Algorithms and programming

Requirements analysis and specification By: Miguel Perez Ojeda

Problem identification and requirements analysis Case Study: soccer tournament

Customer	*University professor*
User	*Soccer teams which can participate in the tournament, tournament referees, sports fans and the community.*
Requerimientos funcionales	RF1: Register a team. RF2: Register a player. RF3: Register a referee. RF4: Preload the necessary information to start a tournament: teams, players, and referees. RF5: Calculate and display the draw (matches) for the group stage and the respective dates of this stage. RF6: Assign the referee team to a match according to their availability (nationality). RF7: Record the score of a match.
Problem Context	*The system to be developed will manage an international football tournament with 8 teams, inspired by the Copa América. It must allow the registration of teams and players, generate the tournament fixture, and record scores, goals, cards, and referee assignments. Teams must be registered in the platform beforehand. Fans will have access to statistics, team and player lists, and prize details. The system must be web-based, accessible from various devices, and ensure high availability and quick response times.*

Non-Functional Requirements	• Response Time: The system must respond to user requests in less than 2 seconds. • Scalability: It must be able to handle an increase in the number of users and data without significantly affecting performance. • Platform Compatibility: The system must be compatible with different operating systems (Windows, Linux, macOS) and modern web browsers, though certain browsers may be recommended to users for optimal performance. • Code Readability: The code must be readable and easy to understand for any developer. • Extensibility: The system must be easy to extend to include new functionalities in the future. • Availability: The system must have 99.9% availability, ensuring it is operational and accessible almost all the time. • Backup and Recovery: It must have mechanisms for data backup and recovery to protect information in case of system failures or data loss. • Security: The system must ensure the protection of team, player, referee, and match result data against unauthorized access and information loss. • Availability and Maintenance: This requirement establishes the need to guarantee high availability and system maintenance.

Process Requirements	● The program must be developed in Java. ● The program must be designed under the object-oriented programming (OOP) paradigm. • The most critical process requirements for this project focus on ensuring quality, collaboration, and flexibility. A development process that includes constant testing is essential to ensure the system works correctly, along with rigorous change management. Effective communication among all stakeholders and the adoption of agile methodologies are key to ensuring that the system adapts to the project's changing needs and meets the highest quality standards. Additionally, the user interface must be very user-friendly.

Identifier and name	*RF1: Register a Team*		
Summary	*The program will carry out the registration of each team, where the user will be asked for information such as: team name, country of origin, list of 20 players, head coach's name, and assistant coach's name..*		
Inputs	**Input Name**	**Data Type**	**Condition / Valid Values**

Postcondition Outputs	Team Name	String	This field cannot be empty.
	Country of Origin	String	Must be an existing and officially recognized country.
	List of 20 Players	String	Exactly 20 players must be entered.
	Head Coach's Name	String	The first letter of the name must be uppercase and follow the structure: first name, second name (optional), last name, and second last name.
Summary	The information of the team and its players has been successfully saved in the system.		
Inputs	**Output Name**	**Data Type**	**Format**
	Team Registration Confirmation	String	“Player [Player_Name] has been successfully registered.”
	Error Message	String	“Invalid information, please verify that it meets the conditions.”

<table>
<tr><td>Identifier and name</td><td colspan="3">RF2: Register a Player</td></tr>
<tr><td>Summary</td><td colspan="3">The program will carry out the registration of each team, where the user will be asked for information such as: team name, country of origin, list of 20 players, head coach's name, and assistant coach's name. Additionally, this function will include the player registration, which includes their number, name, country of origin, and position (goalkeeper, defender, midfielder, or forward). This function will primarily be used by the team coach.</td></tr>
<tr><td rowspan="5">Inputs</td><td>Input Name</td><td>Data Type</td><td>Condition / Valid Values</td></tr>
<tr><td>Name</td><td>String</td><td>This field cannot be empty.</td></tr>
<tr><td>Country of Origin</td><td>String</td><td>Must be an existing and officially recognized country.</td></tr>
<tr><td>Id</td><td>String</td><td>A number from 0 to 99</td></tr>
<tr><td>Position</td><td>String</td><td>It can only be a goalkeeper, forward or midfielder.</td></tr>
<tr><td>Postcondition</td><td colspan="3">The information of the team and its players has been successfully saved in the system.</td></tr>
<tr><td rowspan="2">Outputs</td><td>Output Name</td><td>Data Type</td><td>Format</td></tr>
<tr><td>Player Registration Confirmation</td><td>String</td><td>“Player [Player_Name] has been successfully registered.”</td></tr>
</table>

Identifier and name	*RF3: Register a Referee*		
Summary	The system will allow the storage of information about referees, requiring input of an ID, referee name, country of origin, and referee type (Central or Assistant). A total of 4 central referees and 8 assistant referees are required. This function will primarily be used by the tournament administrator. It is important to ensure that the referee does not belong to the same country as one of the teams they will officiate.		
Inputs	**Input Name**	**Data Type**	**Input Name**
	Referee Name	String	The name must follow the structure: first name, second name (optional), last name, and second last name (optional). No numbers are allowed in this field.
	Referee ID	String	Must contain only numbers, with a minimum of 5 digits and a maximum of 15. Cannot contain periods.
	Country of Origin	String	Must be an existing and officially recognized country. No invented names or unrecognized countries are allowed.

	Authentication of Country of Origin	String	The referee must provide a document (visa or passport) verifying their country of origin to ensure the information is accurate.
	eferee Type	String	The value entered must be either “Central” or “Assistant.” The maximum number of each type cannot be exceeded (8 central referees and 16 assistant referees). If an attempt is made to register more, a warning message will be generated.
Postcondition	The referee information has been successfully saved in the system, and the total number of referees meets the established requirements.		
Outputs	**Output Name**	**Data Type**	**Format**
	Referee Registration Confirmation	String	“Referee [Referee_Name] has been successfully registered.”
	Error Message	String	“Invalid information, please verify that it meets the conditions.”

Identifier and name	*RF4: Preload Necessary Information to Start a Tournament: Teams, Players, and Referees*

Summary	The system will preload essential information needed to start a tournament, including data on teams, players, and referees. This function will streamline the setup process, ensuring that all necessary participants are registered before the tournament begins.		
Postcondition	All necessary information for teams, players, and referees has been preloaded successfully, enabling a smooth start to the tournament.		
Outputs	**Output Name**	**Data Type**	**Format**
	Preloading Confirmation	String	"All necessary information has been preloaded successfully."
	Error Message	String	"Invalid information, please verify that it meets the conditions."

Identifier and name	RF5: Calculate and display the draw (matches) for the group stage and the respective dates of this stage.		
Summary	The system will automatically calculate and display the schedule for the group stage matches, assigning dates and alternating between groups.		
Inputs	**Input Name**	**Data Type**	**Input Name**

Postcondition	The match schedule for the group stage is generated, ensuring intergroup alternation and that the final matchday for each group is simultaneous.		
Outputs	**Output Name**	**Data Type**	**Format**
	Fixture (matches)	String	(Match Date, Team 1, Team 2)
	Match Dates	String	YYYY-MM-DD
	Group A and B Matches	String	PJ, G, E, P, GF, GC, DG, Pts
	Error Message	String	“Invalid information, please verify that it meets the conditions.”

Identifier and name	RF6: Assign the referee team to a match according to their availability (nationality).		
Summary	The system will assign the referee team (1 central referee and 2 assistant referees) to each match, ensuring that none of the referees have the same nationality as the teams playing the match and that they are available for the scheduled date.		
Postcondition	The referee team (1 central, 2 assistants) is assigned to the match. No referee shares nationality with any team involved, and they are all available for the match.		
Outputs	**Output Name**	**Data Type**	**Format**

	Error Message	String	“Invalid information, please verify that it meets the conditions.”

Identifier and name	RF7: Record the score of a match.		
Summary	The system will allow the user to record the final score of a match, including the goals scored by each team and the players who scored. The system will update the match statistics, such as goals, assists, and points for the teams.		
Inputs	**Input Name**	**Data Type**	**Input Name**
	matchIndex	String	The index or ID of the match to be updated.
	homeScore	String	Number of goals scored by the home team.
	awayScore	String	Number of goals scored by the away team.
Postcondition	The score and match statistics are successfully recorded. The points and goal statistics of the teams are updated accordingly.		
Outputs	**Output Name**	**Data Type**	**Format**

	Success Message	String	"The match result has been successfully recorded."
	Error Message	String	“Invalid information, please verify that it meets the conditions.”