Software Requirements Specification

Review Management System (RMS)

Version 1.1

Prepared by Suraj Prakash Sharma (U101114FCS144)

Sandeep Kumar (U101114FCS213)

Rajat Sinha (U101114FCS114)

Rishabh Gangwar (U101114FCS116)

Sajal Kumar Mishra (U101114FCS171)

NIIT University

Date:11 Sept 2016

Table of Contents

Revision History

Name	Date	Reason for Changes	Version
Suraj Prakash Sharma	26 Nov 16	Changes in Requirements	V1.1

1. Introduction

1.1 Purpose

The purpose of this document is to present the brief overview of our "Online Rating System" for University. This document will explain the purpose and features of the system, interface of the system, why we are making this system, benefits and how the user will interact with our system and also how the system will interact with the external environment.

This SRS will explain the whole system with respect to the developer and user.

1.2 Document Conventions

- o Convention for Heading:
  Font Name: Times New Roman.
  Size: 18.

- o Convention for Sub-Heading:
  Font Name: Times New Roman.
  Size: 14.

- o Convention for Body:
  Font Name: Calibri (Body).
  Size: 12.

- o Entire document is justified.

1.3 Intended Audience and Reading Suggestions

This document is made for developers and all those people who are related to the developer team directly or indirectly, so that they can understand our whole system.

Our target customers are mainly students, professors or anyone who is related to the university directly or indirectly.

The best way to read the document is just go section by section as arranged.

1.4 Product Scope

The scope of the product is limited as the product is only for our university. But the main goal behind this product is to make the services at the university better for the near future.

We are providing a rating system for each of the services available at the university like:

Library, Mess, Laundry, General Stores (HOD, Tuck Shop, TMP, Apno Gaon), E-Services (Moodle, ERP), Sport Services etc.

We have done a feasibility analysis on our product to check whether our product is good and worth go for implementation with respect to the user or not and we got positive response which you can see in our feasibility report.

For implementing this web system, the programming language which we are using are HTML5, CSS, JavaScript, PHP, SQL.

1.5 References

1) This project is under the guidance of Prof. Amit Kumar, CSE Department NIIT University.

2) https://docs.google.com/spreadsheets/d/1jCc_GKTm4WWuRAlPRDQrx2cclIcSMXpUrm135U72b08/edit#gid=1797016161

3) Feasibility Report Google Docs Link: https://docs.google.com/a/st.niituniversity.in/document/d/1a584m_e2ykBGV2MdasRQtny0WYSuxWPQoulvUtSI3X0/edit?usp=sharing

2. Overall Description

2.1 Product Perspective

Use Case Diagram of the whole system.

Main Actors: User, Student, Authority

Full System Overview
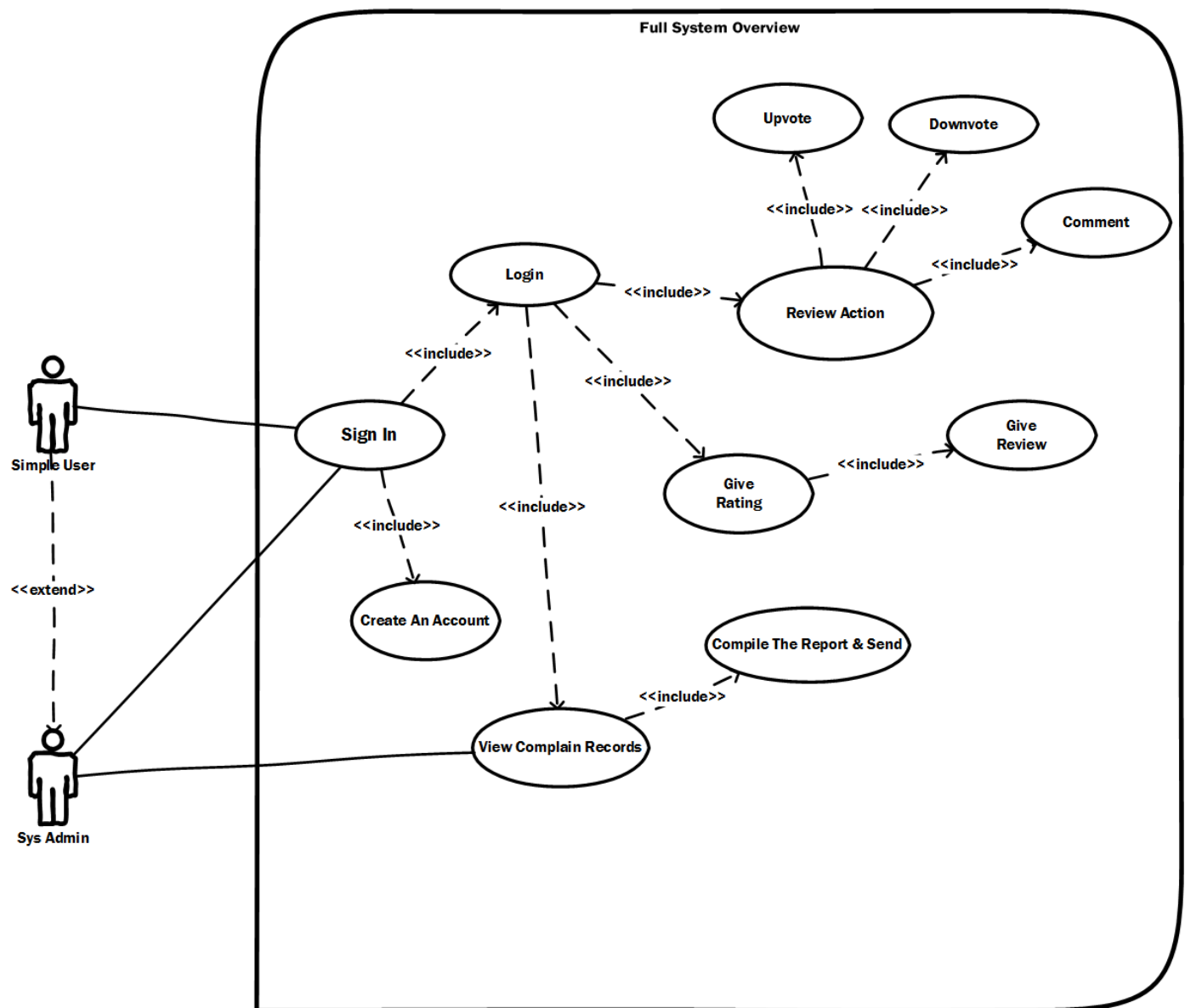
E-R Diagram:

2.2 Product Functions

- o Secure Login/Signup.
- o Allow rating of various services.
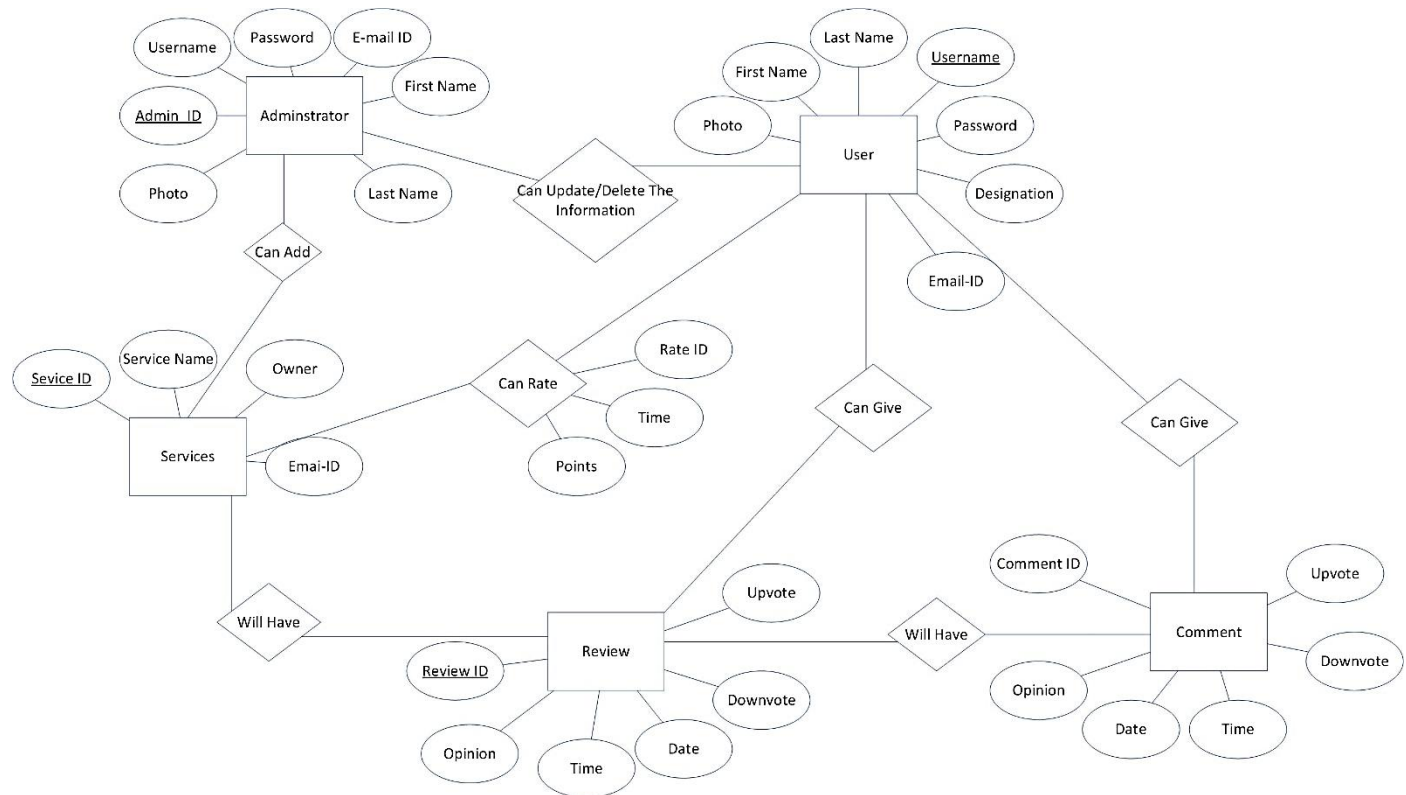- o Allow user to upvote/downvote/comment on various services and reviews.
- o Sending the summary report of a particular service to the concerned organization.
- o Allow administrator to check & view the complains given by the user to every services.

2.3 User Classes and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

2.4 Operating Environment

Every user must have a computer with internet access with the following minimum requirements:

- o Operating System: Windows 8.1 or above.

o   Browser: Latest versions of Internet Explorer, Google Chrome or Mozilla Firefox.
Our Web application will be compatible with mobile phones with internet also.

2.5 Design and Implementation Constraints

- o   One constraint for our web application that is the database which we are using (Relational Database) to store the information provided by the user, because if the number of user increases above some threshold point, then maybe we need to optimize our database.
- o   Security issues related to SQL Injection, for that we need to design our database with the updated PHP version.

2.6 User Documentation

- o   To help user to understand how our website works, we will upload a small video of 2-3 min in our homepage through which user will be able understand clearly understand how our website works and how user can interact with our website and also the goal of our website.
- o   Terms and Conditions.

2.7 Assumptions and Dependencies

- •   It is assumed that all users and the administrator are part of the NIIT University directly or indirectly.
- •   It is assumed that the service would keep backup of all the MySQL data in a different computer.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 GUI- [Images to be added] We are actually working on the user interface in order to make it as simple and elegant as possible for the user. So, this part will be updated soon.

3.2 Hardware Interfaces

3.2.1 Computers with Internet access: Every user must have a computer with internet access with the following minimum requirements:

- o   Operating System: Windows 8.1 or above.

- o   Browser: Latest versions of Internet Explorer, Google Chrome or Mozilla Firefox.

3.2.2. Firewall – Since security is one of our primary concern, a hardware firewall will be setup to stop unwanted user to access the service. This helps in securing the service from

- o   DoS attack, DDoS attack and a firewall is also useful for allowing remote access to a private network through secure authentication certificates and logins.

3.3 Software Interfaces

 3.3.1 **Apache Web Server:** Since the service will be set on a server so, we need a software to run the server that deals with all the MySQL connectivity and request/ reply work.

 3.3.2 **MySQL Servers:** It is necessary to keep all the user and application data stored to be accessible when needed. The whole project depends upon coding of MySQL scripts.

 3.3.3 Web Compiler: The project is nothing but consisting of files which would be coded on a compiler or debugger for maintain flexibility of project.

 3.3.4 **Browser:** User will need to browser which will take care of the interaction with server and run the files provided by server. Hence a browser will take care of all the port management, packet processing and socket programming.

 3.3.5 Firewall: A software firewall for extra security (See 3.2.2)

3.4 Communications Interfaces

 3.4.1 **HTTPS:** Our website will use HTTPS protocols to communicate with the server.

4. System Features

Requirement ID	Name of the Requirements	Short Description	Use Case ID[1]
4.1	Secure Login Feature	It will provide user a high-end security.	UC-1
4.2	Rate & Review Services	Backbone of our system. Through this feature user will be able to rate (out of 5) & review (mandatory) the services.	UC-2
4.3	Upvote/Downvote & Comment on Reviews	This is attached to every review given by the user. This feature is optional with respect to the user.	UC-3
4.4	Sending Review Summary to the concerned authority.	This helps us achieving our goal which is "To Make Services in The Near Future Better".	UC-4

---

[1] All the use-cases with their respective ID's are made below the table.

Use Case Diagrams:

Code-UC1

Description of the Use Case:

User needs to sign up/login in our website to use our services now, to make user information secure and genuine, we ask them to provide some information regarding them while signup like:

    I.   Name

   II.   Last Name

  III.   E-mail ID (According to the valid universal convention for any email)

  IV.   Username (Unique for every user and chosen by the user)

   V.   Password (We will give some guidelines to make a secure password)

  VI.   Photo

 VII.   Institution (Were user study)

VIII.   Designation (Student, Teacher or faculty, Other)

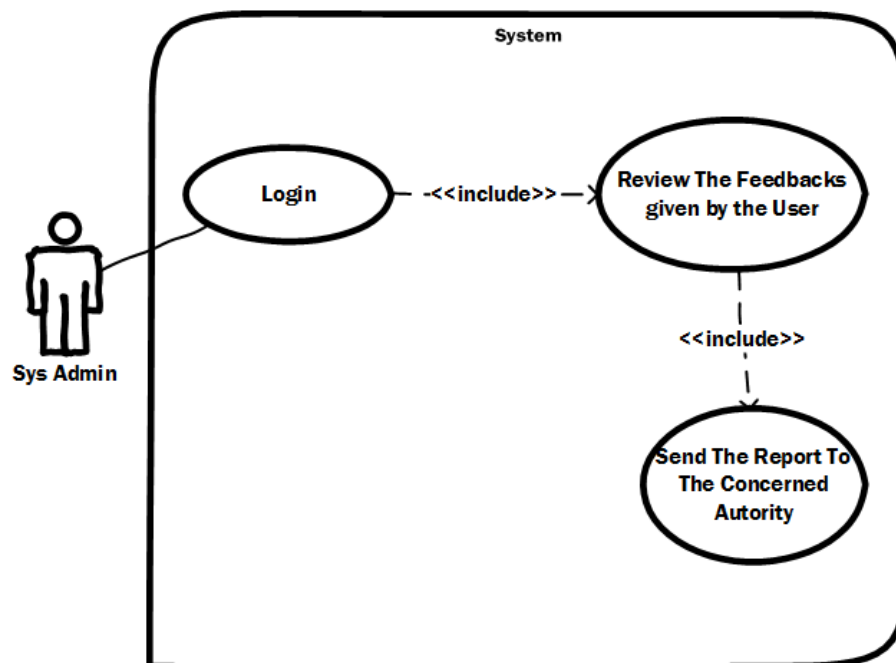 IX.   For security purpose, we ask them to fill the generated CAPTCHA.

CAPTCHA will take care of bot's; it will differentiate whether the user is a human being or a bot.

But, there will be one more problem with this type of signup system, that is what if the user is fake, so to solve this problem, our system will only create the account of the desired user only if user confirms that the email ID which user has provided is genuine.
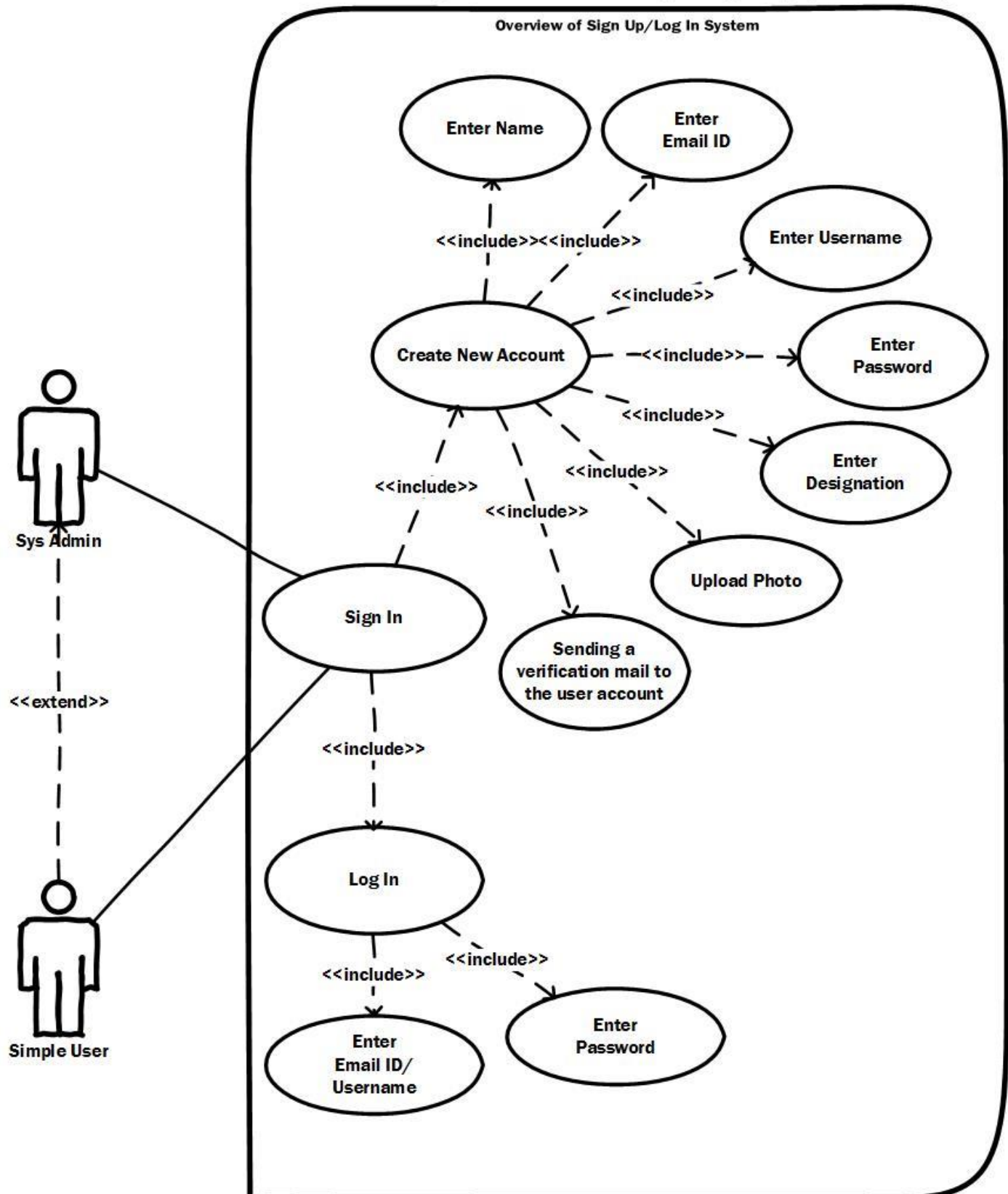
And to check the genuinity of user's email, we will send a confirmation link to the email address which user provided at the time of signup and if user confirms it by opening his email account, then the profile will be created for the desired user, and after that user can use our services.

When user already has an account, he just needs to login through his username, password and write the shown CAPTCHA for security purpose.

The Use Case Diagram for this system is shown below:

Use Case Diagram For Secure Login/Sign Up (Code-UC 1)

Overview of Sign Up/Log In System

Enter Name

Enter Email ID

Enter Username

Create New Account

<<include>> <<include>>

<<include>>

<<include>>

Enter Password

<<include>>

Enter Designation

<<include>>

<<include>>

<<include>>

Upload Photo

Sign In

Sending a verification mail to the user account

Sys Admin

<<extend>>

<<include>>

Log In

<<include>>

<<include>>

Enter Email ID/ Username

Enter Password

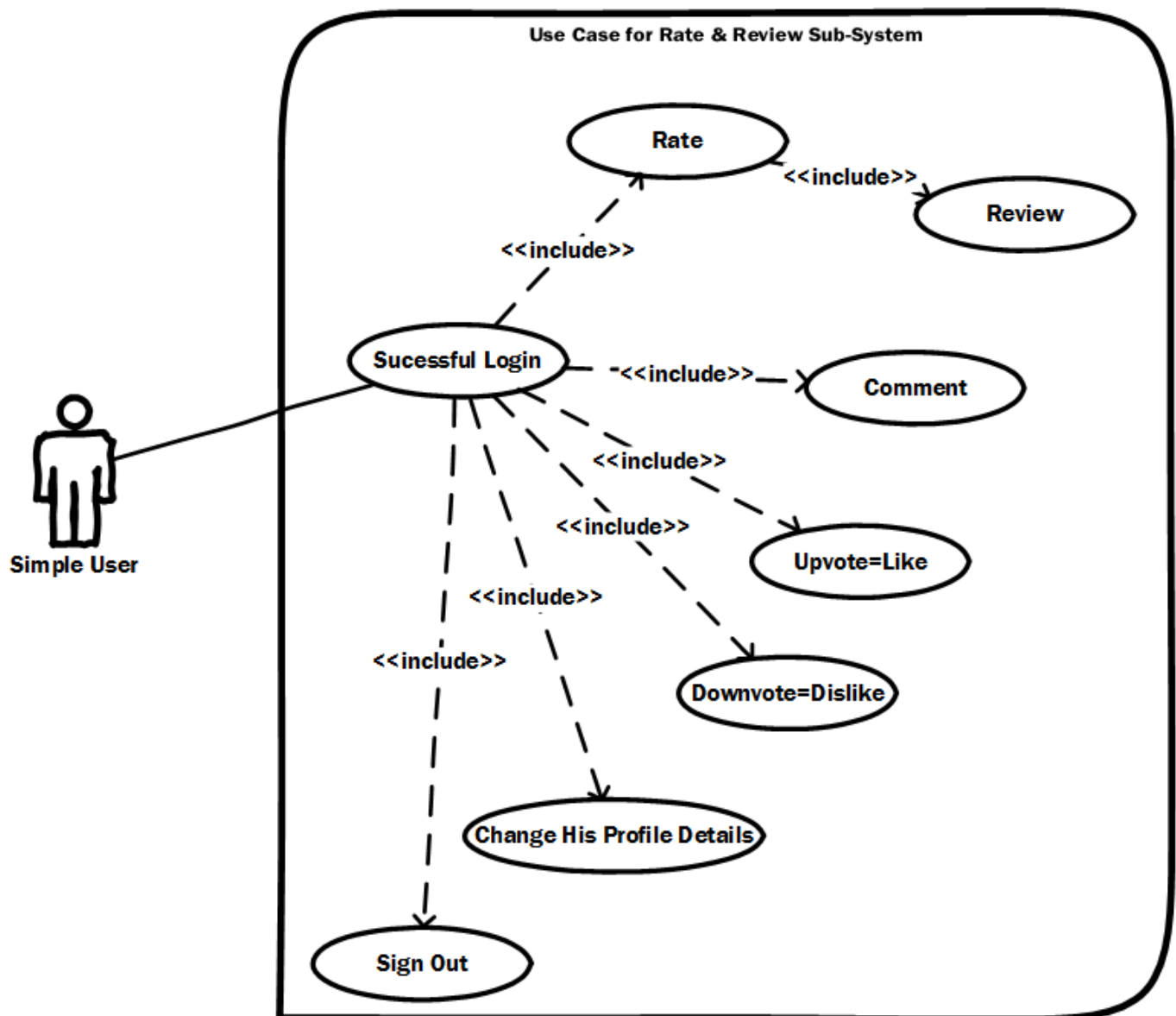Simple User

Code-UC-2

Description of the Use Case:

This feature is the backbone of our system. Through this feature, user will be able to rate and give review (must) about the desired service.

While rating the system, we are not giving anyone a high priority, everyone (user, high authority) is same with respect to the system.

After the successful and secure login/sign up, user will be redirected to the homepage of the system where he will see some tabs, which represent different services provided by the university.

After clicking the service, the user will be redirected to the new page (known as the rating page) where he can rate (out of 5), give review, upvote, downvote, or comment on the review.

User will be able to give the rating out of 5 for the given service, and can write the review for that service or sub-service is optional.

Use Case for Rate & Review Sub-System

5. Other Non-Functional Requirement's

Requirement ID	Name of Requirement	Description
5.1	Performance Requirements	• **Application Services and Technical Support**- Developers will have access to source code to address bugs or system enhancements as necessary. Network Administrator and Database Administrator (DBA) support is also required to maintain system up to point 24x7.
5.2	Safety Requirements	• **Backup Data Storage**- In case of any damage to the system servers, a backup facility will ensure that the service is back to work as soon as possible • **Code Files**- In case of future development a copy of efficiently commented code is kept. This way any new person can understand the working model. And in case of any damage to the hardware systems, the service can be easily implemented again through backups.
5.3	Security Requirements	• **Firewall**- A firewall is needed to stop unwanted users or entities from accessing the service. A physical hardware firewall is required rather than the software's available. • **Administration Features**- System security and access levels are provided in the online system. There are varying levels of system access and functional authority. Each student's access is limited to his/her own registration records. Only authorized system administrator(s) has access to all student registration records. • **Robot File**-The robot.txt file ensures that the web service is safe from web crawlers. And this will take care of many scanning problems.
5.4	Software Quality Attributes	• **Bug/ERROR Report Feature**- Any time the service dysfunction's, a crash report will be sent to the development team to improve the product. Crash report will be sent by the user either automatically or manually depending on the settings set by the user. But if there is any bug report, then user needs to send it by e-mail so that we can admire the user.
5.5	Business Rules	• **Student**- They are only provided with the feature to rate, review, upvote/downvote and comment about the services and reviews. • **Authority**- They can read all the reviews sorted by a sorting algorithm on basis of upvote/downvote frequency (Reviews which has the most upvotes will be displayed first then the review which has less upvote and so on) and he can take his decision on the basis of the summary report send by our system.

6. Other Requirements

6.1 Hardware Requirements

- Network-University network infrastructure (Wired and Wireless).

- Server-A server is needed to deploy the service.

- Client Computers-Mac, Unix and Windows client computers.

Appendix A: Glossary

1) UC-Stands for Use-Case Diagrams which helps us understand the system more clearly.

2) CAPTCHA-Stands for Completely Automated Practical Turing Test to Tell Computers and Human Apart.

3) SRS-Software Requirement Specification.

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List