The overall objective of this project is to create a website that will help the men’s basketball coaches at Drake University more easily identify which players have stronger chemistry, and ultimately play better together, while on the court. This website will help the coaches to better identify their starting five players as well as who to sub in and out during games. It allows the coaches to select which basketball season they are wanting to pull data from as well as which specific game statistics they want to be utilized in the analysis. The website then displays aggregated statistics for groups of players that played together during the season chosen by the coaches. Those aggregated statistics displayed on the website are calculated based off play-by-play statistics for each game that are pulled from the Drake University Men’s Basketball website [1].

In the future, the aggregated statistics will be run through various machine-learning algorithms to actually predict which players would play best together and offer recommendations to the coaches of which players should be on the court at certain times throughout various games.

The idea for this project came directly from the men’s basketball coaches at Drake University when they were touring the Computer Science Department with a prospective student. In the future, the coaches will be involved a lot more with the design of the website as well as the specific tools they would like available for them to use on the website.

All the data used in this project is pulled directly from the statistics archive on Drake University’s Men’s Basketball website [1]. The urllib package in Python is used to open up the HTML for a specific webpage in an archive on Drake University’s Men’s Basketball website [1]. The BeautifulSoup package is used to help parse and clean up the HTML so it is easier to read. Web scraping is used as a means of collecting data for the following four parts of this project:

- Compiling a list of the seasons with available game data.
- Creating a collection of all the games played in a season.
- Creating a list of players on the men’s basketball team at Drake University during a specific season.
- Compiling the play-by-play statistics for each half of every game.

These are used to help aggregate and store all the statistics for each group of five players playing together in a Python dictionary.

The data manipulation phase occurs after all the data has been collected and stored in various data structures, typically lists. In this phase, the main function in the Python code runs through each game’s play-by-play statistics, keeps track of which five players were playing, and stores each group of players as well as their corresponding statistics in a dictionary. Below is a diagram showing how the main Python function manipulates the data, so the desired aggregated statistics can later be displayed on the website.

The website that holds the tools coaches can use to analyze the chemistry between various players is built using Flask, which is a micro-framework commonly used for building web applications in Python. There is only one Python script that needs to run on the computer and the tool on the website to work. Below is an activity diagram that shows how the final webpage with aggregated data for each group of five players that played together during the games is created:

The data manipulation phase occurs after all the data has been collected and stored in various data structures, typically lists. In this phase, the main function in the Python code runs through each game’s play-by-play statistics, keeps track of which five players were playing, and stores each group of players as well as their corresponding statistics in a dictionary. Below is a diagram showing how the main Python function manipulates the data, so the desired aggregated statistics can later be displayed on the website.

The website that holds the tools coaches can use to analyze the chemistry between various players is built using Flask, which is a micro-framework commonly used for building web applications in Python. There is only one Python script that needs to run on the computer and the tool on the website to work. Below is an activity diagram that shows how the final webpage with aggregated data for each group of five players that played together during the games is created:

The website then displays aggregated statistics for each game in the basketball season chosen. It also helps create the full URLs that are used to navigate to the webpage with play-by-play statistics for each game.

This is used to help determine the last five away games, last five home games, or last five games, regardless of where they were played, in the specified season. It also helps create the full URLs that are used to navigate to the webpage with play-by-play statistics for each game.

This is used to help navigate to the webpage with play (i.e. foul, successful free throw, etc) by using stored functions.

The data manipulation phase occurs after all the data has been collected and stored in various data structures, typically lists. In this phase, the main function in the Python code runs through each game’s play-by-play statistics, keeps track of which five players were playing, and stores each group of players as well as their corresponding statistics in a dictionary. Below is a diagram showing how the main Python function manipulates the data, so the desired aggregated statistics can later be displayed on the website.

The website that holds the tools coaches can use to analyze the chemistry between various players is built using Flask, which is a micro-framework commonly used for building web applications in Python. There is only one Python script that needs to run on the computer and the tool on the website to work. Below is an activity diagram that shows how the final webpage with aggregated data for each group of five players that played together during the games is created:

Figure 1: Diagram displaying how data is manipulated and aggregated statistics are calculated for each group of five players.

Figure 2: Activity diagram for the website created to help basketball coaches better analyze which groups of players perform better.

After the data is manipulated and aggregated, the aggregated data is displayed on the final webpage, as shown in Figure 4. The process of displaying the aggregated data on the final webpage is explained in the system overview.

The website then displays aggregated statistics for each game in the basketball season chosen. It also helps create the full URLs that are used to navigate to the webpage with play (i.e. foul, successful free throw, etc) by using stored functions.

The data manipulation phase occurs after all the data has been collected and stored in various data structures, typically lists. In this phase, the main function in the Python code runs through each game’s play-by-play statistics, keeps track of which five players were playing, and stores each group of players as well as their corresponding statistics in a dictionary. Below is a diagram showing how the main Python function manipulates the data, so the desired aggregated statistics can later be displayed on the website.

The website that holds the tools coaches can use to analyze the chemistry between various players is built using Flask, which is a micro-framework commonly used for building web applications in Python. There is only one Python script that needs to run on the computer and the tool on the website to work. Below is an activity diagram that shows how the final webpage with aggregated data for each group of five players that played together during the games is created:

Figure 4: Final webpage that displays the aggregated statistics.

Figure 3: Initial webpage with tool for coaches to use.

REFERENCES