**AI Based Online Exam Proctoring System**



**By Usama Bin Mazhar**

**(IS/BS (CS)/S-16/M/056)**

**Muhammad Waqas Qadir**

**(IS/BS (CS)/S-16/M/027)**

**Supervised by**

Mr. Khawaja Tahir

**A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE DEGREE OF**

**BACHELOR OF SCIENCE IN COMPUTER SCIENCE BS (CS)**

## FEDERAL URDU UNIVERSITY OF ARTS, SCIENCE AND TECHNOLOGY ISLAMABAD 2020

Dedicated to our beloved parents and to all those, whose prayers always pave the way to success for us.

# PROJECT IN BRIEF

|  |  |  |
| --- | --- | --- |
| Project Title | : | AI Based Online Exam Proctoring System |
| Organization | : | Federal Urdu University of Arts, Science and Technology |
| Objectives | : |  |
| Undertaken By | : |  |
| Supervised By | : | Mr. Khawaja Tahir |
| Date Started | : |  |
| Date Completed | : |  |
| Technologies Used | : |  |
| Operating System | : |  |
| Web Server | : |  |
| System Used | : |  |

# ACKNOWLEDGEMENT

Thanks to Almighty Allah for giving us knowledge, power and strength to accomplish this task.

We learned a lot while doing this project and this will certainly help us in our forth coming life. Many friends of us helped us during this project but we are really thankful to the help and support of Mr. Khawaja Tahir Associate Professor from Federal Urdu University of Arts, Science and Technology, Islamabad who helped us in all the phases of this project. His supervision helped us a lot during the times of difficulties. In the end we would like to thank all of our friends for their support and encouragement.

**BADAR KHALIL**

**MUHAMMAD ALTAF**

**SANA AFTAB ABBASI**

# DECLARATION

We hereby declare that this software, neither as a whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No portion of the work presented has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

**BADAR KHALIL**

**MUHAMMAD SAAD ALTAF SANA AFTAB ABBASI**

# CERTIFICATION

It is certified that the contents and form of the project entitled “Computer Science Department

Automation System” has been found satisfactory for the requirements of

***FEDERAL URDU UNIVERSITY OF ARTS, SCIENCE AND TECHNOLOGY,***

***ISLAMABAD***

**For the award of the degree of**

### BACHELOR OF SCIENCE IN COMPUTER SCIENCE BS (CS)

|  |  |
| --- | --- |
| **Supervisor:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Mr. Saeed Ullah |
| **Member 1:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Mr. Badar Khalil |
| **Member 2:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Mr. Muhammad Saad Altaf |
| **Member 3:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Ms. Sana Aftab Abbasi |
| **Dated:** | **HoD: \_**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Dr. Saeed Ullah

# ABSTRACT

This system is designed for Federal Urdu University of Arts science and technology refers to the varied computer machinery and software used to digitally create, collect, store, manipulate, and relay information needed for accomplishing basic tasks like collecting Data of candidates and teachers required for NCEAC accreditation. This system connects the university administration with the teachers (specifically with visiting faculty) and the candidates and will help the admin to inform them about the Date sheet, Timetable, news and Notifications. This system will convert the present procedure of collecting feedback manually on the paper from the candidates about the teachers and their respective course to a fully automated system and thus provide the managing authority an ease to make decisions accordingly. The system also feature a reliable way for administration to generate reports of the profile of the teachers and the candidates in PDF format. The system will also have a web based panel for administration, two front end Android Application for teachers and candidates.

Table of Content

Dedication ................................................................................................................................ I

[ProjectInBrief II](#_Toc321643)

[Acknowledgement III](#_Toc321644)

[Declaration IV](#_Toc321645)

[Certification V](#_Toc321646)

[Abstract VI](#_Toc321647)

CHAPTER 1 .................................................................................................................................... 11

1.1 Introduction ............................................................................................................................... 11

1.1.1 Purpose ............................................................................................................... 11

1.1.2 Scope .................................................................................................................. 12

1.1.3 Business Opportunity ......................................................................................... 12

1.1.4 Problem Statement ............................................................................................. 12

1.1.5 System Position Statement ................................................................................. 13

1.2 Stakeholder and User Descriptions ........................................................................................... 14

1.2.1 Stakeholder Summary ........................................................................................ 14

1.2.2 User Summary ................................................................................................... 14

1.2.3 Stakeholders profile ........................................................................................... 15

1.2.4 Key Stakeholder / User Needs ........................................................................... 15

1.3 System Overview ...................................................................................................................... 16

1.4 System Perspective ................................................................................................ 16

1.5 Summary of Capabilities ....................................................................................... 16

1.6 Assumptions and Dependencies ............................................................................ 16

1.7 Cost and Pricing..................................................................................................... 17

1.8 Licensing and Installation ...................................................................................... 17

1.9 System Main Features ............................................................................................................... 17

1.10 System Priorities ....................................................................................................................... 17

1.11 Constraints ................................................................................................................................ 19

1.12 Quality Ranges .......................................................................................................................... 19

1.13 Other Item Requirements .......................................................................................................... 20

1.13.1 System Requirements......................................................................................... 20

1.13.1.1 Platform .......................................................................................................... 20

1.13.1.2 Printer ............................................................................................................. 20

1.13.1.3 Hardware requirements .................................................................................. 20

CHAPTER 2 .................................................................................................................................... 21

2.1 Introduction ............................................................................................................................... 21

2.1.1 Purpose ............................................................................................................... 21

2.2.1 Scope .................................................................................................................. 22

2.2.2 References .......................................................................................................... 23

2.2.3 Overview: ........................................................................................................... 23

2.2.4 System’s main features: ..................................................................................... 24

2.3 Functional Requirements .......................................................................................................... 24

2.3.1 Security Management ........................................................................................ 24

2.3.2 Course Management .......................................................................................... 24

2.3.3 Date Sheet Management .................................................................................... 26

2.3.4 Timetable Management ..................................................................................... 27

2.3.5 Attendance Management ................................................................................... 28

2.3.6 NCEAC Profile Management ............................................................................ 29

2.3.7 Feedback Management ...................................................................................... 30

2.3.8 News & Notification Management .................................................................... 30

2.4 Non-Functional Requirements .................................................................................................. 31

2.4.1 Security .............................................................................................................. 31

2.4.2 Usability ............................................................................................................. 31

2.4.3 Reliability ........................................................................................................... 31

2.4.4 Performance ....................................................................................................... 32

2.5 Design Constraints .................................................................................................................... 33

2.6 User Help/ manuals ................................................................................................................... 33

2.7 External Interface Requirements ............................................................................................... 33

2.7.1 User Interface ..................................................................................................... 33

2.7.2 Hardware Interface ............................................................................................. 33

2.7.3 Software Interface .............................................................................................. 33

2.8 Licensing requirements ............................................................................................................. 34 CHAPTER 3 .................................................................................................................................... 35

3.1 Use Cases Diagram ................................................................................................................... 35

Use case ID: 01 ................................................................................................................. 36

Use case ID: 02 ................................................................................................................. 37

Use case ID: 03 ................................................................................................................. 38

Use case ID: 04 ................................................................................................................. 39

Use case ID: 05 ................................................................................................................. 40

Use case ID: 06 ................................................................................................................. 41

Use case ID: 07 ................................................................................................................. 42

Use case ID: 08 ................................................................................................................. 43

Use case ID: 09 ................................................................................................................. 44

Use case ID: 10 ................................................................................................................. 45

Use case ID: 11 ................................................................................................................. 46

Use case ID: 12 ................................................................................................................. 47

Use case ID: 13 ................................................................................................................. 48

Use case ID: 14 ................................................................................................................. 49

Use case ID: 15 ................................................................................................................. 50

Use case ID: 16 ................................................................................................................. 51

Use case ID: 17 ................................................................................................................. 52

Use case ID: 18 ................................................................................................................. 53

Use case ID: 19 ................................................................................................................. 54

Use case ID: 20 ................................................................................................................. 55

Use case ID: 21 ................................................................................................................. 56

Use case ID: 22 ................................................................................................................. 57

Use case ID: 23 ................................................................................................................. 58

Use case ID: 24 ................................................................................................................. 59

Use case ID: 25 ................................................................................................................. 60

Use case ID: 26 ................................................................................................................. 61

Use case ID: 27 ................................................................................................................. 62

3.2 System Sequence Diagram........................................................................................................ 63

SSD: 01 Process Registration .......................................................................................................... 63

SSD: 02 Process Login .................................................................................................................... 63

SSD: 03 Add Course ........................................................................................................................ 64

SSD: 04 Search Course .................................................................................................................... 64

SSD: 05 Update Course ................................................................................................................... 65 SSD: 06 Delete Course .................................................................................................................... 65

SSD: 07 Generate Course Report ..................................................................................................... 66

SSD: 08 Add Date Sheet .................................................................................................................. 66

SSD: 09 Search Date Sheet .............................................................................................................. 67

SSD: 10 Update Date Sheet ............................................................................................................. 67

SSD: 11 Delete Date Sheet .............................................................................................................. 68

SSD: 12 Add Timetable ................................................................................................................... 68

SSD: 13 Search Timetable ............................................................................................................... 69

SSD: 14 Update Timetable .............................................................................................................. 69

SSD: 15 Add Attendance ................................................................................................................. 70

SSD: 16 Search Attendance ............................................................................................................. 70

SSD: 17 Update Attendance ............................................................................................................ 71

SSD: 18 Generate Attendance Report .............................................................................................. 71

SSD: 19 Add NCEAC Profile .......................................................................................................... 72

SSD: 20 Search NCEAC Profile ...................................................................................................... 72

SSD: 21 Update NCEAC Profile ..................................................................................................... 73

SSD: 22 Generate Report ................................................................................................................. 73

SSD: 23 Add Feedback .................................................................................................................... 74

SSD: 24 View Feedback .................................................................................................................. 74

SSD: 25 Add News and Notification ............................................................................................... 75

SSD: 26 Search News and Notification ........................................................................................... 75 SSD: 27 Update News and Notification........................................................................................... 76

3.3 Entity Relationship Diagram ..................................................................................................... 77

3.4 Normalization ........................................................................................................................... 78

3.4.1 Normalization for Table CANDIDATE: ................................................................ 78

3.4.2 Table PROVINCE: ............................................................................................ 79

3.4.3 Table DISTRICT: .............................................................................................. 79

3.4.4 Table CANDIDATE: .............................................................................................. 79

3.4.5 Normalization for Table TEACHER: ................................................................ 80

3.4.6 Normalization for Table COURSE: ................................................................... 81

3.4.7 Normalization for Table SMESTERSECTION: ................................................ 81

3.4.8 Normalization for Table CLASS: ...................................................................... 82

3.4.9 Normalization for Table TIMETABLE: ....................................................................... 82

3.4.10 Normalization for Table ATTENDANCE: ........................................................ 82

3.4.11 Normalization for Table DATESHEET: ........................................................... 83

3.4.12 Normalization for Table NEWSANDNOTIFICATION: .................................. 83

3.4.13 Normalization for Table FEEDBACK: ............................................................. 84

3.5 Relational Tables ...................................................................................................................... 84

3.5.1 Relational Table of CANDIDATE: ........................................................................ 84

3.5.2 Relational Table of PROVINCE:....................................................................... 85

3.5.3 Relational Table of DISTRICT:......................................................................... 85

3.5.4 Relational Table of TEACHER: ........................................................................ 85

3.5.5 Relational Table of COURSE: ........................................................................... 85

3.5.6 Relational Table of SMESTERSECTION: ........................................................ 86

3.5.7 Relational Table of CLASS: .............................................................................. 86

3.5.8 Relational Table of TIMETABLE: .................................................................... 86

3.5.9 Relational Table of ATTENDANCE: ................................................................ 86

3.5.10 Relational Table of DATESHEET: ................................................................... 87

3.5.11 Relational Table of NEWSANDNOTIFICATION: .......................................... 87

3.5.12 Relational Table of FEEDBACK: ..................................................................... 87

3.6 Domain Model .......................................................................................................................... 88

3.7 CLASS DIAGRAM .................................................................................................................. 89 Chapter 4 .......................................................................................................................................... 90

4.1 Introduction ............................................................................................................................... 90

4.1.1 Purpose ............................................................................................................... 90

4.1.2 Scope .................................................................................................................. 90

4.1.3 Definitions, Acronyms and Abbreviations ........................................................ 90

4.1.4 References .......................................................................................................... 91

4.1.5 Project Purpose, Scope and Objectives .............................................................. 91

4.1.6 Assumptions and Constraints ............................................................................. 91

4.1.7 Project Deliverables ........................................................................................... 91

4.1.8 Evolution of Software Development Plan ......................................................... 92

4.2 Project Organization ................................................................................................................. 92

4.2.1 Organizational Structure .................................................................................... 92

4.2.2 External Interfaces ............................................................................................. 92

4.2.3 Roles and Responsibilities ................................................................................. 93

4.3 Project Plan ............................................................................................................................... 94

4.3.1 Phase Plan .......................................................................................................... 94

4.3.2 Milestones .......................................................................................................... 94

4.3.3 Schedule Plan ..................................................................................................... 95

4.3.4 Project Resourcing ............................................................................................. 96

4.3.5 Project Monitoring and Control ......................................................................... 96

4.4 Technical Process Plan.............................................................................................................. 96

4.4.1 Methods, Tools and Techniques ........................................................................ 96

4.4.2 Infrastructure Plan .............................................................................................. 97

4.5 Supporting Process Plans .......................................................................................................... 97

4.5.1 Documentation Plan ........................................................................................... 97

4.5.2 Problem Resolution Plan.................................................................................... 97

4.5.3 Process Improvement Plan ................................................................................. 97

4.6 Work Breakdown Structure (WBS) .......................................................................................... 97 CHAPTER 5 .................................................................................................................................... 98

5.1 ASP.NET .................................................................................................................................. 99

5.1.1 Developer productivity ...................................................................................... 99

5.1.2 Improved performance and scalability ............................................................... 99

5.1.3 Enhanced reliability ........................................................................................... 99

5.1.4 Easy deployment ................................................................................................ 99

5.1.5 System requirements for visual studio. NET ................................................... 100

5.2 Windows XP ........................................................................................................................... 100

5.2.1 Easy to use ....................................................................................................... 100

5.2.2 Revolutionize the way remote users work ....................................................... 100

5.2.3 Advanced management, deployment and support tools .................................. 100

5.2.4 System requirements for windows 10 professional ......................................... 101

5.3 Microsoft SQL Server 2014 .................................................................................................... 101

5.3.1 Standard features .............................................................................................. 101

5.3.2 System requirements for Microsoft SQL Server 2014 .................................... 101

Chapter 6 ........................................................................................................................................ 103

1. CONCLUSIONS & FUTURE WORK ................................................................................... 128
   1. Achievements .......................................................................................................................... 128
   2. Limitations .............................................................................................................................. 128

Chapter 8 ........................................................................................................................................ 129

8.1 User Guide ............................................................................................................................... 129

* + 1. Computer Requirements ........................................................................................ 129
    2. Home Page ............................................................................................................. 129
    3. Login ...................................................................................................................... 136
    4. Dashboard .............................................................................................................. 136
    5. Add Course ............................................................................................................ 137
    6. View Course .......................................................................................................... 137
    7. Add News and Notifications .................................................................................. 138
    8. Manage News and Notifications ............................................................................ 138
    9. Add Slides.............................................................................................................. 139
    10. Manage Slides...................................................................................................... 139
    11. Add Upcoming Events......................................................................................... 140
    12. Manage Upcoming Events ................................................................................... 140
    13. Add Testimonials ................................................................................................. 141
    14. Manage Testimonials ........................................................................................... 141
    15. Add Teams ........................................................................................................... 142
    16. Manage Teams ..................................................................................................... 142
    17. Register User ....................................................................................................... 143
    18. View User ............................................................................................................ 143
    19. Update Semester .................................................................................................. 145
    20. Alumni Candidate .................................................................................................... 145
    21. Assign Courses .................................................................................................... 146
    22. Class List ............................................................................................................. 146
    23. Feedback Reports ................................................................................................ 147
    24. Feedback Reports Example ................................................................................. 147
    25. Android Splash Screen ........................................................................................ 148
    26. Android login Screen ........................................................................................... 149
    27. Android Dashboard Entrance .............................................................................. 150
    28. Android Menu...................................................................................................... 151
    29. Android Slides ..................................................................................................... 152
    30. Android Events .................................................................................................... 153
    31. Android News ...................................................................................................... 155
    32. Android Profile .................................................................................................... 156
    33. Android Course Details ....................................................................................... 157
    34. Android Courses .................................................................................................. 158
    35. Android Assigned Teachers................................................................................. 159
    36. Android Feedback Screen .................................................................................... 160
    37. Android Setting.................................................................................................... 161
    38. Android Setting.................................................................................................... 162

Chapter 9 ........................................................................................................................................ 163

* 1. References ................................................................................................................................ 163
  2. Biblography ............................................................................................................................. 163

### List of Figures

3.1 Use Case Diagram ..................................................................................................................... 27

3.2 Process Registration .................................................................................................................. 55

3.3 Process Login ............................................................................................................................ 55

3.4 Add Course ............................................................................................................................... 56

3.5 Update Course ........................................................................................................................... 56

3.6 Serach Course ........................................................................................................................... 57

3.7 Delete Course ............................................................................................................................ 57

3.8 Generate Course Report ............................................................................................................ 58

3.9 Add Date Sheet ......................................................................................................................... 58

3.10Serach Date Sheet .................................................................................................................... 59

3.11Update Date Sheet ..................................................................................................................... 59

3.12Delete Date Sheet ...................................................................................................................... 60

3.13Add Timetable........................................................................................................................... 60

3.14 Search Timetable ...................................................................................................................... 61

3.15Update Timetable ...................................................................................................................... 61

3.16Add Attendance......................................................................................................................... 62

3.17Serach Attendance ..................................................................................................................... 62

3.18Update Attendance .................................................................................................................... 63

3.19Generate Attendance ................................................................................................................. 63

3.20Add NCEAC Profile ................................................................................................................. 64

3.21Search NCEAC Profile ............................................................................................................. 64

3.22Update NCEAC Profile ............................................................................................................. 65

3.23Generate Profiles Report ........................................................................................................... 65

3.24Add Feedback ........................................................................................................................... 66

3.25View Feedback .......................................................................................................................... 66

3.26Add News and Notification ....................................................................................................... 67

3.27Search News and Notification ................................................................................................... 67

3.28Update News and Notification .................................................................................................. 68

3.29ER Diagram............................................................................................................................... 69

3.30Domain Model .......................................................................................................................... 81

3.31Class Diagram ........................................................................................................................... 82

4.1 Organizational Structure ........................................................................................................... 85

4.2 Work Breakdown Structure .................................................................................................. 92

8.1 Shows the appraisal home page ........................................................................................... 138

8.2 Shows the about section ........................................................................................................... 139

8.3 Shows the testimonials of members .......................................................................................... 85

8.4 Shows the latest news and notification ..................................................................................... 92

**CHAPTER 1**

**INTRODUCTION**

### 1.1 Introduction

As the world is facing global pandemic crises, the education system is turning toward new technologies and adopting e-learning platforms. The education system is teeming with technological breakthroughs to keep up with the changing dynamics of e-learning, and one such change is the online web proctored exam. Online proctoring refers to form of assessment which enables candidates to give exams from any location. So to keeping the above in view, we are developing an Artificial Intelligence based exam proctoring system.

#### 1.1.1 Purpose

The purpose of this System is to collect, analyze, and perform relevant and necessary actions immediately without any delay to enhance the overall working performance of the department. It focuses on the functionality needed by the stakeholders these stakeholders are the HOD, Teacher and the Candidates. The previous system in use is totally manual and it is hard manage keep track of data and to perform simple tasks it requires an immense amount of time and energy. The Computer Science department wants to find a solution to reduce its time delays. The major goal of the system is to collect necessary information of the candidates and the teachers required for NCEAC accreditation purposes and to provide them with latest news and notifications regarding timetable, date sheets and many more.

#### 1.1.2 Scope

In our CSDAS (Computer Science Department Automation System) the old and manual system is transitioned to a new computerized system. Different modules will be constructed based on the requirements gathered from the stakeholder and the environment. Main entities like Teacher and Candidate will have separate modules and will be managed by the Admin i.e. Administrator/HOD (Head of Department). The system will allow to manage the official task very efficiently which will increase the overall performance of the department.

#### 1.1.3 Business Opportunity

The previous system that is in use is totally manual and it is hard to use and manage. To keep track of simple tasks it requires an immense amount of time and energy. So, to fulfill the requirements of the stakeholders we are providing a digital solution for managing the official task and this system also provide a better environment for the administration, teacher and the candidate to communicate and share information so easily to save and human resources.

#### 1.1.4 Problem Statement

|  |  |
| --- | --- |
| **The problem** | in pandemic crisis it is problematic to teach online for many teachers, and is difficult to check on candidates’ academic situation where they use ordinary systems to take exams  In the absence of anti-cheating measures to securely conduct the test, the factors, abovementioned, compromise the credibility of remote online assessments forcing organizations to invite the candidates to the venue so that they can take the test in the presence of an acting invigilator. The role of the invigilator is usually performed by the HR or hiring manager who keeps a strict vigil on the candidate’s activities.  Calling the candidates several times to the office venue defeats the purpose of online assessments. New technology has been buzzing in the market that can help you conduct credible remote online assessments safely and securely from the comfort of your location. It is called Remote (Virtual) Proctoring. |
| **Effects** | From the present manual system stakeholders are effected in many ways such as:   * During exams someone else taking the exam in place of the actual candidate . * Searching for answers online using software to generate solution * Searching for answers on the mobile/calling someone to ask for answers * Getting another person in the room to help in answering the questions * Have a friend simultaneously access the exam questions and providing solutions * Leaving the screen or getting up in the middle of the test to use unfair means. |
| **The impact of which is** | The above problem will bi more impactful for candidate. The above problem may help to achieve good grades but candidate have rick of his career in the field in which candidate effort to make his career. |
| **A successful solution would be** | A system which provides simplicity and efficiency in evaluate accordifg to university`s standers. |

**1.1.5 System Position Statement**

|  |  |
| --- | --- |
| **For** | Federal Urdu University of Arts Science  & Technology |
| **Who** | The existing environment has no such system being used, which provides effective evaluation of candidate. |
| **The (Product name)** | AI Based Online Exam Proctoring System (AIBOEPS) |
| **That** | Keeps the record of information of candidate behavior during the exam and also keep the record of candidate result. It will also provide ease in prevention of cheating in exam and will be helpful for candidate and as well as university |
| **Unlike** | Simple online exam system, that is not suitable for conducting exam and the candidate academic progress. |
| **Our Product** | Uses new technologies for development. This system is going to be built by keeping in mind the latest requirements. By providing better performance, security, reliability and scalability so that the system runs efficient and fast. |

**1.2 Stakeholder and User Descriptions**

#### 1.2.1 Stakeholder Summary

|  |  |  |
| --- | --- | --- |
| **Name** | **Represents** | **Role** |
| HOD (Primary Actor) | It will provide all requirements that are required to complete the product | Ensure that the system will have all records. Also add, update & search data. |

#### 1.2.2 User Summary

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | |
| **Administrator** | | It will interact with system after logged in he can add, update, search & delete the data from the system and can also generate reports. |
| **Teacher (Primary**  **Actor)** | | It will interact with system after logged in he can Create Exam, update Exam, search Exam and Maintain candidates result. |
| **Candidate (Primary**  **Actor)** | | It will be able to interact with system after face recognition and then able to give exam, view result and view behavior in exam. |
| **Proctor(Primary**  **Actor)** | | It will interact with system after logged in he can view behavior of each candidate & update candidate behavior report. |

#### 1.2.3 Stakeholders profile

##### 1.2.3.1 Administrator

|  |  |
| --- | --- |
| **Representative** | Not defined |
| **Type** | Primary Actor |
| **Description** | After logged in he can add, update, search & delete the data from the system and can also generate reports. |
| **Responsibilities** | Enter correct entries and also manage role assignments correctly |
| **Success Criteria** | The Administrator is the stakeholder who will define the success if he will totally satisfy with the system |
| **Involvement** | The administrator is the stakeholder who is involved in the project by entering correct information in the system and manage all records of system |
| **Deliverables** | There will be no deliverables provided by this stakeholder |
| **Comments Issues** | With respect to this stakeholder this project is reliable that it should always deliver the correct and timely Information |

#### 1.2.4 Key Stakeholder / User Needs

Existing manages records manually and can’t be searched and updated efficiently as required. This system will provide ease in managing the records and other official task for the administration and will provide easy control over the gathered data for the generation of reports and searching etc.

#### 1.3 System Overview

This system is being made to replace the old online exam system. The main idea is to have easy access to the records of candidate behavior during exam , candidate results and managing them efficiently. This alternate to the manual system will provide the main feature of AI proctoring and result management records. The users (teachers, candidate and administrator, proctor) will have their own accounts, separate according to user category, where they will be able to enter records according to privileges.

#### 1.4 System Perspective

This AI Based Online Exam Proctoring System (AIBOEPS) is a completely new system for the candidate, teachers, proctor and administration of the FUUAST ISB. It has the database of teacher’s details, candidate’s details, and candidate’s behavior details etc.

#### 1.5 Summary of Capabilities

|  |  |
| --- | --- |
| **Benefit** | **Supporting Features** |
| Administrator would manage the system easily as compared to existing manual system | Computerized system will information manage using computers makes management easier |
| User would manage the system easily as compared to existing manual system | Computerized system plus digital information management using computers makes management easier |
| System can generate different types of reports according to given criteria | Computerized system settings will always makes the business running efficiently |

#### 1.6 Assumptions and Dependencies

The default language of the Computer Science Department Automation System is English so the user of the system should be able to read, write and perceive the proper meaning.

On the developing side the implementers should be familiar with the programming languages used.

**1.7 Cost and Pricing**

This is our Final Year Project & have no Cost & Pricing.

#### 1.8 Licensing and Installation

All the installation rights will be mentioned in the license agreement. Installation of the system will take place after complete development.

#### 1.9 System Main Features

* Registration/Login
* Course
* Date sheet
* Time Table
* Attendance
* NCEAC Registration Profile
* Feedback
* News and Notification

### 1.10 System Priorities

**1.10.1 Security Management**

Process Login

**1.10.2 Course Managements**

#### 1.11 Constraints

The system is only compatible for the Windows Platform. Since we are using powerful tools i.e. Visual Studio 2017 with C# and MS SQL Server therefore the system on which this whole system will be installed must be at least single core 2 GHZ. The system also includes Android applications which will be developed using android studio for which it minimum requires API 15: Android 4.0.3 (Ice Cream Sandwich).

**1.12 Quality Ranges**

###  Accuracy

Our system will meet the specified requirements. The result generated will be consistent and according to the requirements gathered by the stakeholders.

###  Performance

The AI Online Proctoring System will have a simple GUI, so that the users of the system will not have any problem in handling the system. The overall experience of the usage will be satisfactory. Standard queries will take no longer than 2-5 second.

###  Security

Our AI Online Proctoring System will provide better security by authenticating users and giving them access to the system according to their privileges.

Without proper authentication no one can access the system.

###  Usability

The system will be easy to use. The AIBOEPS has a simple GUI, which is menu based and has all of the components arranged logically so that the users of the system will not have any problem in handling the system. The overall experience of the usage will be satisfactory.

###  Availability

As the system is a mobile and a Web application it will run for as long as the user wants it to run

#### 1.13 Other Item Requirements

##### 1.13.1 System Requirements

**1.13.1.1 Platform**

The system must use windows platform.

**1.13.1.2 Printer**

Printer is used to print the required form and reports (if required)

**1.13.1.3 Hardware requirements**

Pentium IV or above

**CHAPTER 2**

**REQUIREMENT ANALYSIS**

### AI Based Online Exam Proctoring System (AIBOEPS)

#### 2.1 Introduction

The Computer Science (CS) department automation system of Federal Urdu University of Arts science and technology refers to the varied computer machinery and software used to digitally create, collect, store, manipulate, and relay information needed for accomplishing basic tasks like collecting Data of candidates and teachers required for National Computing Educational Accreditation Council (NCEAC) accreditation. This system connects the university administration with the teachers (specifically with visiting faculty) and candidates and will help the admin to inform them about the Data sheet, Timetable, News and Notifications. This system will convert the present procedure of collecting feedback manually on the paper from the candidates about the teachers and their respective course to a fully automated system and thus provide the managing authority an ease to make decisions accordingly. The system also features a reliable way for administration to generate reports of the profile of the teachers and the candidates in PDF/MS WORD format. The system will also have a web based panel for administration and two front end Android Application for the teachers and the candidates.

##### 2.1.1 Purpose

The purpose of the proposed system is to collect, analyze, and perform relevant and necessary actions immediately without any delay to enhance the overall working performance of the department. It focuses on the functionality needed by the stakeholders these stakeholders are the HoD, Teacher and the Candidates. The current system in use is totally manual and it is hard to manage keep track of data and to perform simple tasks it requires an immense amount of time and energy. The Computer Science department wants to find a solution to reduce its time delays. The major goal of the system is to collect necessary information of the candidates and the teachers required for NCEAC accreditation purposes and to provide them with latest news and notifications regarding timetable, date sheets and many more.

##### 2.2.1 Scope

In our AI Based Online Exam Proctoring System (AIBOEPS)the old and manual system is transitioned to a new computerized system. Different modules will be constructed based on the requirements gathered from the stakeholder and the environment. Main entities like Teacher and Candidate will have separate modules and will be managed by the Admin i.e. Administrator/HoD (Head of Department). The system will allow to manage the official task very efficiently which will increase the overall performance of the department.

**The System has various modules such as:**

* Security Management
* Course Management
* Date sheet Management
* Timetable Management
* Attendance
* NCEAC Registration Profile
* Feedback
* News and Notifications
* **Security Management Module** will provide the access to the system according to the privileges of the user. This module will also provide the user to register into the system and access the functionality after authentication. This will used to authenticate the user accessing the system.
* **Course Management Module** is keeping the record of the Course Contents, Course details, Course codes etc. The purpose of this module will be to provide candidates and teacher with the contents followed by the university regarding their courses and later at the end collect feedbacks about these courses.
* **Date Sheet Management Module** will help the candidates and the teacher to download the date sheets of the final and midterm examination and it will help the Admin to generate the Date sheet easily.
* **Timetable Management Module** will help the teacher and candidates to view or download the timetable and it also help the administration to generate the timetable easily.
* **Attendance Management Module** will help the teachers to add total attendance of the candidates at the end of the semester and then admin can easily generate reports of the short attendance candidates in PDF/MS WORD Format which will be available to both the teacher and candidates on their android application under news and notifications tab.
* **NCEAC Profile Management Module** will allow candidates and the teachers to maintain their profiles with the data required for NCEAC registration. It will help the admin to process this data or generate report as required.
* **Feedback Management Module** will allow the candidates to give feedbacks about their courses and the teachers and it will also help the candidates to request for a teacher to be assigned to them in the next semester. The admin will be able to easily assess the performance of the teachers in their respective subjects.

|  |  |
| --- | --- |
| **Term/Abbreviation** | **Definition** |
| **CSDAS** | Computer Science Department Automation System |
| **GUI** | Graphical User Interface |
| **RUP** | Rational Unified Process |
| **C#** | Programming language |
| **Visio** | Designing tool |
| **NCEAC** | National Computing Educational Accreditation Council |
| **DEO** | Data Entry Operator |
| **UML** | Unified modeling language |

* **News and Notification Management Module** will help teacher and candidates to view latest news, policy change and other notifications and it will help admin to quickly dispatch information to the faculty without any delay.

**2.2.2 References**

### Websites

* [www.msdn.microsoft.com](http://www.msdn.microsoft.com/)
* [www.stackoverflow.com](http://www.stackoverflow.com/)
* [www.pluralsight.com](http://www.pluralsight.com/)
* [www.ibm.com](http://www.ibm.com/)

**Books:**

 Software Requirements - Techniques By Soren Lauesen  Applying UML and Patterns by Craig Larman 2nd Edition

**2.2.3 Overview:**

This document contains the requirements of the Computer Science Department Automation System. All the functional Requirements are collected from the administration of the department. Head of Department of Computer Science provided the requirements and functionalities by himself.

**2.2.4 System’s main features:**

* Security Management
* Course Management
* Date sheet Management
* Timetable Management
* Attendance Management
* NCEAC Registration Profile
* Feedback
* News and Notification

#### 2.3 Functional Requirements

##### 2.3.1 Security Management

###### 2.3.1.1 Process Registration

|  |  |
| --- | --- |
| **SRS-1** | Admin will register candidates and will provide them with their email and password. |
| **SRS-2** | Admin will register candidate by adding their CNIC email contact and a basic password. |

###### 2.3.1.2 Process Login

|  |  |
| --- | --- |
| **SRS-3** | Users can enter the system by entering the username and password |
| **SRS-4** | The user name should be Unique |
| **SRS-5** | Authentication should be provided and the suitable access of the system should be given according to privileges |

##### 2.3.2 Course Management

###### 2.3.2.1 Add new Course

|  |  |
| --- | --- |
| **SRS-6** | Administrator can add a new course into the system |
| **SRS-7** | Each entry will have information about course code, course credit hours, pre-requisite of course, course recommended books, course name/title, course goal, course contents, course categoryetc. |
| **SRS-8** | The course information should be proper and legal |
| **SRS-9** | System should save this information |

###### 2.3.2.2 Search course

|  |  |
| --- | --- |
| **SRS-10** | Administrator, Teacher and Candidates can search for course information |
| **SRS-11** | System should give facility of searching course information on the basis of course name, course code and semester wise. |
| **SRS-12** | System should find the result based on search criteria |
| **SRS-13** | System should display search information |

###### 2.3.2.3 Update Course Information

|  |  |
| --- | --- |
| **SRS-14** | The course to be updated will be searched first by the administrator |
| **SRS-15** | The information of the course being updated should be according to the standards/parameters |
| **SRS-16** | Administrator should be able to update the course information |
| **SRS-17** | System should update this information |

###### 2.3.2.4 Delete Course

|  |  |
| --- | --- |
| **SRS-18** | The course to be deleted will be searched first by Administrator |

|  |  |
| --- | --- |
| **SRS-19** | Administrator should be able to delete the course information from the system |

###### 2.3.2.5 Generate Course Reports

|  |  |
| --- | --- |
| **SRS-20** | Teacher and candidates should be able to generate report of the course information and download course contents. |
| **SRS-21** | The generated report should be in PDF/MS WORD Format and well formatted to take print. |

##### 2.3.3 Date Sheet Management

###### 2.3.3.1 Add Date Sheet

|  |  |
| --- | --- |
| **SRS-22** | Date Sheet of the final and midterm should be added by the administrator |
| **SRS-23** | Date Sheet should be added in the PDF/MS WORD format and users will be able to download it |
| **SRS-24** | System should upload this information |

###### 2.3.3.2 Search Date Sheet

|  |  |
| --- | --- |
| **SRS-25** | The date sheet to be viewed will be searched first |
| **SRS-26** | System should give facility of searching date sheet in the system |
| **SRS-27** | Teacher and candidates can search and view date sheet |
| **SRS-28** | System should display search information |

###### 2.3.3.3 Update Date Sheet

|  |  |
| --- | --- |
| **SRS-29** | Updating date sheet should be according to the parameter semester season i-e Fall 2018, Spring 2019 and final term or midterm. It should be searched first |
| **SRS-30** | Administrator should be able to update the Date sheet |
| **SRS-31** | System should update this information |

###### 2.3.3.4 Delete Date Sheet

|  |  |
| --- | --- |
| **SRS-32** | Administrator should be able to delete the information |

##### 2.3.4 Timetable Management

###### 2.3.4.1 Add Timetable

|  |  |
| --- | --- |
| **SRS-34** | Administrator can add the timetable in the system |
| **SRS-35** | Timetable should be in the PDF/MS WORD format so that it can be viewed and downloaded by the users and it should be well formatted. |
| **SRS-36** | System should save this information |

###### 2.3.4.2 Search Timetable

|  |  |
| --- | --- |
| **SRS-37** | System should give facility of searching timetable |
| **SRS-38** | Teacher and Candidates can search for timetable |
| **SRS-39** | System should display search information |

###### 2.3.4.3 Update Timetable

|  |  |
| --- | --- |
| **SRS-40** | System should give facility of searching timetable information on the basis of semester detail that is Spring 2019 etc. |
| **SRS-41** | The updating of the Timetable should be according to date |
| **SRS-42** | Administrator should be able to update the timetable |
| **SRS-43** | System should update this information |

##### 2.3.5 Attendance Management

###### 2.3.5.1 Add Attendance

|  |  |
| --- | --- |
| **SRS-45** | Regular attendance can be added into the system |
| **SRS-46** | Semester attendance of each course of candidates will be added into the system date wise with class time |
| **SRS-47** | System should save this information |

###### 2.3.5.2 Search Attendance

|  |  |
| --- | --- |
| **SRS-48** | System should give the authority of searching the attendance of the candidates on the basis of its name or ID or course name. |
| **SRS-49** | Administrator and Teacher can search for attendance |
| **SRS-50** | System should display search information |

###### 2.3.5.3 Update Attendance

|  |  |
| --- | --- |
| **SRS-51** | System should give facility of searching attendance information on the basis of candidates name or MIS ID. |
| **SRS-52** | The updating of the attendance should be according to MIS ID. |
| **SRS-53** | Administrator and teacher should be able to update the attendance information |
| **SRS-54** | System should update this information |

###### 2.3.5.4 Generate attendance Report

|  |  |
| --- | --- |
| **SRS-55** | Admin should be able to generate the reports of the candidates both fulfilling the attendance barrier and short attendance candidates in  PDF/MS WORD/MS word format. The generation of reports should be semester wise or overall. |

##### 2.3.6 NCEAC Profile Management

###### 2.3.6.1 Add NCEAC profile

|  |  |
| --- | --- |
| **SRS-56** | Teacher and candidates can maintain their profile after logged into the system according to the NCEAC parameters. |
| **SRS-57** | Candidates should maintain their profile by adding following details candidate name, father name, gender, Roll No (Enrollment Number), admission date, nationality, CNIC number, passport number (optional), date of birth, phone number, email, domicile district, domicile province, mailing address, city (mailing address), SSC degree name, SSC board name, SSC total marks, SSC obtained marks, HSSC degree name, HSSC board name, HSSC total marks, HSSC obtained marks. Teacher will maintain their profile by adding parameters such as name, gender, CNIC, mobile number, email, address, province, city, employment detail, contracts detail, academic rank, joining date, status (active or not), leaving date, academic qualification, degree name, degree type, field of study, core computing teacher (yes or no), degree awarding country, university name, degree start date and degree end date. |
| **SRS-58** | System should save this information |

###### 2.3.6.2 Search NCEAC Profile

|  |  |
| --- | --- |
| **SRS-59** | System should provide the facility of searching the profile of the candidates and teacher with their name or MIS ID for candidates and NIC for teachers. |
| **SRS-60** | Administrator will be able to search the profiles data |
| **SRS-61** | System should display search information |

###### 2.3.6.3 Update NCEAC Profile

|  |  |
| --- | --- |
| **SRS-62** | Candidates and teachers will be able to update the information in their respective profiles |
| **SRS-63** | The updating of the attendance should be according to MIS ID for candidates and CNIC for teachers. |
| **SRS-64** | System should update this information |

###### 2.3.6.4 Generate Report

|  |  |
| --- | --- |
| **SRS-65** | Administrator will be able to generate reports of the candidate’s data batch wise and also admin can generate report of teacher’s data for accreditation purposes in PDF/MS WORD format |

##### 2.3.7 Feedback Management

###### 2.3.7.1 Add Feedback

|  |  |
| --- | --- |
| **SRS-66** | Candidates can add feedback about the course and the teacher and can send request to the admin |
| **SRS-67** | Feedback should be added according to the course and its teachers |
| **SRS-68** | System should save this information |

###### 2.3.7.2 View Feedback

|  |  |
| --- | --- |
| **SRS-69** | Admin can view the feedbacks of the candidates according to the course and semester |
| **SRS-70** | Admin can search the feedbacks of specific courses and of specific semesters |

##### 2.3.8 News & Notification Management

###### 2.3.8.1 Add News and Notifications

|  |  |
| --- | --- |
| **SRS-71** | News and notification should be added by the administrator each news and notification will have title, description, dates and attached files |
| **SRS-72** | News and notification should be added by the administrator for candidates and teachers separately or for both at once |
| **SRS-73** | System should save this information |

###### 2.3.8.2 Search News and Notifications

|  |  |
| --- | --- |
| **SRS-74** | The news and notification will be viewed and searched by admin, teachers and the candidates |
| **SRS-75** | System should give facility of searching the uploaded news and notifications in the system by titles or dates |
| **SRS-76** | System should display search information |

###### 2.3.8.3 Update News and Notifications

|  |  |
| --- | --- |
| **SRS-77** | The news and notification need to be updated should be searched first |
| **SRS-78** | Administrator should be able to update the news and notifications |
| **SRS-79** | System should update this information |

#### 2.4 Non-Functional Requirements

##### 2.4.1 Security

The system requires the users to identify themselves by using login-id and password. Any user who uses the system shall have a login-id and a password

##### 2.4.2 Usability

With proper guidance and use of manual the required training time for a normal users will not be more than one week the interfaces of the system will also guide the user and hopefully the user may not need to read the manuals in order to use this system

**2.4.3 Reliability**

**Availability** The system facility will be available to its users for 24 hours

### Bugs or defect rate Very little

|  |  |
| --- | --- |
| **Robustness** | generate appropriate exceptions and will handle those exceptions effectively |
| **Accuracy**  **Correctness** | The system will generate accurate results and will behave according to the User Requirements  Our system will be according to all the requirements collected. |

In order to make this system fault tolerant this system will

All the requirements will be meeting the User Requirements

#### 2.4.4 Performance

* The CSDAS has a simple GUI, which is self-explanatory and user friendly
* The lists, reports and results generated by the system will not take more than 2-3 sec
* The system will provide correct error detection if some user enters wrong input or login credentials
* The system will run smoothly on minimum systems requirements without any problem
* Responses to queries shall take no longer than 4-5 sec to load onto the screen after the user submits the query
* The CSDAS shall display confirmation messages to user within 1-2 sec after the user submits information to the system. In other words robustness is maintained
* The retrieval of information from the database will not take long time and in case of wrong input the system will prompt a proper message to guide the user

about the error.

* Resources Required

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum** | **Recommended** |
| Processor | 1 Core 2 GHz for PC  For Android application  Android 4.0.3 | Dual Core 2+ GHz  For Android application  Android Version 6 |

RAM 512 MB for PC

1GB for Android

1 GB for PC

2GB for Android

|  |  |  |
| --- | --- | --- |
| Hard-drive | 20 GB for PC  30MB for android | As Required |
| Printer | - | - |

#### 2.5 Design Constraints

Rational Unified Process model (RUP) will be used for the development of the project as it is an iterative model and most widely used commercial model and for designing purposes UML will be used to make diagrams using MS Visio tool. C# .Net language will be used for the development of the server side script while for the front and HTML, JavaScript, Bootstrap, JQuery and Ajax will be used. For the development of the database Microsoft SQL server will be used. For teacher and candidates Android applications will be used which will be developed using Java. So, we must be familiar with these constraints.

#### 2.6 User Help/ manuals

Detailed System manuals are given to the system users and one week training sessions will be held to ensure that the users of the system understands the system fully and can get the maximum benefit from the system

#### 2.7 External Interface Requirements

##### 2.7.1 User Interface

The user interface will be friendly and will be developed by undertaking user intentions and interest. The software will be interactive and self-explanatory with proper

HCI implemented

**2.7.2 Hardware Interface**

The system will interact with the printer if the user wants to print the reports.

##### 2.7.3 Software Interface

|  |  |  |
| --- | --- | --- |
| **Software** | **USE** | **Version** |
| Windows | Operating System | XP,7,8,10 |
| MS Visual Studio (C#) | Front end | 2017 |
| Android Studio | Front end | 2015+ |
| MS SQL Server | Back end | Latest is Better |
| Microsoft Visio | For Modeling | 2010+ |

#### 2.8 Licensing requirements

This system will be developed for the CS department of the Federal Urdu University of Arts Science and Technology and it can only be used in this specific department.

**CHAPTER 3**

### SOFTWARE DESIGN

#### 3.1 Use Cases Diagram



**Figure 3.1: Use Case Diagram**

|  |  |
| --- | --- |
| **Use case ID:** | UC-01 |
| **Use case Name:** | Process Registration |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 30th June,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 2nd July,2019 |
| **Use case Description:** | This use case describes how an Admin will register Teacher, Candidate, himself/herself into a system |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully register accounts of teacher, candidate and himself/herself into the system |
| **Pre-Condition:** | 1. System must be in running state 2. Admin must have the Name, CNIC, and other basic information of teacher, candidate and himself/herself to register them into the system |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Admin must first login into the system using ID and Password |  |  |
|  |  | 2 | System will authenticate the user information from database |
| 3 | Admin will create account of the teacher by entering (Name, CNIC, Courses…) and candidate by entering (Name, CNIC, Semester…) with basic password which user can change later. |  |  |
|  |  | 4 | System will register user and confirmation message will be shown |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully into the system |
| **Extension Points:** | **1a- If user enters the “Wrong ID” or “Wrong Password”**  **-**System prompt user to enter correct ID or Correct Password  **3a-If admin leave any field empty while performing registration**  **-**system prompt user to enter the required field |
| **Priority:** | Low |
| **Frequency:** | Low Frequency of use |
| **Cross-Reference:** | SRS1, SRS2 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-02 |
| **Use case Name:** | Process Login |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 30th June,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 2nd July,2019 |
| **Use case Description:** | This use case describes how a user will Login into a system |
| **Primary Actors:** | <Administrator><Teacher><Candidate> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully log into the system  **Teacher:** Wants to successfully log into the system  **Candidate:** Wants to successfully log into the system |
| **Pre-Condition:** | 1. System must be in running state 2. User must have Login ID/Password |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User will enter the ID & Password |  |  |
|  |  | 2 | System will authenticate the user information from database |
|  |  | 3 | System process information and Confirmation message is shown |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully logs into the system |
| **Extension Points:** | **1a- If user enters the “Wrong ID”**  **-**System prompt user to enter correct ID **1b- If user enters the “Wrong Password”**  **-**System prompt user to enter correct Password **1c- If user leave any blank field**  **-**System will prompt to “Fill the blank fields” |
| **Priority:** | High |
| **Frequency:** | High Frequency of use |
| **Cross-Reference:** | SRS3, SRS4, SRS5 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-03 |
| **Use case Name:** | Add Course |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Add course details into the System |
| **Primary Actors:** | <Admin> |
| **Stake Holders & Interest:** | **Admin:** Wants to successfully Add new Course details |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

3. User must have right to add new course

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add new Course |  |  |
|  |  | 2 | System will show form to the user to entry new course details |
| 3 | User will enter the all information of a new course |  |  |
|  |  | 4 | System will add the information of new course successfully |
| 5 | Step 2-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Cross-Reference:** | SRS6, SRS7, SRS8, SRS9 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-04 |
| **Use case Name:** | Search Course |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Search course details |
| **Primary Actors:** | <Administrator><Teacher><Candidate> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Search course  **Teacher:** Wants to successfully Search an item  **Candidate:** Wants to successfully Search an item |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

3. Course must be present in data base before seaching

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Course |  |  |
|  |  | 2 | System Process and show the Search course form to the User |
| 3 | User will enter the course information |  |  |
|  |  | 4 | System will verify the information and course information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Search an course |
| **Extension Points:** | **1a- User search for an course which doesn’t exist**  -System will prompt to “Course Not Found” **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | High Frequent in use |

SRS10, SRS11, SRS12, SRS13

3. course must be present in data base before updation

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for course record |  |  |
|  |  | 2 | System process and course information is shown to user |
| 3 | User will update the course information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Use case ID:** | UC-05 |
| **Use case Name:** | Update Course |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Update course information |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Update course information |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully update an course information |
| **Extension Points:** | **1a- User search for an course which doesn’t exist**  -System will prompt to “Course Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |

SRS14, SRS15, SRS16, SRS17

3. course must be present in data base before deletion

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for course record |  |  |
|  |  | 2 | System process and course information is shown to user |
| 3 | User will delete the course information |  |  |
|  |  | 4 | System will process and delete the information from the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Use case ID:** | UC-06 |
| **Use case Name:** | Delete Course |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Delete course information |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Delete course information |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully delete an course information |
| **Extension Points:** | **1a- User search for an course which doesn’t exist**  -System will prompt to “Course Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |

SRS18, SRS19

UC-07

|  |  |
| --- | --- |
| **Use case Name:** | Generate Course Report |
| **Use case Prepared by:** | Badar Khalil |
| **Use case updated by:** | Sana Aftab Abbasi |
| **Use case Description:** | This use case describes how a user will Generates Report of course information |
| **Primary Actors:** | <Teacher><Candidate> |
| **Stake Holders & Interest:** | **Teacher:** Wants to successfully Generate a Report of course information **Candidate:** Wants to successfully Generate a Report of course information |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have rights to Generate Reports |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User will select View Course Information |  |  |
|  |  | 2 | System Process and ask for Parameters |
| 3 | User will selects the Parameters and submits |  |  |
|  |  | 4 | System process and Shows course report and contents |
| 5 | User will be able to download this information in PDF/MS Word format |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Generate course Report |
| **Extension Points:** | **1a- if a User doesn’t selects the Parameters**  -System will prompt to “selects the Parameters” |
| **Priority:** | High |
| **Frequency:** | Low Frequent in use |
| **Cross-Reference:** | SRS20,SRS21 |

UC-08

|  |  |
| --- | --- |
| **Use case Name:** | Add Date Sheet |
| **Use case Prepared by:** | Badar Khalil |
| **Use case updated by:** | Sana Aftab Abbasi |
| **Use case Description:** | This use case describes how a user will Add Date sheet |
| **Primary Actors:** | <Admin> |
| **Stake Holders & Interest:** | **Admin:** Wants to successfully Add Date Sheet |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

3. User must have right to add Date Sheet

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add Date Sheet |  |  |
|  |  | 2 | System will show form to the user to add date sheet information or upload file |
| 3 | User will enter the all information required to add date sheet |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |

SRS22, SRS23, SRS24

UC-09

|  |  |
| --- | --- |
| **Use case Name:** | Search Date Sheet |
| **Use case Prepared by:** | Badar Khalil |
| **Use case updated by:** | Sana Aftab Abbasi |
| **Use case Description:** | This use case describes how a user will Search the Date sheet |
| **Primary Actors:** | <Administrator><Teacher><Candidate> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Search the Date Sheet  **Teacher:** Wants to successfully Search the date sheet  **Candidate:** Wants to successfully Search the date sheet |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Date sheet must be uploaded in data base before |

searching **Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for date sheet |  |  |
|  |  | 2 | System Process and shows the Search date sheet form to the User |
| 3 | User will enter the Date sheet details to be searched |  |  |
|  |  | 4 | System will verify the information and date sheet will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Search Date sheet |
| **Extension Points:** | **1a- User search for a Date sheet which doesn’t exist**  -System will prompt to “Date Sheet Not Found”  **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | High Frequent in use |

SRS25, SRS26, SRS27, SRS28

|  |  |
| --- | --- |
| **Use case ID:** | UC-10 |
| **Use case Name:** | Update Date Sheet |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Update date Sheet information |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Update Date sheet |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

3. Date sheet must be present in database before updation

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Date Sheet record |  |  |
|  |  | 2 | System process and Date sheet information is shown to user |
| 3 | User will update the Date sheet information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully update the Date sheet information |
| **Extension Points:** | **1a- User search for date sheet which doesn’t exist**  -System will prompt to “Date sheet Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |

SRS29, SRS30, SRS31

3. Date sheet must be present in data base before deletion

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for date sheet record |  |  |
|  |  | 2 | System process and date sheet information is shown to user |
| 3 | User will delete the date sheet information |  |  |
|  |  | 4 | System will process and delete the information from the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Use case ID:** | UC-11 |
| **Use case Name:** | Delete Date Sheet |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Delete Date Sheet information |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Delete Date sheet information |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully delete an date sheet information |
| **Extension Points:** | **1a- User search for an date sheet which doesn’t exist**  -System will prompt to “Date sheet Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |

SRS32

|  |  |
| --- | --- |
| **Use case ID:** | UC-12 |
| **Use case Name:** | Add Timetable |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will add Timetable |
| **Primary Actors:** | <Admin> |
| **Stake Holders & Interest:** | **DEO:** Wants to successfully Add Timetable |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have right to add Timetable |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add Timetable |  |  |
|  |  | 2 | System will show form to the user to entry new Timetable |
| 3 | User will enter the all information of a new Timetable |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use  SRS34, SRS35, SRS36 |
|  |  |

|  |  |
| --- | --- |
| **Use case ID:** | UC-13 |
| **Use case Name:** | Search Timetable |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Search the Timetable |
| **Primary Actors:** | <Administrator><Teacher><Candidate> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Search the Timetable  **Teacher:** Wants to successfully Search the Timetable  **Candidate:** Wants to successfully Search the Timetable |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Timetable must be uploaded in data base before |

searching

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Timetable |  |  |
|  |  | 2 | System Process and shows the Search Timetable form to the User |
| 3 | User will enter the Timetable details to be searched |  |  |
|  |  | 4 | System will verify the information and Timetable will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |  |
| --- | --- | --- |
| **Post-Condition:** | User successfully Search Timetable | |
| **Extension Points:** | **1a- User search for a Timetable which doesn’t exist**  -System will prompt to “Timetable Not Found” **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** | |
| **Priority:** | High | |
| **Frequency:** | High Frequent in use | |
| **Cross-Reference:** | SRS37, SRS38, SRS39 | |
| **Use case ID:** | | UC-14 | |
| **Use case Name:** | | Update Timetable | |
| **Use case Prepared by:** | | Badar Khalil **Use case Prepared on:** 2nd July,2019 | |
| **Use case updated by:** | | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 | |
| **Use case Description:** | | This use case describes how a user will Update Timetable information | |
| **Primary Actors:** | | <Administrator> | |
| **Stake Holders & Interest:** | | **Administrator:** Wants to successfully Update Timetable | |
| **Pre-Condition:** | | 1. System must be in running state 2. System must authenticate the user | |

6. Timetable must be present in database before updation

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Timetable record |  |  |
|  |  | 2 | System process and Timetable information is shown to user |
| 3 | User will update the Timetable information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully update the Timetable information |
| **Extension Points:** | **1a- User search for**  **Timetable which doesn’t exist**  -System will prompt to “ Timetable Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Cross-Reference:** | SRS40, SRS41, SRS42,SRS43 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-15 |
| **Use case Name:** | Add Attendance |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will add Attendance |
| **Primary Actors:** | <Teacher> |
| **Stake Holders & Interest:** | **Teacher:** Wants to successfully Add Attendance |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have right to add Attendance |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add Attendance |  |  |
|  |  | 2 | System will show form to the user to enter Attendance |
| 3 | User will add Attendance of the candidate |  |  |
|  |  | 4 | System will insert Attendance to the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Cross-Reference:** | SRS45, SRS46, SRS47 |
|  |  |

UC-16

Search Attendance Badar KhalilSana Aftab Abbasi

|  |  |
| --- | --- |
| **Use case Description:** | This use case describes how a user will Search the Attendance |
| **Primary Actors:** | <Administrator><Teacher><Candidate> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Search the Attendance  **Teacher:** Wants to successfully Search the Attendance  **Candidate:** Wants to successfully Search the Attendance |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Attendance must be uploaded in data base before |

searching

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Attendance |  |  |
|  |  | 2 | System Process and shows the Search Attendance form to the User |
| 3 | User will enter the Attendance details to be searched |  |  |
|  |  | 4 | System will verify the information and Attendance will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |  |
| --- | --- | --- |
| **Post-Condition:** | User successfully Search Attendance | |
| **Extension Points:** | **1a- User search for a Attendance which doesn’t exist**  -System will prompt to “Attendance Not Found” **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** | |
| **Priority:** | High | |
| **Frequency:** | High Frequent in use | |
| **Cross-Reference:** | SRS48, SRS49, SRS50 | |
| **Use case ID:** | | UC-17 | |
| **Use case Name:** | | Update Attendance | |
| **Use case Prepared by:** | | Badar Khalil **Use case Prepared on:** 2nd July,2019 | |
| **Use case updated by:** | | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 | |
| **Use case Description:** | | This use case describes how a user will Update Attendance information | |
| **Primary Actors:** | | <Teacher> | |
| **Stake Holders & Interest:** | | **Teacher:** Wants to successfully Update Timetable | |
| **Pre-Condition:** | | 1. System must be in running state 2. System must authenticate the user | |

9. Attendance must be present in database before updation

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Attendance record |  |  |
|  |  | 2 | System process and Attendance information is shown to user |
| 3 | User will update the Attendance information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully update the Attendance information |
| **Extension Points:** | **1a- User search for**  **Attendance which doesn’t exist**  -System will prompt to “ Attendance Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Cross-Reference:** | SRS51, SRS52, SRS53,SRS54 |

UC-18

Generate Attendance Report

Badar Khalil

Sana Aftab Abbasi

This use case describes how a user will Generates Report of Attendance

|  |  |
| --- | --- |
| **Use case Description:** | information |
| **Primary Actors:** | <Teacher><Candidate><Admin> |
| **Stake Holders & Interest:** | **Teacher:** Wants to successfully Generate a Report of Attendance information  **Candidate:** Wants to successfully Generate a Report of Attendance information  **Admin:** Wants to successfully Generate a Report of Attendance information |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have rights to Generate Reports |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User will select View Attendance  Information |  |  |
|  |  | 2 | System Process and ask for Parameters |
| 3 | User will selects the Parameters and submits |  |  |
|  |  | 4 | System process and Shows course report and contents |
| 5 | User will be able to download this information in PDF/MS Word format |  |  |

|  |  |  |
| --- | --- | --- |
| **Post-Condition:** | User successfully Generate Attendance Report | |
| **Extension Points:** | **1a- if a User doesn’t selects the Parameters**  -System will prompt to “selects the Parameters” | |
| **Priority:** | High | |
| **Frequency:** | Low Frequent in use | |
| **Cross-Reference:** | SRS55 | |
| **Use case ID:** | | UC-19 | |
| **Use case Name:** | | Add NCEAC Profile | |
| **Use case Prepared by:** | | Badar Khalil **Use case Prepared on:** 2nd July,2019 | |
| **Use case updated by:** | | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 | |
| **Use case Description:** | | This use case describes how a user will Add NCEAC Profile | |
| **Primary Actors:** | | <Teacher> | |
| **Stake Holders & Interest:** | | **Teacher:** Wants to successfully Add his NCEAC Profile | |
| **Pre-Condition:** | | 1. System must be in running state 2. System must authenticate the user 3. User must have right to add his NCEAC Profile | |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add NCEAC Profile |  |  |
|  |  | 2 | System will show form to the user to enter NCEAC Profile details |
| 3 | User will add his NCEAC Profile details |  |  |
|  |  | 4 | System will insert NCEAC Profile to the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Cross-Reference:** | SRS56, SRS57, SRS58 |

UC-20

Search NCEAC Profile

Badar KhalilSana Aftab Abbasi

|  |  |
| --- | --- |
| **Use case Description:** | This use case describes how a user will Search the NCEAC Profile |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully Search the Attendance |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. NCEAC Profile must be uploaded in data base before |

searching

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for NCEAC Profile |  |  |
|  |  | 2 | System Process and shows the Search NCEAC Profile form to the User |
| 3 | User will enter the NCEAC Profile details to be searched |  |  |
|  |  | 4 | System will verify the information and  NCEAC Profile will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Search NCEAC Profile |
| **Extension Points:** | **1a- User search for a NCEAC** **Profile which doesn’t exist**  -System will prompt to “Profile Not Found” **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | High Frequent in use |
| **Cross-Reference:** | SRS59, SRS60, SRS61 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-21 |
| **Use case Name:** | Update NCEAC Profile |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Update NCEAC Profile information |
| **Primary Actors:** | <Teacher> |
| **Stake Holders & Interest:** | **Teacher:** Wants to successfully Update NCEAC Profile |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

12. NCEAC Profile must be present in database before updation

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for NCEAC Profile record |  |  |
|  |  | 2 | System process and NCEAC Profile information is shown to user |
| 3 | User will update the NCEAC Profile information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully update the NCEAC Profile information |
| **Extension Points:** | **1a- User search for**  **NCEAC** **Profile which doesn’t exist**  -System will prompt to “NCEAC Profile Not Found” **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Cross-Reference:** | SRS62, SRS63, SRS64 |

UC-22

Generate NCEAC Profile Report

Badar Khalil

Sana Aftab Abbasi

This use case describes how a user will Generates Report of NCEAC

|  |  |
| --- | --- |
| **Use case Description:** | Profile. |
| **Primary Actors:** | <Admin> |
| **Stake Holders & Interest:** | **Admin:** Wants to successfully Generate a Report of Attendance information |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have rights to Generate Reports |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User will select View NCEAC Profile  Information |  |  |
|  |  | 2 | System Process and ask for Parameters |
| 3 | User will selects the Parameters and submits |  |  |
|  |  | 4 | System process and Shows course report and contents |
| 5 | User will be able to download this information in PDF/MS Word format |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Generate NCEAC Profile Report |
| **Extension Points:** | **1a- if a User doesn’t selects the Parameters**  -System will prompt to “selects the Parameters” |
| **Priority:** | High |
| **Frequency:** | Low Frequent in use |
| **Cross-Reference:** | SRS65 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-23 |
| **Use case Name:** | Add Feedback |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Add Feedback |
| **Primary Actors:** | <Candidate> |
| **Stake Holders & Interest:** | **Candidate:** Wants to successfully Add his Feedback |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have right to add his Feedback |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add Feedback |  |  |
|  |  | 2 | System will show form to the user to enter Feedback |
| 3 | User will add his Feedback |  |  |
|  |  | 4 | System will insert Feedback to the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |

Less Frequent in use

SRS66, SRS67, SRS68

UC-24

View Feedback

Badar Khalil

Sana Aftab Abbasi **Use case updated on:**

|  |  |
| --- | --- |
| **Use case Description:** | This use case describes how a user will view the Feedback |
| **Primary Actors:** | <Administrator> |
| **Stake Holders & Interest:** | **Administrator:** Wants to successfully view the Feedback |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Feedback must be uploaded in database before |

viewing

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for Feedback |  |  |
|  |  | 2 | System Process and shows all feedbacks to the User with search box |
| 3 | User will enter the Feedback details to be viewed |  |  |
|  |  | 4 | System will verify the information and Feedback will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Search Feedback |
| **Extension Points:** | **1a- User search for a Feedback** **details** **which doesn’t exist**  -System will prompt to “Feedback details Not Found”  **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |
| **Frequency:** | High Frequent in use |
| **Cross-Reference:** | SRS69,SRS70 |

|  |  |
| --- | --- |
| **Use case ID:** | UC-25 |
| **Use case Name:** | Add News and Notifications |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Add News and Notifications |
| **Primary Actors:** | <Admin> |
| **Stake Holders & Interest:** | **Admin:** Wants to successfully Add his News and Notifications |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. User must have right to add his news and notifications |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User Selects Add news and notifications |  |  |
|  |  | 2 | System will show form to the user to enter new News and Notifications details |
| 3 | User will add his notifications details |  |  |
|  |  | 4 | System will insert information to the system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully enters the information into the system |
| **Extension Points:** | **3a- If user leaves any field blank**  -System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |

Less Frequent in use

SRS71, SRS72, SRS73

UC-26

Search News and Notifications

Badar Khalil **Use case Prepared on:**

Sana Aftab Abbasi **Use case updated on:**

This use case describes how a user will Search the News and

|  |  |
| --- | --- |
| **Use case Description:** | Notifications |
| **Primary Actors:** | <Candidate><Teacher> |
| **Stake Holders & Interest:** | **Candidate:** Wants to successfully Search the News and Notifications **Teacher:** Wants to successfully Search the News and Notifications |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user 3. news and notifications must be uploaded in data base |

before searching

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for news and notifications |  |  |
|  |  | 2 | System Process and shows the Search news and notifications form to the User |
| 3 | User will enter the news and notifications details to be searched |  |  |
|  |  | 4 | System will verify the information and news and notifications will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully Search News and Notifications |
| **Extension Points:** | **1a- User search for a News** **and** **Notifications which doesn’t exist**  -System will prompt to “Notifications Not Found” **3a- If user leaves the field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |

High Frequent in use

SRS74, SRS75, SRS76

|  |  |
| --- | --- |
| **Use case ID:** | UC-27 |
| **Use case Name:** | Update News and Notifications |
| **Use case Prepared by:** | Badar Khalil **Use case Prepared on:** 2nd July,2019 |
| **Use case updated by:** | Sana Aftab Abbasi **Use case updated on:** 3rd July,2019 |
| **Use case Description:** | This use case describes how a user will Update news and notifications information |
| **Primary Actors:** | <Admin> |
| **Stake Holders & Interest:** | **Admin:** Wants to successfully Update news and notifications |
| **Pre-Condition:** | 1. System must be in running state 2. System must authenticate the user |

15. News and notifications must be present in database before updation

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User search for news and notifications record |  |  |
|  |  | 2 | System process and news and notifications information is shown to user |
| 3 | User will update the News and Notifications information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

|  |  |
| --- | --- |
| **Post-Condition:** | User successfully update the news and notifications information |
| **Extension Points:** | **1a- User search for** **News** **and** **Notifications which doesn’t exist**  -System will prompt to “ Notifications Profile Not Found”  **3a- If user leaves any field blank**  **-**System will prompt to “Fill the blank field”  **3b- If user enter invalid data in any field**  -System will prompt to enter “Valid Data**”** |
| **Priority:** | High |

Less Frequent in use

SRS77, SRS78, SRS79

**3.2 System Sequence Diagram**

### SSD: 01 Process Registration

Administration

**:**

**System**

Start registration Process (

)

user\_Registration(useranme,user\_password,CNIC,...)

User Registered successfully message shown to the user

**Figure 3.2: Process Registration**

### SSD: 02 Process Login

Administration/Teacher/Student

**:**

**System**

Start Login Process (

)

user\_Login(useranme,user\_password)

successfully Login Message shown to the user

**Figure 3.3:** **Process Login**

### SSD: 03 Add Course

Administration

**:**

**System**

Start Add Course Process ()

enter add\_Course(courseID,CourseName,CourseCreditHours,SmesterOFStudy,........)

Course addded successfully message shown to the user

**\*More Entries**

**Figure 3.4: Add Course**

### SSD: 04 Search Course

Administration/Teacher/Student

**:**

**System**

Start Search Course Process ()

search\_course(courseID/courseName)

Searched courses details will be shown to the user

**\*More Entries**

**Figure 3.5: Search Course**

### SSD: 05 Update Course

Administration

**:**

**System**

Start Update Course Process ()

search\_course(courseID/CourseName)

Course updated successfully message will be shown to the user

**\*More Entries**

Serched course result will be shown to the user

update\_course(CourseName,CourseCrHrs,SmesterOfStudy,CourseContents,......)

**Figure 3.6: Update Course**

### SSD: 06 Delete Course

Administration

**:**

**System**

Start Delete Course Process ()

delete\_course(courseID )

Course deleted successfully message will be shown to the user

**\*More Entries**

**Figure 3.7: Delete Course**

### SSD: 07 Generate Course Report

Administration

**:**

**System**

Start Generate Course Report Process ()

Generate\_course\_report(smesterWise/Overall)

Report will be generated in PDF/MS Word Format and shown to the user

**\*More Entries**

**Figure 3.8: Generate Course Report**

### SSD: 08 Add Date Sheet

Administration

**:**

**System**

Start Add Date sheet Process ()

upload\_dateSheet(dateTimeYear,Smester,dateSheetFile,....)

Date sheet added successfully message will be shown to the use

r

**\*More Entries**

**Figure 3.9: Add Date sheet**

### SSD: 09 Search Date Sheet

Administration/Teacher/Student

**:**

**System**

Start Search Date sheet Process ()

search\_dateShee(smesterName)

Date Sheet Result will be shown to the user

**\*More Entries**

**Figure 3.10: Search Date sheet**

### SSD: 10 Update Date Sheet

Administration

**:**

**System**

Start Update Date sheet Process ()

search\_dateShee(smesterName)

Date Sheet Result will be shown to the user

**\*More Entries**

update\_dateShee(smesterName,dateSheetFile)

Date Sheet updated successfully message will be shown to the user

**Figure 3.11: Update Date sheet**

### SSD: 11 Delete Date Sheet

Administration

**:**

**System**

Start Delete Date sheet Process ()

delete\_dateShee(smesterName)

Date Sheet deleted successfully message will be shown to the user

**\*More Entries**

**Figure 3.12: Delete Date sheet**

### SSD: 12 Add Timetable

**Administration**

**:**

**System**

Start Add Timetable Process()

enter add\_timetable(timetableid,smester,date,....)

Timetable successfully added message will be shown to the user

**\*More entries**

**Figure 3.13: Add Timetable**

### SSD: 13 Search Timetable

**Administration/Teacher/Student**

**:**

**System**

Start Search Timetable Process()

search\_timetable(Smester)

Timetable successfully shown to the user

**\*More entries**

**Figure 3.14: Search Timetable**

### SSD: 14 Update Timetable

**Administration**

**:**

**System**

Start Update Timetable Process()

search\_timetable(Smester)

Timetable successfully shown to the user

**\*More entries**

update\_timetable(Smester,datetime,timetableFile)

Timetable successfully updated message will be shown to the user

**Figure 3.15: Update Timetable**

### SSD: 15 Add Attendance

**Teacher**

**:**

**System**

Start Add Attendance Process()

enter add\_attendance(date,smester,section,....)

Attendance successfully added message will be shown to the user

**\*More entries**

**Figure 3.16: Add Attendance**

### SSD: 16 Search Attendance

**Administration/Teacher**

**:**

**System**

Start Search Attendance Process()

search\_attendance(smester,section,date)

Attendancle successfully shown to the user

**\*More entries**

**Figure 3.17: Search Attendance**

### SSD: 17 Update Attendance

**Teacher**

Start Update AttendanceProcess()

search\_attendance(Smester,section,date)

Attendance successfully shown to the user

**\*More entries**

update\_attendance(Student presence )

Attendance successfully updated message will be shown to the user

**Figure 3.18: Update Attendance**

### SSD: 18 Generate Attendance Report

**Administration**

**:**

**System**

Start Generate Attendance Report Process()

search\_attendance\_report(smester wise/section wise/overall)

Report will be generated in PDF/MS Word formate and shown to the user

**\*More entries**

**Figure 3.19: Generate Attendance Report**

### SSD: 19 Add NCEAC Profile

**Teacher/Student**

**:**

**System**

Start Add NCEAC Profile Process()

enter add\_NCEAC\_profile(student\_enrollment,teacher\_CNIC......)

Add NCEAC Profile successfully added message will be shown to the user

**\*More entries**

**Figure 3.20: Add NCEAC profile**

### SSD: 20 Search NCEAC Profile

**Administration**

**:**

**System**

Start Search NCEAC Profile Process()

search\_NCEAC profile(student\_MIS\_ID/teacher\_CNIC)

NCEAC Profile successfully shown to the user

**\*More entries**

**Figure 3.21: Search NCEAC Profile**

### SSD: 21 Update NCEAC Profile

**Teacher/Student**

Start Update NCEAC Profile Process()

search\_NCEAC Profile(student\_MIS\_ID/teache\_CNIC)

NCEAC Profile successfully shown to the use

r

**\*More entries**

update\_NCEAC Profile(name,address,phone\_number.....)

NCEAC Profile successfully updated message will be shown to the user

**Figure 3.22: Update NCEAC Profile**

### SSD: 22 Generate Report

**Administration**

**:**

**System**

Start Generate Report Process()

search\_NCEAC\_profile(student\_MIS\_ID/teacher\_CNICl)

Report will be generated in PDF/MS Word formate and shown to the user

**\*More entries**

**Figure 3.23: Generate Report**

### SSD: 23 Add Feedback

**Student**

Start Add Feedback Process()

enter add\_feedback(course\_name,teacher\_name,.....)

Feedback successfully added message will be shown to the user

**\*More entries**

**Figure 3.24: Add Feedback**

### SSD: 24 View Feedback

**Administration**

**:**

**System**

Start View Feedback Process()

search\_feedback(course\_name/teacher\_name)

Feedback successfully shown to the user

**\*More entries**

**Figure 3.25: View Feedback**

### SSD: 25 Add News and Notification

**Administration**

Start Add News and NotificationProcess()

enter add\_News\_and\_Notification (Title,description\_date,time,.....)

News and Notification successfully added message will be shown to the user

**\*More entries**

**Figure 3.26: Add News and Notification**

### SSD: 26 Search News and Notification

**Administration/Teacher/Student**

**:**

**System**

Start Search News and Notification Process()

search\_News\_and\_Notification(title/description\_date)

News and Notification successfully shown to the user

**\*More entries**

**Figure 3.27: Search News and Notifications**

### SSD: 27 Update News and Notification

**Administration**

Start Update News and Notification Process()

search\_News\_and\_Notification(title/description\_date)

News and Notification successfully shown to the user

**\*More entries**

update\_News and Notification(title,description\_date,time.....)

News and Notification successfully updated message will be shown to the user

**Figure 3.28: Update News and Notification**

#### 3.3 Entity Relationship Diagram



**Figure 3.29: ER Diagram**

#### 3.4 Normalization

**3.4.1 Normalization for Table CANDIDATE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Candidate\_nic** | **St\_fatherName** | **St\_gender** | **St\_Nationality** | **St\_name** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_enrollmentNumber** | **St\_passportNumber** | **St\_admissionDate** | **St\_dateOfBirth** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_contactNumber** | **St\_email** | **St\_DomicileDstrict** | **St\_DomicileProvince** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_MailingAdress** | **St\_MailingCity** | **St\_SSCDegreeName** | **St\_SSCBoardName** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_SSCTotalMarks** | **St\_SSCObtainedMarks** | **HSSCDegName** | **St\_HSSCBoardName** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **St\_HSSCTotalMarks** | **St\_HSSCObtainedMarks** | **St\_admissionDate** |
|  |  |  |
|  |  |  |
|  |  |  |

**1NF:**

* Primary Key is not repeating so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists because of so it is in 2NF **3NF:**
* Transitive Dependency is Present as non-key Attributes (City, District and Province dependent on each other) so table will take the form

**3.4.2 Table PROVINCE:**

|  |  |
| --- | --- |
| **Candidate\_nic** | **Province** |
|  |  |
|  |  |

**3.4.3 Table DISTRICT:**

|  |  |
| --- | --- |
| **Province** | **District** |
|  |  |
|  |  |

**3.4.4 Table CANDIDATE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Candidate\_nic** | **St\_fatherName** | **St\_gender** | **St\_Nationality** | **St\_name** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_enrollmentNumber** | **St\_passportNumber** | **St\_admissionDate** | **St\_dateOfBirth** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_MailingAdress** | **St\_MailingCity** | **St\_SSCDegreeName** | **St\_SSCBoardName** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **St\_SSCtMarks** | **St\_SSCObtainedMarks** | **St\_HSSCDegreeName** | **St\_HSSCBoardName** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **St\_HSSCTotalMark**  **s** | **St\_HSSCObtainedMark s** | | **St\_admissionDat e** | | **St\_contactNumbe r** |
|  |  | |  | |  |
| **St\_email** | |
|  | |
|  | |
|  | |

**3.4.5 Normalization for Table TEACHER:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **teacher\_nic** | **t\_Name** | **t\_gender** | **t\_PhoneNumber** | **t\_email** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **t\_mailingAddress** | **t\_mailingCity** | **t\_province** | **t\_EmpDetails** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **t\_ContractDetails** | **t\_AcademicRank** | **t\_JoiningDate** | **t\_ActiveStatus** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **t\_DegType** | **t\_isComputingTeacher** | **t\_DegAwardCountry** | **t\_UniName** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **t\_LeavingDate** | **t\_AcademicQualification** | **t\_DegreeName** | **t\_FieldOfStudy** |
|  |  |  |  |
|  |  |  |  |

|  |  |
| --- | --- |
| **t\_DegreeStartDate** | **t\_DegreeEndDate** |
|  |  |
|  |  |
|  |  |

**1NF:**

* Primary Key is not repeating so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists ,so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.6 Normalization for Table COURSE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course\_goal** | **Course\_contents** | **Course\_teaching\_smester** | **Course\_techMappingForm** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **course\_code** | **Course\_crHrs** | **Course\_preRequisite** | **Course\_RecommendedBooks** | **Course\_title** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table Course so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in Course table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.7 Normalization for Table SMESTERSECTION:**

|  |  |  |  |
| --- | --- | --- | --- |
| **SemestersectionID** | **Candidate\_nic** | **SmesterName** | **smesterSection** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table SMESTERSECTION so, it is in 1NF

**2NF:**

* Partial Function Dependency also not exists in SMESTERSECTION table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.8 Normalization for Table CLASS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Class\_id** | **Course\_code** | **SmesterSection\_ID** | **Teacher\_nic** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table CLASS so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in CLASS table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.9 Normalization for Table TIMETABLE:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TimetableID** | **Candidate\_nic** | **Teacher\_nic** | **TimetableDates** | **TimetableShifts** | **Attachments** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table TIMETABLE so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in TIMETABLE table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.10 Normalization for Table ATTENDANCE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **AttendanceDate** | **Class\_id** | **Candidate\_nic** | **value** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table ATTENDANCE so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in ATTENDANCE table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.11 Normalization for Table DATESHEET:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **DatesheetID** | **Candidate\_nic** | **Teacher\_nic** | **DateSheetDate** | **DateSheetShift** | **Attachments** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table DATESHEET so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in DATESHEET table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.12 Normalization for Table NEWSANDNOTIFICATION:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **newsAndNotificati onID** | **Candidate\_ nic** | **Teacher\_ nic** | **News\_ti tle** | **News\_d esc** | **News\_da tes** | **Attachme nts** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table NEWSANDNOTIFICATION so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in NEWSANDNOTIFICATION table, so it is in 2NF

**3NF:**

* Transitive Dependency is also not present here so the table is already in 3NF

**3.4.13 Normalization for Table FEEDBACK:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FeedbackID** | **Candidate\_nic** | **class\_id** | **Feddback\_rating** | **Feddback**  **\_details** | **suggestionRequest** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**1NF:**

* Primary Key is not repeating in the table FEEDBACK so, it is in 1NF **2NF:**
* Partial Function Dependency also not exists in FEEDBACK table, so it is in 2NF **3NF:**
* Transitive Dependency is also not present here so the table is already in 3NF

#### 3.5 Relational Tables

**3.5.1 Relational Table of CANDIDATE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraint**  **s** | **Primary Key** | **Foreign Key** |
| **Candidate\_nic** | BIGINT(13) | Not Null | YES | Nil |
| **St\_fatherName** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_gender** | nvarchar(10) | Not Null | Nil | Nil |
| **St\_Nationality** | nvarchar(12) | Not Null | Nil | Nil |
| **St\_name** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_enrollmentNumber** | nvarchar(40) | Not Null | Nil | Nil |
| **St\_passportNumber** | nvarchar(30) | Null | Nil | Nil |
| **St\_admissionDate** | Date | Not Null | Nil | Nil |
| **St\_dateOfBirth** | Date | Not Null | Nil | Nil |
| **St\_contactNumber** | BIGINT(13) | Not Null | YES | Nil |
| **St\_email** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_MailingAdress** | nvarchar(100) | Not Null | Nil | Nil |
| **St\_MailingCity** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_SSCDegreeName** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_SSCBoardName** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_SSCTotalMarks** | Int | Not Null | Nil | Nil |
| **St\_SSCObtainedMarks** | Int | Not Null | Nil | Nil |
| **St\_HSSCDegreeName** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_HSSCBoardName** | nvarchar(50) | Not Null | Nil | Nil |
| **St\_HSSCTotalMarks** | Int | Not Null | Nil | Nil |
| **St\_HSSCObtainedMar**  **ks** | Int | Not Null | Nil | Nil |
| **St\_admissionDate** | Date | Not Null | Nil | Nil |

**3.5.2 Relational Table of PROVINCE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **Candidate\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **Province** | nvarchar(30) | Unique | YES | Nil |

**3.5.3 Relational Table of DISTRICT:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **Province** | nvarchar(30) | Not Null | Nil | YES |
| **District** | nvarchar(15) | Unique | YES | Nil |

**3.5.4 Relational Table of TEACHER:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **teacher\_nic** | BIGINT(13) | Unique | YES | Nil |
| **t\_Name** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_gender** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_PhoneNumber** | BIGINT(18) | Not Null | Nil | Nil |
| **t\_email** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_mailingAddress** | nvarchar(100) | Not Null | Nil | Nil |
| **t\_mailingCity** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_province** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_EmpDetails** | nvarchar(100) | Not Null | Nil | Nil |
| **t\_ContractDetails** | nvarchar(100) | Not Null | Nil | Nil |
| **t\_AcademicRank** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_JoiningDate** | nvarchar(50) | Not Null | Nil | Nil |
| **t\_ActiveStatus** | nvarchar(10) | Not Null | Nil | Nil |
| **t\_LeavingDate** | nvarchar(50) | Not Null | Nil | Nil |

**3.5.5 Relational Table of COURSE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **course\_code** | nvarchar(20) | Unique | YES | Nil |
| **Course\_crHrs** | Int | Not Null | Nil | Nil |
| **Course\_preRequisite** | nvarchar(50) | Not Null | Nil | Nil |
| **Course\_RecommendedBooks** | nvarchar(100) | Not Null | Nil | Nil |
| **Course\_title** | nvarchar(50) | Not Null | Nil | Nil |
| **Course\_goal** | nvarchar(100) | Not Null | Nil | Nil |
| **Course\_contents** | nvarchar(50) | Not Null | Nil | Nil |
| **Course\_teaching\_smester** | nvarchar(50) | Not Null | Nil | Nil |
| **Course\_techMappingForm** | nvarchar(50) | Not Null | Nil | Nil |

**3.5.6 Relational Table of SMESTERSECTION:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **SemestersectionID** | int | Unique | YES | Nil |
| **Candidate\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **SmesterName** | nvarchar(12) | Not Null | Nil | Nil |
| **smesterSection** | nvarchar(50) | Not Null | Nil | Nil |

**3.5.7 Relational Table of CLASS:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **Class\_id** | Int | Unique | YES | Nil |
| **Course\_code** | nvarchar(20) | Not Null | Nil | YES |
| **SmesterSection\_ID** | Int | Not Null | Nil | Nil |
| **Teacher\_nic** | BIGINT(13) | Not Null | Nil | YES |

**3.5.8 Relational Table of TIMETABLE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **TimetableID** | nvarchar(12) | Not Null | YES | Nil |
| **Candidate\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **Teacher\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **TimetableDates** | Date | Not Null | Nil | Nil |
| **TimetableShifts** | nvarchar(50) | Not Null | Nil | Nil |
| **Attachments** | nvarchar(50) | Not Null | Nil | Nil |

**3.5.9 Relational Table of ATTENDANCE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **AttendanceDate** | date | Not Null | YES | Nil |
| **Class\_id** | int | Not Null | Nil | YES |
| **Candidate\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **Value** | nvarchar(15) | Unique | Nil | Nil |

**3.5.10 Relational Table of DATESHEET:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **DatesheetID** | int | Auto  Increment | YES | Nil |
| **Candidate\_nic** | nvarchar(50) | Not Null | Nil | Nil |
| **Teacher\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **DateSheetDate** | BIGINT(13) | Not Null | Nil | YES |
| **DateSheetShift** | nvarchar(50) | Not Null | Nil | Nil |
| **Attachments** | nvarchar(50) | Not Null | Nil | Nil |

**3.5.11 Relational Table of NEWSANDNOTIFICATION:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraint**  **s** | **Primary Key** | **Foreign Key** |
| **newsAndNotificationI D** | int | Auto  Increment | YES | Nil |
| **Candidate\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **Teacher\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **News\_title** | nvarchar(50) | Not Null | Nil | Nil |
| **News\_desc** | nvarchar(50) | Not Null | Nil | Nil |
| **News\_dates** | date | Not Null | Nil | Nil |
| **Attachments** | nvarchar(50) | Not Null | Nil | Nil |

**3.5.12 Relational Table of FEEDBACK:**

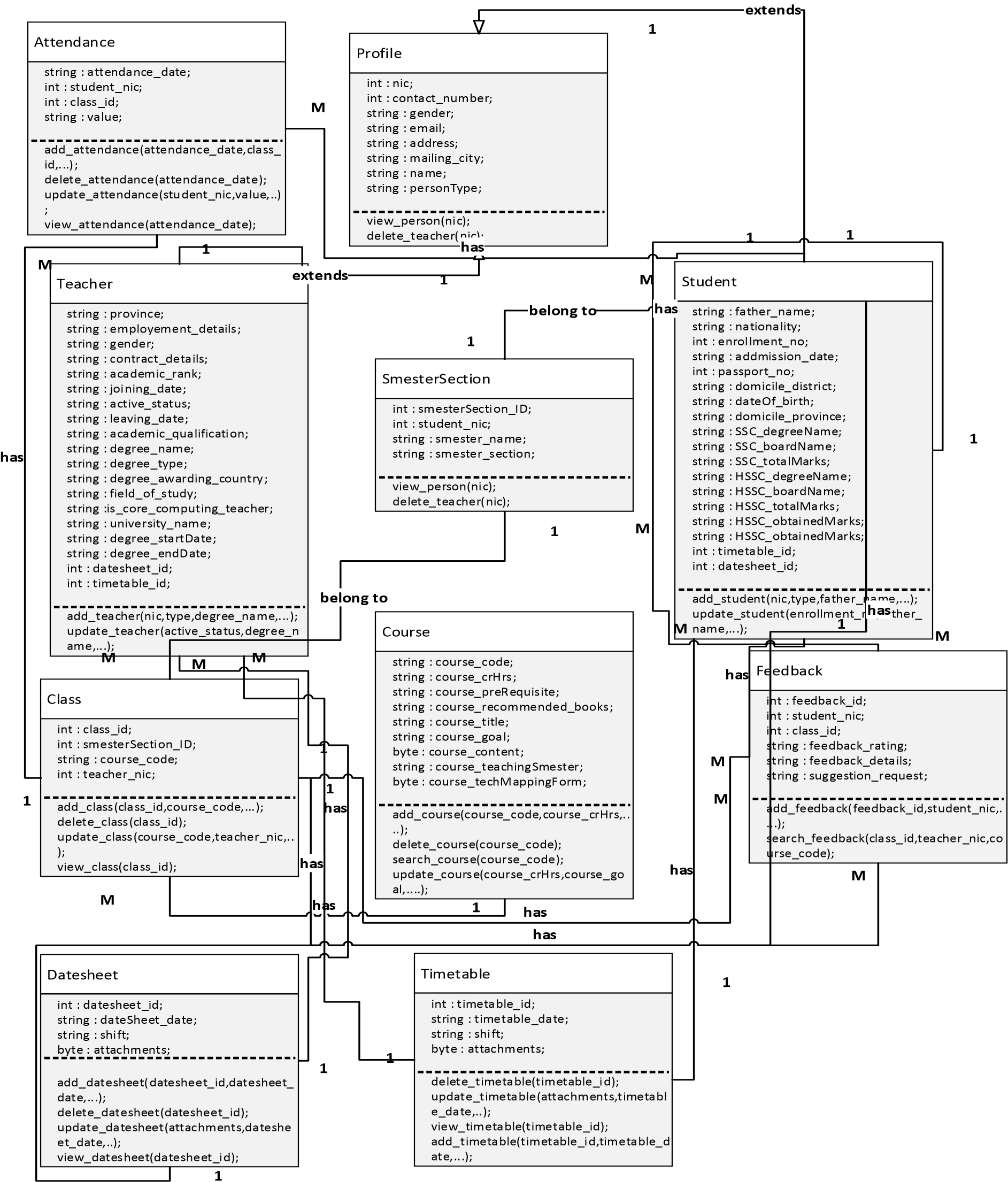
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Constraints** | **Primary Key** | **Foreign Key** |
| **FeedbackID** | int | Auto  Increment | YES | Nil |
| **Candidate\_nic** | BIGINT(13) | Not Null | Nil | YES |
| **class\_id** | Int | Not Null | Nil | YES |
| **Feddback\_rating** | nvarchar(100) | Not Null | Nil | Nil |
| **Feddback \_details** | nvarchar(100) | Not Null | Nil | Nil |
| **suggestionRequest** | nvarchar(100) | Not Null | Nil | Nil |

#### 3.6 Domain Model



**Figure 3.30: Domain Model**

#### 3.7 Class Diagram



**Figure 3.31: Class Diagram**

**Chapter 4**

### PROJECT MANAGEMENT

#### 4.1 Introduction

##### 4.1.1 Purpose

The purpose of this System is to collect, analyze, and perform relevant and necessary actions immediately without any delay to enhance the overall working performance of the department. It focuses on the functionality needed by the stakeholders these stakeholders are the HoD, Teacher and the Candidates. The previous system in use is totally manual and it is hard manage keep track of data and to perform simple tasks it requires an immense amount of time and energy. The Computer Science department wants to find a solution to reduce its time delays. The major goal of the system is to collect necessary information of the candidates and the teachers required for NCEAC accreditation purposes and to provide them with latest news and notifications regarding timetable, date sheets and many more.

##### 4.1.2 Scope

In our CSDAS (Computer Science Department Automation System) the old and manual system is transitioned to a new computerized system. Different modules will be constructed based on the requirements gathered from the stakeholder and the environment. Main entities like Teacher and Candidate will have separate modules and will be managed by the Admin i.e. Administrator/HOD (Head of Department). The system will allow to manage the official task very efficiently which will increase the overall performance of the department.

##### 4.1.3 Definitions, Acronyms and Abbreviations

There are different acronyms used in this artifact. Their detail is given one by one:

* The “Computer Science Department Automation System” is written in short as CSDAS.
* CS stands for “Computer Science”.
* FUUAST stands for “Federal Urdu University of Arts Science and Technology”.

##### 4.1.4 References

* CSDAS SRS
* CSDAS Vision document
* CSDAS Use Case Model

##### 4.1.5 Project Purpose, Scope and Objectives

The AI Based Online Exam Proctoring System (AIBOEPS)is designed for

“Federal Urdu University of Arts Science and technology”. This project will automate the Candidate and Teacher information, Candidate and Teacher Registration, Candidate Attendance and News and Notification and Feedback process. Currently there is no such digital and efficient system that is sufficient to fulfill all the needs of the university administration and faculty members.

##### 4.1.6 Assumptions and Constraints

There are four main constraints in the system. They are given below very briefly.

Their detailed information is available in “Risk Information Sheet”.

* The database (MySQL) engine, we are using has storage capacity problem with it.
* The project may not be completed in time, i.e. at the end of semester.
* Data related to the project may be lost due to system crash. This will result in late completion of project.
* Team members are not familiar with the working of android studio software that is used in the project for building Android application.

##### 4.1.7 Project Deliverables

* Project Charter
* Software Development Plan
* Vision
* Software Requirements Specifications
* Use Case Model
* Fully Dressed Use cases
* Data Model
* Design Model
* Entity Relationship Diagram
* Prototype
* Iteration Assessment Report
* Status Assessment Report
* Release Notes

##### 4.1.8 Evolution of Software Development Plan

This software development plan will be updated subsequently after phase or iteration. The target dates of each phase is given below.

|  |  |
| --- | --- |
| **Phase** | **Target Date** |
| Inception | 22 April, 2019 |
| Elaboration | 1 July, 2019 |
| Construction Iteration 1 | 12 October, 2019 |
| Construction Iteration 2 | 10 January, 2020 |
| Transition | 12 February, 2020 |

#### 4.2 Project Organization

##### 4.2.1 Organizational Structure

The team will be divided into three main groups. Each group is for Inception, Elaboration and Construction phase. These groups will work under the leadership of Project Manager. The details of these three groups can be seen as follow:

**4.2.2**

**faces**

**External I**

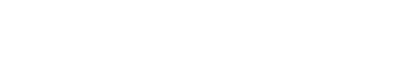
**nter**



**Figure 4.1: Organizational Structure**

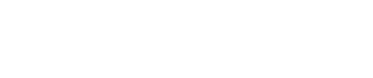


Project Manager



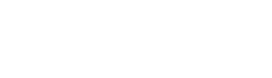
Requirements Specialist,

Designer



Database Designer,

Architect



Programmer,

Test

er

The project team will work with the administration department of Computer Science department of Federal Urdu University. Every module will be tested by the relevant person of the client organization. Through prototype the missing requirements will be included. Every function rejected or objected by the user will be modified or clarified by the team.

##### 4.2.3 Roles and Responsibilities

The following table shows the roles of team members that are represented in the above given diagram. The responsibilities are also given in the table of each role.

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| Project Manager  (Badar Khalil) | Project Manager is the leader of the team. He will manage the resources (time). He coordinates interaction of the teams and user. He assigns the task to team members and makes sure that the team is doing the “right” things. Project manager will also view the status and assess the performance of the project after each phase. |
| Requirements Specialist (Muhammad Saad Altaf) | Requirements specialist, lists all the requirements captured from the user in his environment or by the prototype. He identifies all the functional requirements described by the user. He interacts with the user and understands what he wants to communicate. Then he communicates the user’s desire to team members. |
| Designer (Badar Khalil) | Designer describes the requirements specified by the Requirements Specialist in technical way. He gives these requirements the diagrammatic representation to manage artifacts and other related documents that would be required at elaboration phase. |
| Database Designer (Sana Aftab Abbasi) | Database designer, is responsible for managing the data in database and its representation. He manages the representation of data in tabular format inside the database, defines their relationships and makes sure that data integrity is achieved. |
| Architect (Sana Aftab Abbasi) | Architect is responsible for managing all sorts of artifacts related to software design. He manages the technical activities. He establishes the overall structure for the software and maps the real world scenario to computer system. He identifies the classes and interfaces that would be required to implement the business logic, their relationship and their responsibilities in the project. |
| Programmer (Badar and Saad) | Programmer is responsible for translating the design into executable format. He gives the soul to the body of system, created by the architect. |
| Tester (Sana) | Tester, tests the implemented modules by the programmer. He makes sure that the programmer has made the “things right”. He establishes the test standards to which the system should conform for acceptance. |

#### 4.3 Project Plan

##### 4.3.1 Phase Plan

The project is planned in phases. Each phase may require iterations to complete its allocated task. These phase and iterations are listed below with their allocated time on the basis of estimations.

|  |  |  |  |
| --- | --- | --- | --- |
| **Phase** | **No. of Iterations** | **Start** | **End** |
| Inception Phase | 1 | Week 2 | Week 11 |
| Elaboration Phase | 1 | Week 11 | Week 25 |
| Construction Phase | 2 | Week 25 | Week 41 |
| Transition Phase | 1 | Week 41 | Week 43 |

##### 4.3.2 Milestones

The end of each phase will be considered as the milestone. The details of each milestone along their goals are given below. These details include the necessary documents that would be prepared till the end of milestone.

|  |  |
| --- | --- |
| **Milestone** | **Milestone Description** |
| Inception Phase | The Inception phase will develop the software requirements. It will identify all the functional and non-functional requirements. It will also develop an initial project plan. The major risks will be identified. All the major use cases will be identified and explored. |
| Elaboration Phase | Elaboration phase will result in the overall architecture and design of the system. The conceptual model of system will be mapped to the software model. The database design will be finalized. A prototype will be developed with no functionality available at that level. The software development plan will be revised if needed. |
| Construction Phase | Construction phase will include the overall implementation of business logic of the system. All the modules will be implemented one by one and at the end they will be integrated. All the necessary requirements will be completed to deploy system. |
| Transition Phase | User manuals will be prepared for the user guidance, if required. And the system will be deployed properly in the environment of the user after acceptance test. |

##### 4.3.3 Schedule Plan

The following table shows the high-level schedule plan for the milestones, set for the UMS project. This schedule plan includes the milestone name, iteration with its starting date and expected finishing date. The schedule plan may be reviewed after each phase or an iteration depending upon the work finished in that iteration.

|  |  |  |
| --- | --- | --- |
| **Task Name** | **Start** | **Finish** |
| **Milestones** | **Mon, 22/04/2019** | **Mon, 2/03/2020** |
| Lifestyle Objectives (end Inception Phase) | Mon, 22/04/2019 | Mon, 1/07/2019 |
| Lifestyle Architecture (end Elaboration Phase) | Mon, 1/07/2019 | Sat, 12/10/2109 |
| Initial Operational Capability (end Iteration 1 of  Construction Phase) | Mon, 12/10/2019 | Fri, 10/01/2020 |
| Initial Operational Capability (end Iteration 2 of  Construction Phase) | Fri, 10/01/2020 | Wed, 12/02/2020 |
| Product Release Milestone (end Transition Phase) | Wed, 12/02/2020 | Mon, 2/03/2020 |

##### 4.3.4 Project Resourcing

* **Staffing Plan:** Particular roles have been assigned to individuals of this project. These roles are described in Figure-1.
* **Resource Acquisition Plan:** In fact, no special hardware or software resource is required for the completion of this project.
* **Training Plan:** The team is skilled with all the necessary tools and techniques required for the project. But “Android Studio” software training would be required. A training plan has been devised to learn this software.

##### 4.3.5 Project Monitoring and Control

* **Schedule Control Plan:** “Project Status Report” will be issued through “Actual vs.

Proposed” plan giving at the end of iteration of phase. An “Iteration Assessment Report” will be prepared at the end of iteration. These reports will help the team to be on track and according to their schedule.

* **Budget Control Plan:** Since this project is a course requirement as well. So this project is being developed free of cost. So no budget control plan is required and applicable here.
* **Quality Control Plan:** Informal reviews will be acquired during the development of project. This will ensure that the system is meeting its requirements.
* **Reporting Plan:** Project assessment report will be prepared after iteration. This will ensure that the progress on project in actual is equal to the expected progress.
* **Measurement Plan**:Time estimate has been completed on the basis of project domain and available human resource.
* **Risk Management Plan:** Major risks and their remedies have been discussed in “Risk Information Sheet”.

#### 4.4 Technical Process Plan

##### 4.4.1 Methods, Tools and Techniques

* The standard guidelines of RUP are used
* MySQL Database is used to store information
* Android with Java is used for front-end implementation and business logic
* Back end admin Panel will be created using Bootstrap, HTML, CSS and JS and PHP.
* Crystal Reports software for generating reports

##### 4.4.2 Infrastructure Plan

This project will be deployed at “Federal Urdu University of Arts Science and

Technology”. All the necessary software will be installed as they would be required.

#### 4.5 Supporting Process Plans

##### 4.5.1 Documentation Plan

During the development of project the following documents will be prepared.

* SRS (Software Requirements Specification)
* Vision document
* SDP (Software Development Plan)
* Use Case Model
* Entity Relationship Diagram
* UML design diagrams (as required)
* Project Status Assessment Report
* Iteration Assessment Report

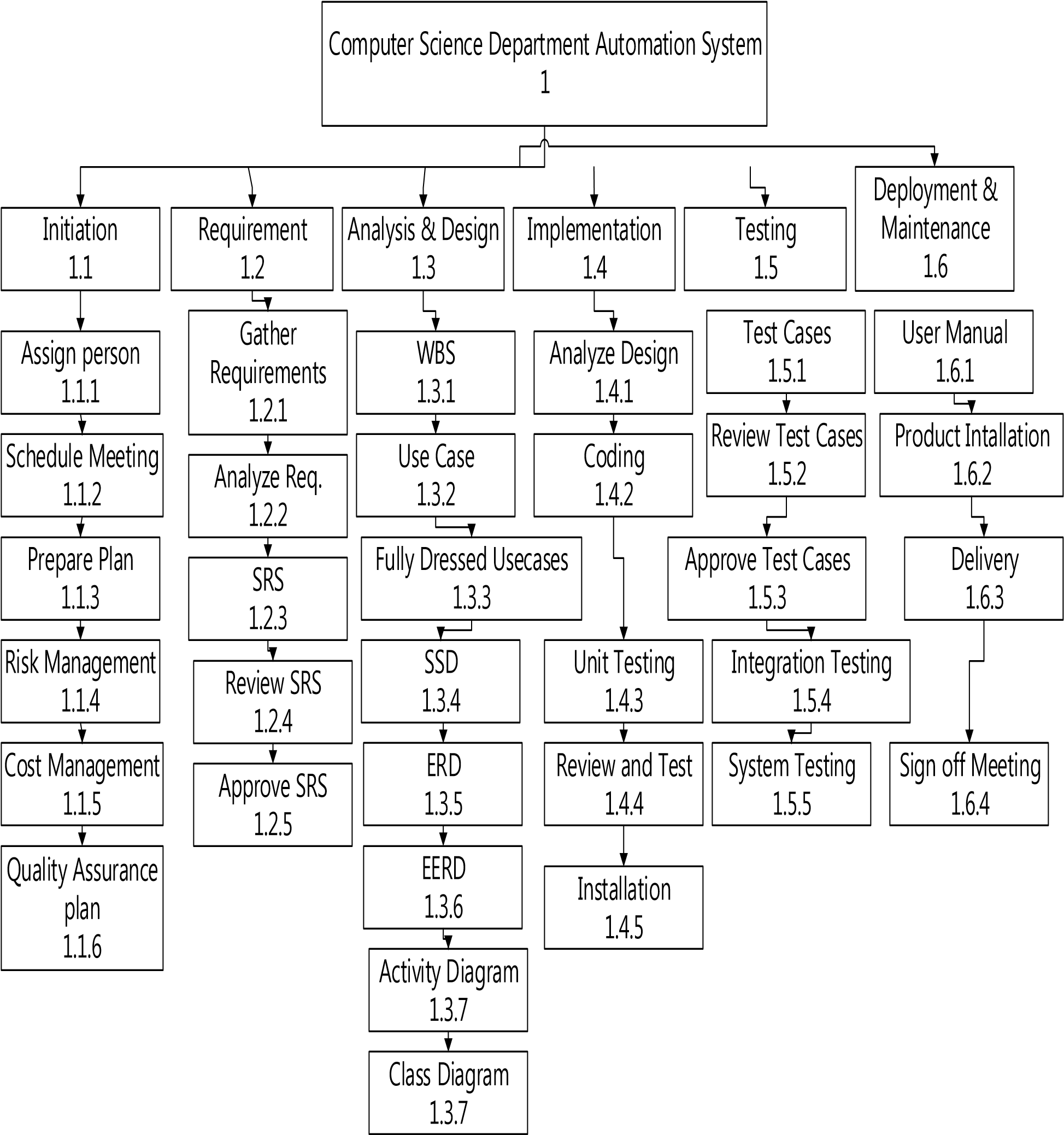
##### 4.5.2 Problem Resolution Plan

Team member meetings are arranged on regular basis. In these meetings all the problems related to project are discussed and their remedies are suggested.

##### 4.5.3 Process Improvement Plan

In all the group meetings, the problems and lessons learned from that problems will be discussed and careful plan will be established to improve the process pace. And try to avoid the repetition of mistakes been done in past.

#### 4.6 Work Breakdown Structure (WBS)



**Figure 4.2: Work Breakdown Structure**

**CHAPTER 5**

### IMPLEMENTATION

#### 5.1 ASP.NET

The Asp.Net is a widely used platform designed by Microsoft to build Web Applications. The Asp.Net Framework was firstly introduced in the year 2001. The platform is used widely around the globe by developers and a lot of API’s and plugin and support is available for Asp.Net.

##### 5.1.1 Developer productivity

The knowledge of the .Net framework itself can be a powerful tool for increasing a developer's productivity. Writing less code is a great technique for making your applications more easily maintainable. Your knowledge of the built-in functionality of the .NET framework is the key to effectively writing concise applications in ASP.NET. Ranging from simple concepts to details about Microsoft's ASP.NET MVC framework, or ASP.NET's Web Forms will get you up to speed in no time.

##### 5.1.2 Improved performance and scalability

ASP.NET is a very popular and a powerful framework that developers across the world prefer to use for application development. This platform is regularly updated with fresh and extended features provided by Microsoft that make it all the more advanced. This has allowed developers to create highly scalable and innovative web applications at a rapid pace. However, there are several ways in which the scalability and performance of web applications can be boosted. With the right blend of advanced features and simple tricks, programmers are able to build popular web applications.

##### 5.1.3 Enhanced reliability

Asp.Net applications are very reliable it is because the Framework has given developer authority to build stable applications which can tolerate exception and other Bugs and errors. This makes the application build using .Net Framework a very reliable and trustworthy. With the passage of time the Microsoft has developed it so well that we can say it is very much stable.

##### 5.1.4 Easy deployment

The deployment of the Asp.Net Application is much easier. Several times we encounter a great and sophisticated web or desktop application that does not poses its appropriate market share just for the reason of poorly written SETUP package and poorly designed deployment strategy. The fact is simple: If your users are not able to easily deploy your application then whatever sophistication or features you provide, they will be unable to even experience your application from the very first place. The Asp.Net previously provided a technique known as XCOPY development and later in Asp.Net 2.0 they also allowed to copy the website on the server or it can be deployed by Pre-Compilation method.

##### 5.1.5 System requirements for visual studio. NET

* 1.8 GHz or faster processor. Dual-core or better recommended
* 2 GB of RAM; 4 GB of RAM recommended (2.5 GB minimum if running on a virtual machine)
* Hard disk space: up to 130 GB of available space, depending on features installed; typical installations require 20-50 GB of free space.
* Hard disk speed: to improve performance, install Windows and Visual Studio on a solid state drive (SSD).
* Video card that supports a minimum display resolution of 720p (1280 by 720);

Visual Studio will work best at a resolution of WXGA (1366 by 768) or higher.

#### 5.2 Windows XP

##### 5.2.1 Easy to use

Microsoft Windows is the prime running Operating System in the World that’s why even every person is familiar with this OS. It is because it has provided users with the capacity to easily manage File and Folders and to easily install new application and run the existing one. While there are some OS where installation of new program and managing of files and folders are very difficult.

##### 5.2.2 Revolutionize the way remote users work

The Platform has revolutionized the way of working of a user remotely from anywhere around the globe and has enabled different user to communicate via a website etc. The Server with Microsoft windows are available to host websites and applications which will then be accessible by all users.

##### 5.2.3 Advanced management, deployment and support tools

One of the another advantage of using Microsoft windows Operating System is that it provides the various advanced management tools for deployment and supports. The Microsoft has also provided AZURE services to maintain host .Net applications and also provided various Libraries for working.

##### 5.2.4 System requirements for windows 10 professional

* Processor: 1 gigahertz (GHz) or more.
* RAM: 1 gigabyte (GB) for 32-bit or 2 GB for 64-bit.
* Hard disk space: 16 GB for 32-bit OS 20 GB for 64-bit OS.
* Graphics card: DirectX 9 or later with WDDM 1.0 driver.
* Display: 800x600

#### 5.3 Microsoft SQL Server 2014

##### 5.3.1 Standard features

SQL Server 2014 enables memory optimization of selected tables and stored procedures. The In-Memory OLTP engine is designed for high concurrency and uses a new optimistic concurrency control mechanism to eliminate locking delays.

SQL Server 2014 provides a new solid state disk (SSD) integration capability that lets you use SSDs to expand the SQL Server 2014 Buffer Pool as nonvolatile RAM. With the new Buffer Pool Extensions feature, you can use SSD drives to expand the buffer pool in systems that have maxed out their memory.

Buffer Pool Extensions can provide performance gains for read-heavy OLTP workloads. The Resource Governor lets you limit the amount of CPU and memory that a given workload can consume.

QL Server 2014 extends the reach of the Resource Governor to manage storage I/O usage as well. The SQL Server 2014 Resource Governor can limit the physical I/O’s issued for user threads in a given resource pool.

##### 5.3.2 System requirements for Microsoft SQL Server 2014

The following memory and processor requirements apply to all editions of SQL Server 2014.

|  |  |
| --- | --- |
| **Component** | **Requirement** |
| Memory[1] | **Minimum:**    Express Editions: 512 MB    All other editions: 1 GB    **Recommended:**    Express Editions: 1 GB    All other editions: At least 4 GB and should be increased as database size increases to ensure optimal performance. |
| Processor Speed | **Minimum:**    x86 Processor: 1.0 GHz    x64 Processor: 1.4 GHz    **Recommended:** 2.0 GHz or faster |
| Processor Type | x64 Processor: AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support    x86 Processor: Pentium III-compatible processor or faster |

### Chapter 6

|  |  |
| --- | --- |
| **Test case ID:** | TC-01 |
| **Test case Name:** | Process Registration |
| **Test case**  **Description:** | This test case describes that can an Administrator register into system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |

#### 6SOFTWARE TESTING

**Test case updated**

### Test case updated by: Badar Khalil 6th june,2019

**on:**

**Primary Actors:** <Administrator> **Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User will enter the user name, user password & CNIC |  |  |
|  |  | 2 | System will insert the user information from database |
|  |  | 3 | System process information and Confirmation message is shown |

**Testing Requirement:**

|  |  |
| --- | --- |
| **Test Condition:**  **Input Data:**  **Expected Result:**  **Actual Result:**  **Priority:**  **Frequency:**  **Acceptance:** | 2. User must have CNIC/Password/CNIC User Name, User Password, CNIC  Administrator successfully register into the system  Administrator successfully register into the system  High  Less Frequent in use  Pass |
|  |  |

1. System must be in running state

|  |  |
| --- | --- |
| **Test case ID:** | TC-02 |
| **Test case Name:** | Process Login |
| **Test case**  **Description:** | This test case describes that can an Administrator, Candidate & Teacher log into system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |

### Test case updated

### Test case updated by: Badar Khalil 6th june,2019

**on:**

**Primary Actors:** <Administrator><Candidate><Teacher>

### Main Success Scenario

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | User will enter the user name & user password |  |  |
|  |  | 2 | System will authenticate the user information from database |
|  |  | 3 | System process information and Confirmation message is shown |

**Testing Requirement:**

|  |  |
| --- | --- |
| **Test Condition:** | 2. User must have Login user name/user password |
| **Input Data:** | User name, Password |
| **Expected Result:** | Administrator, Candidate & Teacher successfully logged into the system |
| **Actual Result:** | Administrator, Candidate & Teacher successfully logged-in into the system |
| **Priority:** | High |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |

1. System must be in running state

|  |  |
| --- | --- |
| **Test case ID:** | TC-03 |
| **Test case Name:** | Add Course |
| **Test case**  **Description:** | This test case describes that can Administrator will Add a course into the system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2019 **on:** |
| **Primary Actors:** | <Administrator> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator selects Add Course |  |  |
|  |  | 2 | System will show form to the user to entry new course |
| 3 | Administrator will enter the all information of a new course |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Administrator must have right to add a course |
| **Input Data:** | Course ID, Course Name, Course Credit Hours, Semester of Study |
| **Expected Result:** | Administrator successfully add information into the system |
| **Actual Result:** | Administrator successfully added information into the system |

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-04 |
| **Test case Name:** | Search Course | |
| **Test case**  **Description:** | This test case describes that can Administrator, Candidate & Teacher will search a course from the system or not | |
| **Test case**  **Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th June,2018 **on:** | |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2018 **on:** | |

**Primary Actors:** <Administrator><Candidate><Teacher>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator, Candidate, Teacher search for course |  |  |
|  |  | 2 | System Process and show the Search course form to the User |
| 3 | Administrator, Candidate, Teacher will enter the course information |  |  |
|  |  | 4 | System will verify the information and course information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

**Test Condition:** 2. System must authenticate the user

3. Course must be present in data base before search **Input Data:** Course ID/Course Name

Administrator, Candidate, Teacher successfully search course **Expected Result:** information from the system

Administrator, Candidate, Teacher successfully searched course **Actual Result:**

information from the system

**Priority:** High

**Frequency:** Most Frequent in use

### Acceptance: Pass Test case ID: TC-05 Test case Name: Update Course

**Test case** This test case describes how an Administrator will Update a course **Description:**

### Test case Sana Aftab Abbasi Test case Prepared 5th june,2019

**Prepared by: on:**

**Test case updated** Badar Khalil **Test case updated** 6th june,2019 **by: on:**

### Primary Actors: <Administrator>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for course record |  |  |
|  |  | 2 | System process and course information is shown to user |
| 3 | Administrator will update the course information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Course must be present in data base before updation |
| **Input Data:** | Course Name, Course Credit Hours, Semester of Study, Course Content |
| **Expected Result:** | Administrator successfully update information in the system |
| **Actual Result:** | Administrator successfully updated information in the system |

### Priority: Less

|  |  |
| --- | --- |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-06 |
| **Test case Name:** | Delete Course |
| **Test case** | This test case describes can an Administrator will Delete a course or not |

**Description:**

Sana Aftab Abbasi

Badar Khalil th

**on:**

<Administrator>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for course record |  |  |
|  |  | 2 | System process and course information is shown to user |
| 3 | Administrator will delete the course information |  |  |
|  |  | 4 | System will process and delete the course information to system |
|  | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. User must have rights to Delete Course |
| **Input Data:** | Course ID |
| **Expected Result:** | Administrator successfully delete course |
| **Actual Result:** | Administrator successfully deleted course |

### Priority: Less

|  |  |
| --- | --- |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-07 |
| **Test case Name:** | Generate Course Report |
| **Test case**  **Description:** | This test case describes can an Administrator will Generates Report of a course or not |

Sana Aftab Abbasi

Badar Khalil th

**on:**

<Administrator>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator will select View course report |  |  |
|  |  | 2 | System Process and ask for Parameters |
| 3 | Administrator will selects the Parameters and submits |  |  |
|  |  | 4 | System process and Shows Item Report |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. User must have rights to Generate Reports |
| **Input Data:** | Semester Wise/Over All |
| **Expected Result:** | Administrator successfully generate report of course |
| **Actual Result:** | Administrator successfully generated report of course |

### Priority: Less

|  |  |
| --- | --- |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-08 |
| **Test case Name:** | Add Date Sheet |
| **Test case**  **Description:** | This test case describes that can a Administrator will Add a date sheet into the system or not |

Sana Aftab Abbasi

Badar Khalil th

**on:**

<Administrator>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator selects Add date sheet |  |  |
|  |  | 2 | System will show form to the user to entry date sheet |
| 3 | Administrator will enter the all information of a date sheet |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Administrator must have right to add a date sheet |
| **Input Data:** | Date Time Year, Semester, Date Sheet files |
| **Expected Result:** | Administrator successfully add information into the system |
| **Actual Result:** | Administrator successfully added information into the system |

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-09 |
| **Test case Name:** | Search Date Sheet |
| **Test case**  **Description:** | This test case describes that can a Administrator, Candidate & Teacher will search a date sheet from the system or not |

**Test case Prepared** Sana Aftab Abbasi **Test case Prepared** th June,2018 5 **by: on:**

**Test case updated** Badar Khalil **Test case updated** th june,20186 **by: on:**

**Primary Actors:** <Administrator><Candidate><Teacher>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator, Candidate, Teacher search for date sheet |  |  |
|  |  | 2 | System Process and show the Search date sheet form to the User |
| 3 | Administrator, Candidate, Teacher will enter the date sheet information |  |  |
|  |  | 4 | System will verify the information and date sheet information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

**Test Condition:** 2. System must authenticate the user

3. Date Sheet must be present in data base before search

**Input Data:** Semester Name

Administrator, Candidate, Teacher successfully search date sheet **Expected Result:** information from the system

Administrator, Candidate, Teacher successfully searched date sheet **Actual Result:**

information from the system

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-10 |
| **Test case Name:** | Update Date Sheet |
| **Test case**  **Description:** | This test case describes how an Administrator will Update a date sheet |
| **Test case**  **Prepared by:** | **Test case Prepared**  Sana Aftab Abbasi 5th june,2019  **on:** |

**Test case updated** Badar Khalil **Test case updated** th june,20196 **by: on:**

### Primary Actors: <Administrator>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for date sheet record |  |  |
|  |  | 2 | System process and date sheet information is shown to user |
| 3 | Administrator will update the date sheet information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Date Sheet must be present in data base before updation |
| **Input Data:** | Semester Name, Date Sheet File |
| **Expected Result:** | Administrator successfully update information in the system |
| **Actual Result:** | Administrator successfully updated information in the system |
| **Priority:** | Less |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Yes |
| **Test case ID:** | TC-11 |
| **Test case Name:** | Delete Date Sheet |
| **Test case**  **Description:** | This test case describes can an Administrator will Delete a date sheet or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2019  **on:**  <Administrator> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for date sheet record |  |  |
|  |  | 2 | System process and date sheet information is shown to user |
| 3 | Administrator will delete the date sheet information |  |  |
|  |  | 4 | System will process and delete the date sheet information to system |
|  | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. User must have rights to Delete date sheet |
| **Input Data:** | Semester name |
| **Expected Result:** | Administrator successfully delete date sheet |
| **Actual Result:** | Administrator successfully deleted date sheet |

### Priority: Less

|  |  |
| --- | --- |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-12 |
| **Test case Name:** | Add Time Table |
| **Test case**  **Description:** | This test case describes that can Administrator will Add a time table into the system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2019 **on:** |
| **Primary Actors:** | <Administrator> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator selects Add Time Table |  |  |
|  |  | 2 | System will show form to the user to entry time table |
| 3 | Administrator will enter the all information of a time table |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Administrator must have right to add a time table |
| **Input Data:** | Time Table ID, Semester, date |
| **Expected Result:** | Administrator successfully add information into the system |
| **Actual Result:** | Administrator successfully added information into the system |

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-13 |
| **Test case Name:** | Search Time Table |
| **Test case**  **Description:** | This test case describes that can Administrator, Candidate & Teacher will search a time table form the system or not |
| **Test case**  **Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th June,2018 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2018  **on:**  <Administrator><Candidate><Teacher> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator, Candidate, Teacher search for time table |  |  |
|  |  | 2 | System Process and show the Search time table form to the User |
| 3 | Administrator, Candidate, Teacher  will enter the time table information |  |  |
|  |  | 4 | System will verify the information and time table information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

**Test Condition:** 2. System must authenticate the user

3. Time Table must be present in data base before search **Input Data:** Semester

Administrator, Candidate, Teacher successfully search time table **Expected Result:** information from the system

Administrator, Candidate, Teacher successfully searched time table **Actual Result:**

information from the system

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-14 |
| **Test case Name:** | Update Time Table |
| **Test case**  **Description:** | This test case describes how an Administrator will Update a Time Table |
| **Test case**  **Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2019  **on:**  <Administrator> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for time table record |  |  |
|  |  | 2 | System process and time table information is shown to user |
| 3 | Administrator will update the new time table information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Time Table must be present in data base before updation |
| **Input Data:** | Semester, Date Time, Time Table File |
| **Expected Result:** | Administrator successfully update information in the system |
| **Actual Result:** | Administrator successfully updated information in the system |

### Priority: Less

|  |  |
| --- | --- |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-15 |
| **Test case Name:** | Add Attendance |
| **Test case**  **Description:** | This test case describes that can a Teacher will Add an attendance into  the system or not |

**Test case Prepared** Sana Aftab Abbasi **Test case Prepared** th june,2019 5 **by: on:**

**Test case updated** Badar Khalil **Test case updated** th june,20196 **by: on:**

<Teacher>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Teacher selects Add Attendance |  |  |
|  |  | 2 | System will show form to the user to entry attendance |
| 3 | Teacher will enter the all information of attendance |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Teacher must have right to add an attendance |
| **Input Data:** | Date, Semester, Section |
| **Expected Result:** | Teacher successfully add information into the system |
| **Actual Result:** | Teacher successfully added information into the system |

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-16 |
| **Test case Name:** | Search Attendance |
| **Test case**  **Description:** | This test case describes that can Administrator, Teacher will search an attendance from the system or not |
| **Test case**  **Prepared by:** | **Test case Prepared**  Sana Aftab Abbasi 5th June,2018  **on:** |
| **Test case updated by:** | **Test case updated**  Badar Khalil 6th june,2018  **on:** |
| **Primary Actors:** | <Administrator><Teacher> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator, Teacher search for attendance |  |  |
|  |  | 2 | System Process and show the Search attendance form to the User |
| 3 | Administrator, Teacher will enter the attendance information |  |  |
|  |  | 4 | System will verify the information and attendance information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

**Test Condition:** 2. System must authenticate the user

3. Attendance must be present in data base before search

**Input Data:** Semester, Section, Date

Administrator, Teacher successfully search attendance **Expected Result:** information from the system

Administrator, Teacher successfully searched attendance **Actual Result:**

information from the system

### Priority: High

|  |  |
| --- | --- |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-17 |
| **Test case Name:** | Update Attendance |
| **Test case**  **Description:** | This test case describes how a Teacher will Update an Attendance |
| **Test case**  **Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th june,2019 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated** 6th june,2019 **on:** |
| **Primary Actors:** | <Teacher> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Teacher search for attendance record |  |  |
|  |  | 2 | System process and attendance information is shown to user |
| 3 | Teacher will update the attendance information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Attendance must be present in data base before updation |
| **Input Data:** | Candidate Presence |
| **Expected Result:** | Teacher successfully update information in the system |
| **Actual Result:** | Teacher successfully updated information in the system |

### Priority: Less

**Frequency:** Less Frequent in use

Pass

**Acceptance:**

|  |  |
| --- | --- |
| **Test case ID:** | TC-19 |
| **Test case Name:** | Add NCEAC Profile |
| **Test case**  **Description:** | This test case describes that can a Teacher, Candidate will Add an NCEAC Profile into the system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared on:** 5th june,2019 |
| **Test case updated by:** | Badar Khalil **Test case updated on:** 6th june,2019 |
| **Primary Actors:** | <Teacher><Candidate> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Teacher, Candidate selects Add NCEAC  Profile |  |  |
|  |  | 2 | System will show form to the user to entry data |
| 3 | Teacher, Candidate will enter the all information |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Teacher, Candidate must have right to add an information |
| **Input Data:** | Teacher CNIC, Candidate Enrollment |
| **Expected Result:** | Teacher, Candidate successfully add information into the system |
| **Actual Result:** | Teacher, Candidate successfully added information into the system |
| **Priority:** | High |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |

|  |  |
| --- | --- |
| **Test case ID:** | TC-20 |
| **Test case Name:** | Search NCEAC Profile |
| **Test case**  **Description:** | This test case describes that can Administrator will search a NCEAC  profile from the system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th June,2018 **on:** |
| **Test case updated by:** | Badar Khalil **Test case updated on:** 6th june,2018 |
| **Primary Actors:** | <Administrator> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for NCEAC  Profile |  |  |
|  |  | 2 | System Process and show the Search NCEAC profile form to the User |
| 3 | Administrator will enter the NCEAC  profile  information |  |  |
|  |  | 4 | System will verify the information and NCEAC profile information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

**Test Condition:** 2. System must authenticate the user

3. NCEAC Profile must be present in data base before search **Input Data:** Teacher CNIC, Candidate MIS-ID

Administrator successfully search NCEAC Profile information from the **Expected Result:** system

Administrator successfully searched NCEAC Profile information from **Actual Result:**

the system **Priority:** High

**Frequency:** Most Frequent in use

### Acceptance: Pass Test case ID: TC-21

**Test case Name:** Update NCEAC Profile

**Test case** This test case describes how a Teacher, Candidate will Update NCEAC **Description:** Profile

**Test case Prepared** Sana Aftab Abbasi **Test case Prepared** th june,2019 5 **by: on:**

**Test case updated** Badar Khalil **Test case updated on:** 6th june,2019 **by:**

**Primary Actors:** <Teacher><Candidate>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Teacher, Candidate search for NCEAC Profile record |  |  |
|  |  | 2 | System process and NCEAC Profile information is shown to user |
| 3 | Teacher, Candidate will update the NCEAC Profile information |  |  |
|  |  | 4 | System will process and update the new information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. NCEAC Profile must be present in data base before updation |
| **Input Data:** | Name, Address, Phone Number |
| **Expected Result:** | Teacher, Candidate successfully update information in the system |
| **Actual Result:** | Teacher, Candidate successfully updated information in the system |
| **Priority:** | Less |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |

|  |  |
| --- | --- |
| **Test case ID:** | TC-23 |
| **Test case Name:** | Add Feedback |
| **Test case**  **Description:** | This test case describes that can a Candidate will Add feedback into the system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared on:** 5th june,2019 |
| **Test case updated by:** | Badar Khalil **Test case updated on:** 6th june,2019 |
| **Primary Actors:** | <Candidate> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Candidate selects Add Feedback |  |  |
|  |  | 2 | System will show form to the user to entry feedback |
| 3 | Candidate will enter the feedback |  |  |
|  |  | 4 | System will insert the feedback to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

|  |  |  |
| --- | --- | --- |
| **Test Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Candidate must have right to add feedback | |
| **Input Data:** | Course Name, Teacher Name | |
| **Expected Result:** | Candidate successfully add feedback into the system | |
| **Actual Result:** | Candidate successfully added feedback into the system | |
| **Priority:** | High | |
| **Frequency:** | Most Frequent in use | |
| **Acceptance:** | Pass | |
| **Test case ID:** | | TC-24 | |
| **Test case Name:** | | View Feedback | |
| **Test case**  **Description:** | | This test case describes that can Administrator will view a feedback from the system or not | |
| **Test case Prepared by:** | | Sana Aftab Abbasi **Test case Prepared on:** 5th June,2018 | |
| **Test case updated by:** | | Badar Khalil **Test case updated on:** 6th june,2018 | |
| **Primary Actors:** | | <Administrator> | |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for feedback |  |  |
|  |  | 2 | System Process and show the Search feedback form to the User |
| 3 | Administrator will enter the feedback information |  |  |
|  |  | 4 | System will verify the information and feedback information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

|  |  |  |
| --- | --- | --- |
| **Test Condition:** | 1. System must be in running state 2. System must authenticate the user 3. Feedback must be present in data base before search | |
| **Input Data:** | Teacher Name, Course Name | |
| **Expected Result:** | Administrator successfully view feedback information from the system | |
| **Actual Result:** | Administrator successfully viewed feedback information from the system | |
| **Priority:** | High | |
| **Frequency:** | Most Frequent in use | |
| **Acceptance:** | Pass | |
| **Test case ID:** | TC-25 | |
| **Test case Name:** | Add News and Notification |
| **Test case**  **Description:** | This test case describes that can Administrator will Add News and Notification into the system or not |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared on:** 5th june,2019 |
| **Test case updated by:** | Badar Khalil **Test case updated on:** 6th june,2019 |
| **Primary Actors:** | <Administrator> |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator selects Add News and  Notification |  |  |
|  |  | 2 | System will show form to the user to entry News and Notification |
| 3 | Administrator will enter the all information of News and Notification |  |  |
|  |  | 4 | System will insert all the information to the system |
| 5 | Step 1-4 repeats if Administrator repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

|  |  |
| --- | --- |
| **Test Condition:** | 1. System must authenticate the user 2. Administrator must have right to add News and Notification |
| **Input Data:** | Title, Description Date, Time |
| **Expected Result:** | Administrator successfully add information into the system |
| **Actual Result:** | Administrator successfully added information into the system |
| **Priority:** | High |
| **Frequency:** | Most Frequent in use |
| **Acceptance:** | Pass |
| **Test case ID:** | TC-26 |
| **Test case Name:** | Search News and Notification | |
| **Test case**  **Description:** | This test case describes that can Administrator, Candidate & Teacher will search a course from the system or not | |
| **Test case Prepared by:** | Sana Aftab Abbasi **Test case Prepared** 5th June,2018 **on:** | |
| **Test case updated by:** | Badar Khalil **Test case updated on:** 6th june,2018 | |
| **Primary Actors:** | <Administrator><Candidate><Teacher> | |

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator, Candidate, Teacher search for News and Notification |  |  |
|  |  | 2 | System Process and show the Search News and Notification form to the user |
| 3 | Administrator, Candidate, Teacher will enter the News and Notification information |  |  |
|  |  | 4 | System will verify the information and course information will be shown to the user |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state

**Test Condition:** 2. System must authenticate the user

3. News and Notification must be present in data base before search

**Input Data:** Title, Description date

Administrator, Candidate, Teacher successfully search News and **Expected Result:**

Notification information from the system

Administrator, Candidate, Teacher successfully searched News and **Actual Result:**

Notification information from the system **Priority:** High

**Frequency:** Most Frequent in use

### Acceptance: Pass Test case ID: TC-27

**Test case Name:** Update News and Notification

**Test case** This test case describes how Administrator will Update News and **Description:** Notification

|  |  |  |
| --- | --- | --- |
| **Test case Prepared by:** | Sana Aftab Abbasi | **Test case Prepared** 5th june,2019 **on:** |
| **Test case updated** | Badar Khalil | **Test case updated on:** 6th june,2019 |

**by:**

### Primary Actors: <Administrator>

**Main Success Scenario:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **User Action** |  | **System Response** |
| 1 | Administrator search for News and Notification record |  |  |
|  |  | 2 | System process and News and Notification information is shown to user |
| 3 | Administrator will update the News and Notification information |  |  |
|  |  | 4 | System will process and update the News and Notification information to system |
| 5 | Step 1-4 repeats if user repeat the process |  |  |

**Testing Requirement:**

1. System must be in running state
2. System must authenticate the user

|  |  |
| --- | --- |
| **Test Condition:** | 3. News and Notification must be present in data base before  updation |
|  | Title, Description Date, Time |
| **Expected Result:** | Administrator successfully update News and Notification in the system |
| **Actual Result:** | Administrator successfully updated News and Notification in the system |
| **Priority:** | Less |
| **Frequency:** | Less Frequent in use |
| **Acceptance:** | Pass |

**Chapter 7**

### 7CONCLUSIONS & FUTURE WORK

#### 7.1 Achievements

I feel very proud after development and implementation of my final project successfully. Before developing through this project although I have a good theoretical knowledge of software engineering, being candidates of computer science, but it is far away from theory to develop a real life system that completely fulfill the user requirements. I had a very good interaction with my internal supervisor. During the development of the project I have achieved the following benefits.

* Project management and scheduling
* How to interact with the user
* System analysis and data collection
* A good knowledge of system designing and modeling
* Latest tools and technologies
* Testing strategies
* User documentation
* During the modeling phase I learnt  UML Diagrams

During testing and implementation finding a bug and its fixture was totally new and healthy experience. One of the very important aspects of the project was how to document our project properly. I have learnt new tools ASP, JS, SQL Server 2014, Adobe Illustrator, and Gif Animator which will help me a long way as I will be starting my careers as IT professional.

#### 7.2 Limitations

* System is built by keeping in view the aspect of scalability, it is able to take any future change and work efficiently.
* The system lacks synchronization which can be implemented later.

**Chapter 8**

### USER GUIDE (as template)

#### 8.1 User Guide

Welcome to the CS Society Automation System! This software is used to digitally create, collect, store, manipulate, and relay information needed for accomplishing basic tasks like collecting Data of candidates and teachers required for NCEAC accreditation. This system connects the university administration with the teachers (specifically with visiting faculty) and the candidates and will help the admin to inform them about the Data sheet, Timetable, news and Notifications.

The system also feature a reliable way for administration to generate reports of the profile of the teachers and the candidates in PDF format. The system will also have a web based panel for administration and two front end Android Application for the teachers and the candidates. We are sure you feel ease to use this system.

##### 8.1.1 Computer Requirements

Before entering the CS Society Automation System, make sure your computer and cellphone meets the following requirements:

* Internet Connection
* Window XP or higher
* RAM 2GB
* Hard disk Space maximum 500MB
* Processor 2GHz
* One of the following latest internet browsers recommended: Microsoft Internet Explorer, or Mozilla Firefox, Opera

##### 8.1.2 Home Page

The main area of the CS Society Automation System home page are described in detail on the following pages.

* Header

CS Society Automation System Title with

* + News and Notification bar
  + Home navigation
  + About navigation
  + Testimonials navigation
  + Member navigation
  + Contact navigation
  + Upcoming Events
  + Our Management Teams
  + Login navigation
  + Registration navigation

* Appraisals
* First appraisals section is below header section  Information about latest events

### Top headers

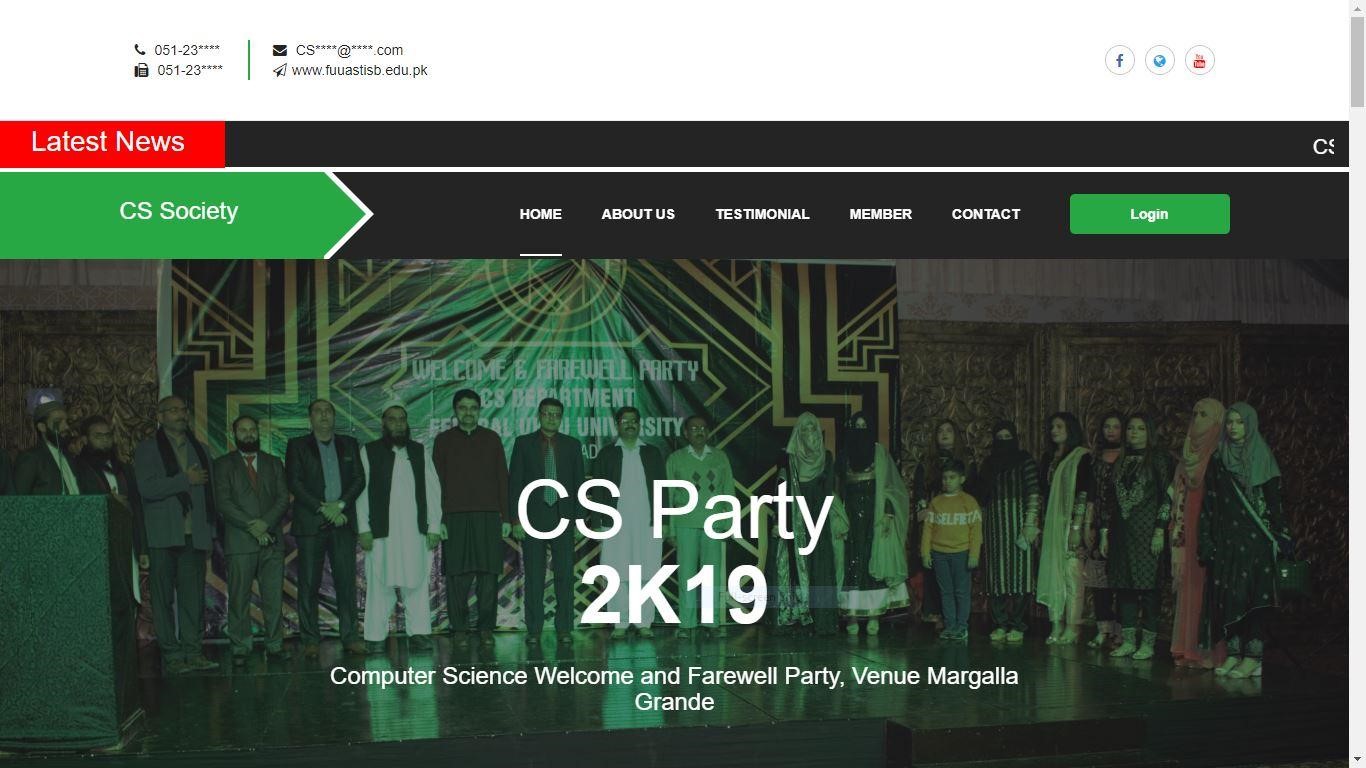


Figure 8.1 shows the appraisal home page

* Slider can show any upcoming event
* Show advertisements, and usual notifications
* Show pictures of past events

### About us



Figure 8.2 Shows the about section

CS department of Federal Urdu University Islamabad is established since a long time providing quality education to the candidates with best teachers in office, about section tells about CS department in brief.

* About us section show the total number of enrolled candidates
* Shows the total number of Alumni, faculty
* Also shows number of lab

#### Testimonials

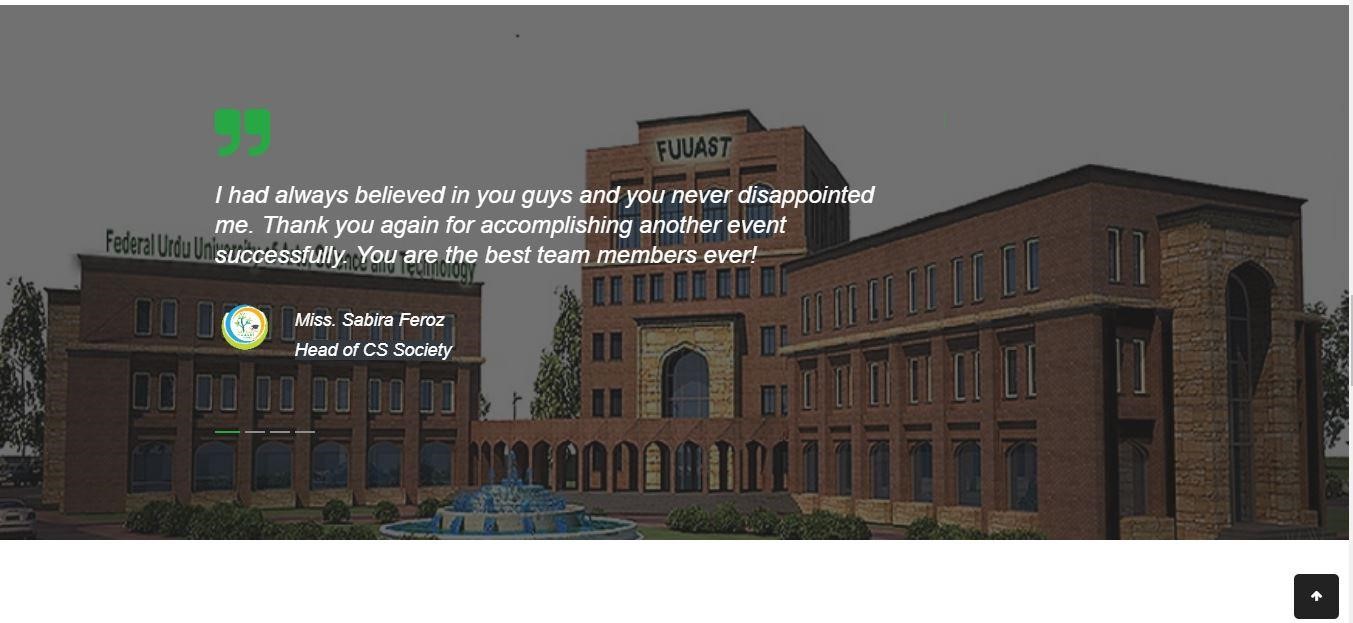


Figure 8.3 shows the testimonials of members

Testimonials section present reviews and feedbacks experience of the official of private and government entities officials about society’s performance and improvements.

#### News and Notification bar



Figure 8.4 shows the latest news and notification

* News and Notification can display on this bar
* CS candidate can remain up to date
* Download File

This bar can be used to display any upcoming events, news or any other information. This bar can also help user to download attached files. News related to official notification of the university will also be displayed here. CS candidates can remain up to date by views only breaking news section of the website.

### Members of CS Society

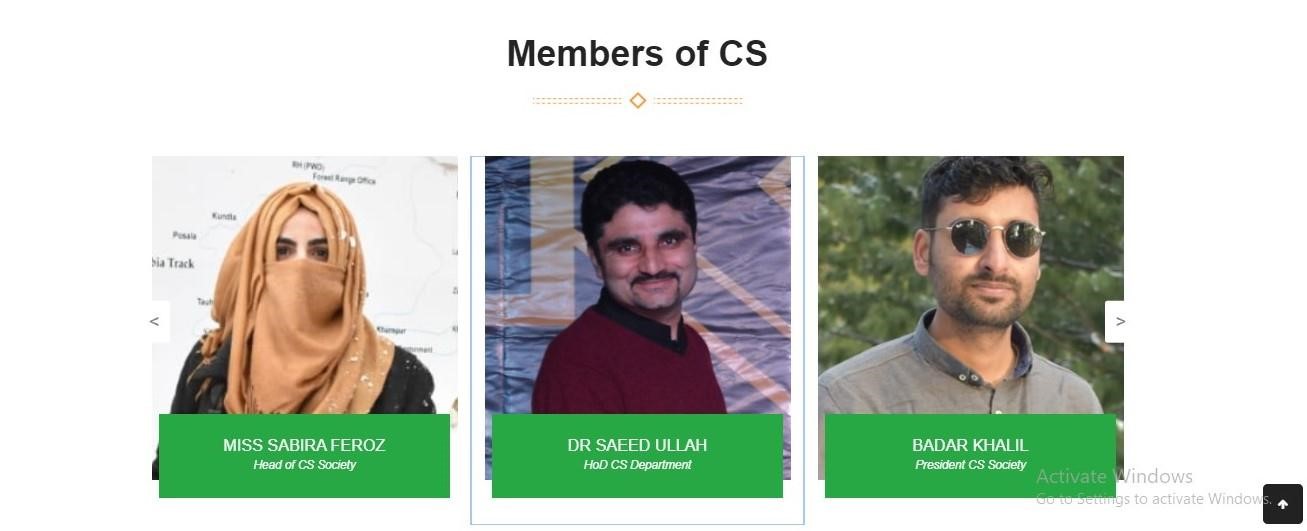


Figure 8.5 shows the members of CS society

CS society has specific ranks assigned to teachers and candidates

* Candidates can know about the officials of society from website
* Candidates can also contact respective officials in case of any queries through their social media accounts
* Head, President and Vice president of society can easily be approached through this website

### Upcoming Events

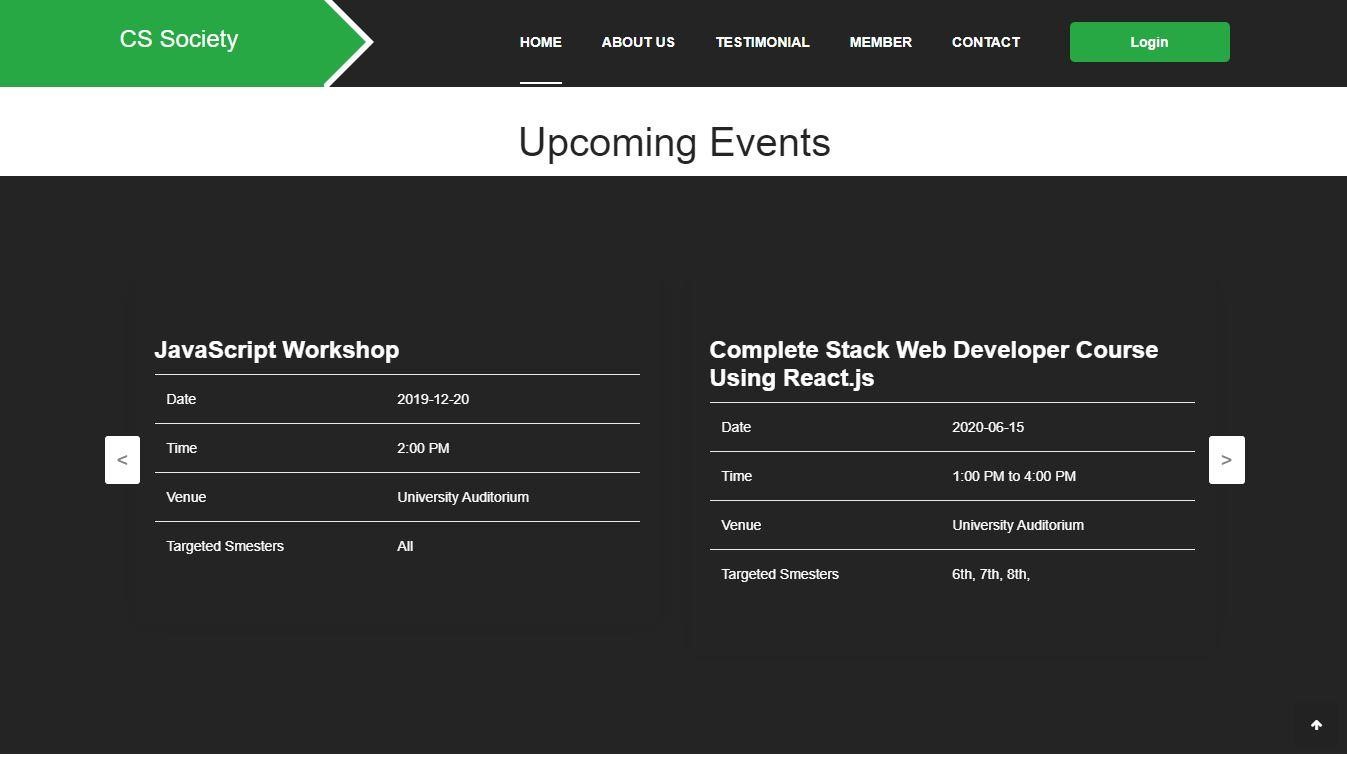


Figure 8.6 shows upcoming events of CS society

* Arranging and managing events are one of key responsibilities of CS society
* User can easily watch upcoming events, titles, dates and other information regarding those events
* All the detail about events can be displayed online using this sit

#### Our Management Team

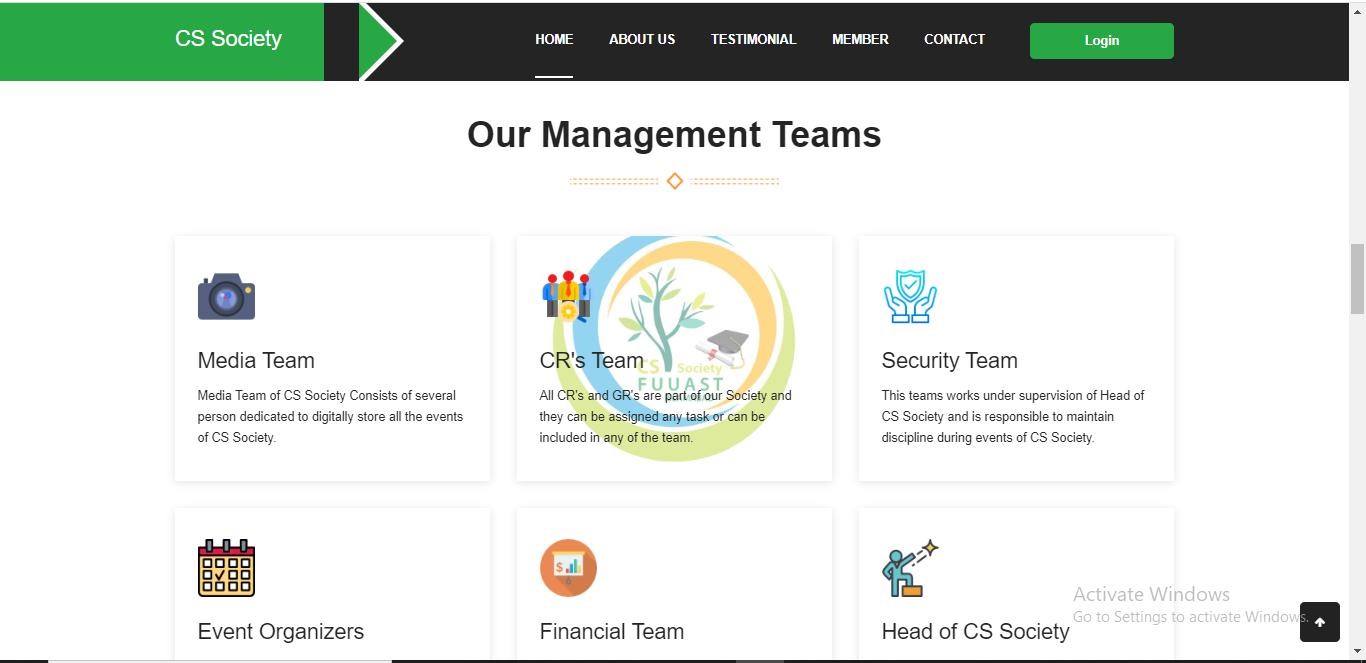


Figure 8.7 shows management teams of CS society

* Our management teams are; media team, security team, CR’s team, event organizers, financial team and head of CS society
* Media team consists of several person dedicated to digitally store all the events of CS society
* All the CR’s and GR’s are the part of our society and they can be assigned any task
* Evet organizers contain persons responsible to organize new events under supervision of CS society
* Security team works under supervision of head of CS society
* Financial team is responsible to deal financial related matters of all the events  Head of CS society has responsibility to coordinate with all teams

#### Contact Us

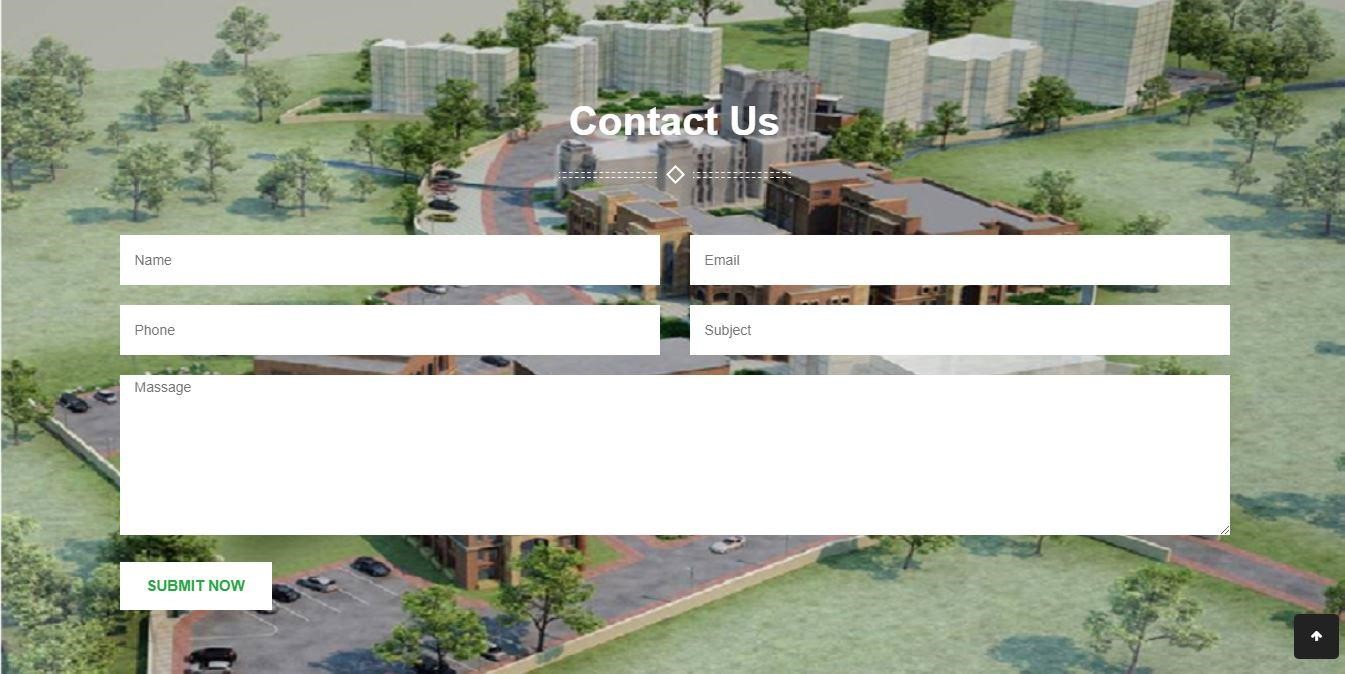


Figure 8.8 shows contact information required to be entered

* Contact forms are displayed so that user can contact admin
* If they found anything improper on the web pages
* Contact form are also helpful for new candidates for getting information about the CS department
* Enrolled candidates can also be benefitted from contact forms

**8.1.3 Login**

### Login Screen



Figure 8.9 show admin login page

* Admin and CR can login through this Admin Panel Login
* Admin panel helps DEO to handle candidates, faculty and departments data

**8.1.4 Dashboard**

### Dashboard

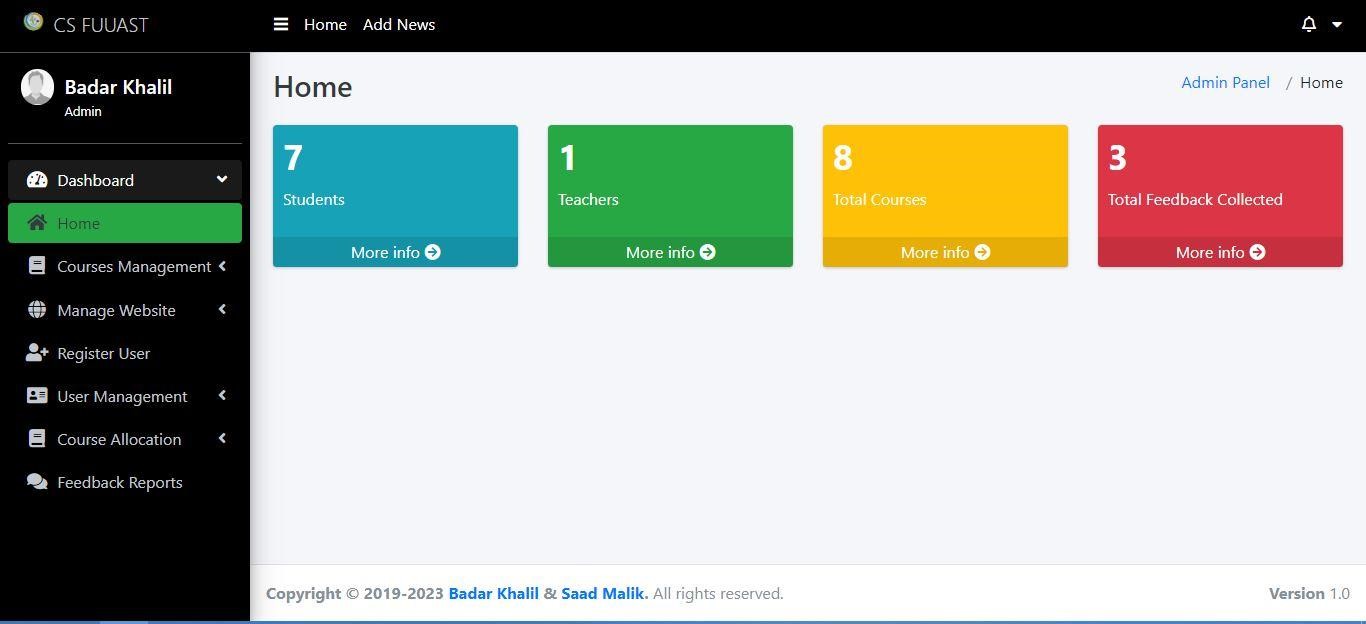


Figure 8.10 show admin dashboard

**8.1.5 Add Course**

### Add Courses

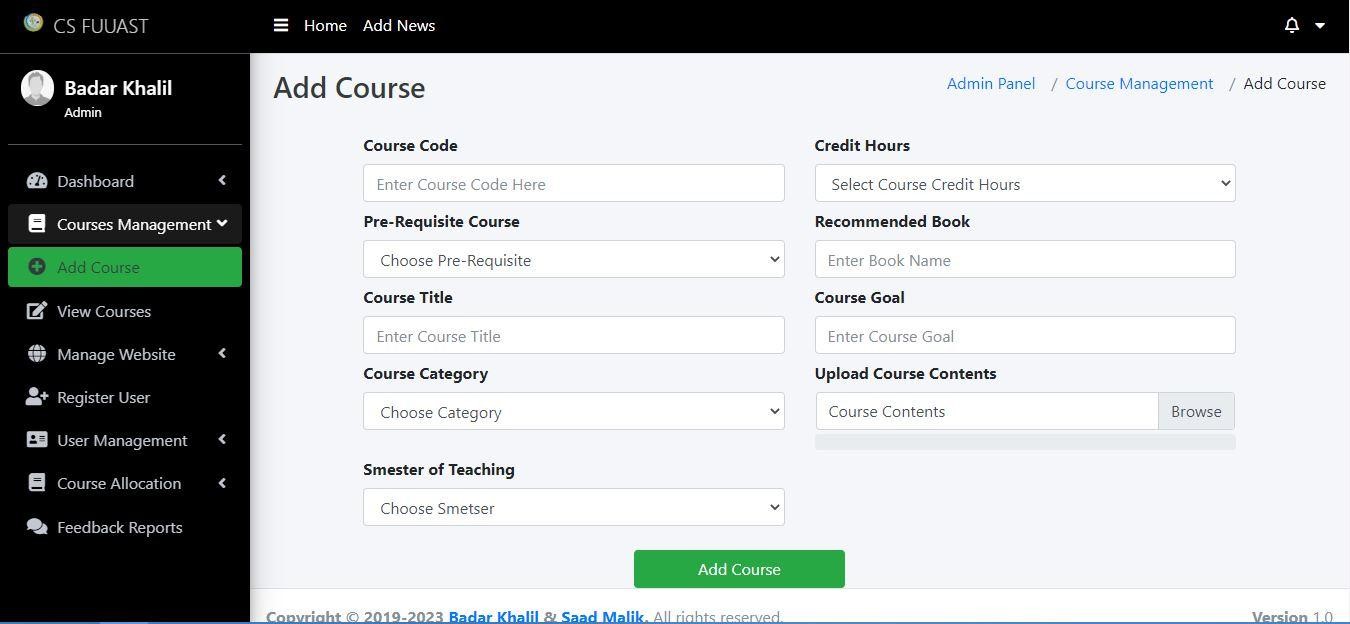


Figure 8.11 show how admin will add courses

**8.1.6 View Course**

### View Courses

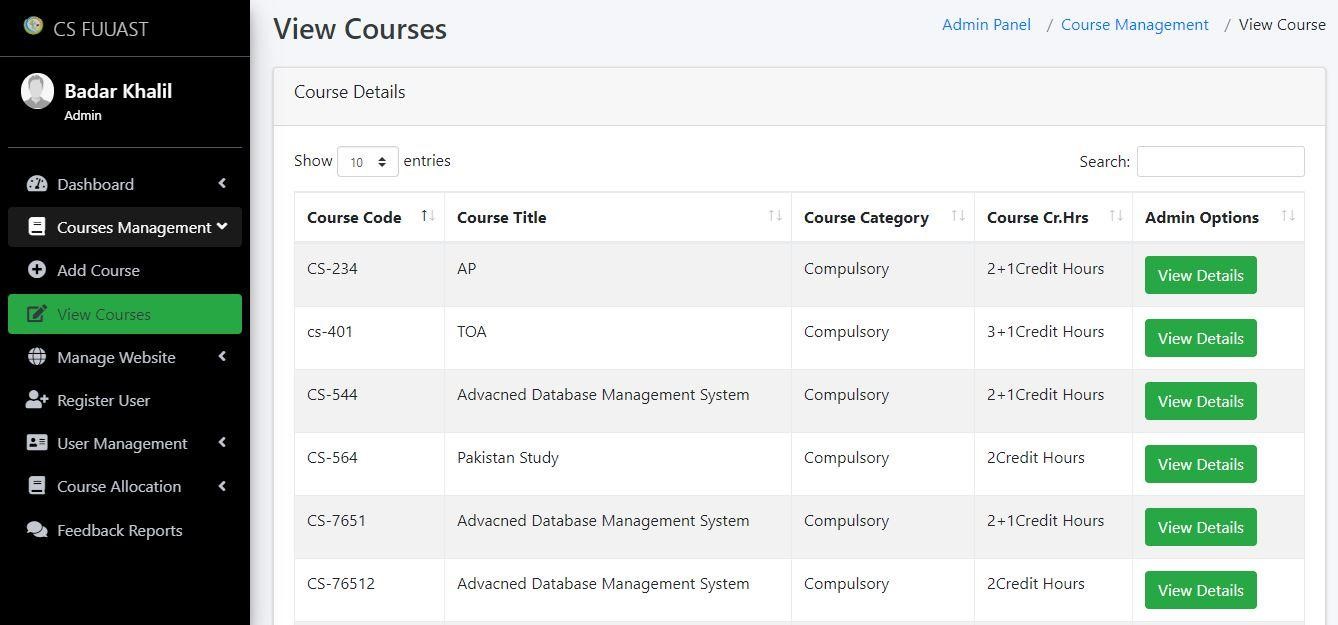


Figure 8.12 show how admins views and edit options of courses

**8.1.7 Add News and Notifications**

### Add News and Notifications

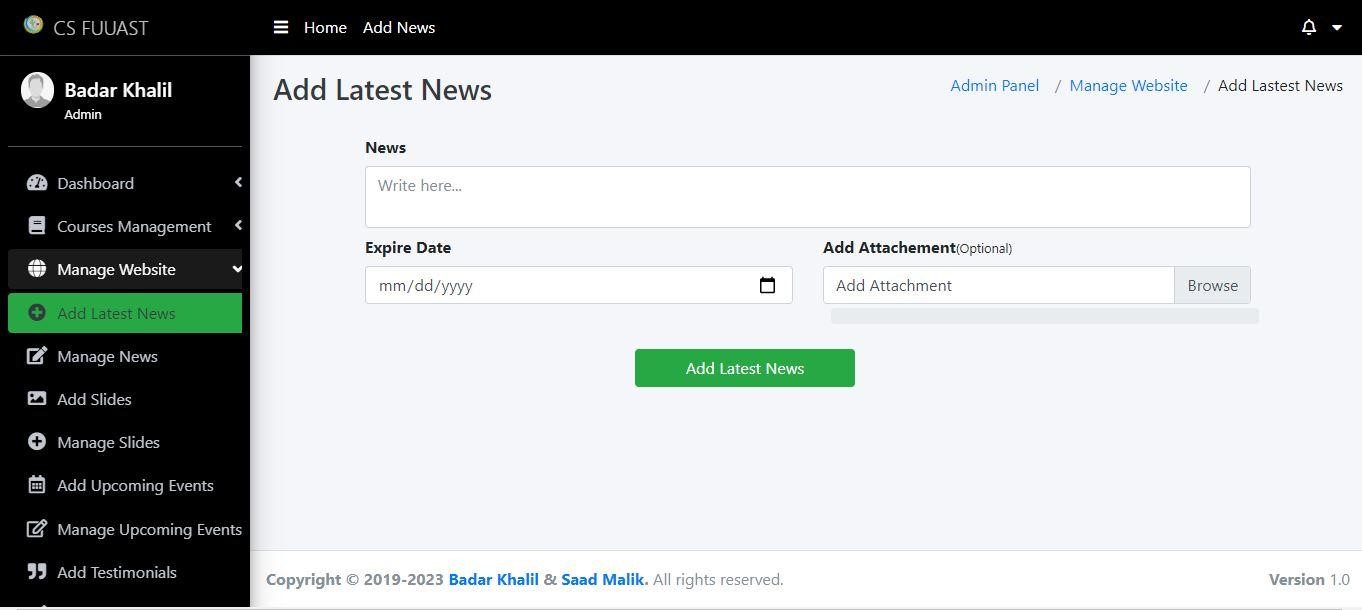


Figure 8.13 show how admin will add news

**8.1.8 Manage News and Notifications**

### Manage News and Notifications

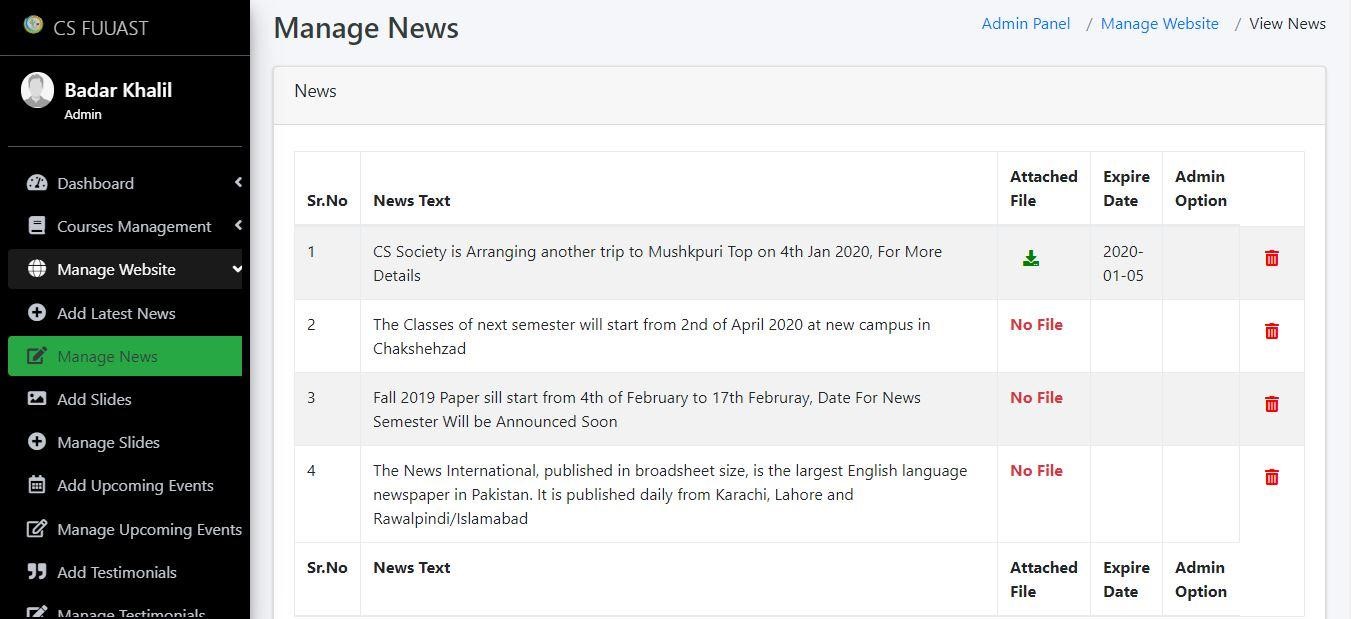


Figure 8.14 show how admin will manage news

**8.1.9 Add Slides**

### Add Slides

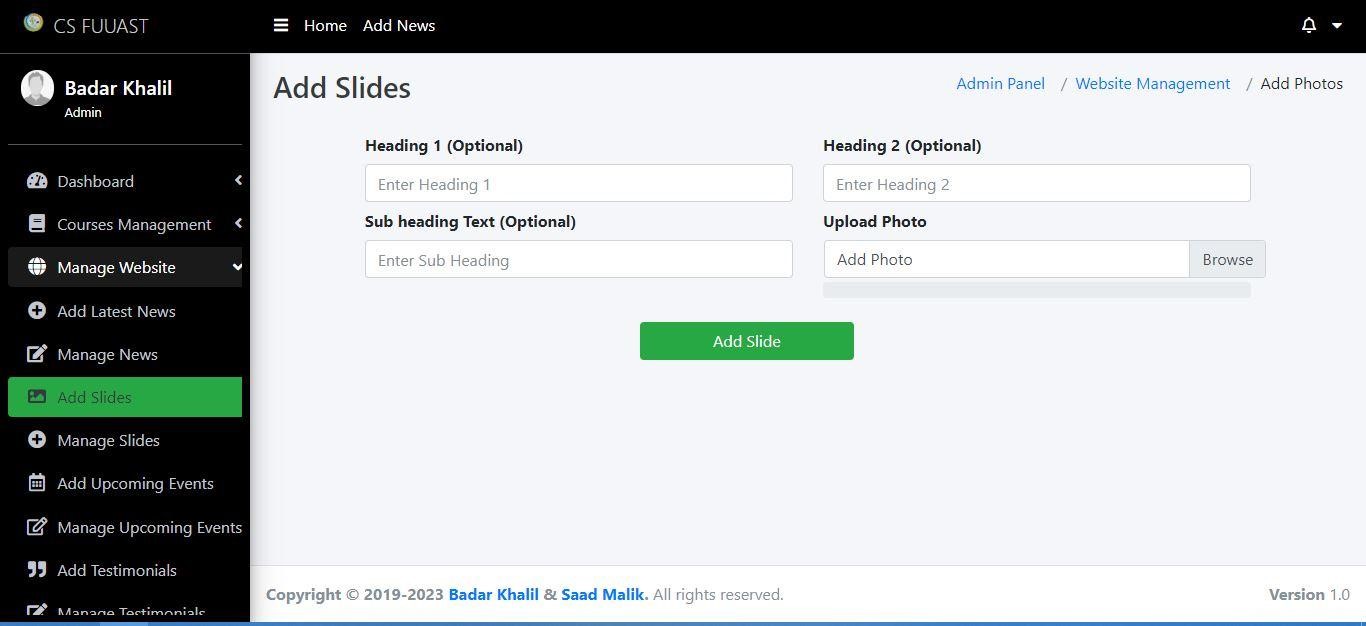


Figure 8.15 show how admin will add slider images

**8.1.10 Manage Slides**

### Manage Slides

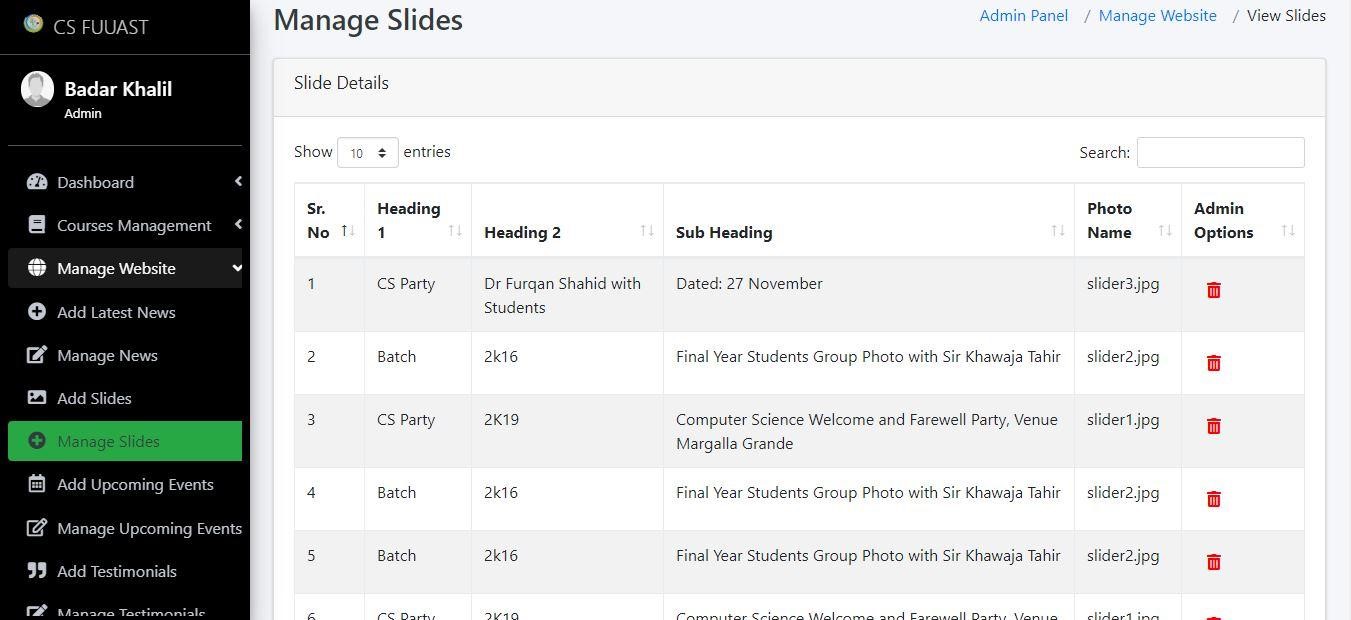


Figure 8.16 show how admin will manage slider images

**8.1.11 Add Upcoming Events**

### Add Upcoming Events

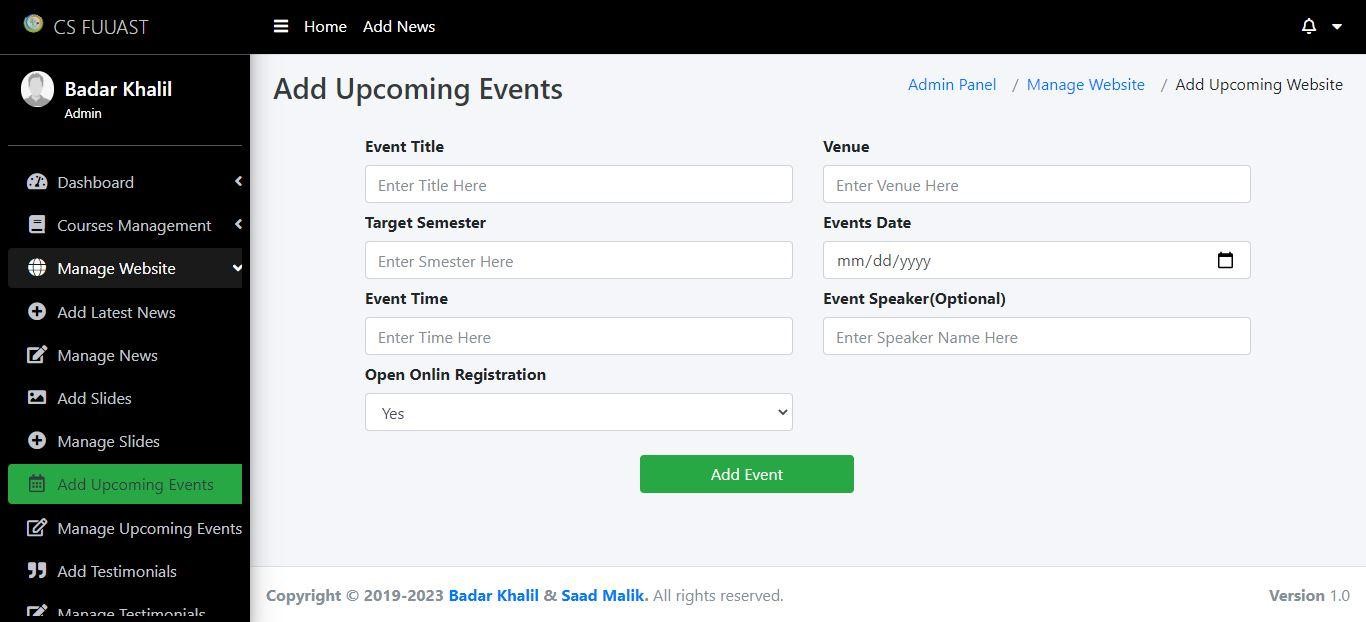


Figure 8.17 show how admin will add upcoming events

**8.1.12 Manage Upcoming Events**

### Manage Upcoming Events

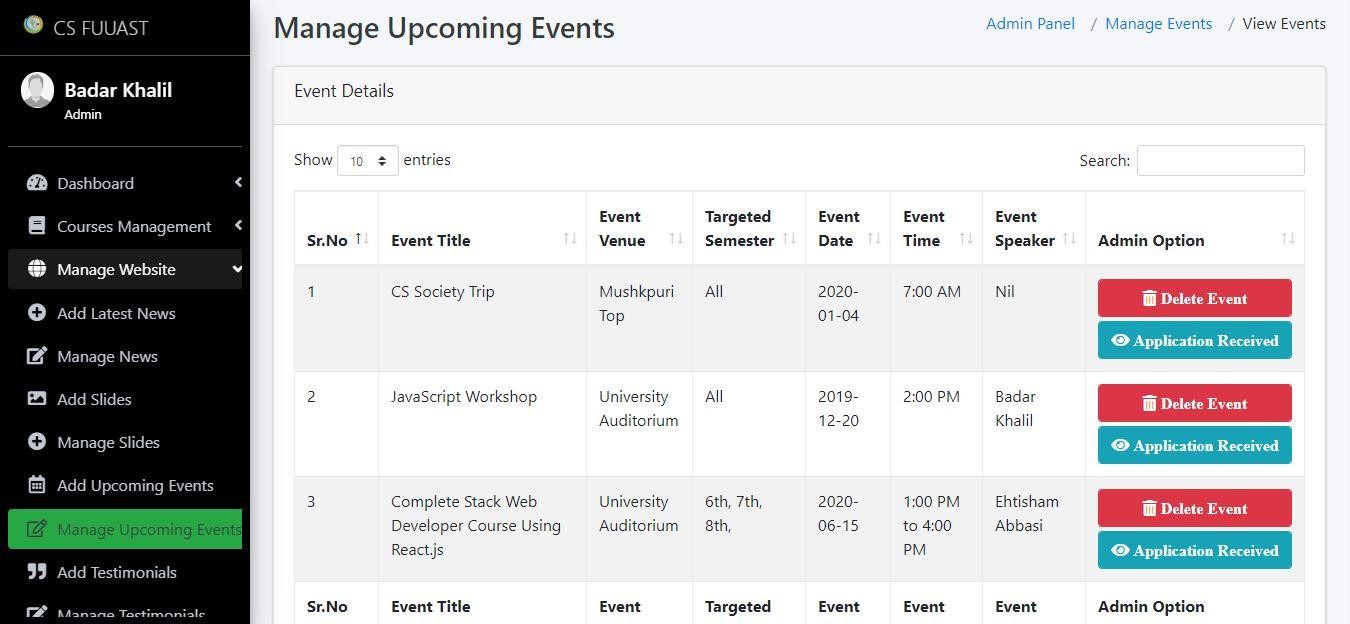


Figure 8.17 show how admin will manage upcoming events

**8.1.13 Add Testimonials**

### Add Testimonial



Figure 8.18 show how admin will add testimonial

**8.1.13 Manage Testimonials**

### Manage Testimonial

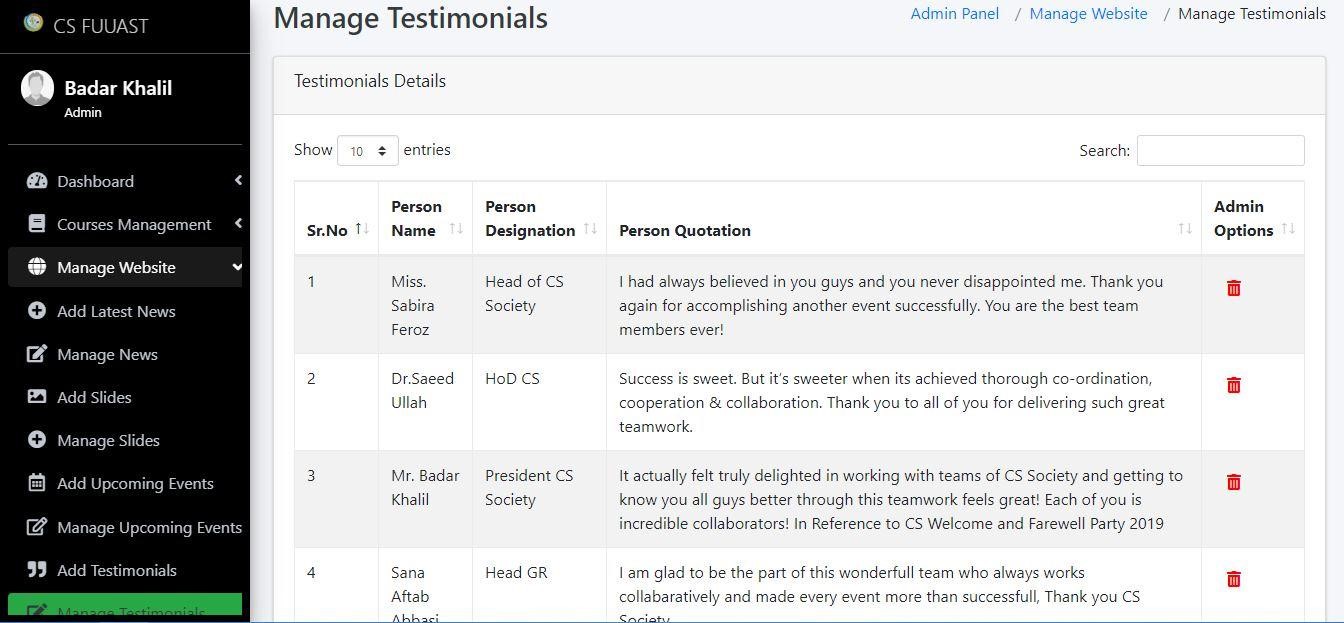


Figure 8.19 show how admin will manage testimonial

**8.1.14 Add Teams**

### Add Teams

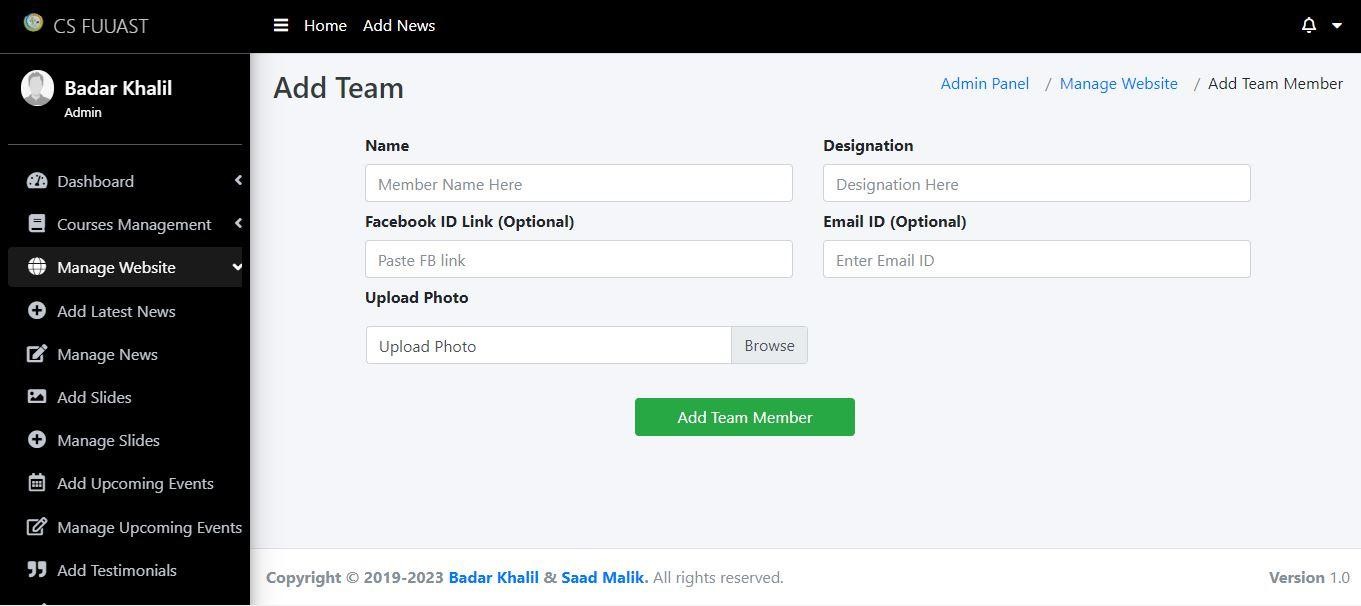


Figure 8.20 show how admin will add teams

**8.1.15 Manage Teams**

### Manage Teams

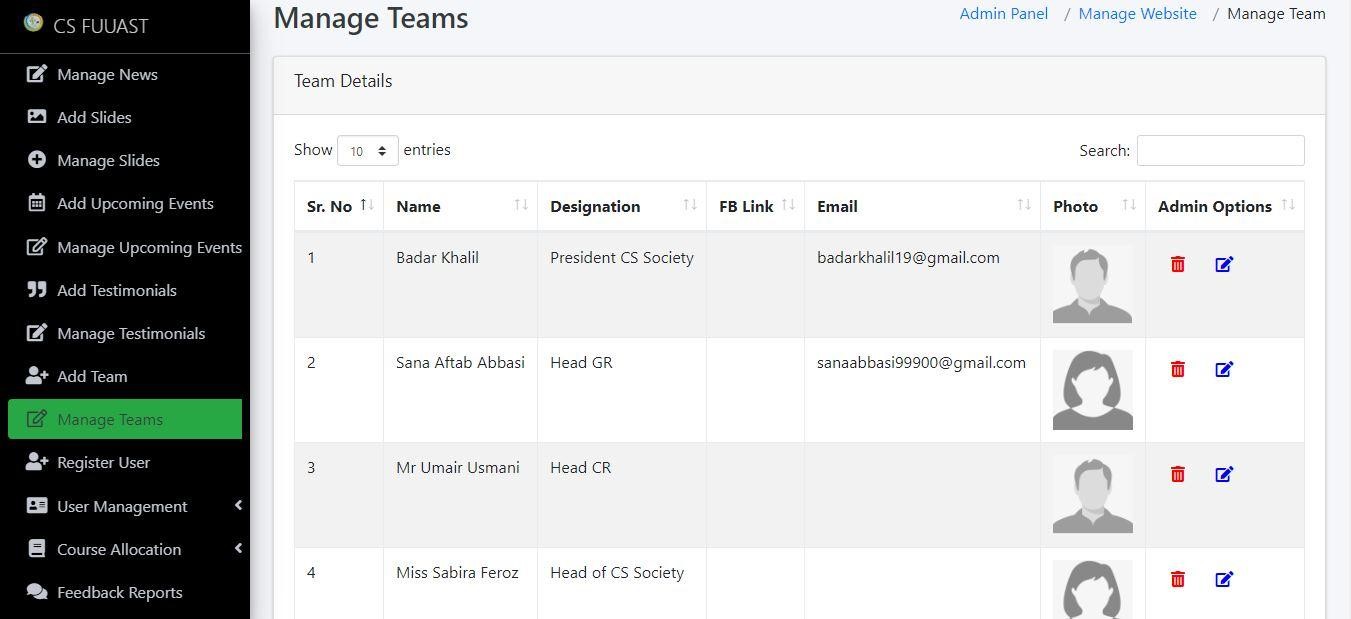


Figure 8.21 show how admin will manage teams

**8.1.16 Register User**

### Register User

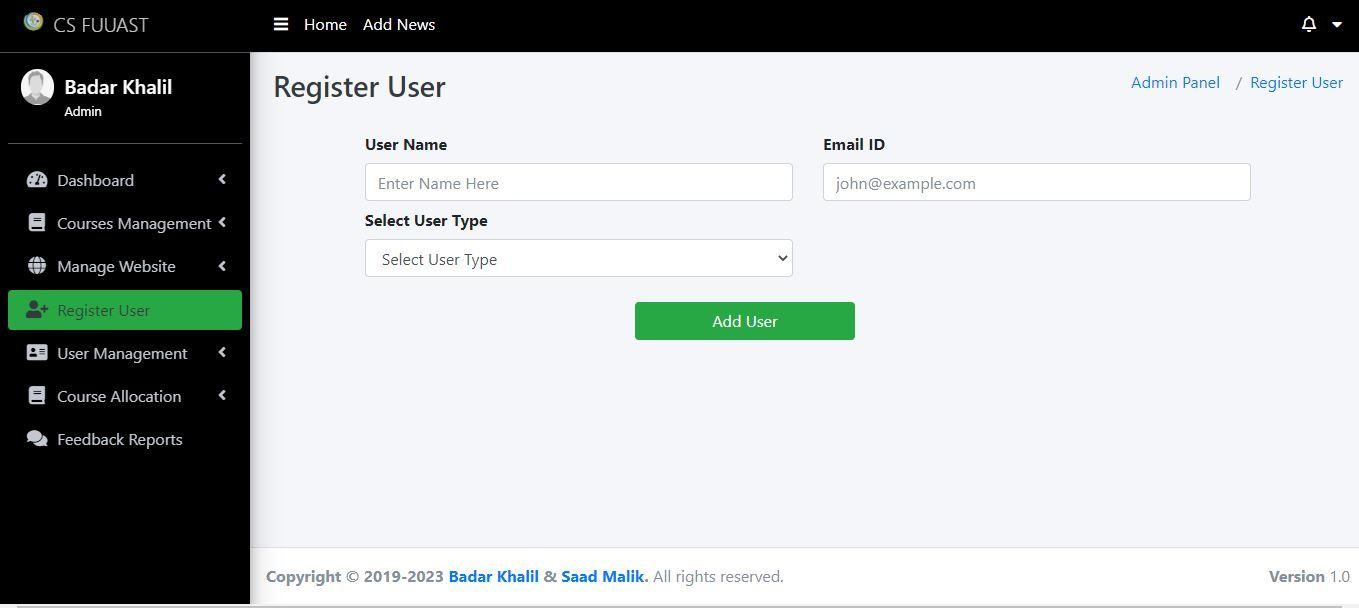


Figure 8.22 show how candidates and teacher will be added into system

**8.1.17 View User**

### View User

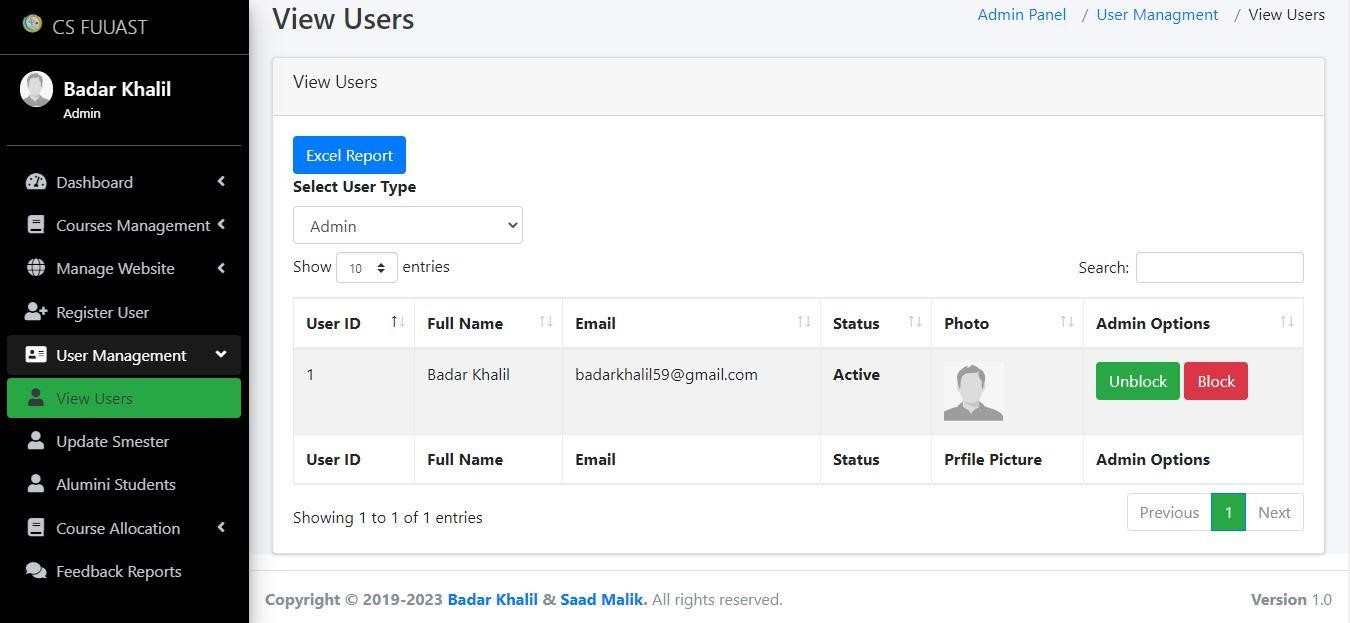


Figure 8.23 show user will be viewed



Figure 8.24 show user will be viewed

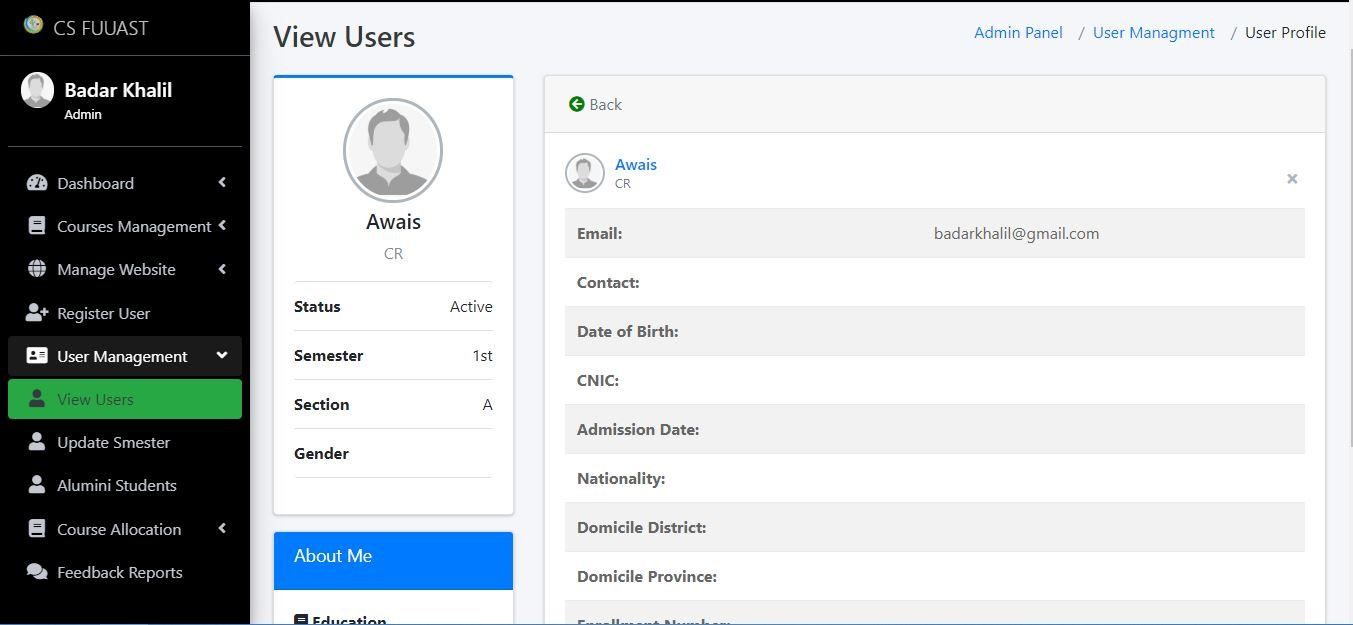


Figure 8.25 show user will profile

**8.1.18 Update Semester**

### Update Semester

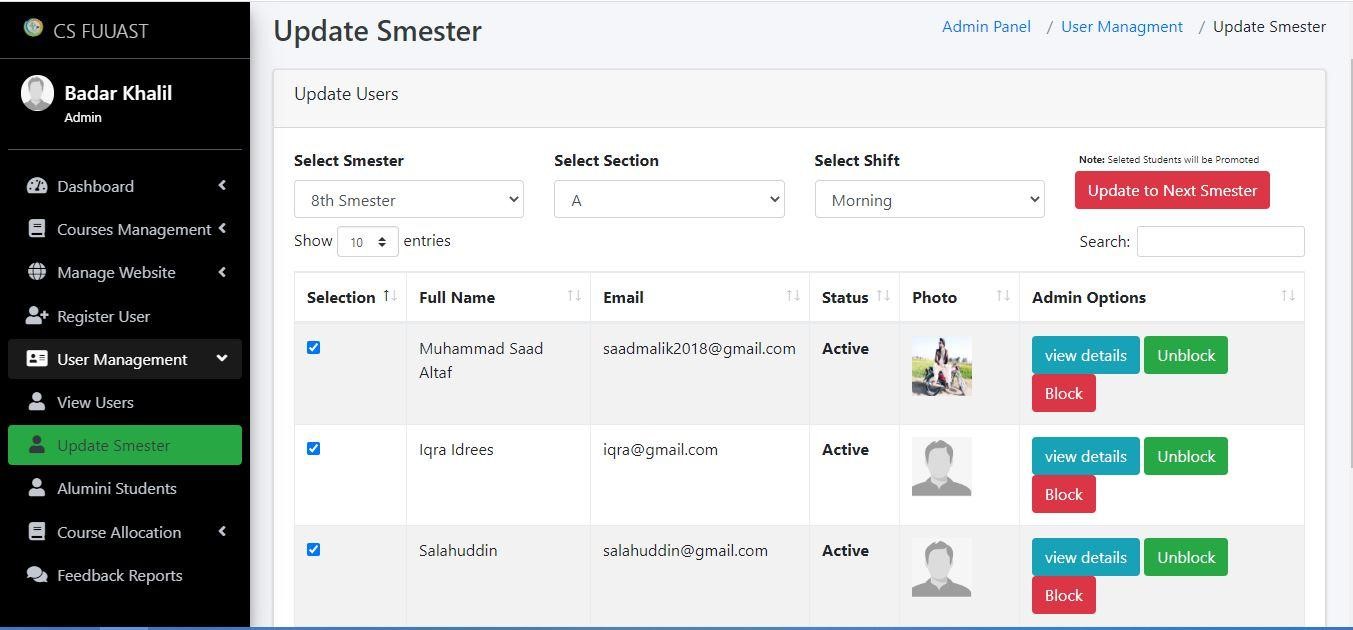


Figure 8.26 show how user promoted to next semester

**8.1.19 Alumni Candidate**

### Alumni Candidate

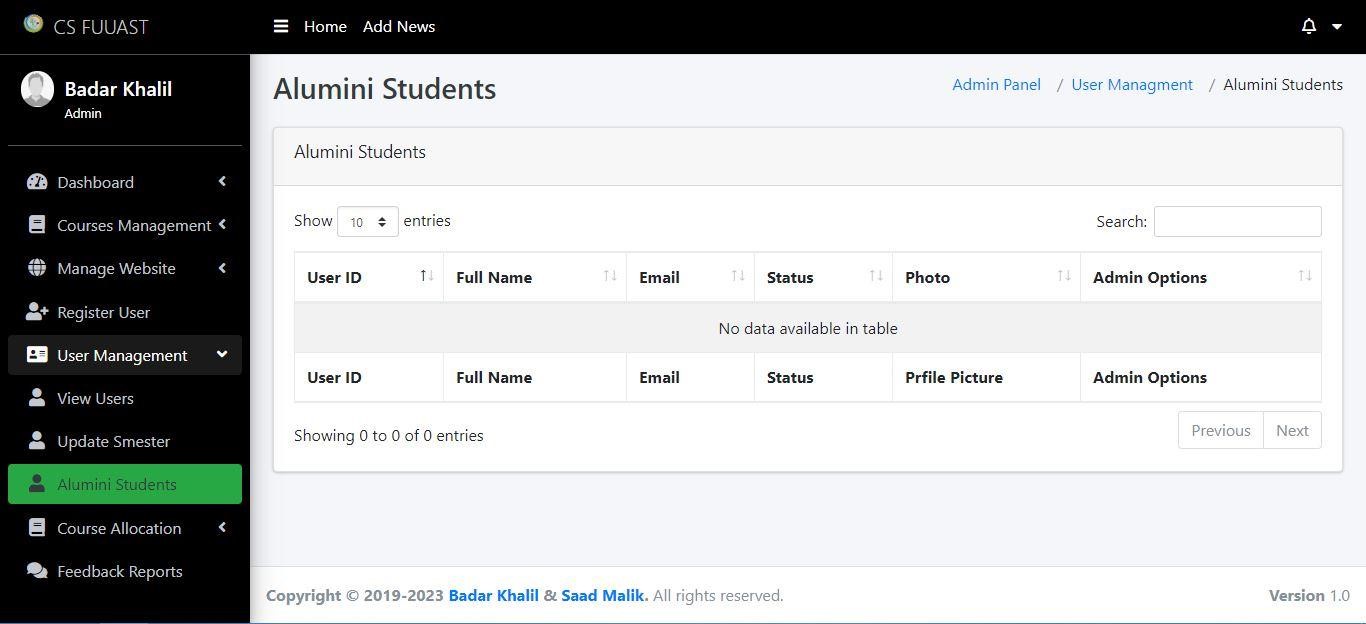


Figure 8.27 show how Alumni record will be displayed

**8.1.20 Assign Courses**

### Assign Courses

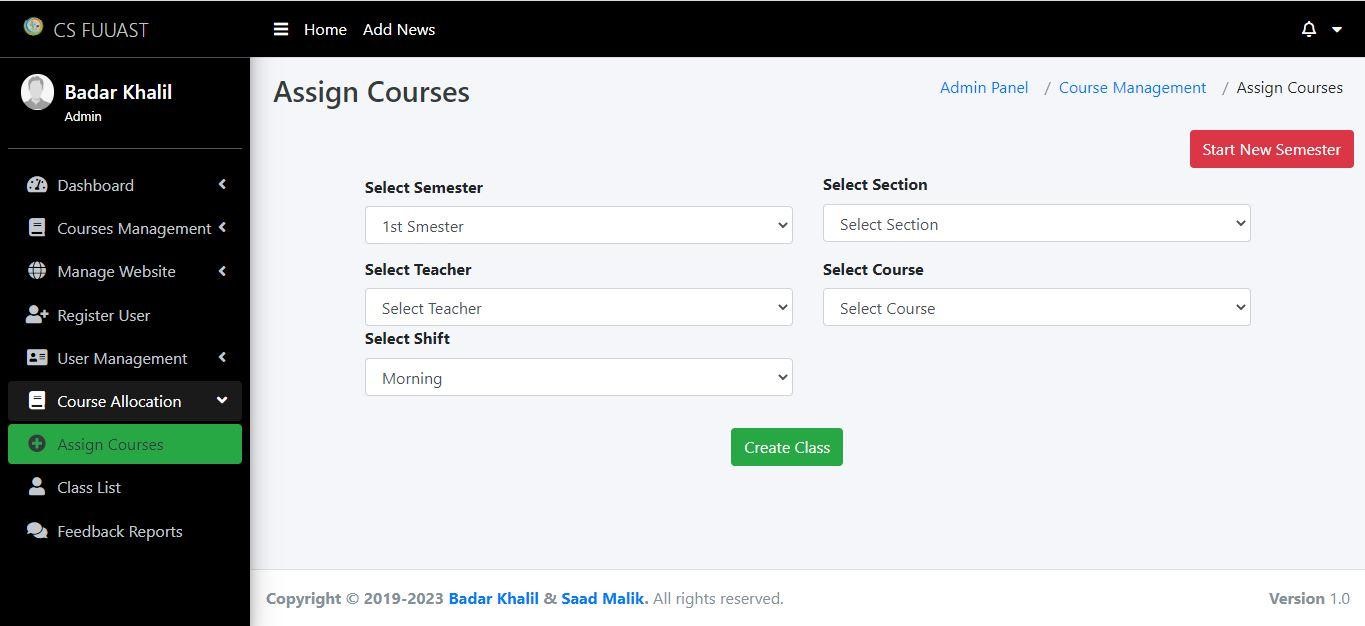


Figure 8.28 show how admin will assign courses

**8.1.21 Class List**

### Class List

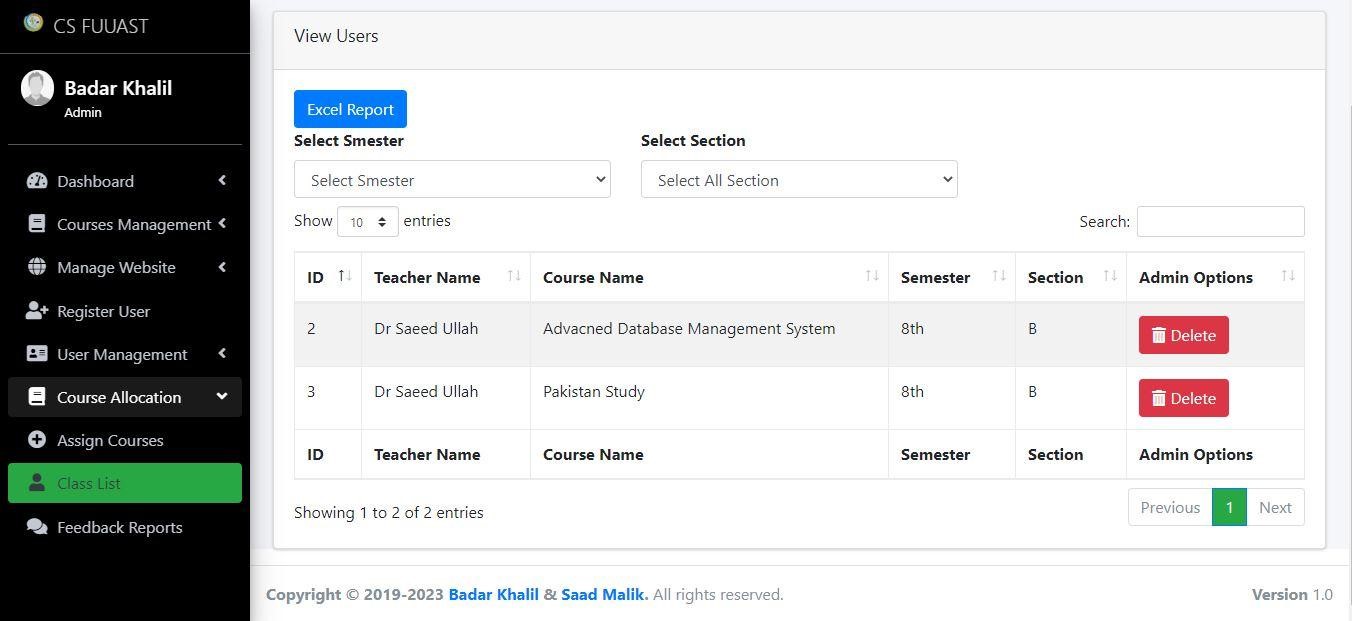


Figure 8.29 shows assigned courses

**8.1.22 Feedback Reports**

### Feedback Reports

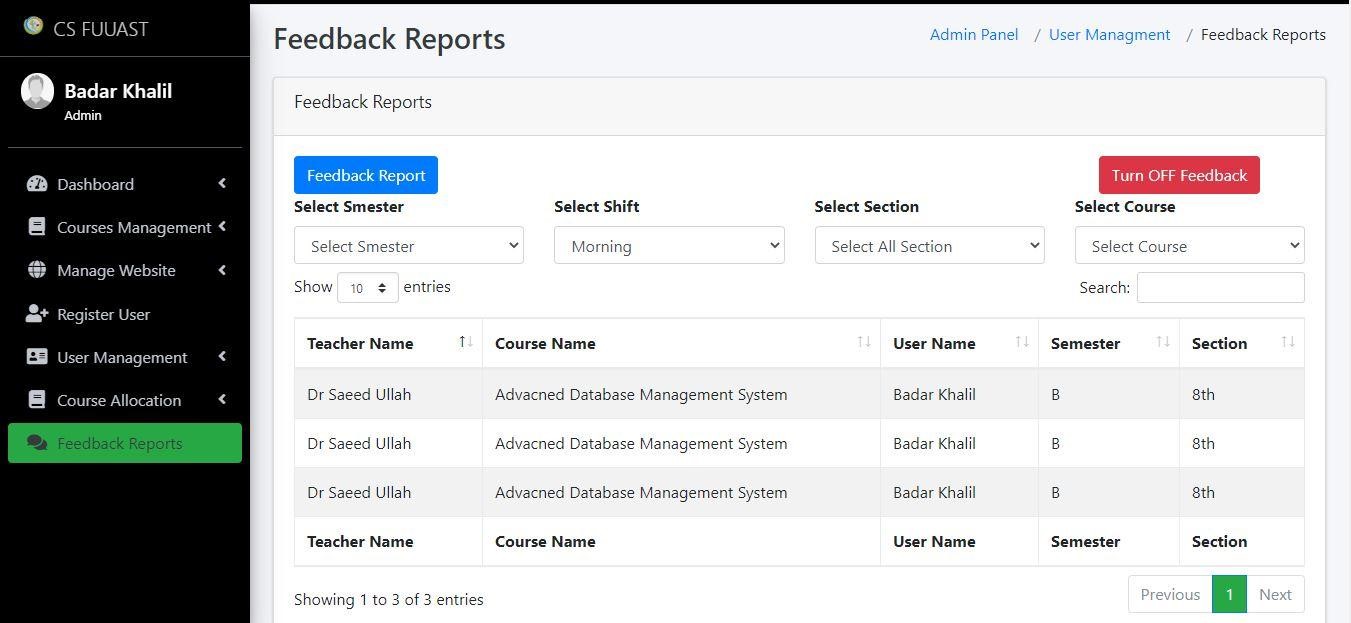


Figure 8.30 shows user feedback and reports can be generated

**8.1.23 Feedback Reports Example**

### Feedback Reports Example PDF

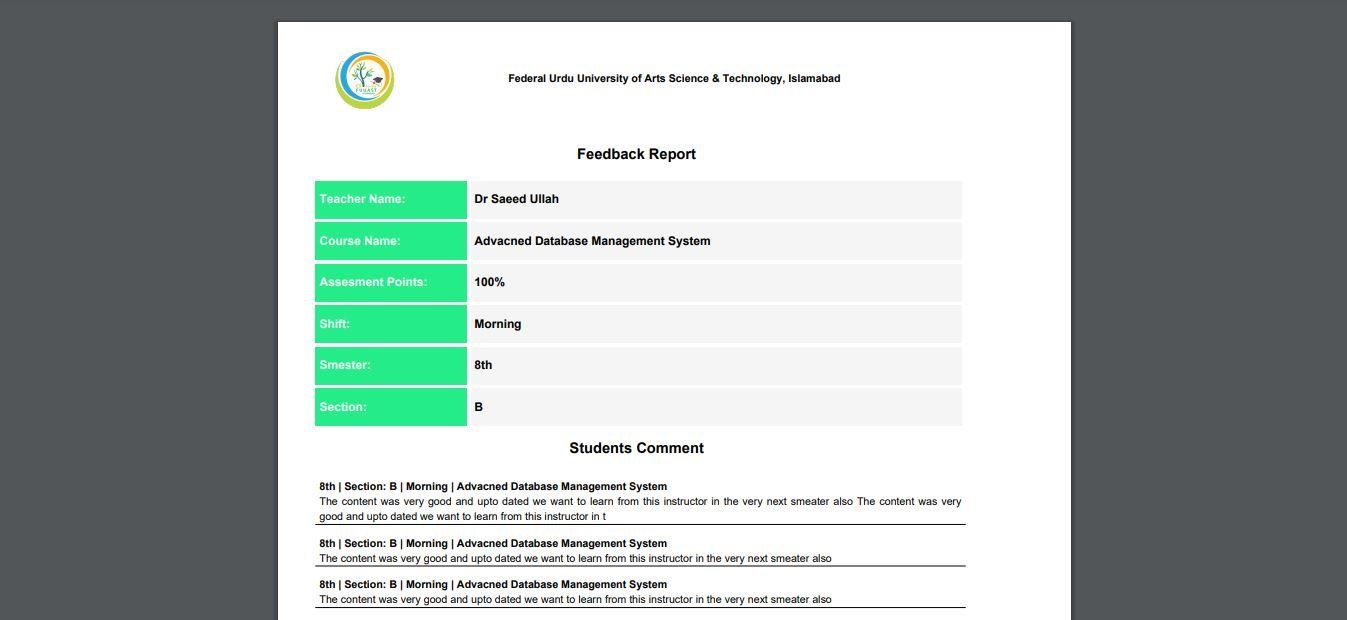


Figure 8.31 Example PDF Feedback report

**8.1.24 Android Splash Screen**

### Android Splash Screen



Figure 8.32 Teacher and Candidate App splash Screen

**8.1.25 Android login Screen**

### Android Splash Screen

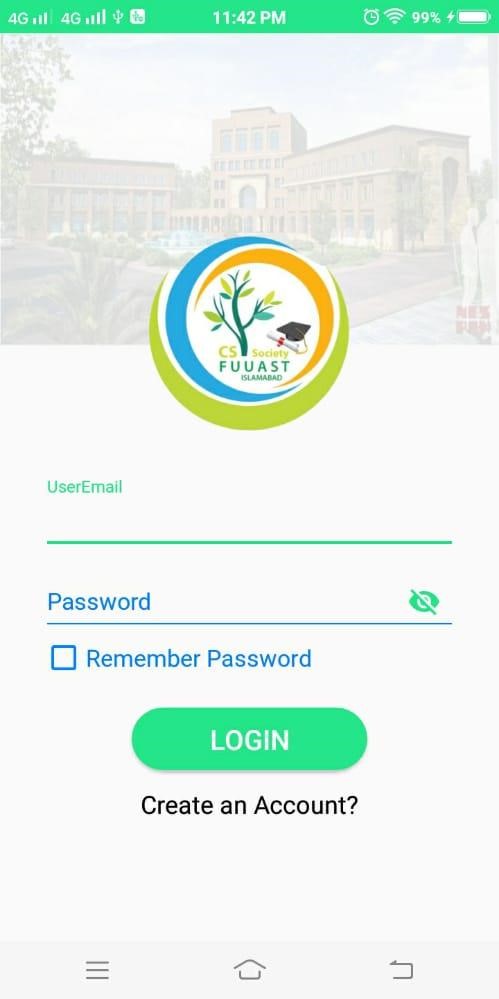


Figure 8.33 Teacher and Candidate App login Screen

**8.1.26 Android Dashboard Entrance**

### Android Dashboard Entrance

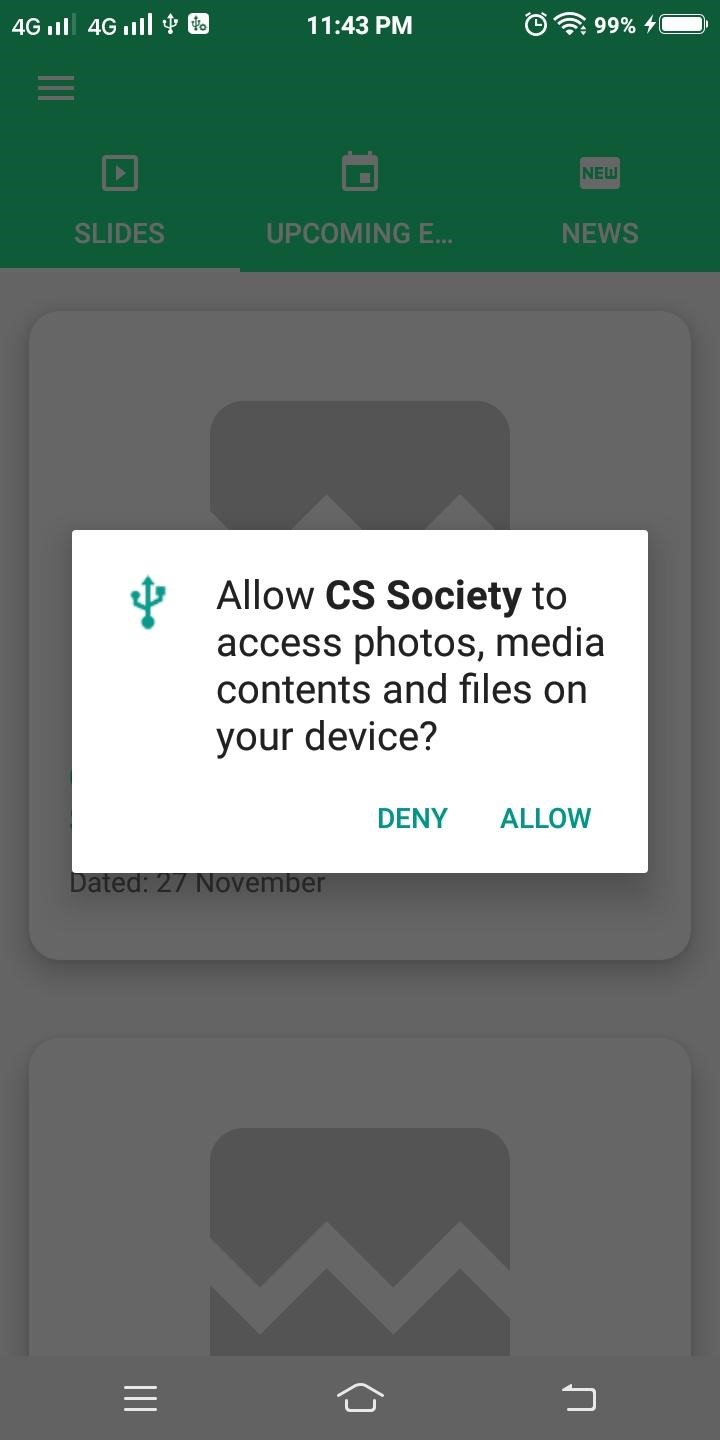


Figure 8.34 Teacher and Candidate App entrance screen

**8.1.27 Android Menu**

### Android Menu

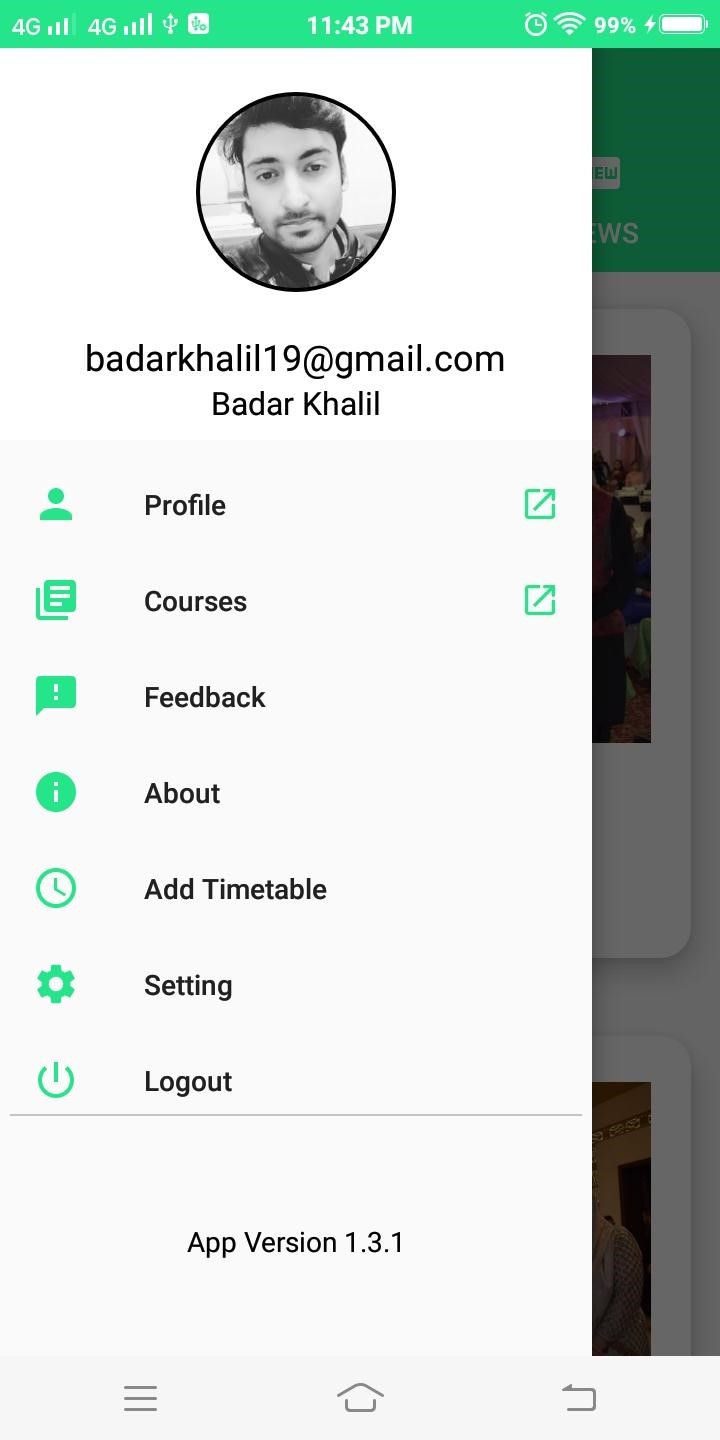


Figure 8.35 Teacher and Candidate App menu screen

**8.1.28 Android Slides**

### Android Slides

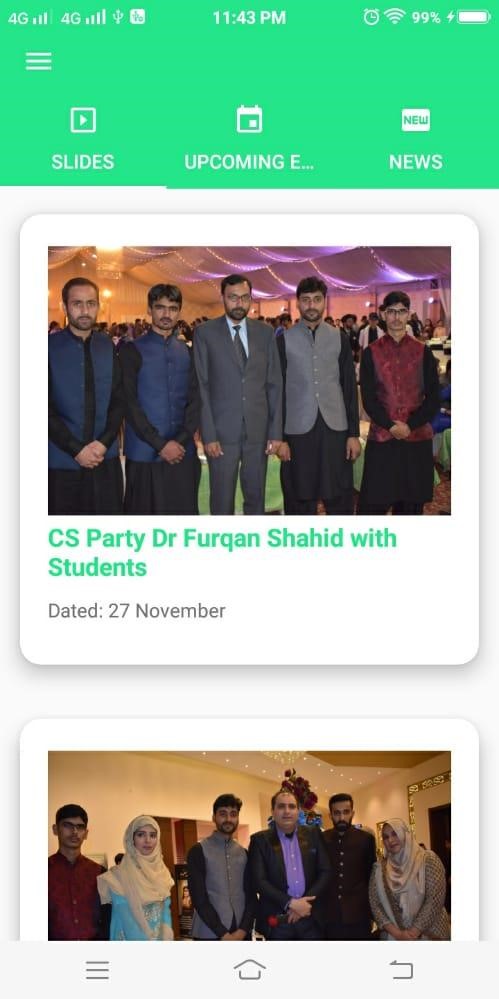


Figure 8.36 Teacher and Candidate App slides show screen

**8.1.29 Android Events**

### Android Events

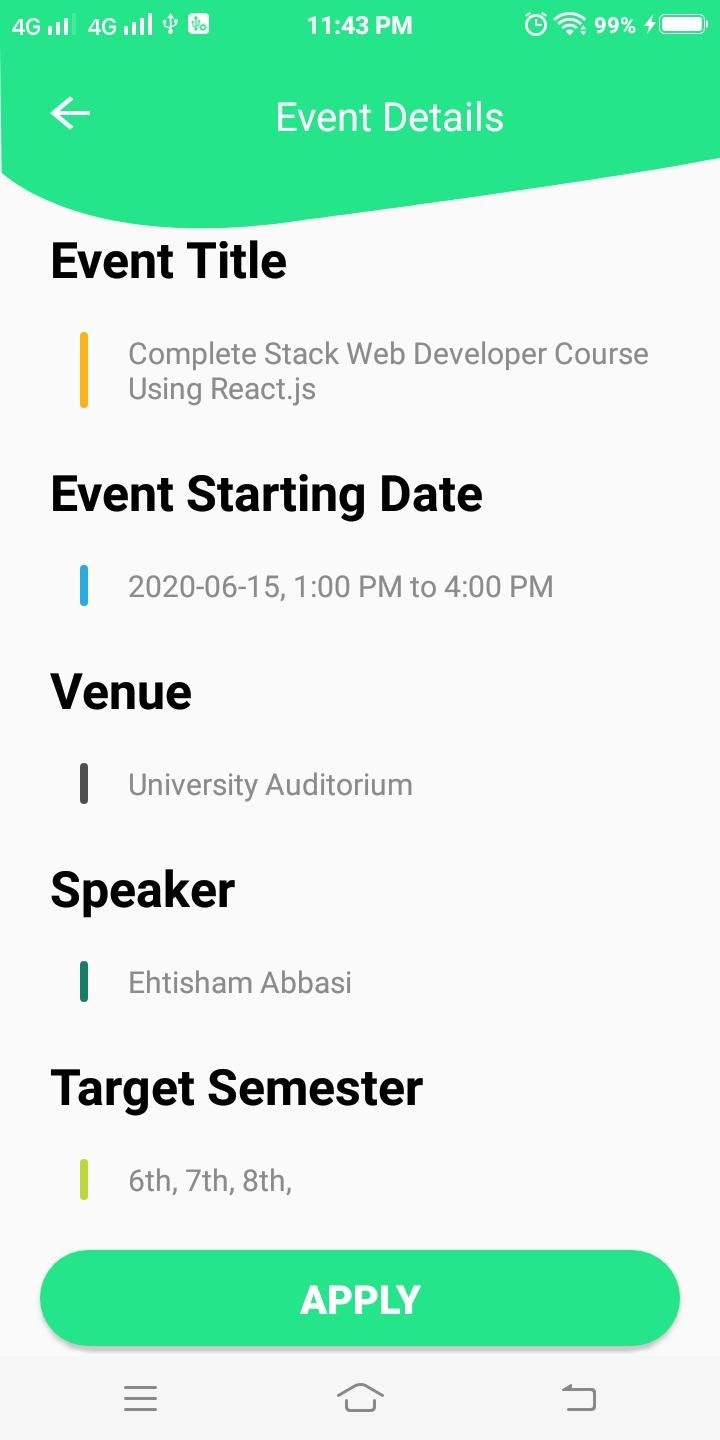


Figure 8.37 Teacher and Candidate events screen to apply



Figure 8.38 Teacher and Candidate events screen

**8.1.30 Android News**

### Android News



Figure 8.39 Teacher and Candidate news screen

**8.1.31 Android Profile**

### Android Profile

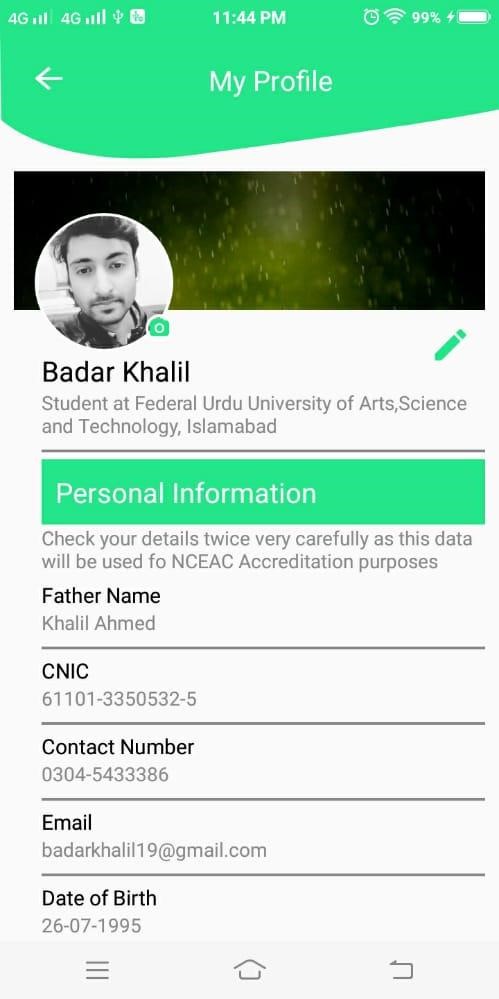


Figure 8.40 Teacher and Candidate profile screen

**8.1.32 Android Course Details**

### Android Course Details

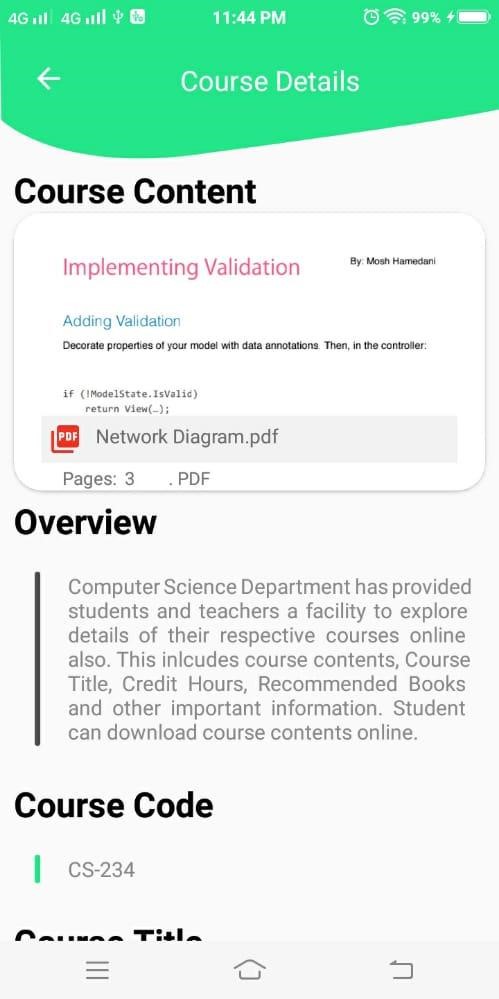


Figure 8.41 Teacher and Candidate Course Details screen

**8.1.32 Android Courses**

### Android Courses

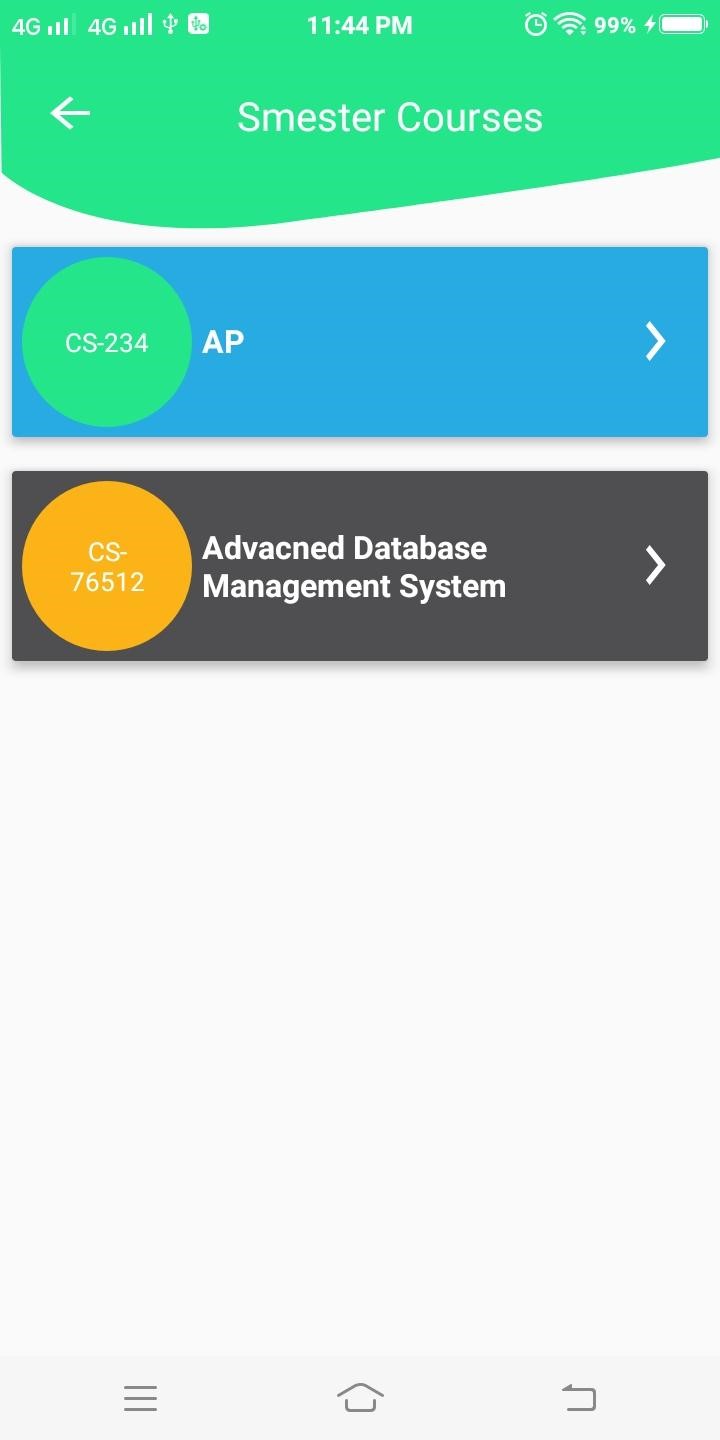


Figure 8.42 Teacher and Candidate Courses screen

#### 8.1.33 Android Assigned Teachers

**Android Assigned Teachers**

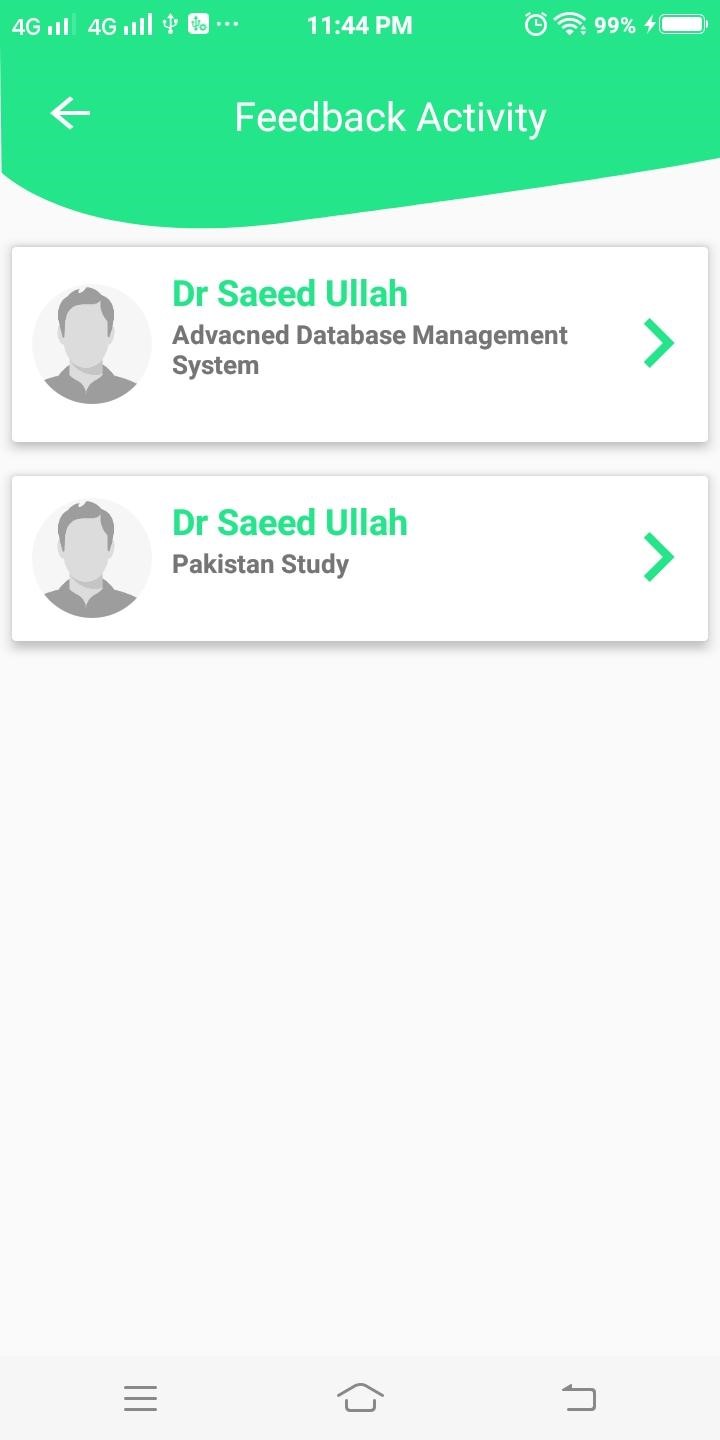
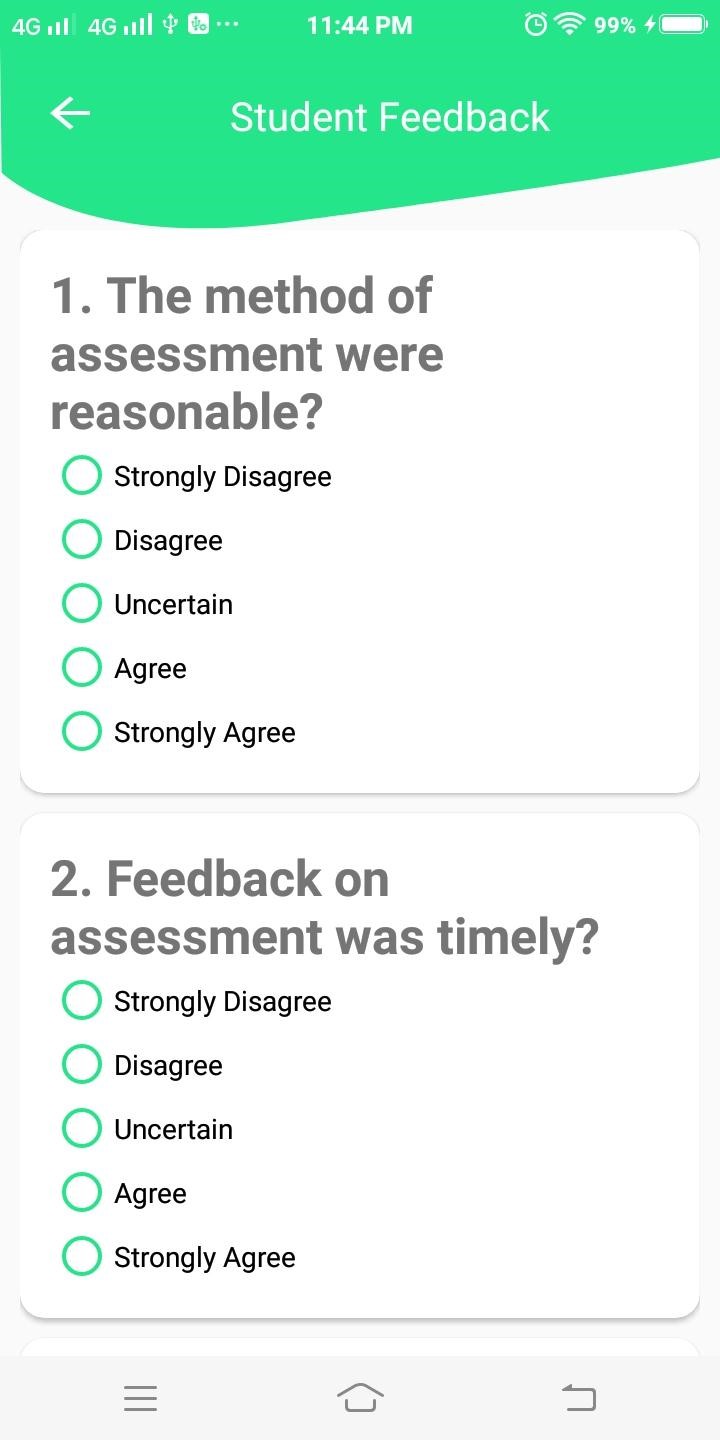


Figure 8.43 Assigned teachers screen

#### 8.1.34 Android Feedback Screen

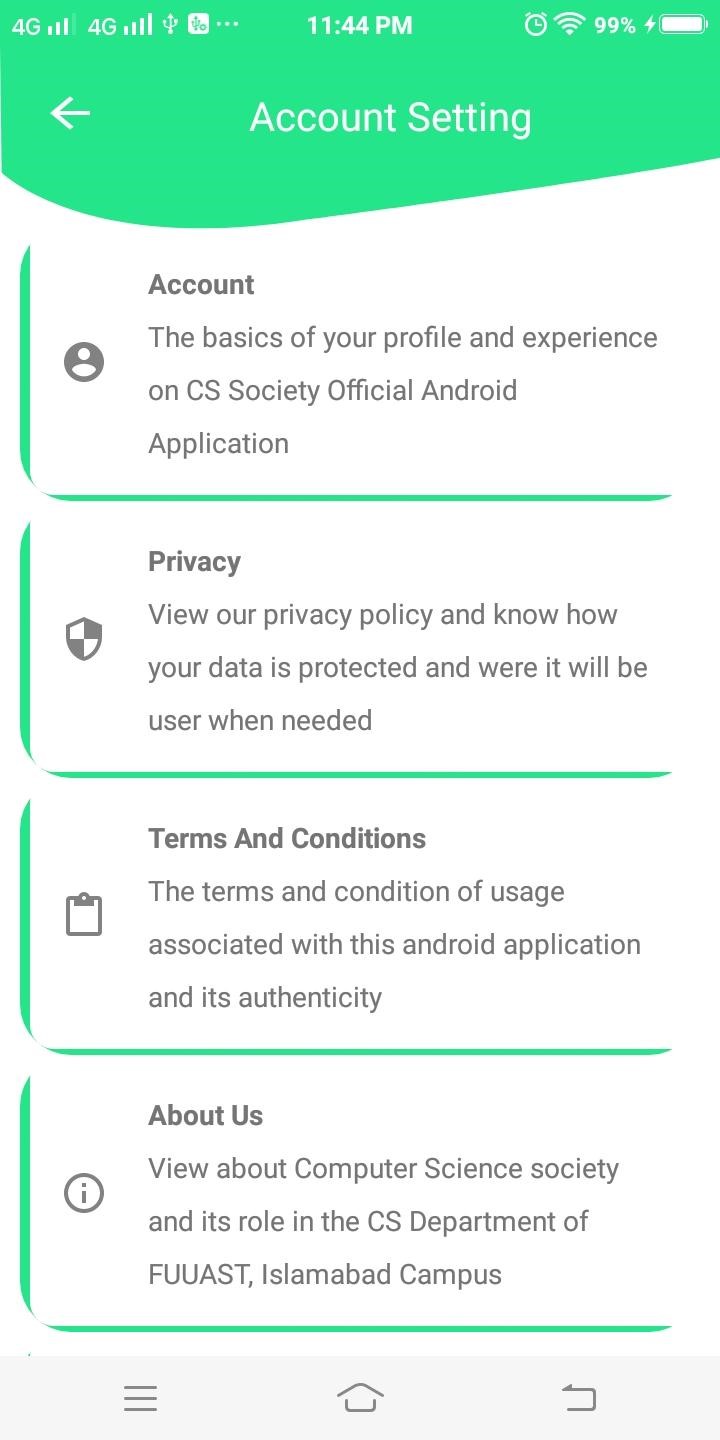
**Android Feedback Screen**



44 Feedback screen

#### 8.1.35 Android Setting

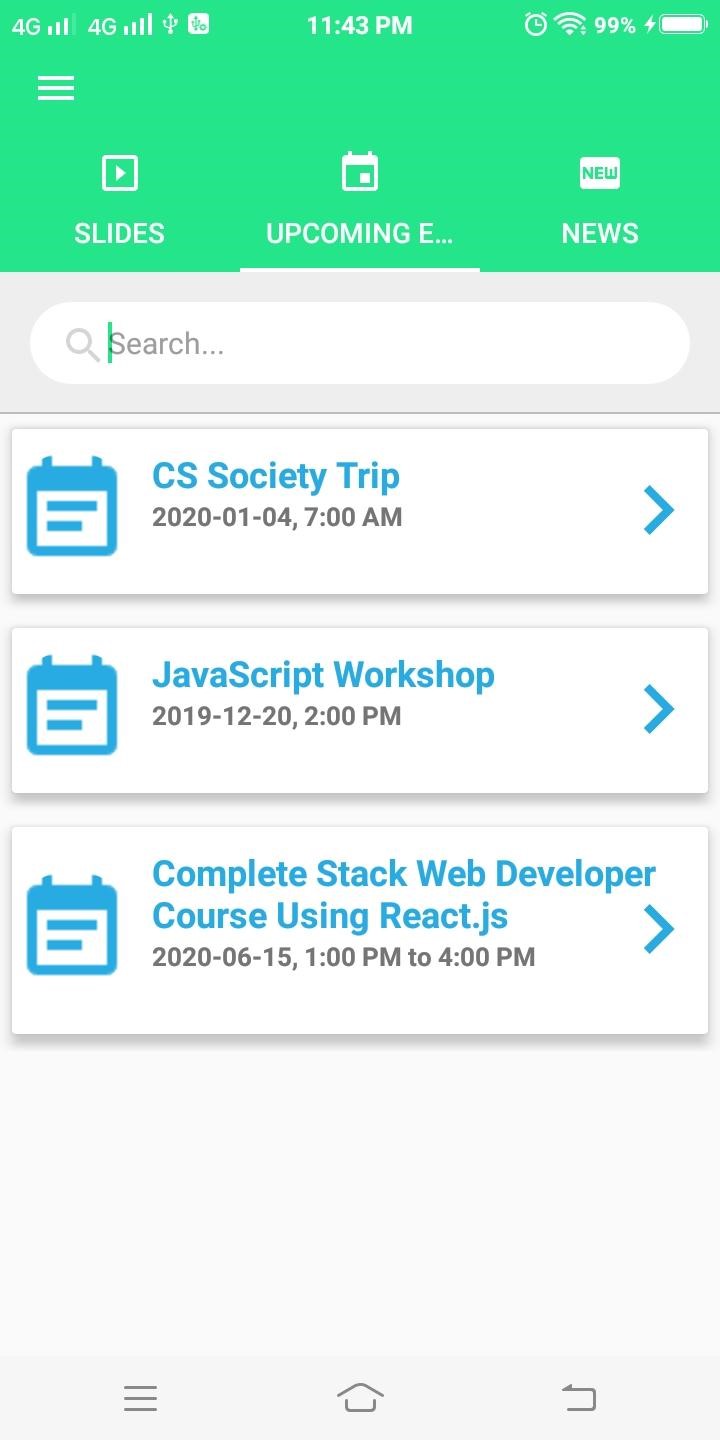
**Android Setting**



45 Feedback Setting

#### 8.1.36 Android Setting

**Android Setting**



46 Upcoming Events

**Chapter 9**

**8REFERENCES & BIBLIOGRAPHY**

### 99.1 References

* [www.msdn.microsoft.com](http://www.msdn.microsoft.com/)
* [www.stackoverflow.com](http://www.stackoverflow.com/)
* www.pluralsight.com  [www.ibm.com](http://www.ibm.com/)

### 59.2 Bibliography

* Software Requirements - Techniques By Soren Lauesen
* Applying UML and Patterns by Craig Larman 2nd Edition