Software Requirement Specification And Analysis

Submitted BY SUPERVISED BY

SUMAYA AKHTER SHAMMEE Dr. Mohammed Shafiul Alam Khan BSSE0821 ASSISTANT PROFESSOR

AFIA SAJEEDA

BSSE0832

Student Information Management System for Residential Institute

2018

LLETTER OF TRANSMITTAL

20th MARCH 2018

The SPL-2 Coordinators

Institute of Information Technology

University of Dhaka

Subject: Submission of Software Requirements Specification of Software Project Lab 2

Dear Sir

With due respect, we are pleased to submit the final report on software requirements specification and analysis of “Student Information Management System for Residential Institute (SIMS)”. Although this report may have some lapses, we tried our utmost to submit an acceptable software requirements specification document.

We would be highly obliged if you overlook our mistakes and accept the efforts given in the documentation.

Sincerely yours

Sumaya Akhter Shammee (BSSE0821)

Afia Sajeeda (BSSE0832)

BSSE 8th Batch

Institute of Information Technology

University of Dhaka

Dr. Mohammed Shafiul Alam Khan

Associate Professor and Project Supervisor

ACKNOWLEDGEMENT

At first, we would like to thank almighty for helping us complete this project. We would like to express our deepest gratitude to all those who provided us the support and encouragement to complete this project. Thanks to our Supervisor teacher Dr. Mohammed Shafiul Alam Khan, Associate Professor, Institute of Information Technology, University of Dhaka, whose continuous suggestions and guidance has been invaluable to us. Also special thanks to all of our respectable teachers for their valuable opinions which help us to improve our insight. Without such stimulating suggestions, the project could not have made this much progress. We are grateful to the Institute of Information Technology for giving us the opportunity to do such a project. Lastly, we would like to thank our classmates. Although they were not on our team, they have always been helpful and provided valuable insights from time to time.

ABSTRACT

The study is made for Student Information Management System for Residential Institute. The scope of the study is to analyze the existing student information management systems in the residential institutes (halls) and to know its functions and drawbacks, and to design the SRS of this system. The objective of this study is to develop an SRS (Software Requirements Specification and Analysis) of Student Information Management System for Residential Institute.

Contents

[**Chapter 1: INTRODUCTION** 1](#_Toc509241341)

[1.1 PURPOSE 1](#_Toc509241342)

[1.2 INTENDED AUDIENCE 1](#_Toc509241343)

[1.3 CONCLUSION 1](#_Toc509241344)

[**Chapter 2: INCEPTION OF SIMS** 2](#_Toc509241345)

[2.1 INTRODUCTION 2](#_Toc509241346)

[2.1.1 LISTING DOWN THE STAKEHOLDER 2](#_Toc509241347)

[2.1.2 RECOGNIZING MULTIPLE VIEWPOINTS 2](#_Toc509241348)

[2.1.3 WORKING TOWARDS COLLABORATION 2](#_Toc509241349)

[2.1.4 COMMUNICATION INITIATION 4](#_Toc509241350)

[2.2 CONCLUSION 4](#_Toc509241351)

[**CHAPTER 3: ELICITATION OF SIMS** 5](#_Toc509241352)

[3.1 INTRODUCTION 5](#_Toc509241353)

[3.2 ELICITING REQUIREMENTS 5](#_Toc509241354)

[3.2.1 COLLABORATIVE REQUIREMENTS GATHERING 5](#_Toc509241355)

[3.2.2 PROBLEM IN THE SCOPE 5](#_Toc509241356)

[3.2.3 QUALITY FUNCTION DEPLOYMENT 6](#_Toc509241357)

[3.2.3.3 EXCITING FEATURES 6](#_Toc509241358)

[3.2.4 USAGE SCENARIO 6](#_Toc509241359)

[**CHAPTER 4: SCENARIO-BASED MODELING OF SIMS** 10](#_Toc509241360)

[4.1 INTRODUCTION 10](#_Toc509241361)

[4.2 DEFINITION OF USE CASE 10](#_Toc509241362)

[4.3 USE CASE DIAGRAMS 11](#_Toc509241363)

[4.3.1 LEVEL – 0 USE CASE DIAGRAM – SIMS 11](#_Toc509241364)

[4.3.2 LEVEL – 1 USE CASE DIAGRAM – SUBSYSTEM 12](#_Toc509241365)

[4.3.3 LEVEL – 1.1 USE CASE DIAGRAM – AUTHENTICATION 13](#_Toc509241366)

[4.3.4 LEVEL – 1.2 USE CASE DIAGRAM – INFORMATION SYSTEM 16](#_Toc509241367)

[4.3.5 LEVEL – 1.3 USE CASE DIAGRAM – COMMUNICATION 18](#_Toc509241368)

[4.3.6 LEVEL – 1.4 USE CASE DIAGRAM – User Management 19](#_Toc509241369)

[4.4 ACTIVITY DIAGRAMS 22](#_Toc509241370)

[ACTIVITY DIAGRAM – 1: AUTHENTICATION (Sign Up) 22](#_Toc509241371)

[ACTIVITY DIAGRAM – 2: AUTHENTICATION (Sign In) 23](#_Toc509241372)

[ACTIVITY DIAGRAM – 3: AUTHENTICATION (Sign out) 23](#_Toc509241373)

[ACTIVITY DIAGRAM – 4: AUTHENTICATION (Account Recovery) 24](#_Toc509241374)

[ACTIVITY DIAGRAM – 5: INFORMATION SYSTEM (Store Information) 25](#_Toc509241375)

[ACTIVITY DIAGRAM – 6: INFORMATION SYSTEM (Edit Information) 26](#_Toc509241376)

[ACTIVITY DIAGRAM – 7: INFORMATION SYSTEM (Search Specific Student) 27](#_Toc509241377)

[ACTIVITY DIAGRAM – 8: INFORMATION SYSTEM (View Students’ Details) 28](#_Toc509241378)

[ACTIVITY DIAGRAM – 9: COMMUNICATION (Hall Notice) 28](#_Toc509241379)

[ACTIVITY DIAGRAM – 10: COMMUNICATION (Complaint Form) 29](#_Toc509241380)

[ACTIVITY DIAGRAM – 11: USER MANAGEMENT (Create Board Member) 30](#_Toc509241381)

[ACTIVITY DIAGRAM – 12:USER MANAGEMENT(Enable/ Disable Account) 31](#_Toc509241382)

[4.5 SWIM LANE DIAGRAMS 32](#_Toc509241383)

[SWIM LANE DIAGRAM – 1: SIGN UP 32](#_Toc509241384)

[SWIM LANE DIAGRAM – 2: SIGN IN 33](#_Toc509241385)

[SWIM LANE DIAGRAM – 3: SIGN OUT 33](#_Toc509241386)

[SWIM LANE DIAGRAM – 4: ACCOUNT RECOVERY 35](#_Toc509241387)

[SWIM LANE DIAGRAM – 5: STORE INFORMATION 36](#_Toc509241388)

[SWIM LANE DIAGRAM – 6: EDIT INFORMATION 37](#_Toc509241389)

[SWIM LANE DIAGRAM – 7: SEARCH SPECIFIC STUDENT 38](#_Toc509241390)

[SWIM LANE DIAGRAM – 8: VIEW STUDENTS’ DETAILS 39](#_Toc509241391)

[SWIM LANE DIAGRAM – 9: HALL NOTICE 40](#_Toc509241392)

[SWIM LANE DIAGRAM – 10: COMPLAINT FORM 41](#_Toc509241393)

[SWIM LANE DIAGRAM – 11: CREATE BOARD MEMBER 42](#_Toc509241394)

[SWIM LANE DIAGRAM – 12: ENABLE/DISABLE USER 43](#_Toc509241395)

# **Chapter 1: INTRODUCTION**

This chapter is a part of our software requirement specification and analysis for the project “Pharmacy Management System”. In this chapter, we focus on the intended audience for this project.

## 1.1 PURPOSE

This document briefly describes the Software Requirement Specification and Analysis of Pharmacy Management System. It contains functional, non-functional and supporting requirements and establishes a requirements baseline for the developing the system. The SRS holds the requirements are independent, uniquely numbered and organized by topic. The SRS serves as a platform to forward user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

## 1.2 INTENDED AUDIENCE

This SRS is intended for several audiences including the customers as well as the project managers, designers, developers, and testers.

* The customer will use this SRS to verify that the developer team has created a product that the customer finds acceptable.
* The project managers of the developer team will use this SRS to plan milestones and a delivery date, and to ensure that the developing team is on the right track when developing the system.
* The designers will use this SRS as a basis for creating the system’s design. The designers will continually refer back to this SRS to ensure that the system they are 0designing will fulfill the customer’s demands.
* The developers will use this SRS as a basis for developing the system’s functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created a software that will fulfill all of the customer’s documented requirements.
* The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

## 1.3 CONCLUSION

This analysis of the audience helped us to focus on the users who will be using our analysis. This document will help each and every person related to this project to perceive the subject matter of the project.

# **Chapter 2: INCEPTION OF SIMS**

In this chapter, the Inception part of the SRS will be discussed briefly.

## 2.1 INTRODUCTION

The renowned genius Albert Einstein has said, “If I had an hour to solve a problem I’d spend 55 minutes thinking about the problem and 5 minutes thinking about the solution.” This means, it is more necessary to dig deep into the facts of the problem rather than jumping to providing a solution. Developing efficient software falls under the same jurisdiction.

Inception is the first phase of requirements engineering. It defines the scope and nature of the problem. The principal target of this stage is to create a basic understanding of the problem, identify the people involved and comprehend the nature of the solution via communication.

For a clear perception of the software requirements, a groundwork is established involving the following steps:

* Listing down the stakeholders
* Recognizing multiple viewpoints
* Working towards collaboration
* Breaking the ice and initiating communication

### 2.1.1 LISTING DOWN THE STAKEHOLDER

According to Sommerville and Sawyer [Som97], “Anyone who benefits in a direct or indirect way from the system which is being developed is a stakeholder.” This implies that stakeholders include the end users of the developed software as well as the people whose activities might be influenced by the tool. Towards the end of inception, the list of stakeholders is usually larger as every stakeholder is allowed to suggest one or more individuals who might be probable stakeholders for the given problem. To identify stakeholders, we consulted a number of small-scale pharmacies in Dhaka, Bangladesh and asked them the following questions:

* Who will be using the product?
* Whose work will this project affect?

We identified the following stakeholders for our project.

* Provost, House Tutors, Assistant House Tutors and Hall Administration Officials
* Students

### 2.1.2 RECOGNIZING MULTIPLE VIEWPOINTS

The list of stakeholders will contribute to the input when requirements are elicited. Every stakeholder has different views of the system and achieves different benefits when the system is developed.

### 2.1.3 WORKING TOWARDS COLLABORATION

Each of the stakeholder constituencies (and non-stakeholder constituency) contributes to the requirement engineering process. The greater the numbers of interactions with multiple stakeholders, the higher is the probability of inconsistency, conflicts and clashes of viewpoints. In such circumstances, requirement engineers finalize the requirements following some steps, which are listed below.

* Finding out the commonality and the conflicting points of stakeholders
* Categorizing stakeholders
* Listing down the requirements based on the stakeholder’s priority points

#### 2.1.3.1 Viewpoints of Stakeholders

We have conducted interviews and prepared questionnaires to collect stakeholder demands and opinions.

**House Tutor and Assistant House Tutor Demands**:

* Complete Student Records on the System
* Scopes to query student information via attribute
* Hall Fee Payment Related Information
* Seat vacancy information
* Room history for every student (Under every registration ID, there will be a record of the rooms where the ID holder has been allocated in the past)
* Student Complain Information (will be viewable to Provost, House Tutors and Assistant House Tutors only)
* Financial Report of Every Student (Socio-economic background information that might be needed for to grant students financial support)
* Extra-curricular activities and club related information of students (e.g. sports, debating, cultural activities)
* Online seat application
* Hall Notice Board

**Hall Administration Officials**:

* Complete Student Records on the System
* Scopes to query student information via attribute
* Seat Vacancy Information
* Hall Notice Board
* Student Complain Information (will be viewable to Provost, House Tutors and Assistant House Tutors only)

**Students**:

* Online seat application
* Hall Notice Board
* Notifications about upcoming events
* Complete Student Records on the System
* Student Complain Information (will be viewable to Provost, House Tutors and Assistant House Tutors only)

#### 2.1.3.2 Common points

All stakeholders agreed on the following necessities:

* Complete Student Records on the System
* Scopes to query student information via attribute
* Student Complain Information (will be viewable to Provost, House Tutors and Assistant House Tutors only)
* Hall Notice Board
* Online Seat Application

#### 2.1.3.3 Contradictions

The stakeholders have not strongly contradicted on any of their demands.

#### 2.1.3.4 Final Requirements

* Complete Student Records on the System
* Scopes to query student information via attribute
* Student Complain Information (will be viewable to Provost, House Tutors and Assistant House Tutors only)
* Hall Notice Board

The final requirements define the scope of SIMS.

### 2.1.4 COMMUNICATION INITIATION

In requirements engineering, the involved individuals can be broadly divided into two clusters: the developers and the stakeholders. Coming from different backgrounds, it will be obvious that these two parties will have different points of views regarding the problem. The stakeholders have more knowledge on facing the problem. Meanwhile, the developers are experienced with providing computerized solutions. Thus, in order to obtain an efficient solution to the problem, it is important to ‘loosen up’ or ‘break the ice’ between the two groups.

Following the ideal guidelines of requirement engineering, some context free questions were asked. The context free questions help throwing light on the stakeholders of the project. The next set of questions includes the context itself so that a better understanding of the problem is obtained. The stakeholder is encouraged to voice out his/her opinions about an alternate solution and also provide recommendations to the developer’s suggestions. The final set of questions focuses on the communication activity itself.

## 2.2 CONCLUSION

The intense hours of developing a software is fruitful only if the users are benefitted and satisfied. Jumping into coding, right after signing up for a project throws both the clients and the developers into the risks of failure. A successful project demands a better perception of the problem. The best and easiest way to sketch out the hints of a solution is to interact with those encountering the problem itself. This is where inception phase comes.

Inception phase has given us the opportunity to create a basic understanding of the problem and perceive an abstract idea of the nature of the solution. Direct interaction with the stakeholders made us come across core points of a solution and realize the effectiveness of communication between two parties. We believe that our groundwork will help us implement an efficient solution to the problem.

# **CHAPTER 3: ELICITATION OF SIMS**

After discussing on the Inception phase, we need to focus on the Elicitation phase. So this chapter specifies the Elicitation phase.

## 3.1 INTRODUCTION

The second phase of requirements engineering is elicitation. The main task of elicitation is to combine elements of problem solving, elaboration, negotiation and specification. Gathering information from stakeholders regarding the problem was not sufficient to design the software. The problems that arose, were encountered following the principles of elicitation.

## 3.2 ELICITING REQUIREMENTS

Stakeholders work together to identify the problems, propose elements of the solutions, negotiate different approaches and specify an initial set of solution requirements. This approach is sometimes called Facilitated Application Specification Technique (FAST).

Elicitation has some sub-phases which are:

* Collaborative Requirements Gathering
* Quality Function Deployment
* Usage Scenario
* Elicitation Work Products

### 3.2.1 COLLABORATIVE REQUIREMENTS GATHERING

During inception, basic questions and answers established the scope of the problem. However, some problems persisted about the scope as the boundary of the system was ill defined and the customers have stated some unnecessary confusing detail. Also, customers/stakeholders do not have a proper understanding about the abilities of the computing environment which results in further discussion regarding the problem domain and product requirements. The requirements were put under re-evaluation by doing following tasks.

* Meetings were conducted with stakeholders and we went into further investigation about their requirements and expectations
* They were inquired about the problems with the existing workflow
* The final requirement list was derived at the end of the meeting

### 3.2.2 PROBLEM IN THE SCOPE

A number of problems were encountered in the course of preparing the software requirement specification and analysis of Student Information Management System of Residential institute.

**Scopes**

* Automation of managerial function of residential institute

**Limitations**

* Timeframe is too short to complete the project
* Inflexible existing workflow

### 3.2.3 QUALITY FUNCTION DEPLOYMENT

Quality Function Deployment (QFD) is a quality management technique that translates the needs of the clients into technical requirements for the software. The prime concern of the QFD is customer satisfaction maximization. In order to ensure this, QFD enforces an understanding of what the customers describe as ‘valuable’ and then deploy these values throughout the engineering process. QFD defines three types of requirements:

* Normal Requirements
* Expected Requirements
* Exciting Features

#### 3.2.3.1 NORMAL REQUIREMENTS

Normal requirements refer to the objectives and the goals that are stated for the product during the meeting with the stakeholders. The presence of these requirements ensures the satisfaction of the customers. The normal requirements for the project are stated below.

* Student database
* Student complain form
* Notice board

#### 3.2.3.2 EXPECTED REQUIREMENTS

The requirements that are implicit to the system might not be brought up during the meeting because of their fundamental nature. Despite being not explicitly mentioned, their presence must be ensured. Otherwise, the product will leave customers dissatisfied. These requirements are called expected requirements and these are stated below.

* Error-free in terms of output
* More efficient than the existing workflow
* Authentication and authorization
* User-friendly

### 3.2.3.3 EXCITING FEATURES

The factors that go beyond the customer’s expectations and prove to be satisfying when present are called exciting features. The exciting features are the so called ‘wow factor’ for our project.

No features have been considered for our project at this stage.

### 3.2.4 USAGE SCENARIO

The system will be available on the predetermined IP. The homepage of the system will hold the noticeboard of the respective residential institute, navigation (option) and a Login panel.

##### Contents of the Webpage

The navigation will house the following information: Institute administration (Provost, House Tutors and Assistant House Tutors, System administrator). The Login panel holds the options to sign into the system, register oneself and signing out (if someone is already signed in at that instance).

#### 3.2.4.1 Types of User and Authentication

The functionalities of the system will vary depending on the user. The system supports 3 types of users: Board Member of Administration (e.g. Provost, House Tutors and Assistant House Tutors), System Administrator (a Board Member or Institute Official) and Student. The authentication module will have 4 subsystems: sign up, sign in, sign out and account recovery. A default system administrator account will be registered into the system. The System Administrator will collect the default username and password from the developer.

##### Sign Up

When the student is signing up into the system, the following information needs to be entered: Registration number, Department, Username, password, confirm password, recovery email. The system checks for empty fields. If any field remains empty, the system will display prompt to fill in the particular field. If the entered registration number does not match any of the registration numbers entered into the database by the System Administrator, the sign up attempt will be terminated with a message telling the individual that he/she is unauthorized for that hall. Otherwise the student will be signed up successfully.

##### Sign In

When the user wants to “sign-in”, he/she must undergo authentication. He/she enters his/her respective username and password. The entered data is matched with the corresponding data stored in Database. If entered data matches the stored data, the user gains access to the system.

##### Sign Out

If a user signs out, he/she will be redirected to the homepage. If account activity remains stalled for 30 minutes, user will be automatically logged out (after the stall duration).

##### Account Recovery

In case a user forgets his/her password or username, he/she will be prompted to enter his/her recovery email address. If the entered recovery email address matches with the email present in the database, a recovery passcode will be sent to that email. Otherwise, a failure message will be sent and the user needs to enter his recovery mail again. When the user enters the recovery passcode, if it is matched with the sent code then he/she will gain access to the system again. Then he will be asked to reset his password. If the entered code does not match with the sent code then the user can request for re-sending recovery passcode for at most 5 times.

#### 3.2.4.2 Information System

Data storing, editing, viewing, searching and account management are handled by the Information System. The information system will have the following subsystems: store information, edit information, search specific student and view student details.

##### Store Information

The System Administrator plays the role of entering the data to be stored in Database. Overall, the Student database will contain the following information: university registration no, name, session, department, class roll, status (resident or non-resident), room no, room history, present address, permanent address, local guardian (name, address and contact number), blood group, photo, co-curricular activity and payment. When entering information if any field is left empty the system administrator must fill that empty field. If the entered information (registration number) matches any existing student record, the system will send message that a similar entry exists and prevent duplication of same record. Otherwise, the information will be stored in database. By default, all the students will be non-resident and account status will be disabled. Room number, co-curricular field, local guardian, blood group, payment fields will be kept empty initially. System admin also enters information of board members.

##### Edit information

This functionality will allow users to edit their information like password, recovery email or phone number. Only the board members can edit details of students. The board member can edit the status from non-resident to resident. If the users want to change their passwords, they will be prompted to enter their current password. If the current password has been entered correctly, the system will allow them to change their password. After confirming the new password, the system will update the password to the database. If the new password does not match with the password in the confirm password field, the users have to re-enter the new password.

##### Search Specific Student

This functionality can be used by Board Members and System Administrator to query student information. The board members or the system administrator can search students by their university registration no, name, session, department, class roll, room no. To search a student the board members or the system administrator enters a search key. If the student record against the key exists, the system will display the student record.

##### View Students’ Details

A student can only see his/her own profile. Board Members and System Administrator can view the profiles of all the Students. For this, at first, they have to sign in to system.

#### 3.2.4.3 Communication

The communication module involves matters regarding the noticeboard and complaint form.

##### Noticeboard

The noticeboard can be viewed by both registered users and unregistered visitors. The Board Members of administration manage the content of the noticeboard. Every notice content has a noticeID, date of generating the notice. Most recent five notices will be shown on the home page.

##### Complaint Form

When a student logs into the system, there will be an option to submit a complaint by a form if he/she is a resident student. The following fields will have to be entered into the form: student name, room no. and complaint. The Board Members will be allowed to view complaint forms.

#### 3.2.4.4 User Management

User Management involves tasks associated with creating board members and enabling/disabling accounts.

##### Create Board member

The system administrator will create accounts for board members. The system administrator enters board member information. If the username does not match with any of existing usernames, account will be created. Account information will be stored in the database.

##### Enable/Disable Account

The System Administration has the authority to enable and disable board members’ accounts. Student accounts can be enabled or disabled by their respective house tutors (Board members). To enable or disable any account, the system administrator or the board members search the account. Then the selected account is disabled by the system administrator or the board members.

# **CHAPTER 4: SCENARIO-BASED MODELING OF SIMS**

This chapter describes the Scenario-Based Model for the Student Information Management System of Residential Institute (SIMS).

## 4.1 INTRODUCTION

When developing software, user satisfaction is given the highest priority. The effective method to identify the requirements to establish meaningful analysis and design model is by determining how end user and other actor wants to interact with the system. Thus, requirements modelling begins with scenario generation in the form of use cases, activity diagrams and swim lane diagrams.

## 4.2 DEFINITION OF USE CASE

A Use Case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a Use Case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A Use Case diagram simply describes a story using corresponding actors who perform important roles in the story and makes the story understandable for the users.

The first step in writing a Use Case is to define that set of “actors” that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be described. Actors represent the roles that people play as the system operators. Every user has one or more goals when using the system.

##### PRIMARY ACTOR

Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software.

##### SECONDARY ACTOR

Secondary actors support the system so that primary actors can do their work. They either produce or consume information.

## 4.3 USE CASE DIAGRAMS

Use case diagrams give the non-technical view of overall system.

### 4.3.1 LEVEL – 0 USE CASE DIAGRAM – SIMS

****

Figure 1: Level 0 use case diagram – SIMS

Name: Student Information Management System for Residential Institute(SIMS)

ID: SIMS-L-0

Primary Actors: System Administrator, Board Member, Student

Secondary Actors: None

#### DESCRIPTION OF USE CASE DIAGRAM LEVEL – 0:

After analyzing usage scenario, we found 3 actors who will directly use the system. Primary actors are those who will play action and get reply from the system whereas secondary actors only produce or consume the information.

The actors of Student Information Management of Residential Institute are stated as follows.

* System Administrator –System Admin
* Board Member – BM
* Student

### 4.3.2 LEVEL – 1 USE CASE DIAGRAM – SUBSYSTEM

****

Figure 2: Level 1 use case diagram – Subsystem

Name: Subsystem of SIMS

ID: SIMS-L-1

Primary Actors: System Administrator, Board Member, Student

Secondary Actors: None

#### DESCRIPTION OF USE CASE DIAGRAM LEVEL – 1:

SIMS has 4 subsystems. These are:

1. Authentication
2. Information System
3. Communication
4. User Management

### 4.3.3 LEVEL – 1.1 USE CASE DIAGRAM – AUTHENTICATION

****

Figure 3: Level 1.1 use case diagram – Authentication

Name: Authentication of SIMS

ID: SIMS-L-1.1

Primary Actors: System Administrator, Board Member, Student

Secondary Actors: None

#### DESCRIPTION OF USE CASE DIAGRAM LEVEL – 1.1:

Authentication is a process on which the credentials provided are compared to those on file in a database of authorized users’ information within an authentication server. The authentication subsystem of SIMS can be divided into four parts. These are stated as follows.

* Sign up
* Sign in
* Sign out
* Account recovery

The system houses 3 types of users. The categories are stated as follows.

* System Administrator
* Board Member (Provost, House Tutor, Assistant House Tutor)
* Student

A default system administrator account will be registered into the system. The System Administrator will collect the default username and password from the developer.

##### 1.1.1 Sign Up

When the student is signing up into the system, the following information needs to be entered: Registration number, Department, Username, password, confirm password, recovery email. The system checks for empty fields. If any field remains empty, the system will prompt to fill in the particular field. If the entered registration number does not match any of the registration numbers entered into the database by the System Administrator, the sign up attempt will be terminated with a message telling the individual that he/she is not linked with that hall. Otherwise the student will be signed up successfully.

**Action/ Reply:**

* **Action 1**: Student enters Registration number, Department, Username, password, confirm password, recovery email.
* **Reply 1**: If any field remains empty, the system will prompt to fill in the particular field. If the entered registration number does not match any of the registration numbers entered into the database, the sign up attempt will be terminated. Otherwise the student will be signed up successfully.

##### 1.1.2 Sign In

When the user wants to “sign-in”, he/she must undergo authentication. He/she enters his/her respective username and password. The entered data is matched with the corresponding data stored in Database. If entered data matches the stored data, the user gains access to the system.

**Action/ Reply:**

* **Action**: User enters username and password.
* **Reply**: Entered data is checked whether it is valid or not. If valid, the system will allow the user to sign in.

##### 1.1.3 Sign Out

If a user signs out, he/she will be redirected to the homepage. If account activity remains stalled for 30 minutes, user will be automatically logged out (after the stall duration).

**Action/ Reply:**

* **Action** : User enter the sign out option
* **Reply**: User will be redirected to the home page.

##### 1.1.4 Account Recovery

In case a user forgets his/her password or username, he/she will be prompted to enter his/her recovery email address. If the entered recovery email address matches with the email present in the database, a recovery passcode will be sent to that email. Otherwise, a failure message will be sent and the user needs to enter his recovery mail again. When the user enters the recovery passcode, if it is matched with the sent code then he/she will gain access to the system again. Then he will be asked to reset his password. If the entered code does not match with the sent code then the user can request for re-sending recovery passcode for at most 5 times.

**Action/ Reply:**

* **Action1**: User enters recovery email
* **Reply1**: A recovery passcode will be sent to the entered email if exists in database
* **Action2**: User enters recovery passcode
* **Reply2**: Security code is checked whether it is entered correctly. If the code is entered correctly user will gain access to the system. If the code is not entered correctly then user can request for resending the code.

### 4.3.4 LEVEL – 1.2 USE CASE DIAGRAM – INFORMATION SYSTEM

****

Figure – 4: Level 1.2 use case diagram – Information System

Name: Information System for SIMS

ID: SIMS-L-1.2

Primary Actors: System Administrator, Board Member, Student

Secondary Actors: None

#### DESCRIPTION OF USE CASE DIAGRAM LEVEL – 1.2:

Data storing, editing, viewing and searching are handled by the Information System.

##### 1.2.1 Store Information

The System Administrator plays the role of entering the data to be stored in Database. Overall, the Student database will contain the following information: university registration no, name, session, department, class roll, status (resident or non-resident), room no, room history, present address, permanent address, local guardian (name, address and contact number), blood group, photo,co-curricular activity and payment. When entering information if any field is left empty the system administrator must fill that empty field. If the entered information (registration number) matches any existing student record, the system will send message that a similar entry exists and prevent duplication of same record. Otherwise, the information will be stored in database. By default, all the students will be non-resident and account status will be disabled. Room number, co-curricular field, local guardian, blood group, payment fields will be kept empty initially. System admin also enters information of board members.

**Action/ Reply:**

* **Action**: System admin enters student and board member information
* **Reply**: If any field remains empty, the system admin will be prompted to fill in the particular field. If the entered information (registration number) matches any existing student record, it will be aborted. Otherwise the information will be stored in database.

##### 1.2.2 Edit information

This functionality will allow users to edit their information like password, recovery email or phone number.

Only the board members can edit details of students. The board member can edit the status from non-resident to resident. If the users want to change their passwords, they will be prompted to enter their current password. If the current password has been entered correctly, the system will allow them to change their password. After confirming the new password, the system will update the password to the database. If the new password does not match with the password in the confirm password field, the users have to re-enter the new password.

**Action/ Reply:**

* **Action**: Board member views and edits student details.
* **Reply**: Database is updated.

For changing passwords, the action/ reply will be as follows.

* **Action1**: User enters current password.
* **Reply1**: Entered password is checked with stored password. If current password has been entered correctly, user will be allowed to change password.
* **Action2**: User enters new password and confirms the password in a separate field.
* **Reply2**: If both fields match, the new password will be updated.

##### 1.2.3 Search Specific Student

This functionality can be used by Board Members and System Administrator to query student information.

The board members or the system administrator can search students by their university registration no, name, session, department, class roll, room no. To search a student the board members or the system administrator enters a search key. If the student record against the key exists, the system will display the student record.

**Action/ Reply:**

* **Action**: The board members or the system administrator enters a search key.
* **Reply**: If the student record against the key exists, the student record will be displayed.

##### 1.2.4 View Students’ Details

A student can only see his/her own profile. Board Members and System Administrator can view the profiles of all the Students. For this, at first, they have to sign in to system.

**Action/ Reply:**

* **Action**: Board member or System admin clicks student details option.
* **Reply**: Students’ details are displayed.

### 4.3.5 LEVEL – 1.3 USE CASE DIAGRAM – COMMUNICATION

****

Figure – 5: Level 1.3 use case diagram – Communication

Name: Communication for SIMS

ID: SIMS-L-1.3

Primary Actors: Board Member, Student

Secondary Actors: System Administrator

#### DESCRIPTION OF USE CASE DIAGRAM LEVEL – 1.3:

The communication module involves matters regarding the noticeboard and complaint form.

##### 1.3.1 Noticeboard

The noticeboard can be viewed by both registered users and unregistered visitors. The Board Members of administration manage the content of the noticeboard. Every notice content has a noticeID, date of generating the notice. Most recent five notices will be shown on the home page.

**Action/ Reply:**

* **Action**: Board member generate notice
* **Reply**: Database is updated with notice.

##### 1.3.2 Complaint Form

When a student logs into the system, there will be an option to submit a complaint by a form if he/she is a resident student. The following fields will have to be entered into the form: student name, room no. and complaint. The Board Members will be allowed to view complaint forms.

**Action/ Reply:**

* **Action1**: Resident student fills up complaint form.
* **Reply1**: All the fields are checked whether they are filled. If any field remains empty, the system will prompt to fill in the particular field.
* **Action2:** Resident student presses the submit button.
* **Reply2**: Complaint will be submitted successfully.

### 4.3.6 LEVEL – 1.4 USE CASE DIAGRAM – User Management

****

Figure – 6: Level 1.4 use case diagram – User Management

Name: User Management for SIMS

ID: SIMS-L-1.4

Primary Actors: System Administrator, Board Member

Secondary Actors: Student

#### DESCRIPTION OF USE CASE DIAGRAM LEVEL – 1.4:

User Management involves tasks associated with creating board members and enabling/disabling accounts.

##### 1.4.1 Create Board member

The system administrator will create accounts for board members. The system administrator enters board member information. If the username does not match with any of existing usernames, account will be created. A mail containing username and password will be sent to the respective board member. The account information will be stored in the database.

**Action/ Reply:**

* **Action1**: The system administrator enters username and password
* **Reply1**: If the username does not match with any of existing usernames, account will be created.
* **Action2**: A mail containing username and password will be sent to the respective board member.
* **Reply2**: The account information will be stored in the database

##### 1.4.2 Enable/Disable Account

The System Administration has the authority to enable and disable board members’ accounts. Student accounts can be enabled or disabled by their respective house tutors (Board members). To enable or disable any account, the system administrator or the board members search the account. Then the selected account is disabled by the system administrator or the board members.

**Action/ Reply:**

* **Action1**: The system administrator or the board members disable/enable the account
* **Reply1**: Selected account is enabled/disabled

## 4.4 ACTIVITY DIAGRAMS

### ACTIVITY DIAGRAM – 1: AUTHENTICATION (Sign Up)



Figure – 7: Level 1.1.1 Activity diagram – Sign Up

### ACTIVITY DIAGRAM – 2: AUTHENTICATION (Sign In)



Figure – 8: Level 1.1.2 Activity diagram – Sign In

### ACTIVITY DIAGRAM – 3: AUTHENTICATION (Sign out)



Figure – 9: Level 1.1.3 Activity diagram – Sign Out

### ACTIVITY DIAGRAM – 4: AUTHENTICATION (Account Recovery)

****

Figure – 10: Level 1.1.4 Activity diagram – Account Recovery

### ACTIVITY DIAGRAM – 5: INFORMATION SYSTEM (Store Information)



Figure – 11: Level 1.2.1 Activity diagram – Store Information

### ACTIVITY DIAGRAM – 6: INFORMATION SYSTEM (Edit Information)

****

Figure – 12: Level 1.2.2 Activity diagram – Edit Information

### ACTIVITY DIAGRAM – 7: INFORMATION SYSTEM (Search Specific Student)

****

Figure – 13: Level 1.2.3 Activity diagram – Search Specific Student

### ACTIVITY DIAGRAM – 8: INFORMATION SYSTEM (View Students’ Details)

****

Figure – 14: Level 1.2.4 Activity diagram – View Student Details

### ACTIVITY DIAGRAM – 9: COMMUNICATION (Hall Notice)

****

Figure – 15: Level 1.3.1 Activity diagram – Hall Notice

### ACTIVITY DIAGRAM – 10: COMMUNICATION (Complaint Form)

****

Figure – 16: Level 1.3.2 Activity diagram – Complaint Form

### ACTIVITY DIAGRAM – 11: USER MANAGEMENT (Create Board Member)

****

Figure – 17: Level 1.4.1 Activity diagram – Create Board Member

### ACTIVITY DIAGRAM – 12:USER MANAGEMENT (Enable/ Disable Account)

****

Figure – 18: Level 1.4.2 Activity diagram – Enable/Disable Account

# 4.5 SWIM LANE DIAGRAMS

### SWIM LANE DIAGRAM – 1: SIGN UP

****

Figure – 19: Swim lane diagram – Sign up

### SWIM LANE DIAGRAM – 2: SIGN IN

****

Figure – 20: Swim lane diagram – Sign in

### SWIM LANE DIAGRAM – 3: SIGN OUT

****

Figure – 21: Swim lane diagram – Sign out

### SWIM LANE DIAGRAM – 4: ACCOUNT RECOVERY

****

Figure – 22: Swim lane diagram – Account Recovery

### SWIM LANE DIAGRAM – 5: STORE INFORMATION

****

Figure – 23: Swim lane diagram – Store Information

### SWIM LANE DIAGRAM – 6: EDIT INFORMATION

****

Figure – 24: Swim lane diagram – Edit Information

### SWIM LANE DIAGRAM – 7: SEARCH SPECIFIC STUDENT

****

Figure – 25: Swim lane diagram – Search Specific Student

### SWIM LANE DIAGRAM – 8: VIEW STUDENTS’ DETAILS

****

Figure – 26: Swim lane diagram – View Students’ Details

### SWIM LANE DIAGRAM – 9: HALL NOTICE

****

Figure – 27: Swim lane diagram – Hall Notice

### SWIM LANE DIAGRAM – 10: COMPLAINT FORM

****

Figure 28: Swim lane diagram – Complaint Form

### SWIM LANE DIAGRAM – 11: CREATE BOARD MEMBER

****

Figure – 29: Swim lane diagram – Create Board Member

### SWIM LANE DIAGRAM – 12: ENABLE/DISABLE USER

****

Figure – 30: Swim lane diagram – Enable/Disable User

# **CHAPTER 5: DATA BASED MODELING OF SIMS**

This chapter describes the Data Based Model for the Student Information Management System for Residential Institute.

## 5.1 INTRODUCTION

Sometimes software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated. The software team chooses to create data models as a part of overall requirements modelling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

## 5.2 DATA OBJECTS

A data object is a representation of composite information that must be understood by the software. Here, composite information means an information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

### 5.2.1 NOUN PARSING AND ANALYSIS

We identified all the nouns to find whether they are in problem space or in solution space from our usage scenario. Attributes belonging to the problem space are denoted by p. Attributes belonging to the solution space are denoted by s.

Table 1: Noun Identification

|  |  |  |  |
| --- | --- | --- | --- |
| Serial no. | Noun | Problem/  solution | Attributes |
| 1 | User | s | 5,6,11,12,15,18,32 |
| 2 | System administrator | s | 5,6,11,12,15,18,32 |
| 3 | Board member | s | 5,6,11,12,15,18,32 |
| 4 | Student | s | 4,5,9,10,11,12,15,18-27,31,32,53 |
| 5 | Username | s |  |
| 6 | Password | s |  |
| 7 | System |  | 51 |
| 8 | Information | p |  |
| 9 | Reg no. | s |  |
| 10 | Department | s |  |
| 11 | Confirm Password | s |  |
| 12 | Recovery email | s |  |
| 13 | Authentication | s |  |
| 14 | Homepage | s |  |
| 15 | Stall duration | s |  |
| 16 | Failure Message | p |  |
| 17 | Recovery passcode | s |  |
| 18 | Name | s |  |
| 19 | Session | s |  |
| 20 | Class roll | s |  |
| 21 | Status | s |  |
| 22 | Resident | s |  |
| 23 | Non-resident | s |  |
| 24 | Room no. | s |  |
| 25 | Room history | s |  |
| 26 | Present address | s |  |
| 27 | Permanent address | s |  |
| 28 | Local guardian | s | 18,29,30 |
| 29 | Address | s |  |
| 30 | Contact number | s |  |
| 31 | Blood group | s |  |
| 32 | imageFileName | s |  |
| 33 | Search key | s |  |
| 34 | Student record | s |  |
| 35 | Profile | p |  |
| 36 | Visitor | p |  |
| 37 | Content | s |  |
| 38 | Noticeboard | s | 37,39,40, |
| 39 | Notice id | s |  |
| 40 | Date | s |  |
| 41 | Complain form | s | 18,24,40,42 |
| 42 | Complaint | s |  |
| 43 | Account | s |  |
| 44 | Default | p |  |
| 45 | Enable account | s |  |
| 46 | Disable account | s |  |
| 47 | Data | p |  |
| 48 | Sign out | s |  |
| 49 | Sign in |  |  |
| 50 | Account recovery |  |  |
| 51 | Message | s |  |
| 52 | Resident student | s | 4,5,9,10,11,12,15,18-22,31,32,53 |
| 53 | Payment Info |  |  |
| 54 | Provost | s | 5,6,11,12,15,18,32 |
| 55 | House Tutor | s | 5,6,11,12,15,18,32 |
| 56 | Assistant House Tutor | s | 5,6,11,12,15,18,32 |

### 5.2.2 POTENTIAL DATA OBJECTS

The following data objects were obtained after grammatical parsing of nouns.

* System: message
* Notice: content, noticeID, date
* User: username, name, password, recovery email, stall duration
* System Administrator: username, name, password, recovery email, stall duration, imageFileName
* Board Member: username, name, password, recovery email, stall duration, imageFileName
* Student: registrationNumber, name department, class roll, present address, permanent address, status, blood group, username, password, confirm password, recovery email, imageFileName, paymentInfo
* Local guardian: name, address, contact number
* Resident student: registrationNumber, room number, local guardian, blood group, paymentInfo
* Complain form: complainID, complaint, date

### 5.2.3 ANALYSIS FOR FINAL DATA OBJECT

* Non-registered user can only view the homepage that contains noticeboard, login panel and navigation
* Provost, House Tutors and Assistant House Tutors are Board Members of the Institute Administration and their activities are more or less the same. This allows us to merge Provost, House Tutors and Assistant House Tutors as a Board Member data object.
* Local guardian information is treated as a separate entity for a better understanding
* Student has Local guardian
* Complain form has complaintID, date and complaint as attributes

### 5.2.4 FINAL DATA OBJECTS

Table 2: Final Data Objects

## 5.3 DATA OBJECT RELATIONS

Data objects are connected to one another in the ways stated below.

Figure – 31: Relationship between Data objects

## 5.4 ENTITY RELATIONSHIP DIAGRAM

Figure – 32: Entity Relationship Diagram

## 5.5 SCHEMA DIAGRAM

A schema is the structure behind data organization. In a schema diagram, all database tables are designated with unique columns and special features, e.g. primary keys, foreign keys.