LoL Replay

Ashton Foulger, Kevin Xue
u1020531@utah.edu, kx6023@gmail.com
U1020531, U1261440
Repo Link: lol-replay

Background and Motivation

As longtime League of Legends players, we have always been interested in the relevant statistics pertaining to our time spent playing the game. Specifically, it would be interesting to create a project that is a deep dive on our performance, based on a plethora of inputs, that shows us a representation of our strengths and weaknesses.

As computer scientists we are able to use our skills to create a project that allows us to analyze the depth of our own games, and showcase a meaningful data visualization, using statistics generated by the Riot API using D3.js for front end rendering purposes.

The Riot API is extremely thorough, widely used, and is free to anyone with a Riot Account. This allows us to have confidence in the quality of data provided. Therefore, we can focus on the data analysis and visualization aspect of this project without needing to worry about the data quality. By having high quality data, we can create a thorough representation about a user's strengths and weaknesses. Therefore creating a “replay” to show players their historical trends with our data visualization project.

Project Objectives

The first question we want to answer is “What are match trends?”. This broad question will include data about match length, as well as provide insights about an individual’s performance based on the length of each game. Detailed match trends will give the player an opportunity to see how well they perform based on KDA ratios when compared to the time spent in a match.

- What are the match trends?
    - What is the number of games vs minutes played?
    - What is the Kill to Death to Assist (K/D/A) ratio by the length of the game?

The second question we want to answer is “When are you most active?”. This question aims to gain insights on how real world factors impact a player’s performance. Some of the data that will be required to answer this question include, number of games per day (ranked and unranked), what times do you commonly play, and what times correlate with wins the most. These data values allow users to see how they perform based on real world factors.

- When are you most active?
    - The amount of games per day?
    - The amount of games played per day (Ranked games)?
    - What hour of the day do you play the most?
    - What hour of the day do you win the most often?

The third question we want to answer is which champion does a player play the best. By providing information about that player's champion stats such as win rate, and mastery. A player

can see what champions they perform the best with when, as well as view other champion specific information, such as mastery.

- Champion specific statistics
    - What champions do you have the highest winrate on (requiring some minimum number of games)?
    - What champions do you have the highest mastery in?

Data

The Riot Games API provides data that is easily accessible by providing a number of query parameters. These parameters include user ID, match ID, league ID, and many more. By using these parameters, we can specify the type of data we want to interact with as well as how specific we want the data to be (player specific, league specific, etc). All that is required to access this data is to have a Riot API account which has three tiers. These API tiers are Development, Personal, and Production. The Development tier allows for users to interact with the API, however refreshes every 24 hours. The Personal tier allows for users to interact with the base API, and does not refresh every 24 hours. Finally, the Production tier allows for users to query for the base API, have increased rate limits, and also have personal requests that are not available through the base API. For our project, we will be using the Personal tier, however if permission is not granted, the Development tier would be functional for our use case as well.

Link : https://developer.riotgames.com/apis

Data Processing

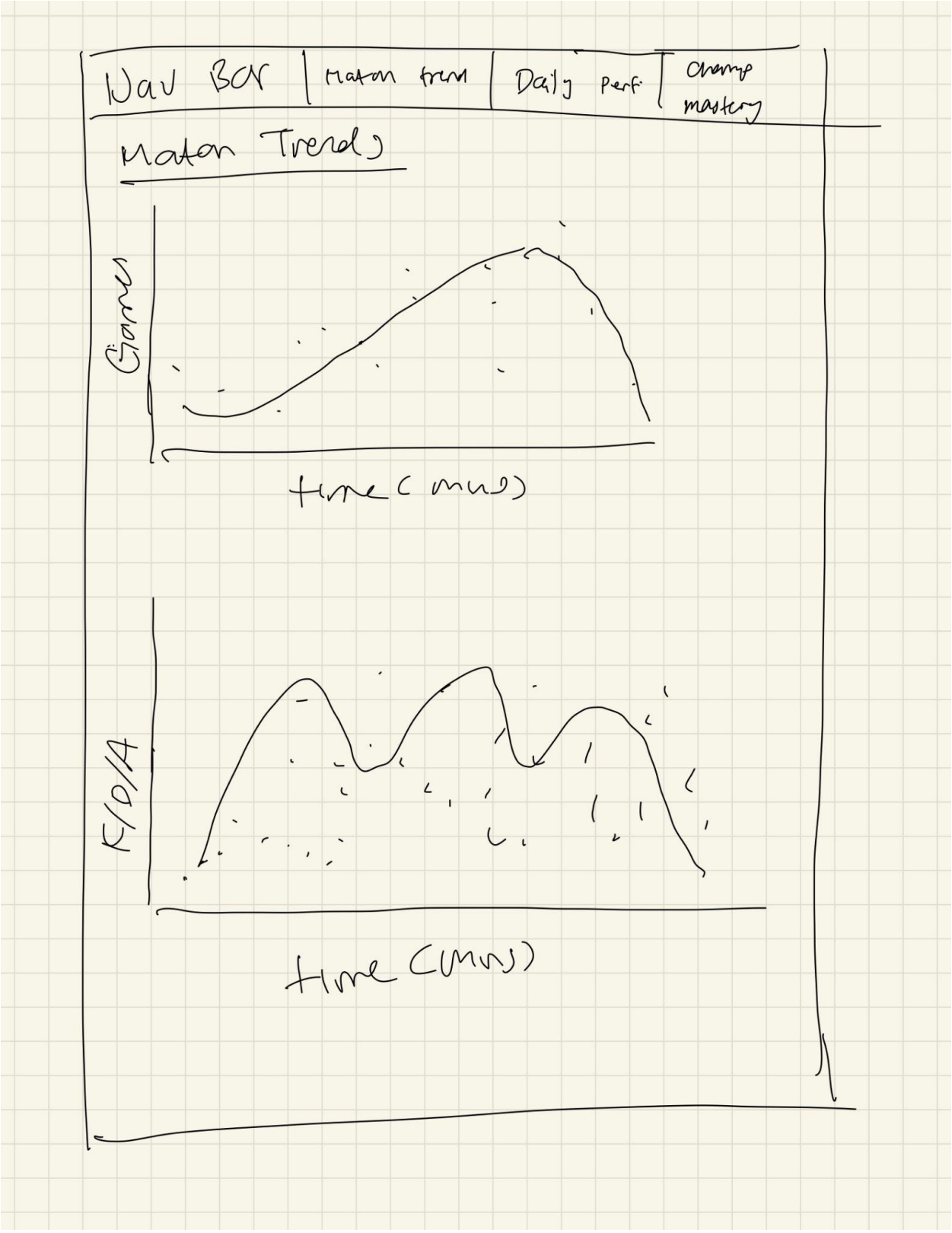
In order to process the data from the Riot API, we will need to parse the data into separate data collections in order to perform relevant calculations on the data. This will require us to ignore unnecessary data that does not pertain to the questions we are attempting to answer. Therefore, data parsing will be required in order to gather data from a specific user when queried by the API.

Information we need to derive from a players match history will go as follows:

- Number of Games
- Number of Games ending at various minutes
- Number of Kills, Deaths, Assists per match
- Match Date and Time
- Ranked Match Data and Time
- Games Won at a specific time of day.
- Gather matches where players played specific champions.
- Gather players champion mastery points

We plan to organize this data by having a function or set of functions to call the API and gather the required information for the data visualizations. By parsing the data through these functions, we can then store the data in various collections in Javascript and then convert these collections into a file type compatible with d3.js.

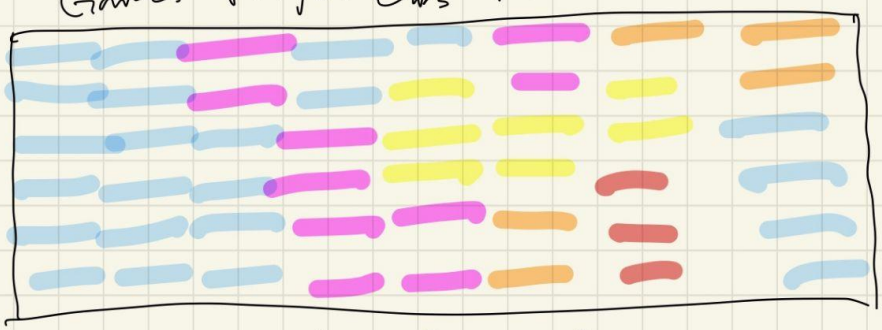
Visualization Design



Nav	Match trends	Daily Performance	Champ Mastery

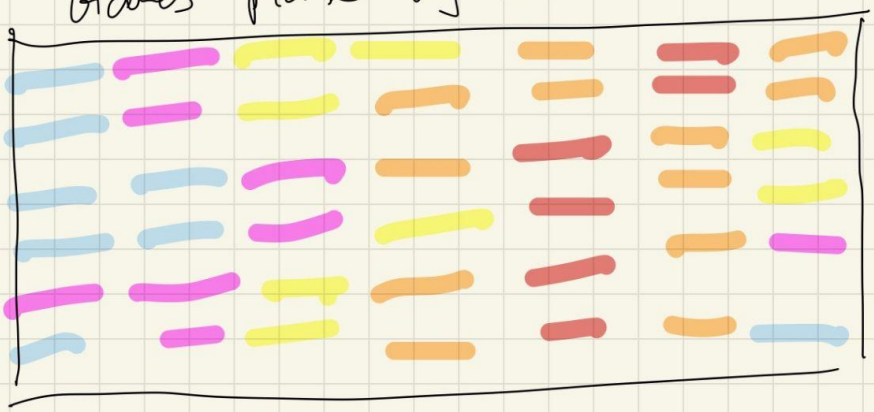
Daily Performance (heat maps)

Games Played (hours)

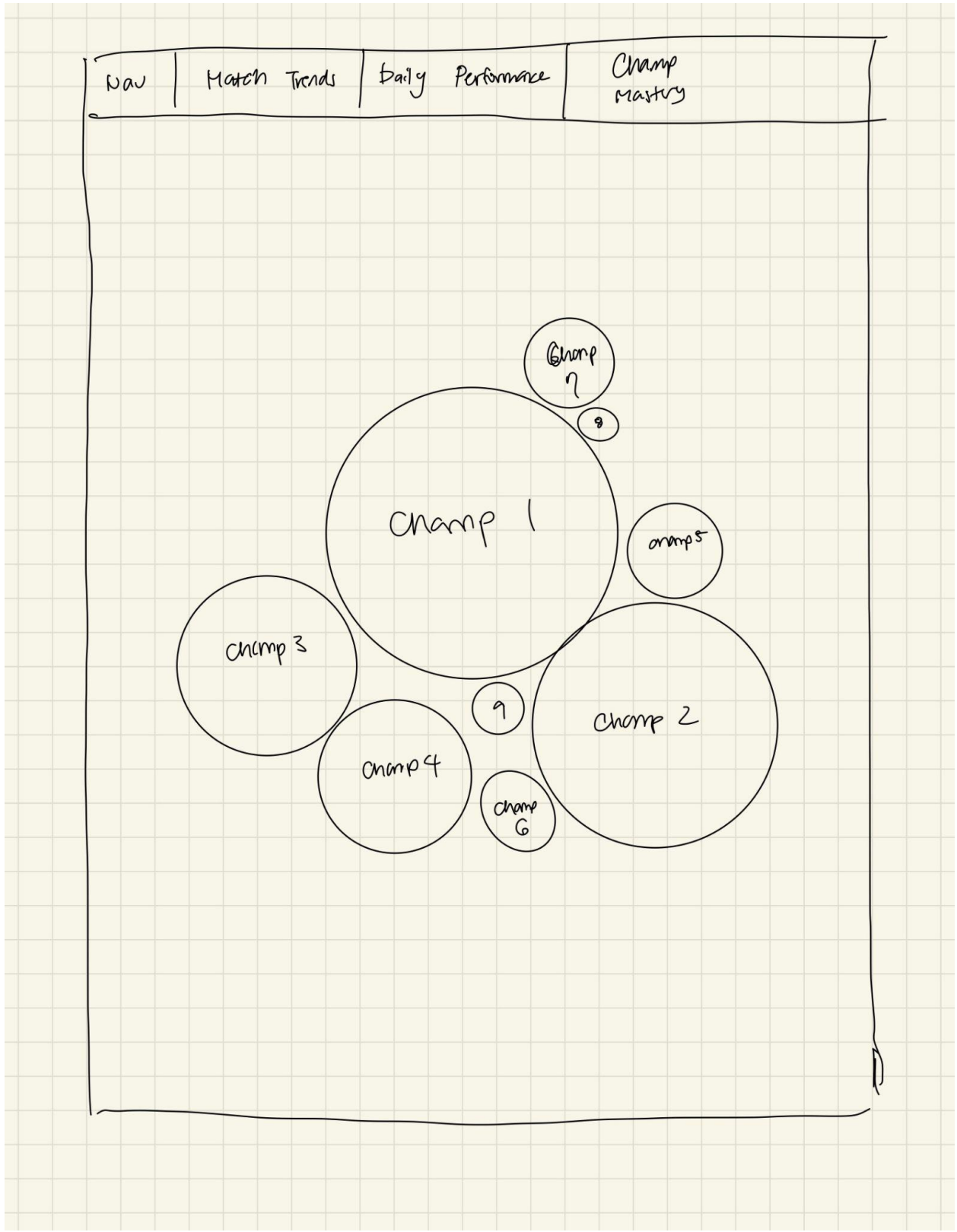


time (hours)

Games played by week



week


Nav
Match Trends
Daily Performance
Champ Mastery
Champ 7
8
Champ 1
champ 5
champ 3
9
Champ 2
champ 4
champ 6

Features

- Show players their number of games vs. minutes played in a well designed visualization
- Show players their K/D/A ratio to the number of minutes per match.
- Show players what time of day they are most active.
- Show players the amount of games played per day.
- Show players a champion mastery graph.

Optional Features

- Show players what champions they have the highest winrate on.
- Show players what champions they lose to the most often.
- Show players what champions they win with most when on their teams.

Schedule

Week #	Ashtons Tasks	Kevin's Tasks
5	Bootstrap React App	Implement Riot API Config
6	Begin Writing Data Parser	Begin Writing Data Parser
7	Data Parser Refinement	Data Parser Refinement
8	Fall Break	Fall Break
9	Graph for Games vs Minutes	Graph for KDA vs Minutes
10	Graph for Most Active	Graph for Games Played Per Day
11	Player Mastery Graph	Player Mastery Graph
12	Graph Refinement	Graph Refinement
13	UI Polish	UI Polish
14	UI Polish	UI Polish
15	Finalize Project	Finalize Project