Hacettepe University

Department of Computer Engineering

BBM384 - Software Engineering Laboratory

Software Design Document

Group 12 - Tesla

Veli Can AYDIN - 21526662

Abdülsamet KALKAN - 21427033

Hakan İMAL - 21426993

Onur VARSAK - 21387485

Sefa Seyda ÖZDOĞAN - 21483421

Barkın ATASAY - 21590911

1. REVISION

Version	Date	Author	Change Description
1.0	25.04.2018	Whole group contributed	E-R diagram and sequence diagram added

2. INTRODUCTION

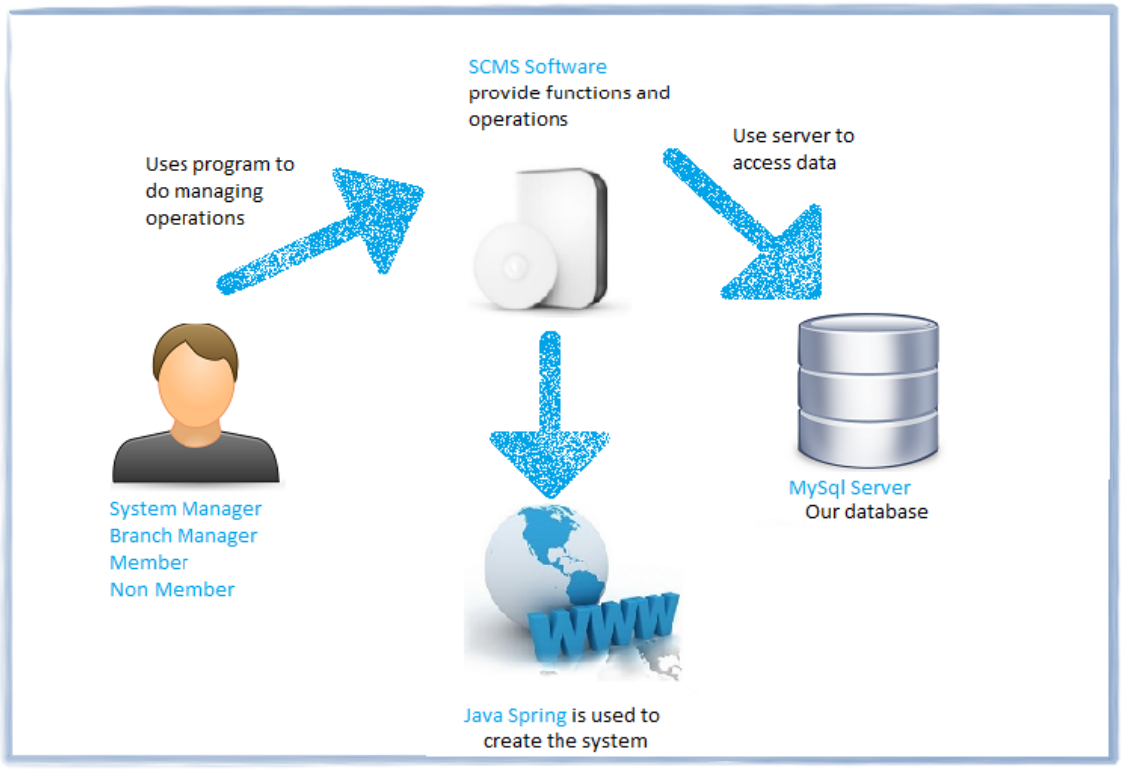
2.1. Purpose and Scope

Details of software design will be explained in this document. The aim is giving general information about the software's design. In audition this document include interfaces design at software component with their functionality.

2.2 Document Overview

In this document we give detailed information about our design limits and the way of how we design. In this document we provide UML diagrams which are sequence, class, E-R and state diagram.

2.3 System Overview



2.4 Definitions, Acronyms, and Abbreviations

Term / Acronym	Definition
SRS	System Requirements Specifications
SDD	Software Design Description
SQL	Structured Query Language
SCMS	Sports Center Management System

3. Design Constraints and Decisions

The Bene-fit Sports Centre Membership System uses internet on internet browser.
The system needs to be internet connected.
The Bene-fit Sports Centre Membership System being designed to easily add, update and delete features later on.
Our system being made as a internet application that can be used 7/24 in internet browser.
Our system depends on the MYSQL database system.
Our system will not be depend on ant specific hardware.
Java Spring framework, CSS, Javascript, MYSQL and MVC being used to develop our system.
Admin is the person who manages our system.
Other operation will be done after login for system users.
Admin and all users must be register in the our system
The system will respond to the all users less than three seconds depend on internet speed and servers speed.
Member and user will be limited considering the performance of the server and database.

4. Design Details

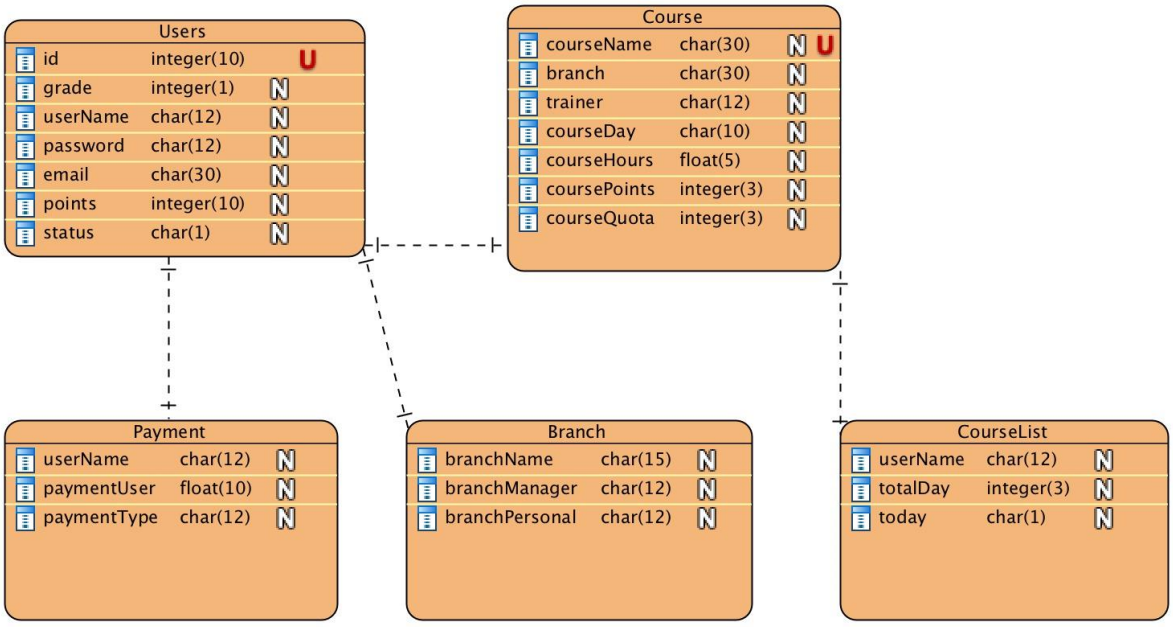
4.1 Software Components

This section is referred in the “SDD - Class Diagram” document.

4.2 Software Behavior

This section is referred in the “SDD - Sequence Diagrams” document.

4.3 Data Model (E-R Diagram)



4.4 User Interface Design

4.4.1 Changes in UI

4.4.2 Explanation of Graphical User Interfaces

A graphical user interface (GUI) is an interface through which a user interacts with a computer, a smartphone, and other electronic devices. This interface establishes a link between the code written by the programmer and the end user. This interface uses icons, menus, and other visual indicator (graphical) notations. This section gives information about the interface of our system.

The interface of the system was developed on Html and Css. It was supported by Bootstrap, an open source JavaScript framework. Moreover, it was developed using Jquery libraries.

When the user visits the site to use the system, they will first see the login screen. If the user is a member of the system, they will be able to log in easily through this screen. On the login screen, the user's email address and the user's password are requested. When the user write the user information correctly, the users logged on to the system. If the user is not a member of the system, click on the link to register on the same page.

When the user enters the registration page to register with the system, some information is required from the user. This information is the user's full name, email and password. The user must enter a unique email address to register with the system. It must also agree to the terms

for registration. After fulfilling these conditions, click on the register button and register on the system.

The user will have an interface that can monitor and manage their own information after login to the system. Since each user in the system has different authorities, the interface may change. Because the authority of a normal user and system administrator is different.

The user opens the Sign Course page when he / she wants to register a course. This page lists the courses that the user can register. The user selects a desired course and clicks on the register button. If the course capacity is available, the user registers the course.

The coach opens the attendance list page when he / she wants to take attendance. The coach selects the course and date for which he / she will take attendance. Then click on the next button and the coach will be directed to the attendance list page. The attendance list page contains the users and user IDs registered in the course. The coach enters each user's attendance into the checkbox. Notification is given on the status column according to the attendance status of the users. The coach clicks on the confirm button after finishing the operation and records saves to the system.

The system admin opens the define course page when he/she want to open a course. The admin writes the name of the course to be defined on this page to the text-box. Then click on the define course button and the course is defined to the system.

The system admin opens the define payment page when he/she want to define a payment method. The admin writes the name of the payment method to be defined on this page to the text-box. Then click on the define method button and the payment method is defined to the system.

The system admin opens the activate members page when he/she want to observe and manage the status of users in the system. This page contains user information consisting of six columns. The information in the column contains the following: member no, name, surname, paying, delete user, user status. Six users are listed on one page. There are page numbers under the page to view more users.

The system admin opens the manipulate customer / trainer / employee page when he/she want to observe and manage the status of users in the system. This page contains users (customer / trainer / employee) information. All users are listed in the interface and the system selects the user that admin wants to manage. After the manipulating process is finished, the system admin saves the page.

Users enter the evaluate course page to evaluate the courses they have already taken. In this page, the questions directed to the users and the level of participation in these questions are measured. There are radio buttons under the questions, users are expected to choose one of

them. Finally, the opinions of users are expected in text-box. The user clicks the send button to finally save the entered information.

Users enter the evaluate trainer page to evaluate the trainer they have already learned. In this page, the questions directed to the users and the level of participation in these questions are measured. There are radio buttons under the questions, users are expected to choose one of them. Finally, the opinions of users are expected about trainer in text-box. The user clicks the send button to finally save the entered trainer evaluations.

Users enter this page to request a course from the management. This page requires two pieces of information from the user. The first is the name of the course the user wants to open. The second is a detailed information box about the course that the user wants to open. When the user enters this information completely, the user clicks the send request button and saves it to the system.

In the system there is a message page between the trainer and the user. The user enters this page when he wants to send a message to the trainer, or when the trainer wants to send a message to a user. User / Trainer selects the Trainer / User that they want to send a message each other. The message text to be sent to the message box is written. Finally, the submit button is clicked and the message is forwarded to the other side.

4.4.3 Error Prevention and Operation Messages

This section is referred in the “SDD - Error and Success Dialogs” document.

4.4.4 Design Considerations

UI design is to make the user’s interaction as simple and efficient as possible. The interface we use in our system is completely user-friendly. Guarantee the integrity and image of the UI is maintained, irrespective of what devices or viewports are used with responsive designs. Defined the layout and composition of the UI, extending to highly detailed specifications for margin sizes, typography, image and colour guides.

5. Requirements Traceability

This section is referred in the "SDD - Requirements Traceability Matrix” document.

6. Appendices

- SDD - Requirements Traceability Matrix
- SDD - Class Diagram
- SDD - Sequence Diagrams
- SDD - Error and Success Dialogs