4 days? 10/24/17 8:00 AM 10/27/17 5:00 PM
16		Meeting with supervisor	1 day? 10/30/17 8:00 AM 10/30/17 5:00 PM
17		Software System Attributes	4 days? 10/31/17 8:00 AM 11/3/17 5:00 PM
18		Meeting with supervisor	1 day? 11/6/17 8:00 AM 11/6/17 5:00 PM
19		Database Requirements	4 days? 11/7/17 8:00 AM 11/10/17 5:00 PM
20		Meeting with supervisor	1 day? 11/13/17 8:00 AM 11/13/17 5:00 PM
21		Additional Material	4 days? 11/14/17 8:00 AM 11/17/17 5:00 PM
22		Meeting with supervisor	1 day? 11/20/17 8:00 AM 11/20/17 5:00 PM

Figure 1. 2 Project Plan 1

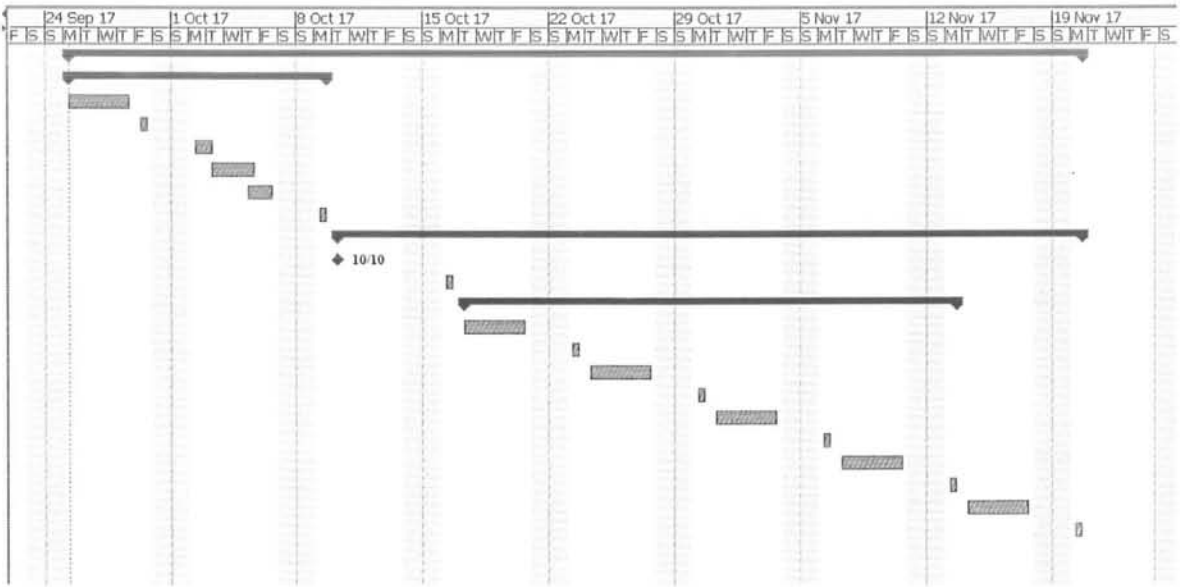


Figure 1. 3 Project Plan 2

1.10 Tasks

Different tasks will be performed to complete the final year project. I will divide the tasks in four main phases.

1.10.1 Analysis Phase

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications. Firstly, all the requirement will be gathered so that client must be satisfy.

Deliverables

At the end of analysis phase, a document will be formed which contains all the requirements.

1.10.2 Design Phase

It consists of a set of steps or phases in which each phase of the Software Development Life Cycle (SDLC) uses the results of the previous one. A Software Development Life Cycle adheres to

important phases that are essential for developers, such as planning, analysis, design, and implementation. These are explained below.

Deliverables

At the end of Design phase, a document will be generated which contains all the relevant information like domain model, use cases and class diagram.

1.10.3 Coding/Implementation Phase

The implementation and coding phase of the software development life cycle is the third phase of the SDLC process. In this phase I have to code and implement the requirements.

Deliverables

At the end of this phase Code will be ready to run and test.

1.10.4 Testing

Software Testing is an evaluation of the software against requirements gathered from users and system specifications. Testing is conducted in this phase. Software testing comprises of Verification and Validation.

Deliverables

After testing a fully functional code without errors/bugs will be presented.

1.10.5 Dependencies and Constraints

To perform above mentioned task, I will require Argo UML, projectlibre and MS word.

CHAPTER NO 2

2 SOFTWARE REQUIUREMENT SPECIFICATION

2.1 System Overview

University Website Evaluation System will evaluate the websites of universities using opinion mining. Registered users will give comments and according to positivity and negativity of comments websites will be rated accordingly.

2.2 Project Overview

Website evaluation system rates the website of different universities in Pakistan using comments of users. User will post a comment or review regarding a website added by admin, then points will be given to that website, thus ranking of universities will be updated. I have used Laravel framework to complete my project. I have used wamp server for database. Note pad++ and PhpStorm is used as an editor.

2.2.1 Laravel

Laravel is a free, open-source PHP web framework, created by Taylor Otwell and intended for the development of web applications following the model–view–controller (MVC) architectural pattern and based on Symfony. The source code of Laravel is hosted on GitHub and licensed under the terms of MIT License. Laravel is a web application framework with expressive, elegant syntax. Laravel attempts to take the pain out of development by easing common tasks used in the majority of web projects, such as authentication, routing, sessions, and caching. Laravel aims to make the development process a pleasing one for the developer without sacrificing application functionality. Laravel is accessible, yet powerful, providing powerful tools needed for large, robust applications.

2.2.2 Project Working

User will select name of any website and then a new window will be open. In new window there will be a space for comment. User has to give review about a website of selected university.

For Example: “This is a **professional** website and **good** developed.”

There are **total 8 words** in above comment and contains **2 positive words** which are shown in bold. There is a function in php called **preg_split ()** which splits each word of comment with a space or comma or full stop and stores in an array. There are two files present in `evaluation_system\public\files` folder named as **pos.txt** and **neg.txt**. The `pos.txt` file contains 2007 positive words while `neg.txt` file contains 4780 negative words. Each word of a comment is compared with words present in both files. There is a counter which will count the positive and negative words. Then formula of chance/likelihood is used.

2.2.3 Calculating chance - the rules of Frequency

The formula $P = x/N$ is used where P is Frequency of any event occurring, N is total number of words in comment and x is number of either positive or negative words occurred in a comment.

Ranking:

Comments are categorized in five categories

- 1 – More Negative comment with negative words and no positive word
- 2 – Less Negative comment with negative words more than positive words
- 3 – Neutral comment with equal positive and negative words
- 4 – Less positive comment with positive words more than negative words
- 5 – More positive comment with positive words and no negative word

Formula for 5-Star rating:

$(5 * \text{Number of comments fall in 5 category}) + (4 * \text{Number of comments fall in 4 category}) +$
 $(3 * \text{Number of comments fall in 3 category}) + (2 * \text{Number of comments fall in 2 category}) +$
 $(5 * \text{Number of comments fall in 5 category}) = \text{Result}$

Then,

$\text{Answer} = \text{Result} / \text{Total number of comments posted}$

This final result (Answer) will be stored as final points.

2.2.4 External Interface Requirements

This section provides a detailed description of all inputs and outputs. It also gives a description of the hardware, software and communication interfaces. Furthermore, it provides basic prototypes of the user interface. On the first page there will be login and registration for user and user can view the current ranking and comments as well.

2.2.5 User Interfaces

Through user interface user will be able to interact with the system. This will be a web-based application for the users. User can use this application through internet. User would have a unique email and password to access this application in addition; user can post comments and admin can add universities websites.

2.2.6 Hardware Interfaces

Hardware interface is basically a physical connection between two technologies. User PC or laptop must be connected with ethernet or wireless device.

2.2.7 Software Interfaces

Universities website evaluation system is a web-based application and it is implemented in PHP; therefore, this system will be run on any operation system. The internet is required to access the system. The system can access through any internet browser.

2.2.8 Communications Protocols

Communication protocols required for this system is; Hypertext transfer protocol (HTTP) for communication over the internet

2.2.9 Software Product Features

Product features are characteristics of our product that describe its appearance, components, and capabilities. A product feature is a slice of business functionality that has a corresponding benefit or set of benefits for that product's end user. My system will have a user-friendly interface. It will contain login, registration, add website and, give comments.

2.2.10 Software System Attributes

There are different System Attributes like debuggability, extensibility, portability, scalability, secure ability, testability and understandability. For databases reliability, availability, scalability and recoverability (RASR), is an important concept.

2.2.11 Reliability

System will be reliable if there will be no occurrence of the failure. My system will be able to work properly means the extent to which it works as and when needed. It will give the proper response to every query performed by the user. The system will be reliable. This system will be user friendly and easy to use.

2.2.12 Availability

My System will be available to every user at any time. It depends upon internet connection speed. All the registered users will be able to give comments and system will rank the universities website accordingly.

2.2.13 Security

Since this system will be hosted on server, all the user data will be kept on that server. Product should be able to protect privacy of user data. User should only be accessed the system through user own credentials and any other user have no access to the user private data. All input needs to be encoded and validated to prevent SQL injection. User can only perform operation under the permission. Some user groups can be configured that they can never have certain permission. Only those users who have Gmail, will be able to registered. System will send a code through an email to verify them because Gmail is considered the most secure email.

2.2.14 Maintainability

There should be an aspect of maintainability for the system. In some cases, maintainability involves a system of continuous improvement learning from the past. In order to improve the ability to maintain systems or improve reliability of the systems is based on maintenance experience. The application should be easy to extend. The code will be written in a way that it favors implementation of new functions.

2.2.15 Portability

This is a web-based system therefore main purpose of developing web-based system is to improve the portability of the system. To improve portability, system should run on variety of platforms and variety of connection speeds. The system should be a lightweight so that it can run on a machine with slow internet connection. To make the web application lightweight, simple libraries and tools will be used at developing phase.

2.2.16 Performance

Since this system is going to be a web-based, it does require a powerful server machine with high band internet access. Server machine should have a powerful CPU and high-speed internet access so that it can handle multiple users at the same time. Another performance requirement is the storage space. Higher storage space means more user and bigger workspace per user. Performance requirements from the user side are: web application should be developed as a lightweight web app so that it can work on almost any platform even with slower internet connections. Expected

number of simultaneous users should be at least 200. System should be able to deal with at least 200 users at the same time. Also, database of the system should handle at least a five hundred of users at any periods and response to any query with in limited time.

2.3 Product Functions:

2.3.1 Use Cases

In software and systems engineering, a use case is a list of actions or event steps typically defining the interactions between a role and a system to achieve a goal. The actor can be a human or other external system.

Table 2. 1 Admin login / sign in

ID	UC1
Name	Admin Sign in
Primary Actor	Admin
Pre-Conditions	i. Admin should on login screen
Post-Conditions	i. Information of the Admin will be match for login from database ii. admin will use admin panel
Main Success Scenario	1. Admin enters the data in given fields (password, email) for sign in 2. Admin clicks on login button. 3. Information related to Admin will be compare from database
Alternative flows or Extensions	1a. If admin clicks on Sign in button without entering email and password 1. System gives error message asks user to enter valid user name and password.
Frequency	Could be nearly continuous

Table 2. 2 Add university Information

ID	UC2
Name	Admin Add University Information
Primary Actor	Admin
Pre-Conditions	I. Admin should on login screen/admin panel
Post-Conditions	I. University Information will be stored in the system
Main Success Scenario	<ol style="list-style-type: none"> 1. Admin enters the Information in given fields (University Name, URI, select Status) 2. Admin clicks on Add button. 3. Information will be added to system
Alternative flows or Extensions	<ol style="list-style-type: none"> 1a. If admin enters the Information that is already present in the system <ol style="list-style-type: none"> 1. System gives message of already present. 2a. If admin clicks on Add button without entering user data <ol style="list-style-type: none"> 1. System gives error message asks user to enter data
Frequency	Could be nearly continuous

Table 2. 3 Reg User Login/Sign in

ID	UC3
Name	Registered User Sign in
Primary Actor	Registered User
Pre-Conditions	i. Registered User should be registered first i. Registered User should on login screen
Post-Conditions	i. Information of the Registered User will be compared from the system ii. Registered User will be logged in and can post comments
Main Success Scenario	1. Registered User enters the data in given fields (password, email) for sign in 2. Registered User clicks on sign in button.
Alternative flows or Extensions	1a. If Registered User clicks on Sign in button without entering email and password 1. System gives error message asks user to enter valid email and password.
Frequency	Could be nearly continuous

Table 2. 4 Add Comment(s)

ID	UC4
Name	Registered User/ Add comment
Primary Actor	Registered User/
Pre-Conditions	i. Registered User/ should on post Comment screen
Post-Conditions	i. Comment will be added successfully in database
Main Success Scenario	<ol style="list-style-type: none"> 1. Registered User/ enter the comment in given fields 2. Registered User/admin clicks on post comment button. 3. Comment will be added to system
Alternative flows or Extensions	<ol style="list-style-type: none"> 1a. If Registered User do not enter the minimum 3 characters <ol style="list-style-type: none"> 1. System gives message to give the comment having minimum 3 characters 2a. If Registered User clicks on post comment button without entering comment <ol style="list-style-type: none"> 1. System gives error message asks user to enter comment.
Frequency	Could be nearly continuous

Table 2. 5 View Comments

ID	UC5
Name	admin View comment
Primary Actor	Admin
Pre-Conditions	i. admin should on Comment screen
Post-Conditions	i. Information will be viewed by any admin
Main Success Scenario	1. User can scroll up and down and view the comment 2. User clicks on University name and views its comments.
Alternative flows or Extensions	
Frequency	Could be nearly continuous

Table 2. 6 View Ranking List

ID	UC6
Name	admin View ranking list of universities
Primary Actor	User / Registered User / admin
Pre-Conditions	i. Admin should on Home screen
Post-Conditions	i. Information will be viewed by admin.
Main Success Scenario	1. admin can scroll up and down and view the ranking
Alternative flows or Extensions	
Frequency	Could be nearly continuous

Table 2. 7 Delete University Information

ID	UC7
Name	Delete University Information
Primary Actor	Admin
Pre-Conditions	Admin should be logged in
Post-Conditions	University information will be deleted from database
Main Success Scenario	1. Admin goes to manage universities panel 2. admin click on small delete button 3. university information will be deleted from database
Alternative flows or Extensions	
Frequency	Could be nearly continuous

Table 2. 8 Edit University Information

ID	UC8
Name	Edit University Information
Primary Actor	Admin
Pre-Conditions	Admin should be logged in
Post-Conditions	University information will be Edit from database
Main Success Scenario	1. Admin goes to manage universities panel 2. admin click on small edit button 3. new window will be open
Alternative flows or Extensions	
Frequency	Could be nearly continuous



Table 2.9 Update Profile

ID	UC9
Name	Update profile
Primary Actor	Registered User/Admin
Pre-Conditions	User must be logged in
Post-Conditions	Profile updated
Main Success Scenario	<ol style="list-style-type: none"> 1. user click on its name on main bar 2. edit the name 3. profile successfully edited
Alternative flows or Extensions	
Frequency	Could be nearly continuous

Table 2.10 Delete Account

ID	UC10
Name	Delete Account
Primary Actor	Registered user
Pre-Conditions	Registered user must be logged in
Post-Conditions	Account will be deleted successfully
Main Success Scenario	<ol style="list-style-type: none"> 1. user click on its name on main bar 2. scroll down 3. click on Delete account button
Alternative flows or Extensions	
Frequency	Could be nearly continuous

Table 2.11 User Registration

ID	UC11
Name	User Registration
Primary Actor	Unregistered User
Pre-Conditions	i. User should on main ranking list screen
Post-Conditions	i. Information of the unregistered User will be stored in the database ii. unregistered User can log in to the system through proper email and password
Main Success Scenario	1. unregistered User enters the data in given fields (name, password, confirm password, email) for registration 2. unregistered User clicks on sign up button. 3. Information related to unregistered User will be added to system
Alternative flows or Extensions	1a. If Registered User enters the data that is already registered 1. System gives message of already registered no need to again registration. 2a. If unregistered User clicks on Sign up button without entering user name, password and email 1. System gives error message asks user to enter valid user name, email and password.
Frequency	Could be nearly continuous

2.3.2 Database Requirements

An SQL Server account required to connect to the Database Engine instance of SQL Server to host the Data Services database. The user account must belong to the MySQL server role on the instance of SQL Server Database Engine. All data is stored in a Data Services database. The computer that hosts this database must run an instance of SQL Server Database Engine.

2.4 FIGURES

2.4.1 Use Case Diagram

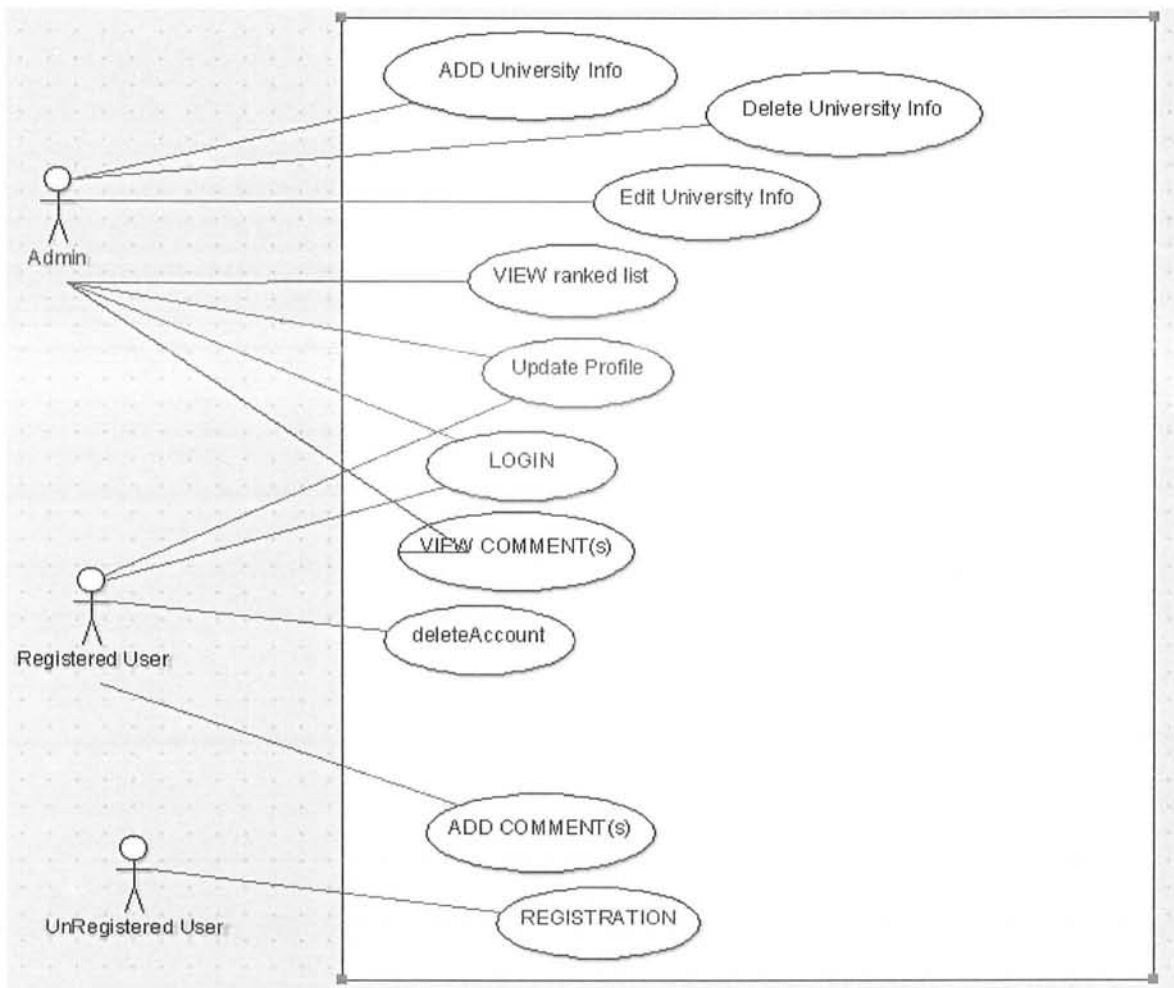


Figure 2. 1 Use case diagram

CHAPTER NO 3

3 SOFTWARE DESIGN DESCRIPTION

This chapter first gives the complete description of software design. It then elaborates the architectural design and detailed description of components of system. Finally, this chapter elucidates the user interface design and interaction diagrams mainly system sequence diagram and class diagram.

3.1 INTRODUCTION

Software Design Description (SDD) is the representation of a software design to be used for communication design information to its stakeholders. It shows how the software system will be structured to satisfy the requirements. The SDD is performed in two stages. The first is a preliminary design in which the overall system architecture and data architecture is defined. In the second stage, that is the detailed design stage, more detailed data structures are defined and algorithms and codes are developed for the defined architecture.

3.2 Design Overview

Design is meaningful engineering representation of something that is to be built. The design mainly focuses on four major areas of concern mainly data, architecture, interfaces and components. Software design is an iterative process through which requirements are translated into a blueprint for constructing the software. It shows how end user can interact with the system therefore mainly focus on user interface design. Design begins with requirement model and at each stage; software design work product is reviewed for clarity, correctness, completeness and consistency with the requirements and with one another. Software design sits at the technical kernel of software engineering and is applied regardless of the software process model that is used. The requirements translated clearly through designing class diagram, sequence diagram, system sequence diagram and user interface interactions.

3.3 Requirement Traceability Matrix

Requirements traceability matrix is a matrix in which we describe that which requirement is mapping with which sequence diagram, test case, and method of class diagram. The purpose of traceability matrix is that when requirements have to be updated then one can update that requirement using traceability matrix instead of going through the whole document. It is often used with high-level requirements and detailed requirements of the product to the matching parts of high-level design, detailed design, test plan, and test cases.

Table 3. 1 Requirement Traceability Matrix

Requirement Id	Requirement Name	Sequence Diagram	Test Case	Class Diagram	Interface	Domain Model
UC1	Admin Login	Sequence Diagram admin login	T1	Admin Login	Admin Login Interface	Login
UC2	Add University Information	Sequence Diagram add University Information	T4	University Information	Manage University Information Interface	University website
UC3	Registered User Login	Sequence diagram Registered User Login	T1	Registered User Login	Login Interface	Login
UC4	Registered User/admin Add comment	Sequence diagram Add comment	T2	Add comment	Comments interface	Comment
UC5	Registered User/admin Delete comment	Sequence diagram Delete comment	T3	Delete comment	Comments interface	Comment
UC6	View Comments	Sequence diagram View Comments	T5	View Comments	Comments interface	Comment
UC7	View Ranking list	Sequence diagram View Ranked list	T6	View Ranked list	Universities List interface	Ranking list

UC8	Delete University Information	Sequence Diagram Delete University Information	T10	University Information	University Information Interface	University website
UC9	Edit University Information	Sequence Diagram Edit University Information	T11	University Information	University Information Interface	University website
UC10	Update Profile	Sequence Diagram Update profile	T8	Profile Information	Profile Interface	Registered user information
UC11	Delete Account	Sequence Diagram Delete Account	T9	Registered Information	Profile Interface	Registered user information
UC12	Registration	Sequence diagram Registration	T7	Registration	Registration interface	Registration

3.4 SYSTEM ARCHITECTURAL DESIGN

Architectural design defines the relationship between major structural elements of the software. It defines the design patterns that can be used to achieve the requirements that have been defined for the system. Architecture design entails the manner in which these components interact and the structure of data that are used by the components. Components or modules are generalized to represent major system elements and their interactions.

3.5 Chosen System Architecture

The chosen architecture for this system is Three Tier Architectural Pattern. Three-tier architecture is a software architecture pattern in which the user interface (presentation), functional process logic (business rules), computer data storage and data access are developed and maintained as independent modules. Interacting between components of system is shown in diagram. The singular quality of three-tier architecture is the separation of the application logic into a distinct logical middle tier of software. The interface tier is relatively free of application processing; windows or web pages forward task requests to the middle tier. The middle tier communicates with the back-end storage layer. Three-tier architecture separates its tiers from each other based on the complexity of the users and how they use the data present in the database. In database tier, the database resides along with its query processing languages. This tier also has the relations that define the data and their constraints at this level. The middle tier resides the application server and the programs that access the database. For a user, this application tier presents an abstracted view of the database. End-users are unaware of any existence of the database beyond the application. At the other end, the database tier is not aware of any other user beyond the application tier. Hence, the application layer sits in the middle and acts as a mediator between the end-user and the database. In user interface tier end-users operate on this tier and they know nothing about any existence of the database beyond this layer. At this layer, multiple views of the database can be provided by the application. All views are generated by applications that reside in the application tier.

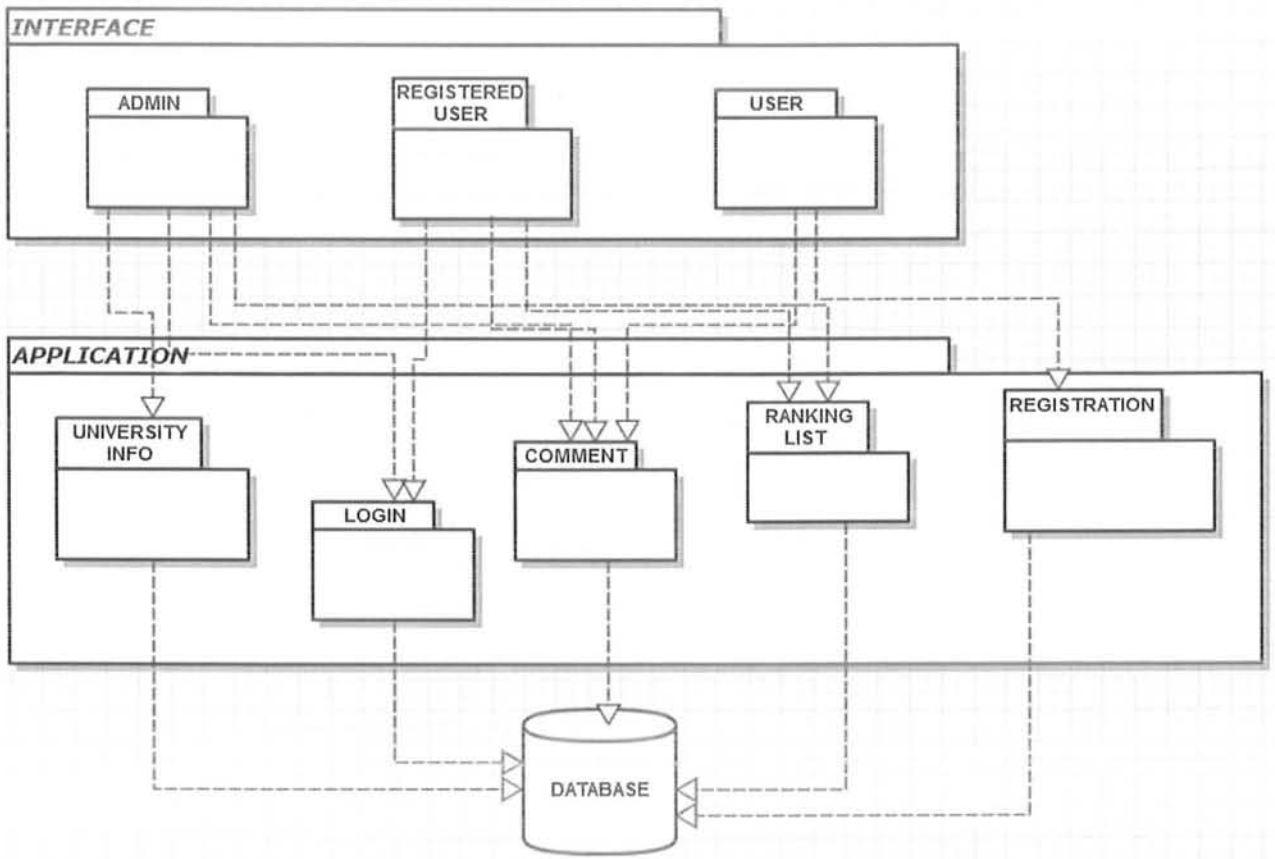


Figure 3.1 Architectural Design

3.5.1 Domain Model

A domain model is a system of abstractions that describes selected aspects of a sphere of knowledge, influence, or activity. The model can then be used to solve problems related to that domain. The domain model is a representation of meaningful real-world concepts pertinent to the domain that need to be modelled in software. The concepts include the data involved in the business and rules the business uses in relation to that data.

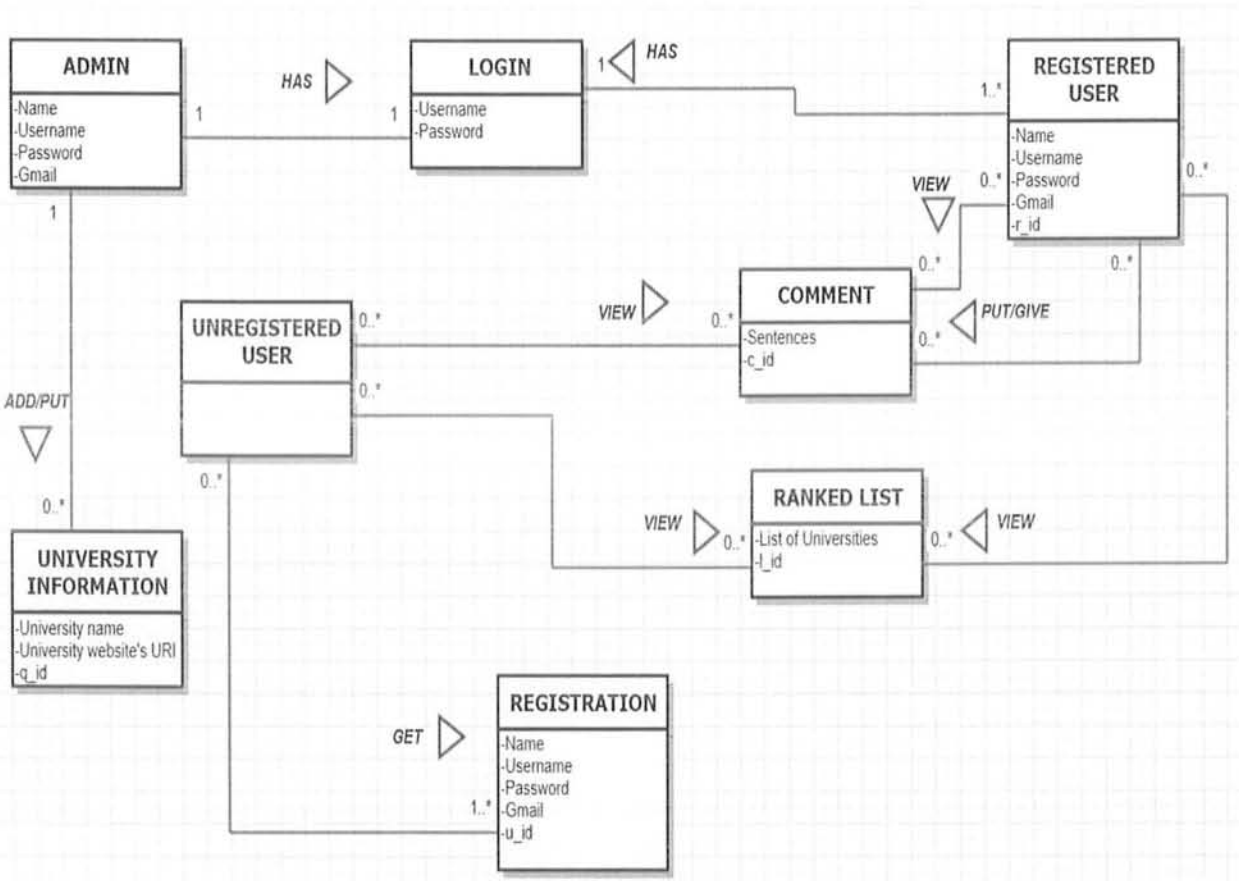


Figure 3. 2 Domain Model

3.5.2 Component Diagram

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities. Thus, from that point of view, component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files, etc.

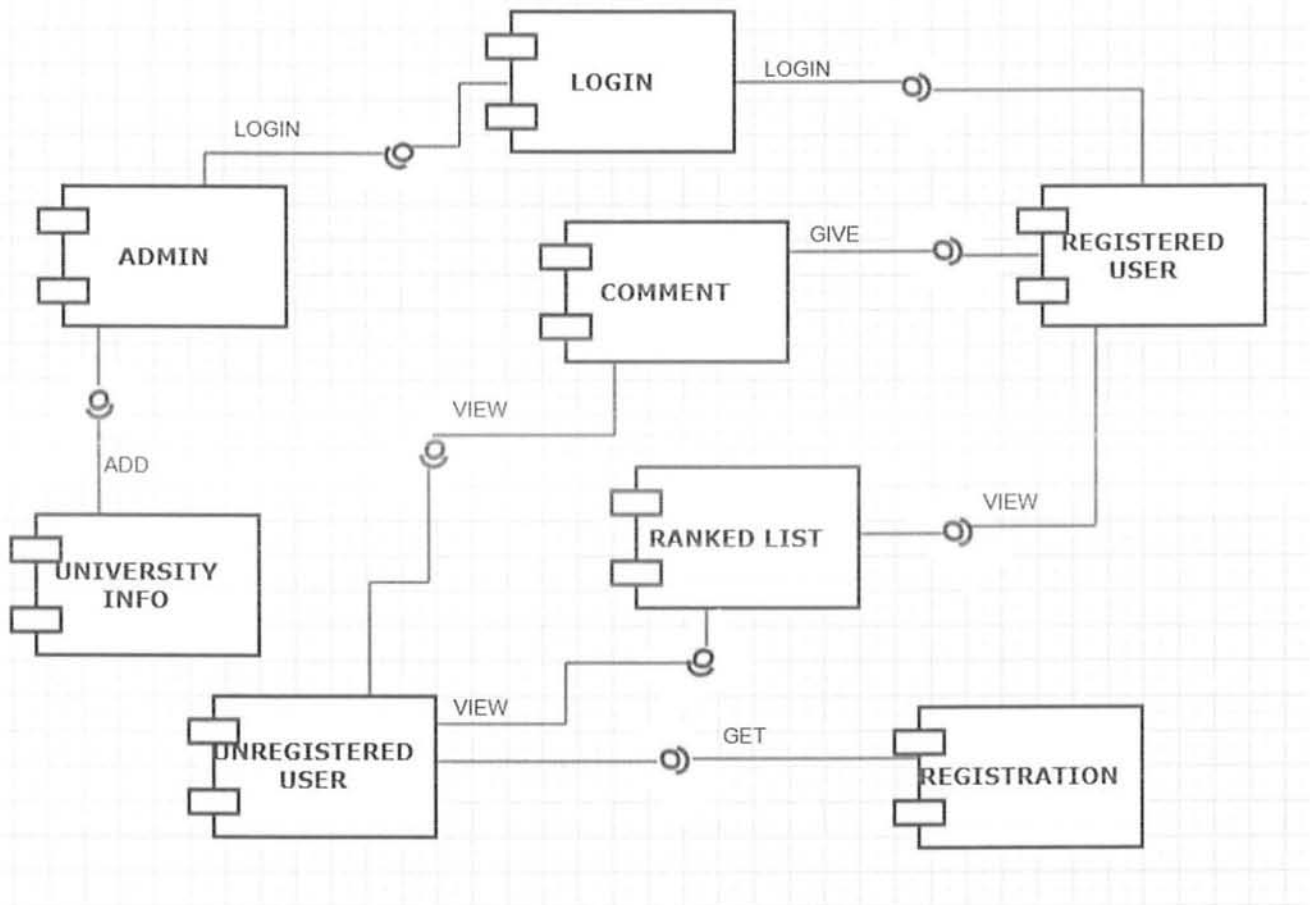


Figure 3.3 Component Diagram

3.5.3 Entity Relationship Diagram

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems.

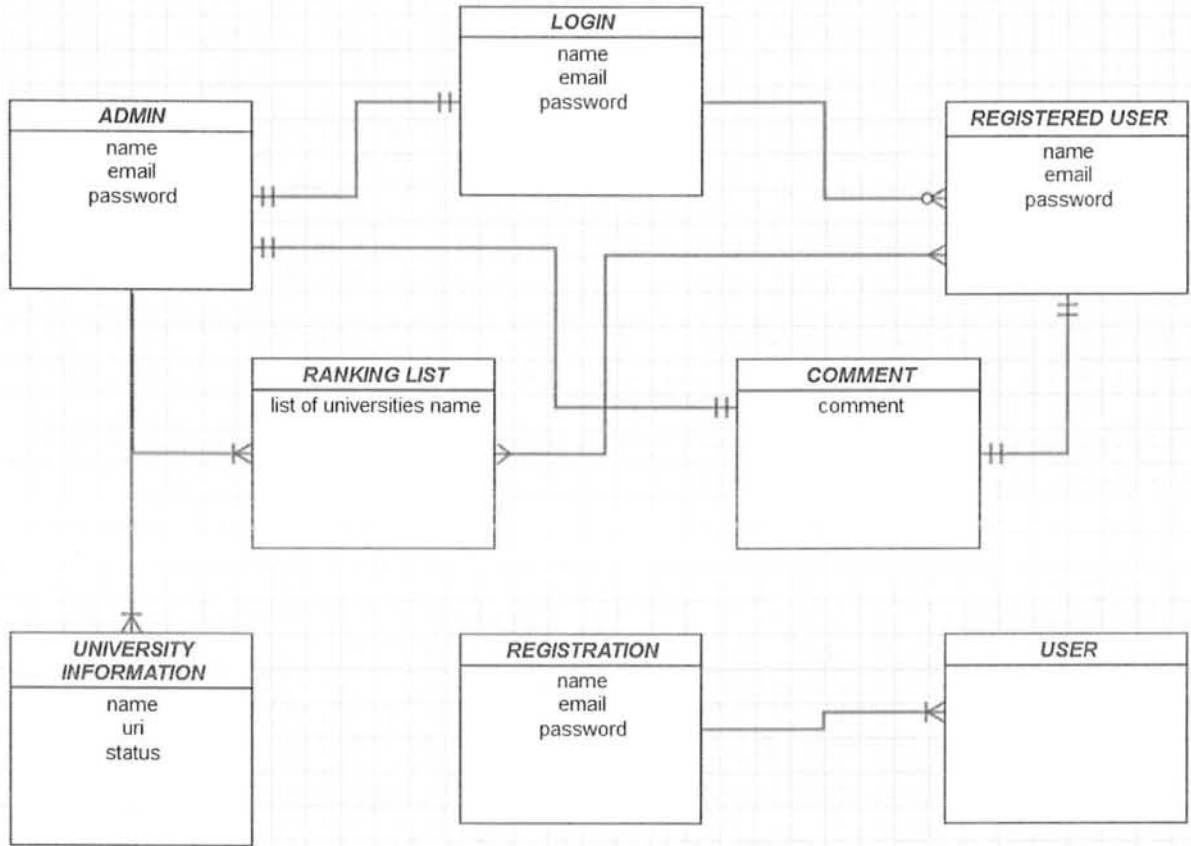


Figure 3. 4 Entity Relationship Diagram

3.6 DETAILED DESCRIPTION OF COMPONENTS

3.6.1 Login

In this component registered user and admin has to login itself. Login is used for authentication. In this user has to give its password and username/email for authentication. User must be registered to login. After login registered user can give his/her comment against any university and also can view ranking list. Admin can add new university name and it's URI.

3.6.2 Admin

Admin must be registered and logged in. Admin has to add university name and it's URI. For example, Quaid-I-Azam University and www.qau.edu.pk. Admin can edit the university name and URI. Also, admin can delete it.

3.6.3 Registered User

Registered User (RU) must be logged in to the system. RU can give comments and can view the comments posted by other peoples. Also, RU can view the ranked list.

3.6.4 Unregistered User

Unregistered user can register him by registration. It can only view comments and ranking list

3.6.5 Comment

Every registered user and admin can post a comment against a university website

3.6.6 Database

There will be all information stored regarding logins, comments, Universities information and ranking list

3.6.7 Ranking list

It will contain university name with ranking number according to comments given by users.

3.6.8 Registration

Here in this component all the unregistered users can get register him. It is required to provide name, username, password, email.

3.6.9 University Information

University information will be required for example: its name and URI. Admin has to add the university information so that users can give comments against the university website.

3.7 USER INTERFACE DESIGN

User interface design creates an effective communication between user and a computer. User interface design begins with the identification of user, task, and environmental requirements.

3.7.1 Description of User Interface

User interface is the part of software and is design in such a way that is expected to provide the user insight of the software. User interface provide the fundamental platform for human- computer interaction. User interface can be graphical, command line and text-based. There are large number of activities performs for designing user interface. These activities include interface analysis and modelling, interface design, design construction and design validation. Interface analysis and modelling focus on profile of user that will interact with the system. Interface design is to define a set of interface objects and actions that enable a user to perform all defined tasks in a manner that meets every usability goal defined for the system. Interface construction normally begins with the creation of a prototype that enables usage scenarios to be evaluated. Finally, interface validation focuses on the ability of the interface to implement every user task correctly, to accommodate all task variations, and to achieve all general user requirements. User interface divided in to two categories; graphical user interfaces (GUI) and command line interface (CLI). Command line interface provides a command prompt, the place where the user types the command and feeds to the system. Graphical User Interface (GUI) provides the user graphical means to interact with the system.

Table 3.2 User Interface

Characteristic	Description
Window	Multiple windows allow different information to be displayed on the user's screen.
Icon	Icons different types of information. On some systems, icons represent files; on others, icons represent processes.
Menu	Commands are selected from a menu rather than typed in a command language.
Pointing	A pointing device such as a mouse is used for selecting choices from a menu or indicating items of interest in a window.
Graphics	Graphical elements can be mixed with text on the same display.

3.7.2 Sequence Diagram

Sequence diagram are used to model the interaction between the actors and the objects in a system and interaction between the objects themselves. A sequence diagram shows that interactions that take place during a particular use case or use case scenario.



3.7.2.1 Admin Login

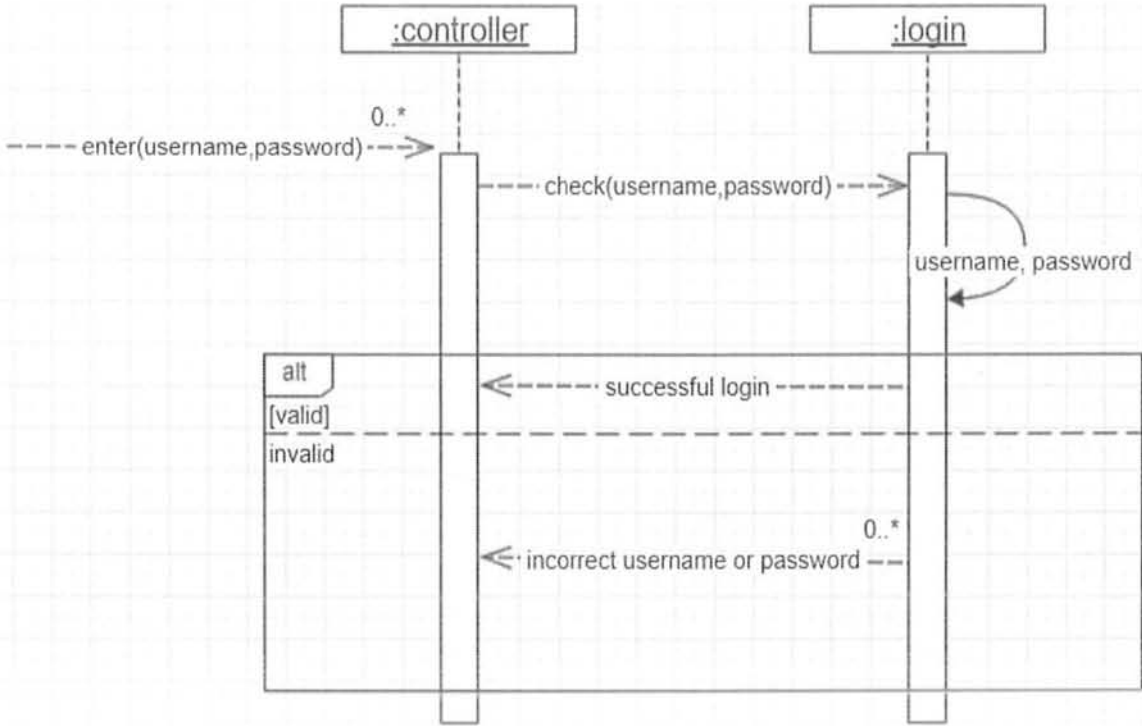


Figure 3. 5 Admin Login Sequence Diagram

Sequence diagram of admin login shows that sequence of interaction that take place when the admin wants to login to the system. User enters id and password to the system then system validates id and password. If admin is authorized then admin will be successfully login to the system else error message will be displayed to the user

3.7.2.2 Registered User login

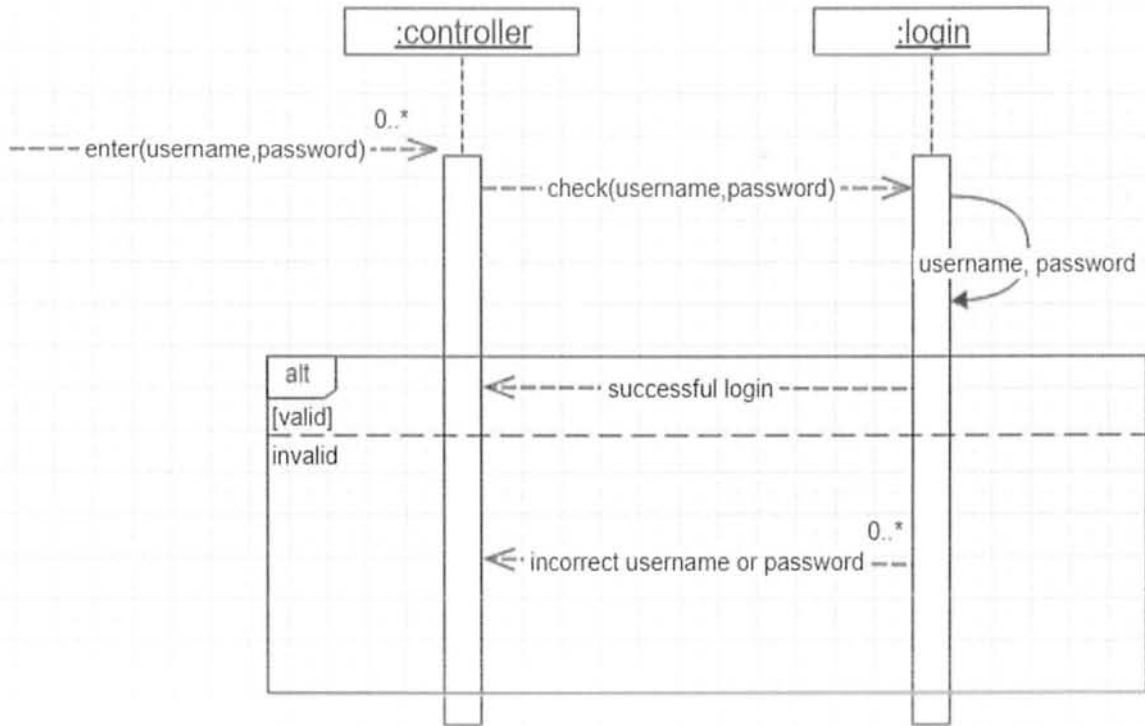


Figure 3. 6 Registered User Login Sequence Diagram

Sequence diagram of Registered user login shows that sequence of interaction that take place when the admin wants to login to the system. User enters id and password to the system then system validates id and password. If registered user is authorized then registered user will be successfully login to the system else error message will be displayed to the user

3.7.2.3 Add university information

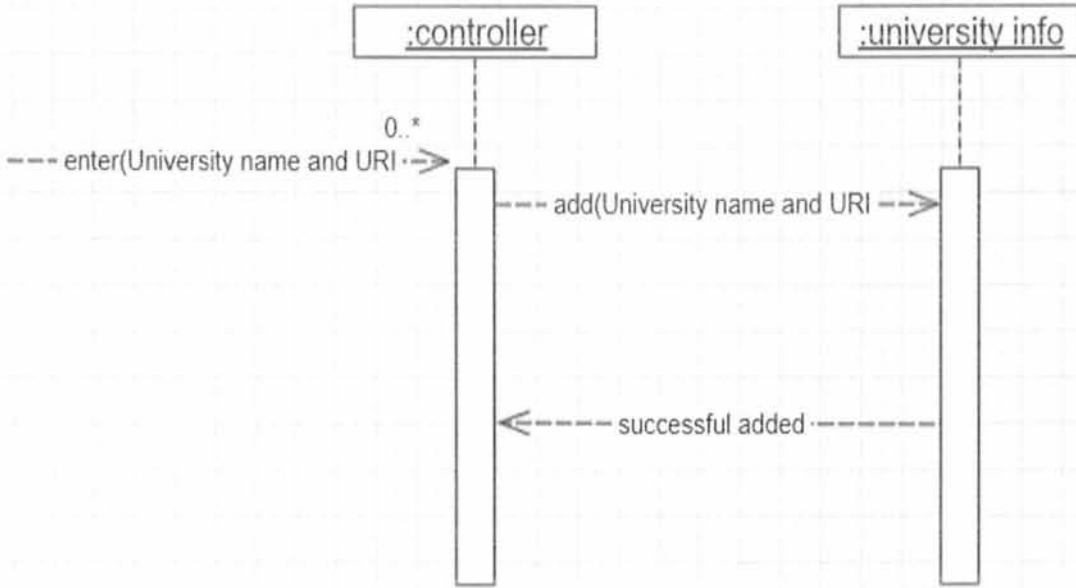


Figure 3. 7 Add university info Sequence Diagram

Sequence diagram of add university information admin has to add university name and it's URI. If it is valid it will be added into system.

3.7.2.4 Edit University Information

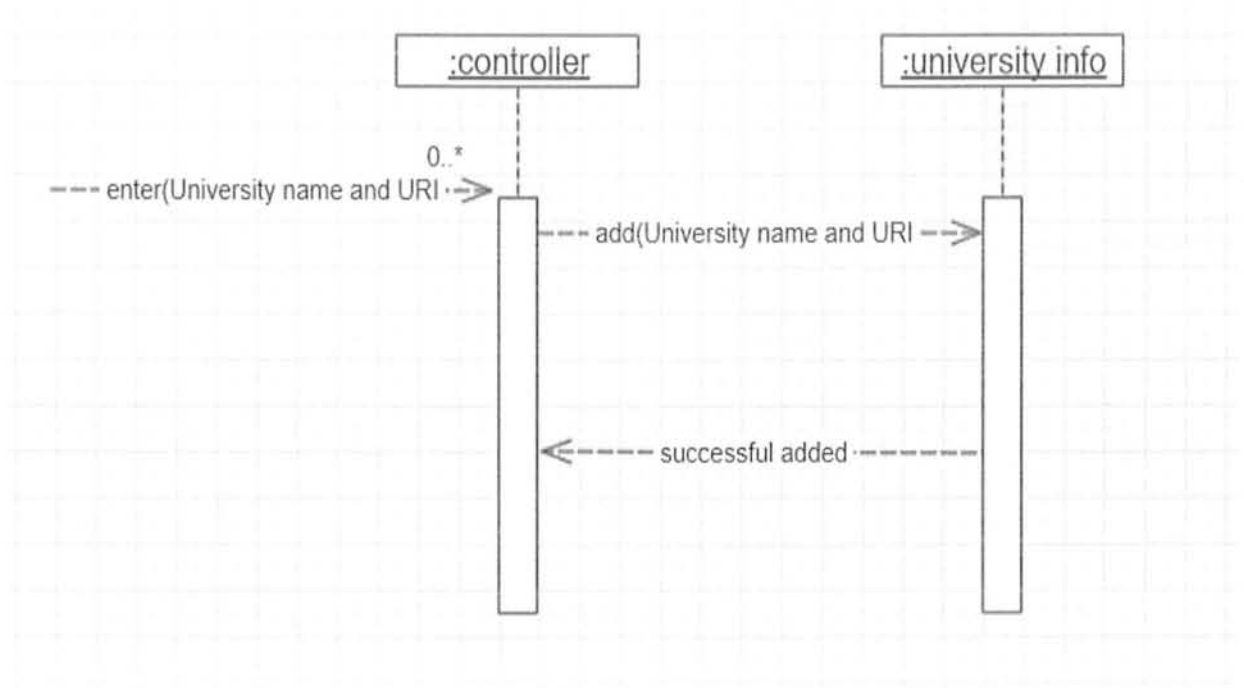


Figure 3. 8 Edit university info Sequence Diagram

Sequence diagram of Edit university information admin can edit university name and it's URI. If it is valid it will be added into system.

3.7.2.5 Delete University Information

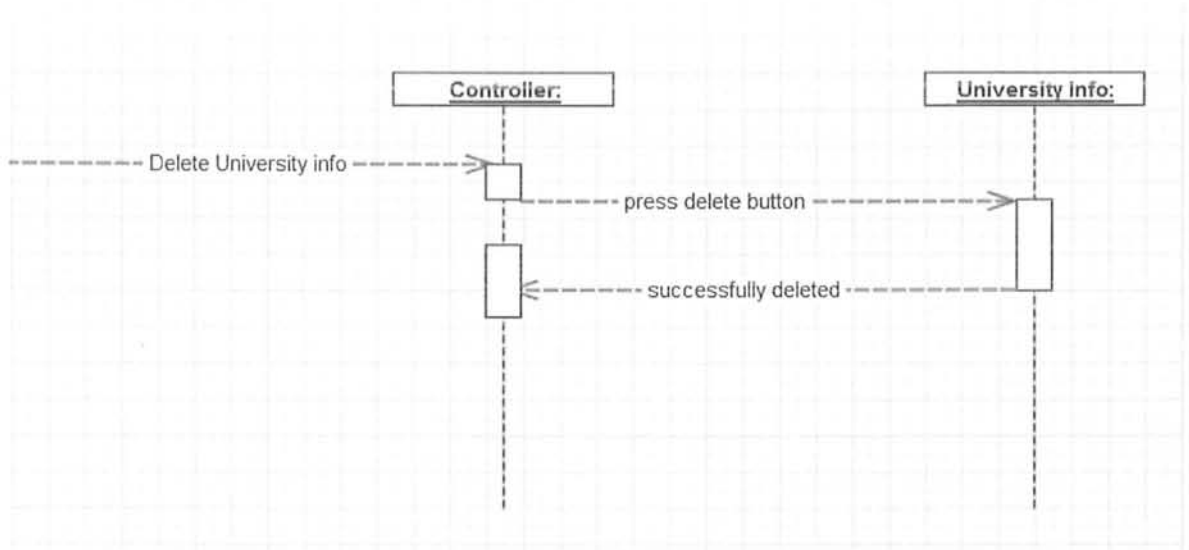


Figure 3. 9 Delete University Info sequence diagram

Sequence diagram of Delete university information admin can delete university name and it's URI. If it is valid it will be added into system.

3.7.2.6 Registered user / Admin Add comments

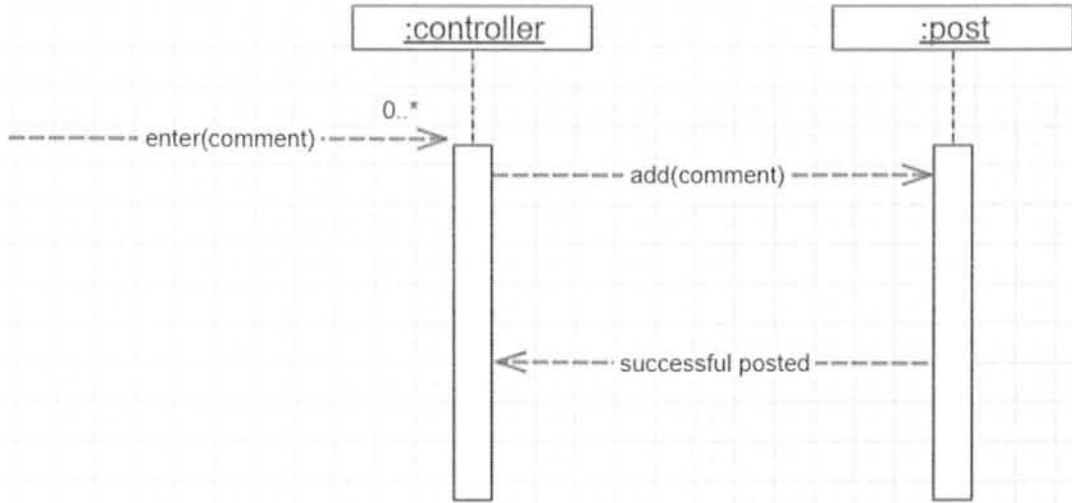


Figure 3. 10 Registered User add comment Sequence Diagram

Sequence diagram of registered user add comment shows that sequence of interaction that take place when the registered user wants to add comment to the system. Registered user will write a comment and submit it to the system

3.7.2.7 View comment sequence diagram

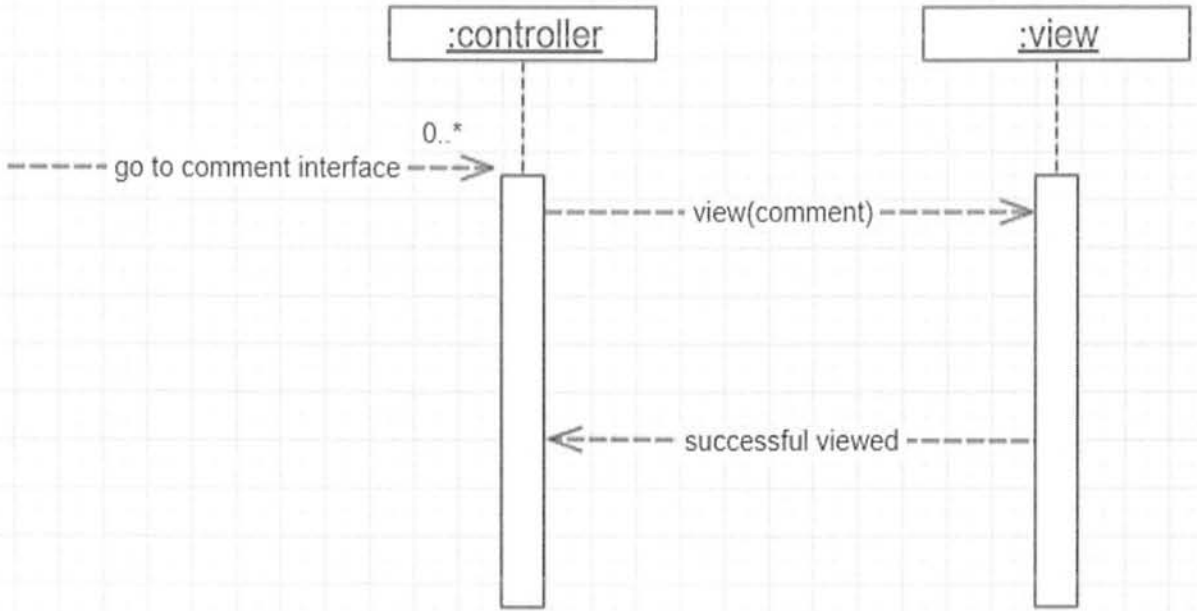


Figure 3. 11 View comment Sequence Diagram

Sequence diagram of View Comment shows that sequence of interaction that take place when the registered user/user/admin wants to see comments. User will click on view comments and then interface will open displaying the comments.

3.7.2.8 View ranking list

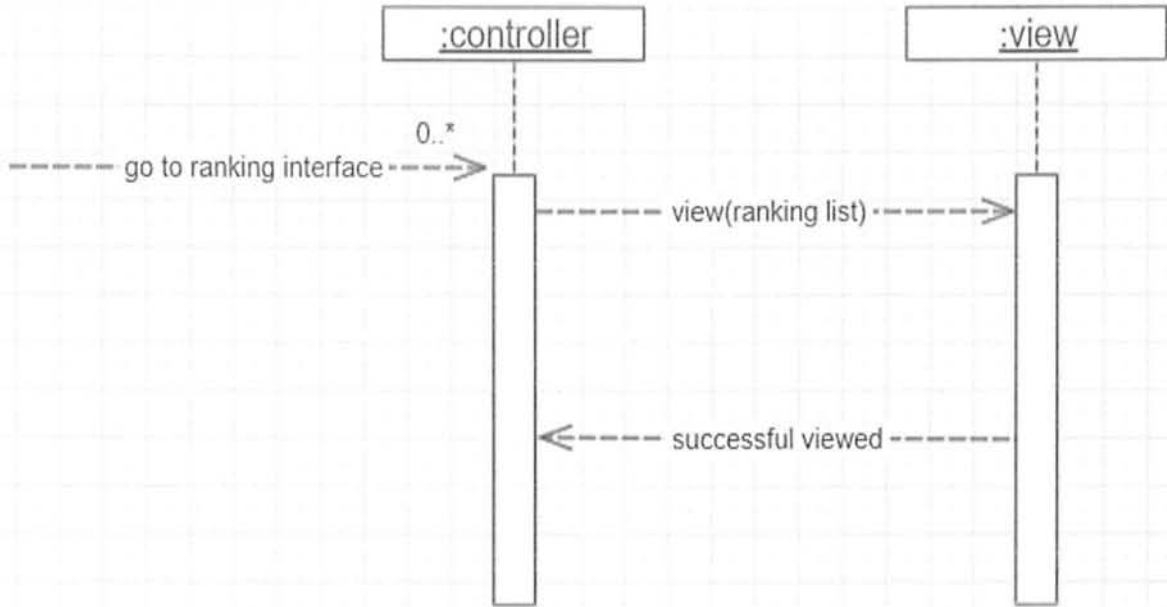


Figure 3. 12 View ranking list Sequence

Sequence diagram of ranking list shows that sequence of interaction that take place when the registered user/user/admin wants to see ranking list. User will click on view ranking list and then interface will open displaying the list of universities name.

3.7.2.9 Registration

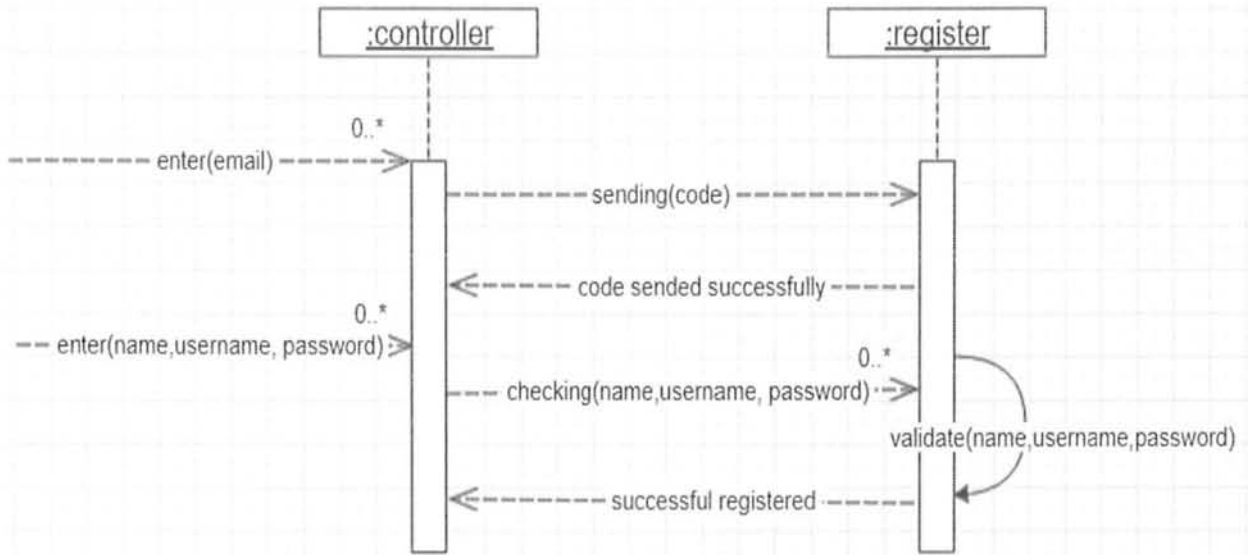


Figure 3. 13 Registration Sequence Diagram

Sequence diagram of registration shows that sequence of interaction that take place when a user wants to get registered to the system. User has to provide necessary information which is email username password. If the information is valid user will be successfully registered.

3.7.2.10 Update profile

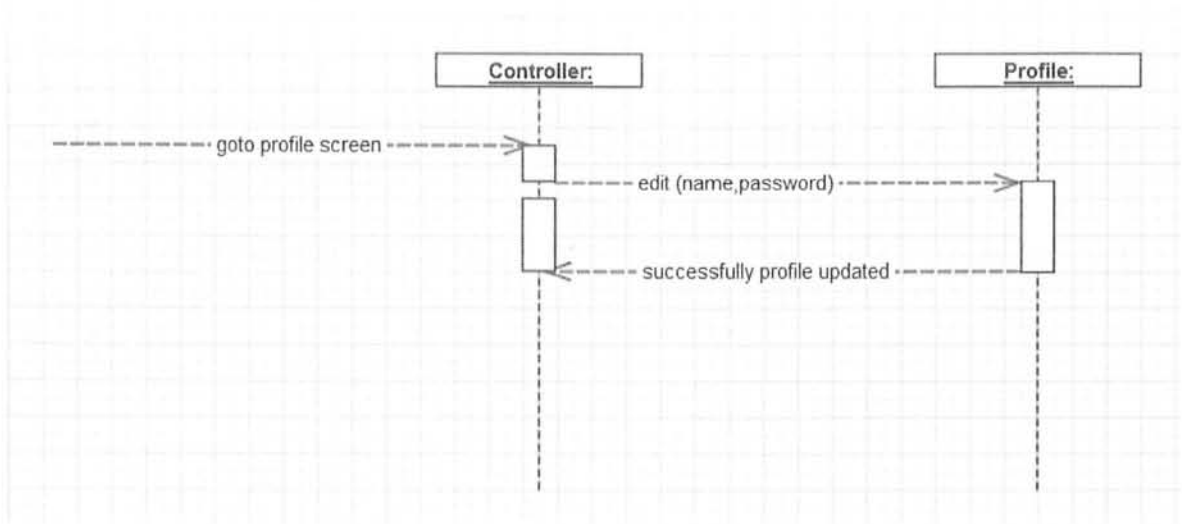


Figure 3. 14 Update Profile Sequence Diagram

Sequence diagram of update profile shows that sequence of interaction that take place when a user wants to update its personal information like name and password. User has to provide necessary information which is email username password. If the information is valid user will be successfully updated.

3.7.2.11 Delete Account

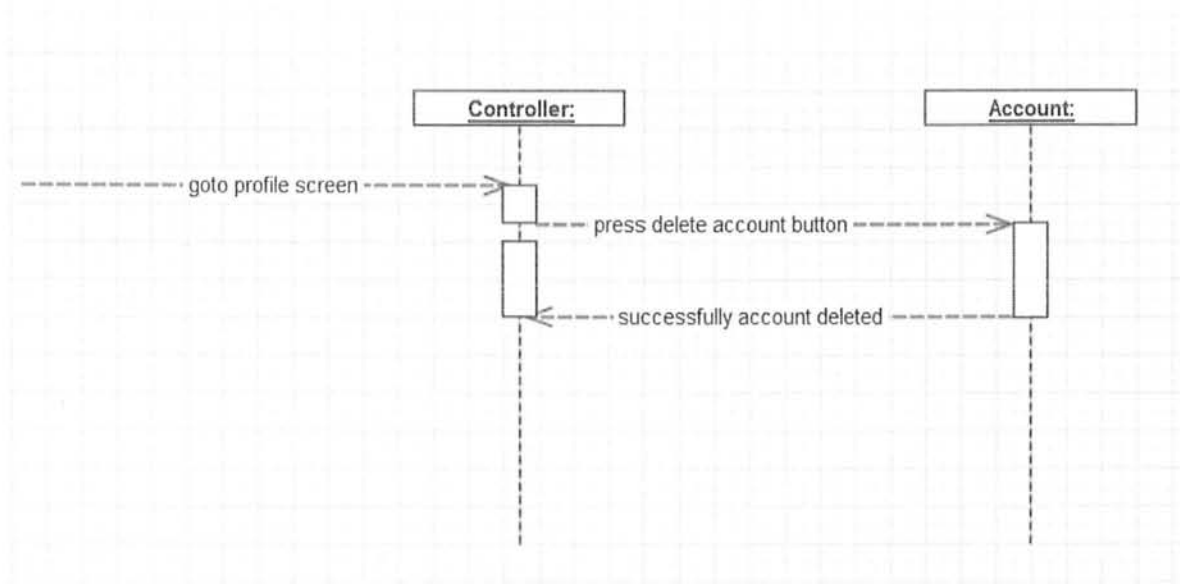


Figure 3. 15 Delete Account Sequence Diagram

Sequence diagram of Delete account shows that sequence of interaction that take place when a user wants to delete its account. User has to go on profile screen then scroll down, there will be a red button of delete account. Just press that button.

3.7.3 Class Diagram

The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application. The class diagram shows a collection of classes, interfaces, associations, collaborations and constraints. It is also known as a structural diagram. The purpose of the class diagram is to model the static view of an application. The class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

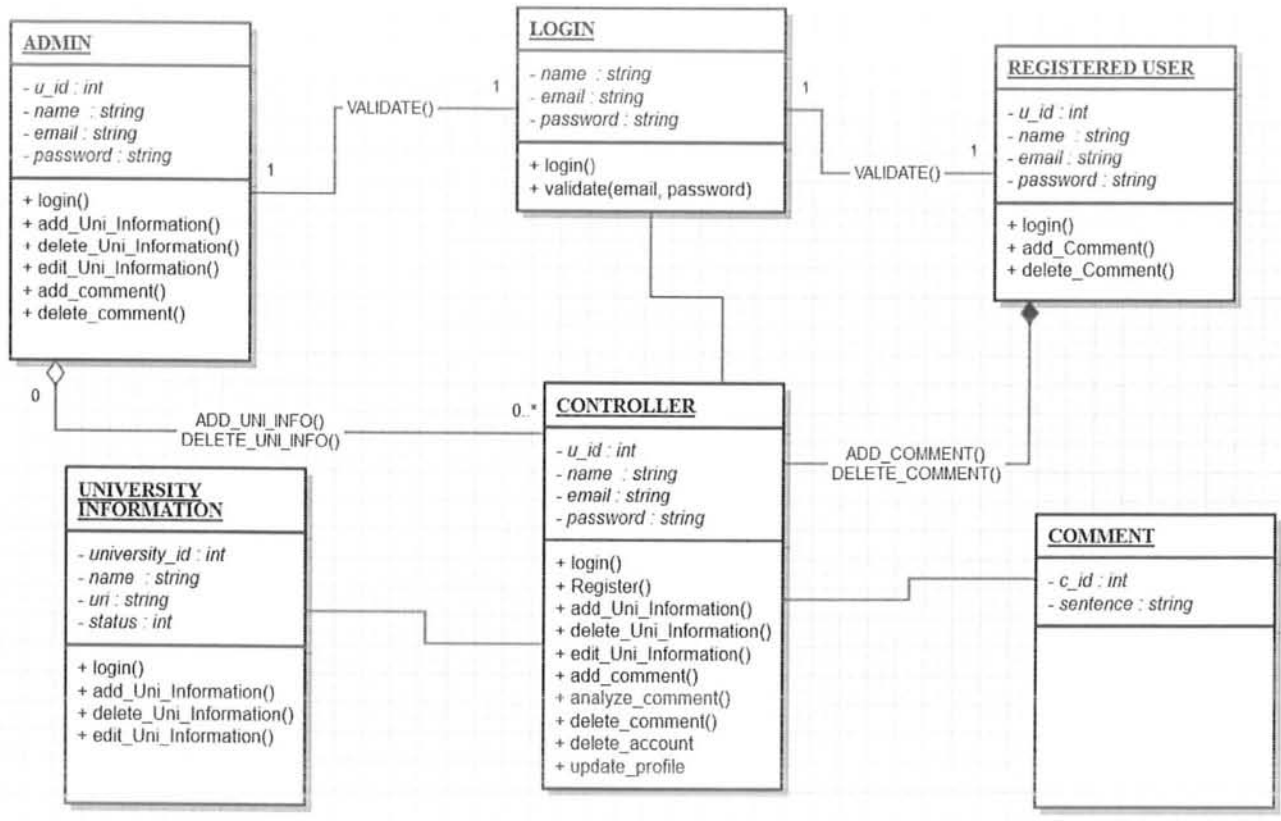


Figure 3. 16 Class Diagram

CHAPTER NO 4

4 SOFTWARE IMPLEMENTATION DOCUMENT

4.1 Introduction

This document describes the project implementation for developing the project planner and scheduler.

4.2 Language Selection

The project implements in the following languages:

- PHP 7.0.1
- Laravel 5.5
- WampServer 3.0.6
- MySQL 5.7.14
- Apache 2.4.23
- Use for database
- HTML/CSS
- Used for designing of web pages
- JavaScript
- Used for PHP is a general-purpose scripting language that is especially suited to server-side web development.

4.3 Tools Selection

- Notepad++
- EasyPHP Server
- Web Browser

4.4 Resources

Here are resources used while making project

4.4.1 Data Tables

Wamp used for store data in tables.

4.4.2 PHP Mailer Class

Used to send mails to newly register employees

4.4.3 Bootstrap Library

Use for the client-side browser to communicate with the server without having to perform a page refresh.

4.5 Screen Images

In this section I have some possible screen shots of my system.

4.5.1 Login Page

The screenshot shows the login page of the 'Universities' Website Evaluation System'. The page has a dark header with the site name on the left and navigation links 'Home', 'Login', and 'Register' on the right. The main content area is a white box with a dark border. At the top of this box is a dark rounded rectangle containing the word 'LOGIN' in white. Below this, there are two input fields: 'E-Mail:' and 'Password:'. Under the 'Password:' field, there is a 'Login Me' button and a link that says 'Oh! I Have Forgot My Password?'. At the bottom of the white box, there is a small copyright notice: 'Copyrights © 2018 www.evaluationsystem.com. All rights reserved.'

Figure 4. 1 Login screen image

4.5.2 Registration

The screenshot shows the registration page of the 'Universities' Website Evaluation System'. The page has a dark header with the site name on the left and navigation links 'Home', 'Login', and 'Register' on the right. The main content area is a white box with a dark border. At the top of this box is a dark rounded rectangle containing the text 'REGISTER YOUR SELF' in white. Below this, there are four input fields: 'Name:', 'E-Mail:', 'Password:', and 'Confirm Password:'. At the bottom of the input fields is a 'Register Me' button. At the bottom of the white box, there is a small copyright notice: 'Copyrights © 2018 www.evaluationsystem.com. All rights reserved.'

Figure 4. 2 Registration page screen image

4.5.3 Home Page

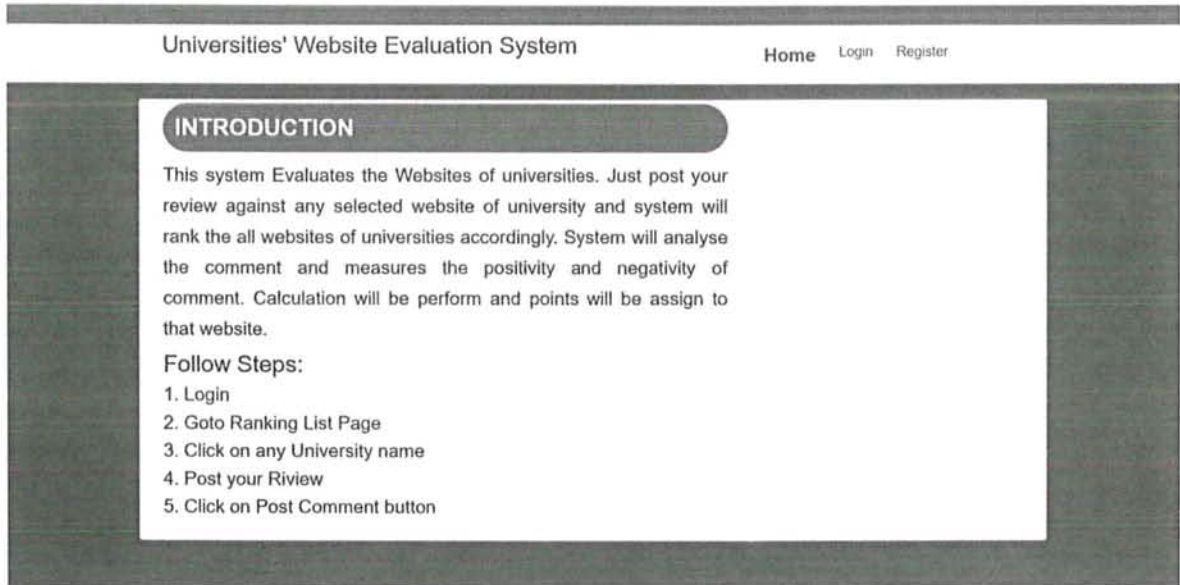


Figure 4. 3 Home page screen image

4.5.4 Add Comments Page

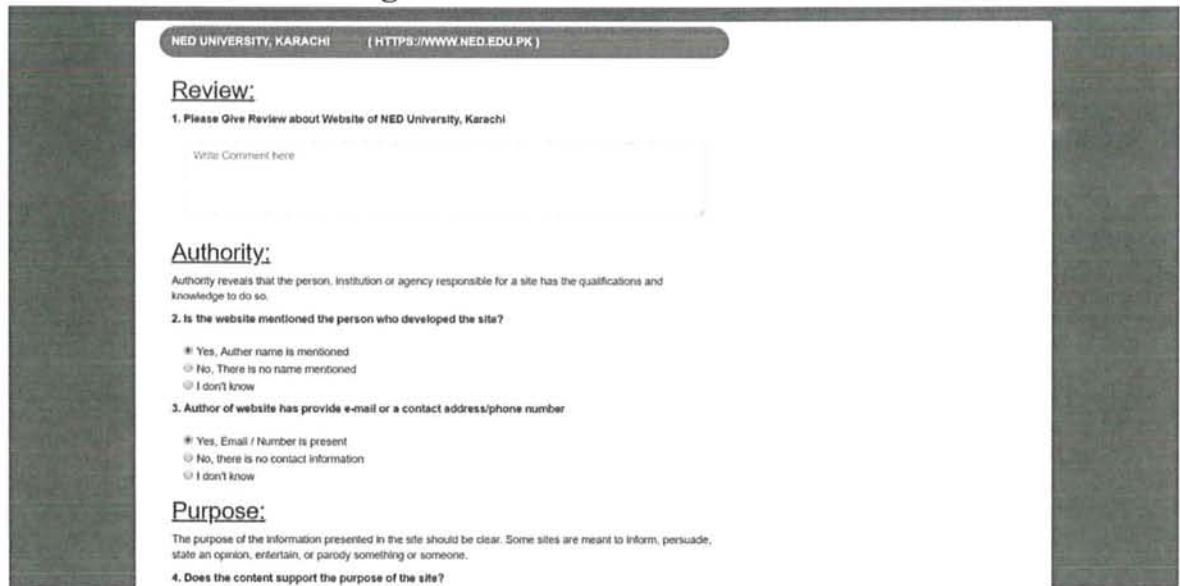


Figure 4. 4 Add comment screen image



4.5.5 Ranking list Page by Admin

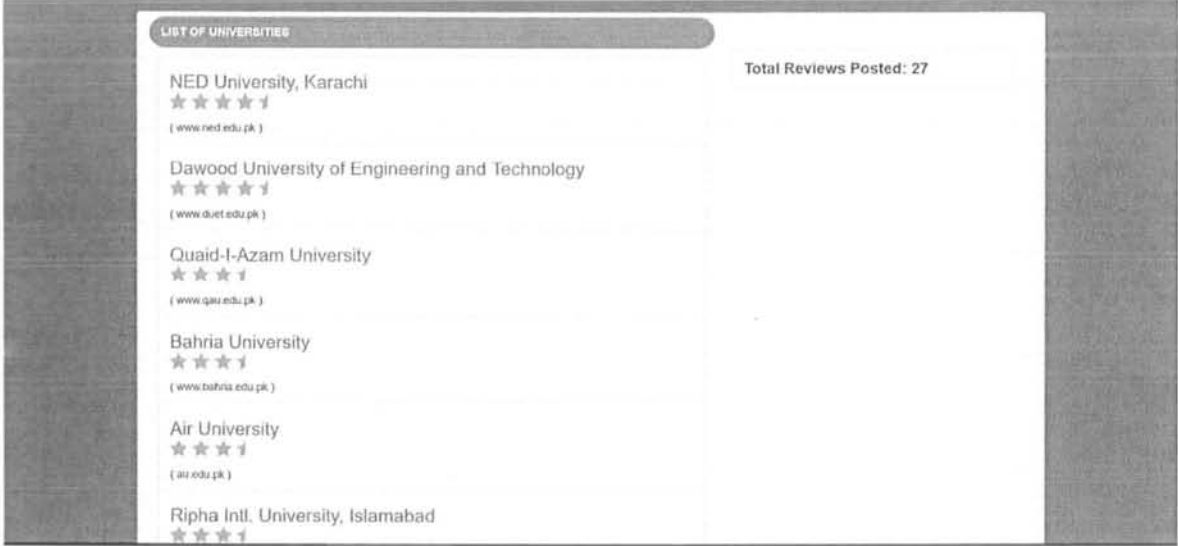


Figure 4.5 View ranking list by admin

4.5.6 Add University Information Page

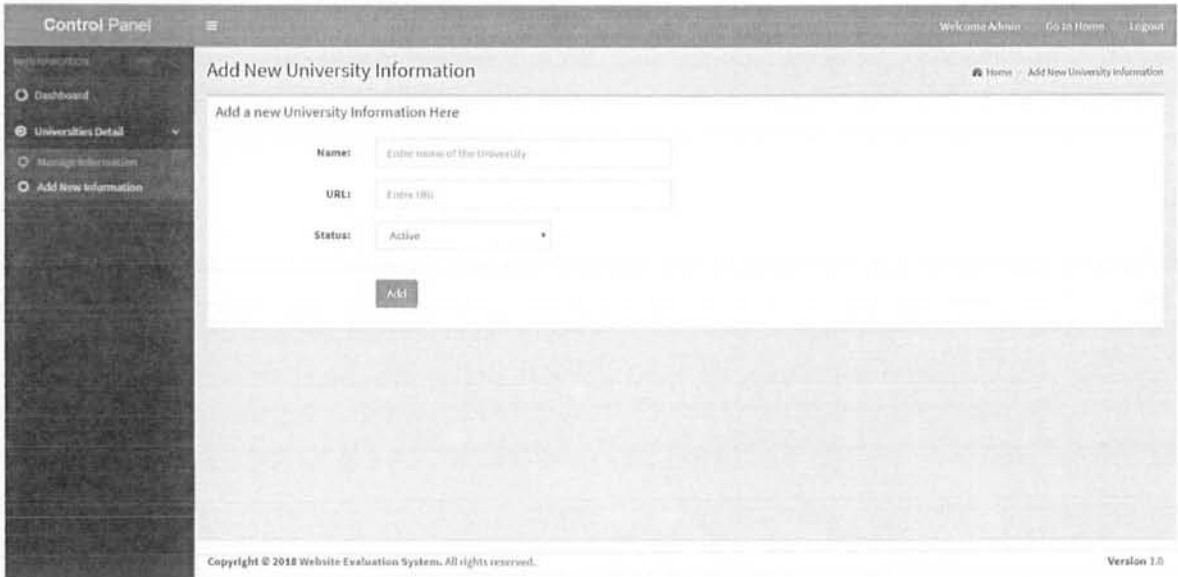


Figure 4. 6 Add university

4.5.7 Edit University information

Control Panel

Welcome Admin | Go to Home | Logout

Universities Detail

Modify University

Home | View All Universities | Modify University

Update university Comsats University Islamabad

Name: Comsats University Islamabad

URL: https://www.comsats.edu.pk

Status: Active

Update

Copyright © 2018 Website Evaluation System. All rights reserved. Version 1.0

Figure 4. 7 Edit university

4.5.8 Delete University information

Control Panel

Welcome Admin | Go to Home | Logout

localhost says
Are you sure you want to delete this?

OK Cancel

Home | View Universities

View Universities

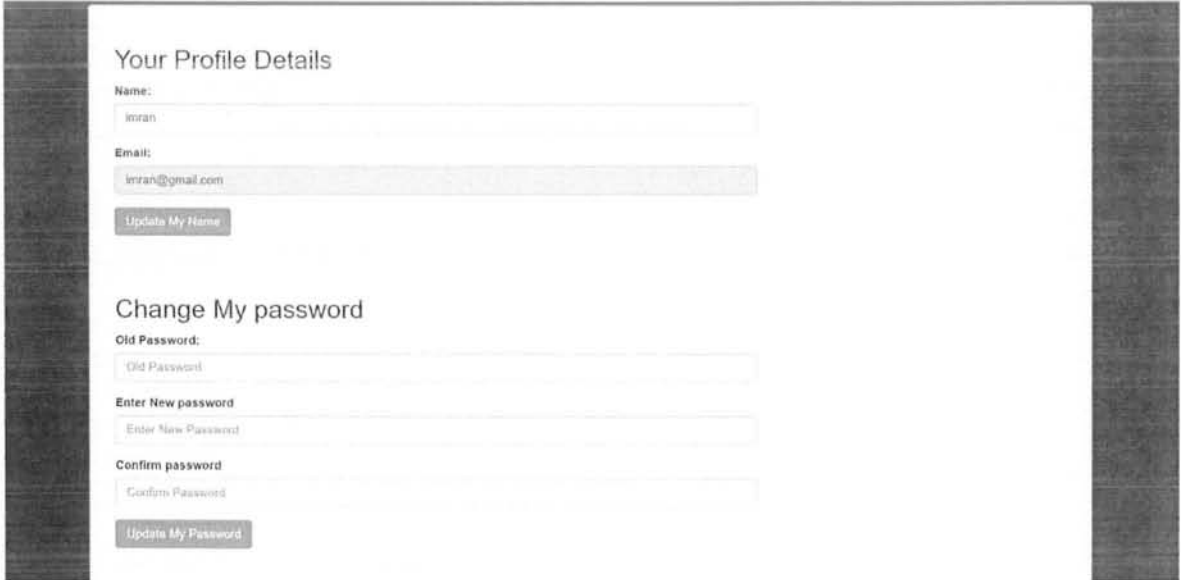
Manage Universities' Information

ID	University Name	University URL	Status	Options
2	Comsats University Islamabad	https://www.comsats.edu.pk	Active	
1	Quaid-i-Azam University	https://www.qau.edu.pk	Active	
3	National University of Science and Technology	https://www.must.edu.pk	Active	
4	NED University, Karachi	https://www.ned.edu.pk	Active	

Copyright © 2018 Website Evaluation System. All rights reserved. Version 1.0

Figure 4. 8 Delete university

4.5.9 Update Account



The screenshot displays a web form titled "Your Profile Details". It contains two main sections: "Name" and "Email", each with a text input field and a corresponding "Update My Name" button. Below this is a section titled "Change My password" with three input fields: "Old Password", "Enter New password", and "Confirm password", each followed by an "Update My Password" button.

Your Profile Details

Name:

Update My Name

Email:

Update My Name

Change My password

Old Password:

Update My Password

Enter New password

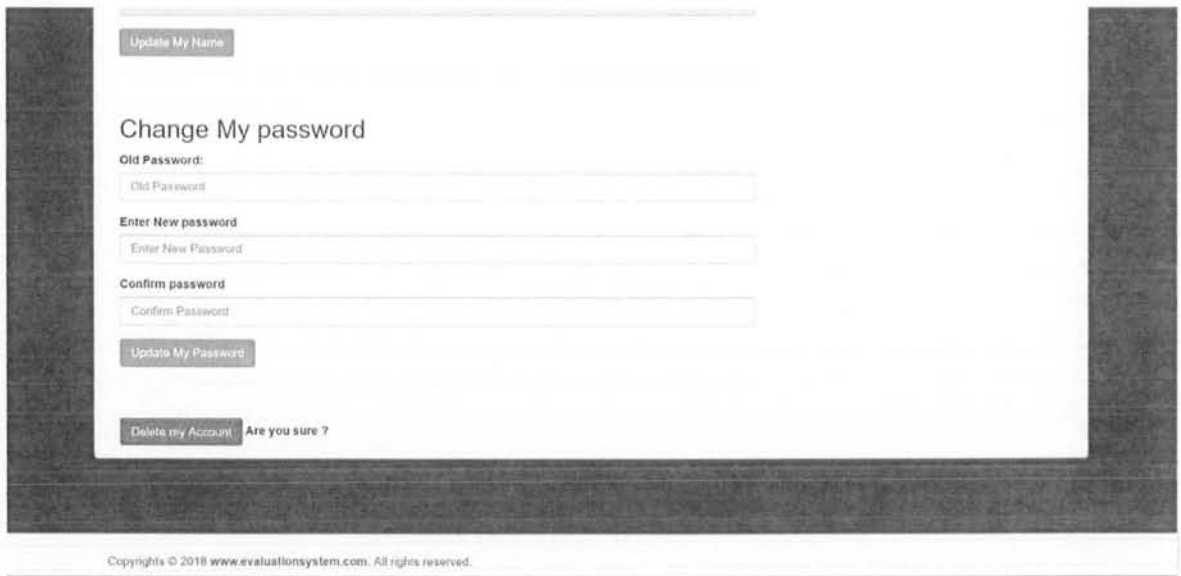
Update My Password

Confirm password

Update My Password

Figure 4. 9 Update account

4.5.10 Delete Account



The screenshot displays a web form titled "Change My password". It contains three input fields: "Old Password", "Enter New password", and "Confirm password", each followed by an "Update My Password" button. At the bottom, there is a "Delete my Account" button and a confirmation text "Are you sure?".

Change My password

Old Password:

Update My Password

Enter New password

Update My Password

Confirm password

Update My Password

Delete my Account Are you sure ?

Copyrights © 2018 www.evaluationssystem.com. All rights reserved.

Figure 4. 10 Delete account

This chapter has given complete description of system implementation. It has elaborated the operational and functions of the system. This chapter further explained the different approaches, language selection, tools and environment for complete implementation a system and finally contained the details description of frame work selected, software used and screen shots of the application.

CHAPTER NO 5

5 SOFTWARE TEST DOCUMENT

This chapter describes software testing processes. This chapter further elaborates the acceptance test cases which are used to test the functional and non-functional requirements after coding of software.

5.1 INTRODUCTION

Testing can only show the presence of errors in the program. It cannot demonstrate that there are no remaining faults. Software test document involves the documentation of artefacts that should be developed before or during the testing of software. Software testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. Testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

5.2 Test Approach

The testing approach describes the process of conducting a test. There are two major approaches in which the testing can be done. One on the basis of execution and other is on the basis of application level. On the basis of execution contains static testing, symbolic testing and dynamic testing. On the basis of application level contains system testing and unit testing. Static testing of program is done without executing the program. It is typically done by a compiler which checks for syntax errors and control flow errors such as unreachable code.

Symbolic testing is carried out by providing symbolic inputs to the software and executing the code by symbolically evaluating the program variables. Dynamic testing requires execution of the program using input data. Here the usual approach is to select the input data values such that desired control paths are executed. Since there can be infinite number of control paths in a program, dynamic test cases are designed to satisfy a minimal number of conditions that indicate the extent of control paths or alternative criteria that are covered in the test cases. System testing is carried out for the entire application and verifies that the product an assemblage of components works as a cohesive whole to satisfy the user requirements. Unit testing, on the other hand, carries out tests at the component (unit) level.

The selected approach for testing is unit testing. In unit testing is the process of testing program components, such as methods or object classes. A unit test case provides the input parameter values and also provides the expected results when the code is executed. The unit test is carried out to verify the results of the module against the expected results

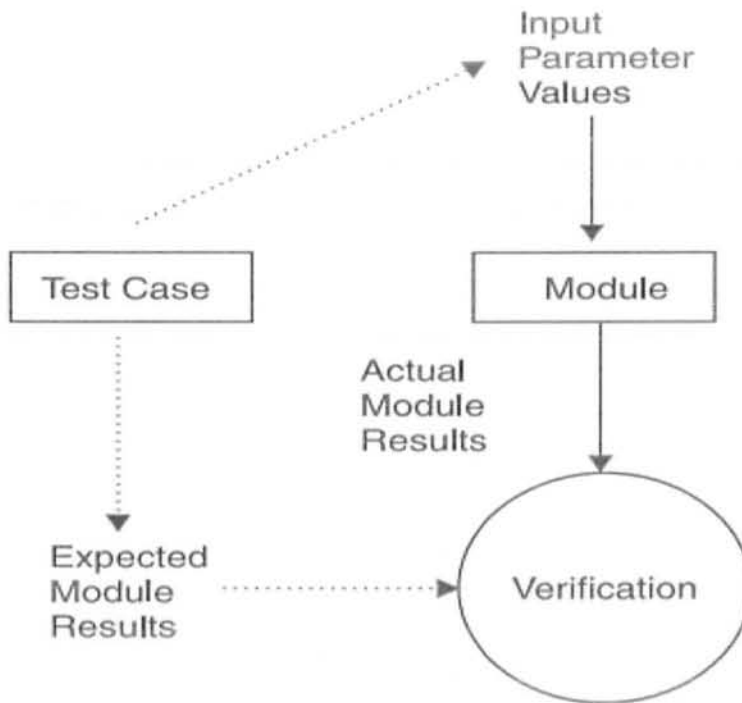


Figure 5. 1 Unit Test

Unit testing contains functional (black box) testing and structural (white box) testing. Black box testing is that box which does not make use of knowledge of the internal logic of the module is unknown. Testing can be made on interface, testing based on function to be computed and input domain testing. White box testing uses internal logic of the modules and it can be system testing, branch testing, and conditional testing.

5.3 TEST PLAN

Test planning is an activity that ensures that there is initially a list of tasks and milestones in a baseline plan to track the progress of the project. Test plan determines the scope and the risk that need to be tested and are not to be tested. Deciding fail and pass criteria.

5.4 Features to be tested

Features of the system that is to be tested using a specific approach. These features include, login (admin/registered user), add comment, delete comment, add university information, and delete university information, registration.

5.5 Features not to be tested

There are some features that will not be tested for example to view comments and to view ranking list.

5.6 TEST CASES

A test case is a document, which has a set of test data, preconditions, expected results and post conditions, developed for a particular test scenario in order to verify compliance against a specific requirement. Test Case acts as the starting point for the test execution, and after applying a set of input values; the application has a definitive outcome and leaves the system at some end point or also known as execution post condition.

Table 5.1 Login

ID	T1
Description	Registered User and Admin can login to the system. It shows that login of Registered user and admin can only be possible if they are already registered and they put valid username and password
Actor	Registered User/Admin
Setup	1. Admin Email: admin@gmail.com Password: imran321. 2. Registered user Email: imran@gmail.com Password: aslam321
Inputs:	1. Enter Email: admin@gmail.com and Password imran321 2. Press login 3. Enter Email: admin@gmail.com and password aslam321 4. Press login
Expected Results	admin as admin should log in successfully imran as user should log in successfully
Result/Verdict	Passed

Table 5.2 Add comment

ID	T2
Description	Registered user and admin can add comment regarding any universities' website. Length of comment must be greater than 3 characters. Registered must be logged in
Actor	Registered User
Setup	Login as Registered user
Inputs:	1. Enter Comment 2. Press Add comment button
Expected Results	comment will be uploaded to the system
Observed Results	comment will be uploaded to the system
Output/Verdict	Passed

Table 5.3 Delete Comment

ID	T3
Description	Comment will be deleted from the system. Only Registered user delete comment
Actor	Registered user
Setup	Login as Registered user
Inputs:	1. Select the option near to comment 2. Click delete button.
Expected Results	Comment will be deleted from the system
Output/Verdict	Passed

Table 5.4 Add University Information

ID	T4
Description	Only admin can add information regarding university. Admin must be logged in. Admin will add university name and its URI
Actor	Admin
Setup	Login as Admin
Inputs:	1. Enter University name and its URI 2. Press Add Information button
Expected Results	Information will be uploaded to the system
Output/Verdict	Passed

Table 5.5 View comment

ID	T5
Description	Registered user and Unregistered user can view comments. No need to logged in
Actor	Registered User and Unregistered user
Setup	Open the comments page
Inputs:	1. Go to home page 2. Press on any university link
Expected Results	comments will be displayed by the system to the user
Output/Verdict	Passed

Table 5.6 View Ranking List

ID	T6
Description	Registered user and Unregistered user can view Ranking List. No need to logged in
Actor	Registered User and Unregistered user
Setup	Open the Ranking List page
Inputs:	1. Go to home page
Expected Results	Ranking List will be displayed by the system to the user
Output/Verdict	Passed

Table 5.7 Registration

ID	T7
Description	Unregistered user can register it self
Actor	Unregistered user
Setup	Open the Registration page
Inputs:	1. Enter Name 2. Gmail: imran@gmail.com 3. Enter password: saleem321 4. press submit button
Expected Results	Unregistered user will be registered
Output/Verdict	Passed

Table 5.8 Update Profile

ID	T8
Description	Registered user can update its profile
Actor	Registered user
Setup	Open logged in page
Inputs:	1. click on your name 2. Enter Name: imran 3. press update button
Expected Results	Profile will be updated
Output/Verdict	Passed

Table 5.9 Delete Account

ID	T9
Description	Registered user can delete its profile
Actor	Registered user or admin
Setup	Open logged in page
Inputs:	1. click on your name 2. press delete account button
Expected Results	Profile will be updated
Output/Verdict	Passed

Table 5. 10 Delete University Information

ID	T10
Description	Only admin can Delete information regarding university. Admin must be logged in. Admin must click on delete button
Actor	Admin
Setup	Login as Admin
Inputs:	1. Click on delete button
Expected Results	Information will be Deleted from the system
Output/Verdict	Passed

Table 5. 11 Edit University Information

ID	T11
Description	Only admin can Edit information regarding university. Admin must be logged in. Admin can Edit university name and its URI
Actor	Admin
Setup	Login as Admin
Inputs:	1. Enter University name and its URI 2. Press Submit Information button
Expected Results	Information will be uploaded to the system
Output/Verdict	Passed

References:

- [1] Ian_Sommerville_Software_Engineering,_9th_Edition____2011
- [2] Applying UML And Patterns LARMAN
- [3] Software Engineering A Practitioner's Approach 7th Edition - Roger Pressman
- [4] <https://medium.com/deep-math.../chapter-4-decision-trees-algorithms-b93975f7a1f1>
- [5] <https://www.techopedia.com/definition/1181/data-mining>
- [6] Design Concepts and Principles, Software Engineering A Practitioner's Approach by R.S. Pressman
- [7] Software Requirement Specifications Software Engineering A lifecycle approach by P. Mohapatra
- [8] Class Diagrams Software Engineering A lifecycle approach by P. Mohapatra
- [9] SoftwareTestingIan_Sommerville_Software_Engineering_9th_edition
- [10] Software engineering A lifecycle approach by P. Mohapatra
- [11] https://www.tutorialspoint.com/software_testing_dictionary/test_plan.htm
- [12] H. Wu and R. Luk and K. Wong and K. Kwok. "Interpreting TF-IDF term weights as making relevance decisions". ACM Transactions on Information Systems, 2008.
- [13] Marie-Francine Moen, "Innovative Techniques for Legal Text Retrieval". Artificial Intelligence and Law vol.
- [14] pp. 29–57, 2001 9 What does tf-idf means." www.tfidf.com (Retrieved on December 2016).
- [15] Source of positive words and negative words file:
ptrckprry.com/course/ssd/data/negative-words.txt for negative file
ptrckprry.com/course/ssd/data/positive-words.txt for positive file

