Software Requirements Specification

Software Requirement Engineering

School Management System

Salman Ahmed Khan 19K-1043

Sadeem Sattar 19K-1102

Abdullah Tilal Khan19K-1103

Presented To: Dr. Abdul Aziz

FAST NUCES KARACHI

December 11, 2020

Table OF Content

Table of Contents

[1. Introduction 1](#_Toc58939633)

[1.1 Purpose 1](#_Toc58939634)

[1.2 Product Scope 1](#_Toc58939635)

[1.3 Document Conventions 1](#_Toc58939636)

[1.4 Reference 1](#_Toc58939637)

[1.5 Overview 1](#_Toc58939638)

[2. Overall Description 2](#_Toc58939639)

[2.1 Product Perspective 2](#_Toc58939640)

[2.2 Product Functions 2](#_Toc58939641)

[2.3 User Characteristics 3](#_Toc58939642)

[2.4 General Constraints 4](#_Toc58939643)

[2.4.1 Technology Constraints 4](#_Toc58939644)

[2.4.2 Interface Constraints 4](#_Toc58939645)

[2.4.3 Safety and Security Constraints 4](#_Toc58939646)

[2.5 Assumptions and Dependencies 4](#_Toc58939647)

[3. External Interface Requirements 4](#_Toc58939648)

[3.1 User Interfaces 4](#_Toc58939649)

[3.2 Hardware Interfaces 4](#_Toc58939650)

[3.3 Software Interfaces 5](#_Toc58939651)

[4. System Functional Requirement 5](#_Toc58939652)

[4.1 Login 5](#_Toc58939653)

[4.1.1 Description 5](#_Toc58939654)

[4.1.2 Requirement detail 5](#_Toc58939655)

[4.2 Add Details 5](#_Toc58939656)

[4.2.1 Description 5](#_Toc58939657)

[4.2.2 Requirement detail 5](#_Toc58939658)

[4.3 View Details 5](#_Toc58939659)

[4.3.1 Description 5](#_Toc58939660)

[4.3.2 Requirement detail 6](#_Toc58939661)

[4.4 Add Announcement 6](#_Toc58939662)

[4.4.1 Description 6](#_Toc58939663)

[4.4.2 Requirement detail 6](#_Toc58939664)

[4.5 View Announcement 6](#_Toc58939665)

[4.5.1 Description 6](#_Toc58939666)

[4.5.2 Requirement detail 6](#_Toc58939667)

[4.6 Set Timetable 6](#_Toc58939668)

[4.6.1 Description 6](#_Toc58939669)

[4.6.2 Requirement detail 7](#_Toc58939670)

[4.7 View Timetable 7](#_Toc58939671)

[4.7.1 Description 7](#_Toc58939672)

[4.7.2 Requirement detail 7](#_Toc58939673)

[4.8 Add Attendance 7](#_Toc58939674)

[4.8.1 Description 7](#_Toc58939675)

[4.8.2 Requirement detail 7](#_Toc58939676)

[4.9 Update Attendance 7](#_Toc58939677)

[4.9.1 Description 7](#_Toc58939678)

[4.9.2 Requirement detail 8](#_Toc58939679)

[4.10 View Attendance 8](#_Toc58939680)

[4.10.1 Description 8](#_Toc58939681)

[4.10.2 Requirement detail 8](#_Toc58939682)

[4.11 Add Marks 8](#_Toc58939683)

[4.11.1 Description 8](#_Toc58939684)

[4.11.2 Requirement detail 8](#_Toc58939685)

[4.12 Update Marks 8](#_Toc58939686)

[4.12.1 Description 8](#_Toc58939687)

[4.12.2 Requirement detail 9](#_Toc58939688)

[4.13 View Marks 9](#_Toc58939689)

[4.13.1 Description 9](#_Toc58939690)

[4.13.2 Requirement detail 9](#_Toc58939691)

[4.14 Generate Fee 9](#_Toc58939692)

[4.14.1 Description 9](#_Toc58939693)

[4.14.2 Requirement detail 9](#_Toc58939694)

[4.15 Print Fee Challan 9](#_Toc58939695)

[4.15.1 Description 9](#_Toc58939696)

[4.15.2 Requirement detail 9](#_Toc58939697)

[4.16 Submit Fee 10](#_Toc58939698)

[4.16.1 Description 10](#_Toc58939699)

[4.16.2 Requirement detail 10](#_Toc58939700)

[5. Other Nonfunctional Requirements 10](#_Toc58939701)

[5.1 Performance Requirements 10](#_Toc58939702)

[5.2 Security Requirements 10](#_Toc58939703)

[5.3 Software Quality Attributes 10](#_Toc58939704)

[5.3.1 Predictability 10](#_Toc58939705)

[5.3.2 Familiarity 11](#_Toc58939706)

[5.3.3 Generalizability 11](#_Toc58939707)

[*5.3.4* *Consistency* 11](#_Toc58939708)

[5.3.5 Dialogue Initiative 11](#_Toc58939709)

[5.3.6 Task Migration 11](#_Toc58939710)

[5.3.7 Substitutivity 11](#_Toc58939711)

[5.3.8 Responsiveness 11](#_Toc58939712)

[5.3.9 Recoverability 11](#_Toc58939713)

[6. Requirement Interaction Matrices 12](#_Toc58939714)

[7. Stakeholder Engagement Matrix 15](#_Toc58939715)

[8. Role Matrix 16](#_Toc58939716)

[9. Use Cases 16](#_Toc58939717)

# Introduction

## Purpose

The software School Management System is designed to facilitate the work of school management. The management system will provide ease to maintain the record of students and teachers. It will provide the day to day automated functionality of school management. It will be useful for administrator management of the institute to manage the record, for account department to generate the fees, for teachers to mark the whole activity of students and for students to view their progress.

## Product Scope

The Software is based on four users that are Admin, Account, Teacher and Student. The software system will be carryout the functionality of their work. It will provide the work efficient for the users of the system. The system will provide the login functionality in order to use the system. The SMS is designed to create an efficient management for students, admin, account and teachers.

## Document Conventions

The abbreviation of project name School Management System will be SMS. The higher priority requirement will be marked bold throughout the SRS.

## Reference

* *Software Requirement Engineering by Ian Somerville and Pete Sawyer*
* *Microsoft Access for database*
* *Lucid chart for Use-Cases*
* *C# for Desktop Application*

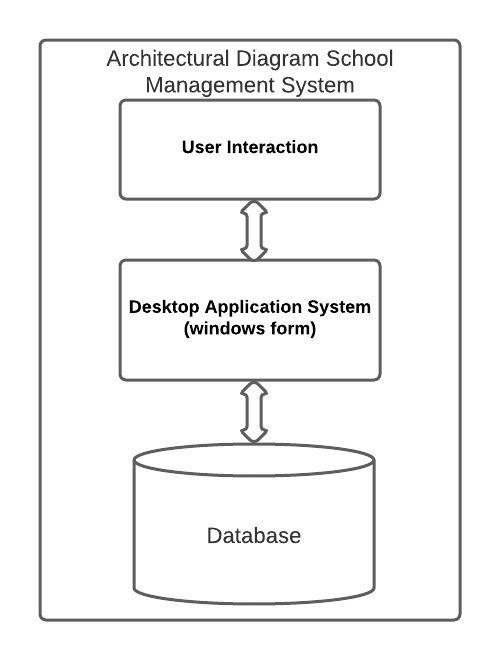
## Overview

*The document will contain the overall description, general constraints, users’ characteristics, specific requirements etc.*

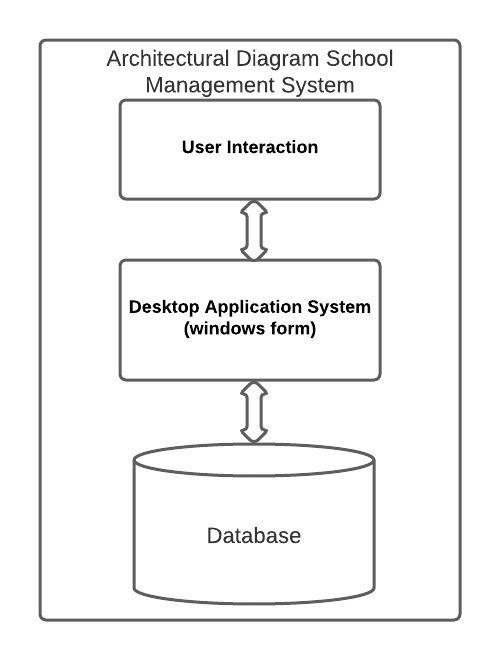
# Overall Description

## Product Perspective

The system SMS is a new system product that is going to replace the old paper based system. It will have the two layers that is the Database Layer and the Application layer through which the user will interact. The records of teachers and students will be recorded in the database and through it can be retrieved for the data process and functions.



Architectural design diagram of School Management System.



## Product Functions

There are four users of the system that are Admin, Account, Teacher and Student.

Functions available for Users are given below:-

**The process of Login is necessary for every user. It require an ID and a Password**

Admin:-

* Login with an ID and Password
* Add Details of new student
* Add Details of new teacher
* View the details of student and teacher
* Add Announcement that can be viewed as per selection that is for teacher, students or account or to every user
* View the announcement
* Set the timetable for every class and for teachers
* View the timetable
* Logout from the system

Account:-

* Login with an ID and Password
* View the details of student which include student-id, name, class, previous fee and fee status
* Add announcement that can only be viewed by specific student and Admin
* Generate the fee for individual student
* Submit the fee
* Logout from the system

Teacher:-

* Login with an ID and Password
* A teacher can view its details
* View the details of only those student to whom the teacher teach as per classes according to timetable
* The detail of student to view include student-id, name, class, attendance and marks
* Add the Attendance of the student
* Update the attendance of the student
* View the attendance of the student
* Add the marks of quiz, mid-term and finals of students
* Update the marks
* View the marks of students
* View the teacher timetable
* Add Announcement that can only be viewed by students and Admin
* View the Announcement set by that teacher and of Admin
* Logout from the system

Student:-

* Login with an ID and Password
* A student can view its details
* A student can view and print the fee challan in order to submit to the accounts
* A student can view the status of its fee
* A student can view its attendance set by teacher
* A student can view the timetable of its class
* A student can view the announcement set by teacher, Admin or Account
* A student can view its marks of quiz, mid-term and final
* Logout from the system

## User Characteristics

**Student:** They are the users studying in the school and they can only have read-only access

**Teacher:** They are the users who teaches to students of different classes and they can only add information related to student progress

**Account:** They are the user who are responsible for the financial check and balance of the students and provide them fee challan and submit their fee

**Admin:** They are the users who are responsible to handle all the details and record of teachers and students and management of the classes by providing the timetable

## General Constraints

### Technology Constraints

*The proposed system is a Desktop Application that will be implemented with C# and based on Windows forms. For the Database, the Microsoft Access database will be equipped to record the data/information.*

### Interface Constraints

*The system is a desktop application that it will work on Windows operating system only*

### Safety and Security Constraints

*The system is designed to be able authorize only the authentic user as mentioned above. Anonymous user will not be allowed to use the system. The system is safeguarded by a login mechanism.*

## Assumptions and Dependencies

The proposed system is to work only with the Windows operating system of version 7 and above. It is assumed that if any other operating system is used the system will failed to launch.

# External Interface Requirements

## User Interfaces

The system will provide the menus option and navigation buttons to reach the specific task for the user. The efficiency will be achieved and it will provide the help to user to navigate.

## Hardware Interfaces

The system is supported with Windows forms and it will only work with the Windows operating system of version 7 and above.

## Software Interfaces

The system will be able to efficiently communicate with the database to retrieve the data for the data process and use for its functions. Also, the data will be provided different for individual teachers and students, thus making it effective

# System Functional Requirement

The functional requirements of the system are as below.

## Login

### Description

The system will have the login functionality in-order to enter in the system. It is a high priority requirement.

### Requirement detail

*The system will asked the user to enter its ID which will be a string, a password which will be integer and choose from the selection box whether it is admin, account, teacher or student*

## Add Details

### Description

*The system will provide the functionality to add the details of teacher and student. This functionality is only provided to admin.*

### Requirement detail

*The functionality of Add details require name, an ID, home address, email address, Nationality, Religion and National ID Number that will be a string. It will also require phone number, password as integer. The system will also require the profile picture which will be in jpg, png or jpeg format. For the details of teacher, it will require qualification, education, experience and field of teaching all will be as string. For students, class field will be integer.*

## View Details

### Description

*The system will provide the functionality to view the details of teacher and student. This will be provided to admin, account, teacher and student. For teacher and student, their respective profile will be complete shown and Admin can also view their details. The account can also view the details of student only.*

### Requirement detail

*The complete details can be viewed by teacher, student and admin. Teacher can also view the details of student to the classes they teach by student id or by name. Teacher can only view their name, student id and class. For account, the student-id, class, phone number and email will be viewed. They can view it through searching by student-id.*

## Add Announcement

### Description

*The system will provide the functionality to add announcement which can only be accessed by teacher, admin and account. Through this functionality, user (teacher, admin, account) can post important announcement.*

### Requirement detail

*The admin can post the announcement by typing the message in a message box and selecting whether to send all students of classes or individual class. Also, teacher can post the announcement and account too.*

## View Announcement

### Description

*The users of the system can view the announcement set by admin, teacher and account.*

### Requirement detail

*The users can view the announcement by selecting the view announcement panel in menu and the entire announcement will be shown which include current and previous.*

## Set Timetable

### Description

*The system will provide the functionality to set the timetable for classes of student and for teachers. This can only be set by Admin.*

### Requirement detail

*The Admin can set the timetable for classes of student which include the selection class, time slot, teacher availability from teacher timetable. These all will be in string format except the class. When the timetable is set, the individual teacher timetable will be automatically updated in the database while setting the class timetable.*

## View Timetable

### Description

*The system will provide the functionality to view the timetable set by Admin. The functionality is accessed to Teachers and Student.*

### Requirement detail

*The timetable for students will include their class time slot and classes. For teachers, their will include the classes they will teach with their time slot. They can view the timetable by selecting the timetable panel from menu slot and their timetable will be shown.*

## Add Attendance

### Description

*The system will provide the functionality to add the attendance of student. This functionality is only provided to teachers.*

### Requirement detail

*The teacher will choose the class and timeslot all students name with student-id will be shown and a dropdown will be provided to select whether to present, absent or leave for every individual student. The attendance will be then added in the database.*

## Update Attendance

### Description

*The system will provide functionality to update the attendance of the student. This is only provided to teacher.*

### Requirement detail

*The teacher can view the attendance and then update it as per it require by changing the status of attendance of individual student and then save it. After that, the attendance will be updated in the database.*

## View Attendance

### Description

*The system will provide the functionality to view the attendance of student set by the teacher. This is provided to teacher and students.*

### Requirement detail

*The attendance can be viewed for every individual student by inputting the student-id of the student. The details of attendance will be retrieved from the database.*

## Add Marks

### Description

*The system will provide the functionality to add the marks of student which include quiz, mid-term and finals. This is only provided to teachers.*

### Requirement detail

*The teacher will select the student whom marks are to be added by inputting the student-id. The teacher can select whether to add quiz, mid-term or finals marks. Their total marks are already be adjusted by default. The teacher will input the marks which will be in float format. The marks will be added in the database of student.*

## Update Marks

### Description

*The system will allow the teacher to update the marks of student.*

### Requirement detail

*The teacher will input the student id in order to update the marks. It will first select the term and change the marks with previous one. The database will be updated.*

## View Marks

### Description

*The marks of student can be viewed by student and teacher.*

### Requirement detail

*The marks can be viewed by selecting the view marks from menu of student panel. For teacher, student-id will be required to view the marks of student.*

## Generate Fee

### Description

*The system will provide the functionality to account only to generate the fee of student.*

### Requirement detail

*The account will select student by inputting student-id, selecting the current and due date from date panel, generating a fee code and inputting the amount of fee. The fee will be added in database with a status of zero as not paid. If the due date is reach, then in next time the amount will be doubled automatically from system.*

## Print Fee Challan

### Description

*The system will allow student to view its fee and print it.*

### Requirement detail

*The student will view its fee from view fee panel from menu. The student will print the challan in order to submit the fee to account.*

## Submit Fee

### Description

*The system will provide the functionality to only account to submit the fee of the student.*

### Requirement detail

*The student will bring the amount and challan manually to the account department. The account will input the student-id, enter the amount and fee code. The status will be updated as one denoting paid status. The database of student fee will be updated and status will be one as paid.*

# Other Nonfunctional Requirements

## Performance Requirements

The system will not slow down while fetching the data from database. Also, the opening of panel pages from menu function will be under a second for individual functional requirements for all users according to their requirements maintaining the quality of the service of the system.

## Security Requirements

The system is safeguarded by a login functions to not allow the external users to login the system. It will provide the access to only the authorized users as security is fundamental requirement of the system. The login id and password of every user will be stored in the database.

## Software Quality Attributes

### Predictability

*User can predict by viewing the login page and home screen that the user click on the button, certain actions will be formed. So, user can predict the behavior.*

### Familiarity

*The metaphors used in project are real-world images. Thus, this creates the user familiar with the system.*

### Generalizability

*The system provides knowledge of the form fill similar to the paper based system.*

### *Consistency*

*All the color of buttons, text, Position of buttons, combo box, layout and appearance are consistent for all categories of users.*

### Dialogue Initiative

*The system will provide dialogue initiative when any field is missing or any input provide is incorrect. Also, user can do dialogue initiate by clicking on buttons in order to open the require page.*

### Task Migration

*The task of user (teacher) in attendance page is that the students will be marked present by default, so user will not have to do it manually. It just needs to mark leave or absent.*

### Substitutivity

*In order to logout from system, the user can click on logout button or even the close button of page to logout the system. Similarly, to return the homepage, user can click on homepage button or the back button***.**

### Responsiveness

*The pages loading time will be same throughout the running of the system. Also, the menu driven approach is also used.*

### Recoverability

*The exceptions will be provided if the user does some error through which the user can view the error and recover it.*

# Requirement Interaction Matrices

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Requirments*** | ***3.1*** | ***3.2*** | ***3.3*** | ***4.1*** | ***4.2*** | ***4.3*** | ***4.4*** | ***45*** | ***4.6*** | ***4.7*** | ***4.8*** | ***4.9*** | ***4.1*** | ***4.11*** | ***4.12*** | ***4.13*** | ***4.14*** | ***4.15*** | ***4.16*** | ***5.1*** | ***5.2*** | ***5.3.1*** | ***5.3.2*** | ***5.3.3*** | ***5.3.4*** | ***5.3.5*** | ***5.3.6*** | ***5.3.7*** | ***5.3.8*** | ***5.3.9*** |
| ***3.1*** |  | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***3.2*** | *0* |  | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***3.3*** | *0* | *0* |  | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.1*** | *0* | *0* | *0* |  | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.2*** | *0* | *0* | *0* | *0* |  | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.3*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.4*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.5*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.6*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.7*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.8*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.9*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.1*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.11*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.12*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.13*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.14*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.15*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***4.16*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.1*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.2*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.1*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *100* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.2*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.3*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *100* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.4*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.5*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.6*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.7*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.8*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |
| ***5.3.9*** | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* | *0* |  |

**KEY Table:**

|  |  |
| --- | --- |
| ***Type of Requirments*** | ***Values*** |
| Independent Requirments | **0** |
| Conflicting Requirments | **1** |
| Overlapping Requirment | **100** |

# Stakeholder Engagement Matrix

**Power**

**Students**

**Security guards**

**Peon**

**Parents**

**Admin**

**Account**

**Teacher**

**Consultant**

**Principal**

**Reactor of school**

**Sindh Government**

**Board of Education**

**Private schools association**

**Interest**

# Role Matrix

Approach: RACI Model

Roles: Admin, Account, Teacher, Principal

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Process/Task** | **Admin** | **Account** | **Teacher** | **Principal** |
| Add details of newly admitted student and teacher | R, A |  |  | I |
| Set the Timetable for the teachers and students | R, A |  |  | I |
| Add Announcement to every users | R, A |  |  | I |
| Add Announcement to students by teacher | A |  | R | I |
| Add Announcement to students by account | A | R |  | I |
| Generate the fees of students | A | R |  | I |
| Submit the fees of the students | A | R |  | I |
| Mark the attendance of the students | A |  | R | I |
| Update the attendance of the student | A |  | R | I |
| Add marks of the student exams | A |  | R | I |
| Update the marks of the students exams | A |  | R | I |

# Use Cases

The usecase has been attached as a separate file with the SRS.