# Software Requirements Specification

for

# Student's Accounts Management System (SAMS)

Version 1.0

Prepared by

| | | |
|---|---|---|
| Sujay Kumar Mandal (Leader) | M120361CA | SujayKumar.Mandal@gmail.com |
| Arvind Singh | M120352CA | Arvind4friends@gmail.com |
| Rakesh Patni | M120364CA | Rakeshpatni409@gmail.com |
| Shashank Mishra | M120363CA | shashank1790@gmail.com |
| Vishwash Kumar Sharma | M120372CA | vishwassharman172@gmail.com |

| | |
|---|---|
| Instructor: | Mr. Kranti Kumar |
| Course: | Object Oriented Programing |
| Lab Section: | Network System Lab |
| Teaching Assistant: | Mr.  Kranti Kumar |
| Date: | 22/01/2014 |

# Contents

# Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
|---|---|---|---|
| 1.0 | Sujay Kumar Mandal<br>Arvind Singh | Student's Accounts Management System keeps track of some very essential | 22/01/2014 |

| Version | Primary Author(s) | Description of Version | Date Completed |
|---|---|---|---|
| | Rakesh Patni<br>Shashank Mishra<br>Vishwash Kumar Sharma | accounts a college student needs to maintain.<br>This version includes Attendance Tracker, Class Schedule , Monetary Account Manager,  Mess Menu Card , Daily Note Pad & Assignment Reminder. | |

# 1 Introduction

## 1.1 Document Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of the SAMS. This document will cover each of the system's intended features, as well as offer a preliminary glimpse of the software application User Interface (UI). The document will also cover hardware, software, and various other technical dependencies.

## 1.2 Product Scope

The SAMS is standalone software. The system is designed to manage the different accounts a student need to maintain. It provides a lot of benefits to the student.

## 1.3 Intended Audience

- Developers – Sujay Kr. Mandal, Rakesh Patni, Vishwash Sharma, Shashank Mishra, Arvind Singh
- Project Manager – Sujay Kumar Mandal
- Users – NITC MCA 2012-15 Batch Students
- Testers - Sujay Kr. Mandal, Rakesh Patni, Vishwash Sharma, Shashank Mishra, Arvind Singh
- Supervisor – Mr. Kranti Kumar
- Documentation Writers - Sujay Kr. Mandal, Rakesh Patni, Vishwash Sharma, Shashank Mishra, Arvind Singh

## 1.4 Definitions, Acronyms and Abbreviations

| | |
|---|---|
| SAMS | Student Account Management System |
| IEEE | Institute of Electrical and Electronics Engineers |
| GUI | Graphical User Interface |
| CIM | Class Information Management |
| JVM | Java Virtual Machine |

| | |
|---|---|
| MAC | Media Access Control |
| OS | Operating System |
| JDK | Java Development Kit |
| IDE | Integrated Development Environment |
| ER diagram | Entity Relationship Diagram |
| UML | Unified Modelling Language |
| MTTF | Mean Time To Failure |
| MTTR | Mean Time To Recovery |

## 1.5 Document Conventions

In general this document follows the IEEE formatting requirements. Used Arial font size 11, or 12 throughout the document for text. Document text is single spaced and maintained the1" margins. For Section and Subsection titles please follow the CONTENTS.

## 1.6 References and Acknowledgments

[1]capstone.cs.ucsb.edu/cs189a/support/**SRS-template**.doc

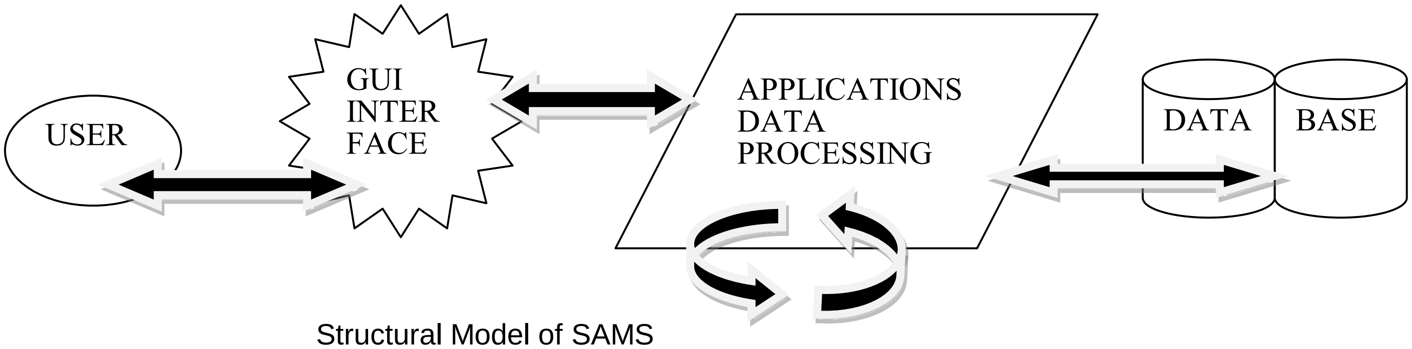
[2]Used the standard IEEE citation guide for this SRS.

[3]The Complete Reference 7th edition by Herbert Schildt (Tata McGraw-Hill)

# 2 Overall Description

## 2.1 Product Perspective

The product Student's Accounts Management System is result of a Green Field project. SAMS is a non-web based application solely for purpose of student's personal use to keep track of attendance , class schedule , monetary account , mess menu .SAMS will also provide facilities to maintain a daily note & also to set assignment reminders . It is a single user application with a single login facility.



Structural Model of SAMS

## 2.2 Product Functionality

- User has to give username & password to login.
- User can update any info recorded.
- User able to logout.
- GUI between user & CIM system.
- SAMS will show the reminders as set on proper date.

## 2.3 Users and Characteristics

SAMS is a single user standalone application. User is able to login using USERNAME & PASSWORD.

## 2.4 Operating Environment

SAMS is platform independent in the sense that you can make a Java application run in Windows, Linux, Mac, and so forth, as long as you don't use libraries specific to one OS, and as long as you have a JVM installed for the appropriate OS to interpret things correctly.

## 2.5 Design and Implementation Constraints

- ➢ JVM need to be installed in the OS.
- ➢ MYSQL Database need to be installed.
- ➢ JDK needed to develop the JAVA application.
- ➢ JAVA language used to implement.

## 2.6 User Documentation

User manual may include a HELP section in the SAMS application GUI . HELP section will describe the functionality of different buttons provided in the GUI. It may also discuss on different entities in the system briefly. Detailed description of all analytic procedure can be documented.

## 2.7 Assumptions and Dependencies

- ❖ Implementation might differ from SRS reasonably, depending on technical difficulties.
- ❖ Completion time depends on learning the required technologies like JAVA.
- ❖ NetBeans IDE used for implementing CIM system JAVA application.
- ❖ Database connectivity & GUI design are two essential parts of implementation process.
- ❖ UML used for modelling the structural design of application.
- ❖ ER diagrams used to describe the database design.
- ❖ Future enhancement scopes will be a main objective during design & implementation.

# 3 Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

- It is important to discuss the interaction between the software and the end users.
- Registered user would be able to get the required and expected information by using the software.
- Guideline for the users which can help them to extract the information.
- References to GUI standards or product family style guides that are to be followed.
- Standard buttons, functions, or navigation links that will appear on every screen, such as a help button.
- Message display conventions.

### 3.1.2 Hardware Interfaces

- Software should be implemented in a hardware-independent fashion and should not rely on any particular hardware interfaces.
- It includes the supported device types, the data and control interactions between the software and the hardware.
- There will be a database  connection required to access the Information by existing storage.
- Required software should be installed properly to the hardware.
- Client Computer OS: Mac, Linux, Windows .

### 3.1.3 Software Interfaces

- Database is required to store the information.
- MYSQL , JAVA required to store/Retrieve the information into the database.
- Supporting Operating system required.
- The interaction between the programs and the system shall be transparent to the user.
- Platform that supports JAVA (JVM need to be installed)

### 3.1.4 Communications Interfaces

- Graphical user interface for user interaction.
- User can explore the information of the class's various activities.
- Software should be accessible in a network simultaneously for the multiple users. (Future enhancement scope)
- Users can Communicate to the Administrator if any modification is required via contact us. (Future scope)

## 3.2 Functional Requirements

- ➢ Student should be able to enter his everyday attendance for every class that he had in a day.
- ➢ Student should have an option to view his current attendance any time.
- ➢ Class time table should be visible and software should able to tell the classes to be held in the current day.
- ➢ There should be an option to make any changes to the existing time table, in case any kind of updating is required.
- ➢ There is an option to maintain a monetary account to have a record of the daily expences of the student.
- ➢ There will be a mess menu card that will show the menu of the mess that student has joined.
- ➢ Student would also be able to change the menu if there is any need required at any time.
- ➢ There should be a daily note pad that would store any special things about the day that student would feel may want to remember for future reference.
- ➢ Student would be able to add any kind of reminders.

## 3.3 Behaviour Requirements

### 3.3.1 Use Case View

- Software shall be capable of completing all the requirements of the user.
- Data can be shortlisted/ sorted on the basis of the requirement of the user.
- Accuracy of the data will be maintained.
- Access for the users will be secured and database/software will be safe always.
- It can be used by the users but can be modified by the administrator only.
- These include properties Like testability, changeability, maintainability, and reusability.
- Software will be able to decide at run time that, which data has to be extracted.

# 4 Other Non-functional Requirements

## 4.1 Performance Requirements

The requirements in this section provide a detailed specification of user interaction with the software and measurements placed on the system performance.

### 4.1.1 Prominent search feature and usage

The home window shall provide the search option .There will be classification of the search based on the sections to which they belong i.e. accounts ,mess etc. The different search options shall be evident ,simple and easy to understand.

### 4.1.2 Response time

- **Database entry**

  Any entry made to the database shall not take more than 2 seconds.

- **Database retrieval**

No retrieval from the database shall take more than 4 seconds.

### 4.1.3 System memory usage

- **Primary memory**

A minimum of 50KB to a maximum of 70KB shall be consumed.

- **Secondary memory**

A minimum of 2500KB to a maximum of 2700KB shall be consumed.

### 4.1.4 Database size

The database shall be built keeping in mind that the maximum duration of a student's stay in the institute is six years. However features shall be added to ensure that enough scalability is provided to the time limit.

## 4.2 Safety and Security Requirements

Only one login and password rights shall be provided in each system. The login requirements shall be prompted whenever the software is installed afresh in the system.

## 4.3 Software Quality Attributes

- **RELIABILITY**

The software should have considerably high MTTF(not less than 1000 hours) and least MTTR(not more than 10 minutes).

- **PORTABILITY**

It should be machine independent :should be compatible with any operating system.

- **FLEXIBILITY**

  The software should be capable of undergoing changes of any nature as and when needed in the future.

- **MAINTAINABILITY**

  Content management system should be embedded with the software.

- **REUSABILITY**

  This software is not intended just for any particular college or course. Hence it has the provision to be reused for any related development.

- **ROBUSTNESS**

  It should be highly resilient to low memory conditions.

- **FAULT TOLERANCE**

  The software is intended to be resistant to and be able to recover from component failure within 10 secs.

# Appendix - Group Log

*Date: 05/01/2014*

First meeting to prepare the SRS comprised of all the members where the topic for the project was decided as Student's Account Management System(SAMS), which would keep information about all the students of the class at the same place . Thereafter SUJAY Kr. MANDAL was chosen as the leader of the group.

*Date: 10/01/2014*

In second meeting a long discussion was held for the various requirements of the project, such as various functional and non functional requirements, hardware requirements and software requirement, etc and also on what all information about the student would be kept.

*Date: 14/01/2014*

In the third meeting the SRS was subdivided into five parts and was distributed among all the five members as one part to each member. Every member was supposed to complete his part in the document before the next meeting.

*Date: 19/01/14*

It was the final meeting where all the parts of the SRS document which were prepared by each member individually were assembled and a composite document was prepared.