

# Use Case Description

## Use Case Name: Create Tournament

### Related Requirements:

### Goal in Context:

Administrator creates a new sports tournament by defining its general settings, format, and branding elements.

### Preconditions:

- The system has no existing active tournament with the same title.

### Successful End Condition:

A new tournament record is stored successfully and appears in the system's list of tournaments.

### Failed End Condition:

The tournament creation process fails due to invalid inputs or missing information.

### Primary Actor:

Administrator

### Secondary Actors:

### Trigger:

Administrator selects the option to create a new tournament from the platform dashboard.

---

### Main Flow

#### Step Action

- 1 Administrator requests the system to create a new tournament.
- 2 Administrator enters general tournament data (name, sport type, start/end dates, Tournament Organizer ,Game Manager).
- 3 The system stores the new tournament and confirms successful creation.

# Use Case Name: Register Team

## Related Requirements:

### Goal in Context:

Team Manager registers a team to participate in a tournament and completes the payment if required.

### Preconditions:

- The registration period is open.

### Successful End Condition:

The team registration is completed and marked as *Pending Approval* or *Approved*.

### Failed End Condition:

The registration fails due to incomplete information or payment issues.

### Primary Actor:

Team Manager,

### Secondary Actors:

Tournament Organizer

### Trigger:

Team Manager selects the option to register a team in an existing tournament.

---

## Main Flow

### Step Action

- 1 Team Manager selects **Register Team** from the tournament page.
- 2 Team Manager enters team details such as team name, division, and players' info and pay fees of registration .
- 3 Team Manager uploads team documents (player IDs, roster, or logo) to System  
Include::Approve Team
- 4 The Tournament Organizer Approves the Team
- 5 Tournament Organizer confirms registration and submits the application.

## Step Action

6 The team is approved and shown his record.

## Use Case Name: Approve Team

### Related Requirements:

### Goal in Context:

Tournament Organizer approves or rejects team registrations submitted to a tournament.

### Preconditions:

- At least one team registration exists in *Pending Approval* status.

### Successful End Condition:

A team's registration status is updated to *Approved* or *Rejected*.

### Failed End Condition:

Approval fails due to missing documents or system errors.

### Primary Actor:

Tournament Organizer

### Secondary Actors:

### Trigger:

Team Manager opens the pending team approval list in the system.

---

## Main Flow

### Step Action

- 1 Tournament Organizer opens the list of teams awaiting approval.
- 2 The System retrieves team details, uploaded files, and payment status.
- 3 Tournament Organizer reviews the details and selects **Approve** .
- 4 The system updates the registration status accordingly.

### Step Action

5 The system sends confirmation notification.

Extensions

3.1 The Tournament Organizer selects Reject

3.2 The Registration isn't uploaded

3.3 The Team Manager Receives a rejection notification

## Use Case Name: Manage Venues and Time Windows

### Related Requirements:

### Goal in Context:

Game Manager configures the tournament venues and their available scheduling windows.

### Preconditions:

- The Teams have been approved.
- Team Enter Time slots

### Successful End Condition:

All venues and time slots are configured and stored successfully.

### Failed End Condition:

Venue setup fails due to time overlaps or invalid location data.

### Primary Actor:

Game Manager

### Secondary Actors:

### Trigger:

Game Manager selects **Manage Venues and Time Windows** from the tournament settings.

---

### Main Flow

**Step Action**

- 1 Game Manager adds a new venue and specifies its capacity and location.
- 2 The address is verified.
- 3 Game Manager sets the available time slots for tournament.
- 4 The schedule is validated.
- 5 The System saves the venue and its time availability successfully.

**Extensions**

4.1 A Conflict has been verified

2.2 Game Manager adds another Venue or change time slot

## Use Case Name: Find Sponsors

**Related Requirements:****Goal in Context:**

Tournament Organizer finds a sponsor for the tournaments

**Preconditions:**

- The tournament has been successfully created, and no conflicts have been found .

**Successful End Condition:**

The Tournament has a sponsor.

**Failed End Condition:**

The Tournament doesn't have any sponsor

**Primary Actor:**

Tournament Organizer

**Secondary Actors:**

**Trigger:**

Tournament Organizer choose to find **Sponsors for the tournament.**

---

**Main Flow****Step Action**

- 1 Tournament Organizer search for sponsors for the tournament .
- 2 The Sponsors are being displayed for the Tournament organizer to choose from them.
- 3 Tournament Organizer chooses one .
- 4 Tournament Organizer accepts the Deal .
- 5 The tournament has a sponsor .

**Extensions**

4.1 The tournament Organizer doesn't accept the deal

4.2 The Tournament Organizer search for another sponsor

## Use Case Name: Publish Schedule

**Related Requirements:****Goal in Context:**

Tournament Organizer publishes the tournament match schedule .

**Preconditions:**

- Teams, venues, and time windows are configured.

**Successful End Condition:**

The match schedule is generated and made publicly available.

**Failed End Condition:**

The scheduling process fails due to missing or conflicting data.

**Primary Actor:**

Tournament Organizer

**Secondary Actors:****Trigger:**

Tournament Organizer decides to **Publish Schedule** .

---

**Main Flow****Step Action**

- 1 Tournament Organizer requests the system to generate the schedule.
- 2 All the venues, groups, and time slots Data are collected .
- 3 The match fixtures and allocated time slots are created.  
  
Tournament Organizer reviews , choose the kind of schedule and confirms the generated schedule.
- 4 Include::Check schedule  
  
The Tournament Organizer check schedule and verify it
- 5 The Tournament Organizer publishes the schedule and makes it visible to all users.  
  
Extensions  
  
4.1 The Tournament Organizer Rejects the generated schedule  
  
4.2 The Tournament Organizer doesn't publishes the generated schedule  
  
4.3 Another schedule is created

## Use Case Name: Record Match Results & Compute Standings

**Related Requirements:****Goal in Context:**

Game Manager enters completed match results and triggers automatic standings computation.

**Preconditions:**

- The match has been scheduled and completed.

**Successful End Condition:**

Results and updated standings are stored and displayed in real time.

**Failed End Condition:**

Data entry errors prevent standings from updating correctly.

**Primary Actor:**

Game manager

**Secondary Actors: Tournament Organizer****Trigger:**

Game Manager selects a match and chooses **Record Results**.

---

**Main Flow****Step Action**

- 1     Game manager opens the Tournament record .  
  
      Game manager enters scores, cards, and match notes.
- 2     Extend:: **Reschedule/Postpone Match**  
  
      **The Tournament Organizer Reschedule or Postpone match if a team didn't play or issue had happened**
- 3     Team points are calculated or updating standing .
- 4     The system stores updated tables and publishes them.

## Use Case Name: Reschedule/Postpone Match

**Related Requirements:****Goal in Context:**

Tournament Organizer reschedules a match to a different date or venue when conflicts arise.

**Preconditions:**

- The match exists and has not been played or confirmed .



**Successful End Condition:**

The match is successfully assigned a new time slot.

**Failed End Condition:**

No valid slot or venue is available.

**Primary Actor:**

Tournament Organizer

**Secondary Actors:****Trigger:**

Tournament Organizer opens a scheduled match and selects **Reschedule/Postpone**.

---

**Main Flow****Step Action**

- 1     Tournament Organizer selects the match to be rescheduled.
- 2     The Tournament Organizer selects the perfect place for the match and time .
- 3     The Tournament updates the schedule and confirms the change.

## Use Case Name: View Tournament Reports

**Related Requirements:****Goal in Context:**

Administrator views final reports summarizing results, statistics, and outcomes.

**Preconditions:**

- All tournament matches are completed.

**Successful End Condition:**

Reports are generated and made available to Administrator .

**Failed End Condition:**

The Administrator cant view Tournament report

**Primary Actor:**

Administrator

**Secondary Actors:**

tournament Organizer

**Trigger:**

Administrator selects **View Reports** from the tournament menu.

---

**Main Flow****Step Action**

- 1     Tournament Organizer initiates report generation.
- 2     The Tournament is summarized into a report.
- 3     The reports are uploaded .
- 4     The Administrator views the report .

## Use Case Name: Define Tournament Rules

**Related Requirements:****Goal in Context:**

Administrator defines the tournament's competition rules, including scoring system, tie-break criteria, and discipline regulations.

**Preconditions:**

- The tournament has been created.
- The Tournament Organizer and Game Manager are authenticated.

**Successful End Condition:**

Tournament rules are stored and linked to the selected tournament and will be enforced by the system.

**Failed End Condition:**

Rule definition fails due to missing mandatory parameters or invalid configurations.

**Primary Actor:**

Administrator

**Secondary Actors:****Trigger:**

Administrator selects *Define Tournament Rules* from the tournament settings

.

**Main Flow****Step Action**

- 1 Administrator opens the rules configuration page for a specific tournament.
- 2 Administrator enters general rules (match duration, number of players, substitutions, etc.).
- 3 Administrator defines scoring rules (points for win/draw/loss).
- 4 Administrator configures tie-break criteria (goal difference, head-to-head, fair play, etc.).
- 5 Administrator sets disciplinary rules (yellow/red card thresholds, suspensions).
- 6 The rules are validated for completeness and consistency.
- 7 The Tournament rules are saved.
- 8 The Rules are successfully linked to the tournament

**Extensions**

6.1 The Tournament Rules are failed to be verified

6.2 the rules aren't saved successfully

6.3 The rules aren't linked to tournament

# Use Case Name: Collect Fees

## Related Requirements:

### Goal in Context:

Tournament Organizer Collect fees from Registration fees and sponsors

### Preconditions:

- Registration is completed by teams successfully .
- The sponsors are confirmed

### Successful End Condition:

Fees are collected and are saved in Tournament history

### Failed End Condition:

The fees are failed to be saved in tournament history

### Primary Actor:

Tournament Organizer

### Secondary Actors:

### Trigger:

Tournament Organizer selects collect fees.

### Main Flow

#### Step Action

- 1 Tournament Organizer requests to collect fees.
- 2 The fees of sponsors and registration is calculated .
- 3 The cost estimation is calculated .
- 4 The Total Profit is calculated
- 5 Tournament Organizer reviews all the profit , cost estimation and fees that are paid via Registration fees or sponsors
- 6 The total fees are saved in tournament history

# Use Case Name: Publish Knockout Schedule

## Related Requirements:

### Goal in Context:

Tournament Organizer publish the knockout bracket for elimination rounds (e.g., round of 16, quarterfinals, semifinals, final).

### Preconditions:

- The qualifications for the knockout phase are completed (groups, prior rounds, or seeding rules).
- The set of teams advancing to the knockout phase is known.
- Venues and time windows are configured.

### Successful End Condition:

A complete knockout bracket with scheduled matches is stored and ready to be published.

### Failed End Condition:

Knockout generation fails due to missing qualified teams or scheduling conflicts.

### Primary Actor:

Tournament Organizer

### Secondary Actors:

### Trigger:

Tournament Organizer selects *publish Knockout Schedule* for the desired knockout stage.

### Main Flow

#### Step Action

- 1 Tournament Organizer initiates knockout schedule generation.
- 2 The system retrieves the list of qualified teams and seeding rules.  
The knockout bracket (pairings, home/away logic if applicable) schedule is created .
- 3 Include:: Check schedule  
Tournament Organizer reviews the generated knockout bracket and schedule.
- 5 The knockout schedule is saved.

**Step Action**

6 The schedule is published

# Use Case Name: Publish League Schedule

## Related Requirements:

## Goal in Context:

Tournament Organizer publish the league schedule

## Preconditions:

- The set of teams are fully registered .
- Venues and time windows are configured.

## Successful End Condition:

A complete league schedule is stored and ready to be published.

## Failed End Condition:

league schedule fails due to missing qualified teams or scheduling conflicts.

## Primary Actor:

Tournament Organizer

## Secondary Actors:

## Trigger:

Tournament Organizer selects *publish* league schedule for the league cup .

## Main Flow

### Step Action

- 1 Tournament Organizer initiates league schedule generation.
- 2 The system retrieves the list of teams.  
The league schedule is created .
- 3 Include:: Check schedule  
Tournament Organizer reviews the league schedule .
- 5 The league schedule is saved.
- 6 The schedule is published .

# Use Case Name: Assign Referees

## Related Requirements:

## Goal in Context:

Tournament Organizer assigns one or more referees to each scheduled match.

## Preconditions:

- Matches have been generated and stored in the schedule.
- Referees are registered and available in the system.

## Successful End Condition:

Each selected match has assigned referees stored in the system.

## Failed End Condition:

Assignment fails because no referees are available for the selected time or due to system errors.

## Primary Actor:

Tournament Organizer

## Secondary Actors:

## Trigger:

Tournament Organizer opens the match assignment screen and chooses *Assign Referees*.

## Main Flow

### Step Action

- 1 Tournament Organizer opens the list of scheduled matches.
- 2 Tournament Organizer selects a specific match for referee assignment.
- 3 The list of available referees are displayed
- 4 Tournament Organizer selects one or more referees and confirms the assignment.
- 6 System saves the referee assignments linked to the match.
- 7 The system notifies the assigned referees through their profiles or communication channel.



# Use Case Name: Enter Time slots

## Related Requirements:

## Goal in Context:

Team manager enter the perfect time slot of each team

## Preconditions:

- The team is approved

## Successful End Condition:

The time slots are saved in system to make schedules

## Failed End Condition:

The time slots are failed to be saved

## Primary Actor:

Team Manager

## Secondary Actors:

## Trigger:

Team Manager chooses to enter the time slots that are suitable for team

## Main Flow

### Step Action

- 1 Team manager chooses to enter the time slot
- 2 The time slots are verified
- 4 The time slots are saved .  
  
The schedule is ready to be generated.
- 5 Extensions
  - 2.1 the time slots are failed to be verified
  - 2.2 the team manager enters another time slots

# Use Case Name: View schedule

## Related Requirements:

## Goal in Context:

views generated schedule

## Preconditions:

- Tournament schedule have been generated and published.

## Successful End Condition:

The generated schedule is approved by all teams.

## Failed End Condition:

An issue has been found in schedule by a team

## Primary Actor:

Team manager

## Secondary Actors:

## Trigger:

Team manager navigates to view schedule

## Main Flow

### Step Action

- 1 Team manager opens the Tournament schedule .
- 2 Team manager and team views the schedule