**2013**

**SUSMITA PODDER**

**1/19/2013**



SCHOOL MANAGEMENT SYSTEM

Table of Contents

[INTRODUCTION AND OBJECTIVE 4](#_Toc346626317)

[INTRODUCTION 4](#_Toc346626318)

[OBJECTIVE 4](#_Toc346626319)

[SYSTEM ANALYSIS 4](#_Toc346626320)

[IDENTIFICATION OF NEED 5](#_Toc346626321)

[PRELIMINARY INVESTIGATION 5](#_Toc346626322)

[FEASIBILITY STUDY 5](#_Toc346626323)

[PROJECT PLANNING 5](#_Toc346626324)

[TRACKING GANTT 5](#_Toc346626325)

[PROJECT SCHEDULING 5](#_Toc346626326)

[PERT CHART (NETWORK DIAGRAM) 5](#_Toc346626327)

[GANTT CHART 6](#_Toc346626328)

[SOFTWARE REQUIREMENT AND SPECIFICATION 7](#_Toc346626329)

[FUNCTIONAL REQUIREMENTS 7](#_Toc346626330)

[TECHNICAL SPECIFICATION 10](#_Toc346626331)

[SOFTWARE ENGINEERING PARADIGM APPLIED 10](#_Toc346626332)

[DATA MODELS 10](#_Toc346626333)

[CONTEXT DIAGRAM 10](#_Toc346626334)

[DATA FLOW DIAGRAM (DFD) 11](#_Toc346626335)

[CONTROL FLOW DIAGRAM 15](#_Toc346626336)

[STATE DIAGRAM / SEQUENCE DIAGRAM 15](#_Toc346626337)

[ENTITY RELATIONSHIP MODEL 15](#_Toc346626338)

[CLASS DIAGRAM / CRC MODEL /COLLABORATION DIAGRAM / USE-CASE DIAGRAM / ACTIVITY DIAGRAM 17](#_Toc346626339)

[SYSTEM DESIGN 18](#_Toc346626340)

[MODULARISATION DETAILS 18](#_Toc346626341)

[SCHOOL MANAGEMENT SERVER 19](#_Toc346626342)

[SCHOOL MANAGEMENT CLIENT 19](#_Toc346626343)

[SCHOOL MANAGEMENT DATABASE 20](#_Toc346626344)

[DATA INTEGRITY AND CONSTRAINTS 20](#_Toc346626345)

[DATABASE AND TABLE DESIGN 20](#_Toc346626346)

[PROCEDURAL DESIGN / OBJECT ORIENTED DESIGN 21](#_Toc346626347)

[USER INTERFACE DESIGN 22](#_Toc346626348)

[TEST CASES 22](#_Toc346626349)

[UNIT TEST CASES 22](#_Toc346626350)

[SYSTEM TEST CASES 22](#_Toc346626351)

[CODING 23](#_Toc346626352)

[COMPLETE PROJECT CODING 23](#_Toc346626353)

[COMMENTS AND DESCRIPTION OF CODING SEGMENTS 23](#_Toc346626354)

[STANDARDIZATION OF THE CODING 23](#_Toc346626355)

[CODE EFFICIENCY 23](#_Toc346626356)

[ERROR HANDLING 23](#_Toc346626357)

[PARAMETERS CALLING / PASSING 23](#_Toc346626358)

[VALIDATION CHECKS 23](#_Toc346626359)

[TESTING 23](#_Toc346626360)

[TESTING TECHNIQUES AND TESTING STRATEGIES USED 23](#_Toc346626361)

[TESTING PLAN USED 23](#_Toc346626362)

[TESTING REPORTS 23](#_Toc346626363)

[UNIT TEST CASES 23](#_Toc346626364)

[SYSTEM TEST CASES 24](#_Toc346626365)

[DEBUGGING AND CODE IMPROVEMENT 24](#_Toc346626366)

[SYSTEM SECURITY MEASURES 24](#_Toc346626367)

[DATABASE / DATA SECURITY 24](#_Toc346626368)

[CREATION OF USER PROFILES AND ACCESS RIGHTS 24](#_Toc346626369)

[COST ESTIMATION 24](#_Toc346626370)

[COST ESTIMATION MODEL 24](#_Toc346626371)

[REPORTS 24](#_Toc346626372)

[FUTURE SCOPE AND FURTHER ENHANCEMENT 25](#_Toc346626373)

[BIBLIOGRAPHY 25](#_Toc346626374)

[APPENDICES 25](#_Toc346626375)

[GLOSSARY 25](#_Toc346626376)

# INTRODUCTION AND OBJECTIVE

## INTRODUCTION

The population of our country is increasing rapidly, but the resources for providing proper education to the children are limited. So we need to utilize our existing school management processes properly with the help of digital technologies.

Most of the schools are managed by a single authority like head master or a group of people like governing body. But in this fast paced world people tend to switch jobs. If the key people leave the school then the quality of education and school management deteriorates. There are so many dependencies on the school management personnel. We can eradicate this dependency by deploying a computerized solution for managing school.

Nowadays both of the parents are employed for most kids and they don’t have enough time to interact with teachers frequently. We need a computerized system to manage interaction between parents and teachers.

We will implement a school management system to address these issues and bring up an efficient system to manage activities of a school using single software.

## OBJECTIVE

School Management System is versatile and complete end-to-end school management software .School Management System is used to enhance the administrative efficiency of educational institutions. It is an interactive platform for all entities viz. Students, Teachers, Management, Parents. It is a simple yet powerful one point integrated platform that connects all the departments of an institution namely office, fee counter, library, hostel, stores, academics, activity center and so on.

# SYSTEM ANALYSIS

## IDENTIFICATION OF NEED

When I was a kid I saw my parents coming to the school for submitting the remuneration and other reasons. The entire data was written using pen and papers so undoubtedly it took lots of time completing even minor queries. Nowadays I see my relatives going to the school to admit their kids and they face huge problems as no proper procedure is followed even by larger and more reputed institutes. I have always felt that there is a need of a proper procedure which could not only save lots of time of both the school authority and the guardians, also easy to maintain the entire process. So I really feel that the modernization of technology could be used to minimize these problems. That is why a proper application or software is required to ease the task of the school administration and save the time of the parents as well.

## PRELIMINARY INVESTIGATION

I started talking to my relatives and neighbors who have recently admitted their child to the schools or they are in the process of admitting them. Naturally, they are the ones who know the problems very well. I also spoke with some of school administrations in my locality about the problems they faced and what exactly would they want if they are given software like School Management System just to get an idea on what could the points be if we want to develop a new application to minimize school management related problems and difficulties. I obviously then consulted with some IT professionals and software engineers and expressed my view to them. I was really amazed to see their positive response on this matter and I got helps in many ways from them. After completing all this process I decided to develop this application for sure.

## FEASIBILITY STUDY

We all know that the numbers of school is growing as fast as Information Technology. Developing an application is very easy these days so people would love to use technology ease their tasks. There is need and there is solution so undoubtedly this software is going to be appreciated by the market.

## PROJECT PLANNING

### TRACKING GANTT



## PROJECT SCHEDULING

### PERT CHART (NETWORK DIAGRAM)



### GANTT CHART



## SOFTWARE REQUIREMENT AND SPECIFICATION

### FUNCTIONAL REQUIREMENTS

#### View and Enter new Student and Employee information

**Introduction**

The details of a new student are stored into a student profile. Only School administration department can enter that details of new student but and teachers can only updated the existing student status. Details of employee are also kept into individual employee profile and it will also be updated by School Administration System.

**Input**

Relevant student and employee data like name, address, contact no., applying for which class.

**Processing**

Employee will enter data in SMS and create a new Student enrolment no, as well as a new code number for Employee.

**Output**

SMS will generate Enrolment no. for Student and Code no. for Employee .Details can be viewed later on whenever required.

#### View and Enter new timetable information

**Introduction**

Employee can view the time table and can also update the timetable information.

**Input**

Employee number, department number and Week range

**Processing**

Employee number and department number must be unique, and when entering timetable of any employee or teacher both values must be valid references. Week range must be between 1to 52.

**Output**

Teacher and Student both can see the time table.

#### Security

**Introduction**

Only the high level members of the School and Network manager will have access to the system for securing their important data from others.

**Input**

System username and password

**Processing**

The network operating system in the department will be used to enforce security. Another security level should also be incorporated to make the system more secure.

**Output**

All data are secured and that can be used in future.

#### Changing Password and Username

**Introduction**

Change existing username and password

**Input**

New username and password

**Processing**

Old username and password will be replaced by user provided new username and password after authenticating.

**Output**

Password and Username can be changed according to the Employee requirement whenever they want to change for better security of the System.

#### Mail Notification

**Introduction**

If holiday is declared suddenly, all students, teachers and employees are informed by sending them a mail.

**Input**

Student and Teacher’s name and email id.

**Processing**

Employee will enter the name, email id and reason of holidays in the SMS and it will generate a message.

**Output**

Employee and Students get a message from SMS.

#### Exam Grade Details

**Introduction**

Data sheets are prepared for individual class. And each datasheet is given via email to the student of the corresponding class.

**Input**

Student name, marks in individual subject, attendance, class performance**.**

**Processing**

Grade card can be generated for individual students. Exam administrators would need to be able to view, update, delete, print and add grade details.

**Output**

A printed Grade card can be given to the student as well as an email can be received by the student with Grade card.

#### Fees Details

**Introduction**

Fees details of all Students are kept and they are reminded after every 4 days after last date. Late Fee is also charged after last date.

**Input**

Student name, enrolment no, remaining fees, last date of deposit amount.

**Processing**

SMS automatically generates a message and send it to the student email id.

**Output**

Students are reminded after every 4 days after last date.

### non FUNCTIONAL REQUIREMENTS

* The software must have a easy to use graphic user interface as it is going to be used by all types of users, like professors and clerks.
* The software should be password protected to secure confidential data.
* The database must be encrypted so that no one could see the accounts information of the organization even if the database is somehow hacked.
* The application must be fast and flexible so that waiting time gets reduced.
* There must be a backup and restore feature so that the valuable data stays secure forever.

## SOFTWARE ENGINEERING PARADIGM APPLIED

Project Category

## DATA MODELS

### CONTEXT DIAGRAM



### DATA FLOW DIAGRAM (DFD)

#### LEVEL 0 DFD



#### LEVEL 1 DFD







#### LEVEL 2 DFD



### CONTROL FLOW DIAGRAM







### SEQUENCE DIAGRAM



### ENTITY RELATIONSHIP MODEL

We will design a RDBMS for School Management System. The entities and their attributes are listed below. Attributes in Bold letter is the unique key.

|  |  |
| --- | --- |
| **Entities** | **Attributes** |
| Student | **Student\_ID**, Student\_DOB, Student\_Name, Student\_Parent\_Name, Student\_Address, Student\_Admission\_Date, Student\_Course\_Name, Student\_Contact |
| Account | **Transaction\_ID**, Transaction\_Amount, Transaction\_Type, Transaction\_Reason,Account\_Balance |
| Admin | **Staff\_ID**, Staff\_Name, Staff\_Permission\_Level, Staff\_Address, Staff\_Admission\_Date, Staff\_Course\_Name, Staff\_Contract\_details, Staff\_Join\_Date, Staff\_Email, Staff\_Role |
| Books | **Book\_ID**, Book\_Name, Book\_Author, Purchase\_Date, Book\_Status, Book\_Description,Purchase\_Amount |
| Faculty | **Faculty\_ID**, Faculty\_Name, Faculty\_Address, Faculty\_Join\_Date, Faculty\_Course\_Under, Faculty\_Contact\_Details, Faculty\_Salary\_Details |
| Course | **Course\_ID**, Course\_Faculty, Course\_Name, Required\_Qualification, Course\_Fees, Course\_Admission\_Date, Students\_Under, Course\_Description |

**Relationship between Entities:**

School Management System has Courses 1 : N

School Management SystemhasStudents 1 : N

School Management System has Faculties1 : N

School Management System has Admin 1 : 1

Studenthas Attendance1 :1

AdminChecksAttendance1 : 1

StudentsreadsBooksM : N

StudentspaysAccount1 : 1

AdminControlsAccount1 : 1

Students Studies in Course N:1



### CLASS DIAGRAM / CRC MODEL /COLLABORATION DIAGRAM / USE-CASE DIAGRAM / ACTIVITY DIAGRAM



# SYSTEM DESIGN

## MODULARISATION DETAILS



School Management System is divided three main modules such as:

1. School Management Server
2. School Management Client
3. School Management Database

### SCHOOL MANAGEMENT SERVER

School Management server is a singleton server designed provide services for school management system. It controls various activities required for the school management system. To manage these activities it has several sub modules such as:

1. Admission Management
2. Student Management
3. Faculty Management
4. Course Management
5. Attendance & Leave Management
6. Library Management
7. Accounts Management
8. Administration Management

### SCHOOL MANAGEMENT CLIENT

School Management System will provide two different clients for the convenience of the user. Desktop client is for doing bulk activities and faster tasks. Web client will allow instant access from anywhere and anytime.

### SCHOOL MANAGEMENT DATABASE

School Management System will have a unified database for storing all the information. It can be a networked database or a database situated in the server machine.

## DATA INTEGRITY AND CONSTRAINTS

?????????? database related issues and limitations

## DATABASE AND TABLE DESIGN

The database used for this software is called **smsdb**. A screenshot from the MySQl workbench is given below. It shows the tables and its columns. The first row is the primary key.



## PROCEDURAL DESIGN / OBJECT ORIENTED DESIGN

## USER INTERFACE DESIGN

## TEST CASES

### UNIT TEST CASES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TEST CASE ID** | **ITEM** | **DESCRIPTION** | **ACTUAL RESULT** | **TESTED BY** |
| SMS – 001 | Login | Enter User ID and Password for Login. | Successfully Logged in. | Susmita |
| SMS – 002 | Cancel | Select Cancel to close Login window. | Successfully Canceled. | Susmita |

### SYSTEM TEST CASES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TEST CASE ID** | **ITEM** | **DESCRIPTION** | **ACTUAL RESULT** | **TESTED BY** |
| SMS – 001 | Login | Enter User ID and Password for Login. | Successfully Logged in. | Susmita |
| SMS – 002 | Cancel | Select Cancel to close Login window. | Successfully Canceled. | Susmita |
| SMS – 003 | Admission | To add a new student enter the Student\_ID, Student\_DOB, Student\_Name, Student\_Parent \_Name, Student\_Address, Student\_Admission\_ Date,Student\_course \_Name, Student\_Contact | New Student is added to the School Management System. | Susmita |
| SMS – 004 | ViewStudentStatus | Enter Student\_ID, Student\_Name, Student\_Course\_Name of the Student. | Show the Student Details. | Susmita |
| SMS – 005 | EditStudentStatus | Select the Student and Click the Edit option. Now edit the Student Details and submit the Details. | Student Details successfully updated. | Susmita |

# CODING

## COMPLETE PROJECT CODING

## COMMENTS AND DESCRIPTION OF CODING SEGMENTS

## STANDARDIZATION OF THE CODING

## CODE EFFICIENCY

## ERROR HANDLING

## PARAMETERS CALLING / PASSING

## VALIDATION CHECKS

# TESTING

## TESTING TECHNIQUES AND TESTING STRATEGIES USED

SMS application will be tested using following strategies to ensure that the application succeeds to complete all the functional and non functional requirements:

### Functional Testing:

Function testing focuses on any requirements for test that can be traced directly to use cases or business functions and business rules. The goals of these tests are to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules. This type of testing is based upon black box techniques; that are verifying the application and its internal processes by interacting with the application via the Graphical User Interface (GUI) and analyzing the output or results. Identified below is an outline of the function testing recommended for SMS:

|  |  |
| --- | --- |
| Test Objective: | Ensure proper target-of-test functionality, including business process validation. |
| Technique: | Execute each use case, use-case flow, or function, using valid and invalid data, to verify the following:   * The expected results occur when valid data is used. * The appropriate error or warning messages are displayed when invalid data is used. * Business rules are properly applied. * Black Box end to end testing of configured processes. Manual validation of required and optional fields. |
| Completion Criteria: | * All planned tests have been executed. * All defects that have been identified have been resolved * All resolutions have been implemented. |

### Regression Testing:

Regression testing focuses on software functionality that may have been previously working however through subsequent changes may have been inadvertently impacted. The goals of these tests are to verify that the broader impact of changes has been verified. Identified below is an outline of the regression testing recommended for each application(s)/module(s) of SMS.

|  |  |
| --- | --- |
| Test Objective: | Ensure that previously passed test cases continue to pass as the new system development is deployed and that surrounding systems that may be impacted by a change are still functioning as expected. |
| Technique: | * Execute previous passed testing suites to ensure the following: * The expected results occur when valid data is used. * The appropriate error or warning messages are displayed when invalid data is used. * Each business rule is properly applied. |
| Completion Criteria: | • All planned regression tests have been executed.  • All identified defects have been resolved. |

### Database & Data Integrity Testing

The databases and the database processes should be tested as a subsystem within the SMS Application. These subsystems should be tested with the target-of-test’s User Interface as the interface to the database.

|  |  |
| --- | --- |
| Test Objective: | Ensure that data is stored correctly, audits can be performed, access is controlled |
| Technique: | * SQL queries will be executed in the DB to verify the data content and correctness. |
| Completion Criteria: | * All planned tests have been executed. * All defects that have been identified have been resolved * All resolutions have been implemented. |

### User Interface Testing:

User Interface (UI) testing verifies a user’s interaction with the software. The goal of UI testing is to ensure that the User Interface provides the user with the appropriate access and navigation through the functions of the target-of-test. In addition, UI testing ensures that the objects within the UI function as expected and conform to corporate or industry standards. Most of this testing will have been done during functional testing. The areas of focus will be on design, layout and navigation of the screens.

|  |  |
| --- | --- |
| Test Objective: | UI testing will verify the screens and the layouts and navigation |
| Technique: | * Verify the design and layout of the screen. * Identify the integration links. * Test the functioning of the links – that the proper page is displayed and correct messages, pop-ups are shown when they need to be displayed etc * Validation of general navigation |
| Completion Criteria: | * All navigation test cases have been executed. * All screens have been verified as per design and layouts * All defects that have been identified have been resolved. |

### Performance Profiling:

Performance profiling is a performance test in which response times, transaction rates, and other time-sensitive requirements are measured and evaluated. The goal of Performance Profiling is to verify performance requirements have been achieved. Performance profiling is implemented and executed to profile and tune performance behaviours as a function of conditions such as workload or hardware configurations

|  |  |
| --- | --- |
| Test Objective: | The purpose of performance profiling is to ensure the performance of the SMS application is up to the desired level. |
| Technique: | * Use a subset of Test Procedures developed for Function and Business Cycle Testing. * Modify data files to increase the number of transactions or the scripts to increase the number of iterations each transaction occurs. * This will be done by using Load Runner or Quick Test Professional (QTP). |
| Completion Criteria: | * Single Transaction or single user: Successful completion of the test scripts without any failures and within the expected or required time allocation per transaction. * Results are recorded and a performance baseline is created for the major logical functions within the scenarios listed above. * All performance defects are reviewed and triaged to an acceptable resolution. |

.

### Load Testing:

Load testing is a performance test which subjects the target-of-test to varying workloads to measure and evaluate the performance behaviours and ability of the target-of-test to continue to function properly under these different workloads. The goal of load testing is to determine and ensure that the system functions properly at the expected maximum workload. Additionally, load testing evaluates the performance characteristics, such as response times, transaction rates, and other time sensitive issues.

|  |  |
| --- | --- |
| Test Objective: | The purpose of load testing is to verify performance behaviour time for designated transactions or business cases under varying workload conditions. |
| Technique: | * Use a subset of Test Procedures developed for Function and Business Cycle Testing. * Scripts will be executed to simulate the peak load for 1 hour and report will be generated and analysed. * This will be done using Load Runner. |
| Completion Criteria: | * Multiple transactions or multiple users: Successful completion of the test scripts without any failures and within acceptable time allocation. * Results are recorded and a performance baseline is created for the major business functions within the scenarios listed above. * All load testing defects are reviewed and triaged to an acceptable resolution. |

### Stress Testing:

Stress testing is a type of performance test implemented and executed to find errors due to low resources or competition for resources. Low memory or disk space may reveal defects in the target-of-test that aren't apparent under normal conditions. Other defects might result from competition for shared resources like database locks or network bandwidth. Stress testing can also be used to identify the peak workload the target-of-test can handle, which is often beyond the production workload.

### Volume Testing:

Volume Testing subjects the target-of-test to large amounts of data to determine if limits are reached that cause the software to fail. Volume Testing also identifies the continuous maximum load or volume the target-of-test can handle for a given period. For example, if the target-of-test is processing a set of database records to generate a report, a Volume Test would use a large test database and check that the software behaved normally and produced the correct report.

### Security & Access Control Testing:

Security and Access Control Testing focus on following key areas of security:

* Application-level security, including access to the Data or Business Functions

Application-level security ensures the authentication and authorization of a user. Authentication ensures that the user is a valid user of the system and authorization ensures that the user has the proper privileges to perform specific tasks on desired resources of the system. Testing will be conducted to validate the rules by taking into considerations the various roles applicable for the system.

### Failover & Recovery Testing:

Failover and Recovery Testing ensures that the target-of-test can successfully failover and recover from a variety of hardware, software or network malfunctions with undue loss of data or data integrity.

Failover testing ensures that, for those systems that must be kept running, when a failover condition occurs, the alternate or backup systems properly “take over” for the failed system without loss of data or transactions.

Recovery testing is an antagonistic test process in which the application or system is exposed to extreme conditions, or simulated conditions, to cause a failure, such as device Input/ Output (I/O) failures or invalid database pointers and keys. Recovery processes are invoked and the application or system is monitored and inspected to verify proper application, or system, and data recovery has been achieved.

### Configuration Testing:

Configuration testing verifies the operation of the target-of-test on different software and hardware configurations. In most production environments, the particular hardware specifications for the client workstations, network connections and database servers vary. Client workstations may have different software loaded⎯for example, applications, drivers, and so on⎯and at any one time, many different combinations may be active using different resources.

### Installation/Deploy & Back out Testing:

Installation testing has two purposes. The first is to ensure that the software can be installed under different conditions⎯such as a new installation, an upgrade and a complete or custom installation⎯under normal and abnormal conditions. Abnormal conditions include insufficient disk space, lack of privilege to create directories, and so on. The second purpose is to verify that, once installed; the software operates correctly and can be backed out successfully. This usually means running a number of the tests that were developed for Function testing before and after the back out.

### Post Production Testing:

The purpose of Post production testing is to verify that, once deployed, the software operates correctly. This usually means running a number of the tests that were developed for Function Testing ensuring that no data is changed/ modified in production. Typically, the business SME’s assist with Post production testing.

### Unit Testing:

Unit testing will take place within the construction phase of the project. After application module has been built to meet design specifications, each component (screen, view, interface, conversion program, etc.) will be tested individually to help confirm that it functions properly as an individual unit. Basic performance testing will also be completed during unit test to resolve obvious issues with performance prior to the System Testing.

The resource responsible for development will conduct testing of their module using the unit test conditions defined by the developer based on detailed design documents. The final step of unit test will be a review by the team lead to obtain their signoff on the component test checklist.

### Smoke Testing:

|  |  |
| --- | --- |
| Test Objective: | Verifies the major functionality at high level in order to determine if further testing is possible. |
| Technique: | * After initial deployment to the test environment validate all critical components of the application prior to proceeding with testing. |
| Completion Criteria: | * Navigation through the application at high level is possible, testing can continue. |

### Data Migration Testing:

This is the process of testing to verify whether or not the data migration (or conversion) has been successfully completed. The testing process will be carried out by running SQL scripts on both the source and destination databases.

The fields which are present in the new data Model in the Destination DB(s) will be migrated from the existing systems source DB(s) to Destination DB(s).

|  |  |
| --- | --- |
| Test Objective: | The objective of this test is to verify that data migration is successful from source DB(s) to destination DB(s). |
| Technique: | * The Team is notified before the data migration. * Team runs queries on the source DB and fetches the data. * Data Migration is done. * Mapped data needs to be determined. * Team runs the queries on the Destination DB and fetches the data. * Cross verification of the data is done to see that data fetched from the old database is same as the data fetched from the new database. * Verification of the table structure. * Verification of record counts. * Verification of the data formatting. |
| Completion Criteria: | * Data fetched from the Source DB(s) and Destination DB(s) matches. * The record count in the Source and the Destination databases should be equal. * No data are truncated. * Data formatting is proper (if required at any instance). * All defects that have been identified have been resolved. |

## TESTING PLAN USED

## TESTING REPORTS

### UNIT TEST CASES

|  |  |  |
| --- | --- | --- |
| Test Case Id | Comments | Status |
| SMS – 001  SMS – 001 | NA  NA | PASS  PASS |

### SYSTEM TEST CASES

|  |  |  |
| --- | --- | --- |
| Test Case Id | Comments | Status |
| SMS – 001 | NA | PASS |
| SMS – 002  SMS – 003  SMS – 004  SMS – 005  SMS – 006  SMS – 007  SMS – 008  SMS – 009  SMS – 010  SMS – 011  SMS – 012  SMS – 013  SMS – 014  SMS – 015  SMS – 016  SMS – 017  SMS – 018  SMS – 019  SMS – 020  SMS – 021  SMS – 022  SMS – 023  SMS – 024  SMS – 025  SMS – 026  SMS – 027  SMS – 028  SMS – 029  SMS – 030 | NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA  NA | PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS  PASS |

## DEBUGGING AND CODE IMPROVEMENT

# SYSTEM SECURITY MEASURES

* School Management System is password protected software. It will be developed such a way that the admin will have complete control on the school’s data.
* Admin can create account with various permission levels, like clerk, librarian, teacher, admin etc. so that the users can see relevant data only.

## DATABASE / DATA SECURITY

* The data of the school will be stored in the database with an encrypted format so even if someone hacks the database somehow still he can make no real harm.
* The software will provide a backup and restore feature in case of loss of data.

## CREATION OF USER PROFILES AND ACCESS RIGHTS

* The software asks for a predefined user-type, username and password to use its feature.
* All the data are not available for all types of user, for example, only an admin can use all the fields of the application. On the other hand, a clerk can only enter data and see data from some selected fields. A librarian can use library related data only.

# COST ESTIMATION

## COST ESTIMATION MODEL

??????????????????????????



# REPORTS

List of reports that are likely to be generated in this software are given below:

* Results & Grade report can be generated
* Class toppers will be given a certificate as a token of appreciation
* List of students can be generated
* List of teachers can be generated
* List of courses can be generated
* Course details can be generated
* Fund details can be generated
* Yearly donation report can be generated
* Salary slips can be created

# FUTURE SCOPE AND FURTHER ENHANCEMENT

* Mobile application could be developed for students’ guardians for querying about various details.
* Support for Linux operating system could be added.
* Online result checking and fees payment feature could be added.

# BIBLIOGRAPHY

* <http://en.wikipedia.org>
* <http://msdn.microsoft.com/en-us/>
* <http://www.microsoft.com/en-us/default.aspx>
* <http://www.codeplex.com/>
* <http://stackoverflow.com/>
* <http://www.codeguru.com/>
* [http://www.w3schools.com](http://www.w3schools.com/)
* [www.mysql.org](http://www.mysql.org)
* School Professionals
* **Programming Java** - E. R. Balaguruswamy

# APPENDICES

## IDE (Visual Studio 2010):

Microsoft Visual Studio is a powerful IDE that ensures quality code throughout the entire application lifecycle, from design to deployment. Whether we are developing applications for SharePoint, the web, Windows, Windows Phone, and beyond, Visual Studio is the ultimate all-in-one solution. Visual Studio includes a [code editor](http://en.wikipedia.org/wiki/Code_editor) supporting [IntelliSense](http://en.wikipedia.org/wiki/IntelliSense) as well as [code refactoring](http://en.wikipedia.org/wiki/Code_refactoring). The integrated [debugger](http://en.wikipedia.org/wiki/Microsoft_Visual_Studio_Debugger) works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building [GUI](http://en.wikipedia.org/wiki/GUI) applications, web designer, [class](http://en.wikipedia.org/wiki/Class_(computing)) designer, and [database schema](http://en.wikipedia.org/wiki/Database_schema) designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for [source-control](http://en.wikipedia.org/wiki/Source_control) systems (like [Subversion](http://en.wikipedia.org/wiki/Subversion_(software)) and [Visual SourceSafe](http://en.wikipedia.org/wiki/Visual_SourceSafe)) and adding new toolsets like editors and visual designers for [domain-specific languages](http://en.wikipedia.org/wiki/Domain-specific_language) or toolsets for other aspects of the [software development lifecycle](http://en.wikipedia.org/wiki/Software_development_lifecycle) (like the [Team Foundation Server](http://en.wikipedia.org/wiki/Team_Foundation_Server) client: Team Explorer).

## Front End - WPF (Windows Presentation Framework)

Windows Presentation Foundation (WPF) provides developers with a unified programming model for building rich Windows smart client user experiences that incorporate UI, media, and documents. Windows Presentation Foundation (WPF) is a next-generation presentation system for building Windows client applications with visually stunning user experiences. With WPF, you can create a wide range of both standalone and browser-hosted applications. The core of WPF is a resolution-independent and vector-based rendering engine that is built to take advantage of modern graphics hardware. WPF extends the core with a comprehensive set of application-development features that include Extensible Application Markup Language (XAML), controls, data binding, layout, 2-D and 3-D graphics, animation, styles, templates, documents, media, text, and typography. WPF is included in the Microsoft .NET Framework, so you can build applications that incorporate other elements of the .NET Framework class library.

## Extensible application Markup Language (XaML)

XAML stands for Extensible Application Markup Language. Its a simple language based on XML to create and initialize .NET objects with hierarchical relations. Altough it was originally invented for WPF it can by used to create any kind of object trees.

Today XAML is used to create user interfaces in WPF, Silverlight, declare workflows in WF and for electronic paper in the XPS standard.

All classes in WPF have parameter less constructors and make excessive usage of properties. That is done to make it perfectly fit for XML languages like XAML.

All you can do in XAML can also be done in code. XAML ist just another way to create and initialize objects. You can use WPF without using XAML. It's up to you if you want to declare it in XAML or write it in code. Declare your UI in XAML has some advantages:

* XAML code is short and clear to read
* Separation of designer code and logic
* Graphical design tools like Expression Blend require XAML as source.
* The separation of XAML and UI logic allows it to clearly separate the roles of designer and developer.

## Programming Framework (.NET 4)

The .NET 4 Framework is Microsoft's platform for building applications that have visually stunning user experiences, seamless and secure communication, and the ability to model a range of business processes. The .Net Framework consists of:

Common Language Runtime – provides an abstraction layer over the operating system

Base Class Libraries – pre-built code for common low-level programming tasks

Development frameworks and technologies – reusable, customizable solutions for larger programming tasks.

The framework's Base Class Library provides user interface, data access, database connectivity, cryptography, web application development, numeric algorithms, and network communications. The class library is used by programmers, who combine it with their own code to produce applications.

## Database/backend - MySQL

MySQL is the world's most popular open source database software, with over 100 million copies of its software downloaded or distributed throughout its history.

The MySQL Community Edition includes:

* Pluggable Storage Engine Architecture
* Multiple Storage Engines: InnoDB , MyISAM, NDB (MySQL Cluster),Memory ,Merge , Archive, CSV
* MySQL Replication to improve application performance and scalability
* MySQL Partitioning to improve performance and management of large database applications
* Stored Procedures to improve developer productivity

## ide for Database –MySQL workbench

MySQL Workbench is a visual database design tool that integrates SQL development,administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. It is the successor to DBDesigner 4 from fabFORCE.net, and replaces the previous package of software,MySQL GUI Tools Bundle.

## Programming Language (C#)

C# is a type-safe, object-oriented language that is simple yet powerful, allowing programmers to build a breadth of applications. C# is a [multi-paradigm programming language](http://en.wikipedia.org/wiki/Multi-paradigm_programming_language) encompassing [imperative](http://en.wikipedia.org/wiki/Imperative_programming), [declarative](http://en.wikipedia.org/wiki/Declarative_programming), [functional](http://en.wikipedia.org/wiki/Functional_programming), [generic](http://en.wikipedia.org/wiki/Generic_programming), [object-oriented](http://en.wikipedia.org/wiki/Object-oriented_programming)([class-based](http://en.wikipedia.org/wiki/Class_(computer_science))), and [component-oriented](http://en.wikipedia.org/wiki/Component-based_software_engineering) programming disciplines. It was developed by [Microsoft](http://en.wikipedia.org/wiki/Microsoft) within the [.NET](http://en.wikipedia.org/wiki/.NET_Framework) initiative and later approved as a standard by [Ecma](http://en.wikipedia.org/wiki/Ecma_International) (ECMA-334) and [ISO](http://en.wikipedia.org/wiki/International_Organization_for_Standardization) (ISO/IEC 23270). C# is one of the programming languages designed for the [Common Language Infrastructure](http://en.wikipedia.org/wiki/Common_Language_Infrastructure).

C# is intended to be a simple, modern, general-purpose, object-oriented programming language.

## Dia for Diagram Drawing & Modeling

Dia is free and open source general-purpose diagramming software, developed as part of the GNOME project's office suite and was originally created by Alexander Larsson. Dia uses a controlled single document interface (CSDI) similar to GIMP and Sodipodi.

Dia has a modular design with several shape packages available for different needs: flowchart, network diagrams, circuit diagrams, and more. It does not restrict symbols and connectors from various categories from being placed together.

Dia is a gtk+ based diagram creation program released under the GPL license.

Dia is inspired by the commercial Windows program 'Visio', though more geared towards informal diagrams for casual use. It can be used to draw many different kinds of diagrams. It currently has special objects to help draw entity relationship diagrams, UML diagrams, flowcharts, network diagrams, and many other diagrams. It is also possible to add support for new shapes by writing simple XML files, using a subset of SVG to draw the shape.

It can load and save diagrams to a custom XML format (gzipped by default, to save space), can export diagrams to a number of formats, including EPS, SVG, XFIG, WMF and PNG, and can print diagrams (including ones that span multiple pages).

## Google Spreadsheet Interface:

*With Google Spreadsheets, we can easily create, share, and edit spreadsheets online. Here are a few specific things we can do:*

* *Import and export these file types: .xls, .csv, .txt and .ods. We can also export data to a PDF or an HTML file.*
* *Format cells and edit formulas so we can calculate results and make data look the way we want it.*
* *Chat in real time with others who are editing our spreadsheet.*
* *Embed a spreadsheet, or a section of a spreadsheet, in our blog or website.*

## Windows Mobile 6 Professional SDK

Windows Mobile is a mobile operating system developed by Microsoft that was used in smartphones and mobile devices. This features a suite of basic applications developed with the Microsoft Windows API. It is designed to be somewhat similar to desktop versions of Windows, feature-wise and aesthetically. Additionally, third-party software development is available for Windows Mobile, and software applications can be purchased via the Windows Marketplace for Mobile. Windows Mobile 6.1 was announced April 1, 2008. It is a minor upgrade to the existing Windows Mobile 6 platform which brings with it various performance enhancements, a redesigned Home screen featuring horizontal tiles that expand on clicking to display more information, although this new home screen is featured only on Windows Mobile Standard edition. This feature was inexplicably left out of the Professional edition. Several other changes such as threaded SMS, full page zooming in Internet Explorer and 'Domain Enroll' have also been added, along with a "mobile" version of the Microsoft OneNote program and an interactive "Getting Started" wizard. Windows Mobile 6.1 also featured improved bandwidth efficiency in its push-email protocol "Activesync" of "up to 40%", this reduced data usage was the cause of considerably improved battery life in many devices.

# GLOSSARY