# 

# Chapter 1

**Introduction**

# Abstract:

Education system forms the backbone of every nation. And hence it is important to provide a strong educational foundation to the young generation to ensure the development of open-minded global citizens securing the future for everyone. Air share is online portal for sharing academic documents between students and faculty. This innovative system allows faculty to share important data as well as notifications with students. Air share consists of tasks such as registering students, attendance record keeping controlling absentees, producing reports, Lecture slides, preparing timetable

# 1.0 Introduction

Education system forms the backbone of every nation. And hence it is important to provide a strong educational foundation to the young generation to ensure the development of open-minded global citizens securing the future for everyone. Advanced technology available today can play a crucial role in streamlining education-related processes to promote solidarity among students, teachers, parents and the school staff.

Air share is online portal for sharing academic documents between students and faculty. This innovative system allows faculty to share important data as well as notifications with students. We propose to build this system on an online server that allows faculty to upload data and students may view search and download required documents. Here students only see and download data of their particular semester. Rest data is hidden. Faculty may access and upload/edit documents to any semester or add any notice as desired.

Air share consists of tasks such as registering students, attendance record keeping controlling absentees, producing reports, Lecture slides, preparing timetable and producing different reports for teachers and parents.

**1.2 Module Description**

This Air Share consists of mainly 4 modules.

**They are:**

* Administrator
* Faculty
* Student
* Parents

**Administrator**

Administrator is the main module of this system. Admin manage & control the total university or school.

**Main features are:**

* User creation
* Add syllabus
* Providing right to files
* Delete Members
* Add exams updates
* Upload files

**Teachers**

Another important module is teachers. Teachers manage student & academic activities

**Main features are:**

* Add student
* Upload
* Files download
* Send notification for the class
* View student list

**Student**

Another important module is student. Collaborate with school activities include exams marks class retinue, communicate with teachers

**Main features are:**

* Upload assignment
* Download Files
* View Class Activity
* View class retinue

**Parents**

Another important module is Parents .parents get child all academic updates communication with school admiration.

**1.3 Scope**

The purpose of any new technology is to make people life easier. This project is database used to manage the school and allows the administrators to register the daily required information of Students, Teachers & office staff.

Air share will organize work inside university and proposed system will do the following tasks:

* Insert student’s information such as student name, student number, address etc.
* Insert employee’s information such as employee name, number, address etc.
* Insert section’s information such as section name etc.
* Insert marks for each student in each subject he/she taken.
* View data about certain student and can edit it like adding or removing a student.
* View data about certain employee and can edit it like adding or removing an employee.

**1.4 Objective**

The Objective of the proposed system is to develop a tool which creates liaison between teacher and students so there is no news or notification left untold. Disseminating information and sharing of important documents gives a social network for Teachers and Students to maintain a good professional relationship by fulfilling all the duties from the either sides.

**Our proposed system has several advantages:**

* User friendly interface
* Fast access to database
* Less error
* More Storage Capacity
* Search facility
* Look and Feel Environment

**1.4 Problem Statement**

In the current system we need to keep a number of records related to the student and want to enter the details of the student and the marks manually. In this system only the teacher or the school authority views the mark of the student and they want to enter the details of the student. This is time consuming and has much cost.

Teachers may want to associate a student with his parent or emergency persons for disciplinary measures which need searching of the students record in the record office. It has been difficult to search a record from thousands of such records and observed that students can take any person claiming that he/she is their parent or emergency person which creates problem in control of students.

Features

* Monitor status of the whole university from all in one dashboard.
* View profile of students quickly with just one click of a button.
* Provide downloadable documents in accordance with any topic.
* Manage exam marks of all students.
* Managing daily attendance is now hassle free.
* Very easy to create and manage class routine schedules.

**Chapter 2**

**Literature Review**

**2.0 Introduction**

This chapter discuss the previous work and the scope of Airshare in detail. Already done work on such system is also discussed here

**2.1 Literature Review**

Many higher education institutions have implemented a learning management system (LMS) to manage online Document sharing and Notification System for the student and the teacher, with varying levels of support provided to staff and students, but often there is little subsequent investigation into the quality of the online sites or the use made of the support structures provided. The past decade has seen enormous growth in the use of learning management systems (LMS) in higher education institutions, with varying levels of support provided to staff and students during the implementation phases. This has, in theory at least, provided the potential for rich learning environments built on social constructivist theories and available to all students, both on campus and those studying at a distance. Debbi Weaver [1], presented his findings on implication for quality on use of learning Management System and discussed methods of ensuring quality in developing Online Learning Management Systems. Gorgi Kakasevski [2]. Used data collection and results in evaluating the effectiveness of usability of Learning Management System Michael Machado [3], compared the usability and Effectiveness of two competing learning management systems.

Liu p [4] investigated the current state of how instructors use technology in online courses. Major findings include that asynchronous discussion was Perceived as being very important or necessary to be used in online courses; While real time chat were perceived as less important or less necessary. Positive Correlations were found between the instructors’ perceived importance and Necessity of the technology and how likely they used it.

Jung et al. [5] investigated the effects of three types of interaction (Academic, collaborative, and social) among online undergraduate students in Korea regarding their satisfaction, participation, and attitude toward online education. Social interaction with instructors and collaborative interaction with Peers were identified as important factors for enhancing learning and active Participation in online discussions. Uzunboylu, [6] A review of the two recent mainline e-Learning projects in the European Union, namely the e-Learning Action Plan, and the e-Learning Program have been fully supported by the European Commission, who provided the necessary infrastructure and equipment, instructor training,

Encouragement, cooperation, delivery of useful services, and promotion of digital literacy. NADIRE CAVUS, discussed the study carried out indicated that students using the advanced collaborative tool have shown statistically significant success rates. Although the common properties between the Advanced Collaborative Tool and the Standard Collaborative Tool, such as the ability to communicate with the instructor, sending messages to each other, and the presence of an editor are very important, these properties are not sufficient for the successful teaching of a programming language in Web-based environment Mario Pinto [8], sought to align the technological tools with organizational practices and processes, in the specific context of an HEI.

**References**

[1] Academic and student use of a learning management system: Implications for quality, Australasian Journal of Educational Technology 2008,

[2] Evaluating Usability in Learning Management System Moodle, Gorgi Kakasevski1, Martin Mihajlov2, Sime Arsenovski1, Slavcho Chungurski1

[3] Blackboard vs. Moodle: Comparing User Experience of Learning Management Systems

[4] Zhai, M., & Liu, S. (2005). Technology use in an online program. 21st Annual Conference on Distance Teaching and Learning. Madison, WI.

[5] Jung, I., Choi, S., Lim, C., & Leem, J. (2002). Effects of different types of interaction on learning achievement, satisfaction and participation in Web-Based Instruction. Innovation in Education and teaching International, 39(2), 153-162.

[6] Uzunboylu, H. (2006). A review of two mainline e-Learning projects in the European Union. Educational Technology Research and Development, 54(2), 2006.

[7] ASSESSING THE SUCCESS RATE OF STUDENTS USING A LEARNING MANAGEMENT SYSTEM TOGETHER

WITH A COLLABORATIVE TOOL IN WEB-BASED TEACHING OF PROGRAMMING LANGUAGES, NADIRE CAVUS, J. EDUCATIONAL COMPUTING RESEARCH, Vol. 36(3) 301-321, 2007

[8] Knowledge Management in Higher Education Institutions: A framework to improve Collaboration, June 2014

**2.2 Chapter Summary**

This chapter discussed the importance of online document sharing and system that are deployed in the industry currently. Web application are major part of these system due to easy access, reachability and effective task performance.

**Chapter 3**

**Planning and methodology**

**3.0 Introduction**

This chapter is related to project scheduling. The project deliverables that have been delivered to project supervisor are also stated in this chapter. This chapter also explain the methodology adopted to accomplish the task.

The following are the project deliverables.

* Project Management Plan
* Software Requirement Specification
* Test Design Document

**3.1 Evaluation of the Plan**

The preliminary draft of the software project management plan were submitted to Mr. Ahmad Mohsin on specified date.

**3.2 Project Plan**

**3.2.1 Project Organization**

The project organization defines the human infrastructure of the project. This task is designed to define the project organization chart the roles, and the relationship of the project team.

**3.2.2 External Interfaces**

The external interfaces for the project would be the members of the project committee.

**3.2.3 Internal Interfaces**

None

**3.2.4 Roles and Responsibilities**

The software developer is responsible for all the documentation to be developed and for all work to be done.

**3.3 Project Management Plan**

A project plan is a formal documentation that describe the execution of the project and project control. The basic usage of the project management plan is to document the planning and scheduling. Communication among the team, project scope and cost. A project plan may be a summarized or detailed document according to need.

**3.3.1 Project schedule**

Project schedule is a part of project management plan that describe what work is need to be done, what resources are to be provided for that work and what timeframes are allotted to a piece of work that need to be performed. Project schedule describes all the work associated that could deliver the project on time. A full and complete schedule helps the project manager to effectively communicate his effort in term of resources necessary to deliver the project.

The below figure describe the component of the project schedule covering scope, requirements, design, coding part, testing documentation, deployment and the time frames allotted respectively.

**3.4 Control Plan**

A control plan is method for documenting the main elements of quality control that are implemented to make sure that quality standards are met for the product that is developed.

**3.4.1 Requirement control plan**

When changes are to be made in the requirement after the **SRS** has been released, the changes brought into the attention of the committee and discussed. Any change that is to be done was with prior approval of the committee and only if feasible and permissible within the constraint’s

Of the project, and resources in term of knowledge and skill of the developer required. Once the changes made to the SRS document, an update version of the SRS released and circulated to the committee.

**3.4.2 Schedule control plan**

To keep stick with the schedule, I had to spend extra time on project. Most of the documentation phase was completed in **Phase-1** of the final year project. During this time period further analysis for the system was done simultaneously.

**3.4.3 Quality Control Plan**

To ensure product quality, black box testing was done simultaneously.

**3.5 Methodology**

A methodology or process model help to organize the activities performed during completion of project. There are a lot of process models currently in practice by software engineers ranging from conventional, time consuming models for large project to iterative, fast project development process models. The waterfall model is choose as process model for this project and all activities are organized and arranged as per methodology defined in this model. Some of the reason for choice of waterfall model as process model are:

* Clear user requirement
* Plenty of time available for project completion
* Division of whole project in two phases P1 and P2 for Design and Development
* Simple and straight forward nature of project

Air Share will be implemented and executed following the iterative approach illustrated below.



As it is a large system that being implemented, some feature may be kept adding with time. My reason for selecting iterative model is that in such type of system, that are processed for decision making purposes, it is common for new analytical details addition, addition of new reports and many other synchronization features. Accordingly one needs to make some modification on software design and requirement analysis. Thus the iterative development process is the most suitable methodology for decision support system. Due to nature of the project the iterative SDLC model is ideal.

**3.6 Summary**

In this section I’ve explained the process of planning and methodologies used in development cycle of this system that include analysis, project schedule and process flow of the whole system. The software requirement specification for the project are discussed in the next coming chapter.

**Chapter 4**

**Software Requirement Specification**

**4.0 Introduction**

Requirement engineering is the one of the most important process in the product development life cycle. The better we perform this portion better we have results. The bottom line of all the product being developed is the Quality. One aspect of the Quality is achieved through better understanding of user requirement and their conformance in the final product. In this chapter, name Software Requirement’s specification, will be documenting all the necessary requirement for “Student Faculty document Sharing” which is web application. Along with requirement this chapter also contains uses cases.

In this chapter following component of SRS and design document will be covered.

* Business Requirements
* Process Flow
* Functional Requirements
* Non-functional Requirements
* Actors
* Use-Cases
* Traceability Matrix
* System Architecture Diagram
* Data Model
* Entity Relationship Diagram
* Class Diagram
* Graphical User Interface

**4.1 Business Requirements**

Student and faculty document sharing system is important for student and teachers. There is need for a system which can ensure that time is preserved as well as used efficiently for everyday in life.

**4.1.1 User Requirements**

1. Log into the system
2. Customize the Profile Setting
3. Manage the access to the user
4. Add or delete files
5. Upload and download files
6. Create user account such as teachers and student
7. uploaded document
8. View the upload document
9. When teacher upload any document and send notification of this document on mobile phone
10. View the history of user activity

**4.2 Process Flow**

A **process flow diagram** (PFD) is a **diagram** commonly used in chemical and **process** engineering to indicate the general **flow** of plant **processes** and equipment. The PFD displays the relationship between major equipment of a plant facility and does not show minor details such as piping details and designations.

Below is the process flow diagram of the system. This diagram show the processes flow operated by three user i.e. Admin, Faculty and Student from the left top process, the faculty and student logs himself in application Air share. From here the F & S can manage the profile setting. From this step, faculty and student can upload their files or document related to your semester and also view and download document. Further on faculty and admin also post new event notice about the classes and general information message from the university. From the right side, there’s process flow for system administrator. After he’s logged in, he can manage every entity of the system include student faculty, admin, classes, subject, and event, notice, time table, upload files records, also manage and view the history. View the list of all the registered members.

C:\Users\zubairsaif\Downloads\Untitled Diagram (5).png

C:\Users\zubairsaif\Downloads\Untitled Diagram (6).png

**4.3 Functional Requirements**

In Software engineering and systems engineering, a functional requirement defines a function of a system or its component. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.

**4.3.1 FR01: User Login**

Below given table describe the component or step for login

|  |  |
| --- | --- |
| FR01-01 | System shall allow user to login |
| FR01-02 | System shall get username from user |
| FR01-03 | System shall get password from allow user |
| FR01-04 | System shall authenticate user when he/she submits username and password on login button |
| FR01-05 | System shall allow user to reset the his username name password when he/she clicks on reset button |
| FR01-06 | System shall allow user to remember his username and password by entering into the checkbox. |

**4.3.2 FR02: User can change his password**

Below given tables illustrate the essential step for FR02

|  |  |
| --- | --- |
| FR02-01 | System shall get username from user |
| FR02-02 | System shall get old password from user |
| FR02-03 | System shall get new password from user |
| FR02-04 | System shall get new password confirmation password from user |
| FR02-05 | System shall check does the old password and does the new password and reconfirmation password match |
| FR02-06 | System shall get maximum length of password at least character |
| FR02-07 | System shall check the length of password |

**4.3.3 FR03: Customizable profile**

The following tables explain the functional requirement of profile setting

|  |  |
| --- | --- |
| FR03-01 | System shall allow user to customize his profile |
| FR03-02 | System shall allow user to manage the notification for mobile |
| FR03-03 | System shall allow user turn on notification for mobile phone |
| FR03-04 | System shall allow user turn off notification for the mobile phone |
| FR03-05 | System shall allow user to manage other services of the system |

**4.3.4 FR04: Upload File**

Below are the component the student faculty and admin shall able to upload document.

|  |  |
| --- | --- |
| FR04-01 | System shall allow to user upload file |
| FR04-02 | System shall allow to check the format of document |
| FR04-03 | System shall allow to check file size |
| FR04-04 | System shall allow to user upload file when click on upload button |
| FR04-05 | System shall authenticate the file size and format |
| FR04-06 | System shall give an error if the file size greater than given limits |
| FR04-07 | System shall give an error if the file format is incorrect |
| FR04-08 | System shall allow to user rename the file name |

**4.3.5 FR05: Download File**

Below are the component the student faculty and admin shall able to download document.

|  |  |
| --- | --- |
| FR05-01 | System shall allow to user when click on download button |
| FR05-02 | System shall give an error when the click button not work proper |
| FR05-03 | System shall give message if file access denied for invalid user |

**4.3.6 FR06: Notification**

Below are component explain the functional requirement 6

|  |  |
| --- | --- |
| FR06-01 | System shall allow to send notification when specified user upload file |
| FR06-02 | System shall allow to user turn on notification on new update when click on turn on button |
| FR06-03 | System shall allow to user turn off notification when click on turn off button |
| FR06-04 | System shall allow to user customizes their notification when click on advance option |
| FR06-05 | System shall allow to user when click on reset as default option |

**4.3.7 FR07: Create Folder**

Below the table explain the FR07

|  |  |
| --- | --- |
| FR07-01 | System shall allow to user when click on new folder button |
| FR07-02 | System shall allow to user rename the folder when right click option |
| FR07-03 | System shall allow to user delete folder when click on delete folder button |
| FR07-04 | System shall allow to user upload file in the folder when uploaded in the folder |

**4.3.8 FR08: Sharing File**

The following table about the document sharing on Facebook on other social media

|  |  |
| --- | --- |
| FR08-01 | System shall allow to user when click on share button |
| FR08-02 | System shall give an error when user can’t select any document for sharing |
| FR08-03 | System shall authenticate the file sharing media and then share it |
| FR08-04 | System shall check if no authenticate the give error message “there is something went wrong ” |
| FR08-05 | System shall allow to check file access type |

**4.3.9 FR09: Admin Login**

Below given the table describe the component or step about Doctor Login

|  |  |
| --- | --- |
| FR09-01 | System shall allow admin to login |
| FR09-02 | System shall get password from admin |
| FR09-03 | System shall authenticate admin when he/she submits username and password on login button |
| FR09-04 | System shall allow admin to reset the his username name password when he/she clicks on reset button |
| FR09-05 | System shall allow admin to remember his username and password by entering into the checkbox. |

**4.3.10 FR10: Login Roles**

In this table explain the different role for the login user

|  |  |
| --- | --- |
| FR10-01 | System shall allow admin to create roles |
| FR10-02 | System shall allow to admin edit roles |
| FR10-03 | System shall allow to admin delete roles |

**4.3.11 FR11: Add Student**

In this table explain creation of student account.

|  |  |
| --- | --- |
| FR11-01 | System shall allow admin to create student account |
| FR11-02 | System shall allow to add student id in text box |
| FR11-03 | System shall allow to admin add user name into the textbox |
| FR11-04 | System shall allow to admin select the class of the student from dropdown menu |
| FR11-05 | System shall allow to admin when press reset button clear all the fields |
| FR11-06 | System shall allow to admin when click on add button system give message student added successfully |

**4.3.12 FR12: Add Faculty**

In this table explain creation of Faculty account.

|  |  |
| --- | --- |
| FR12-01 | System shall allow admin to create Faculty account |
| FR12-02 | System shall allow to select department from dropdown |
| FR12-03 | System shall allow to admin add 1st name  of faculty in field one |
| FR12-04 | System shall allow to admin add 2nd name of faculty |
| FR12-05 | System shall allow to admin when press reset button clear all the fields |
| FR12-06 | System shall allow to admin when click on add button system give message faculty added successfully |

**4.3.13 FR13: Manage the user**

In this step I have done to manage the user

|  |  |
| --- | --- |
| FR13-01 | System shall allow user create account |
| FR13-02 | System shall allow to user add upload document in the system |
| FR13-03 | System shall allow to user delete specified document |

**4.3.14 FR14: Login activity track**

In this step admin view the history of user activity

|  |  |
| --- | --- |
| FR14-01 | System shall allow to admin keep track of user |
| FR14-02 | System shall allow to admin manage login activity |
| FR14-03 | System shall allow to admin see activity log of user |

**4.3.15 FR15: Upload document**

In this step explain about the upload document

|  |  |
| --- | --- |
| FR15-01 | System shall allow admin upload file and folder |
| FR15-02 | System shall allow to admin upload any type of information upload and download |

**4.3.16 FR16: View the user list**

The below table explain about the all user list

|  |  |
| --- | --- |
| FR16-01 | System shall display list of all the user |
| FR16-02 | System shall allow to display the list of admin |
| FR16-03 | System shall allow to display the list of faculty |
| FR16-04 | System shall allow to edit all the user |

**4.3.17 FR17: Add Notice and Event**

In this step explain functional requirement 15

|  |  |
| --- | --- |
| FR17-01 | System shall allow to admin add notice and event into event calendar |
| FR17-02 | System shall to admin remove notice and event from calendar menu |
| FR17-03 | System shall allow to display the list of total events |
| FR17-04 | System shall allow to display list of notice |

**4.4 Non-functional Requirements**

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior or functions.

**4.4.1 NFR01: Performance**

The component regarding the performance of the system are discussed in the table below.

|  |  |
| --- | --- |
| NFR01-01 | The average load time of the starting page of the system must be less than 5 seconds. |
| NFR01-02 | Average processing time taken by the system to complete a transaction/ request by a user should be less than 10 seconds. |
| NFR01-03 | System mean time to failure should not be more than 30 second within 24 hours of use. |
| NFR01-04 | Average system response time should not be greater than 5 seconds. |
| NFR01-05  NFR01-06 | System must successfully run on client machine with 256 MB RAM or above  Multiple user can access the same website but the priority of the admin would be high. |
| NFR01-07 | System shall be able to handle multiple requests over the internet of 100 users at the maximum. |

**4.4.2 NFR02: Security**

Security is most important feature of Air share system. The security criteria is describe in the following table.

|  |  |
| --- | --- |
| NFR02-01 | System must provide access to authorized user only that enter through the login module. |
| NFR02-02 | System must not provide access to any user accepts the designated user to update the database. |
| NFR02-03 | After the end of user session, no information must be saved anywhere on the client machine. |
| NFR02-05 | A proper mechanism shall be adopted in order to avoid any hacker’s attacks on the application. |

**4.4.3 NFR03: Availability**

In centralized system, availability is concerned the top most priority. The NFR’s with respect to availability are discussed in the table below.

|  |  |
| --- | --- |
| NFR03-01 | This web based application shall be available 24 hours a day. |
| NFR03-02 | User may access it from anywhere of the world. |

**4.4.4 NFR04: Documentation**

The documentation meets the criteria given in the following tables

|  |  |
| --- | --- |
| NFR04-01 | User document must be done giving details of the functionalities. |
| NFR04-02 | Online help should available in case of any problem |

**4.4.5 NFR05: Disaster Recovery**

In case of disaster, the recovery parameters are defined in the following tables

|  |  |
| --- | --- |
| NFR05-01 | This should be made sure that a proper recovery system exists in case of any disaster |
| NFR05-02 | In case of client / server crash all information data should be recordable within 30 mints of incidence. |

**4.4.6 NFR06: Usability**

For the system to be useful, it should meet the usability criteria. The defined usability standard for the system are given in the tables below

|  |  |
| --- | --- |
| NFR06-01 | The system should be easy to use for a novice user |
| NFR06-02 | Time to learn overall interface of the system should be minimal |

**4.4.7 NFR07: Defects maintenance**

|  |  |
| --- | --- |
| NFR07-01 | Post release defects of the system must not exceed 1 critical bug per month. |
| NFR07-02 | Post release bug fixing should not take more than 5 hours. |

As said above, non-functional requirements specify the system’s ‘quality characteristics’ or ‘quality attributes.

**4.5 Assumptions and Constraints**

It illustrate how and what components will be used to build the system including the development tools and language, operating system and on which browser the system will be able to run effectively.

**4.5.1 Development Language and tools**

Following are the language and tools used for the development of the system.

**4.5.1.1 Language**

Following are the language used for air share development.

* PHP
* HTML
* CSS
* JAVASCRIPT
* JQUERY
* AJAX
* SQL

**4.5.1.2 Tools**

Following tools were used during the development cycle.

* Xampp server
* Notepad ++
* Dream viewer
* Draw.io
* Microsoft project
* Adobe Photoshop CS6 (To aid GUI)

In addition to the tools, some framework were also used in the development:

* Bootstrap (For a responsive web interface)
* Ajax Control toolkit for php

**4.5.2 Operating system**

The system will be built in Microsoft windows 10 and using Xampp server and will require browser to run. At server side it run in apache server and at client side it may accessed from any machine using any operating system and supported internet browser.

**4.5.3 Browser Support**

The system is able to run in following browsers:

* Microsoft internet Explorer 8, 9, 10, 11
* Firefox
* Google chrome
* opera 9.0 and above
* Safari
* Maxton
* UC brow

**4.6 Actors**

Following is a list of the actors and their responsibilities involved in the Air share system

**4.6.1 Student and Faculty**

Student and faculty will be the person who would access the application end on web browser and will be using this to faculty can add the document time table and other important martial. Student can upload their assignment to appropriate semester and teachers menu and also download the document.

* Log into the system
* Upload the files
* Download the files
* Upload assignment
* View event calendar
* View activity about university
* Manage your profile setting

**4.6.2 System administrator**

The system administrator is responsible for creating account for student, faculty and admin and has to manage these accounts. He’ll be keeping eye on every feedback given to system. He’ll be manage available document. He‘ll be monitoring the activity log of the student and teachers for every types of activity when they are performed and also handle the contact subject and classes.

Responsibilities for the system administrator are as follows:

* Create accounts for student faculty and admin
* Manage the classes and subject details information
* Manage the user role
* View activity log of the student and teachers
* View the activity of admin
* Manage the profile
* Upload the document
* Add new event for classes
* Add notification

# 4.7 Use Cases

In software and systems engineering, a use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modeling Language as an actor) and a system, to achieve a goal. The actor can be a human or other external system.

## 4.7.1 Types of Use Cases:

There are **three** types of use cases: Essential, Concrete and Abstract. They are defined as follows:

**4.7.1.1 Essential Use Cases:**

Essential user casesare expressed in an ideal form that remains relatively free of technology and implementation details; design decisions are deferred and abstracted, especially those related to the user interface.

**4.7.1.2 Concrete or Real:**

Concrete or real use case concretely describes the process in terms of its real current design, committed to specific input and output technologies and so on. When a user interface is involved, they often show screen shots and discuss interaction with the widgets.

**4.7.1.3 Abstract:**

Abstract Use Case is not complete and has no actor that initiates it but is used by another use cases.

**4.7.2 Use Case Diagram**

Diagram that is consist of actors, use cases, system and the relationship between them is called a use case diagram.

Follow is the use case diagram for Air Share.



**4.7.3 Descriptive USE CASES**

The interaction between the actor and the system to accomplish a specific goal is followed by some series of steps. The goals may have some pre-condition to be fulfilled and post-condition as a result. The details description of what the user case defines, pre-condition, steps, alternative flow and post condition is called descriptive use case

**4.7.3.1 UC01: login to the system**

**Brief Description:**

This use case elaborate the login process by which user will log into the system according to the assigned privileges.

**Actors:**

* Teachers
* Student

**Pre-Conditions:**

1. If user want to login he must have an account.

**Main Flow:**

1. Actor enters into the system by entering URL.
2. Login Screen Appears.
3. User will enter Username and Password in the textboxes.
4. User will check the “Remember Me” check box.
5. User will click on the Login Button.
6. System will authenticate the user and give privileges according to user type.
7. System shall store the account information on user’s machine if user has ticked *“Remember Me”.*
8. The system will direct the user to home page.
9. As user is successfully logged in to the system use case ends here.

**Alternative Flows:**

**5(a).** User clears Username and Password by clicking ‘*Reset*’ Button.

**6(a).** System will prompts the user for missing username and password.

**6(b).** If login information is not correct, system will alert for valid login information.

**Exceptions:**

In case Database server is not responding, system will display the message:

“*The database server is down. Please check back after a few minutes*.”

In case web server is down, system will display the message:

“*The web server is down. Please check back after a few minutes*.”

**Post Condition:**

1. User will be logged in to the account.

## 4.7.3.2 UC02: Change Password

### Brief Description:

This Use case elaborate the process by which actor can change the password of his account.

### Actors:

* Teachers
* Student

### Pre- Conditions:

* If user wants to change his password, he will have to login to the system first.
* A user must know their current password then enter a new password twice.

### Main Flow:

1. Use case starts when user/actor clicks on a "change password" link.
2. The system navigates them to a page with a change password form.
3. The form will have three textboxes: current password, new password, and confirm new password, with two buttons submit and cancel.
4. Current password must match their current password and the new password should match its confirmation.
5. If successful, bring them to a new page saying that their password has been changed with a continue button.
6. The cancel button brings them back to the home page as well.

### Exception:

1. In case Database server is not responding, system will display the message:

*“The database server is down. Please check back after a few minutes.”*

1. In case web server is down, system will display the message:

*“The web server is down. Please check back after a few minutes.”*

### Post Condition:

The user record is updated in Database with the new password.

## 4.7.3.3 UC03: Profile View

### Brief Description:

This Use case defines the process by which the User can see his/her profile by clicking on profile button after login.

### Actors:

* Teachers
* Student

### Pre- Conditions:

* If user wants to see his profile, he will have to login to the system first.

### Main Flow:

1. Actors will login to the system by entering valid Username and Password.
2. Home page will appear on the screen with different menus.
3. At the top of the home page there is a profile view option.
4. Actor just have to click on the Profile View.
5. Link will direct user to the profile view page.
6. On profile page all the details of an actor will be seen.
7. Profile Picture
8. Name
9. Designation
10. Contact Information
11. Gender

### Alternative Flows:

4(a) - User will not click on the Profile View

### Exception:

1. In case Database server is not responding, system will display the message:

*“The database server is down. Please check back after a few minutes.”*

1. In case web server is down, system will display the message:

*“The web server is down. Please check back after a few minutes.”*

### Post Condition:

* User can view his profile.

## 4.7.3.4 UC04: Upload files

### Brief Description:

This use case will allow user/actor of the system to upload document files (Pdf, Docx, etc.).

### Actors:

* Teachers
* Student
* Admin

### Pre- Condition:

* If user wants to upload document, he or she will have to login to the system first.

### Main Flow:

1. Use case begins when Actor clicks on file upload and then click on upload
2. The system will direct actor to the upload page of the system.
3. Upload has different parts upload view and download document.
4. User can upload by clicking on document.
5. User can delete document by clicking on delete button.

### Alternative Flows:

1(a) - Actor will not click on the “upload file” and then “upload file page menu”

### Exception:

1. Internet connection is not working.

“*Your internet connection is not working. Please connect to internet and try again.*”

### Post Condition:

User will upload document on general share and he will be able to download by using this use case

## 4.7.3.5 UC05: View, UPDATE & download (dOCUMENT details)

### Brief Description:

This use case will allow to admin teachers view update and download any document. Student can download their assignment and related document.

### Actors:

* Admin
* Teachers
* Student

### Pre- Condition:

In order to view and download document details, Actor must have access to the admin account of the system. He will have to login to the system.

### Main Flow:

1. The use case starts when user will login to the system (If not logged in).
2. User will click on the menu name “View” in the navigation bar, then “View document”.
3. System will direct user to the View & download document.
4. User can view and download all the above mentioned details about the documents here.
5. Admin Teachers can Edit/update these details if required.
6. For Update Admin will click on edit button and make changes in the textbox appear.
7. For save the changes he made, he will click on the “Save Changes” button at the bottom of the page.
8. System will prompt user “Details Successfully Updated”.

### Alternative Flows:

1(a) - Admin will not click on the “View update & download” Link in navigation bar.

7(a) - System will prompts the user for any missing field.

7(b) - If added information is not correct, system will alert for valid information.

### Exception:

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

### Post Condition:

Admin can view and update document details after complete and successful completion of this use case.

## 4.7.3.6 UC06: Sharing Document

### Brief Description:

This use case elaborate the sharing document by which user will share the document on social media or other these task are complete according to the assigned privileges.

### Actors:

* Admin
* Teachers
* Student

## Pre-Conditions:

* If user want to share document he must have to login into the system and have a document for sharing.

### Main Flow:

1. User select the document hi want to share.
2. Share screen will be Appears.
3. System will check the permission on this file if the permission will be accurate then share otherwise give an error message.
4. System will authenticate the user and give privileges according to user type.
5. The system will redirect the user to home page.
6. As user is successfully share document use case ends here.

### Alternative Flows:

If information is not correct, system will alert for valid information.

### Exceptions:

1. In case Database server is not responding, system will display the message:

*“The database server is down. Please check back after a few minutes.”*

1. In case web server is down, system will display the message:

*“The web server is down. Please check back after a few minutes.”*

### Post Condition:

None

## 4.7.3.7 UC07: Dicussion (Student and teachers)

### Brief Description:

This Use case allows user to chat with each other. E.g. student can communicate with Teachers.

### Actors:

* Teachers
* Students

### Pre- Condition:

If user wants to compose message, he will have to login to the system first.

### Main Flow:

1. Use case begins when Actor clicks on discussion and then click on Compose message.
2. The system will direct to the Compose message page.
3. There will be following sections on the page. To, From Body.
4. User will enter his recipient Name. E.g.Ahmad Mohsin.
5. User will enter Subject of the message.
6. User will compose message in the bottom box of page.
7. On pressing send button user can send mail to the recipient he entered.
8. By using Clear button Actor can clear all fields.

### Alternative Flows:

1(a) - Actor will not click on the “Discussion” link.

### Exception:

1. Internet connection is not working.

“*Your internet connection is not working. Please connect to internet and try again.*”

### Post Condition:

Actors/Users will able to send message to anyone.

## 4.7.3.8 UC08: Logout

### Brief Description:

This Use case allows users to logout from the System.

### Actors:

* Admin
* Teachers
* Student

### Pre- Condition:

The user must be logged in preconditions for this use case.

### Main Flow:

1. Use case begins when Actor/User clicks on “Logout” link.
2. The system will move the user session and the user is logged out.

### Alternative Flows:

1(a) - Actor will not click on the “Logout” link.

### Post Condition:

Actors/Users will be logged out of the System.

**4.7.3.09 UC09 Administrator Login**

**Brief description:**

This use case allow admin to login to system

**Actors:** system admin

**Pre-condition**

Admin has required credentials to login

**Main Flow:**

1. Admin goes to web interface of the system
2. Admin types his admin-id in username field
3. Admin shall type his password in the password field
4. Admin shall press submit button
5. System shall authenticate ID and Password
6. System shall take admin to main dashboard

**Alternative Flow**

1. System find no such Admin-id and password in the database.
2. Admin enter wrong user name and password

**Exceptions:**

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

**Post-condition(s):**

Admin is successfully logged into system

## 4.7.3.10 UC10: Add Teachers

### Brief Description:

This use case will allow Admin to add details about teachers. Details include Name, address, phone number, email, description etc.

### Actors:

* Admin

### Pre- Condition:

In order to add teachers details, Actor must have access to the admin account of the system. He will have to login to the admin panel of the system.

### Main Flow:

1. The use case starts when Admin will login to the system.
2. Admin will click on the menu name “Add” in the navigation bar, then “Add teachers Details”.
3. System will direct Admin to the Add teachers Details page.
4. Admin will enter name of the teachers.
5. Admin will enter phone number of teachers.
6. Admin will enter email address of teachers.
7. Admin will enter address of teachers.
8. Admin will click on to the “Add teachers” Button.
9. System will prompt Teachers “Details Successfully Added”.

### Alternative Flows:

1(a) - Admin will not click on the Add teachers Link in navigation bar.

8(a) - System will prompts the user for any missing field.

8(b) - If added information is not correct, system will alert for valid information.

8(c) - Admin will clear fields by clicking on “clear” button.

### Exception:

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

### Post Condition:

Admin can add teachers Details after complete and successful completion of this use case.

## 4.7.3.11 UC11: Add STUDENT

### Brief Description:

This use case will allow Admin to add details about student. Details include Name, address, phone number, email, project, description etc.

### Actors:

* Admin

### Pre- Condition:

In order to add student details, Actor must have access to the admin account of the system. He admin will have to login to the admin panel of the system.

### Main Flow:

1. The use case starts when Admin will login to the system (If not logged in).
2. Admin will click on the menu name “Add” in the navigation bar, then “Add student”.
3. System will direct Admin to the Add student page.
4. Admin will enter name of the student.
5. Admin will enter phone number of student.
6. Admin will enter email address of student.
7. Admin will enter address of student.
8. Admin will enter project student came with and all related details.
9. Admin will click on to the “Add student” Button.
10. System will prompt user “student Details Successfully Added”.

### Alternative Flows:

1(a) - Admin will not click on the Add student Link in navigation bar.

8(a) - System will prompts the user for any missing field.

8(b) - If added information is not correct, system will alert for valid information.

8(c) - Admin will clear fields by clicking on “clear” button.

### Exception:

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

### Post Condition:

Admin can add student details after complete and successful completion of this use case.

## 4.7.3.12 UC012: Delete DOCUMENT

### Brief Description:

This use case will allow Admin and teachers to delete document.

### Actors:

* Admin

### Pre- Condition:

In order to delete document details, user must have access to the admin, account of the system. He will have to login to the admin panel of the system.

### Main Flow:

1. The use case starts when Admin and teachers will login to the system.
2. Admin will click on the menu name “Delete” in the navigation bar, then “Delete document”.
3. System will direct Admin to the delete document file.
4. A list of document will appear on the page.
5. There is a delete button on every field in the list.
6. By clicking on “Delete” button admin can delete the document.
7. System will prompt admin “Are you sure you want to delete this document?”
8. By clicking “Yes” system will prompt again “By deleting this document you will lose all the information about this document. Are you sure you want to delete this?”
9. Pressing “Yes” again will delete the documents from database.

### Alternative Flows:

1(a) – User will not click on the “Delete” Link in navigation bar.

7(a) – If user will press “No” system will not delete the document.

8(a) – If Admin will press “No” system will not delete the document.

### Exception:

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

### Post Condition:

User can delete documents details after complete and successful completion of this use case.

## 4.7.3.13 UC013: Delete teachers details

### Brief Description:

This use case will allow Admin to delete details about teachers.

### Actors:

* Admin

### Pre- Condition:

In order to delete teachers details, Actor must have access to the admin account of the system. He will have to login to the admin panel of the system.

### Main Flow:

1. The use case starts when Admin will login to the system (If not logged in).
2. Admin will click on the menu name “Delete” in the navigation bar, then “Delete teachers”.
3. System will direct Admin to the delete teacher’s page.
4. A list of teachers will appear on the page.
5. There is a delete button on every field in the list.
6. By clicking on “Delete” button admin can delete the teachers.
7. System will prompt admin “Are you sure you want to delete this teachers?”
8. By clicking “Yes” system will prompt again “By deleting this teachers you will lose all the information about this teacher’s r. Are you sure you want to delete this?”
9. Pressing “Yes” again will delete the teachers from database.

### Alternative Flows:

2(a) - Admin will not click on the “Delete teachers” Link in navigation bar.

7(a) – If Admin will press “No” system will not delete the teachers.

8(a) – If Admin will press “No” system will not delete the teachers.

### Exception:

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

### Post Condition:

Admin can delete teacher’s details after complete and successful completion of this use case.

## 4.7.3.14 UC14: Delete STUDENT DETAILS

### Brief Description:

This use case will allow Admin to delete details about student.

### Actors:

* Admin

### Pre- Condition:

In order to delete student details, Actor must have access to the admin account of the system. He (admin) will have to login to the admin panel of the system.

### Main Flow:

1. The use case starts when Admin will login to the system (If not logged in).
2. Admin will click on the menu name “Delete” in the navigation bar, then “Delete student”.
3. System will direct Admin to the delete student page.
4. A list of student members will appear on the page.
5. There is a delete button on every field in the list.
6. By clicking on “Delete” button admin can delete the student.
7. System will prompt admin “Are you sure you want to delete this student member?”
8. By clicking “Yes” system will prompt again “By deleting this student you will lose all the information about this student member. Are you sure you want to delete this?”
9. Pressing “Yes” again will delete the student from database.

### Alternative Flows:

2(a) - Admin will not click on the “Delete student” Link in navigation bar.

7(a) – If Admin will press “No” system will not delete the student.

8(a) – If Admin will press “No” system will not delete the student.

### Exception:

1. In case Database server is not responding, system will display the message:

“The database server is down. Please check back after a few minutes.”

1. In case web server is down, system will display the message:

“The web server is down. Please check back after a few minutes.”

### Post Condition:

Admin can delete student details after complete and successful completion of this use case.

**4.8 Traceability Matrix**

A traceability matrix is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to-many relationship comparison.

**Requirement Traceability Matrix – Parameters include**

* Requirement ID.
* Risks.
* Requirement Type and Description.
* Trace to design specification.
* Unit test cases.
* Integration test cases.
* System test cases.
* User acceptance test cases.

**4.8.1 FR01: USER LOGIN**

The following tables describe the traceability matrix for the student and faculty login functional requirement.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req-No | Actor | Functional Requirement | UC# | Non Functional Requirement | GUI # |
| FR01-01 | Faculty  Student | System shall allow user to login | UC01 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR01-02 | Faculty  Student | System shall get username from user | UC01 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR01-03 | Faculty  Student | System shall get password from allow user | UC01 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR01-04 | Faculty  Student | System shall authenticate user when he/she submits username and password on login button | UC01 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR01-05 | Faculty  Student | System shall allow user to reset the his username name password when he/she clicks on reset button | UC01 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR01-06 | Faculty  Student | System shall allow user to remember his username and password by entering into the checkbox. | UC01 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |

**4.8.2 FR02: USER CAN CHANGE HIS PASSWORD**

The following traceability matrix describe the steps on how to change his password

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req-No | Actor | Functional Requirement | UC# | Non Functional Requirement | GUI # |
| FR02-01 | Admin Teachers Student | System shall get username from user | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |
| FR02-02 | Admin Teachers Student | System shall get old password from user | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |
| FR02-03 | Admin Teachers Student | System shall get new password from user | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |
| FR02-04 | Admin Teachers Student | System shall get new password confirmation password from user | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |
| FR02-05 | Admin Teachers Student | System shall check does the old password and does the new password and reconfirmation password match | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |
| FR02-06 | Admin Teachers Student | System shall get maximum length of password at least character | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |
| FR02-07 | Admin Teachers Student | System shall check the length of password | UC02 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR01-05  NFR01-06  NFR02-01  NFR02-02  NFR02-03  NFR02-04 | 2 |

**4.8.4 FR03: CUSTOMIZABLE PROFILE**

The following traceability matrix describe the setting or customization of profile setting

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req No | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR03-01 | Admin Teachers Student | System shall allow user to customize his profile | UC03 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 3 |
| FR03-02 | Admin Teachers Student |  | UC03 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 3 |
| FR03-03 | Admin Teachers Student | System shall allow user turn on notification for mobile phone | UC03 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 3 |
| FR03-04 | Admin Teachers Student | System shall allow user turn off notification for the mobile phone | UC03 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 3 |
| FR03-05 | Admin Teachers Student | System shall allow user to manage other services of the system | UC03 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 3 |

**4.8.4 FR04: UPLOAD FILE**

In this traceability matrix describe about the file uploading

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req No | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR04-01 | Teachers Student  Admin | System shall allow to user upload file | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-02 | Teachers Student  Admin | System shall allow to check the format of document | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-03 | Teachers Student  Admin | System shall allow to check file size | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-04 | Teachers Student  Admin | System shall allow to user upload file when click on upload button | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-05 | Teachers Student  Admin | System shall authenticate the file size and format | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-06 | Teachers Student  Admin | System shall give an error if the file size greater than given limits | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-07 | Teachers Student  Admin | System shall give an error if the file format is incorrect | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR04-08 | Teachers Student  Admin | System shall allow to user rename the file name | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |

**4.8.5 FR05: DOWNLOAD FILE**

In this traceability matrix discuss about the file downloading

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req No | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR05-01 | Teachers  Student  Admin | System shall allow to user when click on download button | UC05 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR05-02 | Teachers  Student  Admin | System shall give an error when the click button not work proper | UC05 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR05-03 | Teachers  Student  Admin | System shall give message if file access denied for invalid user | UC05 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |

**4.8.06 FR06: SHARING FILE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR06-01 | Student  Teachers | System shall allow to user when click on share button | UC06 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR06-02 | Student  Teachers | System shall give an error when user can’t select any document for sharing | UC06 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR06-03 | Student  Teachers | System shall authenticate the file sharing media and then share it | UC06 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR06-04 | Student  Teachers | System shall check if no authenticate the give error message “there is something went wrong ” | UC06 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR06-05 | Student  Teachers | System shall allow to check file access type | UC06 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |

**4.8.07 FR07: NOTIFICATION**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR07-01 | Student  Teachers | System shall allow to send notification when specified user upload file | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR07-02 | Student  Teachers | System shall allow to user turn on notification on new update when click on turn on button | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR07-03 | Student  Teachers | System shall allow to user turn off notification when click on turn off button | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR07-04 | Student  Teachers | System shall allow to user customizes their notification when click on advance option | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |
| FR07-05 | Student  Teachers | System shall allow to user when click on reset as default option | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR01-04 | 4 |

**4.8.08 FR09: ADMIN LOGIN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR09-01 | Admin | System shall allow admin to login | UC09 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR09-02 | Admin | System shall get password from admin | UC09 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR09-03 | Admin | System shall authenticate admin when he/she submits username and password on login button | UC09 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR09-04 | Admin | System shall allow admin to reset the his username name password when he/she clicks on reset button | UC09 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |
| FR09-05 | Admin | System shall allow admin to remember his username and password by entering into the checkbox. | UC09 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 1 |

**4.8.09 FR11: Add Student**

In this table explain creation of student account.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR11-01 | Admin | System shall allow admin to create student account | UC11 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5 |
| FR11-02 | Admin | System shall allow to add student id in text box | UC11 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5 |
| FR11-03 | Admin | System shall allow to admin add user name into the textbox | UC11 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5 |
| FR11-04 | Admin | System shall allow to admin select the class of the student from dropdown menu | UC11 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5 |
| FR11-05 | Admin | System shall allow to admin when press reset button clear all the fields | UC11 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5 |
| FR11-06 | Admin | System shall allow to admin when click on add button system give message student added successfully | UC11 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5 |

**4.8.10 FR12: Add Faculty**

In this table explain creation of Faculty account.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR12-01 | Admin | System shall allow admin to create faculty account | UC12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 6 |
| FR12-02 | Admin | System shall allow to add department from dropdown menu | UC12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 6 |
| FR12-03 | Admin | System shall allow to admin add 1st name  of faculty in field one | UC12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 6 |
| FR12-04 | Admin | System shall allow to admin add 2nd name of faculty | UC12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 6 |
| FR12-05 | Admin | System shall allow to admin when press reset button clear all the fields | UC12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 6 |
| FR12-06 | Admin | System shall allow to admin when click on add button system give message Faculty added successfully | UC12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 6 |

**4.8.11 FR13: MANAGE THE USER**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR13-01 | Admin | System shall allow user create account | UC13 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5,6 |
| FR13-02 | Admin | System shall allow to user add upload document in the system | UC13 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5,6 |
| FR13-03 | Admin | System shall allow to user delete specified document | UC13 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 5,6 |

**4.8.12 FR11: LOGIN ROLES**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR11-01 | Admin | System shall allow admin to create roles | UC11,12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 7 |
| FR11-02 | Admin | System shall allow to admin edit roles | UC11,12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 7 |
| FR11-03 | Admin | System shall allow to admin delete roles | UC11,12 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 7 |

**4.8.13 FR14: LOGIN ACTIVITY TRACK**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR14-01 | Admin | System shall allow to admin keep track of user |  | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 8 |
| FR14-02 | Admin | System shall allow to admin manage login activity |  | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 8 |
| FR14-03 | Admin | System shall allow to admin see activity log of user |  | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 8 |

**4.8.14 FR15: UPLOAD DOCUMENT**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Req no | User | Functional requirement | UC # | Nonfunctional requirement | GUI # |
| FR15-01 | Admin | System shall allow admin upload file and folder | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 9 |
| FR15-02 | Admin | System shall allow to admin upload any type of information upload and download | UC04 | NFR01-01  NFR01-02  NFR01-03  NFR01-04  NFR02-02  NFR02-03  NFR02-04 | 9 |

**4.09 Hierarchical chart**

An organizational chart, also known as org chart, hierarchy chart, organigram, organization chart, is a diagram mainly used to show the structure of an organization and the relationships in the organization. Creately provides tools to draw organizational charts with amazing ease.

C:\Users\zubairsaif\Downloads\Flow Chart (2).png

**4.10 System Architecture Diagram**

Software Architecture Design is a crucial step for software and application developers to describe the basic software structure by separating functional areas into layers. It depicts how a typical software system might interact with its users, external systems, data sources, and services.



**4.11 DATA FLOW DIAGRAM**

Data Flow Diagrams represent one of the most ingenious tools used for structured analysis. A Data Flow Diagram or DFD as it is shortly called is also known as a bubble chart. It has the purpose of clarifying system requirements and identifying major transformations that will become programs in System design. It is the major starting point in the design phase that functionally decomposes the requirements specifications down to the lowest level of detail.

A DFD consists of a series of bubbles joined by lines. The bubble represents data transformation and lines represent data flow in the system. In the normal convention, a DFD has four major symbols

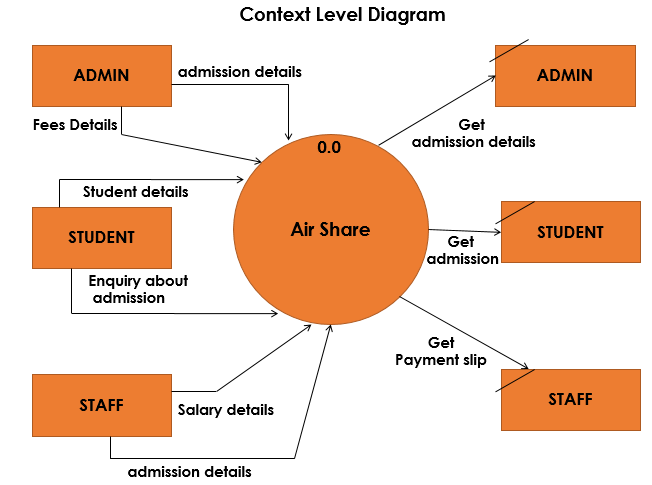
## 4.11.1 Steps to Construct a DFD

Four steps are commonly used to construct a DFD

* Process should be named and numbered for easy reference.
* Each name should be representative of the process.
* The direction of flow is from top to bottom and from left to right.
* When a process is exploded into lower level details they are numbered.
* The names of data stores, source and destination are written in capital.

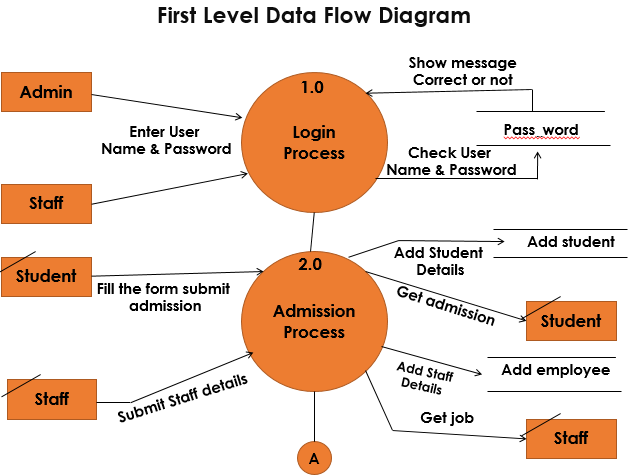
**4.11.2 Context Level**

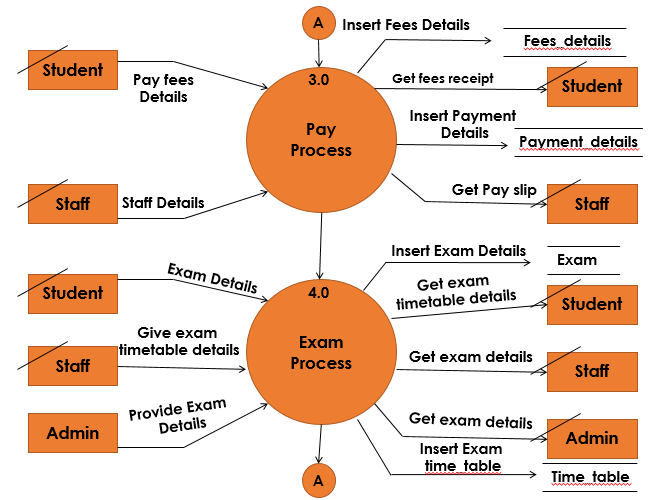
The DFD at the simplest level is referred to as the ‘**Zero Level DFD’** or in the Simple words a ‘Context Analysis Diagram’. These are expanded level by level each explaining its process in detail. Processes are numbered for easy identification and are normally labeled in Block letters. Each data flow is labeled for easy understanding.



**4.11.3 DFD Level 1:**

The next stage is to create the Level 1 Data Flow Diagram. This highlights the main functions carried out by the system. As a rule, we try to describe the system using between two and four functions - two being a simple system. This enables us to keep the model manageable on screen or paper.





**SYSTEM DESIGN**

**4.1 INPUT DESIGN**

Input design is the process of converting the user originated input to a computer based format. The design decision for handling input specify hoe data are accepted for computer processing .Input design is a part of overall system design that needs careful attention.

**OUTPUT DESIGN**

One of the most important features of a system for user is the output it produces. Output design should improve the systems relationship with the user and help in decision making. Computer output is a process that involves designing necessary output that have to be given to various users according to their requirements. Efficient, intelligible output design should improve the system relationship with the user and help and in decision making. A major form of output is the hardcopy from the printer. The output devices are selected by considering the response time requirements, print quickly etc. The print formats and editing for the final printout are very much considered during output design.

The objective of output design is to define the controls and format of all printed documents and reports and screens that will be produced by the system. The output is the most important and direct source of information to the user. For many end users output is the main reason for developing the system and the basis on which they will evaluate the usefulness of the application. Output generally refers to the system results. The output of the system is designed so as to include a number of reports. Reports reflect the output design. Output design is an ongoing activity, which start during study phase itself. Output generally refers to the results and information data are generated by the system. It can be in the form of operational documents and reports.

**Objectives of Output Design**

* Design output to serve the intended purpose
* deliver appropriate quantity of output
* choose the right output method
* Provide output on time

**DATABASE DESIGN**

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data in the database is safe and easily accessed.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

* Data Integrity
* Data independence

Normalization is the process of decomposing the attributes in an application, which results in a set of tables with very simple structure. The purpose of normalization is to make tables as simple as possible. Normalization is carried out in this system for the following reasons.

* To structure the data so that there is no repetition of data, this helps in saving space.
* To permit simple retrieval of data in response to query and report request.
* To simplify the maintenance of the data through updates, insertions, deletions.
* To reduce the need to restructure or reorganize data which new application requirements arise

**Relational Database Management System (RDBMS)**

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, each of which is assigned a unique name. A row in a tale represents a set of related values.

Relations Domains & Attributes: A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values. Every value in a relation is atomic, i.e. Not decomposable.

Relationships Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.

* Entity Integrity enforces that no Primary Key can have null values.
* Referential Integrity enforces that no Primary Key can have null values.
* Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other keys are Super Key and Candidate Keys.
* Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity.

NORMALIZATION

As the name implies, it denoted putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy, which proves to be a great burden on the computer resources.

Normalization is the systematic technique of transforming data subject to a whole range of file maintenance problem into an organized data free from such problem Detecting tables through a number of levels of normalization.

It is carried out in four different steps

1. Represent the unnormalized table or relation.

2. Transform the unnormalized table to the First Normal Form (1NF).

3. Transform of First Normal Form into Second Normal Form (2NF).

4. Transformation of Second Normal Form into the Third Normal Form (3NF).

**These include:**

* Normalize the data.
* Choose proper names for the tables and columns.
* Choose the proper name for the data.

**First Normal Form:**

The first step is to put the data into First Normal Form. This can be done by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the project. This eliminated repeating groups of data.

**Second Normal Form:**

This step helps in taking out data that is only dependent on a part of the key.

**Third Normal Form:**

This step is taken to get rid of anything that does not depend entirely on the Primary Key.

**TABLE DESIGN**

**Graphical User Interface**

A graphical user interface or GUI, pronounced ˈ**ɡuːi** ‘“**gooey**" is a type of interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, as opposed to text-based interfaces, typed command labels or text navigation.

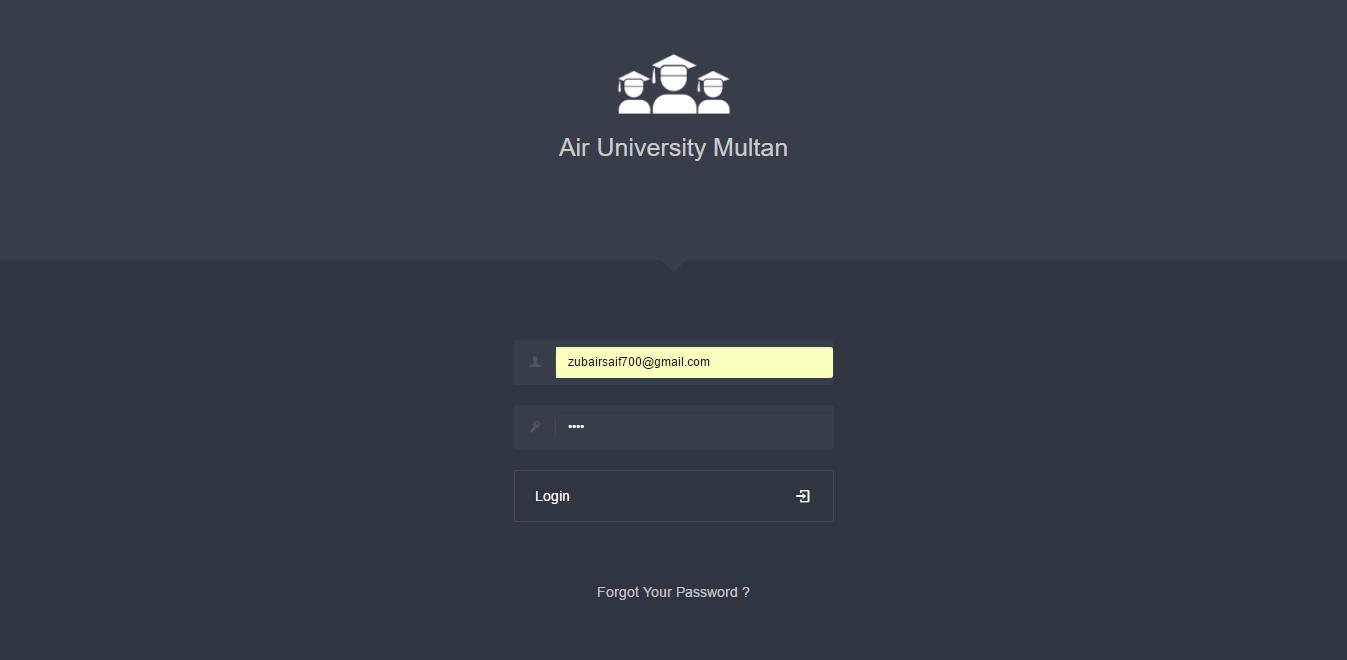
The user interface of **Air Share** is kept fairly simple focusing on making the system’s use fast and simple task.

Following are the goals considered specifically for the user interface

* Easy to learn
* Consistency throughout the system
* Use of guiding icons
* Proper use of labels
* Avoiding complexity

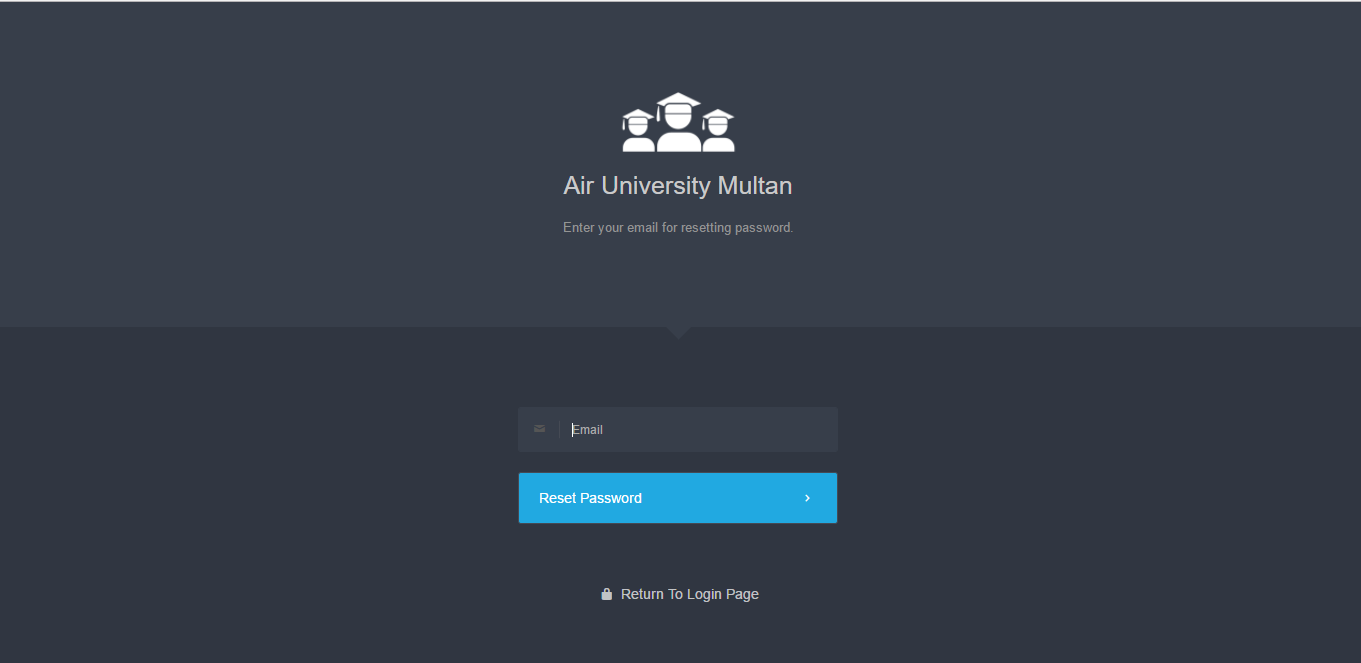
**Admin Login**

Below the figure a screen where admin is allowed to login to system by entering his username and password



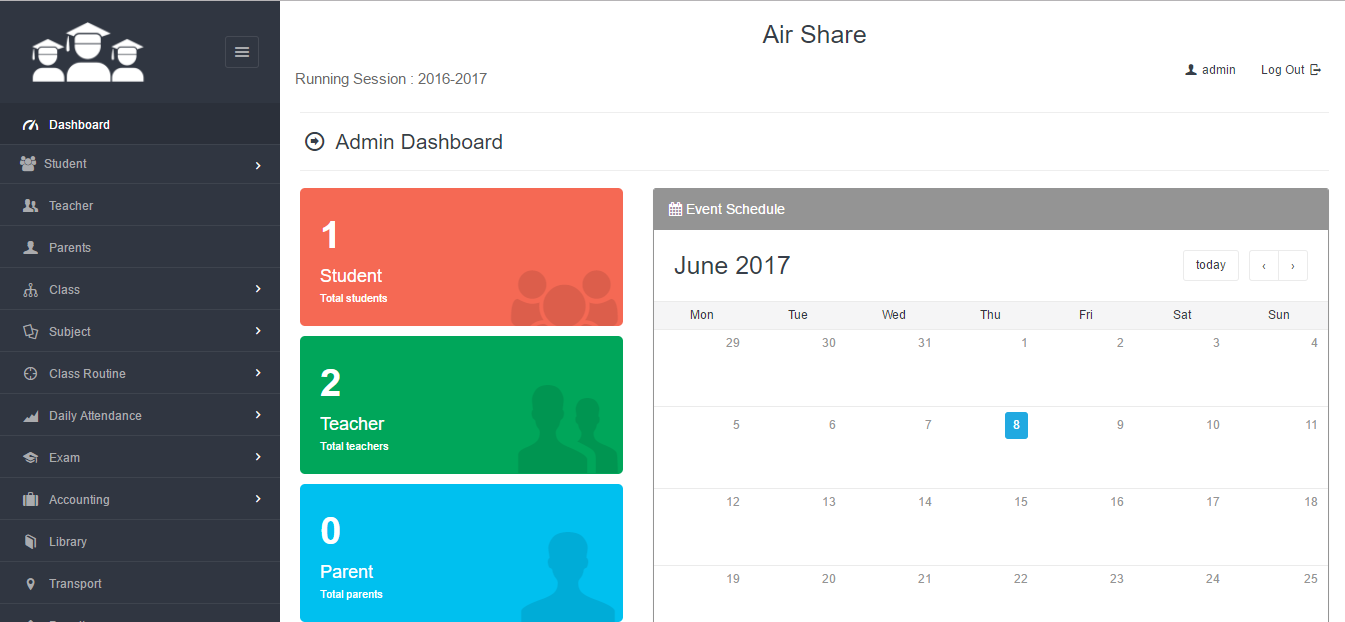
**Forget Password**

In this screenshot user or admin forget your password



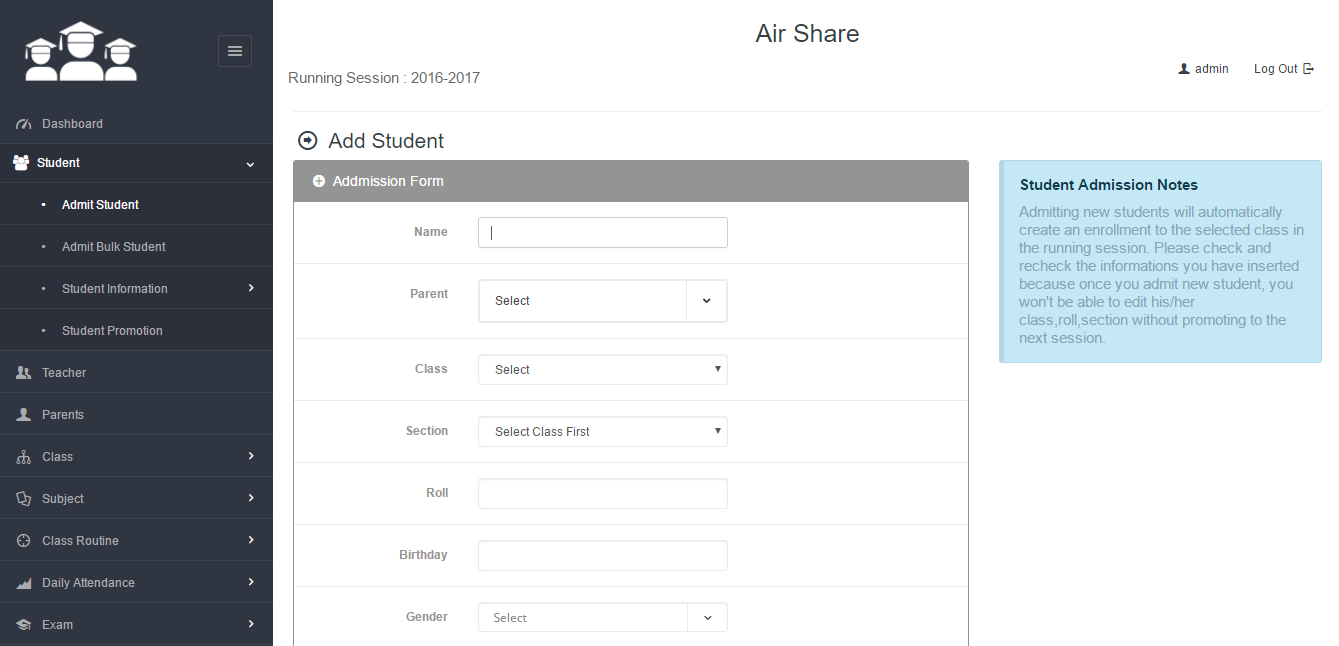
**C Panel**

The below given image is the screenshot for the c-panel dashboard. The dashboard has several option where admin can navigate include history manage user account view user, add classes, add subject, event, profile and logout option and much more.



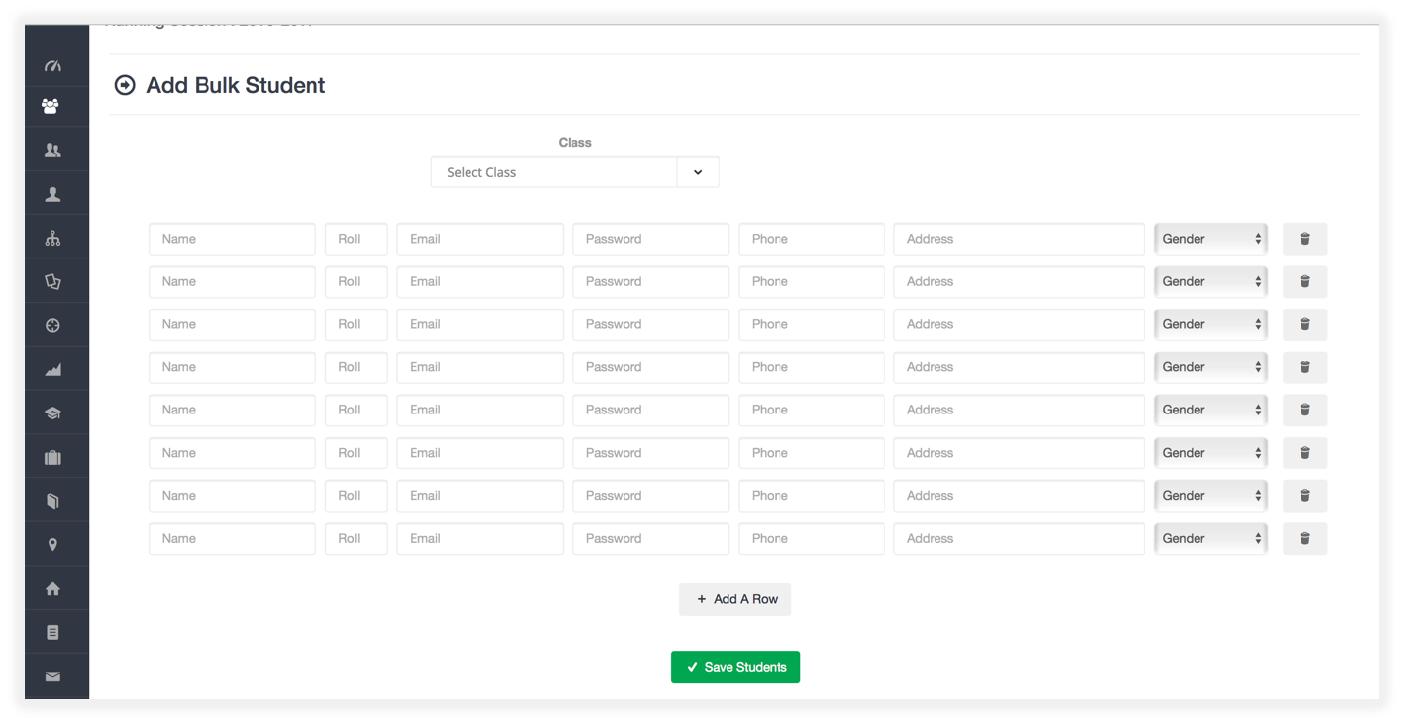
**Student Admission**

Admitting new students to the school can be done from ‘admit student’ submenu under ‘student’ in the navigation. For adding a student, admin will need to fill up the information required and provided by the admission form. It should be kept in mind that, admitting new student will automatically create an enrollment in the running session for the selected class. Check and recheck the information you have inserted while adding student because once you admit him/her to a class, you will not be able to change his/her class without promoting him/her to the next session.



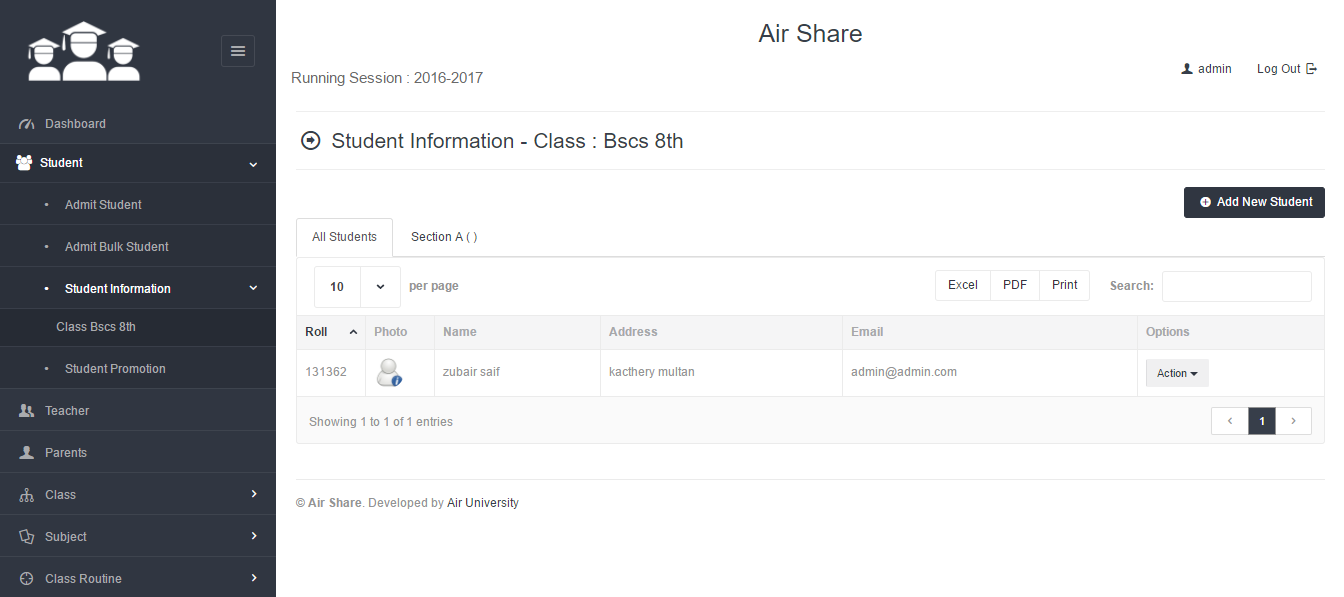
**Bulk Student Admission**

For admitting bulk students (adding multiple students at a time) you will need to go to the ‘admit bulk student’ submenu under the menu ‘student’. Firstly, select class and section and then fill up the information for each student you want to admit. Make sure to put at least name, email and password for each student you want to admit. You can add rows for adding more as much as you need.



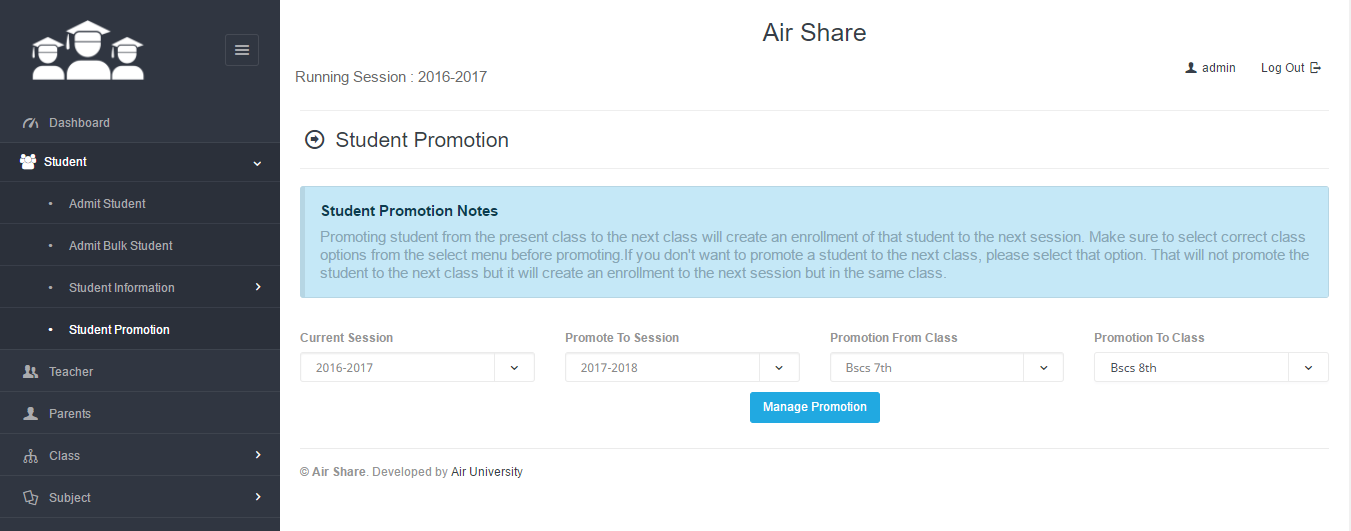
**Student Information**

Student information are shown class wise under the same menu ‘student’. From the page you can get section wise student’s information, his/her profile info, mark sheet. You can also edit student’s basic information form the action button associated with each student.



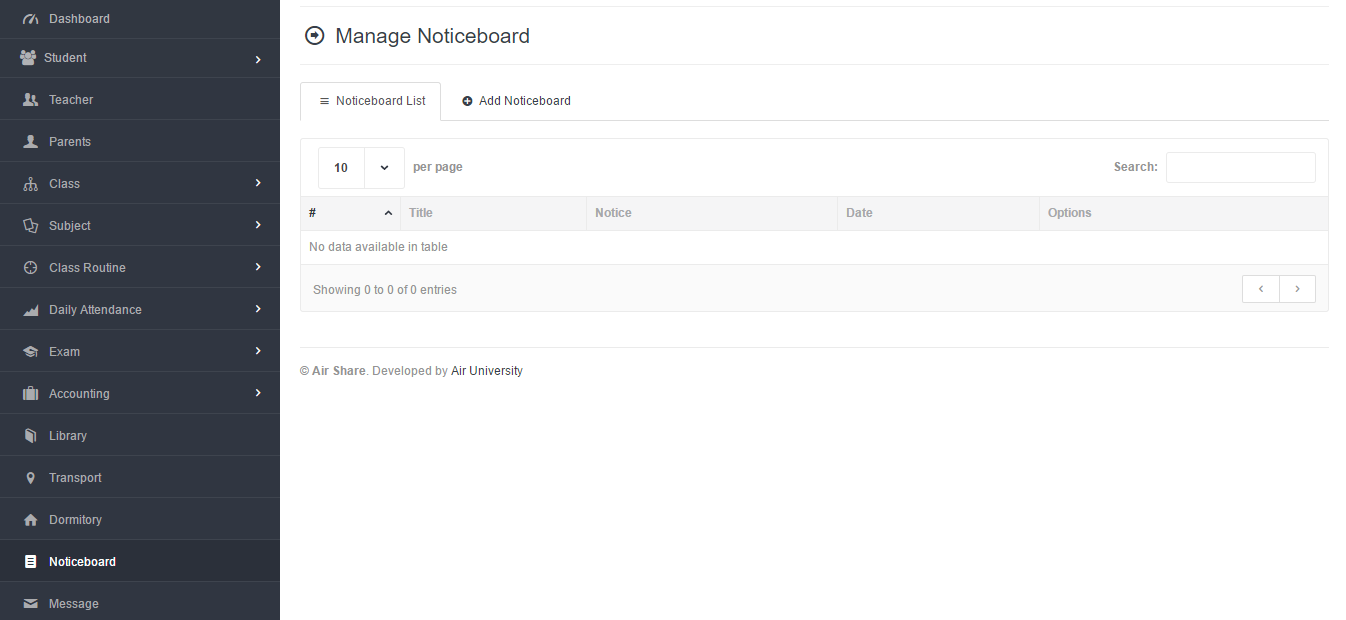
**Student Promotion**

This module is designed to promote students of a particular class to the next class in the next session. ‘Student Promotion’ submenu will be found under the menu ‘student’ in the admin panel. To promote students to the next class, admin needs to select the present class of the students and the class where he/she wants to promote the students. The present session which has been already set from the system settings will be selected automatically as the present session and the next session will be automatically calculated by the software.



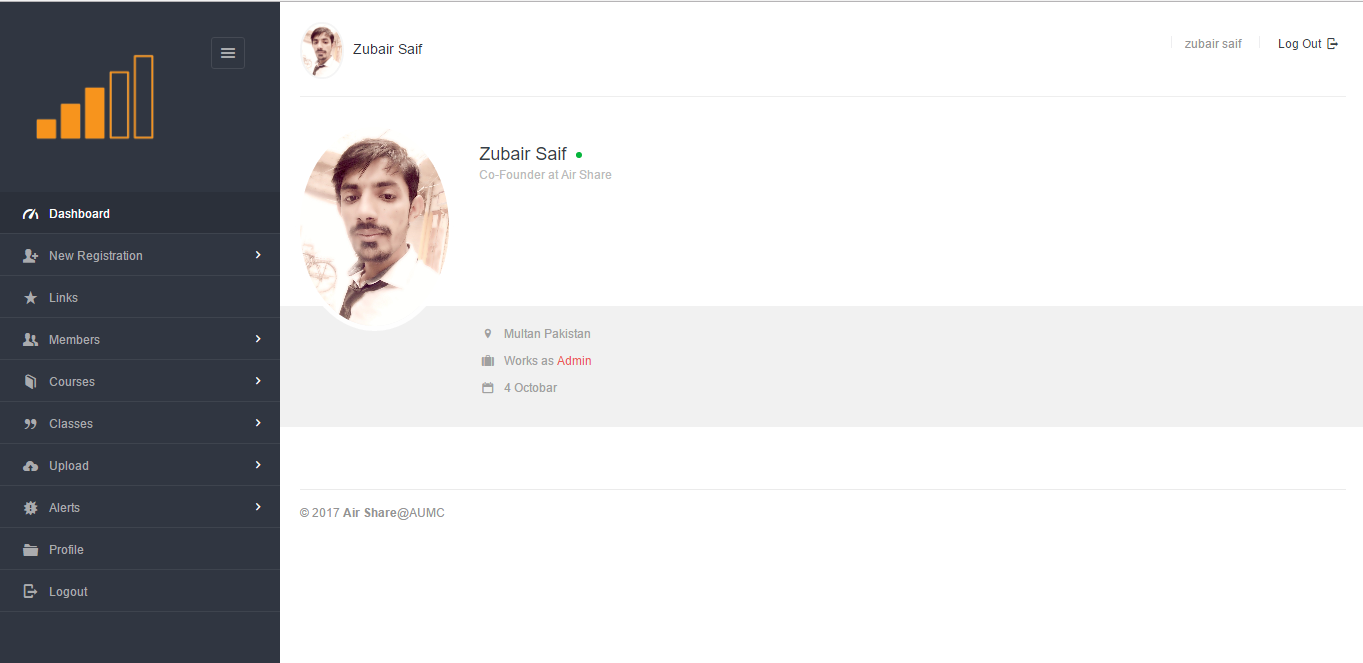
**Add Notice**

In this screenshot where admin can add event, or any activity about the university



**Profile Setting**

In this screenshot describe user profile setting



Conclusion:   
All most all of the Major functionalities have been achieved. In Future, System shall be made more flexible, maintainable and more admin duties will be automated.