Software Requirements Specification

Project Repository: https://github.com/youmengh/software-dev-capstone

TABLE OF CONTENTS

Section 1. Introduction

1.1 Purpose

This document shall identify and explain the software specifications of the Tennis Club Reservation System as determined by the needs of the client. The Software Requirements Specification (SRS) documentation framework is designed to break down each component of the software product from a technical and analytical perspective, such as an overview of how the software will work once completed, and what features are necessary to meet client expectations, based on the Institute of Electrical and Electronics Engineers (IEEE) standard 830. The document will also determine any potential software requirements necessary to ensure an efficient software development environment, including any concerns or limitations. This document has been written to ensure transparency and organization of the entire software development process for both the development team and any interested client or shareholder.

1.2 Scope

The software product is the Tennis Club Reservation System (TCRS). The end product is a web application capable of managing membership, reservations, and payments handled by the tennis club, as well as back end site and account management to be handled by system administrators, and a public directory of consenting member information. All specific client needs shall be taken from the Product Requirements Document (PRD). The software is intended to create ease of access for members, as a place to sign up, pay membership dues, view club information, and reserve courts for tennis, all from a single, centralized web application.

1.3 Definitions, Acronyms, and Abbreviations

Term/Acronym	Definition
SRS	**System Requirements Specification**
IEEE	**Institute of Electrical and Electronics Engineers**
PRD	**Product Requirements Document**
TCRS	**Tennis Club Reservation System**

1.4 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.*

IEEE Computer Society, 1998.

1.5 Overview

The remaining body of this SRS document shall include:

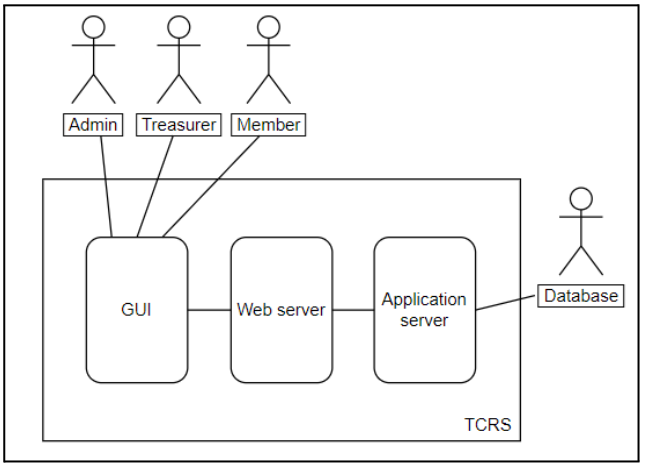
Section 2. A general description of the software and how it solves the problems and requirements of the overall product and client specifications. An overview of potential user characteristics, i.e. the targeted demographic of the software, and therefore the level of technical complexity or ease of access intended. Any limitations, either purposeful constraints or unintended challenges, that the development team will need to consider in the design process, including client guidelines or policies, or limits on access to hardware and software.

Section 3. A more specific technical analysis of the software development strategies and goals necessary to ensure cohesive design and testing aligning with product goals, including performance and function specifications, and any constraints on software functionality. The functional attributes of the product such as security and access control systems, and ease of maintenance for any potential future updates shall be listed. Any additional unspecified relevant information shall also be delineated in this section.

Section 2. General Description

2.1 Product Perspective

The TCRS software is an independent and self contained product. This software shall be accessed on a browser of a computer system with internet access. There are three primary actors with various privileges using this software. The database is a secondary actor.



2.2 Product Functions

The general product functions can be divided into the following subsystems:

a) **Back End** - A back end system including a database of logins at various levels, including standard members and administrators, who will have scaled privileges depending on role.
b) **Reservations** - A system designed to keep track of reservations of the club's tennis courts.
c) **Membership** - A system designed for signing up for membership and for existing members to log in to their accounts.

i) **Guest Information** - A subsystem of membership designed to keep track of the guest information registered by members.

d) **Finance -** A system designed to facilitate any payments or notifications regarding payments to users.

2.3 User Characteristics

Users will be primarily members of the tennis club, but there will be several higher-privileged tennis club employees using this software too. Users may vary between technologically skilled or technologically illiterate, with the average user likely to have general competence with a computer. In order to accommodate the majority of the user base, several components such as account management, site navigation, and UI/UX should be easy to follow and understand.

2.4 General Constraints

Security measures for financial and personal data of members must be in compliance with regulations. Reliability requirements include role-based restrictions for user logins, upholding the maximum member limit upon application, and the ability to handle failed payments. Hardware does not cause any constraints due to the small amount of data being used for the software.

2.5 Assumptions and Dependencies

The following are assumed: users will have internet access and basic competence with technology; server hardware will be able to support a small database of ~1000 members; applicants will select the appropriate membership price according to their age.

Section 3. Requirements

3.1 Functional Requirements

3.1.1 Log in

3.1.1.1 Introduction

This function shall allow users to utilize the software for its intended purpose. This function shall allow the user to view their personal/financial information, manage reservations, and manage their account/membership. Features available shall be role-based, determined by the privileges associated with the submitted credentials.

3.1.1.2 Inputs

This function shall involve three user inputs. The 'Username' input is a string with 8-20 characters, intended to be the user's unique username for their account. The 'Password' input is a string with 8-20 characters, intended to be the user's password associated with their username. The 'Submit' input is a button that confirms the user has entered their credentials accurately and sends the two string inputs to be processed.

3.1.1.3 Processing

The software shall compare the string input credentials to database records, searching for membership accounts associated with the same credentials as entered.

3.1.1.4 Outputs

The software shall output access or rejection to the user according to the result of the database search. If the input credentials match those of a membership in the database, the user will be granted role-based access to the software's features. If the input credentials do not match any memberships in the database, the user remains on the log-in page and is prompted to re-enter log-in credentials.

3.1.2 Register

3.1.2.1 Introduction

This function shall allow users to apply for membership within the club. This function shall provide users with the ability to submit a sign-up form to be processed by the club's management.

3.1.2.2 Inputs

The 'Register' input is a button the user can click to initiate the registration process. The 'Name', 'Address', 'Phone', and 'Email' inputs are strings, intended to be the user's name, address, phone number, and email, respectively. The 'Age' input is a number value intended to be the user's age. The 'credit card number', 'expiration date', 'name', and 'CVV' inputs are optional in case the user pays by check; these inputs are intended to be the user's payment information. The 'submit' input is a button that submits the signup form to club management.

3.1.2.3 Processing

The input information shall be forwarded to club management for review. The software shall count the active members to determine if the applicant shall be added to a waitlist.

3.1.2.4 Outputs

The software shall output information to the user, which will disclose whether they shall be placed on the waitlist or if their application is being reviewed immediately.

3.1.3 Reserve

3.1.3.1 Introduction

This function shall allow the user to reserve a court for a 90 minute period. The user shall be able to see available time slots for reservation and select their choice. The user shall indicate if they are playing singles or doubles matches, and enter the names of all other players. The user shall pay a ten dollar fee for each non-member guest they are playing with.

3.1.3.2 Inputs

The 'Reserve' input is a button shown under each available time slot on the calendar, and clicking it allows the user to initiate a reservation for the indicated time slot. The 'Cancel' input is a button that is shown after the user has initiated the reservation process, and clicking the button shall abort the process of reserving the time slot. The 'Game type' input

is a drop-down menu with options that indicate what type of tennis game the user will be playing. The 'Player Names' inputs are strings for the name of each player the user shall be playing with. The 'Submit' input is a button the user will press when the reservation information form is complete.

3.1.3.3 Processing

The software shall check if the time slot is still available and will update the public calendar and display the selected time slot to be unavailable upon 'Submit' input. The 'Player Names' inputs shall be compared to all member's names in the database to determine if the user shall be billed guest fee(s) for bringing non-members.

3.1.3.4 Outputs

The software shall output a confirmation message to the user if the reservation went through. The software shall output an error message to the user if another user swiftly reserved the same time slot. The output shall also inform the user of a bill if any 'Player Names' inputs were determined to be non-members.

3.1.4 Cancel Active Reservation

3.1.4.1 Introduction

This function shall allow the user to cancel a court reservation that the user has previously reserved.

3.1.4.2 Inputs

The 'Cancel Reservation' input is a button shown to a user managing an active reservation under their name, and pressing it indicates that the user would like to cancel the reservation.

3.1.4.3 Processing

The software shall remove the time slot associated with the input button from the active reservations.

3.1.3.4 Outputs

The software shall output a confirmation message to the user that their previous reservation is now canceled. The time slot shall be made available for reservation on the calendar.

3.1.5 Cancel Membership

3.1.5.1 Introduction

This function shall allow the user to end their membership with the tennis club. The user shall have the option to cancel their membership when viewing/managing their account. The user shall be asked to provide reasoning for cancellation, and a warning screen with all important information about the cancellation shall be shown to the user, followed by multiple confirmation buttons.

3.1.5.2 Inputs

The 'Cancel' input is a button shown to the user when viewing/managing their account.

Pressing the button shall allow the user to initiate a membership cancellation process. The 'Reason' input is a form where the user shall describe their reason for leaving the tennis club. The 'Submit' input is a button the user shall press to submit their notice of cancellation. The 'Confirm' input is a button the user must press as the last layer of confirmation that they would like to cancel their membership.

3.1.5.3 Processing

The software shall check if the user's attempt to initiate their cancellation occurred after the cancellation deadline of 1 October. The software shall save the user's 'Reason' input and notify the database of cancellation after 'confirm' input.

3.1.5.4 Outputs

If the 'Cancel' input occurred after the deadline, the software shall output a message to the user indicating they have missed the deadline to cancel for the upcoming calendar year and the window to resubmit a cancellation during the upcoming calendar year is 1 January - 1 October. Otherwise, the software shall output a form for the user to indicate their reason for cancellation, as well as an informational summary displaying all necessary information about their cancellation. Upon 'Submit' input, the software shall output a final warning message to the user, asking if they shall confirm the cancellation. Upon 'Confirm' input, the software shall output a success message to the user.

3.2 External Interface Requirements

3.2.1 User Interfaces

The primary user interface will be a simple menu page with a bar of header buttons leading to each separate component of the application, including a reservations, membership/account management, and payment tab. A separate button will handle logging in or out. Any errors will be displayed to the user in simple, short error boxes.

3.2.2 Hardware Interfaces

A computer's display system shall allow the user to view the software GUI on their web browser. A keyboard and mouse shall allow the user to interact with the software.

3.2.3 Software Interfaces

An operating system with a web browser is the required user software to access this product. All software features are accessible from a web browser.

3.2.4 Communications Interfaces

HTTP (Hypertext Transfer Protocol) shall be used during communication between the user's web browser and the web server. This shall allow the browser to display the software's GUI and appropriate information from the database.

3.3 Performance Requirements

The software shall support over 1000 simultaneous users. The total number of users shall be never more than 1000 members and some certain staff members. The total number is low enough to allow for reliability in the worst-case scenario of every user simultaneously active. The software shall

support efficient database interaction with information of over 1000 customers.

The software's performance speed on the user's system is dependent on the strength of the user's internet connection. With optimal network connection strength, all transactions shall be processed in a user-friendly duration of less than 2 seconds.

3.4 Design Constraints

3.4.1 Standards Compliance

The TCRS software shall comply with any regulations regarding protection of customer financial data and personal information. Reports shall be formatted in adherence with the IEEE standard 830 model.

3.4.2 Hardware Limitations

The hardware environment of the software is a user's computing system with keyboard and/or mouse input sources and a display system output source. Smartphones/tablets are a possible hardware environment, so the software must not be dependent on a computer mouse input.

3.5 Attributes

3.5.1 Availability

The software shall run on a reliable and popularly used web server to ensure high availability and a high-performance connection between the user and the application server.

3.5.2 Security

User passwords shall be encrypted and unable to be manually accessed by a user. Admin access will be logged to monitor any potential changes. All payment information will be inaccessible to any user, including admins. Security designs will be implemented as early on as possible within foundational code structures and systems to ensure effectiveness. A cryptographic cipher scheme will be chosen to encrypt user passwords and payment information within their respective databases.

3.5.3 Maintainability

The software shall be understandable to developers by having clearly documented code and using modern programming frameworks and languages. The latest version of the widely and commonly used web framework Django shall be used to develop the application. Back end code shall be written in the latest version of the popular programming language Python. The latest version of SQL, one of the most well-known database languages, shall be used to make transactions with the database. The popularity, relevance, and modernity of these frameworks and languages shall aid the software's extensibility. The software shall not rely on third-party API's or script hosts, also aiding its extensibility.

3.6 Other Requirements

Software shall consist of different operations from normal operations (i.e. user mode) to special operations (i.e. employee mode, finance mode, admin mode). Software should generally be live and functional 24/7. It must be online and functional during normal operating hours for users to make reservations and perform any desired tasks or transactions. During off-time, software shall remain live, but may be shut down for maintenance purposes.