



RU's e-Sport Extravaganza

Software for e-Sport event

Verklegt námskeið 1 - T-113-VLN1 - Haustönn 2025

Hópur 1

Andri Már Kristjánsson, Elmar Sigmarsdóttir, Ísak Elí Hauksson,
Kristinn Hrafn Daníelsson, Kristján Hagalín Jónsson og Sindri Freysson

December 12, 2025

Contents

1	Intro	2
2	Requirement List	3
2.1	Functional Requirements	3
2.2	Non-functional Requirements	4
3	User Group Analysis	5
4	Use Cases	6
5	Class Diagrams	15
6	UI-Design	18
6.1	UI-Tree	18
6.2	Tasks	18
6.3	Happy path	18
7	conclusion	26
8	Appendix	27
8.1	Archives	27
8.2	Publish Tournament	27
8.3	Search Bar	27
8.4	Logic to use for multiple instances	27
8.5	States (For diagrams)	27
8.6	Player/Team Notes	27

1 Intro

The goal of this project is to propose, interrogate, design, create a bookkeeping system for hosting and managing an e-sport tournament event, followed by user testing of the software. The software allows organizers to create and manage tournaments and input match results.

2 Requirement List

2.1 Functional Requirements

Nr	Functional requirements	User group	Priority	Completed
1	The user can login as organizer	Organizer	A	✓
2	The user can register as a player	Player	A	✓
3	The user can login as a player	Player	A	✓
4	The user can enter as a spectator without signing in	General	A	✓
5	Organizer can create a tournament	Organizer	A	✓
6	Organizer can add a team to a tournament	Organizer	A	✓
7	When a team captain leaves a team, he assigns a team member to be the new captain	Team Captain	A	✓
8	The Organizer archives tournaments when they finish <i>(Outdated: The System does it for you)</i>	Organizer	A	✓
9	Organizer can publish a tournament	Organizer	A	✓
10	Organizer can input match results	Organizer	A	✓
11	Player can create a team	Player	A	✓
12	Player becomes team captain	Player	A	✓
13	Player can leave a team	Player	A	✓
14	Team captain can add a player to his team	Team Captain	A	✓
15	Team captain can remove a player from his team	Team Captain	A	✓
16	Team captain can leave his team	Team Captain	A	✓
17	The user can view tournament	General	A	✓
18	The user can view tournament schedule	General	A	✓
19	The user can view all teams	General	A	✓
20	The user can view all players	General	A	✓
21	Organizer can manage tournament	Organizer	A	✓
22	Organizer can create a club	Organizer	B	✓
23	When the organizer creates a tournament, he can choose between double elimination and normal	Organizer	B	Dropped
24	Teams are assigned to a club on creation	Team Captain	B	✓
25	The user can view all clubs	Player	B	✓
26	Organizer can modify time of a tournament before it has started	Organizer	B	Dropped
27	Organizer can delete a tournament before it has been published	Organizer	C	Dropped
28	Organizer can edit clubs	Organizer	C	Dropped
29	Team captains can transfer his authority to a player in the team	Team Captain	C	Dropped
30	Team captain can change what club the team is associated with	Team Captain	C	Dropped
31	Organizer can cancel an ongoing tournament	Organizer	C	Dropped
32	Team captain can add ascii art to his team	Team Captain	C	✓
33	Organizer can edit specific match scheduling	Organizer	C	Dropped
34	The user can type in shortcuts for windows to travel to specific window	General	C	Dropped

2.2 Non-functional Requirements

Nr	Non-Functional requirements	User group	Priority	Completed
1	System automatically shuffles and arranges teams in a tournament match, based on quantity of teams and time frame, and server count	System	A	✓
2	When team captain leaves a team if he is alone then the team is archived	System	A	✓
3	Each Player can only be in one team at the same time	System	A	✓
4	The system should not crash due to invalid data by the user	System	A	✓
5	The player personal data should not be visible to the spectator	System	A	✓
6	Only the team captain can add team members the team	System	A	✓
7	Only the organizer can create tournaments	System	A	✓
8	Code must follow Python type hinting	System	A	✓
9	Code must follow snake case naming convention	System	A	✓
10	Doc-strings in code should be sphinx-style (<i>We did not write down raise errors</i>)	System	A	WIP
11	Name and Username should be the lenght of 3-40	System	A	✓
12	Tournament have must have at least 2 teams	System	A	✓
13	Player, Team, Tournament, Servers, Matches, Clubs generate UUID when model classes are created	System	A	✓
14	The user can see a breadcrumb navigation	System	B	✓
15	Only the organizer can modify clubs	System	B	Dropped
16	Give users options to cancel while inputting information	System	B	✓
17	Tournament can have more than 2 teams	System	B	✓
18	The Club Name will be printed in its color	System	C	Dropped
19	Show tournaments brackets visually	System	C	Dropped

3 User Group Analysis

Table 1:

User group	Organizers	Team Captains	Player	Spectators
Background	Age: 19+	Age: 19+	Age: 19+	Age: Any
	Gender: Any	Gender: Any	Gender: Any	Gender: Any
	Education: Graduated from high school	Education: Graduated from high school	Education: Graduated from high school	Education: Graduated from high school
	Abilities: Nothing special	Abilities: Nothing special	Abilities: Nothing special	Abilities: Nothing special
	Computer skills: Very good	Computer skills: Very good	Computer skills: Very good	Computer skills: Various, depends on age and education
	Number: 1	Number: ca. 400	Number: ca. 1750	Number: ca. 1750
User goal	Plan, coordinate and manage a competitive tournament information	To manage a tournament team, enter team information and see tournament information	To enter and see information related to themselves and tournament	To see and keep up with tournament information
Equipment	A good computer with an internet connection	A good computer with an internet connection	A good computer with an internet connection	A good computer with an internet connection
Environment	The physical location is good both at Reykjavík University or at another location with an internet connection	The physical environment can be good both at home and at Reykjavík University	The physical environment is usually good wherever, as long as there is an internet connection available	The physical environment is usually good wherever, as long as there is an internet connection available
Usage	How often: Before and during every tournament	How often: Before every tournament	How often: After tournaments	How often: During and after tournaments
	For how long each time: The system is used for a couple of hours at a time but it can vary depending on the duration of the tournament	For how long each time: The system is used for about 30 minutes each time	For how long each time: The system is used for about 10 minutes each time	For how long each time: The system is used for about 10 minutes each time
	Skills: Very skilled, since they take care of setting everything up with the use of the system and should be familiar with it to do so	Skills: Skilled, since they take care of entering crucial information and should at least be familiar with the system in order to do so	Skills: Mediocre, since they only need to take care of their own personal information	Skills: Not very skilled, since they only use the system to see tournament information
Important	Most important + Team captains	Most important + Organizers	Second most important	Least important

4 Use Cases

User logs in as organizer	
Number	1
Priority	A
Precondition	Program is running
Base flow	1. User selects log in
	2. Inputs the details for the Organizer • Organizer username
Alternative flow	
Post condition	User is now an Organizer and can create and manage tournaments.
Actors	User/Organizer
Author	Elmar Sigmarsdóttir
User registers as player	
Number	2
Priority	A
Precondition	Program is running
Base flow	1. User selects Sign up
	2. User inputs his information • Full name • Date of birth • Address • Phone • Email • Link (optional) • Handle
	User is now registered as a player
Alternative flow	• User tries to sign up with an existing account. User needs to login to his account. See use case(3)
Post condition	User is now a player and cannot be registered as a player again.
Actors	User/Player
Author	Elmar Sigmarsdóttir
User registers as player	
Number	3
Priority	A
Precondition	Program is running
Base flow	1. Player selects log in
	2. Player inputs his Handle
	3. Player is now logged in
Alternative flow	• User doesn't have a player account. User needs to sign up as a player. Use case(2)
Post condition	The player is now logged in and can access information about himself
Actors	Player
Author	Elmar Sigmarsdóttir

User enters as spectator	
Number	4
Priority	A
Precondition	Program is running
Base flow	<ol style="list-style-type: none"> 1. User selects the Spectator option 2. Now the user is a spectator
Alternative flow	
Post condition	The user is now a spectator and can view details from the tournament, (schedule, match results, clubs, teams and players)
Actors	User
Author	Elmar Sigmarsen

Organizer can create a tournament	
Number	5
Priority	A
Precondition	Logged in as organizer, use case(1)
Base flow	<ol style="list-style-type: none"> 1. Organizer selects Create Tournament 2. Organizer inputs tournament information <ul style="list-style-type: none"> • Unique name • Start date • End date • Start time frame • End time frame • Max servers • Venue • Contact email • Contact number 3. User is now registered as a player
Alternative flow	
Post condition	Organizer has now created and has access to a new unique tournament
Actors	Organizer
Author	Elmar Sigmarsen

Organizer adds team to tournament	
Number	6
Priority	A
Precondition	Organizer has already created a tournament, use case (5)
Base flow	<ol style="list-style-type: none"> 1. Organizer selects tournament 2. Organizer selects View Teams. 3. Organizer selects add team 4. Organizer inputs team name
Alternative flow	<ul style="list-style-type: none"> • No team has been made, a player needs to make a team, use case (10) • The team doesn't have enough players, the team captain needs to add players to his team, use case (11)
Post condition	That team is now apart of the tournament
Actors	Organizer
Author	Elmar Sigmarsen

Organizer removes a team from a tournament	
Number	7
Priority	A
Precondition	A team has to have been added to the Tournament, use case (6)
Base flow	1. Organizer selects tournament
	2. Organizer selects view team
	3. Organizer selects remove team
	4. Organizer inputs team name
Alternative flow	
Post condition	That team is no longer apart of the tournament
Actors	Organizer
Author	Elmar Sigmarsen

Organizer starts a tournament	
Number	8
Priority	A
Precondition	The tournament needs to have enough teams.
Base flow	1. Organizer is logged in, case (1)
	2. Organizer selects manage tournaments
	3. Organizer inputs a tournament name
	4. Organizer is now able managing a tournament
Alternative flow	• Tournament doesn't have enough teams in it, organizer needs to add teams to the tournament, use case(6).
Post condition	The organizer is now viewing the tournament and its info and can edit/manage it (publish, manage teams in the tournament, change name/date)
Actors	Organizer
Author	Elmar Sigmarsen

Organizer can publish a tournament	
Number	9
Priority	A
Precondition	The tournament needs to have 16 teams
Base flow	1. Organizer selects a tournament to manage, use case (8)
	2. Organizer selects publish tournament
	3. And confirms
Alternative flow	• Tournament doesn't have enough teams in it, organizer needs to add teams to tournament, use case (6)
Post condition	Needs post condition!!!
Actors	Organizer
Author	Elmar Sigmarsen

Organizer inputs result of a match	
Number	10
Priority	A
Precondition	A tournament needs to have been published
Base flow	1. Organizer selects a tournament to manage, use case (8)
	2. Organizer selects input match results
	3. Organizer gets a list of all matches and selects matches to input results
	4. Organizer is asked who of the two teams won the match and inputs the team name
	5. Player is now team captain of the newly created team.
Alternative flow	
Post condition	Results of a specific match have been made so the winner of that match goes to the next round
Actors	Organizer
Author	Elmar Sigmarsdóttir

Player creates a team	
Number	11
Priority	A
Precondition	
Base flow	1. Player logs in, use case (3)
	2. Captain selects Add player to team.
	3. Player inputs team name
	4. Player inputs Club (no input = no club)
	5. (Now the player is the team captain of an empty team)
Alternative flow	• Player is already a part of a team. Player needs to leave his existing team, see case (14) and then create a team
Post condition	Player is now a team captain of a team
Actors	Player
Author	Elmar Sigmarsdóttir

Team captain adds a player to their team	
Number	12
Priority	A
Precondition	Player must have created a team, use case(10)
Base flow	1. Captain logs in, use case (3)
	2. Captain selects My Team
	3. Captain selects Add Player to Team
	4. Captain inputs Players Handle
	5. That Player is now a part of team
Alternative flow	• A player is already in another team. That player needs to leave his team, see case (14), then the Captain can add that player to his team
Post condition	Team captain has now added a player to his team
Actors	Team captain, Player
Author	Elmar Sigmarsdóttir

Team captain removes a player from their team	
Number	13
Priority	A
Precondition	Team captain must have added a player to their team, use case(11)
Base flow	1. Captain logs in, use case (3)
	2. Captain selects view team
	3. Captain selects remove player
	4. Captain inputs Player's handle
	5. Player has been removed from the team
Alternative flow	• That player removes himself from the team, use case (14)
Post condition	Team no longer has That Player in their team
Actors	Team captain, Player
Author	Elmar Sigmarsdóttir

Player leaves their team	
Number	14
Priority	A
Precondition	Player needs to be in a team.
Base flow	1. Player logs in, use case (3)
	2. Player selects view team
	3. Player selects leave team
	4. (Error message/ confirmation)
	5. Player is no longer a part of any team
Alternative flow	• Team captain removes player from team, case (12)
Post condition	The player is no longer a part of any team and can now be picked onto a new team or create his own
Actors	Player
Author	Elmar Sigmarsdóttir

Team captain leaving his team	
Number	15
Priority	A
Precondition	Player needs to be a team captain.
Base flow	1. Captain logs in, use case (3)
	2. Captain selects view team
	3. Captain selects leave team
	4. (Error message/ are you sure)
	5. Captain selects a Player from team to become new Captain
Alternative flow	• If the team the Captain is leaving has no other players the team is archived, but still has its statistics (wins, etc.)
Post condition	The Captain is no longer a part of the team and is now has the status of just Player (not in team), and the player he selected has the status team Captain of that team
Actors	Captain, player
Author	Elmar Sigmarsdóttir

Any user can see tournaments	
Number	16
Priority	A
Precondition	
Base flow	1. User selects Tournaments
	2. User selects which tournament
	3. User is now viewing a specific tournament
Alternative flow	<ul style="list-style-type: none"> • There are no tournaments available, the organizer needs to create a tournament, use case (5) • A tournament is not published, so the organizer needs to start the tournaments, use case (9)
Post condition	User can now look through that tournament
Actors	Any user
Author	Elmar Sigmarsdóttir

Any user can see tournament schedule	
Number	17
Priority	A
Precondition	
Base flow	1. User selects a tournament
	2. User selects Schedule
	3. User is now viewing the schedule of a tournament
Alternative flow	
Post condition	User can view the schedule of the next round of that tournament (or whole tournament, date)
Actors	Any user
Author	Elmar Sigmarsdóttir

Any user can view teams	
Number	18
Priority	A
Precondition	
Base flow	1. User selects teams (from main menu)
	2. User is viewing a list of all teams
Alternative flow	<ul style="list-style-type: none"> • User can select a tournament see use case (XXX), and in that menu select teams and then the user is viewing only the teams in that tournament
Post condition	The user is viewing a list of teams that he can select from see that teams information (players, statistics)
Actors	Any user
Author	Elmar Sigmarsdóttir

Any user can view players	
Number	19
Priority	A
Precondition	
Base flow	1. User selects players (in main menu)
	2. User is viewing a list of all registered players
Alternative flow	<ul style="list-style-type: none"> • User can select a tournament see use case (XXX), and in that menu select players and then the user is viewing only the players in that tournament • User selects a team, use case (XX), and from the team menu the user can view a list of that teams player
Post condition	The user is viewing list of players that he can select from to see that players information (handle, statistics)
Actors	Any user
Author	Elmar Sigmarsdóttir

Organizer can create clubs	
Number	20
Priority	B
Precondition	
Base flow	<ol style="list-style-type: none"> 1. Organizer selects View Clubs. 2. Organizer selects add club. 3. Organizer inputs information. <ul style="list-style-type: none"> • Name • Club color • Hometown • Country
Alternative flow	
Post condition	Organizer has added a new club.
Actors	Organizer
Author	Elmar Sigmarsdóttir

Organizer can choose type of tournament	
Number	21
Priority	B
Precondition	
Base flow	<ol style="list-style-type: none"> 1. While Organizer is creating a tournament, use case (5) 2. Now there is an extra section which asks the organizer what type of tournament it is (knockout or double elimination)
Alternative flow	
Post condition	Organizer has added selected what type of tournament
Actors	Organizer
Author	Elmar Sigmarsdóttir

Organizer can edit time frame of tournament	
Number	22
Priority	B
Precondition	Tournament has to be created and NOT started
Base flow	<ol style="list-style-type: none"> 1. Organizer selects a tournament to manage, use case (8) 2. Organizer selects Edit Time Frame 3. Organizer inputs new: <ul style="list-style-type: none"> • Start date • End date 4. Organizer selects Edit Time Frame
Alternative flow	
Post condition	The Time frame has been updated for that tournament
Actors	Organizer
Author	Elmar Sigmarsdóttir

Organizer can delete a tournament	
Number	23
Priority	B
Precondition	A tournament can NOT be published
Base flow	<ol style="list-style-type: none"> 1. Organizer selects a tournament to manage, use case (8) 2. Organizer selects Delete 3. Organizer confirms deletion
Alternative flow	
Post condition	Now the tournament has been deleted and no records of it are left
Actors	Organizer
Author	Elmar Sigmarsdóttir

Player can edits their information	
Number	24
Priority	B
Precondition	
Base flow	1. Player logs in, use case (3)
	2. Player selects Edit Information
	3. Player selects the info he wants to edit (all info)
	4. Player has changed his info
	5. Player has changed their information.
Alternative flow	
Post condition	Player has updated their information.
Actors	Player
Author	Elmar Sigmarsen

Team Captain can make team Player a Captain	
Number	25
Priority	B
Precondition	
Base flow	1. Player logs in, use case (3)
	2. Captain selects My Team
	3. Captain selects transfer authority
	4. Captain inputs players' name to transfer authority to
Alternative flow	
Post condition	Now that Player is the Team Captain, and the old Team Captain is just a Player on the team
Actors	Captain, Player
Author	Elmar Sigmarsen

Any user can view clubs	
Number	26
Priority	B
Precondition	
Base flow	1. User selects spectator
	2. User selects clubs
	User is now viewing a list of clubs
Alternative flow	
Post condition	User can view all the clubs created, and can select any club to see that clubs teams
Actors	Any user
Author	Elmar Sigmarsen

Organizer can cancel a tournament	
Number	27
Priority	C
Precondition	A tournament has to have been published
Base flow	1. Organizer selects a tournament
	2. Organizer selects Cancel tournament
	3. Tournament is now canceled
Alternative flow	
Post condition	The tournament is cancelled, all records are not kept, the statistics for the teams is reverted to the state before the tournament started
Actors	Organizer
Author	Elmar Sigmarsen

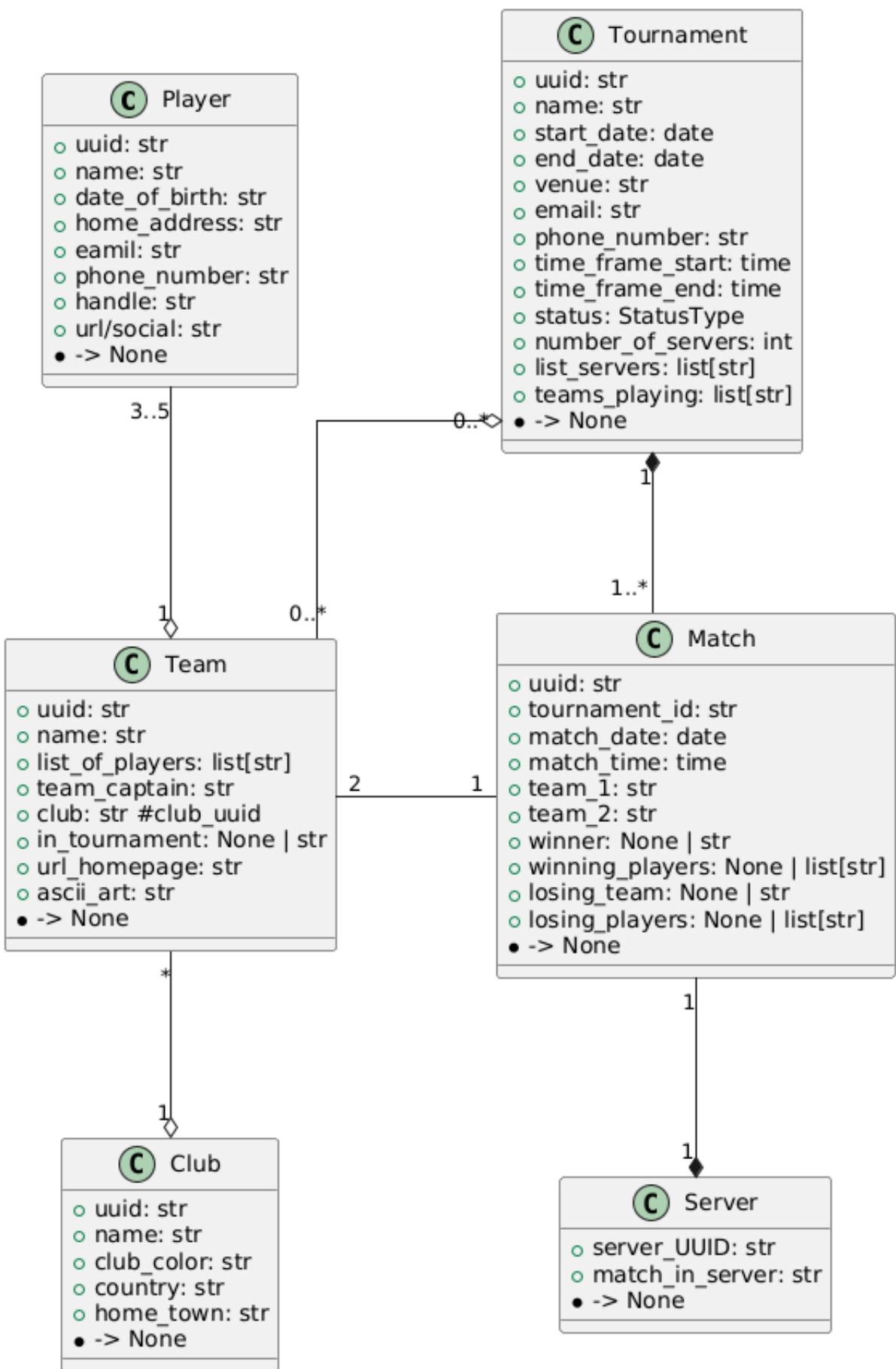
Team captain can edit team name	
Number	28
Priority	C
Precondition	
Base flow	<ol style="list-style-type: none"> 1. Captain is logged in, and selects View Team 2. From View Team menu Captain selects Change Team Name 3. Captain inputs the new team name (has to be unique)
Alternative flow	
Post condition	Now that teams name has a new name, and new or other teams cannot use that name, but can use the previous team name
Actors	Team Captain
Author	Elmar Sigmarsdóttir

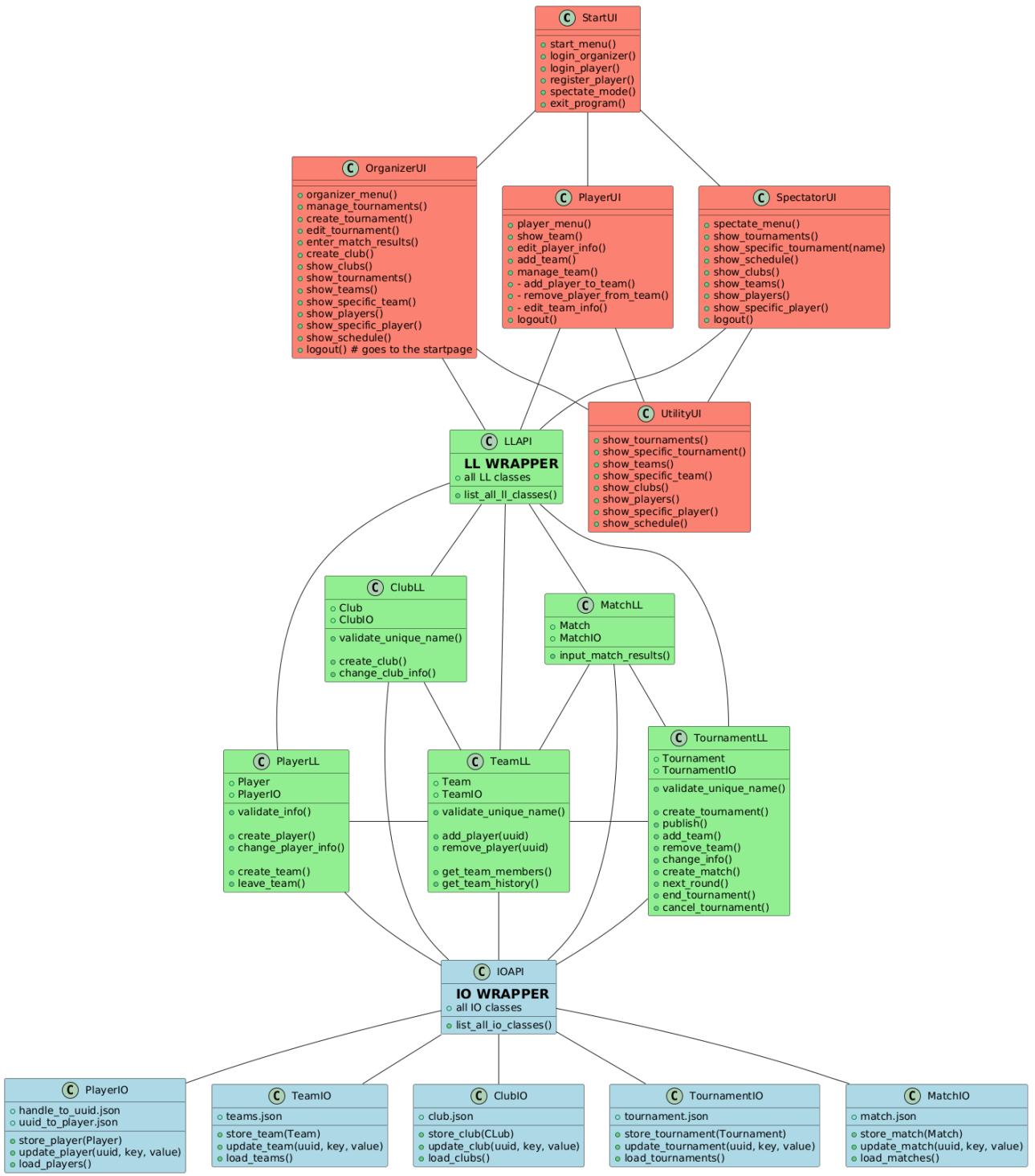
Team captain can change the teams club	
Number	29
Priority	C
Precondition	
Base flow	<ol style="list-style-type: none"> 1. Team Captain is logged in, and selects View Team 2. From View Team menu Captain selects Change Club 3. Captain has a list of the clubs and can select what his teams new club is
Alternative flow	
Post condition	Now the Captains team is in a new club, so the team is listed in the their new club and no longer in the old club
Actors	Team Captain
Author	Elmar Sigmarsdóttir

Team Captain can add ASCII art to his team	
Number	30
Priority	C
Precondition	
Base flow	<ol style="list-style-type: none"> 1. Captain is logged in, and selects View Team 2. Captain selects add ASCII art 3. Captain inputs the ASCII art
Alternative flow	
Post condition	Now the team has ASCII art that can be viewed from any user viewing the team
Actors	Team Captain
Author	Elmar Sigmarsdóttir

5 Class Diagrams

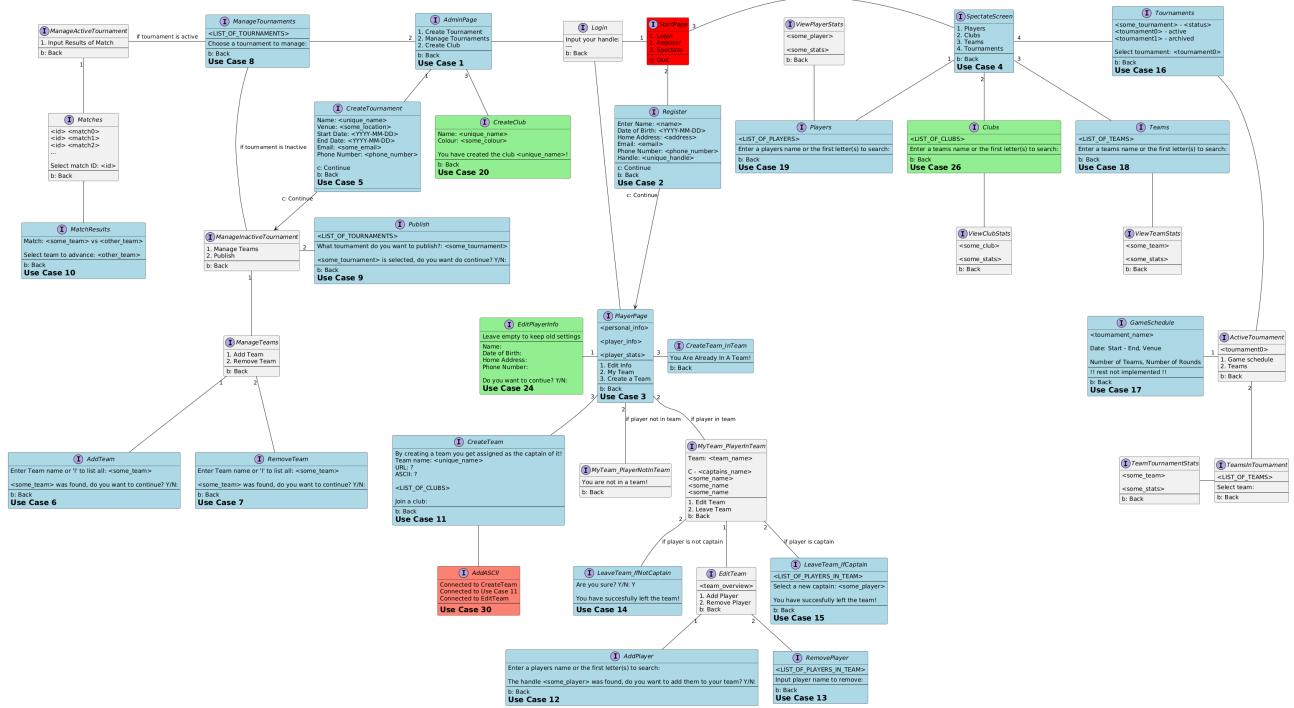
Class diagrams are the main building block in object-oriented modeling. They are used to show the different objects in a system, their attributes, their operations and the relationship between them. The two class diagrams are with model classes and 3 tier design classes. Model classes are a class (bucket) that is just the data, with few functions. While the 3-tier design splits the classes into the user interaction layer, logic layer and data layer, which are connected by so called wrappers that make it so that each layer doesn't impact the rest of the layers to make it easier refactor or change classes or functions without the other layers getting affected. The diagrams below show model classes, 3-tier design and model classes combined with 3-tier design.





6 UI-Design

6.1 UI-Tree



6.2 Tasks

Player / Captain

1. You want to create a account for a player named Miss Piggy, handle: BaconBits, date of birth: june 14th 1938, home address: 44 Bronson Lane, phone number: 1234567, email: mspiggy@gmail.com, socials: none
2. Now that you are a player, you want to create a team named TheMuppets in the club Disney
3. Now that you have created a team you want to add another player called GreenCroaker to your team

Admin

1. You want to log in as an admin (the handle is just “admin”) and create a tournament named MuppetsVsDocHopper that will take place in Sesame Street, it will start 27th of november 2025 and end the 29th of november 2025, the contact info is the gmail: something@something.com, and phone number: 1234567.
2. Now you want to publish the tournament, but you first have to add the TheMuppets team, DocHopper team to the tournament and 14 other teams
3. After publishing the tournament, you start the tournament. The system creates matches between two unique teams.
4. You input the results of the matches, with TheMuppets winning every match. This happens until they win the tournament.

Spectator

1. You want to log in as a spectator and go to view the results of the MuppetsVsDocHopper tournament.
2. You want to see the information on TheMuppets team and see how many games they have won in the past.

6.3 Happy path

Happy path 1 “You want to register yourself as a player”

1. User starts the program and chooses option “2” to register themselves as a new user

Main Menu

1) Log in
2) Register
3) Spectate

please enter an option:

| 2

2. User inputs all necessary information they are prompted to give

Player creation

please enter name:
Miss Piggy

please enter birthdate (yyyy mm dd):
1938 06 14

please enter address:
Muppetstreet 4

please enter phone number:
1234567

please enter handle:
Bacon Bits

please enter e-mail:
mspiggym@gmail.com

3. User is taken to their player menu where they can see their information

Player Page

Name: Miss Piggy
Birthdate: 1938 06 14
Address: Muppetstreet 4
Phone number: 1234567
E-Mail: mspiggym@gmail.com

Team: None
Club: None
Handle: BaconBits

Wins: 0
Points: 0

1) Edit info
2) My team
3) Create a team
b) Back

please enter an option:

|

4.

Happy Path 2 "Now that you are a player, you want to create a team named TheMuppets in the club Disney"

1. User starts the program and chooses option "1" to log in

Main Menu

1) Log in
2) Register
3) Spectate

Please enter an option:

| 1

2. User inputs their user handle

Log in

Input your handle:

BaconBits

3. User is taken to their player menu where they can see their information and chooses option 3 to create a new team

Player Page

Name: Miss Piggy
Birthdate: 1938 06 14
Address: Muppetstreet 4
Phone number: 1234567
Team: None
Club: None
Handle: BaconBits
E-Mail: msPiggy@gmail.com
Wins: 0
Points: 0

1) Edit info
2) My team
3) Create a team
b) Back

Please enter an option:

| 3

4. User is made aware that they will be made captain of the team they are creating and choose "c" to continue

By creating a team you are assigned as the captain of it!

c) Continue
b) Back

Please enter an option:

| c

5. User is prompted to give information and inputs "TheMuppets" as a name and leaves the rest blank since it is not needed

6. User then gets a list of all available clubs and chooses the "Disney" club to join it

<u>Team Creation</u>
Enter team name:
The Muppets
Enter URL for the team (optional):
Enter ASCII art (optional):
<u>Available clubs</u>
xxxXX
xxXxx
Disney
ClubName
Enter a Club to join:
Disney

Happy Path 3 "Now that you have created a team, you want to add another player called GreenCroaker to your team"

1. Player is already loged in as a player and selects

<u>Player Page</u>
Name: Miss Piggy
Birthdate: 1938 06 14
Address: Muppetstreet 4
Phone number: 1234567
Team: None
Club: None
Handle: BaconBits
E-Mail: mspiggy@gmail.com
Wins: 0
Points: 0
1) Edit info
2) My team
3) Create a team
b) Back
please enter an option:
1 2

2. User inputs "1" to edit their team

<u>My Team</u>
Team: The Muppets
C - BaconBits
BlueBignose
SmilerBear
1) Edit team
2) Leave team
b) Back
1

3. User inputs "1" again to add a player to their team

<u>Edit team</u>
1) Add player
2) Remove player
b) Back
1

- User inputs the handle of the player they want to add

Add Player
Enter a players name or the first letter(s) to search:
Green Croaker

- The player is found and the user inputs "Y" to confirm that they want to add them to their team

Add Player
Enter a players name or the first letter(s) to search:
Green Croaker
The handle GreenCroaker was found do you want to add them to your team Y/N:
Y

Happy Path 4 "You want to log in as an admin (the handle is just "admin") and create a tournament named MuppetsVsDocHopper

- User inputs "1" to log in

Main Menu
1) Log in
2) Register
3) Spectate
please enter an option:
1

- User inputs "Admin" in order to log in as an admin

Log in
Input your handle:
Admin

- User is met with the admin options and inputs "1" to create a tournament

Admin Menu
1) Create Tournament
2) Manage Tournaments
3) Create Club
b) Back
please enter an option:
1

- User is prompted for the needed info in order to be able to create a tournament and inputs the information needed

Create tournament
Enter tournament name:
MuppetsVsDocHopper
Enter venue name:
Sesame Street
Enter start date
2025 11 27
Enter end date
2025 11 29
Enter contact E-mail:
Something@Something.com
Enter contact phone number:
1234567

Happy Path 5 "Now you want to start the tournament, but you first have to add the TheMuppets team and the DocHopper team to the tournament"

- User goes back to the admin menu and inputs "2" in prder to manage their tournaments

Admin Menu

1) Create Tournament
2) Manage Tournaments
3) Create Club
b) Back

Please enter an option:

1 2

- User is met with a list of all their tournaments and if they are active or inactive and user inputs the name of the tournament they want to manage

Manage Tournaments

Tournament name 1, active
X X X X X X X X, unactive
MuppetsVs Doc Hopper, unactive
Choose a tournament to edit!

MuppetsVs Doc Hopper

- User inputs "1" to manage the teams in the tournament

Manage Teams

1) Add Team
2) Remove team
b) Back

1

- User gets prompted to add a teamname to add to the tournament, the program confirms that the team was found and user inputs "y" to add it

Add Team

Enter team name or "l" to list all teams:

TheMuppets

Add Team

Enter team name or "l" to list all teams:

TheMuppets

The Muppets was found, do you want to add it? Y/N

Y

- User re-does the previous 2 steps but inputs another teamname

Add Team

Enter team name or "l" to list all teams:

DocHopper

Add Team

Enter team name or "l" to list all teams:

DocHopper

Doc Hopper was found, do you want to add it? Y/N

Y

- User goes back to the tournament manage menu and inputs "2" to publish it

Manage Unactive Tournament

1) Manage teams
2) Publish
3) Edit Tournament
b) Back

2

- User inputs the name of the tournament they want to publish and confirm to publish it

Publish

What tournament do you want to publish?

TheMuppetsVs Doc Hopper

Publish

What tournament do you want to publish?

TheMuppetsVs Doc Hopper

TheMuppetsVs Doc Hopper was selected
do you want to continue? Y/N

Y

Publish
What tournament do you want to publish?
<u>TheMuppetsVs DocHopper</u>
TheMuppetsVs DocHopper was selected
do you want to continue? Y/N:
<u>Y</u>
TheMuppetsVs DocHopper has been published

Happy Path 6 "Now that all the rounds have finished except for the last round of the tournament, you want to input the results that TheMuppets won the tournament"

1. User goes back to the manage tournament menu and inputs the name of the tournament they want to input the results to

Manage Tournaments
tournament name 1, active
X X X X X X X X, active
<u>MuppetsVs Doc Hopper, inactive</u>
Choose a tournament to edit:
<u>MuppetsVs Doc Hopper</u>

2. Since the tournament is now active the user gets a new menu and inputs "1" to input results

Manage active tournament
1) input results of match
2) Next round
3) Cancel tournament
b) Back
please enter an option:
<u>1</u>

3. User is met with a list of all matches left in the tournament and inputs "1" to select match number 16

Match results
Match Nr Team 1 Team 2
Match 30 TheMuppets DocHopper
1) Match 30
b) Back
please enter an option:
<u>1</u>

4. User inputs the name of the winning team for that round

Select a winner
Match 30 TheMuppets DocHopper
<u>TheMuppets</u>

5. Since all the matches of the tournament have finished, the program prints out who won the tournament

Tournament results
TheMuppets won the tournament!
b) Back

Happy Path 7 "You want to log in as a spectator and go to view the results of the MuppetsVsDocHopper tournament"

1. User inputs "3" in the main menu to spectate

Main Menu
1) Log in
2) Register
3) Spectate
please enter an option:
<u>1 3</u>

2. User is taken to the spectators page and inputs "4" to see the tournament history

Spectate Page

1) Game schedule
2) Clubs
3) Teams
4) Tournament History
b) Back

please enter an option:

| 4

3. User is met with a list of previous tournaments and can see next to each tournament who the winner was

Tournament History

xxxxx, Winner: x
FinishedTournament, Winner: Bla
Muppets Vs Doc Hopper, Winner: TheMuppets
AAAAA, Winner: B
BIGCHUNGUS, Winner: WhatstheDoc

b) Back

Happy Path 8 "You want to see the information on TheMuppets team and see how many games they have won in the past"

1. User inputs "3" on the main menu to spectate

Main Menu

1) Log in
2) Register
3) Spectate

please enter an option:

| 3

2. User is taken to the spectators page and inputs "3" to see team information

Spectate Page

1) Game schedule
2) Clubs
3) Teams
4) Tournament History
b) Back

please enter an option:

| 3

3. User is taken to a team lookup menu and inputs "2" to search for a specific team

Teams

1) See all teams
2) Search for team
b) Back

2

4. User inputs the desired team name and then the teamname, along with how many wins the team has gotten and the club it is printed for the user

Search for team

Enter a players name or the first letter(s) to search:
TheMuppets

TheMuppets Wins:1 Club:Disney

7 conclusion

8 Appendix

8.1 Archives

Tournament

- When the tournament is over then it is archived

Team

- The place that a team goes to when a team captain is alone in a team and leaves the team (AKA: IF a team becomes empty)
- Stores empty teams and their scores (Should be able to view it)

8.2 Publish Tournament

When managing the tournaments the Organizer has the option of publishing a tournament and by pressing that option, he will get a list of available tournaments (AKA: Tournaments that have 16 or more teams and tournaments that aren't already published). Then he will input what tournament he wants to publish and that tournament

8.3 Search Bar

- Can input the name of a player directly
- IF unsure about the player options, then should be able to list every name
- The ability to search for the first letters to narrow down the list

Example: Searching for available players for teams

Add a Teams

- Enter Team name or l for the list of all team or q to quit: abcd
 - abcd was found are you sure you want to continue
- Y Yes
- N No

Remove a Team

- Enter Team name or l for the list of all teams in the tournament or q to quit:
 - abcd is not a part of the tournament, do you want to continue
- Y Yes
- N No

8.4 Logic to use for multiple instances

8.5 States (For diagrams)

A team needs to have at least 3 players to join a tournament

A team can have a maximum of 5 players? If it has 5 players, the captain cannot add more.

A player can have 3 states (Captain, In Team, Not In Team)

- Captain: Can view team and add or remove players
- Player in team: Can view his team, and can leave his team
- Player NOT in team: Can create a team

If captain leaves his team there can be two outcomes

- Captain leaves and selects a new captain from the team
- If the team is empty the captain leaves but the team still exists and has all statistics

A player can NOT leave his team if the team is in a tournament

8.6 Player/Team Notes

Color: simple length check

- Venue: simple length check
- Email: needs to go through a check
- Phone number: 7 Numbers
- Address: (strings)(space)(number)

- When taking input we will have a validator in the logic layer that returns whether it is valid or not
- Valid player handle: Just about the length (3-12) NEED TO HAVE **ADMIN** in the handle file
- Date stored YYYY-MM-DD

- o Validation

All need to be numbers that are not the “-”

Year:

Month: 01 <= month <= 12

- If the C color requirement will be used then there should only be options for the user (Can only choose color out of few chosen colors)
- Randomize team list for tournaments
- At least 16 teams in a tournament
- Teams cannot Play in multiple matches at the same time
- The main interactions from the user is with shown steps like: 1) Login, 2) Sign Up, b) back ... something like that (can also have some commands)
- Everyone should be able to see all active Teams (in an active tournament)