WD Diploma - Capstone Project

**Project Title:** *PeelSoccerLeague*

**Author:** *Jyotish Melwani*

**Cohort:** *Web Development PT 2022*

## Project Overview

### 1.1 Description

This is application will allow people throughout the Peel Region to connect with each other to schedule soccer games on a casual basis.

Games can be organized by a user for where and when players can meet and coordinate

### 1.2 Problem

Finding people to play soccer with it is always a challenge especially when you are trying to play casually.

Then assigning people to a position becomes a struggle when you don’t know the strengths and weakness of each player.

This application looks to solve that issue by having a roster of available players who can join a scheduled game and choose an available position.

### 1.3 User Profile

*Who is the end user? How will they use your application? How does the application help them? Are there any special considerations that your design must take into consideration for the end user?*

The end user is the person looking to be part of soccer games that occur throughout the PEEL region on a casual basis without having to know people in the squad.

The user would be able to input the following information as part of their details: Name, Position, Contact Information, Availability.

### 

### 1.4 Requirements: Use Cases and Features

A user would be able to log in and input their details to register.

From there they can join to play in any game.

Games will be capped at 11 players on each side with each player selecting a position they would like to play. There will be 2 substitutes allowed for either side

### 1.5 Tech Stack and APIs

* *Technologies that will be used in solution*
* *Any libraries that will be used to save time or provide more functionality (make sure you have researched and understand how to use the libraries and any potential limitations.*
* *APIs/Web sites that will be required in the solution for data*

React – Client-side code available via browser

Express – Web application framework

NodeJS – Runtime environment

MongoDB – Document Database

API for creating, editing, and getting single player information

API for creating, editing, and getting all player information

API for creating, editing, and getting a game

## 2. Client-Side Implementation

### Site Map

*List of pages/screens that your project will implement with brief descriptions. You can use* [***draw.io***](https://www.draw.io/) *to make technical diagrams. e.g.)*

*In list format:*

* *Game*
* *Log-In*
  + My Game
* *About Us*
  + *Team*
  + *Culture*
  + *Careers*
* *Contact Us*
  + *Address*
  + *Email*
  + *Phone*

### Screen Details

*Mockups and/or descriptions of input/output or state that the screens will need. You can use online tools like* [***Balsamiq***](https://balsamiq.com/index.html)*, or pictures of hand drawn mockups, whichever is easiest to identify key ideas of the user interface.*

## 3. Server-Side Implementation

### 3.1 End-Point Descriptions

*A list of endpoints your server will implement, HTTP methods for the end points, and any parameters that the endpoint will accept to fulfill the request. e.g.:*

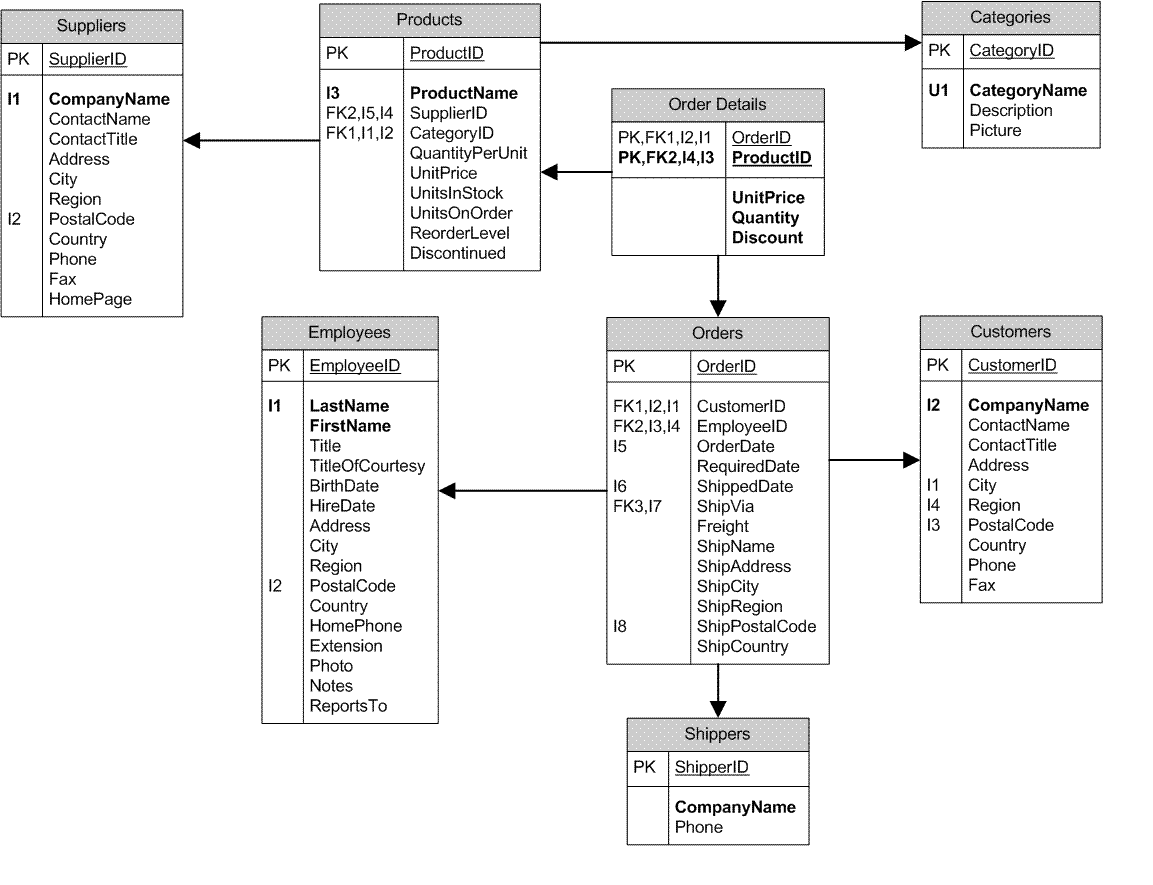
|  |  |
| --- | --- |
| **End-point** | **Response Format** |
| *HTTP GET - /todos* | *{ todos : [ { text : ‘STRING’, done : TRUE/FALSE, ID : NUMBER } … ] }* |
|  |  |

### 3.2 External APIs that will be consumed

*If your project uses APIs (e.g. twitter, google maps, etc), a list of the APIs that will be used, how the API will be used. API should have been thoroughly investigated to confirm it can do what you think it can do.*

### 3.3 Database Structure

*If your project includes a database, the structure should be described including tables with fields (for relational dbs), or documents and attributes (for non-relational dbs). If your table does not include a database, describe any persistent storage that you will be using (e.g. flat-files, directories). If desired, diagrams can be drawn using tools such as* [***draw.io***](https://www.draw.io/)

**

### 3.4 Authentication/Authorization and Security

*Description of any user profile/login functionality that you need, as well as any authentication/authorization that your project will use.*

User can sign in and view games they have created to keep track

## 4. Project Roadmap

### Phase 1

*A subset of the features that you have listed in Section 1.4.* ***This is what you are committing to have completed for demo day.******Your project will be marked based on what you commit to in Phase 1.*** *Think about what you can reasonably complete within 8 - 10 working days, and what will demonstrate your skills in an interesting way to the audience on Demo Day. Note that the more clearly you have planned out your project in the preceding sections, the easier it is to focus on implementing your project step by step.*

*You must scope out Phase 1 as accurately as you can as a 1 week “mini-sprint”. Assuming that you begin your project on Tuesday of Week 9, and everything is due on Wednesday of Week 10, that gives 7 working days to work on your final project. The more detailed you can be, the more you can be assured that you can complete your plans. Also note that you may encounter “problems”, consider having 1 day of buffer time to account for unexpected problems. e.g.)*

The first phase will include a functional site a database of players can be stored. Players can mark their availability and can join games that are scheduled for a day. Players will be able to choose an available position which they will commit to playing.

|  |  |
| --- | --- |
| **Day** | **Goal** |
| *1 (Wed)* | *Create database and design models* |
| *2 (Thurs)* | *Build server boilerplate and hook up to database* |
| *3 (Friday)* | *Define all Endpoints in express, connect endpoints to any external web api* |
| *4 (Mon)* | *Implement functionality of endpoints - create JSON response and verify endpoint responses* |
| *5 (Tues)* | *Build React App overall structure and high-level components* |
| *6 (Wed)* | *Connect smart components to end-point APIs* |
| *7 (Thurs)* | *Test/debug end-to-end functionality of app* |
| *8 (Fri)* | *Implement CSS/Styling* |
| *9 (Mon)* | *Code Clean up, last minute testing/debugging* |
| *10 (Tues)* | *DEPLOYMENT* |
| *11 (Wed)* | *DEMO DAY* |

### Phase 2

*This is what features you may complete if you have extra time, or what you would work on next after Phase 1 is complete.*

Phase 2 will have a landing page for the website and will have a user profile page, where users can store their information to make creating games and joining games easier

Another feature would be for players to join in as a substitute in event of a cancellation or injury

### Phase 3

*This is nice to have features or future enhancements that are more complex, or take the project to a more completed, production-ready state.*

Some fields must be rented out. Thus, the game organizer would be required to pay a fee which she/he would then collect from all participants.

Phase 3 to include a feature using Splitwise API (<https://dev.splitwise.com/#tag/other>) to track what each player owes the organizer and would remind players if they owe an organizer.

Phase 3 will also a feature a voting component allowing players on a team to choose a captain to choose formations

## 5. Demo Day Information

Please fill out the Google Forms that will be sent out via Slack. This information will be used to complete your presentation and to showcase your profile on the BrainStation website.