
Software Requirements Specification for SPORTS

Version 1.0 approved

Prepared by HaiFeng Huang,

Wan Wang,

Prajwal Umesha,

Poorna Bengaluru Shivaji Rao

CSCI 5801, Fall 2024

09/23/2024

● Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	3
3.1 User Interfaces	3
3.2 Hardware Interfaces	3
3.3 Software Interfaces	3
3.4 Communications Interfaces	3
4. System Features	4
4.1 System Feature 1	4
4.2 System Feature 2 (and so on)	4
5. Other Nonfunctional Requirements	4
5.1 Performance Requirements	4
5.2 Safety Requirements	5
5.3 Security Requirements	5
5.4 Software Quality Attributes	5
5.5 Business Rules	5
6. Other Requirements	5
Appendix A: Glossary	5
Appendix B: Analysis Models	5
Appendix C: To Be Determined List	6

Revision History

Name	Date	Reason For Changes	Version
HaiFeng Huang Wan Wang,	09/23/2024	First version of for user requirements SPORTS	Version 1.0
Prajwal Umesha Poorna B S Rao	10/07/2024	Revised version	Version 1.1

1. Introduction

1.1 Purpose

System for production of recreational team scheduling (**SPORTS**) is a lightweight system that is developed for small-scale sports leagues. The system is designed to help manage various sports events, including venue scheduling, team registration, and match scheduling. This document covers the main functions and features of the SPORTS system, focusing on the core system parts that meet user needs. This version is the initial release of the SPORTS system, which is suitable for managing regional and local sports events.

1.2 Document Conventions

The following conventions are used in this SRS document:

- All “shall” statements indicate mandatory requirements, “should” statements indicate recommended but not mandatory requirements, and “may” statements indicate optional requirements.
- Specific terms and key nouns in the text are emphasized by being displayed in bold.
- Use cases are represented using standard UML (Unified Modeling Language) symbols and diagrams to help visualize system interactions.

1.3 Intended Audience and Reading Suggestions

This SRS is intended for the following audiences:

- Developers: This document provides the development team with the detailed requirements needed to build the SPORTS system.
- Project managers: For project planning and management to ensure that the project is implemented as required.
- Testers: Used as the basis for system testing to ensure that each function of the system is implemented as specified.
- Users and venue management: For an understanding of the system functions and possible future improvements.
- Stakeholders: Stakeholders will interact with the system in various functional modules of the system. (Stakeholders include: Stadium owners, Players, team members, city officials, league officials)

The document structure is organized according to the priority and logical order of the requirements. It is recommended to read the overall system description in Part 2 first, and then read the specific functional requirements according to the role requirements.

1.4 Product Scope

SPORTS is an event management system designed for local and regional sports leagues to simplify the organization, scheduling, payment, and statistical analysis of events. The system supports the management of multiple sports and multiple venues, and provides customized scheduling, registration and payment processing, weather monitoring, and compliance checks with city regulations. The goal of the SPORTS system is to reduce the complexity of event management and improve the efficiency of event organization, while providing flexible scheduling and analysis functions.

1.5 References

Reference

2024Fall-CSCI5801-Assignment1RequirementsDeliverable.pdf
Unified Modeling Language Specification, Version 2.5.1
IEEE Std 830-1998
Minneapolis Parks & Recreation Department, 2023 Edition.

2. Overall Description

2.1 Product Perspective

The SPORTS system was developed against the background of the complex logistics planning needs of professional sports leagues, which encompass the coordination of players, team staff and venue operations. Even at the venue level, these events require specialized software to coordinate ticketing, scheduling and security. However, existing enterprise-level software is often prohibitively expensive for local and regional sports organizations. In addition, enterprise-level software offers a plethora of features that are beyond the needs of smaller events, such as customization for individual teams and venues.

The SPORTS system aims to fill this gap by providing a generic solution that helps small sports leagues coordinate their participating teams, available venues, and necessary staff. The system will streamline the end-to-end processes of a sports league, from team registration and scheduling to season results aggregation. The SPORTS system provides users with a streamlined logistics management process.

The SPORTS system is a standalone product that does not rely on any existing system or product family. It focuses on the management of small-scale sports leagues and, although it can be extended and integrated with other systems, its design goal is to solve the core management needs of small-scale events. The system mainly interacts with the user's input/output devices and payment gateways.

2.2 Product Functions

Here is only a high level summary for product functions, details will be provided in Section 3

- Determining sports rules and team restrictions: Depending on the sport and league, the system will manage the relevant game rules and team participation restrictions.
- User registration: The system allows users to register for one or more sports and handles registration information.
- Payment management: The system requires users to pay for registered leagues and provides support for the corresponding payment gateway.
- League and game management: The system will manage and display the schedule, game records, and related venue concessions (such as food and beverage sales) for multiple leagues.
- Venue scheduling and conflict detection: Ensures that the schedule does not generate time or venue conflicts when scheduling venues.
- Game records and data analysis: The system will record the results of the game and generate relevant statistics and analysis reports to help users view the overall performance of the league.
- Weather and city regulations monitoring: The system will monitor real-time weather conditions and city regulations to ensure that the event goes smoothly.

[uml class image placeholder]

TODO: draw a system UML graph in here

2.3 User Classes and Characteristics

The primary user classes anticipated for the SPORTS system include:

League Managers

- **Frequency of Use:** High – frequent interaction, especially during league setup and throughout the season for management and scheduling.
- **Technical Expertise:** Moderate – basic computer literacy is required. Familiarity with managing league structures and schedules is necessary.
- **Security or Privilege Levels:** High – league managers will have administrative access to create, modify, and delete leagues, register teams, and manage payments.
- **Key Characteristics:** Responsible for overseeing multiple leagues, setting up schedules, ensuring that games do not conflict, and maintaining league rules and regulations. They will have access to analytics and reporting features.
- **Importance:** Critical – the system must meet their needs as they are the primary administrators of the platform.

Team Managers

- **Frequency of Use:** Medium – moderate interaction to manage their team's registration, payments, and scheduling.
- **Technical Expertise:** Low to Moderate – basic computer skills are needed to navigate registration, schedule games, and view league standings.
- **Security or Privilege Levels:** Medium – team managers will have limited administrative access to manage their own team's details, registration, and payments, but will not have control over the entire league.
- **Key Characteristics:** Responsible for registering teams, making payments, viewing game schedules, and managing the team roster.
- **Importance:** Important – essential to ensure a smooth team registration and management experience.

Venue Owners

- **Frequency of Use:** Low to Medium – occasional interaction to manage venue availability and conflicts.
- **Technical Expertise:** Low – basic ability to use the system for viewing schedules and confirming venue bookings.
- **Security or Privilege Levels:** Medium – access to venue management features to block out unavailable times and prevent scheduling conflicts.
- **Key Characteristics:** Responsible for managing the availability of venues, ensuring that games do not conflict, and communicating any venue-specific constraints or regulations.
- **Importance:** Medium – necessary for successful venue management, but interaction is less frequent.

Players

- **Frequency of Use:** Low – occasional interaction to view schedules, standings, and team information.

- **Technical Expertise:** Low – minimal technical expertise required. Most interactions will involve logging in to check game times or team standings.
- **Security or Privilege Levels:** Low – players will have read-only access to schedules and standings.
- **Key Characteristics:** Primarily interested in viewing game schedules and standings, with minimal interaction beyond that.
- **Importance:** Low – their user experience should be smooth, but they have fewer system interactions compared to managers and owners.

City Officials and Parks & Rec Staff

- **Frequency of Use:** Low – occasional interaction to verify venue usage, city regulations, and game scheduling.
- **Technical Expertise:** Low – basic ability to navigate schedules and confirm adherence to city policies.
- **Security or Privilege Levels:** Low – read-only access to schedules and league information to ensure compliance with local regulations.
- **Key Characteristics:** These users are focused on ensuring that league activities comply with city policies and that venue usage is properly managed.
- **Importance:** Low – necessary for system compliance but minimal interaction with the system.

2.4 Operating Environment

(Tentative, TBD)

Hardware platform: The system is mainly deployed on cloud servers and supports processors based on the x86 architecture. The minimum recommended configuration is 4 GB of RAM, 40 GB of storage space, and a stable network connection. For small-scale event management, it is also supported to run on a local server, which requires a corresponding hardware configuration.

Operating system: The system will support the following operating systems and their versions:

- Linux: Ubuntu 20.04 and higher, CentOS 7.9 and higher
- Windows: Windows Server 2019 and higher, Windows 10 and higher
- MacOS: MacOS 11.0 Big Sur and higher (for development and testing environments)

Database: The system will use MySQL or PostgreSQL as the back-end database, and is compatible with other SQL-based database systems.

Web browser: The client will run in a modern browser and is compatible with the following browsers:

- Google Chrome: version 90 and higher
- Mozilla Firefox: version 88 and higher
- Safari: version 14 and higher
- Microsoft Edge: version 90 and higher

Programming language and framework:

- The backend was developed using Python and the Django or Flask frameworks.
- The frontend uses HTML5, CSS3, JavaScript and the React or Vue.js frameworks.

Compatibility: The SPORTS system requires compatibility with the following external components or services:

- Payment gateway: compatible with payment processing platforms such as Stripe and PayPal.
- Weather API: uses an external weather API (such as OpenWeather or AccuWeather) to obtain real-time weather information.
- City regulation system: can be integrated with city or local management platforms to detect and handle regulations and permits related to the venue.

2.5 Design and Implementation Constraints

Regulatory Policies: The system must comply with local and national regulations regarding data protection and privacy. For example, compliance with the General Data Protection Regulation (GDPR) or equivalent local regulations will be required for handling user data such as personal information and payment details.

(extra part, TBD)

Hardware Limitations: The system will need to function on a range of devices, including both desktops and mobile devices with varying hardware capabilities. This requires efficient memory and resource management, ensuring the system can run smoothly even on low-end devices. There is a timing requirement for the system to process game schedules and venue allocations within 2 seconds under typical load conditions.

Database and Technology Constraints: The system will use MySQL or PostgreSQL as the primary database for managing teams, leagues, and schedules. Integration with these databases is required, and any changes to the database schema should maintain backward compatibility. Furthermore, the use of RESTful APIs for external communications, such as real-time weather updates or score reporting, must be adhered to.

Parallel Operations: The system should support concurrent operations from multiple users, such as team managers registering teams or venue owners updating schedules. Careful consideration must be given to ensuring data consistency and preventing race conditions, particularly when scheduling games across shared venues.

Language and Platform Constraints: The SPORTS system will be implemented using Python as the primary programming language due to its flexibility and wide library support for web and database integration. The system must be cross-platform, supporting major operating systems like Windows, macOS, and Linux. Web-based components should be designed to function on modern browsers (Chrome, Firefox, Safari).

Security Considerations: All user data, including personal information and payment data, must be encrypted in transit and at rest. Secure communication protocols such as HTTPS will be mandatory. The system must also include user authentication, including role-based access control, to ensure that users only have access to the appropriate sections of the system.

Programming Standards: The development team will adhere to industry-standard programming practices, including following PEP 8 for Python code style and utilizing version control (e.g., Git) for managing development. The system will also use automated testing frameworks to ensure robustness and maintainability.

2.6 User Documentation

- User manual documentation on how to use the product will be provided.
- In case the user manual is not sufficient to solve the issues, help will be provided to the users during the business hours through emails.

2.7 Assumptions and Dependencies

- It is assumed that all users, including league managers, team managers, and venue owners, have access to reliable internet connections. The system's real-time functionalities, such as live score updates and scheduling, depend on constant internet access.
- It is assumed that the hardware resources available to run the system, both server-side and client-side, are adequate for handling up to 500 concurrent users without significant performance degradation.
- The system will assume that external weather services providing updates for venue scheduling will remain operational and accurate. The scheduling algorithm relies on real-time weather data to prevent venue conflicts due to inclement weather.
- The assumption is that the third-party payment processing system integrated into the SPORTS platform will remain available and provide secure transactions for league registration payments.
- It is assumed that all user data, especially regarding personal details and payment information, will be securely transmitted and stored in compliance with local and international regulations (e.g., GDPR).
- The system depends on third-party libraries and APIs for handling tasks such as payment processing, weather forecasting, and live score updates. If any of these services experience downtime or disruptions, certain system functionalities (e.g., payment processing or weather-based scheduling) may be affected.
- The system relies on external database management systems (MySQL or PostgreSQL) for data storage and retrieval. Any changes in the availability or version compatibility of these databases could affect system performance and reliability.
- The SPORTS system is dependent on external authentication mechanisms to provide secure access to users. Any updates or changes to these authentication protocols (e.g., OAuth or multi-factor authentication systems) must be incorporated to maintain security.
- The system is also dependent on ongoing compliance with local and international data protection laws. Changes in regulations could necessitate system updates to ensure compliance, potentially affecting project timelines or system functionality.

3. External Interface Requirements

3.1 User Interfaces

The SPORTS system shall provide an intuitive graphical user interface (GUI) for various users, including league managers, team managers, and venue owners. The interface will include clear navigation through tabs or dropdowns for managing leagues, registering teams, and scheduling games. Key elements such as buttons for "Register," "Schedule Game," and "View Standings" will

be consistently placed across screens. Error messages will be displayed at the top of the screen in red when invalid input is detected, and help buttons will be available on all major user tasks. The system will follow a responsive design, ensuring usability on both desktop and mobile platforms. GUI standards for color schemes, fonts, and layouts will adhere to accessibility guidelines for readability and ease of use.

3.2 Hardware Interfaces

The SPORTS system shall interface with common server hardware and cloud storage for data storage and retrieval. It will support integration with mobile devices for real-time score updates during games. The system shall be compatible with typical input devices such as keyboards, mice, and touch screens. Communication between the software and hardware will be conducted over secure communication protocols, ensuring that data, including schedules and scores, is safely transferred to and from the system.

3.3 Software Interfaces

The SPORTS system shall integrate with standard relational databases such as MySQL or PostgreSQL to manage team registrations, league information, and venue scheduling. The system will also interact with external APIs for weather forecasting to prevent scheduling conflicts. Operating systems supported include Windows, macOS, and Linux. Libraries for real-time data processing, such as web sockets for live score updates, will be integrated. All inter-software communication will follow RESTful API standards, ensuring smooth data exchange between modules.

3.4 Communications Interfaces

The SPORTS system shall communicate over standard web protocols such as HTTPS for secure data transmission. All external communications, such as updates on league schedules or payment confirmations, will be sent via email to users. The system shall format all communication messages in plain text with optional HTML formatting for newsletters or updates. Data synchronization between the mobile app and the web version will be handled in real-time through secure channels, ensuring no loss of information.

4. System Features

4.1 Use Case 1: Players Register and View Personal Schedule

Iteration: First Draft

Summary: Players can register for leagues as well as classes and view their personal schedule information through the SPORTS system. In addition, players can register for one or more sport(s) (i.e. volleyball, baseball, softball, pickleball, kickball). Registration includes payment processing, and the schedule is updated to reflect new registrations.

Basic Course of Events:

- Players log in to the system.
- Players select the sport and league they want to register for.
- The system displays the league details, and the player fills in the registration information and pays the fee.
- The system confirms the registration and updates the player's personal schedule.
- Players can view their personal schedule.

Alternative Paths: The player can view a summary of his personal schedule after registering for multiple leagues.

Exception Paths: If the payment fails, the system prompts the user to re-enter the payment information or change the payment method.

Extension Points: None

Trigger: The player wants to register for a league and view his schedule.

Assumptions: The player has already created a personal account in the system.

Precondition: The league and schedule have been entered into the system.

Postcondition: The player has successfully registered, and their schedule is updated with new league registrations.

Author: Wan Wang

Date: September 23, 2024

Functional Requirements:

Requirement #: 4.1.1

Requirements Type: System

Date: 10/05/2024

Introduction: The system can be accessed to register only when the customer logs in with his/her credentials for either registration or viewing purposes. The system shall support a secure authentication process so that only authorized users are allowed to proceed.

Rationale: Secure login systems ensure that unauthorized access to systems by users is prevented and therefore sensitive user information is protected.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Credentials of the Player (username/email and password)

Requirement Description: The system shall require the players to log in by entering their credentials (such as username and password) to proceed so that only authorized users can move further.

1. If correct credentials have been provided, the user shall access it for registration or any other relevant detail such as viewing league or team information.
2. If the credentials are wrong, the system shall display a specific error message *“Incorrect username or password, Please try again”* to re-input the correct data.
3. The system shall allow the players to reset or recover their password after confirming their identity (through a one-time code or a security question) via email verification if they forget their credentials.

Outputs:

If credentials are correct: Grant access to the user dashboard. If credentials are incorrect: Display the message *“Incorrect username or password. Please try again.”*

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Conflicts: None

Test Cases: TBD

Requirement #: 4.1.2

Requirements Type: System

Date: 10/05/2024

Introduction: To enhance usability, the system shall give a sport and league selection facility along with an option to select classes to the user from a set of predefined lists. Once such a choice has been made, players are presented with comprehensive league information such as schedules and teams.

Rationale: Providing a predefined list of sports and leagues along with classes will simplify the user registration process, ensuring users receive accurate and relevant information.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Selected sport and league

Requirement Description: The system shall allow players to choose a sport and league from a predefined set.

1. If there is a valid sport and league along with the class chosen, then the system shall present to the player detailed information about the league, such as rules, schedules, and teams involved.
2. If no leagues are available for the sport selected, then the system shall display a message *“No leagues available for this sport. Please select another sport.”* and prompt them to select another sport.

3. If no classes are available for the selected option, the system shall prompt “Please select another class”.
4. If the player doesn't choose a sport or league, then a message “*Please select a sport and league to proceed.*” will be displayed asking the player about their choice.
5. In case of an invalid selection by the player or if the selection is not available, then an error message will be displayed stating “*Invalid selection. Please choose from the available options*” along with a refreshed options list. Also, the player will be able to go back and change the present selection at any time during the processing.

Outputs:

If a valid sport and league is chosen: Display detailed information about the league, including rules, schedules, and teams. If no leagues are available: Display the message “*No leagues available for this sport. Please select another sport*”. If no selection is made: Display the message “*Please select a sport and league to proceed*”. If invalid selection: Display message “*Invalid selection. Please choose from the available options*” along with a refreshed options list.

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.1

Conflicts: None

Test Cases: TBD

Requirement #: 4.1.3

Requirements Type: System

Date: 10/05/2024

Introduction: The system should display a form to the user with which he/she will be able to finish the registration.

Rationale:

Author: Poorna Bengaluru Shivaji Rao

Inputs: Registration information

Requirement Description: The system will provide all the selected League and class details on the screen and prompt players to fill out a registration form including mandatory fields such as their name, age, and contact details.

1. If all the fields are filled as required (e.g., valid age range, contact format), the system proceeds with the registration and displays a confirmation of registration stating “*Your registration has been confirmed*”.
2. The system shall warn a user to correct if any of the required fields are missing or

contain invalid entries by displaying an error message stating “*Please fill in all required fields correctly*” and highlighting the fields that need modification.

Once the user's information has been brought to a state where it complies with the league requirements, the system will confirm this with a success message as stated above. If errors occur in processing, the system will notify the user and provide options in case support is needed.

Outputs:

Display a registration form with fields for name, age, and contact details. If registration is successful: Display the confirmation message “*Your registration has been confirmed.*” If required fields are missing or invalid: Display the error message “*Please fill in all required fields correctly*” and highlight the fields that need correction.

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.2

Conflicts: None

Test Cases: TBD

Requirement #: 4.1.4

Requirements Type: System

Date: 10/05/2024

Introduction: The system shall handle the transaction process to allow a player to register to their chosen league as straightforward as possible. This system will ensure that players can register with a secured payment facility. All payment methods will be offered for different league registrations.

Rationale: Efficient payment methods will help to enhance the user experience and timely registration. At the same time, the system informs the players promptly about the status of their payment and thus instills confidence in the registration process and minimizes any potential problems that may occur in making the payment, hence deterring them from joining the league.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Payment information

Requirement Description: The system shall facilitate the payment process for players who want to join the league more easily by providing different options (such as credit cards, debit cards, or online banking) in which one can input their payment details. Once payment details are put into the system, it shall process the transaction and send notifications regarding the status of the payment to the player as stated below:

1. In case the payment is done, there will be a confirmation message for all the participants concerning the registration stating, “*Your payment has been processed successfully, and your registration is complete*” and updating the player’s registration status accordingly.

2. If the transaction has failed, the system will throw an error message stating, *“Payment failed. Please check your payment details and try again”*, allowing them to re-enter the details or to choose a different payment method.

Outputs:

If payment is successful: Display the message *“Your payment has been processed successfully, and your registration is complete”* and update the player’s registration status accordingly. If payment fails: Display the message *“Payment failed. Please check your payment details and try again”* and provide options to re-enter details or choose a different payment method.

Persistent Changes: Payment transaction records.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.3

Conflicts: None

Test Cases: TBD

Requirement #: 4.1.5

Requirements Type: System

Date: 10/05/2024

Introduction: This feature aims to confirm a player’s registration once the payment has been done successfully. Additionally, it also updates the player’s schedule and ensures that they receive timely and accurate status of their registrations.

Rationale: The confirmation of the registration and schedule updates is promptly done through notifications, which in turn gives players more confidence and satisfaction. Such a process reduces any kind of confusion among players, keeps them organized, and ensures they remember the times they are committed to league activities.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Payment information

Requirement Description: The system aims to confirm that the player indeed was registered upon receiving a successful payment and updates the player's schedule. The players will be informed of their status, regardless of the outcome of their payment

1. The registration with updated schedules shall be confirmed to the player if the payment has been processed.
2. If the payment fails, the system at this point sends a notification to the player, showing an unsuccessful transaction along with an error message stating, *“Unable to process your request”*, and requests for repayment.

Outputs:

If payment is successful: Display a message confirming registration and updated schedule. If payment fails: Display the message *“Unable to process your request”* and request repayment.

Persistent Changes: Registration Confirmation Status – Success or Failure; Updated Personal Schedule.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.4

Conflicts: None

Test Cases: TBD

Requirement #: 4.1.6

Requirements Type: System

Date: 10/05/2024

Introduction: This feature is meant to promote players' abilities to access their schedules in real time. By providing the means for players to always check their schedules, the system enhances their engagement and better prepares them for league activities.

Rationale: By showing players their newly updated schedules, they are informed of any commitments. In this regard, it enhances better organization and planning among the players, thus helping them use their time wisely.

Author: Poorna Bengaluru Shivaji Rao

Inputs: None

Requirement Description: The system shall grant players access to view their updated personal schedule at any time. It shall provide the player with information on upcoming events like matches, practice sessions, and all other major events.

1. If a player requests a schedule, the system shall display the current schedule along with updates
2. If no schedule is found, the system informs the player that no events have been scheduled for now by displaying “No schedule found!”.

Outputs:

If a schedule is requested: Display the current schedule with upcoming events. If no schedule found: Display the message “*No schedule found!*”

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.5

Conflicts: None

Test Cases: TBD

4.2 Use Case 2: Team manager Register Team and Manage Payments

Iteration: First Draft

Summary: The team manager can register a team for a league, pay fees, and view the team's match history. The system allows registration for multiple teams and provides flexible payment methods.

Basic Course of Events:

- The team manager logs in to the system.
- The team manager selects the league to register for.
- The team manager enters the team information and makes the payment.
- The system confirms the registration and generates the match history.
- The team manager can view the team's match history and payment status.

Alternative Paths: The team manager can select different payment methods or register multiple teams for different leagues at once.

Exception Paths: If the payment fails, the system prompts to re-enter the payment information.

Extension Points: None

Trigger: The team manager wants to register the team for the league and pay the fee.

Assumptions: The team manager has registered and has access to the system.

Precondition: The league and payment gateway have been configured.

Postcondition: The team manager successfully registers the team and pays the fee, and can view the match records.

Author: Wan Wang

Date: September 23, 2024

Functional Requirements:

Requirement #: 4.2.1

Requirements Type: System

Date: 10/05/2024

Introduction: This feature will implement the functionality for securing team registration by ensuring that the managers log in with their credentials. These will shield team registrations from unauthorized personnel and guarantee the integrity of the registration system.

Rationale: The team managers shall be given a secure login to protect sensitive information concerning the teams from being accessed by the unauthorized. This further promotes accountability in the running of the teams and helps maintain the overall security of the league's registration system.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Team manager credentials (username and password)

Requirement Description: The system shall provide the capability of login for team managers to authenticate themselves for accessing the available options for registering teams. The aforementioned login process is quite important for securing the functions of team management and team registration is only initiated by those who have the authority to do so.

1. If the correct credentials (username and password) have been entered, the system shall present options to select leagues and register teams.
2. The system will generate an error message for wrong inputs of credentials stating *“Incorrect username or password, Please try again”* and the manager has to re-input the login credentials.
3. In addition, in case the team manager forgets their password, then the system will display a recovery option for the password in order to reset it securely via email verification once their identity is confirmed.

Outputs:

If credentials are correct: Provide access to league selection and team registration options. If credentials are incorrect: Display the message *“Incorrect username or password. Please try again.”* If password recovery is initiated: Display the message *“Password recovery link has been sent to your registered email.”*

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.1

Conflicts: None

Test Cases: TBD

Requirement #: 4.2.2

Requirements Type: Functional

Date: 10/05/2024

Introduction: This feature will implement the functionality for the team manager to register for a league and make himself the manager.

Rationale: Team managers shall act as the single point of contact for the teams and will be able to manage multiple teams and leagues.

Author: Prajwal Umesha

Inputs: Selected sport and league

Requirement Description: The system shall allow the team manager to select any sport and league from a predefined dropdown list.

1. If a sport and league are validly selected, then the system shall display all the details related to that league, such as rules, schedules, and registration requirement.
The system shall provide a form that shall contain the following fields:
 - Manager's name
 - Team Name
 - Sport Selection
 - League selection
 - Contact Information - Email, Phone Number
2. When no leagues are available for the selected sport, the system shall indicate this to the team manager and return to the Sport selection screen.
3. If the manager does not select a sport and/or league, after some time a message shall appear on the screen asking for selection.

4. In the case of a wrong or unavailable selection, the system shall present the error with a new list of available options.
5. The user should be able to create an account for the league as a team manager.

Outputs: Confirmation of registration if successful.

Persistent: Changes None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.2

Conflicts: None

Test Cases: TBD

Requirement #: 4.2.3

Requirements Type: Functional

Date: 10/05/2024

Introduction: This feature implements the feature for the team manager to register players for the league.

Rationale: Team manager is able to manage the details of all team members and register players to the team..

Author: Prajwal Umesha

Inputs: league, player details

Requirement Description: The system will allow the team manager to register players for the league.

1. The system shall provide a form that includes the following fields:
 - Player's name
 - Player's age
 - Player's position
 - Contact information - phone number, email
 - Team affiliation
2. The team manager shall first fill in their details, followed by each player's details.
3. If all required fields are completed, the system shall proceed with the registration.
4. The system shall validate the information entered and notify the manager to correct any invalid or missing fields.
5. Once all fields conform to league requirements, the system shall confirm the successful registration of the team.

Outputs: Confirmation of team registration.

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.2, Requirement 4.2.2

Conflicts: None

Test Cases: TBD

Requirement #: 4.2.4

Requirements Type: Functional

Date: 10/05/2024

Introduction: This feature allows a team manager to pay the registration fee for the league.

Rationale: The payment is needed for the registration

Author: Prajwal Umesha

Inputs: Payment information

Requirement Description: The system shall facilitate the payment process for team managers. The payment process shall be done with a third party(TBD) provider only.

1. The system shall present a payment form with the following fields:
 - Payment method - credit card, debit card, or PayPal
 - Cardholder's name
 - Card number
 - Expiry date
 - CVV
2. Upon successful payment, the system shall display a confirmation message.
3. In case of a failed transaction, the system shall display an error message prompting the manager to review and correct payment details.
4. The system shall send an email confirmation of the payment to the manager, with a summary of the transaction.
5. The system shall send an email confirmation of the payment to the event manager, with a registration confirmation of the team.

Outputs: Payment status - Success or Failure.

Persistent Changes: Payment transaction records.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: Requirement 4.1.3

Conflicts: None

Test Cases: TBD

Requirement #: 4.2.5

Requirements Type: Functional

Date: 10/05/2024

Introduction: The system shall generate and display a team's match history after successful registration and payment.

Rationale: This feature provides team managers with a comprehensive view of past matches, including outcomes and statistics.

Author: Prajwal Umesha

Inputs: Payment confirmation, match details

Requirement Description: The system shall display a team's match history.

1. The team history page will be empty upon registration and updated as matches are completed.
2. The system shall display details such as:
 - Match date and time
 - Venue
 - Opponent
 - Match outcome - win/loss
 - Score
3. As matches are played, the system shall update the match history with the relevant details.

Outputs: Team match history.

Persistent Changes: Match history records.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Conflicts: None

Test Cases: TBD

Requirement #: 4.2.6

Requirement Type: System

Use Case: Team's Roster Update

Introduction: The system can allow the team managers to update their team's roster including adding or removing players.

Rationale: It can help the team information is current and reflects any changes due to transfers or injuries.

Author: Haifeng Huang

Inputs: Team roster updates

Requirement Description: The system can allow the team manager to upload their current team's roster and edited it anytime

1. Team manager can update the document or edit the roster such as:
 - Add new team member
 - Remove injured or transferred team member
 - Change some general information about the team

Outputs: Confirmation of roster updates.

Persistent Changes: Updated team roster information.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Test Cases: TBD

4.3 Use Case 3: View and Manage Venue Schedule

Iteration: First Draft

Summary: The event manager uses the system to schedule competitions and ensure there are no conflicts with the venue's availability. The system assists in preventing conflicts with automatic time slot adjustments.

Basic Course of Events:

- The event manager selects a venue and a date.
- The system displays the available time slots for that date.
- The event manager selects an available time slot and enters the competition details.
- The system checks for any time conflicts.
- If there are no conflicts, the system confirms that the competition has been successfully scheduled.

Alternative Paths:

If there is a time conflict at the venue, the system will prompt the user to select another time slot or venue.

Exception Paths:

If, at step 5, the system detects a conflict at the venue after the match has been entered, the system will automatically adjust other possible time slots for selection.

Extension Points: None

Trigger: The venue owner wants to schedule a match.

Assumptions: The venue information has been entered into the system and there are free slots available for scheduling.

Precondition: The venue must be free and there must be no conflicting matches scheduled.

Postcondition: The match has been successfully saved and is displayed in the schedule.

Author: Wan Wang

Date: September 23, 2024

Functional Requirements:

Requirement #: 4.3.1

Requirements Type: Functional

Date: 10/05/2024

Introduction: The system shall allow venue owners to provide details about the venue, including facilities and capacity.

Rationale: This feature provides organizers with venue information, enabling them to host leagues at appropriate locations.

Author: Prajwal Umesha

Inputs: Venue details

Requirement Description: The system shall allow venue owners to enter venue details using a form that includes the following fields:

1. Venue name
2. Address
3. Supported sports
4. Number of courts/grounds for each sport
5. Seating capacity

Venue owners shall also have the option to update or modify these details as needed.

Outputs: Venue information displayed to organizers.

Persistent Changes: Venue details.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Conflicts: None

Test Cases: TBD

Requirement #: 4.3.2

Requirements Type: Functional

Date: 10/05/2024

Introduction: The system shall allow event owners to select a venue and date for scheduling competitions. It will then display available time slots for the selected date to facilitate scheduling.

Rationale: This feature helps organizers easily manage venue schedules and select appropriate dates and time slots for league matches.

Author: Prajwal Umesha

Inputs: Venue details, date, time slots

Requirement Description: The system shall allow venue owners to list the availability of the venue for scheduling.

1. Venue owners shall be provided with a form containing the following fields:
 - Venue name (dropdown with registered venues)
 - Date (date picker)
 - Available time slots (time picker for start and end times)
 - Number of available courts/grounds (number field)
2. Once a venue is selected, the owner shall input the available dates and times in the provided form.
3. The system shall display a calendar showing the venue's availability to organizers for scheduling purposes.
4. The form shall allow venue owners to modify or update availability as required.

Outputs: Updated venue availability details displayed on the calendar.

Persistent Changes: Venue availability records.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements:None

Conflicts: None

Test Cases: TBD

Requirement #: 4.3.3

Requirements Type: Functional

Date: 10/05/2024

Introduction: The system shall allow organizers to book a venue for a league for a specific duration. This allows the venue to be available for the league completely.

Rationale: This feature allows organizers to book a venue for the duration of the league, ensuring its availability for conducting matches.

Author: Prajwal Umesha

Inputs: Booking details (venue, date, time slots)

Requirement Description: The system shall allow the organizer to book the venue for the league.

1. The organizer shall be able to look at the calendar for available dates and time slots for the venue.
2. The system shall display a booking form with the following fields:
 - Venue name (dropdown list with registered venues)
 - League name (dropdown of available leagues)
 - Date (date picker)
 - Time slots (start and end times, time picker)
3. Organizers shall be given an option to book the venue based on available slots.
4. Upon proceeding to book, the system shall check for conflicts and display a confirmation message that the venue is booked.
5. Once the booking is confirmed, the venue's availability shall be updated on the calendar, preventing others from booking the same time slots.
6. If there is a scheduling conflict due to double booking or unavailable time, the system shall display an error message informing the organizer of the conflict and suggesting alternate times.

Outputs: Confirmation of the venue booking, or an error message in case of conflict.

Persistent Changes: Venue booking records and calendar updates.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements:None

Conflicts: None

Test Cases: TBD

4.4 Use Case 4: View League Schedule and Standings

Iteration: First Draft

Summary: League officials can view the complete league schedule and standings, including exporting match reports and results data for analysis.

Basic Course of Events:

- The league official logs in to the system.
- The system displays the current schedule and standings.
- The league official selects a specific league to view the detailed schedule and results data.
- The system displays standings, match results, and team performance.
- The league official can export the schedule and results report for further analysis.

Alternative Paths: The league official can choose to view the match schedules and results for multiple leagues.

Exception Paths: If no match records are available, the system will prompt the league official to try again later.

Extension Points: None

Trigger: The league official wants to view the match schedules and results for the entire league.

Assumptions: The league official has access to the data for multiple leagues.

Precondition: The system has recorded the match results and ranking data.

Postcondition: The league official successfully views the match schedule and results.

Author: Wan Wang

Date: September 23, 2024

4.5 Use Case 5: Monitor Venue Regulations and Weather

Iteration: First Draft

Summary: system is able to monitor the weather and safety regulations in real-time at venues to ensure compliance with city regulations.

Basic Course of Events:

- City officials log in to the system.
- The system displays current weather conditions and regulatory information at the venue.
- City officials select a venue to view regulations and weather warnings.
- The system confirms if the venue complies with city regulations based on the weather conditions.
- City officials can issue compliance reports and notify relevant departments.

Alternative Paths: City officials can select different venues to view regulations and weather information.

Exception Paths:

If the system fails to obtain data from the weather API, the city official is prompted to try again later.

Extension Points: None

Trigger: City officials want to check the weather and safety regulations at the venue.

Assumptions: Venue information, weather API, and regulatory data are configured.

Precondition: The system is connected to the weather and regulatory APIs.

Postcondition: The city official successfully monitors the weather and regulations at the venue to ensure compliance.

Author: Wan Wang

Date: September 23, 2024

4.6 Use Case 6: Maintain Match Records and Analyze Standings

Iteration: First Draft

Summary: The system records game results, maintains standings, and generates statistical analysis for league officials and users to view. It automatically updates team performance data and recalculates rankings when results are modified.

Basic Course of Events:

- After the game, the referee or league official logs in to the system.
- The referee enters the final results of the game, including the score and key statistics (e.g. points scored, fouls, etc.).
- The system validates and saves the game results.
- The system automatically updates the league standings and team statistics.
- Users (e.g. league officials or team managers) can view game analysis, ranking changes and team performance statistics.

Alternative Paths: The referee or league official can manually modify the game record after the game is over, and the system will automatically recalculate the rankings and analysis.

Exception Paths: If data entry is incorrect, the system prompts the user to re-enter or confirm the data. If validation fails, the system requires administrator intervention.

Extension Points: None

Trigger: After the game is over, the referee or league official needs to record the game results and update the league rankings.

Assumptions: The match has ended and the referee has permission to log in and record the results.

Precondition: The basic information about the league, teams and matches already exists in the system.

Postcondition: The results of the match have been recorded, the league standings and analysis data have been updated, and users can view the updated statistics.

Author: Wan Wang

Date: September 23, 2024

Functional Requirements:

Requirement #: 4.6.1

Requirements Type: System

Use Case: Match Results Entry

Date: 10/07/2024

Introduction: League officials must enter match results into the system promptly after a game, ensuring that league standings are updated in real time.

Rationale: Keeping match results and standings up-to-date improves the accuracy and reliability of the league standings.

Author: Wan Wang

Source: League management

Inputs: Match result information

Requirement Description: The system shall enable league officials to input match results after each game, automatically updating league standings and statistics.

League officials log the results and statistics of completed matches.

The system processes and updates standings and stats in real time.

Outputs: Confirmation of match results and updated standings.

Persistent Changes: Updated league standings and team/player statistics.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None**Conflicts:** None

Support Materials: Match result entry guide

Test Cases: Verify that the match results are entered, and standings are updated accordingly.

Requirement #: 4.6.2

Requirement Type: System

Introduction: The system can provide a search function for users to find suitable opportunities.

Rationale: Team players can use the system to search the team or leagues more convenience when they type some specific criteria.

Author: Haifeng Huang

Inputs: None

Requirement Description: The system display as a searching windows and users can type words in the searching bar:

1. Users can type the teams' full name or the leagues' full name to find the object.
2. If users forget the full name, some key words also can help the system to find the object.

3. If users do not know anything about the teams or leagues, they also can type the keywords to find all of objects which have the keywords tag, such as:
 - Location
 - Skill level
 - Foundation history
 - Award record
 - Famous members
 - Impressive match

Outputs: Search results for teams or leagues.

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Conflicts: None

Test Cases: TBD

Requirement #: 4.6.3

Requirement Type: System

Use Case: User Log Recording

Introduction: The system can record the logs of user access and actions to enhance security and accountability, tracking the activities of users.

Rationale: These records can help in monitoring system usage and identifying any unauthorized access attempts.

Author: Haifeng Huang

Inputs: Security and user access logs

Requirement Description: The system can record the log which comes from the user activities

1. The recording include all of the user's activities such as:
 - Searching history
 - Download files
 - Text
 - Deleted files
 - Transfer files activities
 - Edited information
2. If there are some threats or the database has risk to be attacked, these log records can help to find where the threat comes from and how to transpass the firewall to attack the data.

Outputs: User access and action logs.

Persistent Changes: Stored security logs.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Conflicts: None

Test Cases: TBD

4.7 Use Case 7: Scheduling league matches at Venue

Iteration: First Draft

Summary: The system checks for time conflicts when scheduling a match to ensure that there are no overlapping matches scheduled for the same venue at the same time.

Basic Course of Events:

- The venue owner or league official logs into the system and selects the venue where the match is to be scheduled.
- The system displays the current schedule and available time slots for the venue.
- The venue owner or league official selects a time slot and enters the match details.
- The system checks to see if the selected time slot is already occupied by another match.
- If there is no conflict, the system confirms that the match has been successfully scheduled.
- The system updates the venue schedule to reflect the new match.
- The system shall allow modifications to schedules in case a team drops out or the schedules need to be changed due to the weather conditions.

Alternative Paths:

If the user selects a time slot that conflicts with another match, the system prompts the user to select a different time slot or venue.

Exception Paths: If another administrator attempts to schedule a match in the same time slot after the match has been successfully scheduled, the system will prompt a time conflict and prevent the new schedule.

Extension Points: None

Trigger: The user attempts to schedule a match for a venue.

Assumptions: The venue information and the current schedule have been entered into the system.

Precondition: The venue schedule must have been entered into the system, and it needs to be ensured that there are no other matches at the venue during the scheduled time period.

Postcondition: The system successfully prevents conflicts in the venue schedule and ensures that there are no conflicts in the match schedule.

Author: Wan Wang

Date: September 23, 2024

Functional Requirements:**Requirement #:** 4.7.1**Requirements Type:** Functional**Use Case:** Create Venue Schedule**Date:** 10/09/2024

Introduction: It shall provide functionality to the users, who are the venue owners or league officials, to create a new schedule for the venue by choosing available time slots and filling in the details of the matches. In this way, the feature contributes to the venue schedule being updated in real time for transparency and organization.

Rationale: Creating a schedule for a venue ensures that the venue's availability is properly managed and that all upcoming matches are documented and can be viewed by authorized users.

Author: Poorna Bengaluru Shivaji Rao

Source: Venue Scheduling Guidelines

Inputs: Available venue schedule, match details

Requirement Description:

- The system should provide the capability for users to view the availability of the venue in terms of time slots before being able to schedule a match.
- The system should provide a user interface to input match details: teams, date, and time of the match.
- The system should update the venue schedule in case of confirmation.
- The system should allow administrators to view and modify the schedules of venues.

Outputs: Updated venue schedule with newly added match details.

Persistent Changes: The system shall update the venue schedule database to reflect the new match entry.

User Satisfaction: 5

User Dissatisfaction: 3

Related Requirements: Requirement 4.7.1

Conflicts: None

Test Cases:

Verify that the system allows users to view available time slots.

Verify that the system updates the schedule after a match is successfully added.

Requirement #: 4.7.2**Requirements Type:** Functional**Use Case:** Prevent Scheduling Conflicts at Venue**Date:** 10/09/2024

Introduction: The system shall manage scheduling conflicts in venues to ensure that no two matches can have the same time slot. This feature helps to maintain the organization of the process of venue scheduling, thus every match can get a separate time slot.

Rationale: Scheduling conflicts need to be canceled to avoid overbooking of resources and to ensure smooth operations of competitions.

Author: Poorna Bengaluru Shivaji Rao

Source: Venue Scheduling Guidelines

Inputs: Venue schedule, match details

Requirement Description:

- The system will automatically check for conflicts in an attempt to schedule a new match and notify the user in case the selected time slot is occupied.
- In this case, the system should flag a conflict and ask the user to choose some other time slot or venue.
- In case of a successful match scheduling, when another administrator tries to schedule another match on the same time slot, the system shall notify the administrator about the conflict and hinder the new schedule.
- This system is supposed to update the venue schedule in case of a match confirmation.

Outputs: Confirmation of match scheduling or conflict notification.

Persistent Changes: Updated venue schedule.

User Satisfaction: 5

User Dissatisfaction: 3

Related Requirements: None

Conflicts: None

Test Cases: Verify that the system detects scheduling conflicts and prevents double booking of venues.

Requirement #: 4.7.2

Requirements Type: Functional

Use Case: Modify Schedule in Case of Conflict

Date: 10/09/2024

Introduction: At any level of conflict during the scheduling process, the system shall provide capability for manual overriding of the schedule by an owner of a venue or league officials. Since manual intervention is kept to a minimum, it reduces any possibility of an overlap when matches are scheduled.

Rationale: By allowing changes in the schedule at the time of a conflict, the system obtains more flexibility, and match planning proceeds without having to restart the scheduling process.

Author: Poorna Bengaluru Shivaji Rao

Source: Venue Scheduling Guidelines

Inputs: Conflict details, available time slots

Requirement Description:

- The system should allow the facility to change a time slot or a venue in case of a conflict.
- The system shall automatically propose available time slots or alternative venues in line with the schedule and user preferences.
- If no viable options exist at the current venue, the system shall prompt the user to select an alternate date or venue.
- In case of a new time slot or venue chosen, the system shall update the schedule and inform all concerned people about the changes.

Outputs: Confirmation of modified match schedule.

Persistent Changes: Updated venue and match schedule.

User Satisfaction: 5

User Dissatisfaction: 3

Related Requirements: None

Conflicts: None

Test Cases: Verify that the system allows modification of the schedule when a conflict occurs and suggests alternate solutions.

Requirement #: 4.7.3

Requirements Type: System

Use Case: Player Notifications

Date: 10/07/2024

Introduction: Players should be notified of important league updates, such as schedule changes.

Rationale: Keeping players informed through notifications improves engagement and league satisfaction.

Author: Wan Wang

Source: Notification system

Inputs: None

Requirement Description: The system shall send notifications to players about modified league schedules to ensure that they are always informed of changes via an email.

The system shall provide an option to the event organizers to send email to the team managers in such cases.

Outputs: Notifications to players.

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements: None

Conflicts: None

Support Materials: Notification system guidelines

Test Cases: Verify that players receive timely notifications regarding updates.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The SPORTS system shall ensure that scheduling operations (i.e., registering teams and allocating games to venues) are completed within 2 seconds of user input to avoid delays. The system should handle up to 500 concurrent users without a noticeable decline in performance. For real-time score updates, the system must process the update and reflect changes under 1 second.

Requirement #: 5.1.1

Requirements Type: Performance

Date: 10/09/2024

Introduction: The system shall ensure that scheduling operations (i.e., registering teams and allocating games to venues) are completed within 2 seconds of user input to avoid delays.

Rationale: A fast and responsive system is critical for maintaining user engagement and ensuring the smooth operation of scheduling and real-time score updates.

Author: Wan Wang

Inputs: User scheduling input, real-time score updates

Requirement Description: The system shall meet the following performance criteria:

1. The system shall handle up to 500 concurrent users without a noticeable decline in performance.
2. Scheduling operations such as registering teams and allocating games to venues shall be completed within 2 seconds of user input.
3. The system must process real-time score updates and reflect changes within 1 second.

Outputs: Scheduling logs, real-time score update logs

Persistent Changes: Data regarding schedule and score updates

User Satisfaction: 5

User Dissatisfaction: 3

Related Requirements: TBD

Conflicts: None

Test Cases: TBD

5.2 Safety Requirements

The system shall prevent data loss or corruption, ensuring that game schedules, team registrations, and league standings are securely stored and maintained. There shall be a backup

system in place that runs nightly to store all data. If the system detects any potential conflicts in venue scheduling due to weather or other external factors, the system shall immediately notify league organizers to take corrective action.

Safety Requirements

Requirement #:5.2.1

Requirements Type: Safety

Date: 10/05/2024

Introduction: A mechanism in the system shall be put in place to ensure that the safety of players is guaranteed on account of physical and health hazards arising from league activities, such as during matches or training sessions.

Rationale: Keeping players safe reduces injuries and helps in ensuring a safe participation environment that is good for participants, thus guaranteeing them a better experience and well-being.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Health status of players and activity schedule

Requirement Description: The system shall request any special considerations due to health concerns when registering players.

1. The activities shall include safety measures like necessary first aid on location and people who are trained to hold the matches.
2. Any injuries during the league activity reported shall be recorded, and the system shall get in touch with the emergency contact and medical personnel immediately.

Outputs: Player safety protocols and injury reports.

Persistent Changes: Health information records and injury logs.

Outputs:

Persistent Changes: Health information records and injury logs.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements:

Conflicts: None

Test Cases: TBD

Requirement #:5.2.2

Requirements Type: Safety

Date: 10/05/2024

Introduction: The system shall provide players with safety information and resources pertinent to their participation in league activities.

Rationale: The readily available safety information helps players make more informed choices and helps establish a safety culture in the league.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Safety resource content, player access requests

Requirement Description: SCS shall maintain a current library of the safety procedures, best practices, and emergency procedures available to the players.

1. Players shall receive mandatory training about safety procedures before league activities that shall be recorded on the players' profiles.
2. The system shall provide the player with feedback mechanisms to report any safety concerns or recommendations for enhancing the prevailing safety practices.

Outputs: Safety training information and completion records of training.

Persistent Changes: Safety training logs and records of feedback.

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements:

Conflicts: None

Test Cases: TBD

Requirement #:5.2.3

Requirements Type: Safety

Date: 10/05/2024

Introduction: The system shall ensure that personal data is handled in such a way that it is kept safe, thus protecting players' privacy and ensuring compliance with data protection legislation.

Rationale: Personal data protection is critical for maintaining trust in the players and fulfilling legal requirements on data privacy.

Author: Poorna Bengaluru Shivaji Rao

Inputs: Player personal data

Requirement Description: The system shall implement data access controls in order to ensure that access to personal data shall be obtained only by persons having the right to access because of their positions or the performance of certain duties.

1. All personal data shall be made anonymous or pseudonymized wherever possible so as to make any direct and indirect identification of sensitive information impossible.
2. The system shall define an appropriate data retention policy, including the period the personal data shall be stored and processes for secure deletion if the data is no longer required.

Outputs: Access logs, data retention reports

Persistent Changes: Personal data handling logs

User Satisfaction: 5

User Dissatisfaction: 5

Related Requirements:

Conflicts: None

Test Cases: TBD

5.3 Security Requirements

User authentication shall be enforced through secure login protocols, requiring unique credentials for each user role (i.e., venue owners, team managers, league officials). Data encryption shall be implemented for all user including payment information and personal details. The system shall comply with GDPR laws.

Requirement #: 5.3.1

Requirements Type: Security

Date: 10/06/2024

Introduction: The system must ensure that all data transmitted between the client and server is encrypted to prevent unauthorized access..

Rationale: Data encryption protects information from being insecure during transmission.

Author: Prajwal Umesha

Inputs: Data communication

Requirement Description:

1. The system shall use TLS and SSL to encrypt all data transmissions between the client and server.
2. All data transmitted shall be encrypted both at rest and in transit.

Outputs: Secure data transmission..

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 2

Related Requirements: Nonce

Conflicts: None

Requirement #: 5.3.2

Requirements Type: Security

Date: 10/06/2024

Introduction: The system shall ensure that the user is authenticated before modifying any details

Rationale: Ensuring the user is authenticated ensure that the modifications that are done are by the right person and not a hacker.

Author: Prajwal Umesha

Inputs: Username and Password

Requirement Description:

1. The system shall authenticate the users before payments are done.
2. The system shall authenticate the users for any modifications to the registrations.

Outputs: Noce.

Persistent Changes: None

User Satisfaction: 5

User Dissatisfaction: 2

Related Requirements: 4.1.1 and 4.2.3

Conflicts: None

5.4 Software Quality Attributes

The system must demonstrate adaptability to various types of sports that are present in the bulletin, ensuring flexibility for different league structures. Availability must be maintained at 99.5% uptime to ensure continuous access for users. The system shall have updated documentation for both the user and development sides. Reliability measure that is, mean time to failure, should be no less than 100 hours. Portability across different platforms, including desktop and mobile devices, must be ensured to increase accessibility.

5.5 Business Rules

The system shall enforce business rules related to venue booking, ensuring that no more than one event is scheduled at the same venue and time. League-specific rules, such as team size limitations and game durations, must be customizable by league officials but adhere to city and venue regulations.

● 6. Other Requirements

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

Different Access Levels and Roles for System Users

- Date: 30th October 2024
- Responsible Party: Prajwal Umesha
- Reason: Requires business rules and security policies clarification.

Operating Environment

- Date: 0th October 2024
- Responsible Party: Prajwal Umesha
- Reason: Depends on hosting and infrastructure decisions.

External System Interfaces

- Date: 0th October 2024
- Responsible Party: Poorna Bengaluru Shivaji Rao,
- Reason: Depends on confirmed third-party services

Test Cases

- Date: 0th October 2024
- Responsible Party: Poorna Bengaluru Shivaji Rao,
- Reason: Depends on finalized requirements