***Ballpark Bookie - Software Development Plan***

**4.1 Plan Introduction**

This software development plan provides the details of the planned development for the Ballpark Bookie CSCI which provides an application to provide users with general and specific predictions about Major League Baseball games.

The application’s final output will be a list of the predicted winners of upcoming Major League Baseball games. The user will be able to search through a list of all of the upcoming games and be presented with a prediction for who will win each individual game. The application will provide this information through a series of steps performed in the backend. The application will need to create, update, and maintain a database of MLB statistics on players and teams. The statistics will be pulled from an API that provides access to such data. The predicted winners will come from an algorithm, which is unique to Ballpark Bookie, that takes into account the enormous amount of data present in MLB statistics and weigh the probable influence of the data on the outcome of each individual game. Ballpark Bookie will appeal to a range of potential users, ranging from novice betters to avid sports fans, that want an easy-to-use, accurate application that can predict the outcomes of games. People may want to know the likely outcome of a game based on their interest in the sport, investments in betting, participation in fantasy baseball, or desire to attend a game where their favorite team wins.

Development will include multiple different the creation and linking of multiple layers. The application will require a front-end to be designed as a web application that will display the final predictions calculated by the algorithm. The front-end will need to communicate with a backend that performs multiple functions. The backend must maintain and reference a database of MLB statistics and the calculated outcomes. The statistics will be collected by the communication between an API and our application. The predicted outcomes will come from an algorithm that parses through the database of statistics and calculates the most likely winner based on weighing the implication each statistic tends to have on the outcome of games. The algorithm will be developed based on observations of the relation between past games’ statistics and outcomes and be tweaked based on the accuracy of the algorithms calculations. All of these different parts will work together in the creation and use of Ballpark Bookie.

Sub-Task Completion Dates:

* 25 September 2019 – Front-End of Web App Completed
* 6 November 2019 – Project Preliminary Design Completed
* 23 October 2019 – Database w/ Statistics from API Completed
* 13 November 2019 – Final Prediction Algorithm Completed
* 27 November 2019 – Front-End Displaying Basic Calculated Predictions Completed
  + Ready for Alpha Presentation
* 11 December 2019 – Front-End Displaying Advanced Calculated Predictions

**4.1.1 Project Deliverables**

Our team has access to all the data from the mysportsfeed.com API. There are detailed stats for all major American sports and we will use their baseball data. The user will get access to accurate predictions on MLB games.

1. The first part we will work on is the algorithm that will give us the percent chance that one team wins over another team. We plan on having this complete by 10/22. It is important to get this one done first because we can later call on the percent chance of a team winning to help predict other outcomes.
2. We will want to setup our database by 10/23. It will not be complete but we will be using it to store data so that we do not have to make an API call every single time.
3. Next we will work on the algorithm that predicts scores. The completion of the first algorithm will help greatly with this. The plan is to have this done by 11/4. This algorithm might not be as accurate as the previous algorithm because there is a huge variety of outcomes (instead of just two) so we might have to come back to this later in the development stage to keep working on it and making it better.
4. Next we want an algorithm that will predict how a team will do in an entire season of baseball, not just one game. It should be finished by 11/13. This will be the most difficult algorithm so it will be important to get everything above done at their respective due dates.
5. Lastly we want to implement all this into our webpage. There will be places within this web app to find where the user can see the odds of their team making it to the playoffs or world series. The user will be able to see the projected score of a game and the percentage chance of a certain team winning. Implementing should be a lot easier to finish than the parts before it. We hope to have this done by 11/27.

**4.2 Project Resources**

This section provides information about the resources required to create the application Ballpark Bookie. The section is split into two parts, one describing the hardware and the other describing the software.

**4.2.1 Hardware Resources**

Ballpark Bookie is a web application and will thus be accessible on any device through a web browser. Our application requires only Google Chrome and connection to the internet to work. Therefore the user will need OS X Yosemite 10.10 or later to run Chrome. In order to access the internet the computer needs 300 MB (500 MB recommended) of free hard drive space, 256 MB or higher of memory, G3 or higher, or Intel Processor, and an updated operating system.

**4.2.2 Software Resources**

Software Tools used for Ballpark Bookie:

* Visual Studio Code
* React
* Bootstrap v4
* AWS
* MySportsFeed API

**4.3 Project Organization**

The project will be divided into four major parts/functions. The four parts are: front-end, database, API interaction, calculation (using developed algorithm).

The **front-end** will be a clean, clutter-free web-page that displays all relevant information to the user in an easy to understand manner. The front-end will show the various teams and MLB games for which predictions are available. The development of the front-end will include creating a Bootstrap based web-app with forms for input of desired teams, tables displaying predictions, and buttons allowing the user to submit requests for information/predictions. Traditional HTML, JavaScript, and node/Bootstrap will be used in this development.

The **database** will be a simple way to store vast amounts of data present in MLB statistics. The database will store the current recorded statistics per day/week and therefore be updated on a daily/weekly basis. The database’s development will include the creation of the underlying data structure, the choice of DBMS, and the various queries required to pull the relevant data to calculate predictions.

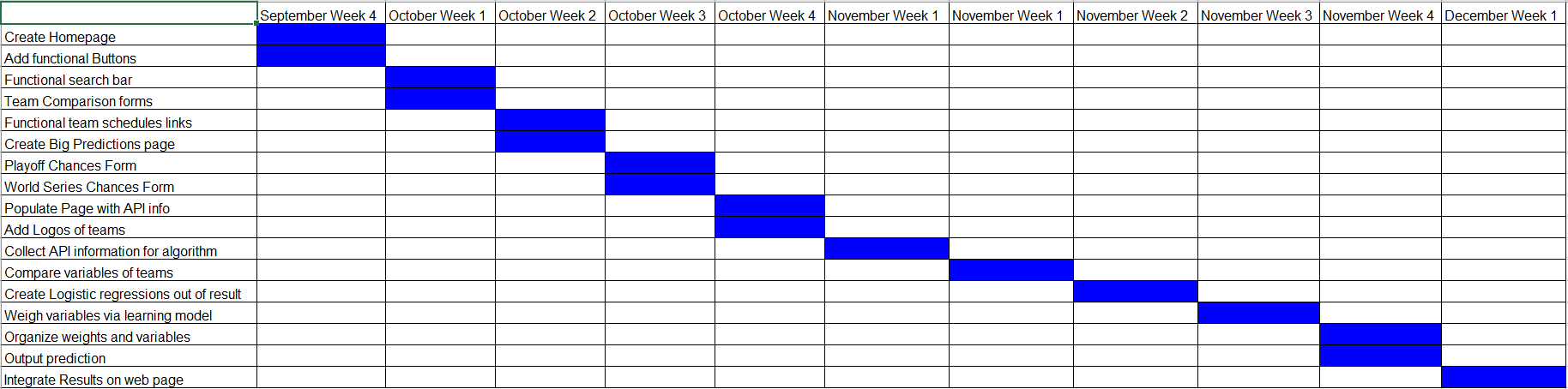
The **API** interaction will take place in the backend. An API for a service that provides MLB statistics will be called on a regular basis to update the application’s database. The API calls will be less frequent and therefore be less taxing on the application, as a whole, through the integration between the API and the database. The API will need to be understood and called in an efficient fashion so that there is no superfluous calls or information collected.

The development of an **algorithm** to create predictions is the most integral part of the app. The algorithm will need to be developed by assigning initial weights to each recorded statistics and calculating the predicted winner of a given game. The algorithm will be further perfected through analysis of its accuracy on completed games and tweaking weights of statistics on predictions. Logistic regression will be used in the algorithm and be an integral part of the generation of predictions.

**4.4 Project Schedule**

This section provides schedule information for the Ballpark Bookie project.

**4.4.1 GANTT Chart**



**4.4.2 Task/Resource Chart**