

Requirements Definition

Introduction

We've been contracted to create management software for a mini-golf tournament. This document contains the requirements set by the client for the development of the project. We first discuss the functional and non-functional requirements for the project, and then talk about future development plans.

Actors (Functional):

Player requirements: 1.X

- 1.1 Players need to be able to sign up for the tournament.
- 1.2 Players need the ability to add money to their accounts.
- 1.3 Players add their score to the leaderboard when they finish the tournament.
- 1.4 Players can order drinks at any time during their playthrough.

DrinkMeister requirements: 2.X

- 2.1 DrinkMeisters receive orders for drinks in their app portal.
- 2.2 DrinkMeisters complete orders in the app, and send or deliver the drinks to the players.

Sponsor requirements: 3.X

- 3.1 Sponsors register to sponsor an event using the calendar in the Sponsor view of the application.

Manager requirements: 4.X

- 4.1 Managers create tournaments.
- 4.2 Managers are able to convert account types of users.
- 4.3 Managers receive all of the proceeds from the tournament.
- 4.4 Managers are able to modify the drink menu.

Non-Functional

Tournament System: 5.X

- 5.1 UI Interface to create & end tournaments
- 5.2 Define & display a leaderboard

User Database & Login System: 6.X

- 6.1 Sign up & log in from homepage
- 6.2 Dashboard specific to user account type

Drink Menu for Users: 7.X

- 7.1 List name of drink, price, and order button
- 7.2 Within user dashboard

Event Calendar: 8.X

- 8.1 Shows planned tournaments
- 8.1 Allows sponsors to create a request to sponsor a tournament

Compatibility: 9.X

- 9.1 Web application
- 9.2 Useable on mobile devices

Future Features - 10.X

- 10.1 A Sponsor sign-up option to bypass managers needing to convert *Player* accounts into *Sponsor* accounts.

Use Case Diagrams:

Participating Actor: Manager

Entry Conditions:

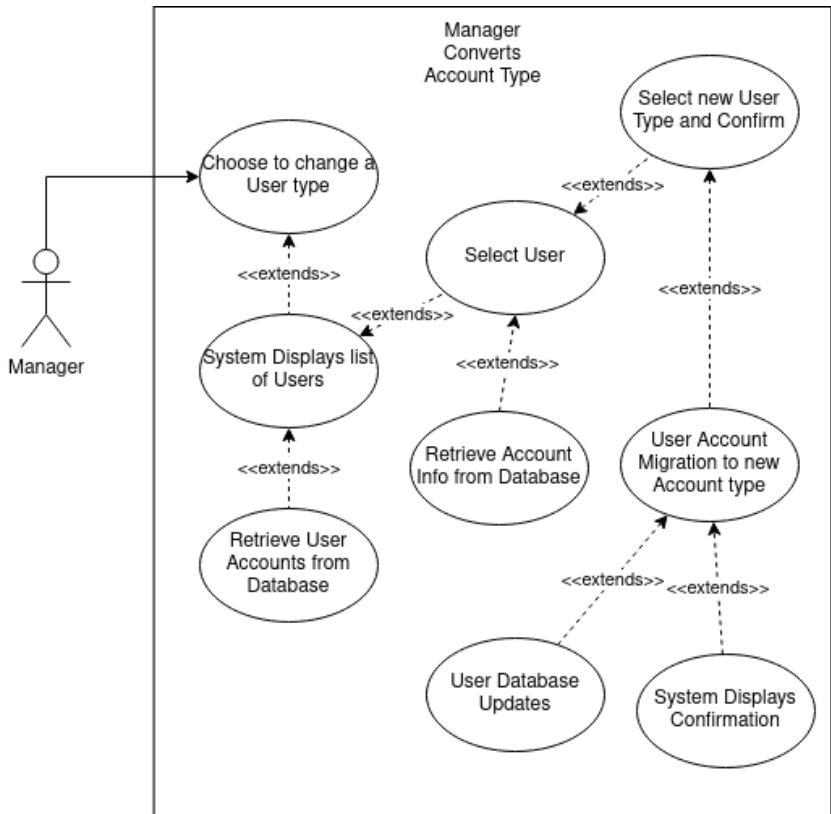
-Manager selects to convert account type.

Exit Conditions:

-Manager converts account successfully.
-Manager cancels account conversion.

Event flow:

1. Manager chooses to change a user type.
2. System displays list of users.
3. Manager selects user to convert.
4. Manager selects new user account type.
5. System converts user type and updates database.
6. System displays confirmation of change.

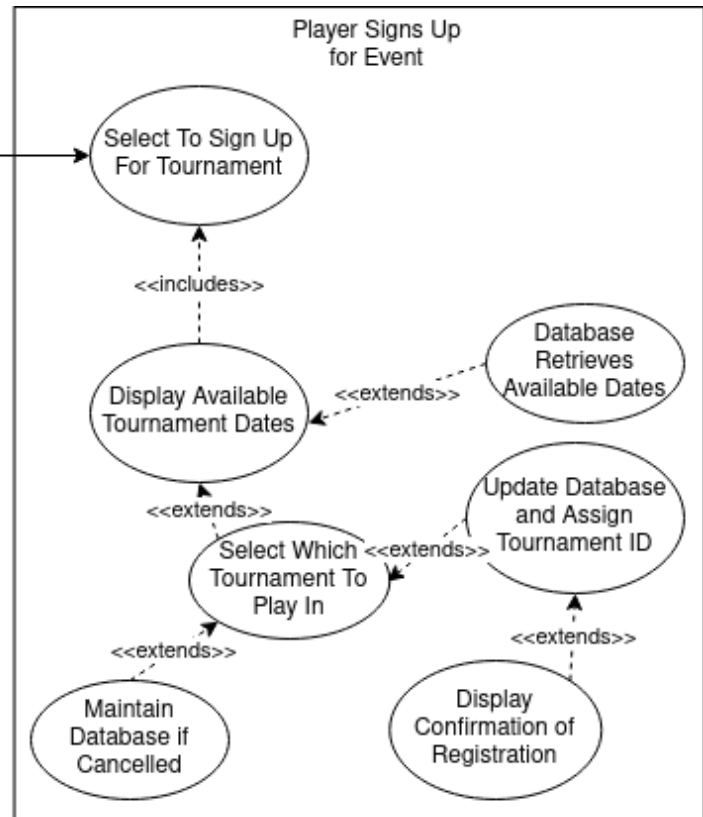
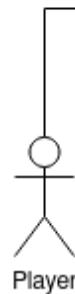


Participating Actor: Player

Entry Conditions:
-Player logs in

Exit Conditions:
-Player signs up
-Player cancels signing up

- Event flow:
1. Player chooses to sign up for a tournament
 2. System displays tournament days available to sign up.
 3. Player selects date of tournament they wish to enter.
 4. System assigns player a unique tournament ID
 5. System displays confirmation of registration.



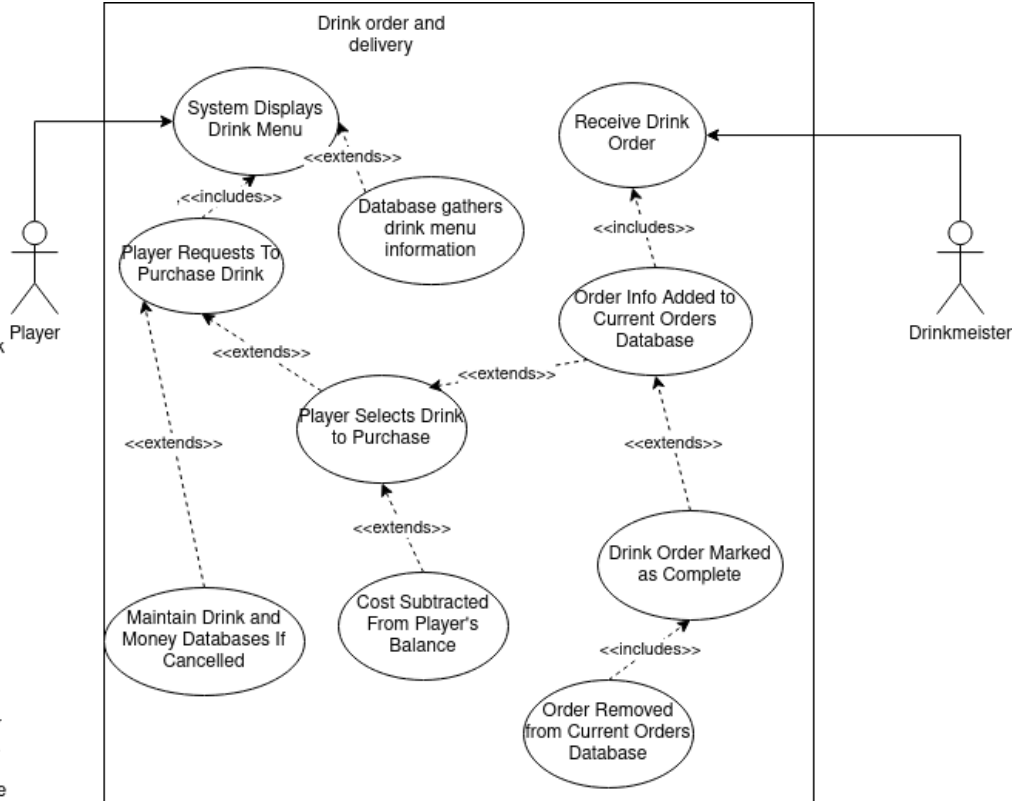
Participating Actors: Player,
Drinkmeister

Entry Conditions:
-Player logs in

Exit Conditions:
-Drinkmeister completes
drink delivery
-Player does not have
enough money to complete
transaction
-Player cancels delivery

Event flow:

1. Player requests view drink menu
2. System collects drink menu from database
3. System displays drink menu to player
4. Player requests to purchase drink.
5. Player selects drink to purchase
6. Cost is subtracted from player balance
7. Drink order and player location is added to current orders database
8. Order details appear for drinkmeister
9. Drinkmeister marks order as complete after delivered.
10. Order details removed from current orders database

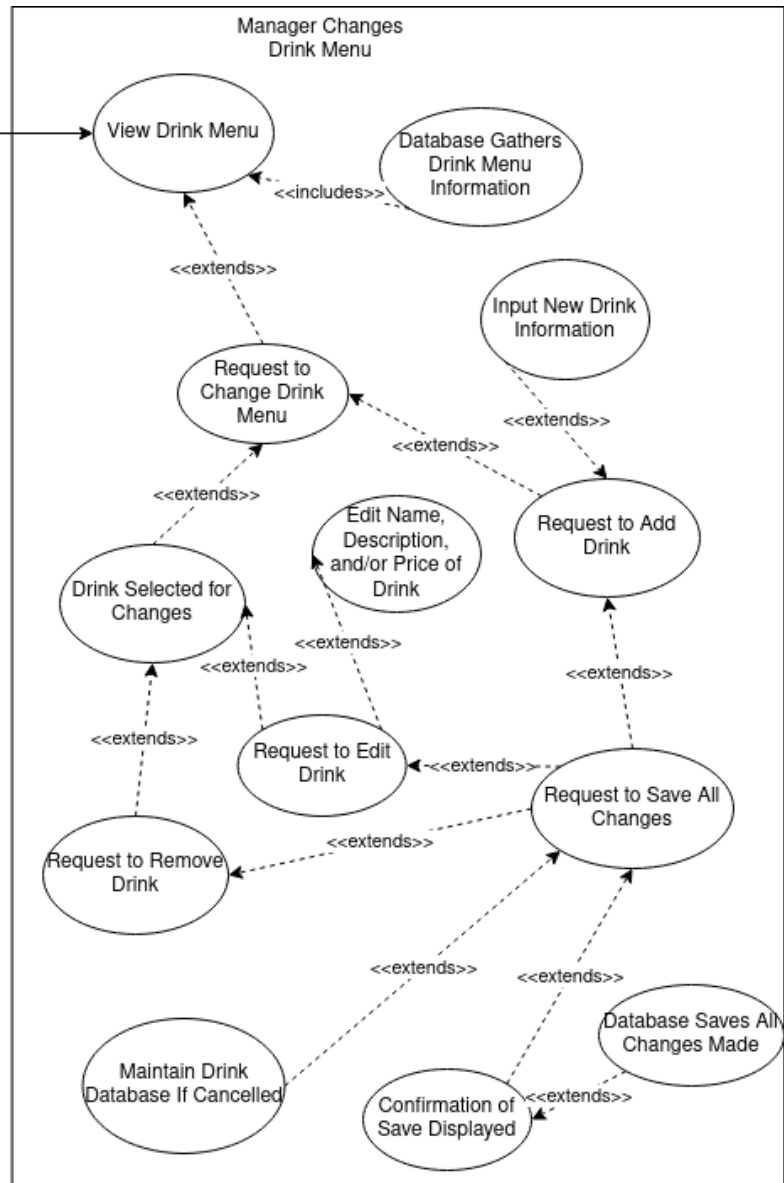
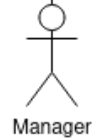


Participating Actor: Manager

Entry Conditions:
-Manager logs in

Exit Conditions:
-Manager saves changes made
-Manager cancels action

- Event flow:
1. Manager opens drink menu
 2. Manager requests to change menu
 3. Manager chooses to add, remove, or edit one or more drinks
 4. Manager makes changes to drinks in the menu
 - a) Manager edits a drink's name, description, and/or price.
 - b) Manager adds a drink with a name, description, and price.
 - c) Manager removes a drink from menu.
 5. Manager requests to save changes or discard them
 6. Changes are saved to database if save selected



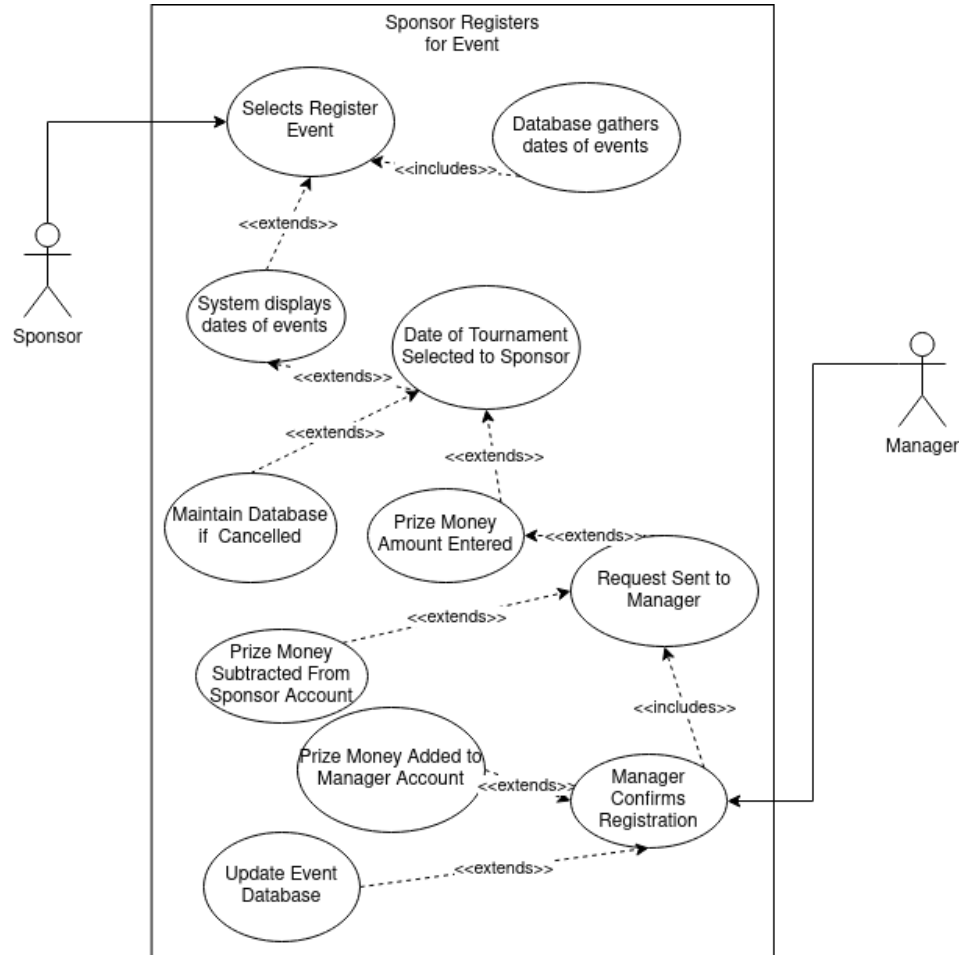
Participating Actors: Sponsor,
Manager

Entry Conditions:
-Sponsor logs in

Exit Conditions:
-Manager approves
registration
-Sponsor cancels action

Event flow:

1. Sponsor requests to register event
2. System displays available tournament dates to sponsor
3. Sponsor selects date of tournament to sponsor
4. Sponsor enters how much money to enter into prize pool
5. Sponsor sends request to manager to approve registration
6. Manager approves and money is transferred to manager

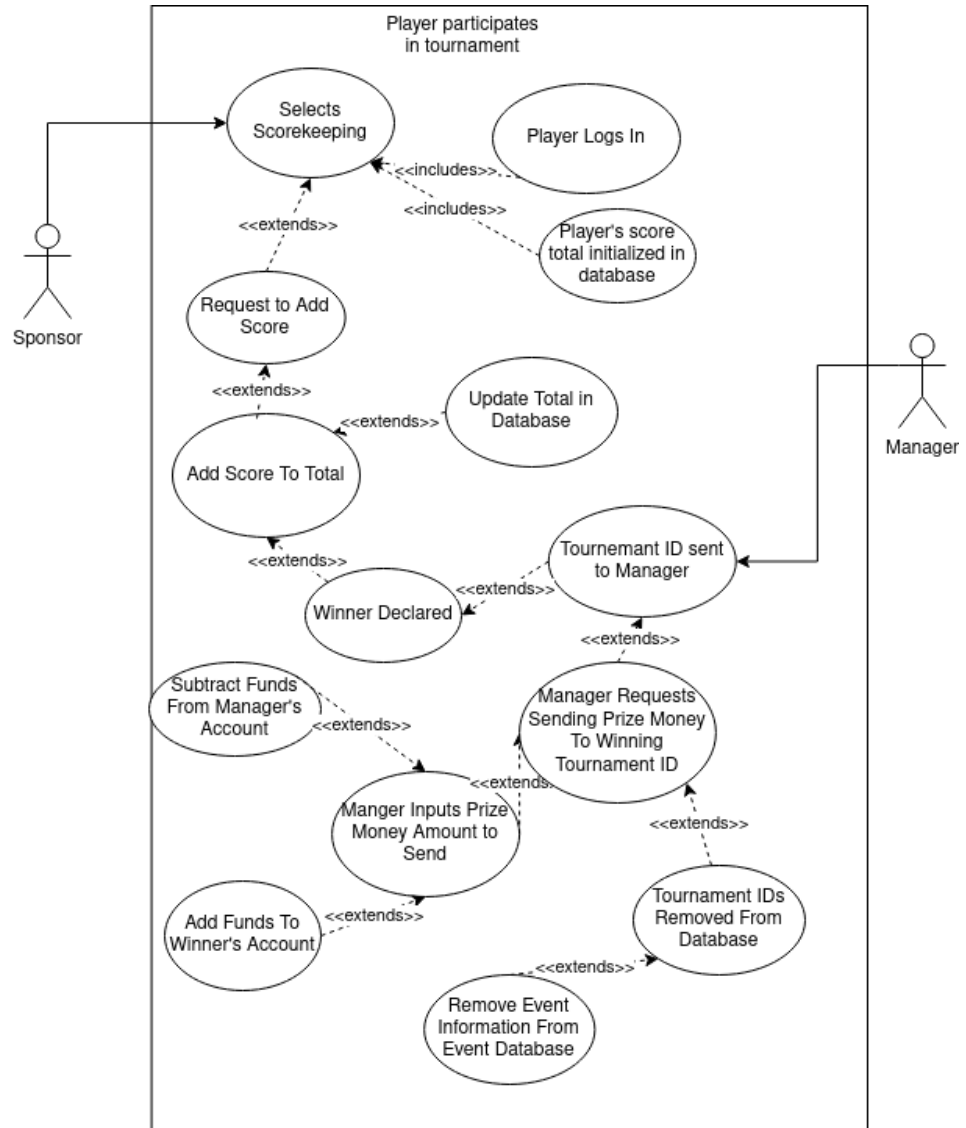


Participating Actors: Player,
Manager

Entry Conditions:
-Player signs up for
tournament

Exit Conditions:
-Manager awards prize
money to winner

- Event flow:
1. Database of player's total score initiated
 2. Player logs in
 3. Player selects scorekeeping
 4. At each hole, player requests to update score.
 5. Player inputs score at hole
 6. Database adds score to total
 7. After the tournament ends and winner is declared, manager selects winning player to transfer prize money to.
 8. Money transferred to winning player's account.
 9. All tournament IDs removed from database
 10. Tournament event info removed from event database

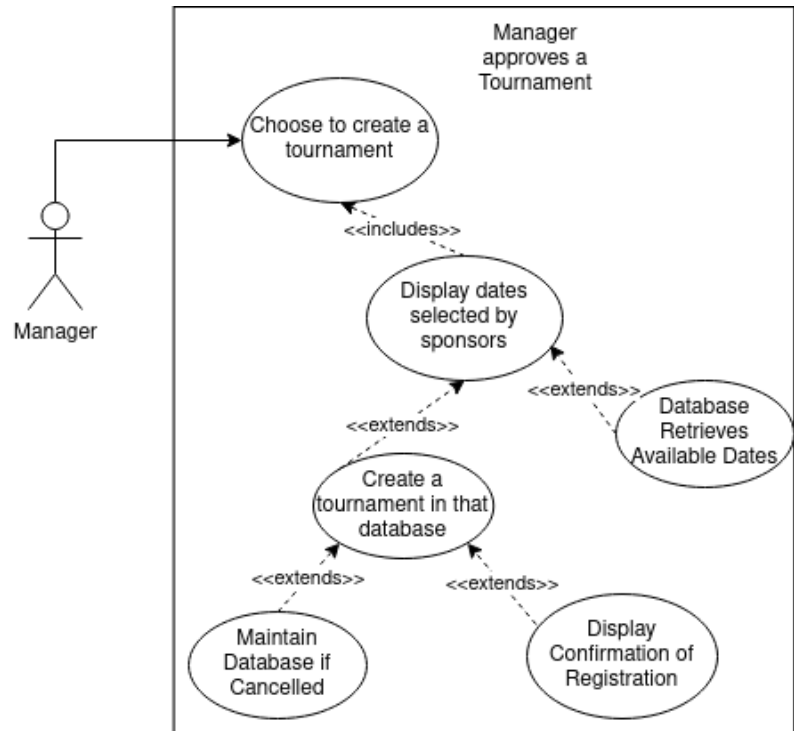


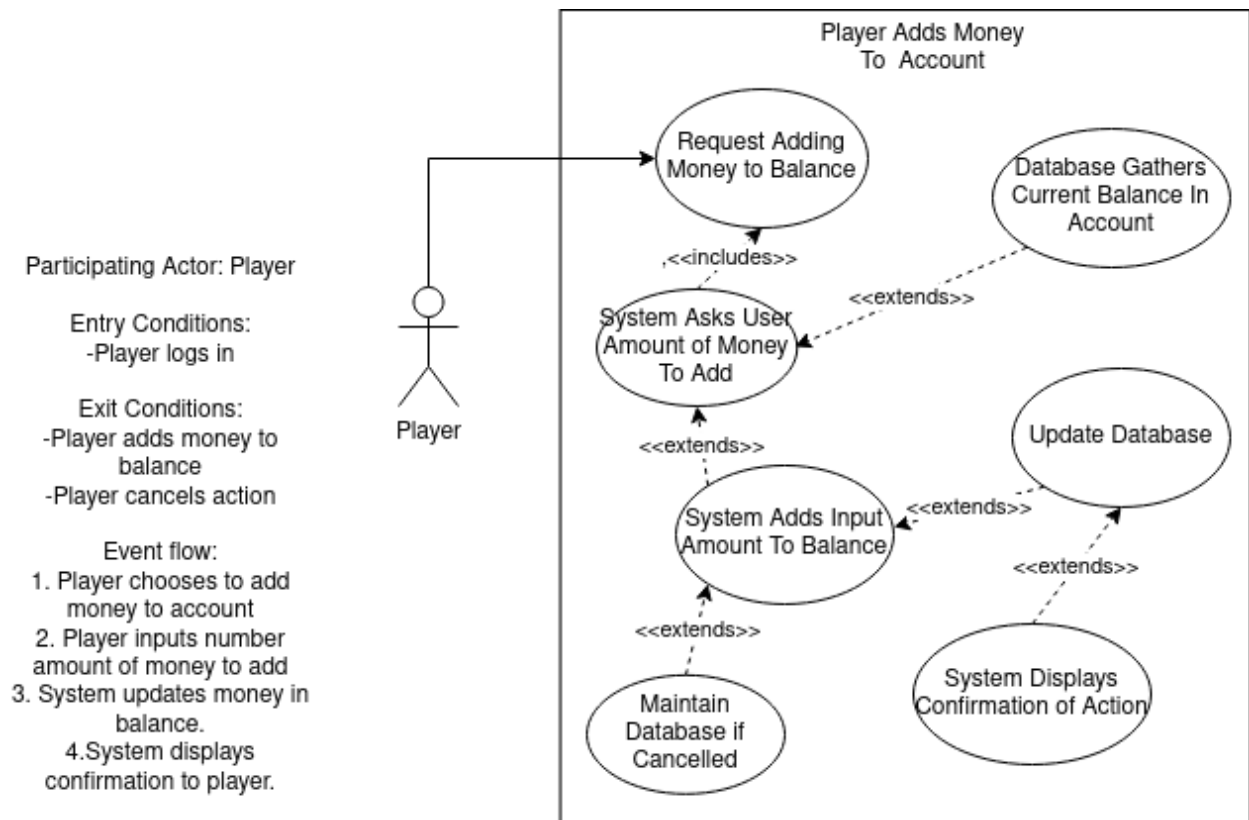
Participating Actor: Manager

Entry Conditions:
-Manager selects to approve

Exit Conditions:
-Manager creates new tournament
-Manager cancels creation

- Event flow:
1. Manager chooses to create a tournament
 2. System displays dates that have been sponsored by sponsors.
 3. Manager chooses desired tournament date.
 4. System creates a tournament in the database.
 5. System displays confirmation of registration.





Glossary

Tournament:

The tournament is a mini-golf competition. There are a variable (?) number of holes determined by the manager, and the players move from hole to hole at their own pace to compete for the lowest possible number of strokes to complete the course. The scores are posted to a leader board when each player finishes.

Players:

Players are the people taking part in the tournament. They move through the holes playing mini-golf, updating their scores and ordering drinks in the app. They are the most populous user, making up the vast majority of our user count.

DrinkMeisters:

DrinkMeisters are the employees that are helping run the tournament. They are in charge of fulfilling drink orders, and delivering them to players.

Manager:

The *Manager* organizes each tournament and collects the money used by the players and sponsors. They manage the accounts of the other users in the tournament.

Sponsors:

Sponsors provide money to host the tournament, in exchange for being able to advertise their products or services to the players. The money they provide is used for the prize pool.