



---

# Software Requirements Specification

for

## Gym Management System

Version 1.0.0

Prepared by

**Roll Number:** B190474CS

**Name:** Prashant Shekahr

**Project Owner:** Piyush Keshwani

**Course:** CS4097D Object Oriented Systems Laboratory

**Date:** 06/10/2022

<b>1 Introduction</b>	<b>3</b>
1.1 Document Purpose	3
1.2 Product Scope	3
1.3 Definitions, Acronyms and Abbreviations	3
1.4 Document Conventions	3
1.5 References and Acknowledgments	3
<b>2 Overall Description</b>	<b>4</b>
2.1 Product Overview	4
2.2 Product Functionality	4
<b>3 Specific Requirements</b>	<b>5</b>
3.1 User Interfaces	5
3.1.1 login page	5
3.1.2 sign up page	5
3.1.3 Home - page	6
3.1.4 Add Member page	6
3.1.5 Add Trainer	7
3.1.6 Payment :	8
3.2 Functional Requirements	8
3.3 Use Case Model	10
3.3.1 Use Case #1	10
3.3.2 Use Case #2	11
3.3.3 Use Case #3	11
3.3.4 Use Case #4	11
3.3.5 Use Case #5	11
3.3.6 Use Case #6	11
3.3.7 Use Case #7	12
3.3.8 Use Case #8	12
3.3.9 Use Case #9	12
3.3.8 Use Case #10	12

# **1 Introduction**

The "Gym Management System" is a desktop-based java project and has been deployed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and in some cases reduce the hardships faced by the existing system.

## **1.1 Document Purpose**

This documentation covers the features and functionality of Gym Management System v1.0.0. The purpose of the project is to maintain the details of the Gym . The main purpose of this project is to maintain ease.

## **1.2 Product Scope**

Gym management system is one of the most effective ways for modern gyms to achieve success and optimize their business potential and it has great future scope as well. This technology has progressed well beyond checking members into the gym and processing dues.

## **1.3 Definitions, Acronyms and Abbreviations**

- UI/UX - User Interface/User Experience
- F1 - Functional Requirement 1

## **1.4 Document Conventions**

In general this document follows the IEEE formatting requirements. Arial font size 11, or 12 is used throughout the document for text. Times font size 18 is used for main headings. Italics are used for comments. Document text is kept single-spaced and the 1" margins are maintained in this document.

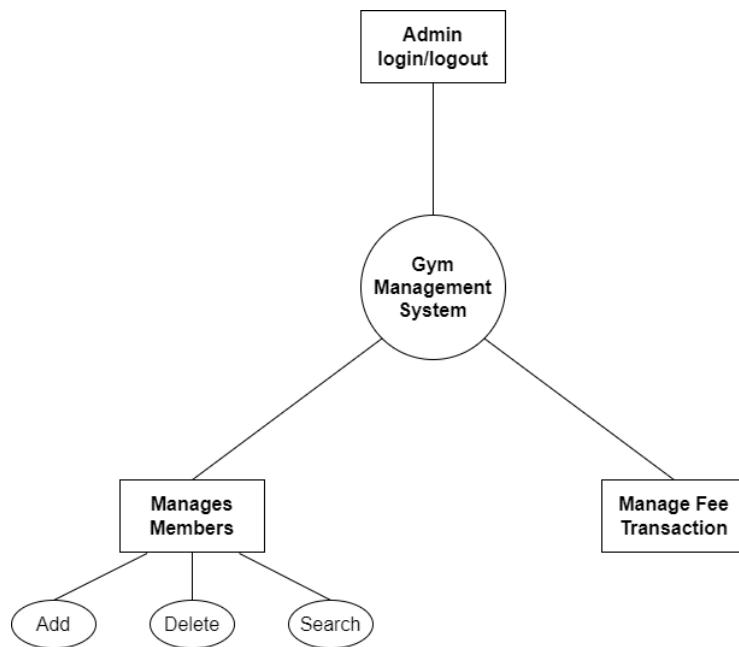
## **1.5 References and Acknowledgments**

- <http://www.cse.msu.edu/~cse870/IEEEExplore-SRS-template.pdf> (Courtesy: Dept. of CSE, Michigan State University)
- <https://app.diagrams.net/>

## 2 Overall Description

### 2.1 Product Overview

The Gym management System has an Admin module and one relational database. Admin uses a login portal to login with their ID and Password after successful Sign Up. Now Admin can see the Total Members, Add, Update, Delete, Search Member and can Add, check Fee Details, change his/her password and can logout. A relational Database is used to store all the data such as list of registered members, etc.



### 2.2 Product Functionality

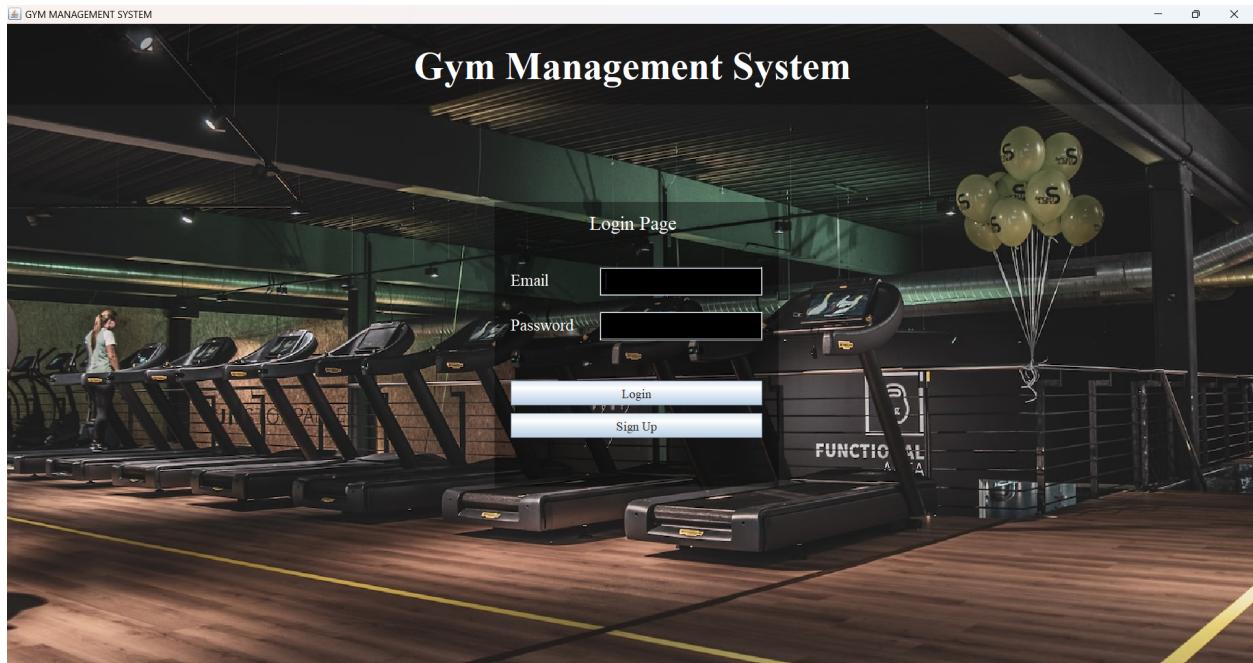
- The Gym Management System allows admin to Sign Up and then Login to the System.
- It allows admin to manage the members like see the Total Members, Add, Update, Delete, Search Member.
- Admin can Add and Check Fee Details, change his password and can logout.

# 3 Specific Requirements

## 3.1 User Interfaces

### 3.1.1 login page

Admin will login the application.



### 3.1.2 sign up page

Admin will login the application.

GYM MANAGEMENT SYSTEM

### Admin Sign Up

Email

Name

Password

Confirm

**Create New Account**

### 3.1.3 Home - page



### 3.1.4 Add Member page

The Member will be registered by Admin.

The screenshot shows a Windows application window titled "GYM MANAGEMENT SYSTEM". At the top, there are three radio buttons: "Add Member" (selected), "Add Trainer", and a third option with a "choose" button. Below this, the title "Add Member" is centered. The form contains six input fields arranged in two rows of three. The first row has "Name" and "Email" (both with dropdown arrows) on the left, and "Address" on the right. The second row has "Gender" (with a dropdown arrow) and "DOB" on the left, and "Time" and "DOJ" (both with dropdown arrows) on the right. At the bottom are two buttons: "Submit" and "Cancel".

### 3.1.5 Add Trainer

The Trainer will be registered by admin.

This screenshot shows the same "GYM MANAGEMENT SYSTEM" window, but the "Add Trainer" radio button is selected. The title "Add Trainer" is centered above the form. The input fields are identical to the "Add Member" form: Name, Email, Address, Gender, DOB, Time, and DOJ. The "Submit" and "Cancel" buttons are at the bottom.

### **3.1.6 Payment :**

Payment entry will be entered by Admin.

The screenshot shows a window titled "GYM MANAGEMENT SYSTEM" with a sub-title "Payment". The window contains four input fields: "Member Id" (text box), "Date" (text box), "Month" (dropdown menu showing "January"), and "Amount" (dropdown menu showing "650"). At the bottom are two buttons: "Submit" and "Cancel".

## **3.2 Functional Requirements**

### **F1: Admin SignUp**

Initially, when we run the project, the application will ask the admin to sign up. The admin can sign up using the sign up portal. After that, all the details of the admin will be added to the database and he/she will be redirected to the login page with a proper message on the screen.

### **F2: Admin Login**

Admin can login using the login portal after signup. Once an admin successfully logs in, he/she will be redirected to the **Home Page** with a proper message on the screen.

### **F3: Add Members**

After entering proper details of members, when admin presses the button **Add Member** details will get added into the database.

**F4: See Total Members**

Admin can see total members present in the database.

**F5: Search Members**

Admin can search Members by their Name and Id.

**F6: Fee Payment**

Admin can add member fee payment by using their Member Id.

**F7: Check Fee Details**

Admin can check Member fee details by using their member Id.

**F8: Update Members**

Admin can update Members by using their member Id.

**F9: Delete Members**

Admin can delete Members by using their member id.

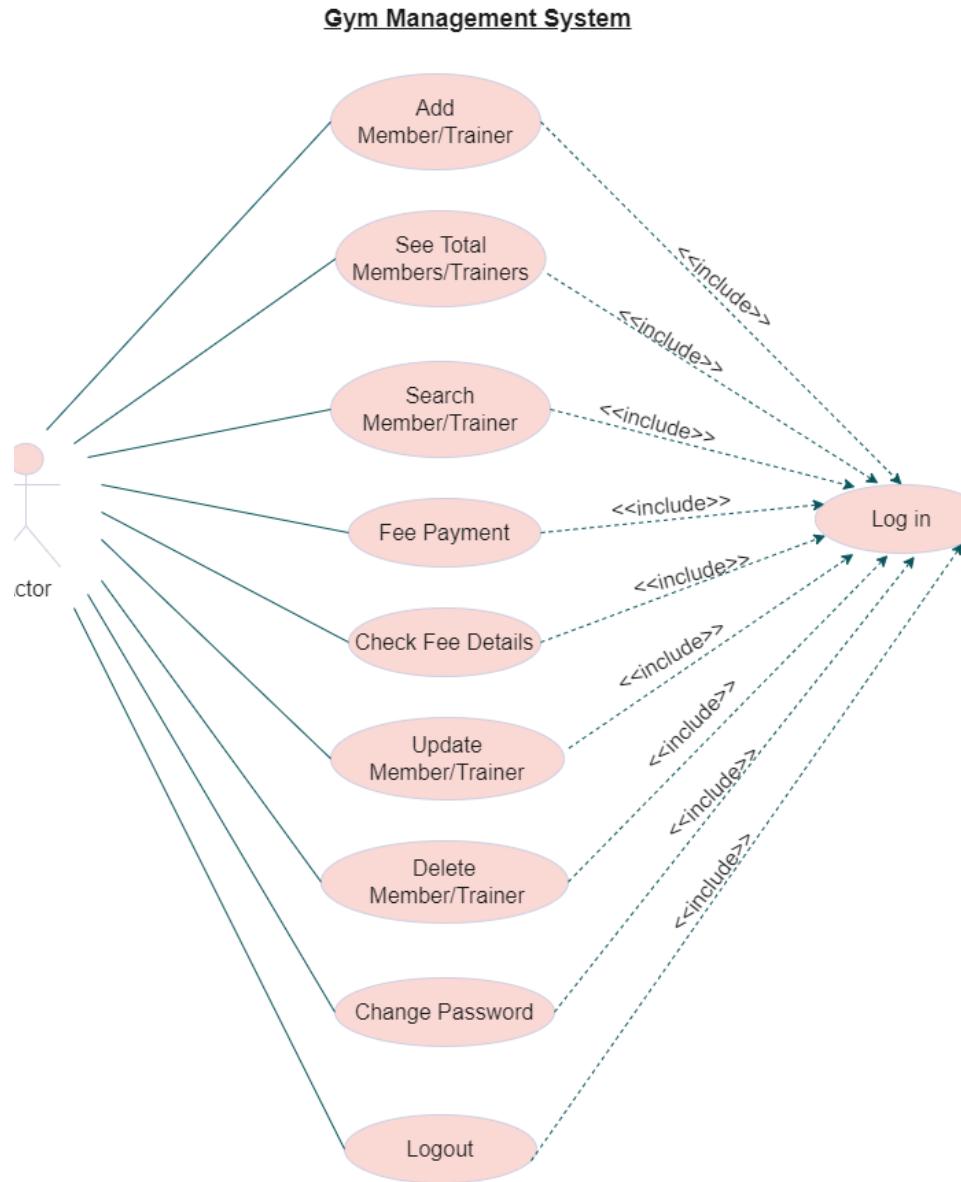
**F10: Change Password**

Admin can change Password if he wants to change.

**F11: Logout**

Admin can logout.

### 3.3 Use Case Model



#### 3.3.1 Use Case #1

**Purpose** - Admin can login to their respective account.

**Requirements Traceability – F2**

**Actors** – Human

### **3.3.2 Use Case #2**

**Purpose** - Admin can Add members to the system.

**Requirements Traceability** – F3

**Actors** – Human

### **3.3.3 Use Case #3**

**Purpose** - Admin can see list of total members.

**Requirements Traceability** – F4

**Actors** – Human

### **3.3.4 Use Case #4**

**Purpose** - Admin can search members from added members.

**Requirements Traceability** – F5

**Actors** – Human

### **3.3.5 Use Case #5**

**Purpose** - Admin can add member fee payment by using their Member Id.

**Requirements Traceability** – F6

**Actors** – Human

### **3.3.6 Use Case #6**

**Purpose** - Admin can check Member fee details by using their member Id.

**Requirements Traceability** – F7

**Actors** – Human

### **3.3.7 Use Case #7**

**Purpose** - Admin can update details of existing members.

**Requirements Traceability** – F8

**Actors** – Human

### **3.3.8 Use Case #8**

**Purpose** - Admin can delete Members.

**Requirements Traceability** – F9

**Actors** – Human

### **3.3.9 Use Case #9**

**Purpose** - Admin can change his/her password whenever he wants.

**Requirements Traceability** – F10

**Actors** – Human

### **3.3.8 Use Case #10**

**Purpose** - Admin can logout from the system..

**Requirements Traceability** – F11

**Actors** – Human

