SOFTWARE REQUIREMENT SPECIFICATION

JANUARY 28, 2025

ACTIVEZONE

Prepared by: ASWIN C S
AL AMEEN N
ANANYA KRISHNAN
ARATHY R

Submitted in partial fulfillment
Of the requirements of

CSD334 MINIPROJECT

TABLE OF CONTENTS

1. Introduction

1.1 Purpose

The purpose of ACTIVEZONE is to provide an efficient platform for users to book turfs seamlessly. The system will enable users to search for available turfs, make bookings, and manage reservations. It also offers turf managers a dashboard to oversee and update turf availability.

1.2 Overview

The SRS is organized in such a way:

- **Section 1- Introduction** This gives an introduction of the project.
- **Section 2-Overall Description** This provides the user needs,assumption and dependencies.
- **Section 3-Functional Requirements** This gives the module description of the project.
- **Section 4-External Interface Requirement** This gives the interface of various interfaces like the User,Hardware,Software and Communicattion.
- **Section 5-Non Functional Requirements** This gives the requirements like Performance,Security and Software Quality.

1.3 Independent Audience and Use

This software is intended for:

- **End Users:** Individuals looking to book turfs for various sports activities.
- **Turf Managers:** Managing turf availability, bookings, and schedules.

1.4 Scope

The project focuses on creating a turf booking platform with the following features:

- User authentication and authorization.
- Searching for available turfs.
- Booking and managing turf reservations.
- Turf availability management by turf managers.
- Mobile-friendly, responsive interface.

1.5 References

- Django Documentation: https://docs.djangoproject.com
- PostgreSQL Documentation: https://www.postgresql.org
- Bootstrap Documentation: https://getbootstrap.com

2. Overall Description

2.1 **User Needs**

Users need an intuitive and fast platform to:

- Search for turfs by date, time, and location.
- Book and manage reservations without hassle.
- Receive booking confirmations and updates.

Turf managers need a tool to:

- Update and manage turf availability in real-time.
- Monitor and approve/cancel bookings.

2.2 **Assumptions and Dependencies**

- The application will be hosted locally during development, with future deployment on cloud servers.
- PostgreSQL will be the database system, and sensitive data will be securely stored.
- Users will have internet access to use the system.
- Email notifications depend on SMTP configuration.

3. Functional Requirements

3.1 **Module Descriptions**

- **User Module:**
    - **Registration:** Users can sign up with their email,phone number and password.Verification through email or OTP can be implemented for security.
    - **Authentication:** Secure log in and log out functionality.
    - **Profile management:** Users can update personal details ,view past bookings and manage their preferences.
    - **Booking history:** Displays past and upcoming bookings,along with payment details if applicable.
- **Search Module:**
    - **Turf Filtering:** Users can search for available turfs based on location,date and time.
    - **Live Availability Updates:** Turf managers can update real-time availability,ensuring accurate information for users.
    - **Turf Details Page:** Each turf has a dedicated page displaying images,location,pricing and available slots

- **Booking Module:**
    - **Slot Selection:** Users can choose available time slots for booking.After selecting a slot,the user receives a confirmation email and a unique booking ID.
    - **Modification and Cancellation:** Users can reschedule or cancel bookings based on turf policies.Secure online payment options are provided.
- **Turf Management Module:**
    - **Turf Availability Control:** Turf managers can update available slots and maintenance schedules.
    - **Booking Approval System:** Managers can approve or reject bookings based on demand.
- **Admin Module:**
    - **User Management:** Admins can view,edit,or deactivate user accounts if necessary.
    - **System Logs and Monitoring:** Admins can review system logs for debugging and security monitoring.
    - **Reports and Analytics:** Generate reports on user engagement,booking trends,and revenue insights.
    - **Dispute Handling:** A dedicated module to manage customer complaints and refund requests.

4. External Interface Requirements

4.1 **User Interface**

- The user interface will be designed using HTML, CSS, JavaScript.
- Features include:
    - A home page displaying available turfs.
    - User-friendly forms for registration and login.
    - Dashboard for users to view/manage bookings.
    - Responsive design for mobile and desktop.

4.2 **Hardware Interface**

- The system requires a standard computer or smartphone with internet access.
- Recommended hardware:
    - CPU: 2.0 GHz or higher.
    - RAM: 4 GB or higher.
    - Storage: 100 MB for client-side cache.

4.3 **Software Interface**

- **OS:** Windows 7 or above
- **Front End:** HTML, CSS, JavaScript
- **Back End:** Django (Python)

- **Database:** PostgreSQL
- **Server:** Django Development Server(for local testing)

4.4 **Communication Interface**

- HTTP/HTTPS protocols for client-server communication.
- SMTP for email notifications (e.g., booking confirmations).

5. Non-Functional Requirements

5.1 **Performance Requirements**

- The system should support up to 100 concurrent users without noticeable performance degradation.
- Page load times must not exceed 2 seconds under normal usage.

5.2 **Safety Requirements**

- Data backups will be performed daily to prevent loss.
- Error handling mechanisms will ensure that users are not stuck during unexpected failures.

5.3 **Security Requirements**

- All passwords will be hashed and stored securely.
- Sensitive data will be encrypted.
- Only authorized users can access protected sections.

5.4 **Software Quality Requirements**

- **Maintainability:** The code will follow clean coding standards and include comments.
- **Scalability:** The system will be designed to add features like payment gateways in the future.
- **Usability:** The application will provide a seamless and intuitive user experience.