

CS 115 Release Plan

LoL Stats.

Product owner: Logan Collinwood.

Team members: Griffin Meyer, Michael Le, Johannes Pitz, and Brandon Chai.

LoL Stats – a League of Legends analytics web application built on Laravel, React, and D3.

Goal:

To build an eSports web application providing analytics for League of Legends gameplay by fully utilizing the Riot Games API. Using the data provided, and applying our own algorithms to break down player stats and provide rich, meaningful ways to improve your gameplay.

Task Listing:

This section lists the user stories, in priority order from most important (top) to least important (bottom). Within each user story, there needs to be a list of tasks required to implement the user story, along with the time estimate for each tasks (preferably less than or equal to 6 ideal hours). This should look like:

Note:

All time estimates are PER group member, as they include building out the API and frontend to support these features.

Sprint 1:

- As a developer I want to be able to work on the project.
 - 3 points
- As a developer, I want to be able to track analytics about users accessing my website in real time
 - 1 point
- As a player I want to access the website online.
 - 3 points
- As a player, I want to be able to see stats about myself to be able to analyze and improve my gameplay.
 - 21 points
- As a player loading into a game, I want to quickly view stats so I can outplay my opponent.
 - 5 points
- (As a mobile app user, I want to be able to use the site quickly on my phone so that I don't have to tab out to see stats. I want to be able to view the website on my mobile phone, so the interface needs to be responsive)
 - 2 points
- As a teammate, I want to be able to see summoner statistics filtered for a specific role so I can recommend them what to improve on.
 - 5 points

Sprint 2:

Match Analysis

- As a player, I want to see champion picks, bans, runes and masteries selected for a match
 - 5 points
- As a strategist, I want to see what people do at a certain time frame so that I can strategize the best way to win.
 - 13 points
- I want to see farm, damage, and other relevant match data at a specific time in the match
 - 8 points
- As a player, I want to analyze ward placement and it's benefit by viewing ward placement over time and locations on d3 heatmap
 - 5 points
- As a player, I want to see every kill logged and displayed on an image of the in game map
 - 2 points
- As a League of Legends player, I want to be able to pick great meta-champions so I can win solo queue.
 - 5 points
- As a LoL mentor, I want to show my students stats so that they can see how wins relate with game stats.
 - 7 points

Sprint 3:

Community Section

- As a frequent LoLStat user, I want to view live Twitch Streams in a section of the website, so that I don't have to load a new page.
 - 8 points
- As a LoLStat user, I would like to watch a Twitch Stream on the community section of the website so I can also view stat analysis of the players.
 - 8 points
- As a trendy person, I would like to read interesting and noteworthy articles bookmarked and catalogued so that I can keep up-to-date with news.
 - 8 points
- Finalize product deployment strategies (branding and domain), build community endpoints
 - 8 points

Team roles:

Logan Collingwood: Product Owner, Backend Lead

Griffin Meyer: ScrumMaster (Sprint 1), Backend {full stack as needed}

Brandon Chai: Frontend Lead

Johannes Pitz: Frontend {full stack as needed}

Michael Le: Frontend {full stack as needed}

Initial task assignment:

Logan Collingwood: Work on story 2, build up schema

Griffin Meyer: Work on story 2, identify what needs to be stored

Brandon Chai: Work on story 1, design application layout (unified frontend design across modules)

Johannes Pitz: Work on story 1, given player JSON start using D3 to create charts

Michael Le: Work on story 1, pull in frontend static assets (item icons, etc)

Sprint schedule:

Week 2: release and sprint planning; setup environment/tools; initial presentation

Week 3: finalize release and sprint planning; sprint 1

Week 4: sprint 1

Week 5: sprint 2

Week 6: sprint 2

Week 7: sprint 2

Week 8: sprint 3

Week 9: sprint 3; wrapup, final presentation

Week 10: final presentation; demo (acceptance test)