

League of Legends Champion Mastery Viewer

Erik Gunderson & Darren Lin

April 26th, 2018

1 Introduction

In the popular game *League of Legends*, players play as champions in five versus five matches on a large map with three lanes, towers, and various other defenses. These champions, as they are used, gain champion mastery, which is akin to experience in various other games. Earning enough champion mastery will increase the champion's level, up to level five, wherein the player has to do very well with that champion in several matches in order to earn the right to bring that champion to levels six and seven.

Champion mastery, as a result, can give insight into how much a player has played as a particular champion. Our goal was to create a simple tool that allows users to easily view any player's champion masteries, which will give the user an idea as to how much a player has played each character, as well as infer how skilled a player is with particular champions.

The audience for this tool would be mainly League of Legends players, as they would be more familiar and more interested in knowing the favorites of other players, and could analyze their own champion masteries if they felt so. The tool is still simple and intuitive enough so that even people unfamiliar with League of Legends would be able to use the tool themselves if they felt interested in looking at information about people on League of Legends.

With making this tool, the main question we ask is the following: Would displaying the champion masteries of players in League of Legends in a visual format give more insight as to the skill with which that player plays those champions? And not only that, would this format give insight as to what kinds of champions that player is good at? We believe that through a simple visual format like a bar graph, a user will be able to easily answer those questions.

2 Related Work

Observing trends in the usage of particular champions as well as data regarding the skill of players, while not a subject seen in academia, is a concept seen in many tools created by hobbyists and fans of *League of Legends*. MasteryPoints and ChampionMasteryLookup are two popular tools for performing the task our tool does, and while they provide information on champion mastery points and level, they do not give a lot of filtering and sorting options, something our tool aims to fix.

3 Design Evolution

In the initial idea, this project was to involve getting ranking information on players and showing the total player population, in a streamgraph format. We moved on to a bar graph idea after changing from the initial idea, and drafted up a basic sketch of the UI we wanted, as shown below. A basic prototype of this UI was implemented in Matplotlib, though missing all of its planned interactive features as well as lacking the champion images to be used as labels.

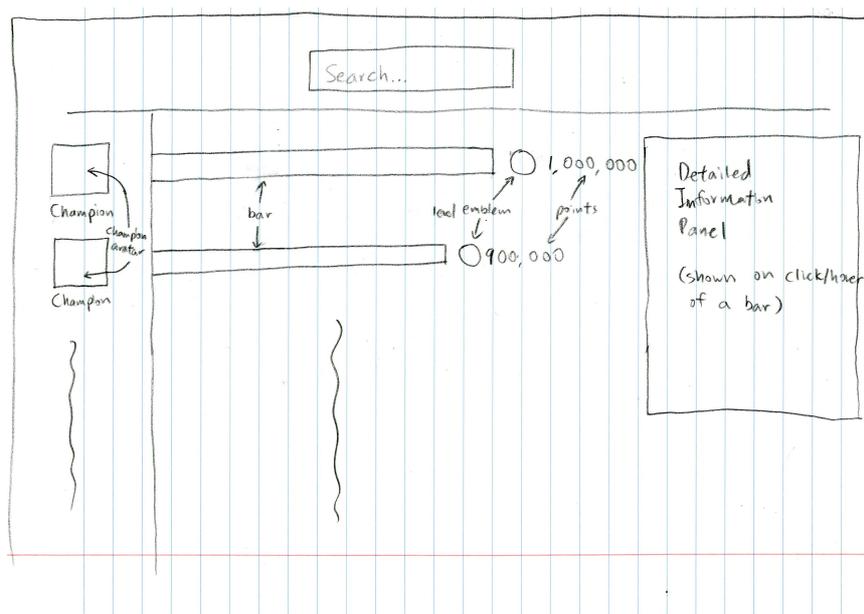


Figure 1- Draft of the final visualization UI. Missing planned filter features and does not show plan to have sticky UI on the detailed information panel.

Figure 2 shows the finished visualization. There is a search bar at the top, which allows the user to search by summoner name. The bar represents the number of points that champion has earned, while the symbol to the right shows what level that character

has obtained via its symbol. The symbols themselves have tooltips explaining what they mean for users who do not know what they mean. There are also checkboxes at the top that filter out champions by role. The filter is an OR filter, so the user can select all of the roles they want to look at. Figure 2 is using a static set of data for testing, as at this point searching for a summoner by name had not been implemented into this version.

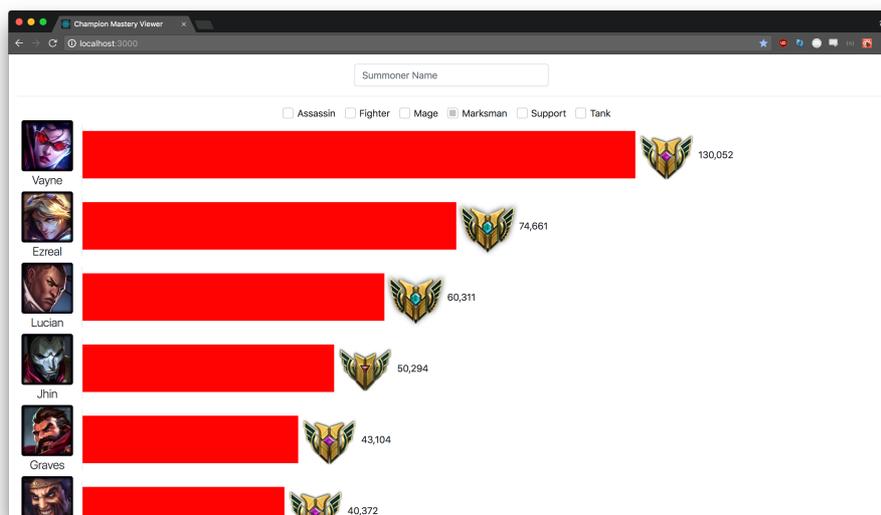


Figure 2- The final visualization in the last testing phase, making use of the filter to only show Marksmen.

4 Feedback

In class, a rough Matplotlib diagram (shown in Figure 3) was displayed, showing the basic outline of the planned design, but without any of the interactive features other than allowing the user to search for a player they wished to look for. The five people that were met in class were each asked if the general format was effective in displaying the information. Everyone agreed that a bar graph format was a good method of displaying the information, and various features were recommended, like adding scrolling, filtering champions rather than always picking the top ones, and adding coloring to the bars to differentiate levels or champion types better.

There were a few other suggestions, such as being able to compare two or more players' information with each other, as well as seeing if the user could see the champion masteries of the player whose information they are looking at. The former would be difficult to arrange well with the UI, though not impossible, but it will not be included in the final version. The Riot Developer API does not give any information about friends,

so the latter would be impossible. The general feedback, at least, was useful for getting the final implementation figured out.

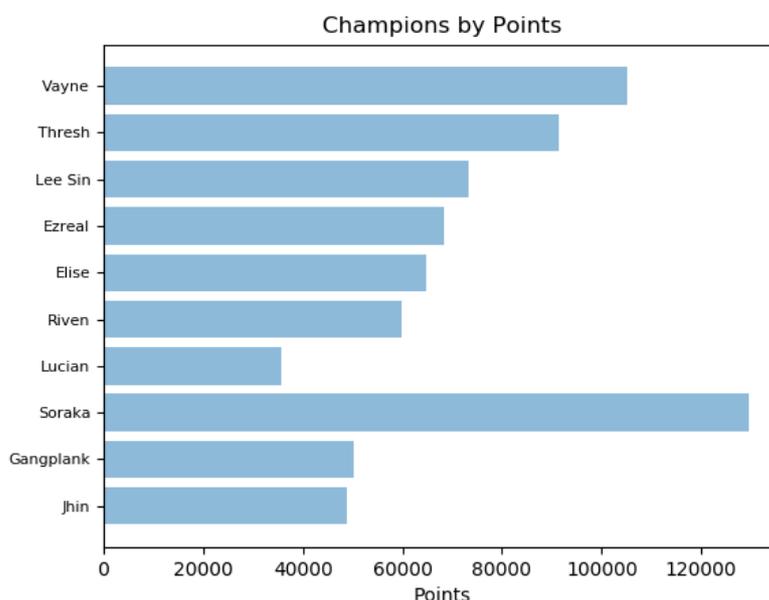


Figure 3- Visualization for debugging shown in class, made in Matplotlib. For testing, the user could input the number of champions with highest mastery, sorted by level first, then mastery points.

5 Implementation

Our final visualization makes use of the Riot Developer API to access three specific endpoints: summoner, champion-mastery, and static-data. The summoner endpoint contains information about summoners (players) and can be used to search for summoners by their ID or their summoner names. The champion-mastery endpoint is what actually contains the lists of all of the champions a summoner has used, and requires a summoner ID to search through. The static-data endpoint is what contains all of the information about every champion in the game, including their roles and links to their image files.

When a user searches for a summoner by name, the tool uses the summoner endpoint to obtain the summoner's ID. This ID is then used with the champion-mastery endpoint to obtain the champion mastery list, which lists each champion by their own ID. This list then cross-references the static-data list of all champions to get the names and roles of each champion on the list for the purposes of grabbing their names, portraits, and roles.

With the information gathered, it then visualizes it using ReactJS and Bootstrap, two

libraries that assist with creating UIs, resulting in the final visualization in Figure 4. The filters use the gathered role data from the static-data endpoint to filter out the champions requested.

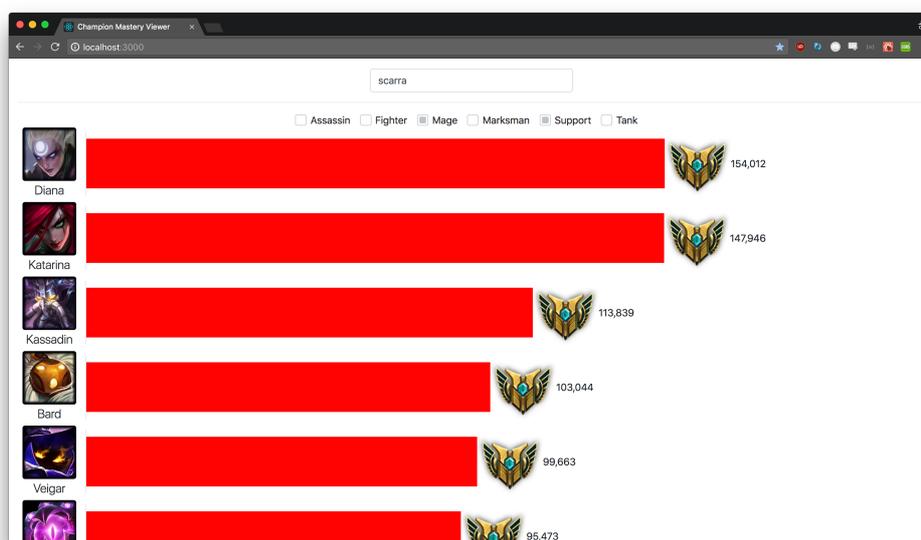


Figure 4- The final visualization, searching for the summoner “scarra” and filtering by mages and supports simultaneously.

6 Proposed User Study

If a user study were to be undertaken for this project, it would involve determining how intuitive the tool is and what information can be determined from using it. We believe that the tool, as it exists right now, will be more intuitive for players of the game, but less-so for people who do not play the game. In order to determine this, we would need a fair amount of people to use the tool, ranging from about 70 people to about 100 people. Preferably, there will be a decent ratio of people who are not familiar with *League of Legends* to people who are familiar with *League of Legends*. Most of these people will be a fair mix of people in the 20 year old to 30 year old range, though that is not a hard limit in any way. The only real important thing would be to have a mix of both players and non-players.

The researchers would include ourselves and some people who are more familiar with *League of Legends* and would be able to design questions based off of the kind of information present in our tool. Their role will be to look for information on players that would be useful in designing the tasks to give to the study participants. Questions that might be asked would pertain to things like what a summoner’s preferred

and least preferred champion is, what roles a summoner tends to play more or less of, and comparing how much two different players use certain champions.

The experiment itself will involve each participant being first asked about their experience with League of Legends. Regardless of if they are experienced or not, every participant will be given an explanation about champion mastery, the way champions earn mastery levels, and a very basic explanation of the tool they are about to use. After this explanation, the participants would be given some amount of time to work with the tool freely for about fifteen or so minutes, before giving them the kinds of questions mentioned above. These questions will involve the user looking up players given to them by the person overseeing the experiment, and seeing as the data might change between users, the tool will be using a cache of data rather than live data, so as to standardize the answers to questions.

The rapidity and correctness of the answers will be measured for each general type of question, those being comparisons between players, champion preference-based questions, and role preference-based questions. Afterwards, each participant will be asked to give their comments on the tool and what they thought of its design and usability, and to point out anything that might be confusing or unintuitive.

All participants who complete the tasks will be assigned afterwards to a group based off of their familiarity with the game, and all resulting data will be split up between groups to compare how each does with each other in regards to how speedy they were as well as how correct they were. Their comments will be considered without regard to group, as anything that could help improve the design or add more useful features is welcome.

7 Work Division

Darren created a basic framework for grabbing player information in Python using the found CSV files for the initial idea, then worked on creating the final visualization in JavaScript using ReactJS and Bootstrap. Erik focused on creating the test visualization in Matplotlib, while also determining how data could be obtained and parsed from the summoner, champion-mastery, and static-data endpoints.

8 Conclusion

At this point, our visualization is feature complete and allows a user to look up any summoner from *League of Legends* and allow them to see the champion masteries of all of their champions. Given what other tools of this kind are able to do, there is room for improvement, not only in regards to the design of the visualization, but also to features

that allow more direct comparisons, as well as other forms of sorting, filtering, and other features. At the least, this champion mastery viewer allows people to make direct comparisons simpler.

References

“Riot Developer API Documentation.” *Riot Games*. N.p., n.d. Web. 29 Mar. 2018.

<https://developer.riotgames.com/api-methods/>

“Mastery Points.” N.p., n.d. Web. 26 Apr. 2018.

<https://www.masterypoints.com>

“Champion Mastery Lookup.” N.p., n.d. Web. 26 Apr. 2018.

<https://championmasterylookup.derpthemeus.com>

“ReactJS.” *React JS*. Facebook Open Source, n.d. Web. 19 Apr. 2018.

<https://reactjs.org>

“Bootstrap.” *Bootstrap*. N.p., n.d. Web. 19 Apr. 2018

<https://getbootstrap.com>

“Champion Mastery Icons” *League of Legends Wikia*. N.p., n.d. Web. 26 Apr. 2018.

http://leagueoflegends.wikia.com/wiki/Category:Champion_mastery_icons