Project Topic

Mana Staff Management System

Abstract

D.S. Senanyake College is one of many institutes in Sri Lanka that keeps track of immense amounts of data. The college currently operates with a paper based system where information about staff and students is stored manually. This system has proven to cause a lot of problems such as inefficiency in performance, loss of data integrity due to human error etc. The proposed system is a web-based system that will be used by the staff to manage information. This includes and is not limited to student details, student marks and grades, teachers’ timetable details, attendance marking and leave management. An event management module will also be created for staff members to handle school events.

Developing the system must be done with caution as most of the intended users are not computer savvy and the UI has to be simple and intuitive enough for inexperienced users. Security is also a prime concern as the system will be deployed on the internet and therefore needs adequate protection from any known threats.

From this project, we hope to overcome the difficulties faced in the manual system and serve as a suitable replacement for the current system.

Declaration

We declare that the project work entitled “**Mana Staff Management System**” for “**D.S. Senanyake College**” was carried out by us. This project was undertaken as part of the academic curriculum (2nd year 2nd semester ITP module) and was done according to the Sri Lanka Institute of Information Technology rules and norms. This is our original work.

Project Details

|  |  |
| --- | --- |
| Project Title | Mana Staff Management System |
| Project ID | ITP-14-B1-06 |

Group Members

|  |  |  |
| --- | --- | --- |
| **Reg. No** | **Name** | **Signature** |
| IT13008338 | Peiries M.S.E |  |
| IT13014650 | Rathnayake R.M.D.A |  |
| IT13013424 | Liyanage M.C |  |
| IT13031312 | Joseph V.S |  |
| IT13028206 | Jayakody J.A.I |  |
| IT13024000 | De Silva G.L.N.A.M |  |
| IT13006426 | Mendis B.T.M |  |
| IT13001304 | Alimudeen M.Y |  |

Acknowledgement

We would like to acknowledge the support of the Old Boys Association of D.S. Senanyake College who provided us with the resources we need when it came to data on the school.

We are also grateful to our Project Supervisor Mr. Prasanna S. Haddela for his constant support in our project and for giving us his valuable insight and opinions. Mr. Haddela was instrumental in the development phase and this project would have been entirely different without him.

We would also like to thank Ms. Laksri Wijerathna and Ms. Geethanjali Wimalaratne for guiding us during lectures and also for providing the required material to do our project.

Table of Contents

Abstract i

Acknowledgement ii

Declaration iii

Table of Contents iv

List of Figures v

List of Tables v

List of Acronyms and Abbreviations v

1. Introduction 1

1.1 Problem Statement 1

1.2 Product Scope 1

1.3 Project Report Structure 1

2. Methodology 2

2.1 Requirements and Analysis 2

2.2 Design 2

2.3 Implementation 2

2.4 Testing 2

3. Evaluation 3

3.1 Assessment of the Project results 3

3.2 Lessons Learned 3

3.3 Future Work 3

4. Conclusion 4

5. References 5

Appendix A: Design Diagrams 6

Appendix B: Test Results 7

Appendix C: Selected Code Listings 8

List of Figures

*<Insert a list of figures; mention the number name and the page number of each figure>*

List of Tables

*<Insert a list of tables; mention the number name and the page number of each table>*

List of Acronyms and Abbreviations

*<Insert a list of acronyms and abbreviations; describe the specific meaning of those terms in your project>*

# Introduction

## Problem Statement

The system to be developed will be based on the current manual system they have in place. The reports generated will mimic reports and other data files already in use, so as to make the transition to the new system easier.

The Ministry of Education (under which the school operates) requires several reports from the school as it is the governing body. When producing these reports, they have to be made as similar to the existing ones as possible while still enhancing their current production model, e.g. Generating reports faster than they are being produced currently.

## Product Scope

Mana is a staff management system to be developed for D.S Senanayake College, Colombo. During the requirement gathering phase it was understood that at present the school is finding it hard to keep track of the data manually. All changes to the system have to be worked out manually. The intended system will focus on the staff information management, but will still deal with student information management, marks and grading, event planning, student attendance.

With a workforce of more than 300, it is highly inefficient and not to mention time consuming to store and manage data manually, not to mention the several thousand students.

Developing the system must be done with caution as most of the intended users are not computer savvy. Any user interface has to be simple and intuitive enough for inexperienced users. Security is also a prime concern as the system will be deployed on the internet and therefore needs adequate protection from any known threats.

**Key Benefits of the proposed system**

* All the staff, student, attendance and grading information will be available to anyone authorized to view this information.
* The automated system can provide access levels to different users to restrict access to material that they are not authorized to.
* The various detailed reports generated by this system will provide useful information to the teachers using this system.
* As it is web-based, the system can be accessed anywhere, at any given time without restrictions on platform either.
* Single centralized storage of information.
* Easy record keeping.

**Objectives**

* Provide a system the client can use to store and manage their staff members’ information.
* Fast and efficient retrieval of data of staff members.
* Ensure no data is lost to system errors by making multiple backups.
* Reduce the data entry time to a minimum so that users don’t have to waste time.
* Maintain a single copy of the information for many users to work on thereby eliminating data duplication and maintaining the integrity of data.
* Eliminating human error by having form validations therefore data accuracy is maintained.
* Provide descriptive statistical reports to the users for reviewing purposes.

**Goals**

* Provide an easy user experience with simple functions to introduce users to computers.
* Eliminate the limitations that are present in the manual system.
* Provide a system that rapidly processes data and information is generated quickly.

## Project Report Structure

< Introduce how the rest of the chapters of the project report is organized>

# Methodology

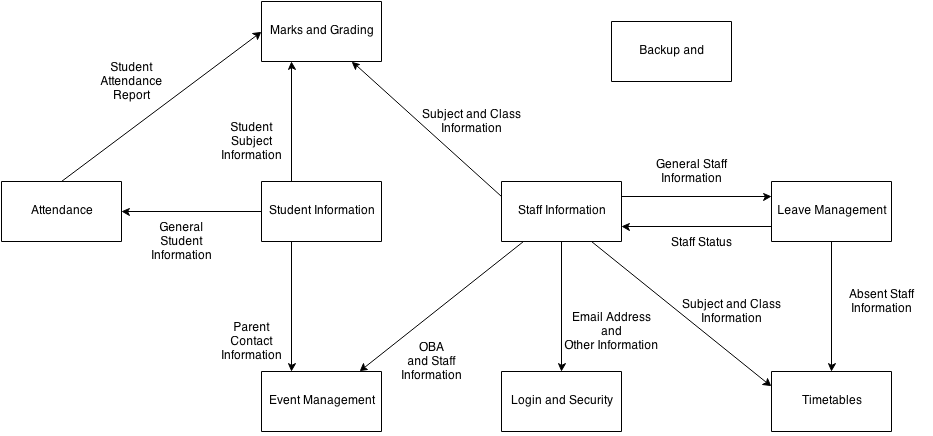
## Requirements and Analysis

We obtained several reports generated by the school and it helped us greatly to get a clear idea of how the current manual system operated. Furthermore meetings were held once every two weeks with our client. In the first few meetings we carried out question and answer sessions to obtain the requirements set. After we had an idea of the system, we developed dummy interfaces and obtained our clients feedback and understood how to improve our system.

### Product Functions

* Staff Management
* Leave Management
* Time Tables
* Student Information Management
* Marks and Grading
* Attendance
* Security
* Backup

The following is a diagram of how the mentioned function will interact with each other and constitute the system as a whole.

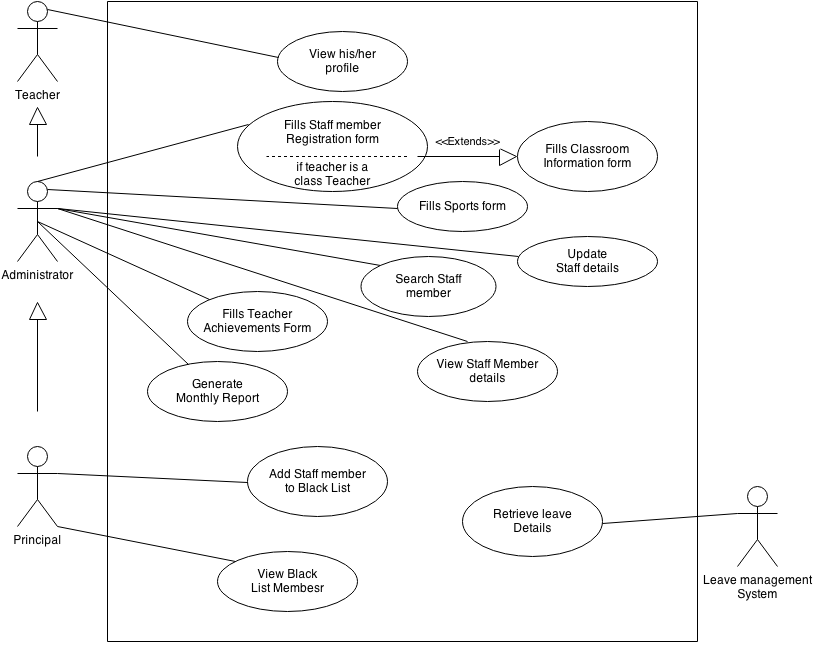


### Use Case Diagrams

##### **Staff Registration and Management System**

Figure

4.1.1



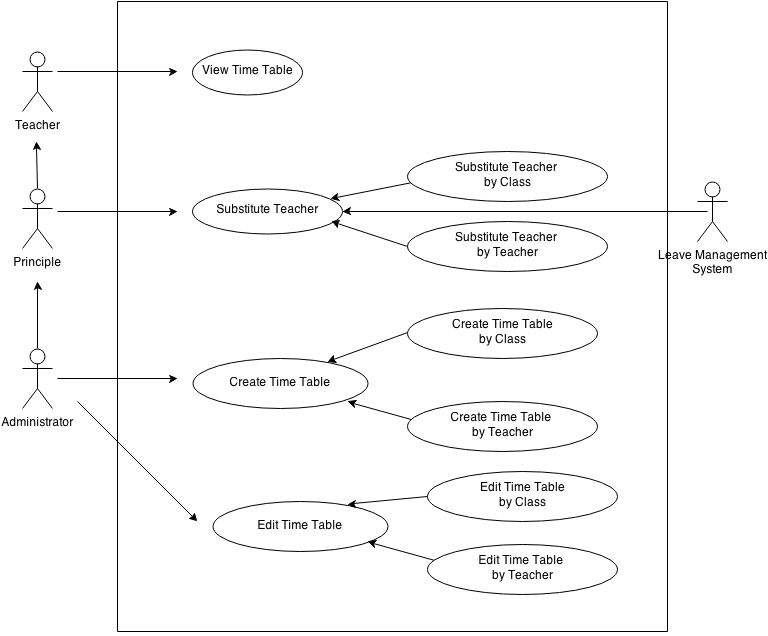
Staff Registration & Management

##### 

##### **Leave Management System**

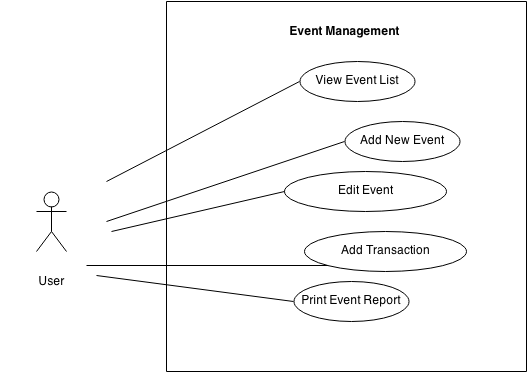


**Time Table**

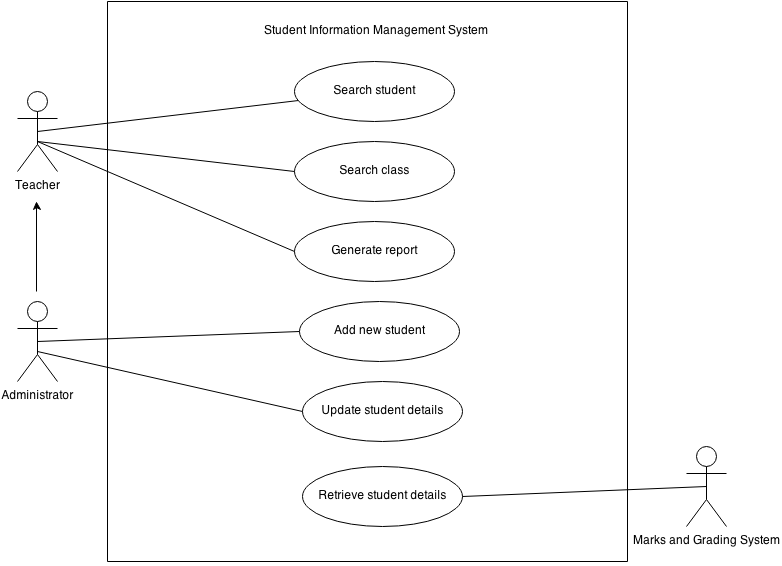


Time Table

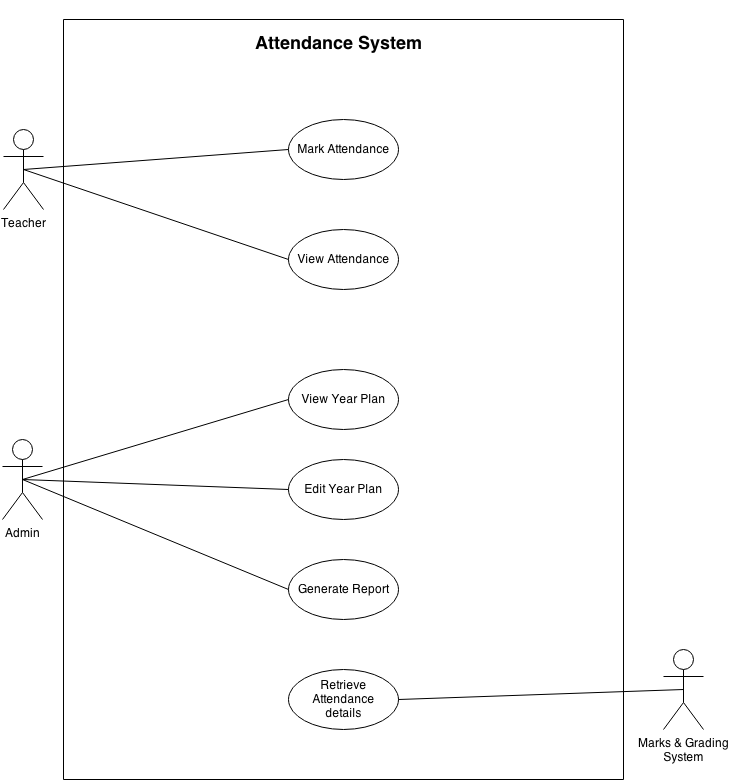
**Event Management System**



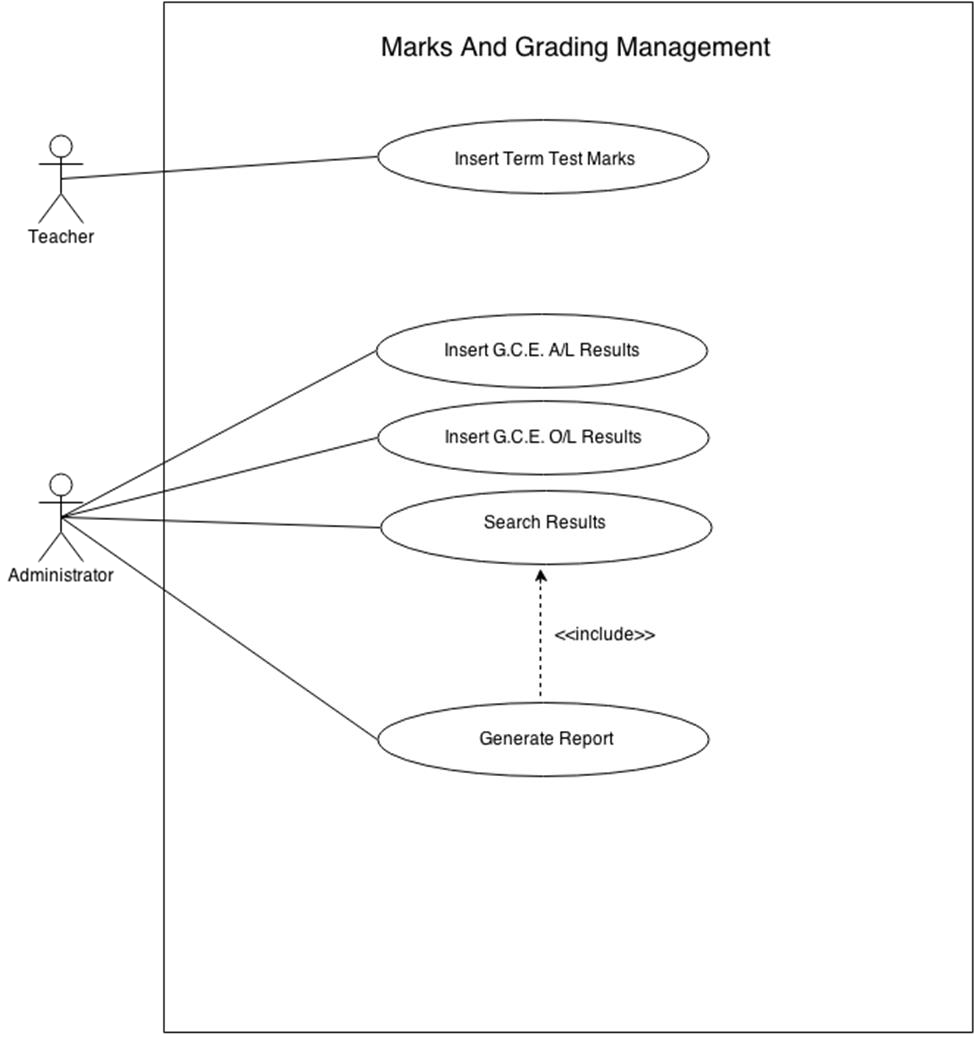
**Student Management System**



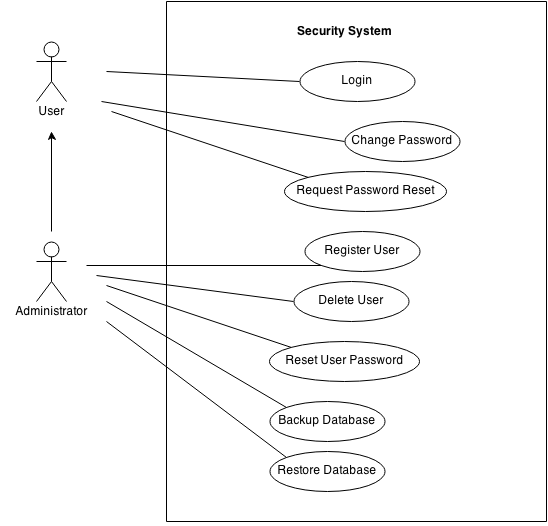
**Attendance System**



**Marks and Grading System**



**Security System**



### Use Case Scenarios

### Staff Registration and Management System

|  |  |
| --- | --- |
| Use Case  Name | Fill Staff Member Registration Form |
| Primary Actor | Administrator |
| Main Flow | 1). Login to the System  2). Administrator selects the Register Staff Member tab in the website.  3). Website displays the register interface with Staff registration form.  4). Administrator enters Staff member details and submit.  5). System will display a successful message and use case ends. |
| Extensions | 4) a) If Administrator provides wrong information system will displays an error  Message.  b) If staff member is a Class teacher needs to fill classroom Information form. |

|  |  |
| --- | --- |
| Use Case  Name | Fills sports Form |
| Primary Actor | Administrator |
| Pre-condition | Should have filled the staff member registration form and submit |
| Main Flow | 1) Login to the system.  2). Administrator selects the Sports form tab  3). Website displays the Sports form interface. 4). Administrator enters Sports details and submit |
| Extensions | 4) a) If Administrator didn’t fill required fields system will displays an error message |

|  |  |
| --- | --- |
| Use Case  Name | Fills Teacher Achievement Form |
| Primary Actor | Administrator |
| Pre-condition | Should have filled the staff member registration form and submit |
| Main Flow | 1) Login to the system.  2). Administrator selects the Teacher Achievements tab.  3). Website displays the Teacher Achievements interface.  4). Administrator enters details and clicks submit. |
| Extensions | 4) a) If Administrator didn’t fill required fields system will displays an error Message. |

|  |  |
| --- | --- |
| Use Case  Name | View Staff Details |
| Primary Actors | Administrator, principal |
| Main Flow | 1. Login to the system. 2. Click the “Staff management”. 3. Click the “View staff details” tab. 4. System prompts the Assign interface. 5) View the staff member Details |

|  |  |
| --- | --- |
| Use Case  Name | Search Staff Details. |
| Primary Actors | Administrator. Principal |
| Main Flow | 1. Login to the system. 2. Click the “Search Staff member”. 3. System prompts the Assign interface. 4. User can search Staff member by select the required field. 5. System views search results.   6)If he wants he can generate a report |
| Extensions | 5) a .if user didn’t select any required Field it will display all the details of staff members |

|  |  |
| --- | --- |
| Use Case  Name | Add Staff Member to Black List. |
| Primary Actor | principal |
| Main Flow | 1. Login to the system. 2. Click the “Add Staff member to Black List”. 3) System prompts the Assign interface. 3. Principal enter staff member details and clicks submit. 4. System display successful message. |
| Extensions | 5) a) If principal didn’t fill required field system will displays an error Message |

|  |  |
| --- | --- |
| Use Case  Name | View Black List member details. |
| Primary Actor | Principal |
| Main Flow | 1. Login to the system. 2. Click the “View Black List”. 3. System prompts the Assign interface   4)If principal needs he /she can generate a report |

|  |  |
| --- | --- |
| Use Case  Name | Generate Monthly Staff Report |
| Primary Actor | Administrator |
| Secondary Actor | Leave management System |
| Pre-condition | Should have filled the staff member registration form and submit |
| Main Flow | 1. Login to the system. 2. User selects the view staff Details form tab in the website. 3) Select view staff details with leave details 4) Clicks print button. |

|  |  |
| --- | --- |
| Use Case  Name | Update Staff member Details |
| Primary Actor | Administrator |
| Pre-condition | Should have filled the staff member registration form and submit |
| Main Flow | 1) Login to the system.  2). Administrator selects the Assigned tab  3). Website displays the related interface.  4). Administrator update and submit |
| Extensions | 4) a) If Administrator didn’t fill required fields system will displays an error message |

|  |  |
| --- | --- |
| Use Case  Name | View His/her profile. |
| Primary Actor | Teacher |
| Pre-condition | Teacher Should have registered to the system previously |
| Main Flow | 1. Login to the system 2. View His/ Her Profile with unread Notifications |

**Leave Management System**

|  |  |
| --- | --- |
| Use Case  Name | Apply for Leave |
| Primary Actor | Clerk |
| Pre-Conditions | Clerk has authorized login |
| Main Flow | 1. Clerk enters the Staff ID 2. Staff ID is validated and checked if exists 3. Clerk enters the start date 4. Clerk enters the end date 5. Clerk enters the number of leave days 6. Clerk picks the leave type 7. System generates the number of leave days left 8. Clerk clicks apply for leave after finished 9. System logs the leave request and sends for approval |
| Extensions | 2 a. If Invalid Staff ID is entered, display error message.  2 b. Go back to step 1.     1. a1. If Staff member doesn’t exist, display error message. 2 b1. Go back to step 1.      1. a. Invalid dates entered, display error message.   3 b1. Repeat step.    7a. Number of leave days over, display message. |

|  |  |
| --- | --- |
| Use Case  Name | Check Previous Leave History |
| Primary Actor | Staff Member |
| Pre-Conditions | Staff has a valid user login |
| Main Flow | 1. Staff member selects View Leave History from the side menu 2. Staff member sees his/her leave history |

|  |  |
| --- | --- |
| Use Case  Name | Cancel Leave |
| Primary Actor | Clerk |
| Pre-Conditions | Clerk has authorized login |
| Main Flow | 1. Clerk selects cancel leave from the side menu 2. Clerk enters the staff id 3. System checks approval status if approved/rejected 4. Clerk selects the leave request he wants to cancel and confirms it |
| Extensions | 2 a. If Staff ID is invalid, display error message.  2 b. Go back to step 1.     1. a1. If Staff member doesn’t exist, display error message. 2 b1. Go back to step 1.      1. a. If Leave is already approved or rejected, display message.      1. a. If No leaves has been requested, display message. |

|  |  |
| --- | --- |
| Use Case  Name | Approve Leave |
| Primary Actor | Principal |
| Pre-Conditions | Principal has an authorized login |
| Main Flow | 1. Principal selects the approve leave from main menu 2. Principal selects leave request from the list 3. Principal expands details 4. Principal selects approves/rejects leave request 5. System updates staff leave days left |
| Extensions | 2 a. No leave requests, display message saying no leave requests. |

|  |  |
| --- | --- |
| Use Case  Name | Generate Leave Report |
| Primary Actor | Principal |
| Pre-Conditions | Principal has an authorized login |
| Main Flow | 1. Principal selects leave report from the main menu 2. Principal see generate leave report form 3. Principal selects what sort of leave report it should be 4. Principal clicks generate report 5. System generates report 6. Principal can see the generated report |

|  |  |
| --- | --- |
| Use Case  Name | Retrieve Leave Details |
| Primary Actor | Time Table System |
| Main Flow | 1. Time table system requests list of Teachers on leave for teacher substitution 2. Time table system gives relevant parameters for data retrieval 3. System generates teachers on leave 4. System sends list of teachers to time table system |

|  |  |
| --- | --- |
| Use Case  Name | Retrieve Staff Status |
| Primary Actor | Staff Information System |
| Main Flow | 1. Staff information system requests current status of teacher/staff member 2. System generates staff status 3. Returned to the Staff information system |

**Time Table**

|  |  |
| --- | --- |
| Use Case Name | View Time Table |
| Primary Actor | Principle , Administrator |
| Main Flow | 1. Login 2. User will redirect to the Main Interface 3. User Click View Time Table Tab 4. User will Redirect to Time Table Interface 5. Then User Select View time table tab 6. User Enters Teacher’s Name or Class 7. Relevant Time Table will Display |

|  |  |
| --- | --- |
| Use Case Name | Create Time Table by Teacher |
| Primary Actors | Administrator |
| Main flow | 1. User Log in to the System 2. User will Redirect to the Main Interface 3. User Clicks Time Table Tab 4. User will Redirect to Time Table Interface 5. Then User Select Create time table tab 6. User Tick ‘ by teacher ’ Combo Box 7. Text box will appear to enter Teacher’s name 8. User enters the teacher’s name 9. User Feed the time table 10. User press save 11. System prompts “Time Table Successfully Save” |

|  |  |
| --- | --- |
| Use Case Name | Create Time Table by Class |
| Primary Actors | Administrator |
| Main flow | 1. User Log in to the System 2. User will Redirect to the Main Interface 3. User Clicks Time Table Tab 4. User will Redirect to Time Table Interface 5. Then User Select Create time table tab 6. User Tick ‘ by class ’ Combo Box 7. Text box will appear to enter class 8. User enters the class 9. User Feed the time table 10. User assign Teachers for subjects 11. User press save 12. System prompts “Time Table Successfully Save” |

|  |  |
| --- | --- |
| Use Case Name | Substitute Teacher by Teacher |
| Primary Actors | Administrator , Principle |
| Main flow | 1. User Log in to the System 2. User will Redirect to the Main Interface 3. User Clicks Substitute Teacher Tab 4. User will Redirect to Substitute Teacher Interface 5. User Ticks ‘ by teacher ’ Combo Box 6. Text box will prompt to enter Teacher’s name 7. User enters the teacher’s name 8. System Prompts relevant teacher’s time Table 9. User Selects relevant(concerned) period 10. Available Teachers list will appear with their contact details 11. User selects one of the available Teacher 12. System sends email and SMS for selected teacher |

|  |  |
| --- | --- |
| Use Case Name | Substitute Teacher by class |
| Primary Actors | Administrator , Principle |
| Main flow | 1. User Log in to the System 2. User will Redirect to the Main Interface 3. User Clicks Substitute Teacher Tab 4. User will Redirect to Substitute Teacher Interface 5. User Ticks ‘ by class ’ Combo Box 6. Text box will prompt to enter class 7. User enters the class 8. System Prompts relevant class Table 9. User Selects relevant(concerned) period 10. Available Teachers list will appear with their contact details 11. User selects one of the available Teacher 12. System sends email and SMS for selected teacher |

**Event Management System**

|  |  |
| --- | --- |
| Use Case Name | View Event List |
| Primary Actor | Principal |
| Pre-Condition | Should have valid login |
| Main Flow | 1.Login  2.Click Event Management in the Main Interface |

|  |  |
| --- | --- |
| Use Case Name | Add New Event |
| Primary Actor | Principal |
| Pre-Condition | Should have valid login |
| Main Flow | 1.Click Event Management in the Main menu  2.Click Add New Event button in the Event List interface  3.Fill the Event details  4.Click Save Event |

|  |  |
| --- | --- |
| Use Case Name | Edit Event |
| Primary Actor | Principal, Event Manager |
| Pre-Condition | Should have valid login |
| Main Flow | 1.Login  2.Click Event Management in the Main Interface  3.Click Manage button next to the related event  4.Edit required fields  5.Click Save Event |

|  |  |
| --- | --- |
| Use Case Name | Add Transaction |
| Primary Actor | Event Manager |
| Pre-Condition | Event is already created |
| Main Flow | 1.Login  2.Click manage button in the related event  3.Fill the transaction details |

|  |  |
| --- | --- |
| Use Case Name | Print Transaction Report |
| Primary Actor | Event Manager |
| Pre-Condition | Transactions were fed to the system |
| Main Flow | 1.Login  2.Click manage button in the related event  3.Click Print Transaction Button |

**Student Management System**

|  |  |
| --- | --- |
| Use Case  Name | Add new student |
| Primary Actor | Administrator |
| Pre-Conditions | Admin has authorized login |
| Main Flow | 1. Selects the add new student from menu 2. Enters the student details   3.Clicks submit button  4.Receives a successful message from the system |

|  |  |
| --- | --- |
| Use Case  Name | Update student Details |
| Primary Actor | Administrator |
| Pre-Conditions | Admin has authorized login |
| Main Flow | 1. Selects the update student details 2. Update the necessary student details 3. Clicks submit button 4. Receives a successful message from the system |

|  |  |
| --- | --- |
| Use Case  Name | Search Student |
| Primary Actor | Administrator and Teacher |
| Main Flow | 1. Selects the search student 2. Fill in the student admission number 3. Clicks the search button 4. Can view the searched results of the student |
| Extensions | 2.a : If the admission number is wrong, user will receive an error message from the system |

|  |  |
| --- | --- |
| Use Case  Name | Search Class |
| Primary Actor | Administrator and Teacher |
| Main Flow | 1.Selects the search class  2.Fill in the grade/year and the class  3.Clicks the search button  4.Can view the searched results of the student |
| Extensions | 2.a : If the grade and the class is wrong, user will receive an error message from the system |

|  |  |
| --- | --- |
| Use Case  Name | Generate Report |
| Primary Actor | Administrator |
| Main Flow | 1. User selects report from the main menu 2. User see generate report form 3. User selects what sort of report it should be 4. User clicks generate report 5. System generates report 6. User can see the generated report |

|  |  |
| --- | --- |
| Use Case  Name | Mark attendance. |
| Primary Actor | Teacher |
| Precondition | Teacher should have log in to the system. |
| Main Flow | 1. Log in to the system. 2. Select keep attendance. 3. Mark attendance. 4. Save. |

##### **Attendance System**

|  |  |
| --- | --- |
| Use Case  Name | View Attendance. |
| Primary Actor | Teacher |
| Precondition | Select class, grade and date. |
| Main Flow | 1. Teacher select view attendance from attendance. 2. Teacher enter student details and get attendance. |

|  |  |
| --- | --- |
| Use Case  Name | View year plan. |
| Primary Actor | Administrator |
| Precondition | Select year plan then view. |
| Main Flow | 1. Log in to the system. 2. Select the year plan. 3. View year plan. |
| Use Case  Name | Edit year plan. |
| Primary Actor | Administrator |
| Precondition | Go attendance interface and select year plan. |
| Main Flow | 1. Log in to the system. 2. Select year plan. 3. Edit year plan. 4. Save. |

**Marks and Grading System**

|  |  |
| --- | --- |
| Use Case  Name | Insert Term Test Marks |
| Primary Actor | Teacher |
| Main Flow | 1. Select Insert term test marks from Menu. 2. Teacher enters class details and term details and submits. 3. Teacher get student list of preferred class. 4. Teacher enters subject marks and remarks and submits. 5. Teacher gets success massage from the system. |
| Extensions | 2 a. If Teacher leave combo boxes empty an error massage will display. |

|  |  |
| --- | --- |
| Use Case  Name | Insert G.C.E. O/L results |
| Primary Actor | Administrator |
| Main Flow | 1. Select Insert G.C.E O/L results from Menu 2. Administrator enters index number and submit 3. Administrator gets Subject list along with the name, index number and year 4. Administrator enters G.C.E. O/L results 5. Administrator gets the success massage from the system |
| Extensions | 2 a .Administrator gets an error massage if he enters an invalid index number |

|  |  |
| --- | --- |
| Use Case  Name | Insert G.C.E. A/L results |
| Primary Actor | Administrator |
| Main Flow | 1. Select Insert G.C.E A/L results from Menu 2. Administrator enters index number and submit 3. Administrator gets Subject list along with the name, index number and year 4. Administrator enters G.C.E. A/L results 5. Administrator gets the success massage from the system |
| Extensions | 2 a .Administrator gets an error massage if he enters an invalid index number |

|  |  |
| --- | --- |
| Use Case  Name | Search Results |
| Primary Actor | Administrator |
| Main Flow | 1. Administrator select Search results from main menu. 2. Administrator selects what sort of result it should be. 3. Administrator clicks search. 4. Administrator can see the results 5. If he wants he can generate a report |
| Extensions | 2 a. If He enters wrong details, an error massage will display |

|  |  |
| --- | --- |
| Use Case  Name | Generate Report |
| Primary Actor | Administrator |
| Main Flow | 1. select report item from the menu 2. Administrator sees the generate report form 3. Administrator selects what sort of report it should be. 4. Click generate report 5. Administrator can see generated report 6. Prints the report |

##### **Security System**

|  |  |
| --- | --- |
| Use Case  Name | Login |
| Primary Actor | All Users |
| Preconditions | User has login credentials provided to them |
| Main Flow | 1. Enter username and password 2. Submit |
| Extensions | 2 a. If username/password combination is incorrect, log the attempt and allow a reattempt. |

|  |  |
| --- | --- |
| Use Case  Name | Change Password |
| Primary Actor | All Users |
| Preconditions | User is logged in |
| Main Flow | 1. Enter new password and confirm 2. Submit |
| Extensions | 2 a. If new passwords do not match, display an error message and prompt reattempt. |

|  |  |
| --- | --- |
| Use Case  Name | Request Password Reset |
| Primary Actor | All Users |
| Main Flow | 1. Enter email address 2. Confirm password reset |

|  |  |
| --- | --- |
| Use Case  Name | Register User |
| Primary Actor | Administrator |
| Main Flow | 1. Enter user information 2. Select user’s access rights 3. Provide user a password 4. Submit |

|  |  |
| --- | --- |
| Use Case  Name | Delete User |
| Primary Actor | Administrator |
| Main Flow | 1. Enter email 2. Confirm deletion 3. Send user email notifying account deletion |

|  |  |
| --- | --- |
| Use Case  Name | Reset User Password |
| Primary Actor | Administrator |
| Main Flow | 1. Select User from list of Users that have requested password resets 2. Confirm Selection 3. Provide user a password 4. Submit |

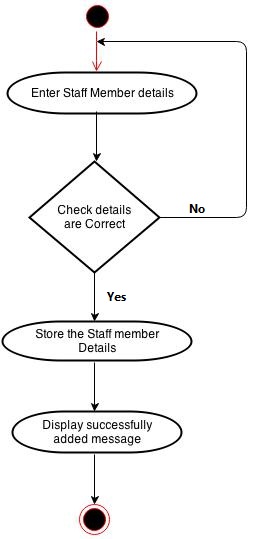
|  |  |
| --- | --- |
| Use Case  Name | Back up database |
| Primary Actor | Administrator |
| Main Flow | 1. Select Backup database |
|  | 2. Confirm back up |

|  |  |
| --- | --- |
| Use Case  Name | Recover Database |
| Primary Actor | Administrator |
| Main Flow | 1. Select back up from list 2. Confirm choice |

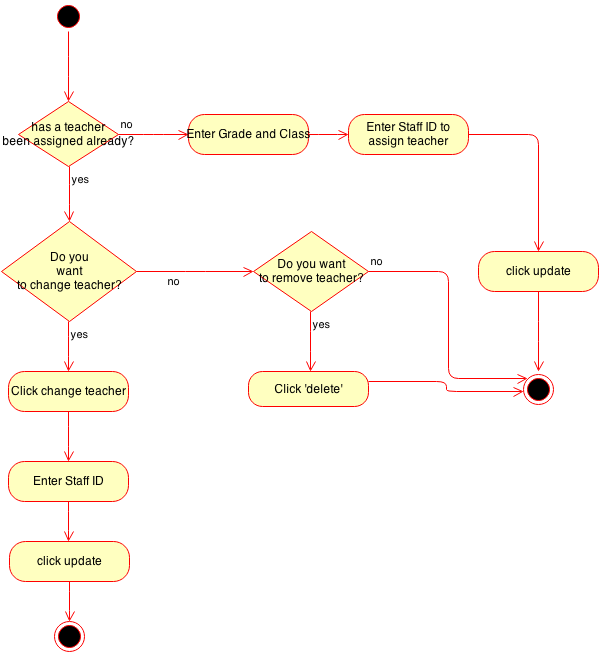
### Activity Diagrams

**Staff Management Module**

Register Staff Member

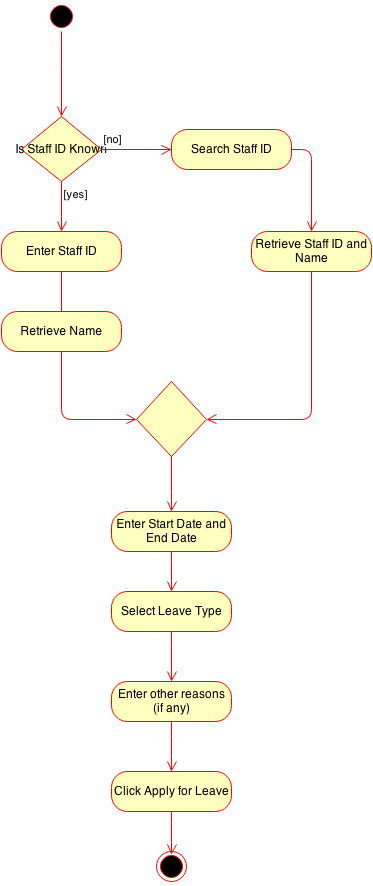


Assign Teacher to Class

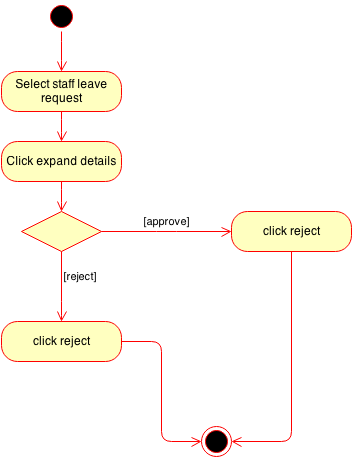


**Leave Management Module**

Apply for Leave

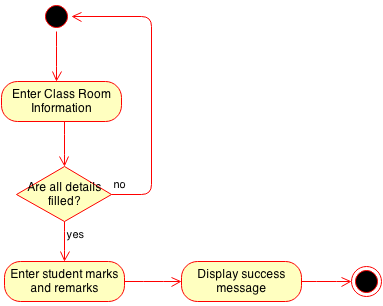


Approve Leave

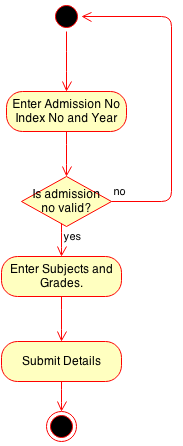


**Marks and Grading Module**

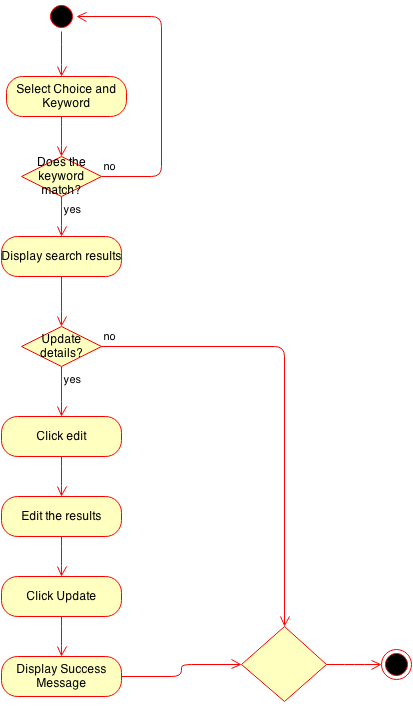
Enter Term Test Marks



Advanced Level and Ordinary Level Input

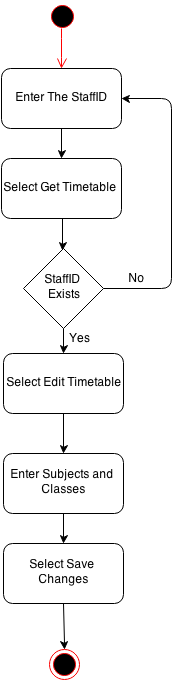


Search and Update Marks

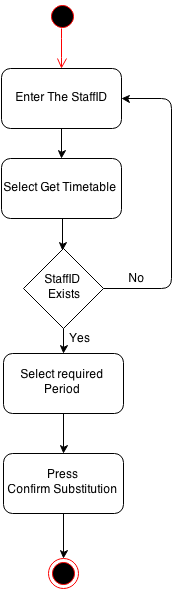


**Timetable Module**

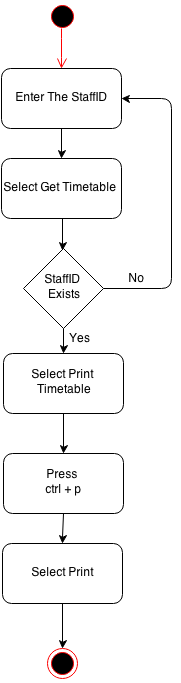
Create Timetable



Substitute Teacher

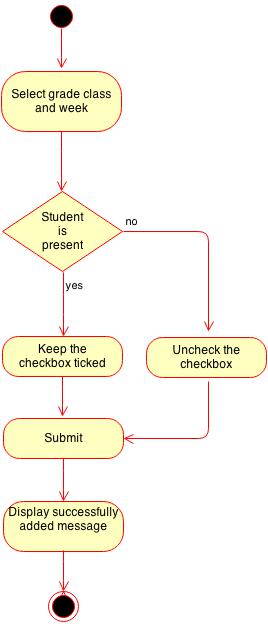


Print Timetable



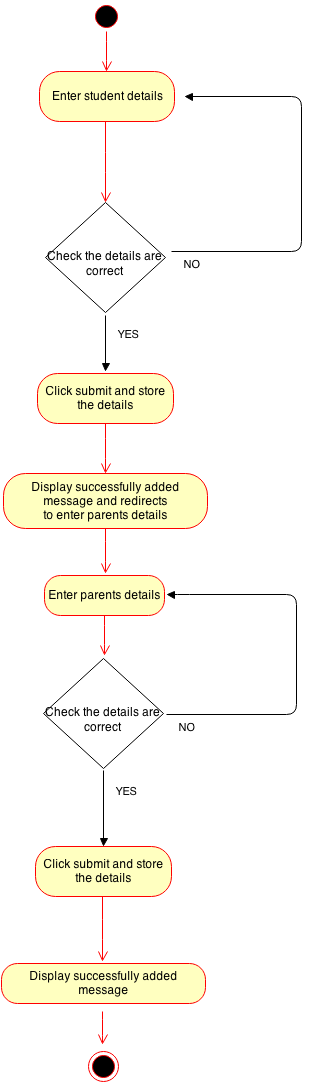
**Attendance Module**

Mark Attendance

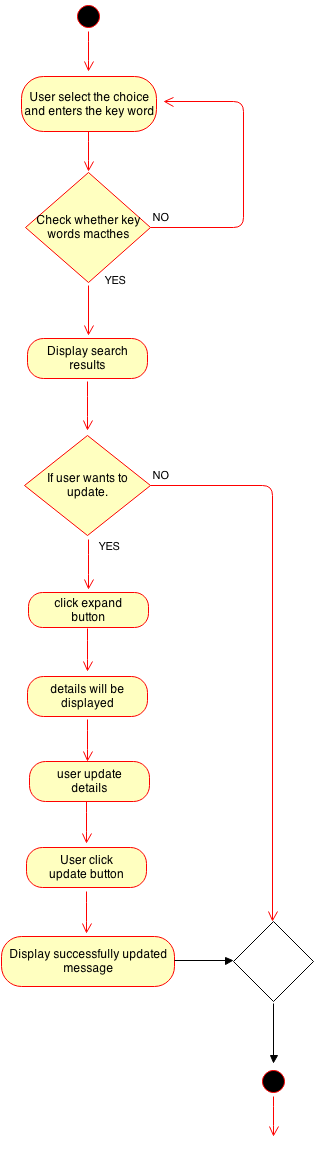


**Student Information Management Module**

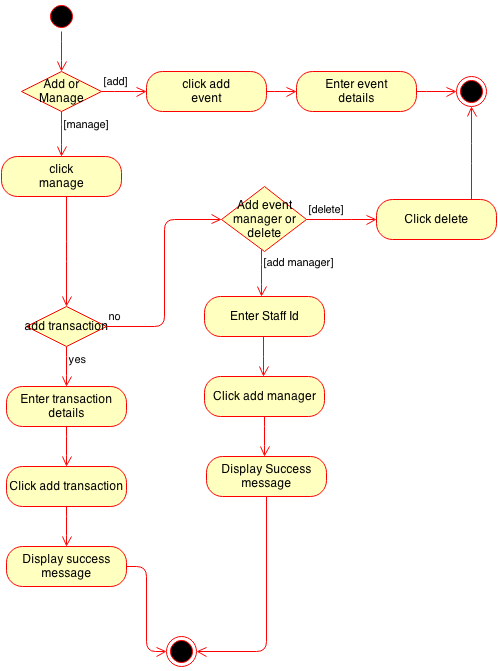
Register Student



Search Student



**Event Management Module**



## Design

< Design can be explained with the aid of diagrams. Explain the design of the software using UML diagrams such as Deployment diagram, component diagram, class diagram, Sequence diagram, State Charts, Activity diagrams etc. and explain the database design using the ER diagrams and E-R diagrams to Logical model Database schema>

## Implementation

< Here major module structures should be comprehensively explained. Any reusable code and development tools used must be explained. Also explain the choice of DBMS, Implementation Languages, Code of special algorithms used must be included to appendices. Do not just copy and paste all the code>

## Testing

<Describe your test plan and Evidence that all aspects of the system have been tested>

# Evaluation

## Assessment of the Project results

< What techniques were used to analyze the data and what the result of the analysis were. Any failures to achieve given objectives should be analyzed. Identify deficiencies in the final product and highlight any improvements could be made.>

## Lessons Learned

<Explain the Lessons learned.>

## Future Work

< Explain the suggestions for future work>

# Conclusion

< This section sums up the whole project. Discuss the realization of the original objectives/goals and how work can be taken further. Highlight the weaknesses/limitations of your proposed technique but you must always suggest a solution to all these (especially in future work). Also highlight the benefits of developing this project to the client organization>

# References

<Include a list of references done in the IEEE referencing style>

Appendix A: Design Diagrams

<Include the main UML diagrams in the main text and supplementary design diagrams can be included here>

Appendix B: Test Results

<Additional test result tables and figures can be included here>

Appendix C: Selected Code Listings

<Code of special algorithms implemented can be included here>