

**207**

**Software Engineering Theory**

Gym System

**Final Project**

Full Name: Rania Ahmed ElDahrany

ID: 203045

Full Name: Fady Adel Saad

ID: 203089

Nov 2020

Contents

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope 3](#_Toc402452673)

[Definitions, acronyms, and abbreviations 3](#_Toc402452674)

[Requirements 3](#_Toc402452675)

[Functional Requirements 3](#_Toc402452676)

[Non Functional Requirements 4](#_Toc402452677)

[System Models 4](#_Toc402452678)

[Use Case Model 4](#_Toc402452679)

[Use Case Tables 5](#_Toc402452680)

# Introduction

## Software Purpose

## Our program provides memberships and courses to keep track of our members, staff and schedules. Although a wide variety of scheduling solutions is available.

## Software Scope

## This system aims to automate gym and fitness admission process as the admission process in gyms and selecting a trainer is difficult.

## This system will help to overcome such problems by booking the desired slot online and pay fees by electronic money transfer.

## Gym System helps the gym to boost annual with features like client scheduling, staff and member management, auto-renewing memberships.

## Scalable to meet the needs of a single or multiple location club, or a large franchise, Gym software meets all your gym business needs all online.

## Definitions, acronyms, and abbreviations

# The objective of the Gym System is to provide a system which handles the information of the people coming into the gym and maintaining their health care. It takes care of all their health information. It even maintains the data of what and all medicines used by the people who join the gym. Data will be stored in the database. It also maintains the people’s attendance, gym records.

# Requirements

## Functional Requirements

## Admin, members, and trainers can login using credentials.

## Admin can view the package details of an individual member.

## Admin can manage packages by adding cost, discount and deleting old packages.

## Admin can also view the member’s attendance taken by the trainer.

## Members can view their profile, list of trainers, the package and payment details.

## Members can give feedback on their trainers.

## A trainer can take member’s attendance daily.

## Non-Functional Requirements:

* **System should be easy to use.**
* **System should be available 24 hours.**
* **System should be able to work on all operating systems.**
* **System should response at the time.**
* **System should allow trainers to change their password in case of security.**
* **System should have high security against hacking.**
* **System should handle expected and non-expected errors in ways that prevent loss in information**

**and long downtime period.**

# System Models

## Use Case ModelDiagram Description automatically generated

## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0001 | |
| Use Case Name: | Login | |
| Actors: | Member, admin, and trainer | |
| Pre-conditions: | none | |
| Post-conditions: | * Access the system. | |
| Triggers: | * Allow users to access the system by entering valid username and password. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and Password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter incorrect username and Password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: First increment. * Frequency: Many times, daily. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection. * Cannot login if you are not one of the actors. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0002 | |
| Use Case Name: | Manage packages | |
| Actors: | Admin | |
| Pre-conditions: | Login | |
| Post-conditions: | * Add, remove, or edit the packages. | |
| Triggers: | * The admin can edit the packages | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| 3- User can select the manage page from the list. |  |
|  | 4- System will show all the packages |
| 5- User can add cost, make discounts, and delete old packages. |  |
|  |  | 6- System will save changes in the database. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter wrong username or password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: Second increment. * Frequency: Daily. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0003 | |
| Use Case Name: | Take member’s attendance. | |
| Actors: | Trainer | |
| Pre-conditions: | Login | |
| Post-conditions: | * Take the daily attendance of members that registered in his attendance table. | |
| Triggers: | * Allow the Trainer to take the attendance daily in the class. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and Password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| 3- User can select the attendance page from the list. |  |
|  | 4- System will show the attendance table. |
| 5- User can take member’s attendance. |  |
|  |  | 6- System will save changes in the database. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter wrong username or password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: Second increment. * Frequency: Daily. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0004 | |
| Use Case Name: | Give feedback | |
| Actors: | Members | |
| Pre-conditions: | Login | |
| Post-conditions: | * Increase or decrease in the trainers’ rating. | |
| Triggers: | * Members can give feedback on the trainers. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and Password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| 3-User can select the feedback page from the list. |  |
|  | 4- System will show the feedback page. |
| 5- User can give feedback and rating on trainers. |  |
|  |  | 6- System will save the feedback and the rating in the database. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter wrong username or Password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: Third increment. * Frequency: Monthly. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection. * Cannot change the rate or the feedback | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0005 | |
| Use Case Name: | View the attendance | |
| Actors: | Admin | |
| Pre-conditions: | Login | |
| Post-conditions: | * Check the Members’ attendance taken by the trainers. | |
| Triggers: | * Allow the Admin to view the user’s attendance taken by the trainer. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and Password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| 3- User can select the attendance table from the list. |  |
|  | 4- System will show the attendance table. |
| 5- User check the members’ attendance. |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter incorrect username or Password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: Third increment. * Frequency: Daily. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0006 | |
| Use Case Name: | View the packages’ details of members | |
| Actors: | Admin | |
| Pre-conditions: | Login | |
| Post-conditions: | * Check the members’ selected packages. | |
| Triggers: | * Admin can view the package details of an individual member. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and Password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| 3- User can select members’ packages from the list |  |
|  | 4- System will open members’ selected packages list. |
| 5- User can check the members’ selected packages |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter incorrect username or Password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: Second increment. * Frequency: Daily. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection. * Cannot change members’ selections. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 0007 | |
| Use Case Name: | View list of trainers, packages, and payment details | |
| Actors: | Member | |
| Pre-conditions: | Login | |
| Post-conditions: | * Member can choose his suitable trainer, package, and way of paying. | |
| Triggers: | * embers can view their profile and list of trainers. They can also view the package and payment details. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username and Password. |  |
|  | 2- System checks the validity of the login details if valid the system will open and show the home page. |
| 3- User can select his suitable trainer, package, and way of paying. |  |
|  | 4- System will save changes in the database. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter incorrect username and Password. |  |
|  | 2- Invalid login or Password.  3- System stops. |
| Includes: | * When to be available: Third increment. * Frequency: Daily. * Priority: Essential, must be implemented. | |
| Notes and Issues: | * Cannot login without internet connection. * Cannot make change in any package or list. | |