Software Requirement Specification

For

Project

<CCBFS>

Version 1.0 approved

Prepared and Developed by Team A-1

For: Country Club

As of 11 August 2017

Table of Contents

1. Introduction
   1. Purpose
   2. Intended Audience
   3. Project Scope
   4. References
2. Overall Description
   1. Product Features
   2. User Classes and Characteristics
   3. Operating Environment
   4. Design and Implementation Constraint
   5. User Documentation
   6. Assumptions and Dependencies
3. System Features
   1. Functional Requirements
   2. Non-Functional Requirements
4. Additional Appendixes
   1. Appendix A
   2. Appendix B
   3. Appendix C
5. **Introduction**
   1. Purpose

The purpose of this software product is to allow club members to make bookings online. It also helps the manager to manage the country club in a more organized and orderly manner.

The Purpose of this document is to entail all the details that the user is left unambiguous about what is and isn’t being developed. A common agreement.

* 1. Intended Audience and Reading Suggestions

This is intended for country club members and the manager who will be using our software product to reduce communication overhead. This allows our clients to have a shorter and pleasant booking experience. This document intends to inform the users of the purpose and scope of this particular ‘CCFBS’ software product.

* 1. Project Scope

The scope of this product is to enable country club members to view the availability and prices of the facilities as well as making and changing their bookings. It also allows members to check their membership tier and the discounts entitled to them.

They can also update their personal particulars and store their booking preferences as well as upgrading their membership.   
  
It also allows the club manager to create, delete, amend and update the club’s facilities for booking and set the various charging rates.

The software also will be able to generate a report which graphically displays the popularity of facilities and the frequency of a facility is in used and in popularity .

1. **Overall Description**

2.1 Product Perspective

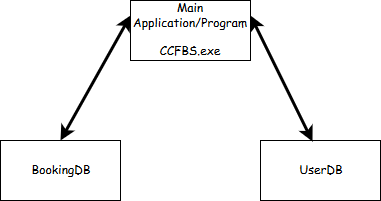
This is a new, self-contained product. The system will be in 3 main parts, the Application/Program(main executable) , the User Database , and the Booking Database.

The program is where any User interactions will take place. Any Interactions with User Database and Booking Database shall be covered by the program.

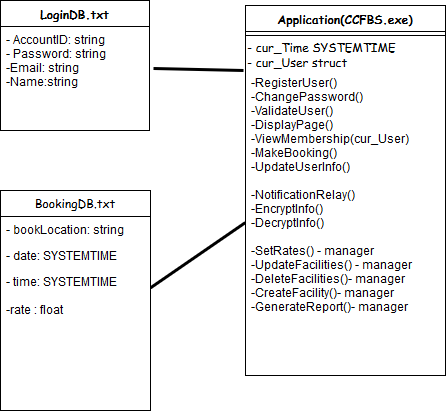
User Database will contain encrypted personal information and User data related to the program shall be kept here.

Booking Database will contain encrypted Booking information and relevant user data shall be kept here.

Component diagram of CCFBS



2.2 Product Features



2.3 User Classes and Characteristics



General Use Case 1.0 Typical operations of what a member and what a manager can do.

-Users of the system should be able to register if they have not been registered in the system.

-Users should be able to authenticate themselves via their saved entries in UserDB.

Once validated, Users can update and view their profile information or search for facilities to book within their privileges and book within available bookings left.

Managers would be able to view current booking trends within the facilities and change any existing reservations, delete and create new facilities as well as change the current rates.

2.4 Operating Environment

To Be written in C++.

Rationale: Extensibility and maintainability

To be run using Windows Operating System.

Rationale: Best supported and user-friendliness.

2.5 Design and Implementation Constraints

The software product will be written entirely in C++.

2.6 Assumptions and Dependencies

The software product will run on Windows.

1. System Features
   1. Functional Requirements

3.1.1 User Class 1 – Club member

3.1.1.1 Functional requirement 1.1

FR001:

TITLE: Login to the facility booking system

DESC: Given that the club member has an account in the country club database, he/she should be able to login to the facility booking system. The club member then will be able to view available facilities for booking.

3.1.1.2 Functional requirement 1.2

FR002:

TITLE: Retrieve password

DESC: Given that the club member has registered, the club member should be able to retrieve his/her password via email.

3.1.1.3 Functional requirement 1.3

FR003:

TITLE: Club member registration

DESC: Given that the club member has not registered an account with the country club, the club member should be able to register through the software application. The club member must provide a username, password and email address.

3.1.1.4 Functional requirement 1.4

FR004:

TITLE: Update account information

DESC: Given that the club member has an account in the country club database, and has logged into the system, he/she should be able to modify certain existing information pertaining to his/her account details via the software application.

3.1.1.5 Functional requirement 1.5

FR005:

TITLE: Search available booking facilities (should confirm what can be searched/sorted by – better to include in SRS)

DESC: Given that the club member has an account and has logged on to the system, the club member should be allowed to search for available booking facilities via the software application.

3.1.2 User Class 2 – Club manager

3.1.2.1 Functional requirement 2.1

FR006:

TITLE: Administrator login

DESC: Given the administrator account has been created, the club manager should able to log on to the country club booking system after entering the correct login credentials.

3.1.2.2 Functional requirement 2.2

FR007:

TITLE: Manage country club facilities

DESC: In order to have a list of facilities to book, an administrator should be able to manage the facilities.

**Scenario: Add a new facility**

Given the administrator is logged in, when the administrator creates a new facility, the new facility should be added to the list of facilities.

**Scenario: Update an existing facility**

Given the administrator is logged in, when the administrator edits an existing facility, the facility should be updated in the list of facilities.

**Scenario: Delete a facility**

Given the administrator is logged in, when the administrator deletes a facility, the deleted facility should be removed from the list of facilities.

**Scenario: Set charging rate**

Given the administrator is logged in, when the administrator sets the charging rate for the facility, the recent charging rate should be updated in the list of facilities.

* 1. Non Functional Requirements

NFR001:

TITLE:

DESC:

NFR002:

TITLE:

DESC:

NFR003:

TITLE:

DESC:

NFR004:

TITLE:

DESC:

NFR005:

TITLE:

DESC:

NFR006:

TITLE:

DESC: