

# USER REQUIREMENT SPECIFICATION (URS)

**Start Date:** 24.04.2022

**End Date:** 10.06.2022

**Location:** Rachelsmolen R10

**Made by:** Martin Todorov

**Tutor:** Ravelo Sánchez, Jesús J.N

## Table of Contents

1. Introduction.....	2
2. Functional requirements .....	4
3. Use cases .....	7

## 1. Introduction

The company DuelSys inc. wants a software solution to allow their customers (sport associations) to

manage their sport tournaments. For now, the software must support a round-robin tournament system for badminton, but DuelSys inc. also wants the software to have the potential to support other types of tournament systems and sports.

A tournament has multiple players competing in badminton games to determine who is the best (e.g. gold, silver and bronze medal). To determine this, the purpose of the software is to register all the results of each game.

This software solution will be used by sport association staff (staff) to organize tournaments and by players to find information about the tournament(s) they want to participate in.

## 2. Functional requirements

The requirements

Below are the requirements for this assignment, and they are divided into three categories

- Core requirements
- Major requirement
- Minor requirement

### Core requirements

#### **Non-functional requirements**

NFR-01: Maintainable and extendable

Proper OO principles must be applied to ensure good maintainability and extensibility of the code base.

NFR-02: Bug free system

Appropriate testing techniques must be used when implementing the system to ensure proper functioning.

NFR-03: Secure software

Only authorized people may make use of the system and can only access data they are authorized for. Passwords and user input must also be handled appropriately.

#### **Functional requirements**

#### FR-01: Manage Tournaments

Staff must be able to manage (CRUD operations) the tournaments. Make sure to include the information given in Process of running a tournament Phase 1.

- *This requirement must be implemented in a desktop application.*

#### FR-02: Support registering players

When a player is interested in participating in a tournament, they can visit the sport association website, retrieve the list of available tournaments and register themselves for it.

- *This requirement must be implemented in a web application.*

#### FR-03: Support generating tournament schedule

Staff must be able to generate the tournament's schedule.

- *This requirement must be implemented in a desktop application.*

#### FR-04: Support registering the results of the games

When a game between two players is finished, the results must be registered in the system by staff.

#### FR-05: Support showing tournament information and results

Any interested party (e.g. a sport enthusiast, a player) must be able to retrieve information about any given tournament.

- *This requirement must be implemented in a desktop application.*

## **Major requirement**

### **Functional requirements**

FR-06: Support multiple tournament systems

Extend the software solution to also support different tournament systems. It should be possible, for a staff member, to specify what tournament system should be used when creating a new tournament. For now, at least one of the following tournament systems (in addition to round-robin) is required:

- o Single-elimination
- o Double-elimination
- o Double round-robin

## Minor requirements

### Functional requirements

FR-09: Support challenge games

Extend the software solution to also support challenge games. It should be possible for a player to challenge another player for a game. When the challenge is created, the opponent can either accept or reject the challenge. If it is accepted, the result of the game can be registered by one of the players.

### 3. Use cases

<b>UC-01</b>	Log in
<b>Actor</b>	Any user
<b>Precondition</b>	Web App must be started
<b>Main success scenario</b>	<ol style="list-style-type: none"><li>1. The user inputs his account details</li><li>2. The system processes the user details</li><li>3. The system informs that the user was successfully logged in</li><li>4. The system redirects the user to the home page</li><li>5. End of use case</li></ol>
<b>Extensions:</b> Incorrect or missing data input	<ol style="list-style-type: none"><li>2.1 Data was missing in some/all fields<ol style="list-style-type: none"><li>1. System informs the user that there was no data entered</li><li>2. End of use case</li></ol></li><li>2.2 Wrong data input<ol style="list-style-type: none"><li>1. System shows that the input data was in the wrong format</li><li>2. End of use case</li></ol></li></ol>

<b>UC-02</b>	Register
<b>Actor</b>	User

<b>Precondition</b>	Web App must be started
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. The user inputs his details.</li> <li>2. The user clicks on register</li> <li>3. System verifies if the data is correctly input.</li> <li>4. The system confirms that the new user account was successfully created</li> <li>5. System redirects user to login page</li> <li>6. End of use case</li> </ol>
<b>Extensions:</b> Employee with the same data already exists in the system.	3.1 User with the same data <ol style="list-style-type: none"> <li>1. The system prompts that there is an employee with the same data.</li> <li>2. The system informs the user that the input data is not in the correct format.</li> </ol> 3.2 End of use case

<b>UC-03</b>	Add tournament
<b>Actor</b>	Admin
<b>Precondition</b>	Admin must be logged in the desktop application
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. Admin inputs the information needed about a new tournament</li> <li>2. Admin submits the new tournament.</li> <li>3. System validates the data.</li> <li>4. System confirms that a new tournament was added</li> <li>5. End of use case</li> </ol>
<b>Extensions:</b> Incorrect or missing data input	3.1 Data was missing in some/all fields <ol style="list-style-type: none"> <li>1. System informs the admin that there was no data entered</li> <li>2. End of use case</li> </ol> 3.2 Wrong data input <ol style="list-style-type: none"> <li>1. System shows that the input data was in the wrong format</li> <li>2. End of use case</li> </ol>

<b>UC-04</b>	Remove a tournament
<b>Actor</b>	Admin
<b>Precondition</b>	Admin must be logged in the desktop application
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. Admin selects the tournament.</li> </ol>



	<ol style="list-style-type: none"> <li>2. Admin clicks on remove tournament.</li> <li>3. The system confirms the tournament has been removed</li> <li>4. End of use case</li> </ol>
<b>Extensions:</b> Can't remove a tournament that has already concluded	<ol style="list-style-type: none"> <li>2.1 Selected tournament has already finished</li> <li>1. System informs Admin that he can't remove a tournament that is already over</li> <li>2. End of use case</li> </ol>

<b>UC-05</b>	Update tournament details
<b>Actor</b>	Admin
<b>Precondition</b>	Admin must be logged in the desktop application  Tournament status is Open
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. Admin selects the tournament.</li> <li>2. Admin changes the information he wants and clicks update</li> <li>3. The system confirms the tournament has been updated</li> <li>4. End of use case</li> </ol>
<b>Extensions:</b>	

<b>UC-06</b>	Open a tournament
<b>Actor</b>	User
<b>Precondition</b>	User must be logged in the web application
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. System displays tournaments</li> <li>2. User selects the tournament.</li> <li>3. System opens the details about a tournament</li> <li>4. End of use case</li> </ol>
<b>Extensions:</b>	

<b>UC-07</b>	Show tournament results
<b>Actor</b>	Any user
<b>Precondition</b>	User must be logged in the desktop application

<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. User requests a concluded tournament from the previous tournaments page</li> <li>2. System retrieves information and results of that tournament</li> <li>3. System displays the tournament and the results</li> <li>4. End of use case</li> </ol>
<b>Extensions:</b>	

<b>UC-08</b>	Challenge a user
<b>Actor</b>	User
<b>Precondition</b>	User must be logged in the web application
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. User opens challenge page and selects another user</li> <li>2. User selects challenge to a game</li> <li>3. System asks for confirmation</li> <li>4. End of use case</li> </ol>
<b>Extensions:</b>	

<b>UC-09</b>	Accept a challenge
<b>Actor</b>	User
<b>Precondition</b>	User must be logged in the web application
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. User opens challenges page</li> <li>2. System displays the challenges a user has</li> <li>3. User selects a challenge from another user</li> <li>4. System asks for an accept or deny</li> <li>5. End of use case</li> </ol>
<b>Extensions:</b>	

<b>UC-10</b>	Register results of a match/challenge
<b>Actor</b>	Admin
<b>Precondition</b>	Admin must be logged in the desktop application
<b>Main success scenario</b>	<ol style="list-style-type: none"> <li>1. System displays the matches/challenge</li> <li>2. Admin selects a match and inputs the scores of the players</li> <li>3. System verifies the results</li> </ol>

	4. End of use case
<b>Extensions:</b> Invalid score	3.1 System displays that the score the Admin input don't correlate to the appropriate sport match rules 1. System notifies the Admin that the results are incorrect 2. End of use case

<b>UC-11</b>	Sign up for a tournament
<b>Actor</b>	User
<b>Precondition</b>	User must be logged in the web application
<b>Main success scenario</b>	1. System displays the available tournaments 2. User selects the tournament. 3. User selects Join tournament 4. System adds user to the tournament 5. End of use case
<b>Extensions:</b> Tournament is full User has already entered this tournament	4.1 Tournament is full 1. System informs user that the tournament has reached its maximum number of players 2. End of use case 4.3. User has already entered the tournament 1. System displays a message that the user has already entered this tournament 2. End of use case