



KICKIT DOCUMENTATION

CSCI201, Summer 2017

Abstract

This document contains project specifications for Kickit, a Dynamic Web Application.

Team Members: Jerry Yuzhou Ge, Xavier Yifei Liu, Natalie Monger,
Bradley Nguyen, Jason Woodruff
monger@usc.edu, yifeil@usc.edu, namnguyen@usc.edu, yuzhouge@usc.edu, jdwoodru@usc.edu

CONTENTS

HIGH-LEVEL REQUIREMENTS.....	2
Server.....	2
Registered Users Versus Guests.....	2
Registered Users	2
Guests.....	2
Game Administrators.....	3
Main Page.....	3
Game Lobby.....	4
Multithreading Feature.....	4
TECHNICAL SPECITIFICATION.....	5
Task Layout.....	5
Login Page GUI.....	6
Create Account GUI.....	6
Homepage Navigation Menu.....	6
Homepage Navigation Content.....	7
Create a Game GUI.....	8
Join a Game GUI.....	8
Game Lobby Previews.....	8
Server.....	9
Registered Users Versus Guests.....	9
Game Administrators.....	10
Main Page.....	10
Game Lobby.....	11
Multithreading Feature.....	11
DETAILED DESIGN.....	12
Hardware and Software Requirements.....	12
Database Schema.....	13
Class Diagram.....	14
User.....	14
GameGroup Data.....	14
GameGroup Methods.....	15
Player.....	16
Class Hierarchy.....	17
GUI.....	18
Sign In Page.....	18
Register Page.....	19
Landing Page.....	20
Features on Landing Page.....	20
Bottom of Landing Page.....	21
Main Page.....	22
My Events Page/Game Lobby Previews Pages.....	23
Create a Game GUI.....	24
TESTING.....	25
DEPLOYMENT.....	38

High-Level Requirements

Kickit should be broken into two different clients: the server and the user. The user client allows the game players to sign in. The server manages user data, game schedules, and user-generated content. Below is a detailed description of these attributes.

Server

The server client should allow multiple users to communicate with each other. The server will store user data (such as location, skill level, age, gender, etc.) and game data (such as chat logs and game attributes). All communication, information, and interaction about games will be handled through the server.

Registered Users versus Guests

People can use Kickit in two different ways: as a registered user or as a guest. A guest will be able to view the homepage of recent games going on. A registered user will have all the functionalities that a guest has, but with additional functionalities described below.

Registered Users

A registered user will have attributes such as name, skill level, location, gender, age, and an option to be a game administrator. A registered user can do these things:

- Create a game (the user will become an administrator with this option)
- Join a game
- Converse with other users in a game lobby

Guests

A guest will be able to view the Kickit platform homepage. However, a guest cannot have the same functionalities that a registered user has. If the guest wants to perform one of the options described above, the guest will be prompted to register for Kickit.

Game Administrators

A registered user who creates a game is the administrator of that game. They can set attributes for that game such as:

- Sport Type
- Maximum Number of Players
- Time (Start, end time)
- Group / Game Name
- Description / Comments (such as Apple Employees after hours casual soccer game)
- Skill Level (Beginner / Intermediate / Advanced / Any)
- Gender (M / F / Any)
- Age (10-15 / 15-20 / 20-30 / 30-40 / ... / Any)
- Location (Add google map screenshot or use plug-ins)

The administrator can change these attributes at any time and can lock the game lobby when they feel it is ready. This will remove the game lobby from the main page. It is up to the players in the lobby to communicate decide if they want to stay in the game after any changes have been made.

Main Page

The main page, which displays once a user logs in, will display a list of game lobbies they can join. The games displayed will have matching attributes with that of the user (location, skill level, age, gender). The user will have the ability to filter the game results based on sport type, time, recently created, etc. The main page will also have a list on the side of the page that displays all games the user has joined so far with game time reminders and links to the game lobbies.

Game Lobby

The game lobby should display detailed game information (like type, location, time, skill level, age, gender). Users who join the lobby will see other joined players and enjoy a chatting feature for communication. The user can also choose to leave or stay in this group.

Multithreading Feature

Kickit will have a multithreading feature when different users want to join the same group at the same time and when many users are talking in a game lobby at the same time.

Technical Specification

Task	Estimated Time
Frontend:	
Login Page GUI	.5 hr
Create Account GUI	3 hr
Homepage Navigation Menu	2 hr
Homepage Main Content	8 hr
Create a Game GUI	3 hr
Game Lobby Previews	8 hr
Server:	
MySQL/JDBC Setup	2 hr
Networking	8 hr
Server Backend	8 hr
Backend:	
Game Lobby Backend	8 hr
User Communication	8 hr
User Information	8 hr
Filter Backend	8 hr
Multithreading	5 hr
Documentation:	
Project Proposal	.5 hr
High Level Requirements Document	2 hr
Technical Specifications Document	3 hr
Detailed Design Document	6 hr
Testing Document	2 hr
Test Cases/Quality Assurance	8 hr
Code Complete Document	3 hr
Deployment Document	1 hr
Make Presentation Slides	1 hr
Practice Presentation with Group Members	2 hr
Total Estimate	108 hr

Login Page GUI

The Login GUI will display a GUI that allows a user to log in to the program or register. There will also be an option for the visitor to bypass the login page and view the game lobby as a guest.

Create Account GUI

If the user decides to create an account instead of logging in, by clicking on the create account button, a registration page will appear requesting information. The sample will include these fields in a form for the user to fill out:

- First Name
- Last Name
- Username
- Password
- Email
- Sports of interest
- Skill level
- Location
- Age
- Gender

Homepage Navigation Menu

After successful login or registration, the user will be led to the game homepage. The homepage displays the user information in the top right-hand corner, and the user can click in this area to edit his or her profile. Each page will contain a navigation bar with buttons for different functionalities. Here are the details of this feature:

- For a Registered User
 - An arrow displays in the upper right-hand corner
 - If the user clicks on this arrow, a drop-down menu will appear with these options:

- Settings
 - Edit Profile
 - Sign Out
- For a Guest
 - There will be a “Join Kickit” button in the upper right-hand corner for all guests visiting the page.

Homepage Main Content

Top Menu Items/Buttons

There are two main ways to navigate the homepage: 1) a user can filter his or her search results, or 2) the user can create a game. More details on these functionalities:

- Filter Search Results
 - A “Filter” button will feature a drop-down menu with ways to filter the games according to:
 - Sport type
 - Location
 - Time
 - Age
 - Gender
- Create a Game
 - A “Create Game” button will prompt a user to the Create a Game GUI.

Main Content

In addition to these two buttons, the rest of the homepage will be dedicated to displaying a list of games to join. If the results are unfiltered (when a user first logs in), it will display the most recent games. Otherwise, it will display the filtered results according to what the user wants to see.

Right Sidebar

On the right side, the homepage will display a list of games the user has joined. A user can click on a game to go to the game lobby (the individual page for the game).

Create a Game GUI

This GUI will prompt a user to create a game. It will be a form that will be submitted (post). The user will be asked to answer these questions:

- What is the name of the game?
- Where will the game be held?
- What time will the game start?
- Provide a description of the game.
- What skill levels can play?
- What type of game is it?

Join a Game GUI

This GUI will simply prompt the user to join a game with a message to the effect of, “Are you sure you want to join this game?”

Game Lobby Previews

Games will be displayed in rectangles with the following information:

- Name of Game/Group
- Location of Game
- Time of Game
- Description of Game
- Skill Level for Game
- Background photo of the sport type.

Server

The server client should allow multiple users to communicate with each other. The server will store user data (such as location, skill level, age, gender, etc.) and game data (such as chat logs and game attributes). All communication, information, and interaction about games will be handled through the server.

Registered Users versus Guests

People can use Kickit in two different ways: as a registered user or as a guest. A guest will be able to view the homepage of recent games going on. A registered user will have all the functionalities that a guest has, but with additional functionalities described below.

Registered Users

A registered user will have attributes such as name, skill level, location, gender, age, and an option to be a game administrator. A registered user can do these things:

- Create a game (the user will become an administrator with this option)
- Join a game
- Converse with other users in a game lobby

Guests

A guest will be able to view the Kickit platform homepage. However, a guest cannot have the same functionalities that a registered user has. If the guest wants to perform one of the options described above, the guest will be prompted to register for Kickit.

Game Administrators

A registered user who creates a game is the administrator of that game. They can set attributes for that game such as:

- Sport Type
- Maximum Number of Players
- Time (Start, end time)
- Group / Game Name
- Description / Comments (such as “Apple Employees after hours casual soccer game”)
- Skill Level (Beginner / Intermediate / Advanced / Any)
- Gender (M / F / Any)
- Age (10-15 / 15-20 / 20-30 / 30-40 / ... / Any)
- Location (Add google map screenshot or use plug-ins)

The administrator can change these attributes at any time and can lock the game lobby when he feels it is ready. This will remove the game lobby from the main page. It is up to the players in the lobby to communicate to decide if they want to stay in the game after any changes have been made.

Main Page

The main page, which displays once a user logs in, will display a list of game lobbies they can join. The games displayed will have matching attributes with that of the user (location, skill level, age, gender). The user will have the ability to filter the game results based on sport type, time, recently created, etc. The main page will also have a list on the side of the page that displays all games the user has joined so far with game time reminders and links to the game lobbies.

Game Lobby

The game lobby should display detailed game information (like type, location, time, skill level, age, gender). Users who join the lobby will see other joined players and enjoy a chatting feature for communication. The user can also choose to leave or stay in this group.

Multithreading Feature

Kickit will have a multithreading feature when different users want to join the same group at the same time and when many users are talking in a game lobby at the same time.

Detailed Design

Hardware Requirements

Windows:

Windows 8 (Desktop)

Windows 7 Windows Vista SP2

RAM: 128MB

Disk Space: Recommended 512MB

Processor: Minimum Pentium 2 266MHz processor

Mac OS X:

Intel-based Mac running Mac OS X 10.8.3+, 10.9+

Linux:

Oracle Linux 5.5+

Oracle Linux 6.x (32-bit), 6.x (64-bit) Oracle Linux 7.x (64-bit)

Red Hat Enterprise Linux 5.5+ (32-bit), 6.x (64-bit)

Ubuntu Linux 12.04 LTS, 13.x

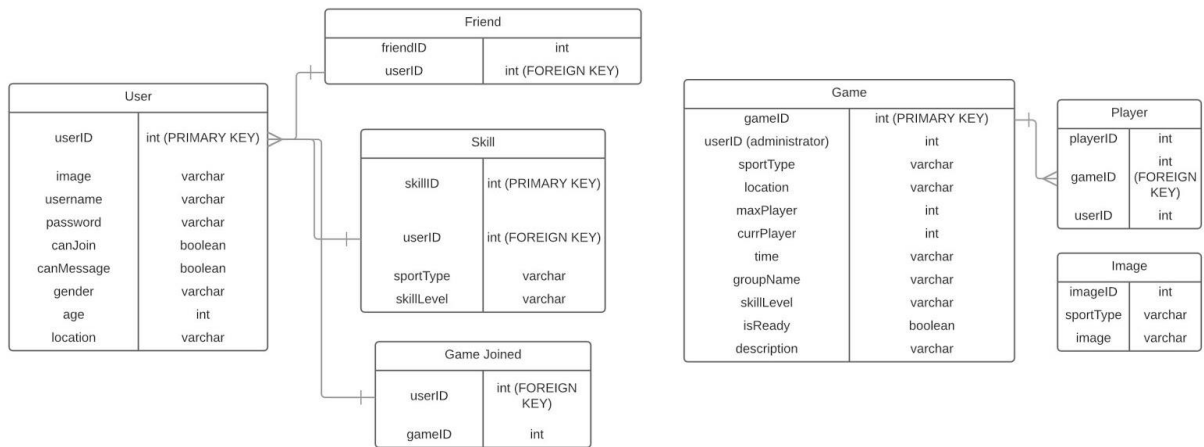
Software Requirements

Java 8 Eclipse IDE for Java EE Developers (Luna/Mars)

Web Requirements

View in Browser, e.g. Chrome

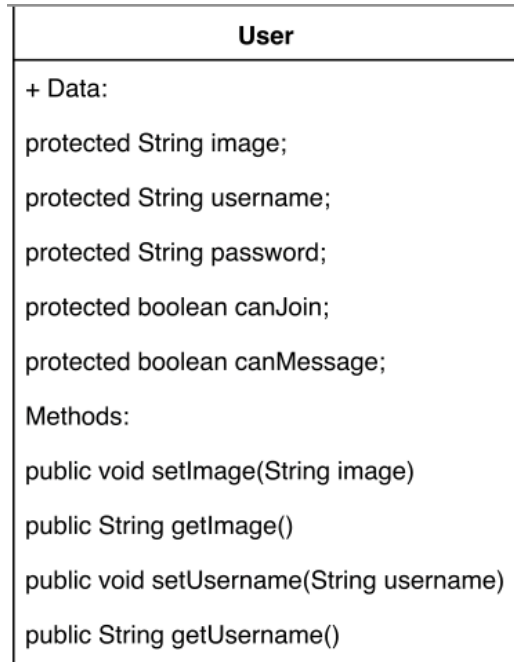
Database Schema:



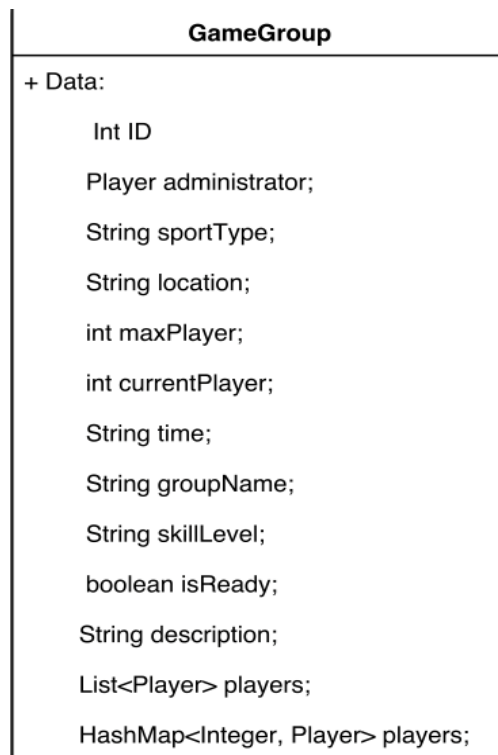
Client

Class Diagram:

User



Game Group Data



Game Group Methods

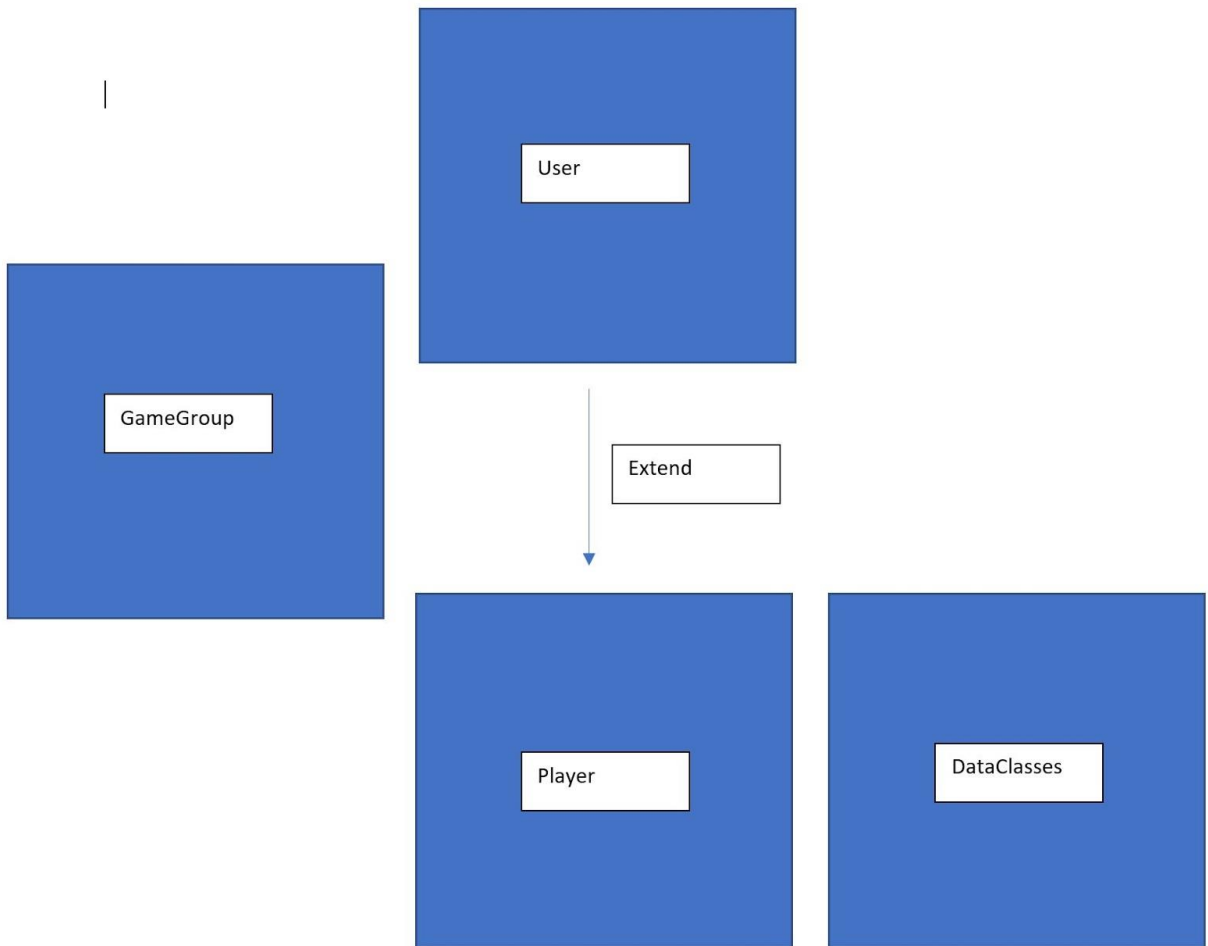
Methods:

```
public void ready()
public void unready()
public void addPlayer(Player p)
public void removePlayer(Player p)
public boolean isFull()
public void setAdministrator(Player admin)
public Player getAdministrator()
public String getSportType()
public void setSportType(String sportType)
public void setLocation(String location)
public String getLocation()
public void setMaxPlayer(int maxPlayer)
public int getMaxPlayer()
public void setCurrentPlayer(int currentPlayer)
public int getCurrentPlayer()
public void setGroupName(String groupName)
public String getGroupName()
public void setTime(String time)
public String getTime()
public void setSkillLevel(String skillLevel)
public String getSkillLevel()
public void setDescription(String description)
public String getDescription()
```


Player

Player
<p>+ Data:</p> <p>int id;</p> <p>ArrayList<Player> friendList;</p> <p>private ArrayList<Player> blackList;</p> <p>private HashMap<String, String> skills;</p> <p>private String gender;</p> <p>private int age;</p> <p>private String location;</p> <p>private HashMap<Integer, GameGroup> gamesJoined;</p> <p>private HashMap<Integer, GameGroup> gamesAdministrated;</p> <p>Methods:</p> <p>public void changeSkill(String sportType, String skillLevel)</p> <p>public void addFriend(Player p)</p> <p>public void removeFriend(Player p)</p> <p>public void addBlack(Player p)</p> <p>public void removeBlack(Player p)</p> <p>public void setID(int id)</p> <p>public int getID()</p> <p>public void setGender(String gender)</p> <p>public String getGender()</p> <p>public void setLocation(String location)</p> <p>public String getLocation()</p> <p>public void setAge(int age)</p> <p>public int getAge()</p>

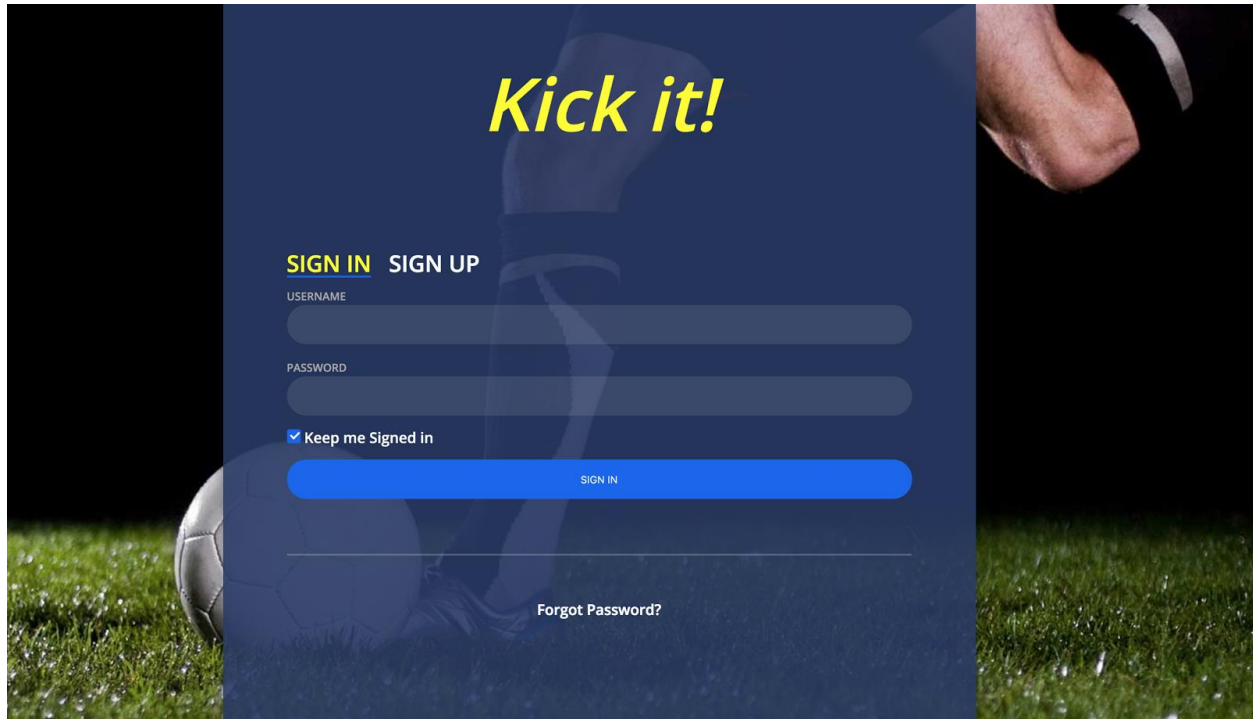
Classes Hierarchy



GUI

Sign In Page

The Sign In page is the first page that the user reaches when he or she enters the Kickit website. It features fields to enter a username and a password so the user can explore the website.



Register Page

The Register Page page is included with the Sign In page. If the user clicks on the “Sign Up” tab, he or she will be directed to this GUI. It features fields to create a username and a password. It asks the user to repeat the password. Furthermore, it asks for some profile attributes: gender, age range, and location.

Kick it!

SIGN IN SIGN UP

USERNAME

PASSWORD

REPEAT PASSWORD

GENDER: MALE ☒ FEMALE ☐

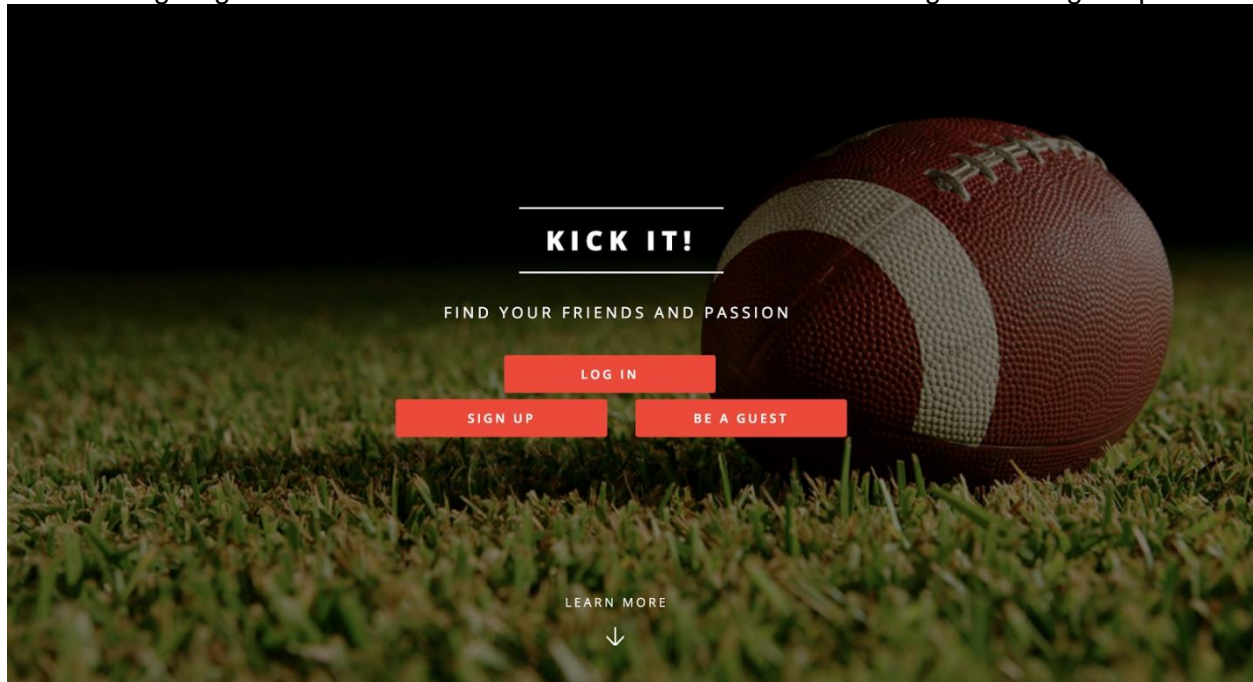
AGE RANGE: 10-15

LOCATION: USC

SIGN UP

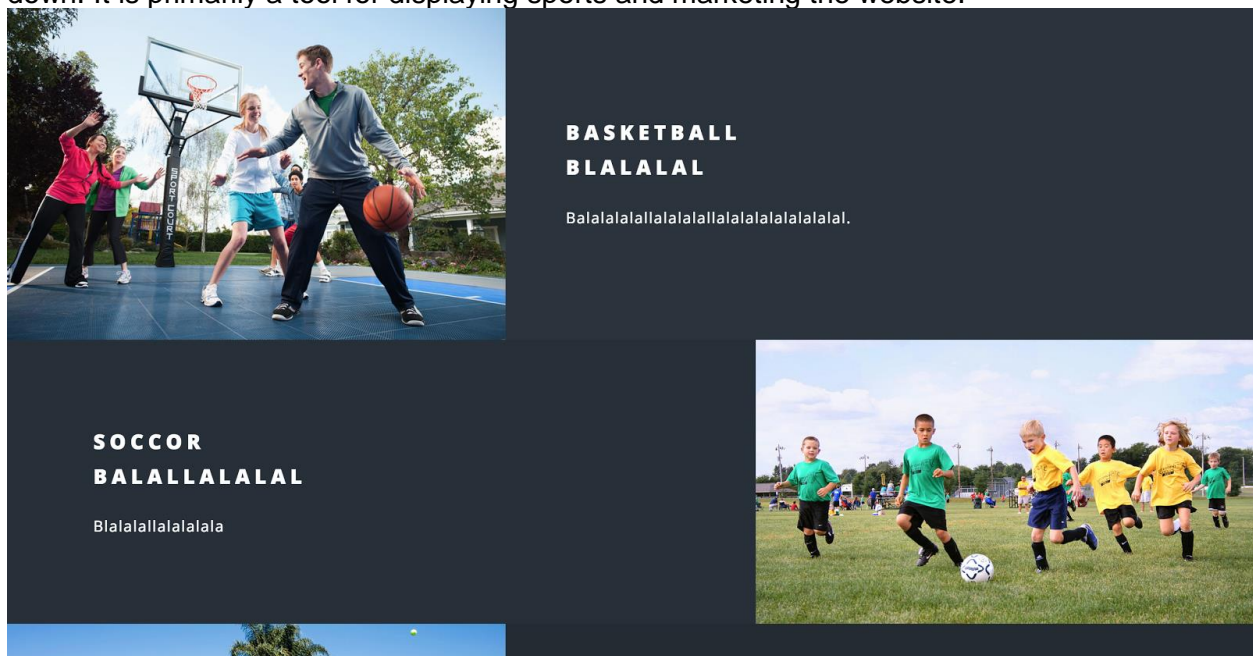
Landing Page

The Landing Page is the GUI the user reaches once he or she either signs in or signs up.



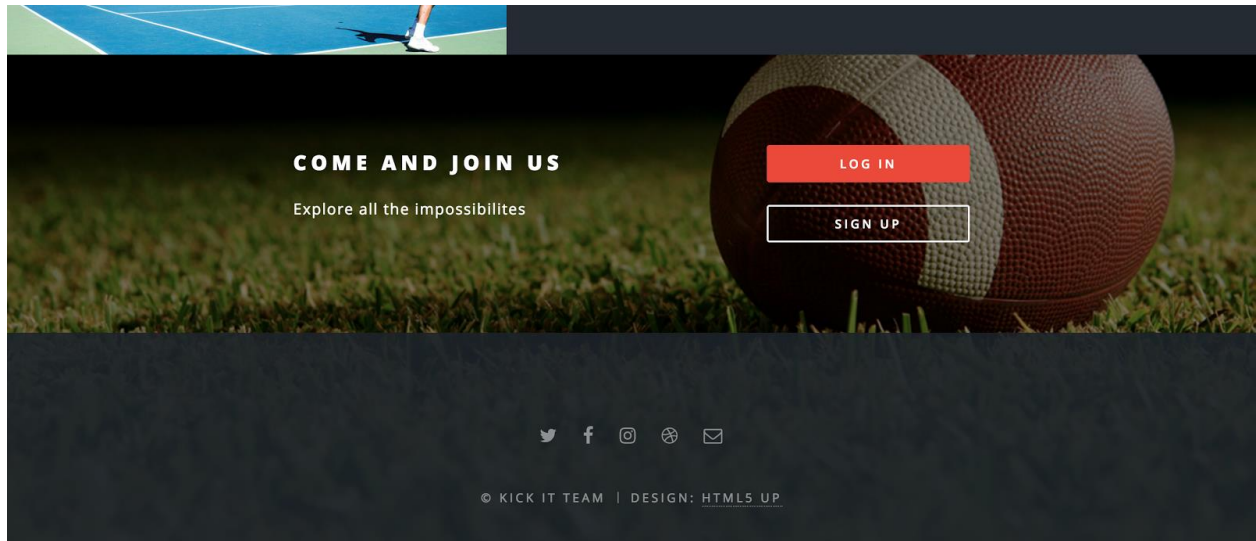
Features on Landing Page

The following GUI shows a sample of the detail on the Landing Page after the user scrolls down. It is primarily a tool for displaying sports and marketing the website.



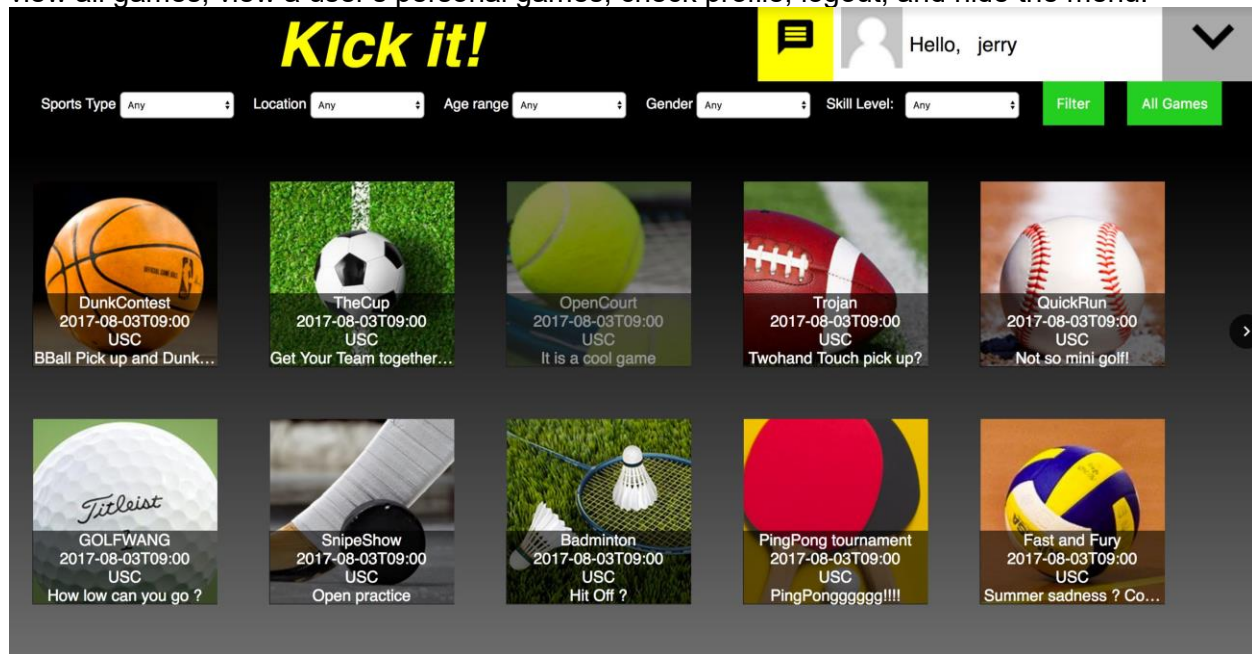
Bottom of Landing Page

The bottom of the landing page features a prompt for the user to log in or sign up, which connects to the same GUI featured above.



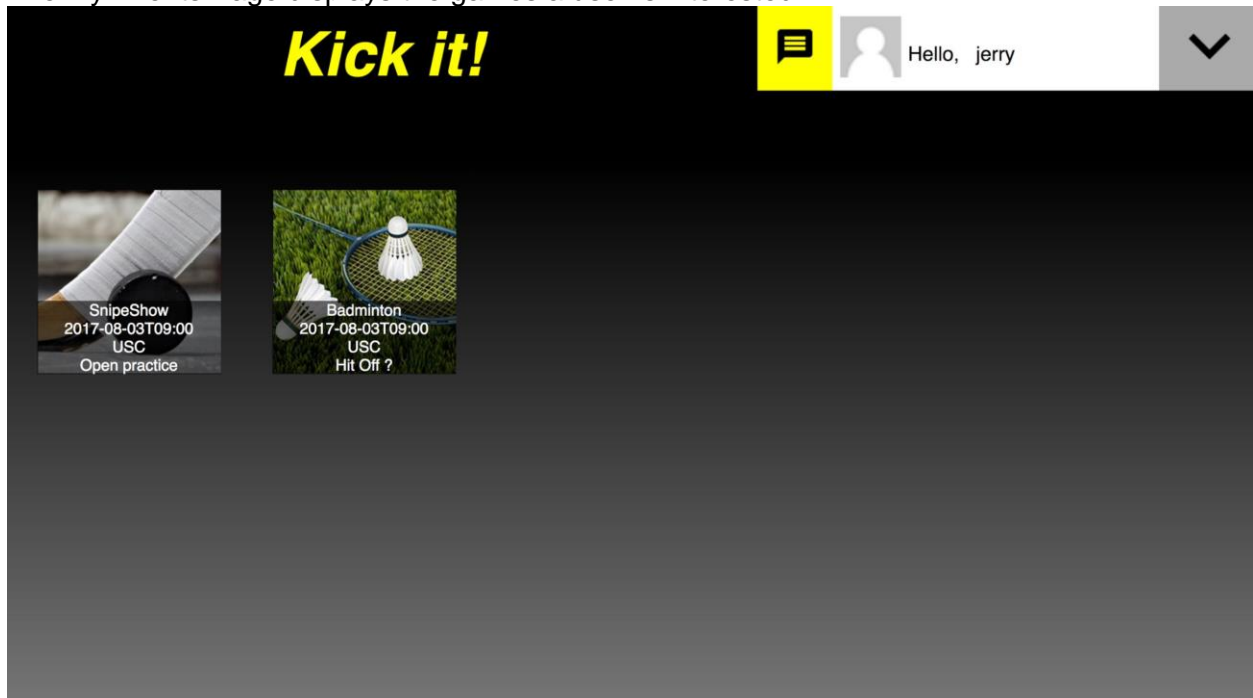
Main Page

The Main Page displays current games and features a sidebar with options to create a game, view all games, view a user's personal games, check profile, logout, and hide the menu.



My Events Page / Game Lobby Previews Page

The My Events Page displays the games a user is interested in.



Create a Game GUI

The Create a Game GUI is displayed when a user clicks on “Create an Event” from the Main Page.

The screenshot displays a web application for creating a game. The browser's address bar shows the file path: `file:///Users/yuzhouge/Desktop/Kickit/CS201_FinalProject_Kickit/myevents.html`. The browser's tab bar includes tabs for 'Apps', 'WeDesign', 'Programming', 'Mathematics', 'Blogs', 'Algorithms', 'Graphics', 'Front-End Program', 'Hokkaido Trip', 'CS201', and 'CSCI 201L'.

The application interface consists of a red header with the text **Kick it!**, a blue sidebar on the right, and a yellow main content area.

Blue Sidebar (Navigation):

- Create an Event
- All Events
- My Events
- Show my profile
- Log Out
- Hide Me!

Yellow Main Content Area (Form):

Create a new game


Game Title:

Game Image: No file chosen

Game Type:

Game Time/Date: -- --

Game Location:



Skill Level:

Testing

Tests for Sign Up/Register Page:

Test #	1
Description	User attempts to create an account
Steps	a. In sign up page b. User fills out all forms c. Press sign up button
Expected Result	User successfully creates an account and can explore the site

Test #	2
Description	User attempts to create an account with empty username
Steps	a. In sign up page b. Leave the username field blank c. Press sign up button
Expected Result	Receives an error message saying that username needs to be filled

Test #	3
Description	User attempts to create an account with empty password
Steps	a. In sign up page b. Leave the password field blank c. Press sign up button
Expected Result	Receives an error message saying that password needs to be filled

Test #	4
Description	User attempts to create an account with empty repeat password
Steps	d. In sign up page e. Leave the repeat password field blank f. Press sign up button
Expected Result	Receives an error message saying that repeat password needs to be filled

Test #	5
Description	User attempts to create an account with two passwords that don't match
Steps	a. In sign up page b. Enter two different passwords in password and repeat password field c. Press sign up button
Expected Result	Receives an error message saying that passwords don't match

Test #	6
Description	User attempts to create an account with username that already exists
Steps	a. In sign up page b. Enter an existing username in database c. Press sign up button
Expected Result	Receives an error message saying that this user already exists

Test #	7
Description	User attempts to create an account with password length of less than 6
Steps	a. In sign up page b. Fill the username and enter a password "1234" in both password and repeat password field c. Press sign up button
Expected Result	Receives an error message saying that the password is too short

Test #	8
Description	User attempts to create an account with illegal password
Steps	a. In sign up page b. Fill the username and enter all illegal characters like "@@@\$%^&%*@!#!" c. Press sign up button

Expected Result	Receives an error message saying that the password format is wrong
-----------------	--

Tests for Landing Page:

Test #	9
Description	Go to Login Page from Landing Page
Steps	a. In landing page b. User clicks on login button
Expected Result	Page updates to login page

Test #	10
Description	Go to Sign Up Page from Landing Page
Steps	a. In landing page b. User clicks on sign up button
Expected Result	Page updates to sign up page

Test #	11
Description	Go to Be A Guest from Landing Page
Steps	a. In landing page b. User clicks on be a guest button
Expected Result	Page updates to main page with guest information

Tests for Login Page:

Test #	12
Description	User attempts to login
Steps	a. In login page b. User enters username and password c. Press login button

Expected Result	User successfully logs in and can explore the site
-----------------	--

Test #	13
Description	User enters a username that doesn't exist in the database, and tries to log in
Steps	<ol style="list-style-type: none"> In the login page Enter an invalid username in the username field Presses the login button
Expected Result	Receives an error message saying that the username is invalid

Test #	14
Description	User enters a username with an incorrect password
Steps	<ol style="list-style-type: none"> In the login page Enter a valid username and an incorrect password field Press the login button
Expected Result	Receives an error message saying that the password is invalid

Tests for Main Page:

Test #	15
Description	User presses Menu Drop Down button
Steps	<ol style="list-style-type: none"> In main page Click on the menu dropdown arrow (top right arrow)
Expected Result	Display a menu of: <ol style="list-style-type: none"> Create an Event All Events My Events Show my full profile Log Out

Test #	16
--------	----

Description	User presses Notification Box button (registered user)
Steps	c. In main page d. Click on the message box button (speech bubble icon)
Expected Result	Display all the recent incoming messages of this user in a drop down box

Test #	17
Description	Guest User clicks Notification box button
Steps	a. In main page b. Click on the notification box button (speech bubble icon)
Expected Result	Show an alert box saying that the user needs to register to see notifications

Tests for Create Event Functionality:

Test #	18
Description	Registered user clicks Create an Event button
Steps	a. In main page b. Click the Create an Event button
Expected Result	Open the Create an Event page where user can set game attributes and host the game on the server.

Test #	19
Description	Guest user clicks Create an Event button
Steps	a. In main page b. Click the Create an Event button
Expected Result	Show an alert box that notifies the user they must register to create an event

Tests for My Events Functionality:

Test #	20
--------	----

Description	Registered User clicks My Events button
Steps	a. In main page b. Click My Events button
Expected Result	Open the My Events page where all the events that the user has joined/created are displayed. User may open game lobbies he or she has joined/created or go back to main page.

Test #	21
Description	Guest user clicks My Events button
Steps	a. In main page b. Click My Events button
Expected Result	Show an alert box notifying the user that they must register to create and join events

Tests for Edit Profile Functionality:

Test #	22
Description	Registered User clicks Edit Profile button
Steps	a. In main page b. Click Edit Profile button
Expected Result	Open the edit profile page where the user can edit his or her profile details

Test #	23
Description	Guest user clicks Edit Profile button
Steps	a. In main page b. Click Edit Profile button
Expected Result	Show an alert box notifying the user that he or she must register to edit his or her profile

Test for Log Out Functionality:

Test #	24
--------	----

Description	Click Log Out button
Steps	a. In main page b. Click the Logout button
Expected Result	Logs the user out and closes the thread. Redirects user to landing page.

Test for Join Game Functionality:

Test #	25
Description	Registered User clicks Join a Game button
Steps	a. In main page b. Click one of the game summary buttons on the main page
Expected Result	Open the game info page where the user can see the full details of the game. The user may join or go back if he or she wishes.

Test #	26
Description	Guest User clicks Join a Game button
Steps	a. In main page b. Click on one of the game summary buttons on the main page
Expected Result	Show an alert box notifying the user that they must register to view/join a game

Tests for Page Navigation:

Test #	27
Description	Create a Game Lobby from Create an Event
Steps	a. In Create an Event page b. User clicks on the Create Game button
Expected Result	Redirect the user to the newly created Game Lobby. Create new GameGroup in database with selected attributes. Game lobbies are displayed on the server.

Test #	28
Description	Go to Game Lobby from My Events
Steps	a. In My Events page b. Click on a game summary in the page
Expected Result	Redirect the user to the Game Lobby the user selected

Test #	29
Description	Save Profile Edits in Edit Profile
Steps	a. In Edit Profile page b. Click Save Profile button
Expected Result	Save the user's profile edits, update their database, and redirect to the main page

Test #	30
Description	Display next page of games
Steps	a. In main page b. Click the right arrow on the bottom of the screen
Expected Result	Game summaries on the page should be replaced with the next set of games that could not fit on the main page. If no more games exist (rightmost set of games), do nothing.

Test #	31
Description	Display previous page of games
Steps	a. In main page b. Click the left arrow on the bottom of the screen
Expected Result	Game summaries on the page should be replaced with the previous set of games that could not fit on the main page. If the user is already on the original set of games in the main page (leftmost set of games), do nothing.

Tests for Filtering Games:

Test #	32
Description	Main page filter game type
Steps	a. In main page b. Change the game type filter
Expected Result	All the displayed games should change to match the game type filter

Test #	33
Description	Main page filter age range
Steps	a. In main page b. Change the age range
Expected Result	All the displayed games should change to match the age filter

Test #	34
Description	Main page filter location
Steps	a. In main page b. Change the game location
Expected Result	All the displayed games should change to match the location filter

Test #	35
Description	Main page filter skill level
Steps	a. In main page b. Change the skill level filter
Expected Result	All the displayed games should change to match the skill level filter

Test #	36
Description	Main page filter upcoming date

Steps	a. In main page b. Change upcoming date
Expected Result	All the displayed games should change to match the game starting date.

Tests for Create Game Page:

Test #	37
Description	User attempts to create a game with an empty title
Steps	a. In create game page b. Let game title be empty c. Press create game button
Expected Result	Receives an error message saying that the game needs a name/title

Test #	38
Description	User attempts to create a game without uploading a game image
Steps	a. In create game page b. Do not upload the game image file c. Press create button
Expected Result	The game can be created with a default game picture

Test #	39
Description	User attempts to create a game with the time/date empty
Steps	a. In create game page b. Let the time/date field be empty c. Press create button
Expected Result	Receives an error message saying that the game needs a date/time

Test #	40
Description	User attempts to create a game with game location empty

Steps	<ul style="list-style-type: none"> a. In create game page b. Let the game location field be empty c. Press create button
Expected Result	Receives an error message saying that the game needs a location

Test #	41
Description	User attempts to submit the form and create game
Steps	<ul style="list-style-type: none"> a. In create game page b. With all fields filled c. Click submit button
Expected Result	Link the page to a new created game lobby

Test #	42
Description	User attempts to reset the form
Steps	<ul style="list-style-type: none"> d. In create game page e. With all fields filled f. Click reset button
Expected Result	User should still stay at the current page and all fields should be empty

Test #	43
Description	Image display: Create a new game of sport type Soccer
Steps	<ul style="list-style-type: none"> 1. Create a new game of type soccer 2. Check the GUI to see the new game
Expected Result	A new game with a picture of a soccer ball should display

Test #	44
Description	Game summary boxes and corresponding images to game type
Steps	<ul style="list-style-type: none"> 1. Navigate to main page 2. Check the picture next to it

Expected Result	The image displayed should be a ball for the corresponding sport
-----------------	--

Tests for User Events Page:

Test #	45
Description	User attempts to click on joined and created games button
Steps	a. In my events page b. The logged in user clicks on his or her joined and created page
Expected Result	Redirects the page to the clicked game lobby page

Test #	46
Description	Switch page button
Steps	a. In my event page b. User clicks on the left or right arrow to switch my events page if the user has that many games joined
Expected Result	User can switch between all these pages.

Test for View and Edit Profile Functionality:

Test #	47
Description	User updates password (new password and new repeat new password don't match)
Steps	a. In my edit profile page b. Type in new password and a different password in repeat your password field. c. Click on update
Expected Result	Error message that the new password and repeat new doesn't match.

Test for Message Functionality:

Test #	48
Description	Enable networking feature for user

Steps	<ol style="list-style-type: none"> 1. User logs in 2. On server side, check that the user is in the HashMap
Expected Result	Adds user to HashMap with key: username and value: session.

Test #	49
Description	User sends message to server on creation
Steps	<ol style="list-style-type: none"> 1. User logs in 2. On server side, check that the user sent a message with their ID in it
Expected Result	The user should have sent a message with its unique ID in it.

Deployment

To deploy this application within Eclipse, import the CS201_FinalProject_Kickit.zip file using the built in import function. This should generate a project called Kickit with root folders src, images, assets, and a MYSQL script. The WebContent folder contains the page jsps. The src folder contains the client, server, servlet, and backend classes and code. Assets such as images will also be included in the project folder.

We are using two outside libraries:

- mysql-connector-java-5.1.40-bin.jar
- jquery-3.2.1.js

Step-by-step instructions on how to run the program:

- To execute the server for Kickit, run the Server inside the networking package.
- To start Kickit, open the landing.html with a browser such as Chrome.
- In the landing page, you can choose to login if you already have an account or sign up if you do not have an account.
- If you do not want to login or register, you can go to the main page by clicking the “Be a Guest” button.
- After you have logged in, you can then start to explore Kickit!
- In the main page, you can see all the upcoming games. You can choose to join then or create your game. Have Fun!