

2020



# UNIVERSITY OF LINCOLN

## “BUILDS OF LEGENDS”

Design & Development of a user-oriented website to help newer players learn the game “League of Legends”

### Abstract

The overall aim of this project was to produce a user-friendly website based on the design of League of Legends on which experienced players can submit advice to other players or aspiring players, creating a ‘knowledgebase’. The primary field of this project is Human-Computer Interaction (HCI) and therefore users and design were a priority throughout. This project is also relevant in the field of web development as the latest web-based programming standards have been used to code a fully functioning, responsive website.

## Contents

Introduction .....	3
Literature Review.....	4
Literature Exploring the Problem Domain .....	4
Exploring Player Experience in Ranked League of Legends (Mora-Cantallops and Sicilia, 2018) ...	4
Mechanics and Metagame: Exploring Binary Expertise in League of Legends (Donaldson, 2017) ....	5
Literature Regarding Existing Solutions .....	6
A Tutor Agent for MOBA Games (Nascimento Silva and Chaimowicz, 2017).....	6
Using API Data to Understand Learning in League of Legends (Gerber, Sweeney and Pasquini, 2019) .....	7
Methodology.....	8
Software Development .....	8
Project Management .....	10
Gantt Chart.....	10
Trello .....	11
Microsoft Teams .....	12
Toolsets and Machine Environment .....	12
Programming Languages.....	13
Hosting .....	15
Research Methods .....	16
Design, Development & Evaluation .....	17
Requirements Elicitation, Collection & Analysis.....	17
System Requirements .....	22
Design.....	22
Development.....	31
Testing.....	38
Software Testing .....	38
Usability Testing.....	42
Potential Future Operation & Maintenance.....	44
Evaluation & Conclusion .....	45
Reflective Analysis .....	49
Demonstration Video.....	50
References .....	50
Appendices.....	52
(1) Requirements Elicitation Survey.....	52
(a)     Survey Post in Facebook Group .....	52

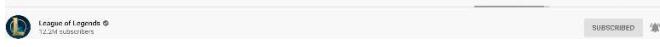
(b)	Survey Questions and Results .....	53
(c)	Summary of Survey Findings.....	56
(d)	User Stories .....	57
(2)	Designs.....	58
(a)	Surface Plane: Designs made in Adobe XD .....	58
(3)	Implementation Code.....	61
(a)	Basic layout and themes .....	61
(b)	Signup/Login System .....	63
(c)	Write/Submit Guide System .....	66
(d)	Displaying Guides stored in the Database to Users.....	70
(e)	Edit User Information System.....	71
(4)	SUS.....	72
(a)	The SUS Response Form .....	72
(b)	User Responses to the SUS Form .....	76

## Introduction

“League of Legends is a team-based strategy game where two teams of five powerful champions face off to destroy the other’s base” – ([leagueoflegends.com](http://leagueoflegends.com), n.d.). League of Legends is a free-to-play multiplayer online battle arena (MOBA) game which draws 8 million concurrent players, making it the most popular game on PC (Messner, 2019). As the concept of the game is very difficult to understand through words alone, it is highly recommended to watch the video attached below before reading this project. Whilst League of Legends offers a variety of game-modes, some of which occur temporarily or on special occasions, this project focuses on the most popular mode which takes place on a map called ‘Summoner’s Rift’ in which two teams of five compete to destroy one another’s base first. The five players take on 5 different roles (top, jungle, middle, ADC, and support) and each player chooses from a roster of almost 150 characters, known as champions. Each champion has a very different set of abilities and playstyles that usually make them more suited to some roles more than others. Throughout a match (lasting an average of approximately 30 minutes), players buy items with the currency they earn by completing objectives such as killing enemy champions and destroying enemy turrets. There are six item slots in total, so a complete build would consist of six completed items. With 5 roles, 150 champions, and over 175 items, there are thousands of different ways to play which is why the game can be very overwhelming for new players. Whilst a tutorial is provided, only the very basics are covered on just a small pool out of the large selection of available champions. For this reason, new players are often left ill prepared to play the game alone which can cause frustration due to a lack of understanding and progression. Furthermore, League of Legends is often played competitively and has a complex ranking system ranging from the Iron tier to the elite Challenger division. Research into the rank distributions later in this project reveals that the majority of the player-base fall in the lower ranks suggesting that the game does not provide sufficient guidance in helping players improve. Frustrations caused by these factors can lead to a toxic environment and a poor user experience. Therefore, the aim of this project is to create a widely accessible resource in which players can improve their game knowledge and subsequently their skill level. This resource will be a website in which experienced players can submit advice to newcomers or those seeking to improve. The website will be created using the latest standards in web-based programming, including HTML, CSS, SQL, PHP, and jQuery (JavaScript). The focus of this project is Human-Computer Interaction (HCI) and therefore user experience will be the priority of the site, ensuring adequate usability and accessibility. Furthermore, the design stage will be carried out in depth and carry as much importance as the development. Since this project uses user-centred design, users will be involved where possible and appropriate. The Coronavirus outbreak unfortunately significantly impacted and hindered this project and user involvement could not be carried out to the desired degree, and development was slowed due to difficulty in accessing resources and guidance. Nevertheless, this project was carried out to a high standard and offers a sufficient degree of uniqueness in this niche field.



<https://youtu.be/BGtROJeMPeE>



## Literature Review

To ensure the relevance and a sufficient degree of uniqueness of a project, it is important to investigate relevant literature to explore both the problem domain and previous/existing solutions. Gaining a deeper understanding of the issues at hand and the current situation regarding the game and player experience can illustrate the purpose of this project as well as highlight potential aims and requirements. As well as this, research into existing solutions can identify failings and therefore the need for a different solution such as the one proposed in this project.

### Literature Exploring the Problem Domain

[Exploring Player Experience in Ranked League of Legends \(Mora-Cantallops and Sicilia, 2018\)](#)  
This study conducted by Marçal Mora-Cantallops and Miguel-Ángel Sicilia explored player experience by surveying more than 400 players in the database of the largest eSports organisation in Spain and carrying out a series of tests based on two models; Player Experience of Need Satisfaction (PENS) and Social Presence in Gaming Questionnaire (SPGQ). The results of this study include showing how player experience differs across different levels of competence (rank) inside the game and that players perceive team collaboration as an essential factor for success. This research can provide insight into potential ways to improve player experience by highlighting what players struggle with and their opinions on the game. Figure 1 shows the results after players were asked for their perceived skill/rank tier i.e. which rank they think they should be based on the skill level they feel that they play at, and their actual ranks checked using API data. These results suggest that many players evaluate their own skill level at a higher rank than they actually are currently. For example, at the time of this study Bronze was the lowest possible Rank prior to the introduction of the Iron tier in 2019, and only 5.02% of the 400 players asked said that they felt like this was the right rank for them based on their skill level, whereas almost 10% more (14.16%) of those players were in the actually Bronze division. By comparison, Challenger is the highest achievable rank in which less than 0.02% of players achieve and none of the participants were in, and 6 of the participants said that they felt that they were good enough to be in Challenger. Furthermore, the fact that none of them were in the tier below, Master, shows that the highest rank those 6 people could have been in would be Diamond, yet they felt like they deserved to be at least two tiers higher. This data was gathered as this difference in desired and achieved rank can be a potential cause of frustration which would have a negative impact on player experience. Moreover, this study highlights players' need for guidance as it is likely that many do not have a good understanding of why they are not at a higher rank and therefore many would benefit from guidance how to improve their skill or game knowledge in order to achieve the rank they desire. The study also found that, when comparing two groups; the masses (90% in Bronze, Silver and Gold) and the elite (10% in Platinum and above), elite players felt more competent/skilled at the game as validated by their placement and therefore perceived the game as more balanced/fair. These are also factors likely to affect player experience which would suggest that players may get more enjoyment out of the game if their rank improved. Overall, learning about this study helped to validate the value of helping to players improve their skill and knowledge, and therefore rank, in order to improve player experience and their enjoyment. Furthermore, since players know and feel that their actions are dependent on their teammates' actions as much as their co-players actions are affected by their own, it is clear that every player on a team must perform to the best of their ability to ensure success. This awareness may suggest that players would be willing to help each other by submitting and reading help since the better the skill of the player-base, the better experience for each and every player.

Tier	API (Real)	% (Real)	N (Reported)	% Reported
Bronze	62	14.16	22	5.02
Silver	127	29.00	65	14.84
Gold	123	28.08	146	33.33
Platinum	84	19.18	122	27.85
Diamond	42	9.59	69	15.75
Master	0	0.00	0	0.00
Challenger	0	0.00	6	1.37

Figure 1: A Comparison of Player Perceptions and Actual Rank Distributions (Mora-Cantallops and Sicilia, 2018)

### Mechanics and Metagame: Exploring Binary Expertise in League of Legends (Donaldson, 2017)

This article examines the significance of two types of expertise in the popular multiplayer video game League of Legends. League of Legends play was analysed through the lens of a binary model of expertise, outlining examples of the in-game and out-of-game practices used by players in their pursuit of competitive success. The hypothesis of this study was that forms of out-of-game ‘metagame’ expertise are a particular form of importance in League of Legends. Previous studies suggest that video gaming does not take place in a sociocultural vacuum, instead, players often gather knowledge from previous experience of engagement with external resources in order to achieve success. Such research could include reading websites or watching videos and livestreams in order to learn more. In previous studies, two categories of gameplay have been referred to; the players’ handling of the game, which concerns the physical and motorial activities of gameplay, and the player’s meaning-making activities which involves their understanding of the game in terms of how the game is to be played, their role within it and the culture around it. Another concept identified in multiplayer games is the idea of ‘multifaceted proposition’ of learning, with the first stage being the accumulation of knowledge of the game rules and the second stage being the understanding of social rules, which may deviate from the rules or change depending on playgroup (such as regions, levels, and ranks). This research demonstrates how external sources are an essential part of improving in the game as well as how the in-game tutorials are not sufficient in providing the necessary game knowledge. Online games such as League of Legends are regularly patched, these patches often contain nerfs and buffs and alter the ‘meta’ which relates to the champions and items that are currently popular and considered as strong. In order to keep the game balanced and average win rates to around 50%, the game is frequently patched which can cause confusion among new players and make it difficult for existing players to keep up. Furthermore, as League of Legends can be known for its toxic player-base, making choices contradictory to the meta can result in a negative team attitude which can negatively impact player experience. As well of this, the release of the game came with a ‘Summoner’s Code’, which is an implicit set of rules created by the game developers, which state that players should always support their team and many players consider refusing to conform to the meta as breaking this code. These factors make the use of an external resource such as the website to be created in this project in order to learn about relevant metas very clear. Whilst the proposed guide site may not be able to help players improve their mechanical abilities, since this study shows that skills of this kind only make up around half of a players success, a resource to aid the other half relating to acquiring broader game knowledge will undoubtedly be helpful in improving players’ skills. Donaldson explains how “the inbuilt mechanisms [in League of Legends] for learning are not extensive enough to give players anything beyond a basic understanding of gameplay. New players receive only limited assistance from the game system in developing mechanical expertise—a brief tutorial mode offers players an explanation of the basic control scheme and other basic mechanics, but little attention is given to higher strategy. Each champion also comes with a recommended item build, although players must often deviate from this in order to compete effectively against certain enemy team compositions”.

Therefore, it is almost expected that for players to advance, they must find external guidance or gain experience slowly through long periods of trial and error. Since evidence suggests that players struggling to learn and keep up with the game is still a pressing issue, it is clear that another resource aiming to help players learn would be a welcome addition.

## Literature Regarding Existing Solutions

### A Tutor Agent for MOBA Games (Nascimento Silva and Chaimowicz, 2017)

This study presents an approach to helping novice players of MOBA games overcome problems such as steep learning curves and unwelcoming communities in the form of an Artificial Intelligence (AI) agent that plays alongside the player analysing their performance and giving tips about the game. Like Donaldson, this article also explains the issue with the in-game tutorial available and also explores the issues of the matchmaking system in which players of similar skill levels play together; “games often present a tutorial where the main gameplay is presented, aiming to teach newcomers how to play. Requiring the player to complete tutorial matches or playing against bots is also a very common practice, trying to assure that the player understands the concepts presented. However, in MOBA games, there are some features or roles that require the cooperation of more than one player. To group players of similar skills in a team, these games implement a matchmaking system, but, in a cooperative scenario, this approach may not perform well. The performance can be particularly poor in initial matches, as the combination of players with low skill or no knowledge of some features in the same team will not help them in learning the game”. For this reason, this study proposes an alternative method of helping new players learn the basics as well as improving their entertainment. The AI agent created was designed to play alongside a human team acting as a tutor, providing tips and support whilst aiming to improve their experience and gameplay. The tutor replaces one of the human players and will go unnoticed by others, which will consider it a regular player. The tutor supports the whole team, but it focuses on supporting a specific player, who it identifies as a partner and analyses their gameplay. This study was proposed since players of MOBA games are known for their ‘toxic’ behaviour which can include raging at novice players therefore novice players can often be greeted with unfriendliness, impatience, and even bullying. Whilst Riot Games do not tolerate bullying and provide players with a means of reporting such behaviour so that it can be identified and actions such as banning players can be taken, this system is not perfect and therefore remains an issue. New players are particularly at risk now that the game is eleven years old and many new accounts are not actually inexperienced players, but existing players that have made an account to start fresh, play with lower-level friends, or have had their previous account banned. This means that the new players of today are frequently being matched with experienced, impatient, and potentially toxic players on new accounts. To combat these situations, other online games have implemented ‘adopt-a-newbie’ program in which experienced players help new players for rewards. The idea of matching players of similar skill levels to create a friendly environment in which they can learn and improve together is somewhat misguided as there is nobody that can teach, and it is common to find players relying on video tutorials to guide them. The benefit of the method implemented in this study is that players are given specific guidance based on their gameplay and can also reduce frustration by ensuring that they are playing with a reliable, non-toxic partner. The agent was developed by extracting domain knowledge from expert players, in-game tutorials, videos and streams and integrating with the in-game messaging system in order to send warnings and tips. The success of the agent was measured by analysing the Kill/Death/Assist (KDA) of the players they helped. This metric is the ratio of the kills and assists a player has scored to the number of times they have died, the better the KDA, the better the performance. Figure 2 shows the average KDA of players before, during, and after playing alongside the tutor agent. All six players except from player C performed better whilst playing alongside the tutor which suggests that the tutor was successful. However, the KDA of the

majority of players declined after playing with the AI agent and while most of these were better than before, this suggests that the tutor was not a sufficient solution in providing guidance to help players consistently learn and improve. They also surveyed the players during this study which revealed that all of the players had previous experience with MOBA games and most of them found the behaviour of the agent to be very robotic. These findings, alongside the very low validity of studying just six players from a game with over 27 million daily players, suggest that this approach was not an ideal solution. Furthermore, although this method was not technically classed as cheating, it is evident that it would give players a very unfair advantage over the enemy team. Investigating this study highlights the purpose of carrying out this project. Since an AI approach was somewhat of a failure, perhaps a web-based approach, more easily available to all players of all abilities, would be more successful.

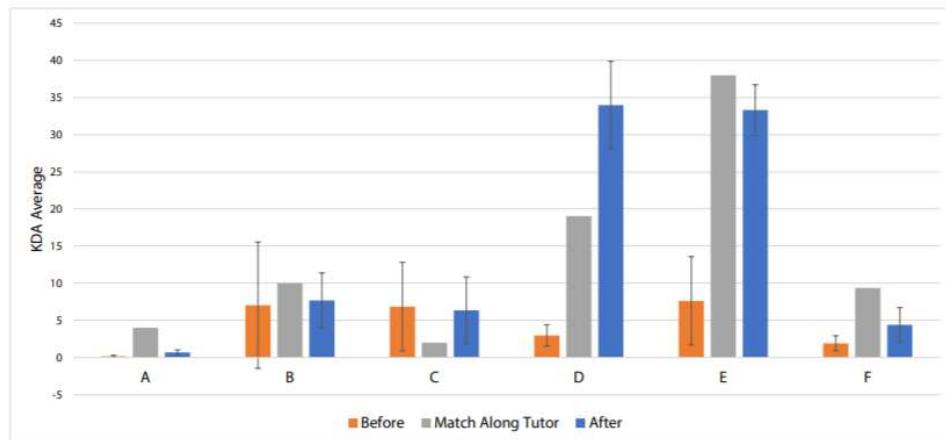


Figure 2: KDA of Players before, during, and after playing alongside the tutor agent (Nascimento Silva and Chaimowicz, 2017)

#### Using API Data to Understand Learning in League of Legends (Gerber, Sweeney and Pasquini, 2019)

This mixed-methods research was conducted to understand the impact of learning and player growth in a League of Legends summer camp in which eighteen adolescents engaged in three consecutive days of various team building activities. Data collected during this study included stats from the participants' game matches both before and after camp using an API, and qualitative data gathered by interviewing the campers. The hypothesis of this study was that players with better teamwork skills perform better, and therefore the aim of the summer camp was to build the participants teamwork skills in order to assess whether their game performance improved after attending. Since the very best players of the game are professionals of the game who often compete in established teams during eSports events, it would suggest that good teamwork is an essential factor in succeeding in the team-based game. However, since many players play alone, particularly to compete in Ranked Solo/Duo, teamwork strategy is likely to be a weakness for many casual players. An API was used to gather information about over 800 ranked games played across the participants, before and after attending the camp. The metrics closely analysed were the players' vision scores and assists as these are indications of team cooperation – a good vision score indicates that players have used items to grant map vision to their team so that they are less likely to be surprised by the enemy team and therefore be killed, a high number of assists indicate that players help their fellow teammates takedown opponents in a collaborative manner. Figure 3 shows the results in the form of a comparison between various performance metrics before and after attending the camp. Most metrics show minimal differences, however the all of the differences are improvements. Whilst kills is generally a statistic where higher means better, the slight decline in kills and the increase in assists suggest that players played less selfishly after the camp, sharing kill rewards with the team in order for the strength to be

fairly distributed, rather than ensuring they secured the glory for themselves. It is thought that many team members having a good number of kills is better than just one player having many since the team is more likely to succeed since rather than a single player being relied upon, every team member has the strength to contribute individually. Furthermore, the slight decrease in death counts and slight increase in win rate also suggest that players were more successful after taking part in the team-building activities. Whilst most of these results are fairly insignificant, the drastic increase in vision score was a very significant result. The average vision score of the players approximately doubled after the camp which suggests that they learned the importance of supporting their fellow team members as vision is a very beneficial aspect to all members at all times. The qualitative data obtained during the study included revelations that the players felt that patience and communication was central to the team's success and if communication broke down, it impacted their ability to win or establish a trusting environment. Whilst this study was a success, it is rather inapplicable to the majority of players. The research benefitted just 18 players out of the 27 million the game receives every day. Not everyone can get to know people they play with or take part in physical teambuilding activities. For this reason, a resource that is widely available to all players is more likely to help the player-base as a whole. The finding that good teamwork does indeed lead to better game performance is helpful for this project as the concept of players writing guides in order to help others could indicate that players would indeed be willing to contribute due to the understanding that supporting others leads to the wider success, and that the very activity of helping others could serve as a way to build players' teamwork skills.

Variable	Pre-Camp Mean (Post-Camp Mean)	Pre-Camp Standard Error (Post-Camp Standard Error)	T Statistic
Wins	0.51 (0.53)	0.017 0.017	0.68
Kills	7.41 (6.92)	0.197 0.184	-1.80
Deaths	6.37 (6.04)	0.133 0.130	-1.76
Assists	6.69 (7.12)	0.176 0.180	1.69
Vision Score	8.41 (16.5)	0.017 0.017	15.22**

\*Significant at  $\alpha = 0.05$ , \*\*Significant at  $\alpha = 0.01$ .

N = 842 in both groups.

Figure 3: A Comparision of Player's Performance Metrics before and after attending a team-building summer camp (Gerber, Sweeney and Pasquini, 2019)

## Methodology

### Software Development

Software Development is the process of taking a set of requirements (a problem statement), analysing them, designing a solution to the problem, and then implementing that solution on a computer (Dooley, 2017). The software development life cycle (SLDC) (Figure 4) is a process that produces software with the highest quality in the shortest time which includes a detailed plan for how to develop, alter, and maintain a software system (Stackify, 2017). There are many different models of the SLDC and the Waterfall method (Figure 5) is one of them. The core methodology chosen for this project is the traditional Waterfall method however, elements and ideas from iterative and agile processes were also considered throughout.

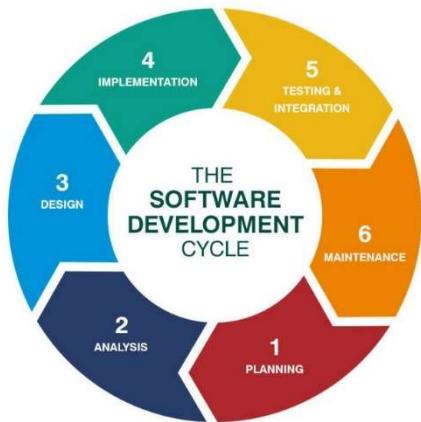


Figure 4: The SLDC (Synothe, 2017)

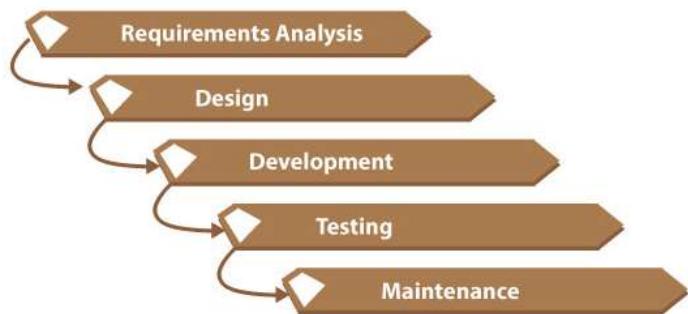


Figure 5: The Waterfall Model (WADIC, n.d.)

The waterfall method is a step by step process of development that follows a strict path with defined stages along the way. The first stage of the Waterfall method is the Requirements phase. This is where the requirements of the project are gathered and analysed. As there is no defined way that requirements should be gathered using the waterfall method, various techniques from different methodologies could be used at this stage. This is an advantage because it allows the freedom of choosing the most suitable methods of requirements elicitation for the project at hand. Since the requirements stage is fully completed before the development takes place, it ensures that potential problems are considered early and also avoids scope creep as requirements will not be continuously added along the way. The requirements stage of the waterfall method ensures that all requirements are clearly defined and documented, and that they define exactly what the system should do. The next stage is the design phase. This is where the requirements gathered from the first stage are carefully reflected upon in order to design a system that will meet the defined requirements. An advantage of completing all designs thoroughly before beginning development is that it helps to decide how, and ensures that, different components of a system will work together and integrate since they are considered and designed all at once. Furthermore, this is another way of avoiding scope creep as it does not leave an opportunity to add new features into design that would slow down the development and make the timings unpredictable. Then, the middle stage is where the actual development and coding take place. In some projects, the design phase might help in deciding which programming languages to use but since this project is about web development, the programming languages available are limited and the most appropriate are already obvious. By the end of the development stage, a working piece of software is complete. This allows for more accurate timing predictions. Once the system is developed, it must then be tested. The testing phase can involve both unit testing in which developers run a series of tests to spot bugs and errors, and to ensure that all requirements are being met. The testing stage can also involve users in which users can provide feedback on the usability and accessibility of the system so that it can be made as user-friendly as possible before release. The final stage at the bottom of the waterfall is the maintenance/operations stage. This is where any issues identified during the testing phase are addressed in order to get the system ready for release.

While the waterfall model has seen a slow phasing out in recent years in favour of more agile methods, it can still provide a number of benefits, particularly for projects that require the stringent stages and have strict deadlines (Powell-Morse, 2016). One benefit of the waterfall method is that it forces structured organisation, and this helps to meet targets and stay on track. Thanks to the well-defined requirements and design stages pre-development, the processes of the project have been made easier

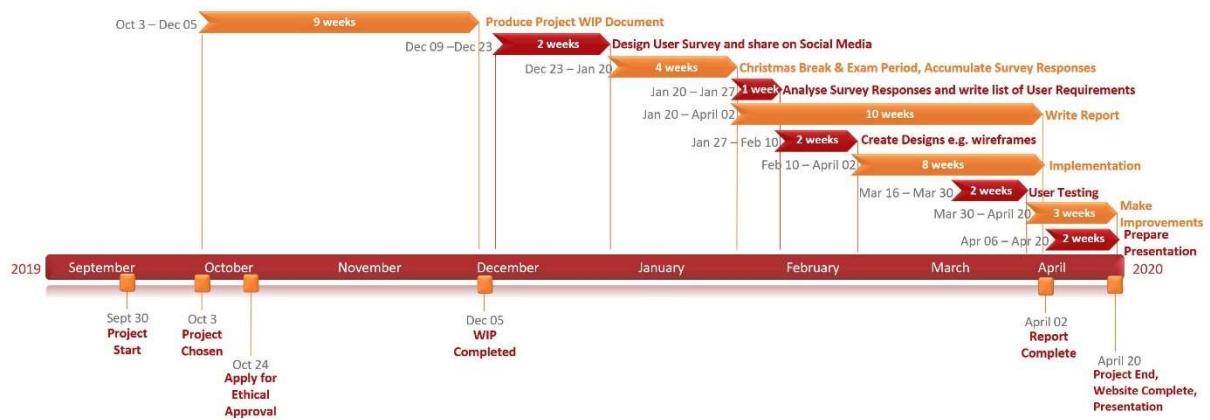
to document than if large amounts of improvisation and adaptation took place. Another advantage is that potential issues can be spotted early and changes to design can be made easily since sketches and mock-ups are much easier to edit than code. Moreover, the waterfall method is suited to milestone-focused development. Due to the strict deadlines and documentation requirements of this project, the waterfall method was very suitable in aiding to meet milestones and deadlines. As well as this, most agile processes require constant involvement of users and this was not feasible for this project. Not only was it difficult to dedicate the time and to find willing participants for user involvement, but this was worsened due to the coronavirus outbreak. Effective methods of user involvement include focus groups and observations which were made impossible due to the extraordinary circumstances. Therefore, all user involvement was required to be carried out online and kept to a minimum. For these reasons, it is clear that the waterfall method was the most appropriate choice of methodology for this project.

## Project Management

Project management techniques refer to the tools and processes used to aid in carrying out a project whilst maintaining scope and timeframes.

### Gantt Chart

As discussed in the Work-in-Progress (WIP) document, a Gantt chart was created to plan the schedule of this project. Gantt charts are a visual representation of tasks against time in the style of a bar chart. They represent critical information such as the duration of tasks and overlapping activities. They are useful because they are relatively simple to create and understand, and they serve as a timeline that illustrates how the project will progress (Kashyap, 2019). The Gantt Chart is a very important and suitable tool for the waterfall methodology since a complete chart often takes the same shape as the steps in the waterfall and is therefore in sync with the methodology. An advantage of making a Gantt chart before beginning development is that all necessary tasks and activities are considered, decided, and planned. This meant that it was less likely to forget a step or to carry out tasks in an illogical order. Furthermore, it allowed for estimates to be made regarding the timeframe of the project overall and the individual tasks necessary to complete.

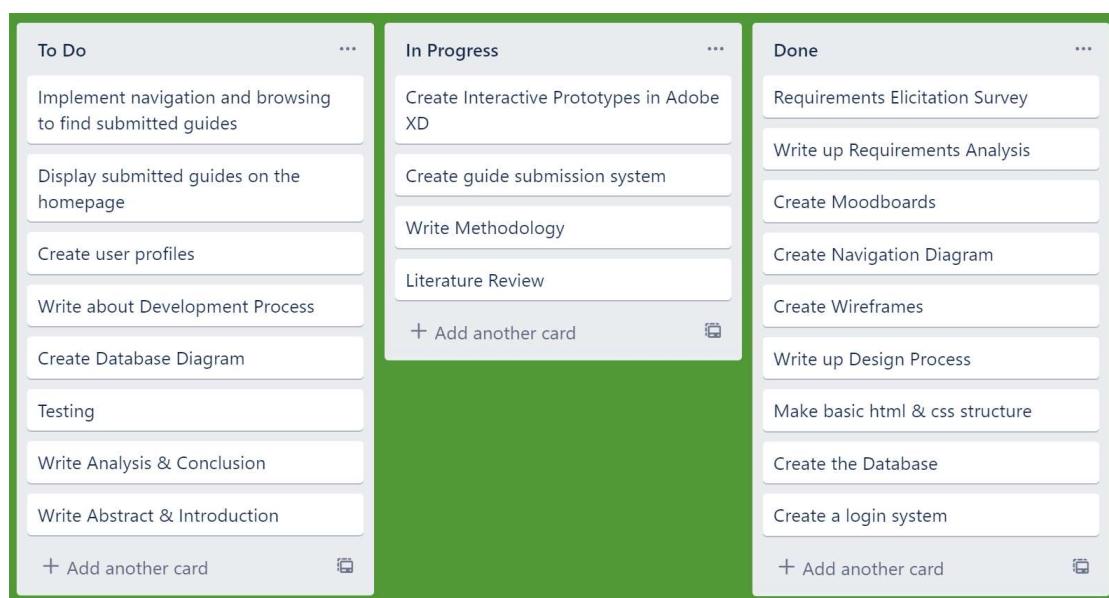


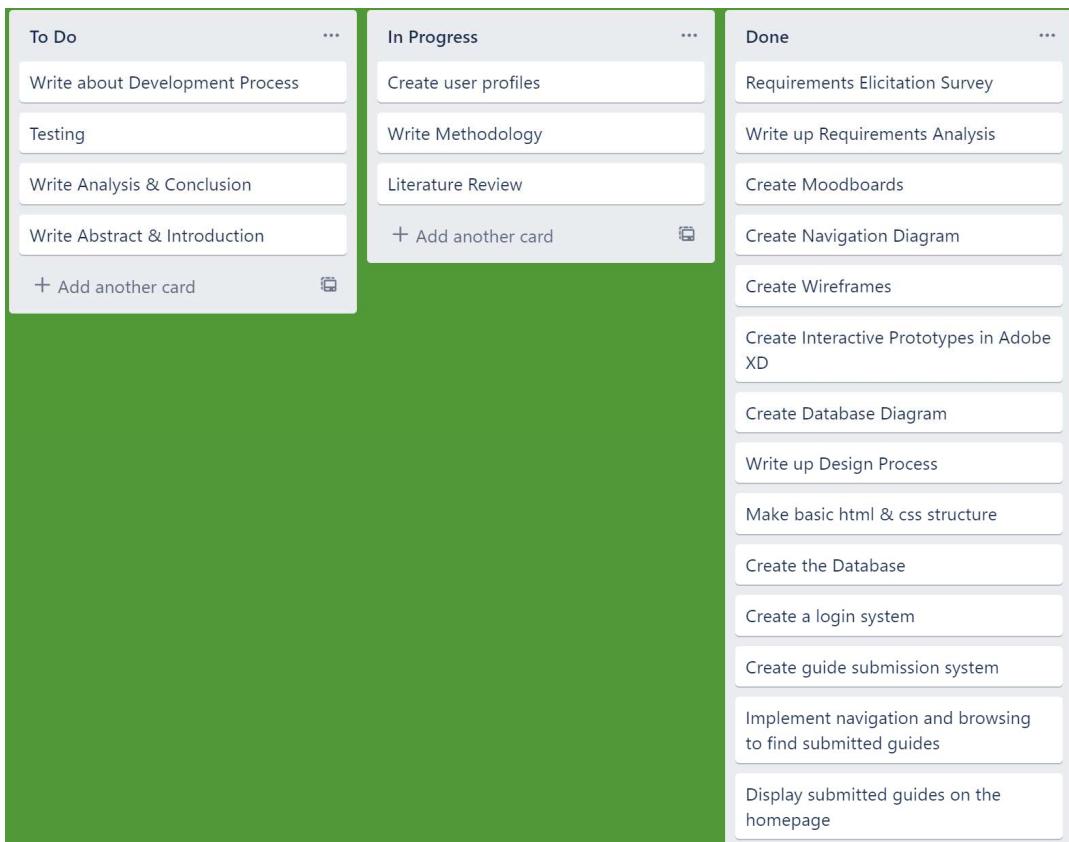
However, the Gantt chart was somewhat of a failure during this project. While it was undoubtedly useful in making initial estimates, deciding tasks, and identifying milestones, it was not used to its fullest potential. Firstly, some of the timeframes set proved to be too ambitious which led to the project processes lagging behind and large amounts of stress caused by the build-up of tasks and unknown timeframes. As well as this, the timeframes and deadlines changed in order to adapt to the extraordinary circumstances caused by the coronavirus. However, the Gantt chart was not updated or remade to adapt to these new timescales which rendered it largely useless towards the end of the

project. This was due to the changes taking place during the hectic development stages of the project that could not be interrupted and therefore updates to the Gantt chart were not prioritised and ultimately forgotten about.

### Trello

Trello is a free project management tool that organises your project into boards. These boards can then be used to organise tasks into sections which can be defined by the user. A common approach to using Trello boards which was used in this project is to organise tasks into 3 columns; 'to do', 'doing' and 'done'. Whilst Trello is often praised as a collaborative working tool that helps teams work together and stay on track, it was also very useful in this solo project. The images below show the Trello board created for this project at two different points of progress. Due to the creation of the Gantt chart and the rigorous requirements elicitation phase involved in the Waterfall method, it was fairly easy to add specific tasks to organise into the Trello board as the necessary steps had already been considered. Furthermore, Trello's easy to use interface made organising and updating the board very easy as tasks can simply be dragged and dropped to the relevant lists. This helped to keep track of the progress and identify what needs to be done next. It also helps to spot work patterns and reflect on the process and success of the project since the activity feed keeps a log of when updates are made. From the examples below, it is clear that the report was temporarily put on hold whilst development was carried out since the 'methodology' and 'literature review' writing tasks remained in progress whilst many other tasks relating to development were completed. In reflection, more effective use of Trello could have been made by updating the board more frequently instead of moving several tasks in bulk. Nevertheless, it proved to be a valuable tool throughout the project in aiding staying on track and ensuring all tasks were completed.





## Microsoft Teams

This was a rather unexpected tool to be utilised in this solo project. However, due to the coronavirus pandemic, it was important to stay in touch with academics to maintain support in this project. A Computer Science student is not necessarily equipped with the necessary expertise to carry out such a project before it begins and therefore it was important to continue the learning process throughout. Guidance and explanations from expert academics provided much more valuable insight than unexplained code snippets found online, therefore online communication tools used to collaborate with academics became an important tool during this project. Microsoft Teams offered improved quality and advanced capabilities over other means provided so was therefore chosen as a communication tool. Furthermore, it supports instant messaging, filesharing, screenshare, and voice chat which meant that everything was in a single, convenient place, saving time compared with using various tools for these features.

## Toolsets and Machine Environment

A toolset relates to any kind of software tool and these can be helpful in both software development and project management, but since a variety of project management tools have already been discussed in the previous section, this section will focus on those tools which aided development. Many different tools came in handy throughout this project, as is to be expected of a project this size, and all of the tools used have been listed in the table below. Some tools are very straightforward, self-explanatory or have no major competitors therefore only a selection of the toolsets chosen will be discussed. Most of the tools listed are free to use and therefore this will not be discussed as a relative comparison or advantage.

Design	Documents	Programming	Hosting	Storage	Browser	Video
Pic Collage, Moqups, Adobe XD, DB Designer	Microsoft Word, Microsoft Publisher, Google Docs, Google Forms, Visual Studio Code	HTML, CSS, JavaScript, jQuery, JSON, AJAX, PHP, SQL (MySQL)	XAMPP, phpMyAdmin	Google Drive, OneDrive	Google Chrome	Screencast-O-matic, YouTube

### Programming Languages

HTML is the language used to create the layout and structure of the website and CSS is the language used to add styling such as colours, fonts, and advanced positioning. HTML and CSS are standards in the web industry and therefore no decision needed to be made regarding which programming languages to use for the front end of the website. Nevertheless, it was important to be aware of the current standards and make use of the newest features in order to futureproof the site. For example, many new tags were introduced in the latest version of html, HTML5, that were important to learn and utilise. Using these tags enrich the semantic value of the site and make the code cleaner and easier to understand. For example, using the tag <header> makes the purpose of the section far clearer than a generic <div>. Furthermore, HTML5 forms were made much more elegant and offer new capabilities with new types of inputs and fields which reduced the need to use the much more complicated JavaScript for a relatively basic feature. Many JavaScript scripts running on a page can slow down response times so since forms were an important feature of the site, it was beneficial to be able to avoid large amounts of JavaScript since HTML alone can now handle advanced forms.

JavaScript has also become a standard of modern websites with any advanced features or dynamic information. It can be used to update and change both HTML and CSS based on different conditions and can also calculate, manipulate, and validate data (W3schools.com, n.d.). JavaScript can create responsive and interactive elements for web pages which is essential in enhancing user experience. However, JavaScript can be considered relatively complex, especially when compared with the simplicity of HTML and CSS. Therefore, for this project, jQuery was used. jQuery is a lightweight, “write less, do more”, JavaScript library which offers an easier way of implementing JavaScript. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish and wraps them into methods that you can call with a single line of code. jQuery also simplifies complex features of JavaScript, such as AJAX calls and DOM manipulation (W3schools.com, n.d.). There are many JavaScript Libraries, but jQuery seems to be the most popular by far. Figure 6 shows that 73.4% of websites use jQuery and that it has a 97.5% market share. Figure 7 shows the results of a survey in 2013 in which over 1500 people were asked about their JavaScript Library of choice. Over half, 52%, voted for jQuery and it seems that its popularity has continued to increase over the years. As well as its simplicity and popularity, jQuery offers many other benefits such as interoperability, speed, and efficiency. Therefore, jQuery was a clear choice for this project and allowed for the implementation of many advanced, attractive features designed to enhance user experience.

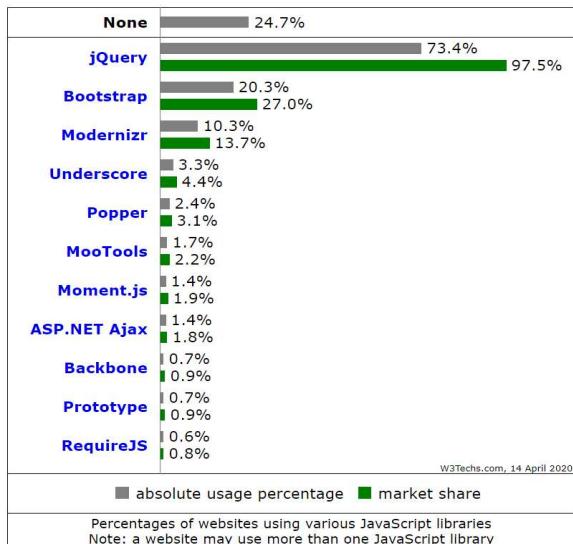


Figure 6: Usage and Market Share of JavaScript Libraries (W3techs.com, 2020)

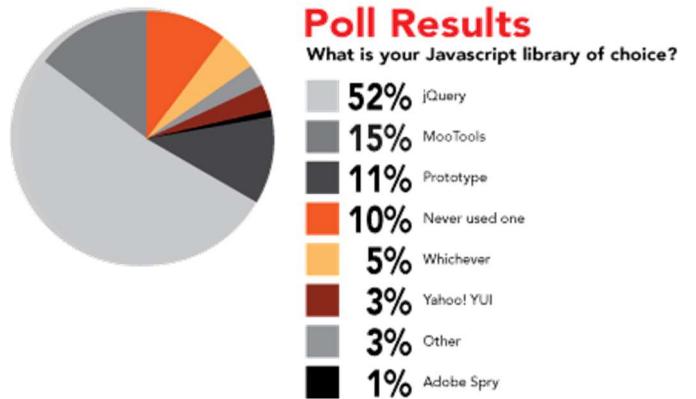


Figure 7: Poll results of asking people about their favourite JavaScript Library (Coyier, 2013)

Many different programming languages can be used in server-side/backend development. Figure 8 shows the programming languages used in various popular websites and from this it is clear that there is no clear choice for a server-side language. Whilst PHP is only used in 4 of the 7 examples and is used solely in none other than Yahoo, PHP was chosen as the single server-side language for this project. PHP is a widely used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML (PHP.net, n.d.). PHP's similarity and compatibility with HTML is extremely advantageous. With knowledge and experience in HTML, it is relatively easy to make the step to PHP and there is no steep learning curve. This saved crucial development time since it would have likely taken much longer to learn an unfamiliar language. Since HTML can be used alongside and inside PHP, it kept the complexity to a minimum since it reduced the need for many different files of many different types – everything was able to be kept in once place, neat, and compact. Furthermore, PHP is widely supported and is compatible with all operating systems and browsers. This maximises the likelihood that the website will be compatible with users' devices and software which is important since League of Legends is a widely available game. Furthermore, PHP easily connects with databases which were an essential part of this project. This capability also meant that everything could be done in one convenient location. Having many different toolsets and files can soon make things very complicated, cause compatibility issues, or make bugs and errors more difficult to identify and solve. Another major advantage is that PHP is known as the fastest programming language and therefore the site can be easily and quickly loaded which improves user experience since they are less likely to be made impatient. Even without exploring the specifics of comparisons between other languages, PHP was quickly identified as the best choice for this project and therefore valuable time was saved avoiding extensive research into a new language.

Website	Client-side programming language	Server-side programming language(s)
Google	JavaScript	C, C++, Go, Java, Python, PHP (HHVM)
Facebook	JavaScript	Hack, PHP (HHVM), Python, C++, Java, Erlang, D, XHP, Haskell
YouTube	JavaScript	C, C++, Python, Java, Go
Yahoo	JavaScript	PHP
Amazon	JavaScript	Java, C++, Perl
Wikipedia	JavaScript	PHP, Hack
Twitter	JavaScript	C++, Java, Scala, Ruby

Figure 8: Programming Languages used in Popular Websites (IONOS Digitalguide, 2019)

SQL is a language widely used to communicate with databases. While this is the standard for relational databases, the choice between using a relational database with SQL or a non-relational database with NoSQL needed to be made. Since non-relational databases are most useful for unstructured, varied data, the choice of using a relational database was easy to make since all of the required information could easily be organised. Whilst this choice was straightforward, there are many different types of databases to choose from. Figure 9 shows the most used databases of 2018 and the most popular, MySQL was the one elected in this project. MySQL is globally renowned for being the most secure and reliable database management system used in popular web applications like WordPress, Facebook and Twitter (Datamation, 2016). Whilst the website made for this project would not be obtaining sensitive data such as addresses and card details, security and confidentiality should still be a priority for any user-oriented project. Furthermore, the reliability and availability means it can be counted on to have high uptimes and be capable of managing high demand at high performances. Since the database required for this project was very simple, there was no need to search for a type offering specific advanced capabilities or features aimed at niche markets. Therefore, MySQL was a suitable choice for this project.

## Hosting

A hosting service is an essential part of web development since it is required to view the product online, through a browser. As this was a solo project, local hosting was used throughout development because access is only necessary on the developer's local machine. This avoids the potential costs and unreliability of online hosting services since it does not need to be viewed by many people. When it comes to local hosting, there are two main competitors; XAMPP and WAMP. Whilst either would have been an appropriate choice for this project as both include PHP and MySQL, XAMPP was chosen as it offers some benefits over WAMP. Although XAMPP includes more than WAMP such as several extensions and Perl, as these were not necessary for this project, this is not a relevant advantage. However, XAMPP is known for its clean, simple interface which is ideal for beginners (Digamber, 2020) and therefore is easier to set up and configure than WAMP. With no previous experience in web hosting, the user-friendliness of XAMPP was very advantageous as it did not take long to install and configure. Furthermore, XAMPP is a cross-platform service whereas WAMP is only compatible with Windows (MAMP is the alternative for Mac OS). Since some lab machines have Mac OS and it is the

## > Top Databases

Database use is dominated by SQL-based choices, with MySQL still the most commonly utilized database solution.

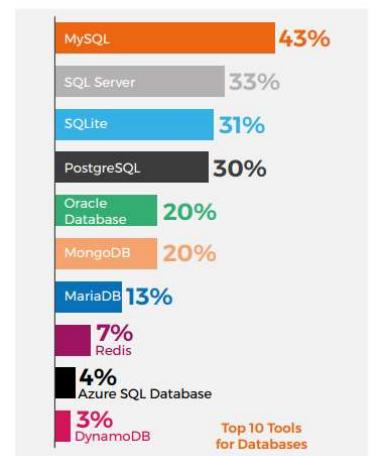


Figure 9: Top 10 Database Tools (Medium, 2018)

OS of choice for academics in the Web field, using XAMPP made it much easier to share files with academics in order to get programming support.

## Research Methods

Whilst this section may seem relatively short, various research methods have been used extensively throughout this project and have thus been discussed in their relevant sections. Research methods are the strategies, processes or techniques utilised in the collection of data or evidence for analysis in order to uncover new information or create better understanding of a topic (Libguides.newcastle.edu.au, 2019). As this project user-centred design, some degree of user involvement was essential. User-centred design is a design philosophy and a process in which the needs, wants, and limitations of the end user of an interface or document are given extensive attention at each stage of the design process. This helps to put customers at the heart of the online experience (Rouke, 2017). For this reason, users were involved in the requirements elicitation phase in order to gain insight into what potential users would really want out of the system. This research was conducted using surveys and the justification for doing so as well as the findings will be covered in the following section. Furthermore, users will also be involved during the testing and evaluative stages towards the end of the development process. Typically, during usability testing, participants will try to complete typical tasks while observers watch, listen and takes notes (usability.gov, n.d.). However, the testing stage of this project was significantly impacted and hindered by the coronavirus pandemic as interaction with participants was not possible. Ideally, these tests would have been carried out via a focus group. A focus group is a gathering of deliberately selected people who participate in a facilitated discussion intended to elicit consumer perceptions about a particular topic or area of interest. Focus group methods permit alternative ways of obtaining information from consumers without using a survey as survey instruments tend to be viewed as scientific, particularly when they produce quantitative data (Devault, 2019). However, as focus groups were made impossible due to COVID-19, surveys unfortunately had to be used during user testing too. However, it was made a priority to gather both quantitative and qualitative data during research. Further information about the research methods used during user testing can be found in the relevant section: Testing (Usability Testing). Quantitative research relates to measuring the quantity of something. This type of research is useful for performing statistical analysis and for finding an exact answer. Whilst both cases of user involvement were carried out using a mixed-methods study as both quantitative and qualitative data was obtained, quantitative research proved most useful in identifying the target audience and quantifying players' opinions and desires. This made it clear which features to prioritise in development. Qualitative data refers to measuring the quality of something and is therefore obtained using open-ended questions to provide participants with an opportunity to share their thoughts, feelings, and opinions freely. It was important to gather qualitative information in the form of feedback post-development to understand users' opinions on the finished topic in order to assess its success in meeting its requirements. Both forms of research discussed so far have been examples of primary research which is where the project leader collects data themselves. However, a significant amount of secondary research (understanding and analysing the findings of other researchers) has also been carried out which has already been discussed in the Literature Review. Conducting secondary research helped to gain a deeper understanding in the field and therefore the problem domain as well as identifying the successes and failures of other research projects proposing solutions to a similar product. This help to verify the purpose of carrying out this project and identify features that should be prioritised. The research methods used during this project have been discussed throughout and proved to be the most suitable choices given the current situation and the nature of this project. Whilst not all research methods were ideal, they have been used appropriately throughout.

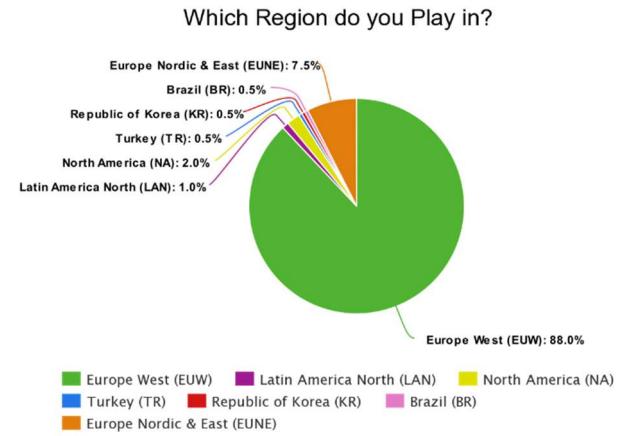
## Design, Development & Evaluation

### Requirements Elicitation, Collection & Analysis

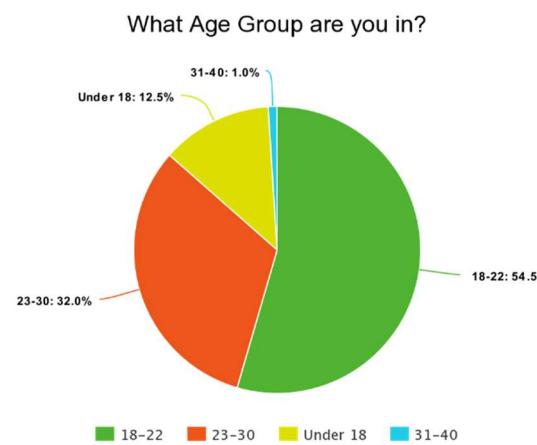
Requirements Engineering is arguably the most important phase of the Software Development Life Cycle (SDLC) since it defines exactly what is to be developed and can avoid potential issues later in development if done thoroughly. A study by the Standish Group noted that the three most commonly cited root causes of project failures, responsible for more than a third of projects running into problems are; a lack of user input (13% of projects), incomplete requirements and specifications (12% of projects) and changing requirements and specifications (12% of projects) (Mohapatra, 2010). All three of these issues can be addressed and avoided during Requirements Engineering. Requirements Elicitation is the first stage of the Requirements Engineering phase in which potential users or stakeholders can inform developers regarding what they would like the system to do for them and how they might use it. There are many methods of gathering this information and the results of doing so can define the scope of the project and the features the system must include. Surveys have been a primary source of data collection within Human-Computer Interaction (HCI) since the early days of the science and they are still popular today since they allow researchers to collect information based on users' satisfaction, opinions, ideas, and evaluations regarding a system in a relatively quick and easy manner (Ozok, 2009). An online survey was conducted for this project in order to gather user requirements and information regarding the target audience. There are many benefits to using online surveys such as being able to reach a large number of people and being unrestricted by geographical location. This meant that a fairly large sample size was able to be used resulting in a higher validity of results. Furthermore, since League of Legends is a popular game worldwide, responses could be gathered from players in various regions. It is evident that a survey was the most appropriate elicitation method for this project since players in various locations were able to submit responses at their leisure whereas methods such as focus groups and interviews would have resulted in a much smaller sample size and a much greater population bias.

The survey was created using a Google Form from which a link could be generated and shared for people to submit responses. Firstly, a selection of fellow players were personally asked to complete the survey. Whilst these responses alone would have resulted in bias due to likely being similar levels etc., it was good to ensure that a handful of responses were more likely to be answered seriously and honestly since a friend is more likely to care about the success of the project than a stranger. Then, the friend/contact system in-game was utilised to gather more responses. Since the rank and level of players is displayed in this list, it was useful for ensuring that responses were being gathered from a variety of abilities. Furthermore, although all of these contacts were restricted to the Europe West (EUW) region, it meant that responses could be gathered from various countries rather than people known locally. To gather as many responses as possible, the survey was then shared on a Facebook page dedicated to the game (Appendix 1a Figure 18). Unfortunately, the only group that allowed this was specific to the EUW region which meant that gathering responses from all over the globe was unlikely. Nevertheless, this allowed far more people to be reached than other methods such as paper-based surveys or interviews. This was a success since the post received 24 likes and 42 encouraging comments. Furthermore, a total of 200 responses were gathered resulting in a high validity of results and a diverse range of answers were collected.

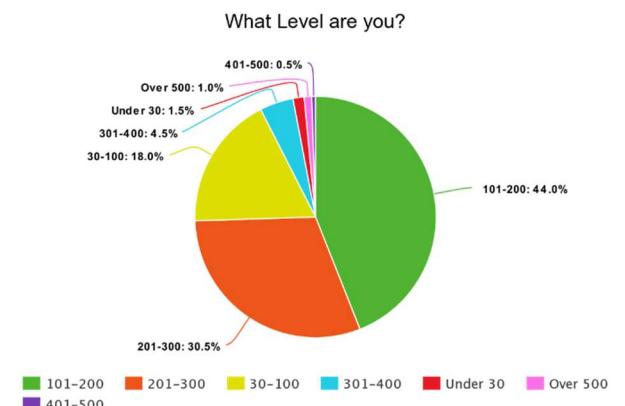
The survey was not only useful for determining what the users want, but also useful in evaluating who the users are. The survey conducted asked the players several questions about themselves in order to gain information regarding the product's potential target audience without asking for any unnecessary personal information that could raise concerns of data protection, privacy, and confidentiality. Personal data such as names and addresses were not collected, and the survey was answered anonymously. The first question in the survey asked players which region they play in (*Figure 12*). This was important to know because while the survey could be answered remotely anywhere in the world, the ability to share the link to the survey was somewhat restricted to the local region, both in-game and on Facebook groups for specific regions. As suspected, the majority of responses (88%) came from Europe West (EUW) so it was important to note that this is not representative of the League of Legends overall player-base and that trends identified in this study might not apply to different regions. The survey then asked players what age group they are in (*Figure 11*) as this is a key data point in analysing target audience and age can be a factor of accessibility considerations. The results of this showed that none of the respondents were over the age of 40 and the majority (54.5%) of participants were aged between 18 and 22. This means that accessibility considerations for elderly people are unnecessary, and the design should appeal to young people. As well as this, it was useful to gather information regarding users' playing habits. The primary research revealed that 60% of participants play daily (*Appendix 1b Figure 19*) and more players said they play 'just for fun' (34.5%) than for any other reason (*Appendix 1b Figure 20*). These results show that League of Legends is still a very popular game and its player-base is dedicated to it. However, surprisingly, the majority of players prefer playing for fun rather than competitively which highlights the need for a guide site to offer advice applicable to any player rather than just those seeking to improve their rank. Moreover, surveys such as these can help to identify and justify the purpose of a project. In this case, since the aim is to enable experienced players to provide advice to less experienced players, it was useful to identify this ability gap. The survey results showed that more participants were between levels 101 and 200 than any other level range (*Figure 10*). This showed that there are plenty of experienced players capable of submitting advice. However, only 1.5% of respondents were under level 30 (*Figure 10*) which might suggest that there is a small pool of inexperienced players, but it is likely due to the fact that newer or inexperienced players would not be passionate or enthusiastic enough to join a Facebook Group dedicated to the game. The survey also asked participants what rank they achieved at the end of the latest Season (*Figure 13*) as this is also an important indication of ability and players



*Figure 12*



*Figure 11*



*Figure 10*

often seek guidance and improvement in order to climb the ranking ladder. The results showed that more players finished in Gold IV than any other rank which is approximately the middle rank and that, surprisingly, more players achieved ranks above Gold IV than below. However, it was clear that these results may not be representative of the actual distribution, especially since those more advanced at the game are more likely to be enthusiasts and therefore join Facebook groups. For this reason, secondary research was also carried out to investigate the actual rank distribution at the end of Season 9 (November 2019). The data found (Figure 14) shows that while Gold IV really is the rank that more players achieved, a higher proportion falls below the average ranks than above. Furthermore, the player-base is fairly evenly distributed therefore there is a large proportion of players experienced enough to provide accurate guidance as well as a large pool of less advanced players who could greatly benefit from the advice. Players were also asked what was the highest rank they've ever achieved (Appendix 1b Figure 21) and these results skewed to the right suggesting that more players have previously achieved a better rank than they attained at the end of last season. This could suggest that the game has become more difficult over time so that even though players may have been playing for a long time, new sources of guidance can always be helpful to enable players to maintain and improve their skills. To further back this up, only 39.9% of respondents said that they achieved the rank that they wanted or expected to (Appendix 1b Figure 22) suggesting that the majority of players would indeed like to improve their rank. In addition, almost half (48%) of participants rated the overall difficulty of the game as either 'difficult' or 'very difficult' (Appendix 1b Figure 23) and the difficulty of the ranking system as either 'difficult' or 'very difficult' (Appendix 1b Figure 24). This is substantial evidence that the game does not alone provide enough support and guidance to aid players in understanding the game knowledge or developing their skills.

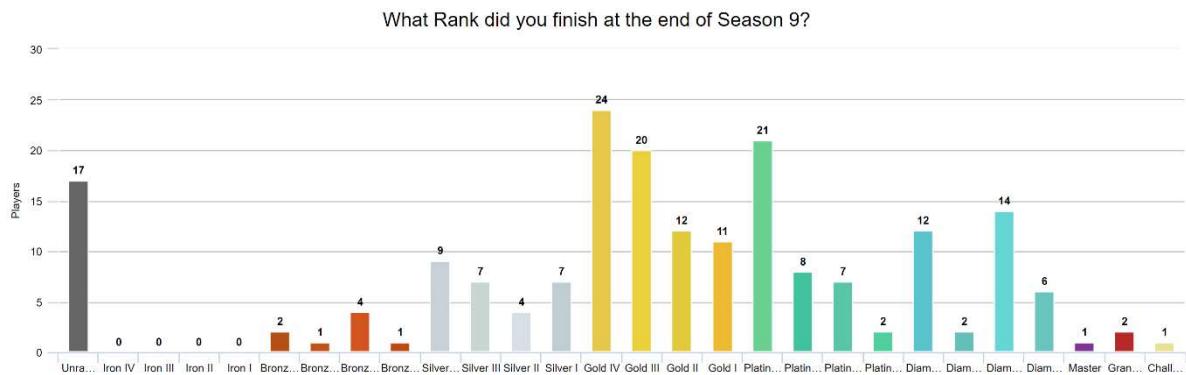


Figure 13

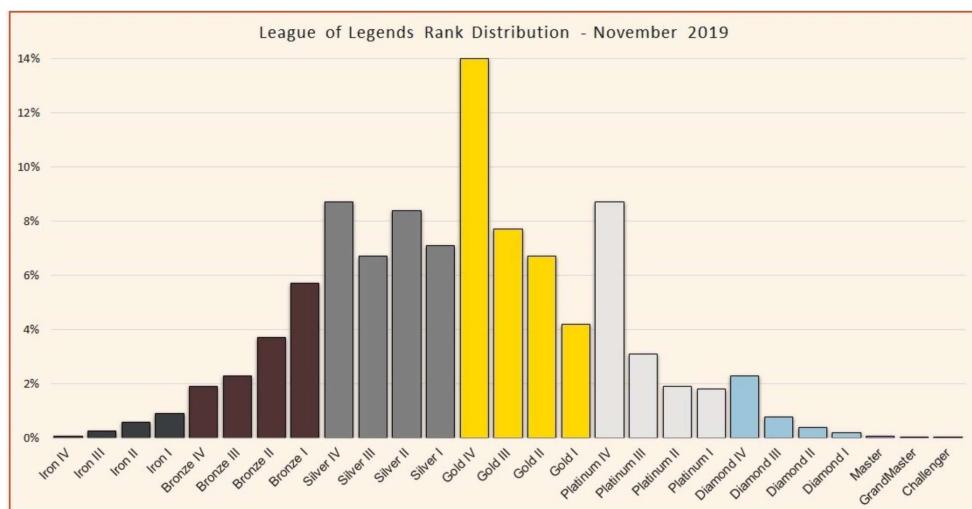


Figure 14 (Esports Tales, 2019)

The primary research conducted was also extremely valuable in identifying when and why players use existing guide sites and what they want to see from them. The survey asked how often players use guide sites (Appendix 1b Figure 25) and the most popular response (73 votes) was ‘every time I play a champion I haven’t played in a while’ and ‘every time I play a new champion’ was also a highly voted option with 59 ticks. Interestingly, the least popular response was ‘every time I rank up’ with just 2 votes which suggests that guide sites are not only useful for those wanting to keep up with other players in their rank but for players experimenting with different champions in regular gameplay. This further highlights the need for the system to be welcoming for inexperienced players and those playing casually. Players were also asked about what features they find important when using guide sites (Figure 15) and the majority of players (61%) answered ‘easy and quick access to the information I am looking for’. This result highlights the need for the site to be quick and responsive and graphic-based rather than text-based so that players can quickly navigate and see the advice quickly, perhaps in the middle of a fast-paced intense game. Other responses with a high number of votes included; ‘having a variety of build options for each champion’, ‘being able to see how to adapt your build to various situations’, ‘being able to see counter-picks’, and ‘having guides updated to each patch’. These responses show that building is an element of the game that players tend to struggle with most and the need for a new system that can show adaptive guides rather than static. Players were also asked about what skills they would like to work on (Figure 16) in order to identify what kind of advice would be most useful. Only 7 people said that there are no skills that they would like to work on which suggests that the vast majority of players would like to improve their skills. This is also backed up by the fact that only 11% of respondents said that they are not intending or hoping to improve their skill and/or rank in the next Season (Appendix 1b Figure 26). Unfortunately, many skills that were the most highly voted by many players as things they would like to work on are not something that can be easily helped through reading or looking at advice but instead come with practice. Examples of these include map awareness, objective control, learning when to freeze lane and when to push, and improving creep-score. Nevertheless, many of the skills that a guide site can help people with were also highly voted. Some of these include; learning more roles, learning more champions, mastering a role, mastering the kits of a small champion pool and adapting builds appropriately in various situations. These are all skills that the system should cover, and they will all be beneficial in helping players to improve their abilities.

Overall, conducting this primary research has shown that there is definitely room for a new guide site on the web in order to help players dynamically and adaptively since existing sites often fail to do so. 84.5% of participants answered yes or maybe when asked if they would be interested in a new website that would combine some of the most important features in various existing sites/apps into one handy place (Appendix 1b Figure 27). Furthermore, this research also provided useful information regarding the target audience which will undoubtedly help in design considerations, as well as insight into the guidance that the system must be able to support and provide.

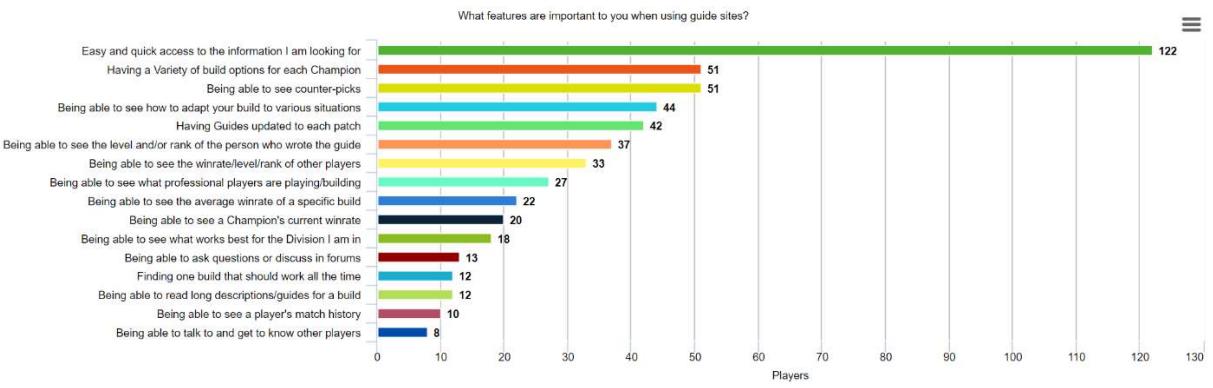


Figure 15

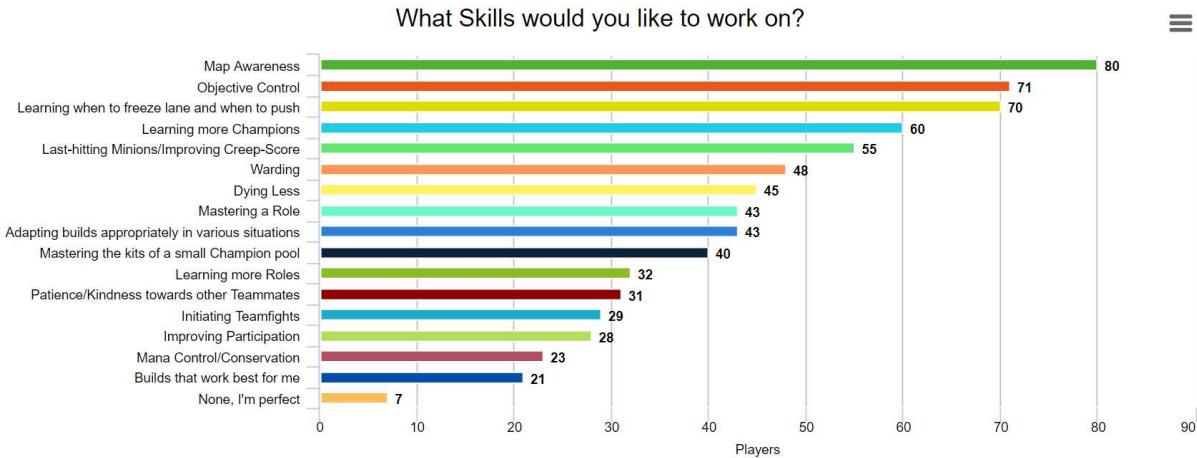


Figure 16

Once the responses had been gathered, it was then time to analyse the results and use them in requirements elicitation. Firstly, a complete list of findings was created in order to summarise the results (Appendix 1c). This was done since the automatically generated charts by Google Docs can be somewhat difficult to read and require a lot of scrolling or clicking to view the summaries of each question. The list identified the key findings from which information regarding target audience and user requirements could be extracted. This list will undoubtedly be a helpful resource throughout the project to serve as a reminder of scope and usability and accessibility considerations. However, this list alone was insufficient in thoroughly determining user requirements. To enhance understanding, the agile approach of 'user stories' was used. A user story describes functionality that will be valuable to a user of a system software. User stories are composed of three aspects; a written description of the story used for planning and as a reminder, conversations about the story that serve to flesh out the details of the story, and tests that convey and document details that can be used to determine when a story is complete (Cohn, 2004). In other words, user stories are short descriptions about an aspect of functionality written from a user's perspective typically written in the form: as a <type of user>, I want <some goal> so that <some reason>. User stories were selected as a method of requirements elicitation instead of use cases in this project because they are far more time-efficient and are easier for anyone to read. Although user stories are typically used in agile methodologies, they are also useful in an iterative approach to the waterfall method as testing and evaluation will also involve users before the product is completed, hence user-centred design. However, since users had already been involved in submitting survey responses, the traditional concept of user stories being written by users on index cards was not applied in this scenario. Instead, as a means of gaining a deeper understanding of survey responses, 10% of the 200 responses gathered were randomly

selected to create user stories for. So, 20 user stories were fabricated by scrutinising 20 unique survey responses (Appendix 1d). This activity helped to gain an understanding of the different types of users as well as the features required for various reasons. Furthermore, this list of user stories can help to prioritise features and plan task order. Using a combination of these methods of analysis meant that a complete list of functional requirements could therefore be thoroughly derived.

### System Requirements

- The system must represent the League of Legends aesthetic by using the same colour palette and other design features
- The system must avoid difficult navigation or long load times by having a minimal number of pages
- The system must be usable on various devices with different screen resolutions
- Users must be able to create an account and log in to access the full capabilities of the system
- Logged-in users must be able to create and submit guides that will be automatically displayed to all users
- The guides must follow a set layout and require mandatory information such as the name and role of the Champion, at least one build recommended for a defined situation, the patch version of the game it is written for, and the rank, level, and region that the guide is tailored to.
- Information must be easy to access and to visualise so the system must use visual cues and avoid large amounts of text where possible
- The system must allow all users to easily find guides they are looking for and refine their searches by Champion, Role, Level, Rank, Region, Patch, and Rating
- The system must display useful categories on the home screen such as the most recent guides or most popular Champions
- Logged-in users must be able to access and customise their profile

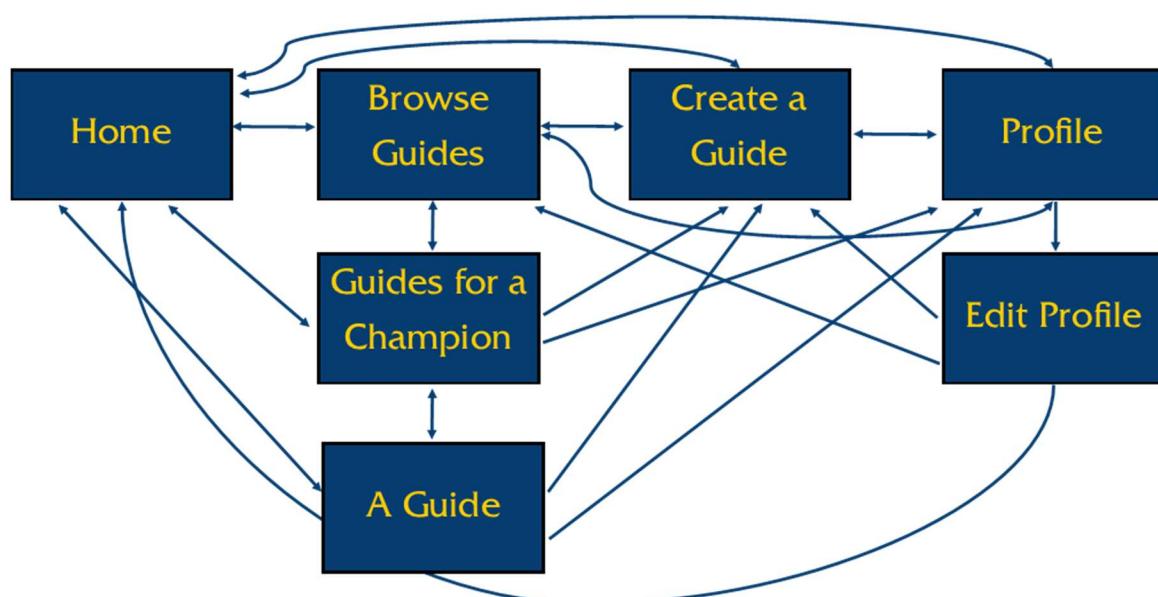
### Design

The design phase is the second stage of the Software Development Lifecycle in which the established requirements are used to decide the structural and aesthetic properties of the system. There are several ways to approach this phase and many different types of designs. In this project, design was tackled by considering the five planes of user experience (UX). This is the concept that there are five distinct layers of UX design, ranging from abstract to concrete. These five planes help to break down the design process into steps that reflect the different levels of abstraction involved in the design. It starts with a basic concept/idea and builds up to the visual design aspect. The first two planes were largely covered in the Requirements Engineering phase, but they are crucial to reflect upon in order to create carefully considered designs that match the defined specification. The first plane is the strategy plane which considers the reason for the system, why it is being created, who it is being built for, why people are willing to use it, and why they need it (Elgabry, 2016). As established, the system will be a website aimed at League of Legends players in which experienced players can offer advice to newer players to help them learn the game and develop a deeper understanding of game knowledge. Throughout the design phase, it is important to keep this in mind at all times to ensure that the correct product is being designed and therefore the correct product will be built. Furthermore, the information gathered via the survey helped to identify the target audience and the need for the product and certain features. For example, it was revealed that the website does not need to have extensive accessibility features suited for the elderly since the age group of the target audience is younger than 40, a majority agree that the game is not sufficient in teaching game knowledge and most would like to improve their skills so a need for a guide resource is clear, and a large proportion would like to know more about builds so the site must offer this as a feature. All of the information

gathered is essential to consider during the design stage to ensure that features are being included that people desire and considerations are being made to ensure optimum usability and accessibility for the target audience. The second plane is Scope. This is the stage where the functional requirements are defined. This is important because the requirements can serve as a checklist to ensure that every feature has been incorporated into the design. As well as this, it can help to avoid ‘scope creep’ which could result in new ideas being implemented taking time and attention away from the core, original requirements. The system requirements have already been defined in the previous section. To reiterate from a user’s perspective, a user of the website will be able to...

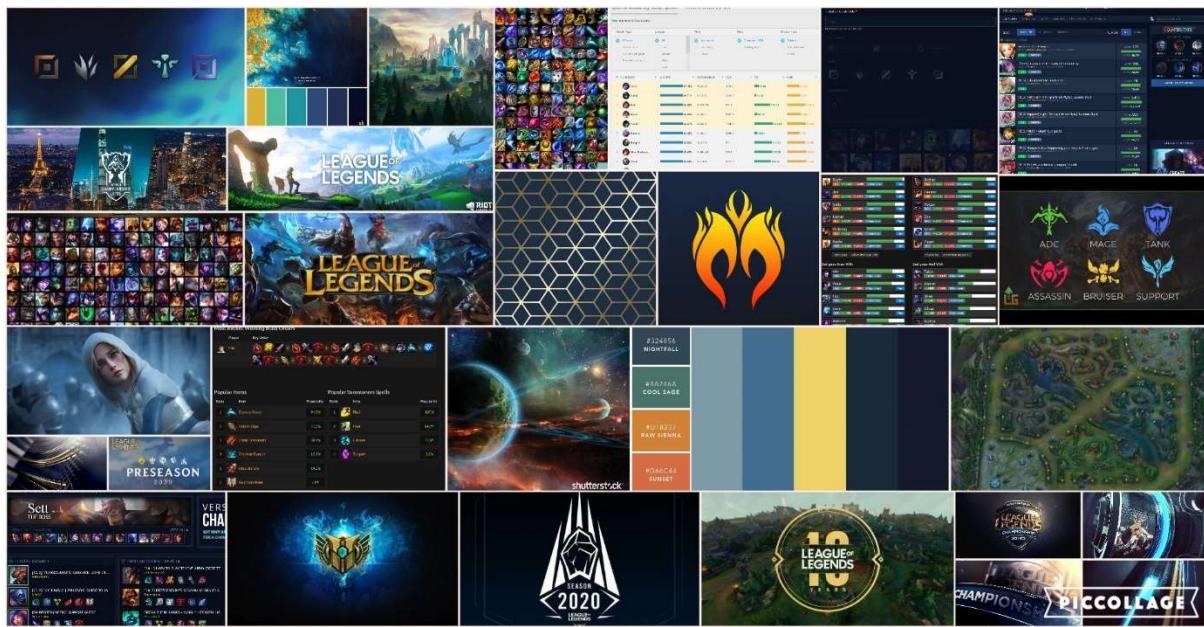
- Read and write guides for champions which mainly cover builds and may be focused on specific scenarios such as a counter
- Easily see the newest or most relevant guides
- Search for guides based on various features such as champion, role, region, patch, level and division
- Create a profile in order to be able to submit guides which displays their summoner name, level, rank, and region as well as the guides they have submitted

The third plane is Structure which considers how users get to each page and where they could go from it. This stage of the design phase is useful for deciding which pages must be designed and created, and the necessary navigation features to navigate between them. The structure helps to define how users interact with the product, how the system behaves when a user interacts with it, and how it is organised and prioritised (Elgabry, 2016). Furthermore, structure is split into two components; interaction design and information architecture. Interaction design defines the very core of human-computer interaction; how users can react with the system and how the system behaves in response to these actions. Good interaction design helps users carry out tasks they set out to complete, makes the user aware of interactivity and functionality, informs users about state changes, and helps to prevent user error or mistakes. Information architecture defines the arrangement of content elements and how they are organised. Good information architecture organises, categorises, and prioritises information based on user needs, makes it easy for users to understand and navigate through the information, and is appropriate for the audience (Elgabry, 2016). At this stage, the pages, the type of content to be displayed on each page, and the navigation features can be decided upon and designed. A helpful way of displaying this is by creating a diagram to show what the pages will be, how users can get to each of them and where they can go from them.



The diagram above shows the structure of the site-to-be. The pages at the top are accessible from all pages (hence the spiderweb of arrows) which would suggest that they would need to be incorporated into a static navigation feature such as a button bar that would appear on all pages in the same style. As well as this, it helps to clarify how users can find a guide they are looking for; perhaps by searching for their champion, then by browsing through the list of guides specifically for them until they find one most suited for them. Or perhaps a featured guide on the homepage sparked their interest so they clicked on it to find out more. An issue with this design is that it lacks detail. It is difficult to portray the different ways and reasons for ultimately arriving at the same place such as in the examples described. This highlights the need for more detailed design to build on this fairly high-level concept.

Before moving onto the next plane, design concepts were developed through the form of moodboards. A moodboard is a collection of assets and materials intended to communicate the style, voice, direction, and language of a particular design or project. (Gidalevitz, n.d.) Before deciding how the site should be laid out, it was useful to have an idea of the general aesthetics and the themes it should portray. An advantage of using moodboards is that they are quick and easy to create so they can easily be changed if necessary and can present various options and comparisons to be able to choose between more clearly. They act as a good foundation in order to move on to the next stage of design and prototyping. As well as this, they encourage creative inspiration as they can be an enjoyable ‘warm-up’ activity to get ideas flowing and spark the motivation to create more advanced designs. A colour palette can often be decided through the creation of moodboards as similarities between elements can be spotted and therefore frequently occurring or well-matched colours can be identified. Since this project entails creating a tool for an existing game with an already defined colour palette, it was essential to ensure a similar colour scheme was being used so that users instantly feel familiarity when using the site. Furthermore, accessibility considerations can be taken into account at this stage. League of Legends is largely usable to those that may suffer with colour blindness so it was important to avoid reds and greens when choosing colours so that the site will be accessible in this way too. The process of creating moodboards proved extremely valuable in identifying the colour scheme of the game and even finding out the exact colour codes used. This meant that the development process could be sped up as the colour codes could be noted and easily added into CSS styling. Furthermore, moodboards can be a quick way to decide upon a style for the design. Rather than having to create several prototypes in different styles which take much longer, UI elements, textures, and fonts can be collated into a moodboard and be decided upon in this earlier stage. Two moodboards were created for this project; one to portray UI elements and colour palettes, another to gather theme ideas and inspirations.

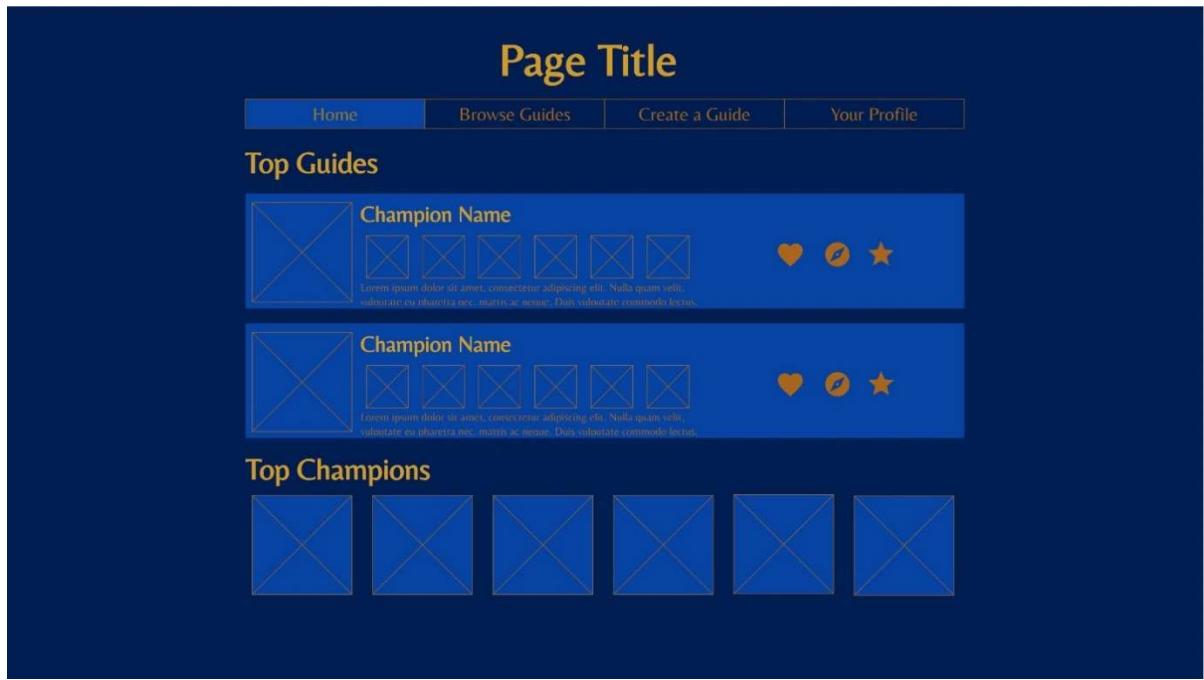


The first moodboard, shown in the image above, combined aspects of League of Legends, the wider community surrounding the game, and existing tools and sites related to it. The colours blue and gold immediately come to mind when the game is considered and finding various images and colour palettes with this in mind sparked the idea of doing more in-depth research to discover the game's real colour palette. Furthermore, this moodboard helped to identify that aspects of the game are largely represented by visual cues; rather than long lists of item names and champion names, players are conditioned to rely on images and icons to identify such things. This is because the game is very fast paced so encouraging players to be able to identify things by a glance saves much more time than having to read them. For this reason, it is clear that the site must contain ample visual cues and not rely on users to type or read. This supports Nielsen's theory of recognition over than recall in usability heuristics; "minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another" (Nielsen, 1994). Since players may be unfamiliar with names, it is essential that the site must offer the same interactivity indications that they will be used to from the game.

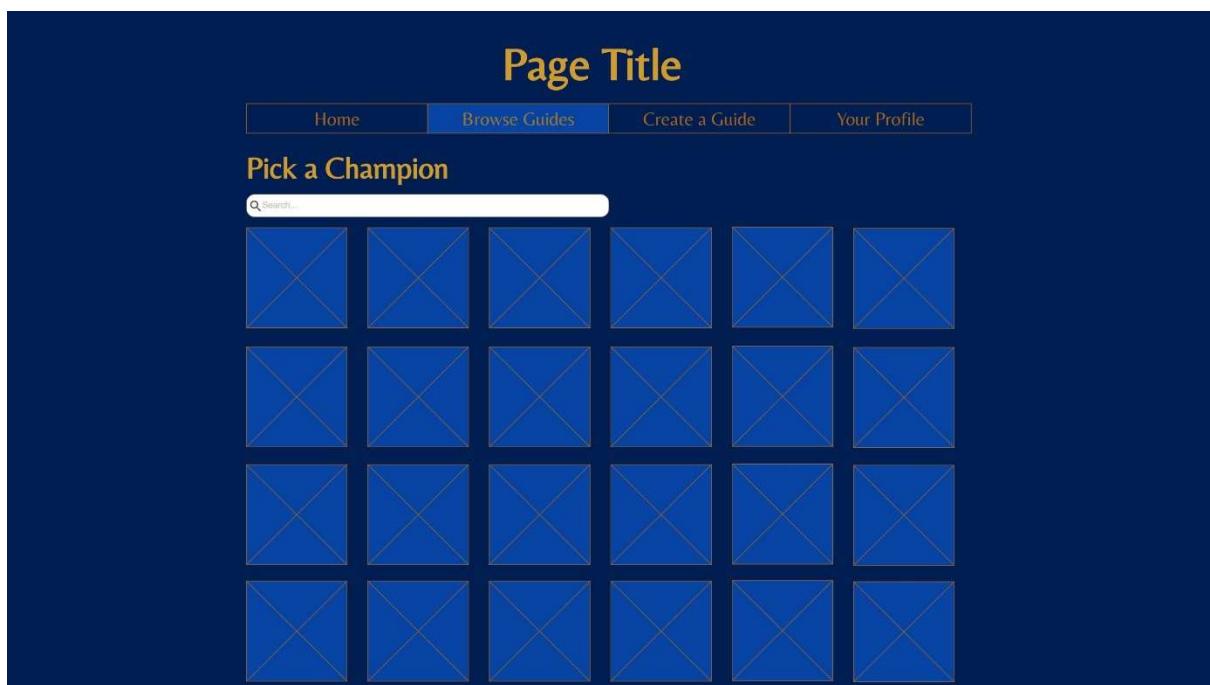


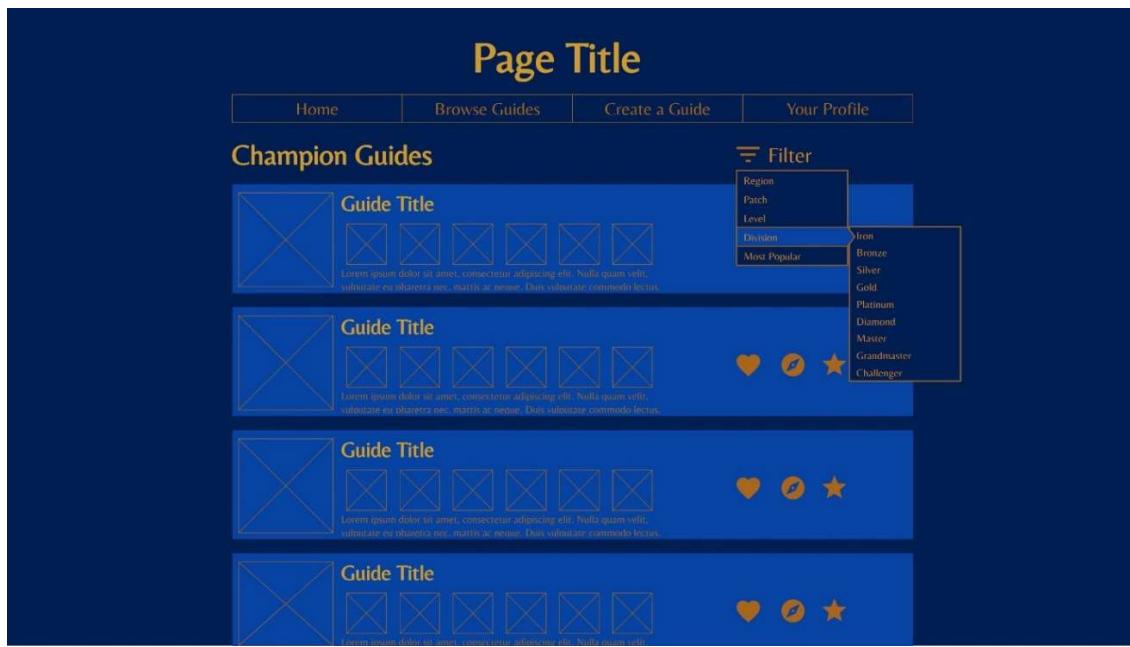
The second moodboard, shown in the image above, involved exploring elements of the game's UI and researching into the colour schemes and UX features. This resulted in the actual colour scheme of the game to be discovered and noted for future use, thus speeding along the development process. Furthermore, the actual font used by League of Legends was found and installed ready to be imported and used both in prototypes and the site. As well as colours and fonts, the styling concepts were put into perspective. For example, through collating the various images, it was clear to see that the design is rather minimalistic and avoids clutter. Other features such as frequently occurring rectangular and circular shapes were also identifiable. Moreover, this process allowed affordances to be determined. Affordances are properties or features of components that present a prompt on its use. While this usually refers to real-world examples such as using a door to exit, the game also has its own affordances that can be used to aid a player's recognition when using an external site. For example, players will be accustomed to seeing an icon of an open book to represent mages, and a golden emblem to represent the gold division. These existing icons and features should be utilised in the site in order to aid the user's understanding and minimise the necessity for reading.

The next stage is the Skeleton plane which adds more clarity by determining the visuals, presentation, and arrangement of elements. In other words, this is where the user interface begins to take shape and interactions are more carefully considered. Accessibility is also considered at this stage. As previously established, the site must be accessible to those with colour blindness, however more disabilities were considered at this stage. Since the game is very reliant on physical reactions and a high level of cognitive processing, the game is not accessible to those with severe motor, visual or cognitive disabilities and therefore are not a target audience of the site. However, the site can be made accessible to people with autism and dyslexia by avoiding clutter and large pieces of text, and by using a large font and many visual cues such as images. The skeleton plane can be broken down into smaller parts; interface design, navigation design, and information design. Interface design refers to how elements will be arranged to enable users to interact with the system, navigation design refers to where navigation features will be and how users can get from one part of the site to another, and information design refers to the presentation of information. The skeleton plane is a useful way of answering crucial questions; what the site will look like, how content will be arranged, how interactions will be presented, how can users navigate, and how information will be presented clearly. A popular design tool during this stage are wireframes. A wireframe is a static sketch or mock-up that represent the visual format of the product and the layout of the content. Wireframes were chosen as a design method for this project because they allow potential problems or oversights to be caught and managed early, and they are much easier to change than code. Furthermore, they can help in prioritising content since a fully functional homepage and its content features should be prioritised before the pages deeper in navigation and less crucial to the functionality of the system. Although wireframes are sometimes sketch-like and monochrome, as the colour palette was chosen from the moodboards, it was helpful to create the wireframes with this scheme in order to visualise the look of the site more clearly. The wireframes created show the main pages that will be included in the website and how they will be laid out. These wireframes provide a useful foundation upon which to begin development as the basic HTML and CSS that will be required is made relatively clear. However, wireframes are not the perfect final design as important features such as navigation cannot be displayed.

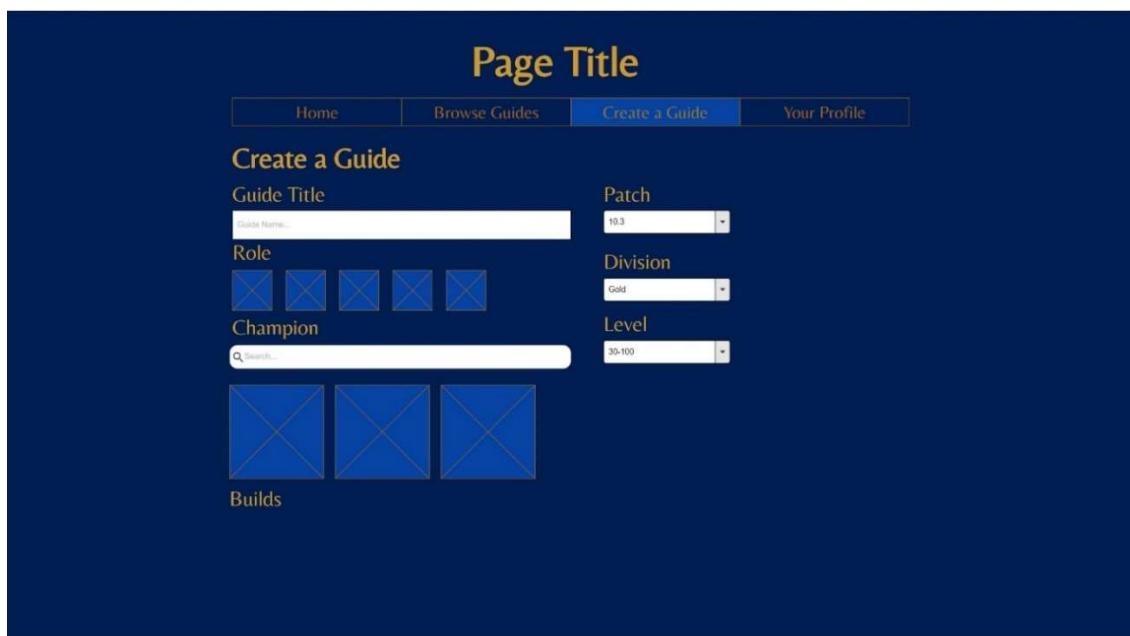


This wireframe represents the layout of the homepage. As can be seen from the many image placeholders, this page will have many images in order to capture the users' interest and allow them to easily find the information they are looking for by avoiding having to read large amounts of text and by using familiar cues from the game. Furthermore, this shows the type of information that will be displayed on this page. Instead of immediately bombarding users with all of the information in the system, it provides a selection of samples such as the most recent guides and most popular champions. This will be helpful for users that aren't looking for something specific as these suggestions might spark inspiration. This page also shows what a preview of a guide will look like. Users will immediately be able to see the name of the guide, the picture of the champion, images of the core build suggested, a short description, and information about the author such as their level, rank, and region. This is ample information to allow the users to decide if they are interested enough to click on a guide to learn more without displaying an overwhelming amount of cluttered information.

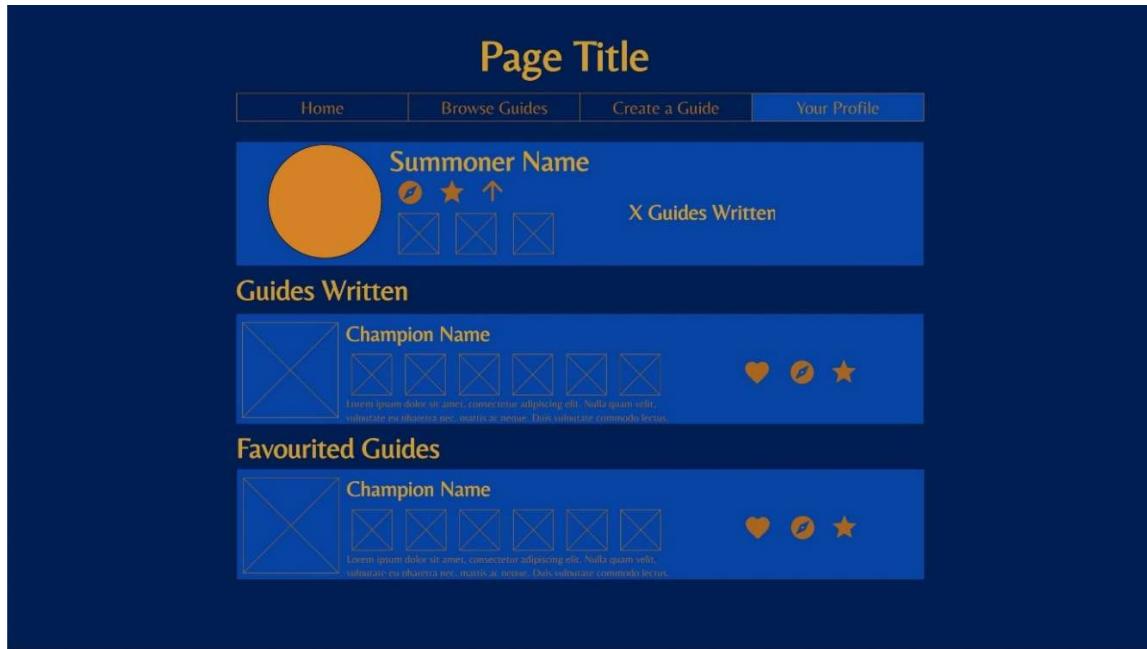




The two wireframes above show early ideas regarding how users will be able to search and browse for guides more specific to what they are interested in. The first wireframe shows a very important feature that must be implemented, the search bar. This would allow users to specify keywords that they are looking for and then would be presented with guides relevant to their search terms. There are around 150 champions in total in League of Legends so rather than having to scroll through a long list, it would often be much quicker for users to be able to type the name of the champion they are looking for. Initially, the idea was to have an array of champion images displayed on the 'browse guides' page so that users could scroll and click on an image of a champion they wish to see guides for. However, it was later decided that this adds an unnecessary stage of navigation and restricts the search field to champion names only. Instead, the 'browse guides' page could display all guides in date order and allow users to search or filter by any criteria they want such as ranks, roles, and regions as well as champion names. A filter tool would also be a good feature to implement to be able to quickly refine by criteria with fewer options such as region but as a search bar can allow users to search for these too, it will be prioritised during development.



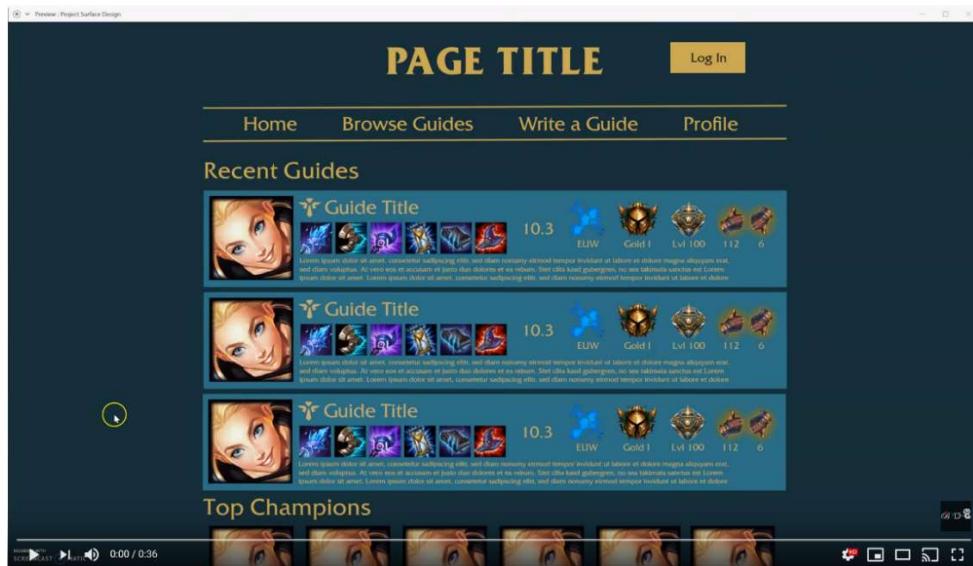
The wireframe above represents an early design of the page on which users will be able to create and submit a guide. This design helped to establish some fields that were important to include such as giving the guide a title, choosing the role by clicking an image of a familiar icon, defining the champion the guide is written for, and the patch version of the game it was written for. Some other important information was also considered such as the rank and level that the guide is aimed at, but later developments revealed that it would be easier for the users if they do not need to enter this information themselves each time they write a guide but if it is instead extracted automatically from their user information entered and signup and updated via their profile. Furthermore, the creation of this wireframe highlighted that more planning was needed as it was still not clear at this stage how the users would actually choose the items in the builds that they recommend.



This final wireframe shows the planned layout of a users' profile page. A selection of profile pictures will be available for the user to choose from that are pre-existent in the game, and there will be no need to implement an upload image system in order to avoid privacy issues. As well as this, each user profile will have a 'bio' section at the top which displays their username and information about them such as their rank, level, region, and how many guides they have written. The profile will also be where users can see all of the guides that they themselves have contributed. However, this design lacks detail such as how the profile information can be edited.

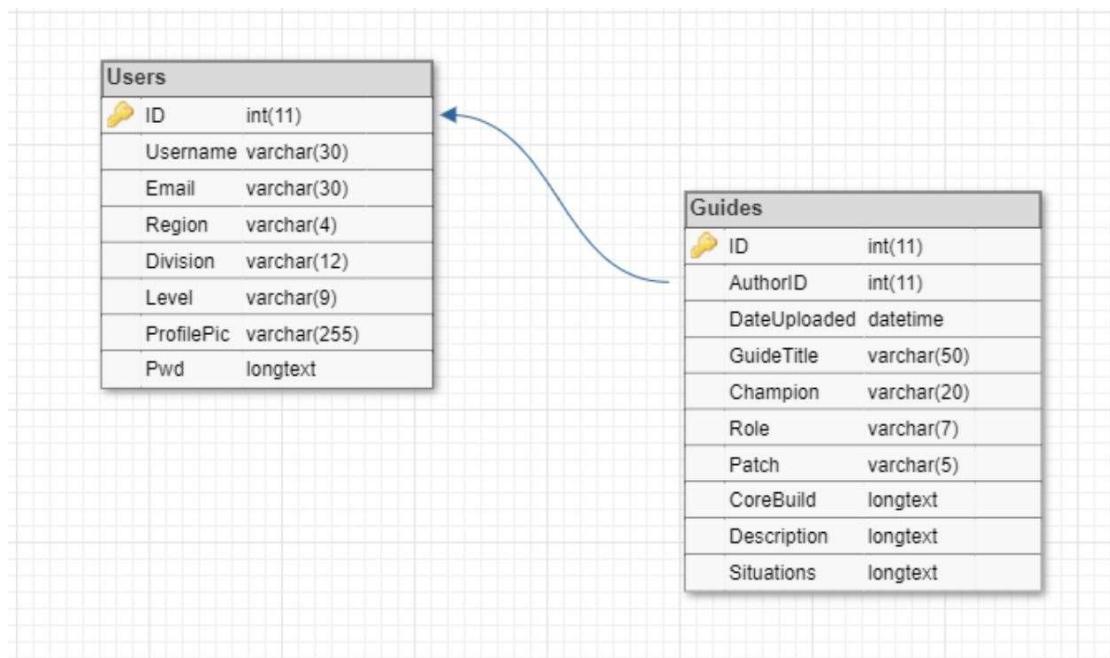
The final stage of the five planes of User Experience is the Surface Plane. This final layer is where the visual features of the system are fully established. Rather than using placeholders to represent the general layout, more specific elements such as images, icons, buttons, and text are used here. Many features of the effects and interactivity are decided at this stage but since CSS and other web-based programming languages are powerful tools, more advanced capabilities may be discovered and implemented during development than can be represented in a design tool such as Adobe XD. Furthermore, the advantage of making an interactive prototype design at this stage not only establishes the static aesthetics of the finished site but also represents some of the basic navigation and interactivity features such as how users can get from page to page and what happens when a button is clicked. The video below shows a demonstration of the interactive prototype design in use. Even from the thumbnail, it is clear that the design is much more established now than compared with the wireframes. Real images from the game have been used, and whilst only a small selection have

been repeated several times in order to save time rather than finding and saving them all, it makes the aesthetics of the system much clearer. Since all of the images from the game will have to be saved during development to be used in the site, it was an inefficient use of time to do this for the prototype as well therefore the variation of images is limited for now. As well as images, the actual colour scheme has been established as exact colour codes were able to be input to match with the game's colour palette as discovered while creating moodboards whereas the wireframe colours were a very inexact approximation. Moreover, decisions regarding functionality have also been made and designed since the surface plane such as all necessary buttons and how the users will be able to add items to builds – a drag-and-drop feature was decided upon. The video shows which elements are to be clickable and what happens when various elements are clicked which helps to gain a deeper understanding of how the site will function and feel. However, not all planned features were able to be represented using this tool such as hover effects, pop-ups/reveals, selections, and the drag-and-drop. Static images of each page designed at this stage can be found in the Appendix (Appendix 2a Figures 28-34).



<https://youtu.be/f6lpcojrlWU>

With the visual designs complete, the functionality can be further planned and designed before beginning development. Designing the login/signup system and the form page to write and submit a guide has made it clear that a database will be necessary to create and the fields it will require. Creating a database schema diagram before beginning development can save time by avoiding having to make frequent changes to the database structure using various tedious statements such as ALTER and DROP. The simple diagram below shows the structure of the database to be implemented. Just two tables will be required; one to store user information and one to store submitted guides. As the focus of this project is on the user experience, the database design is relatively simple so storing large amounts of information with complex relationships between many tables is unnecessary to achieve the goals of this project. The columns in each table and their relative datatypes are shown, many of these were known due to designing the required fields that the users will fill in. The tables are linked by using the AuthorID as a foreign key to the UserID. This will mean that information can be displayed about the user that wrote a guide by using JOIN statements or other simple queries. This diagram has been improved after development took place so while it was not a fool-proof method of avoiding necessary changes, it helped to decide the structure of the information and reduced how much the database needed to be tweaked throughout development.

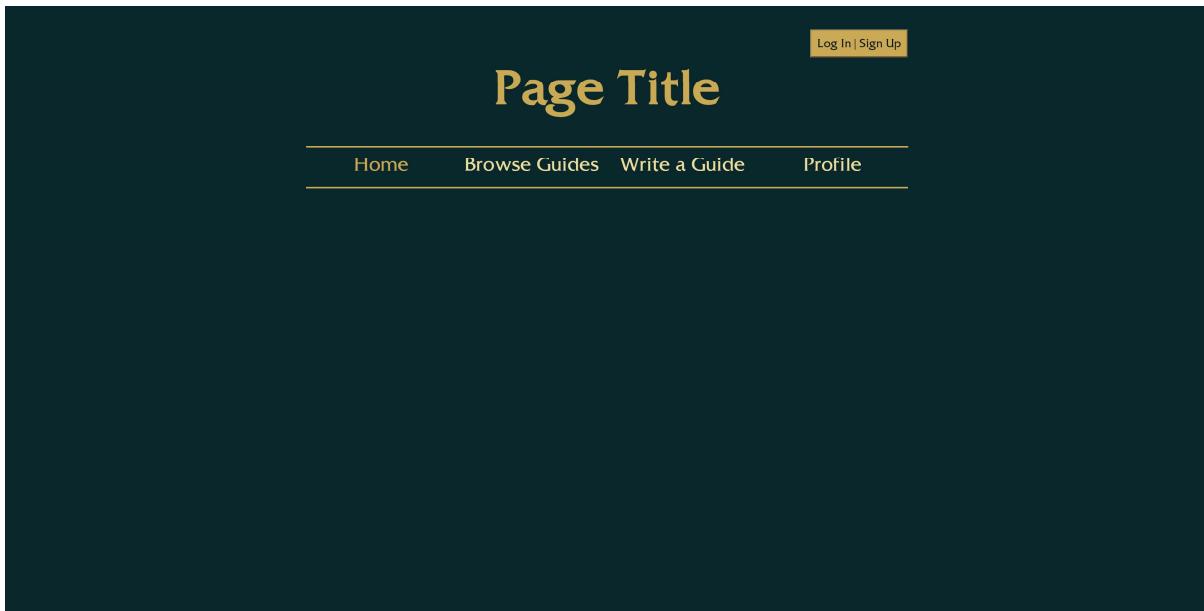


## Development

The development/implementation phase is the central stage of the SLDC Waterfall Model. This is where the actual system is created through programming. This is usually the longest stage of the process as it is often the largest and most complex task. For this reason, it is often broken down into manageable subtasks as shown in the Project Management section. These subtasks are often referred to as units which can then be used to carry out unit testing which is where a single piece of the system is tested to ensure it works by itself before testing how it integrates with the rest of the system. Once the implementation is complete, all of the requirements should have been met which is why it is essential to keep the requirements in mind at all times. Furthermore, factors such as experience can also impact what is done, what is prioritised, and how long is spent on each subtask. For example, previous experience with front end development involving HTML, CSS and SQL helped to speed up these parts of coding however features requiring the much more complex JavaScript/jQuery and the never before experienced PHP required more time due to the learning process and the more advanced nature of the code. As the design section was carried out carefully and thoroughly, few decisions needed to be made regarding aesthetics and layout as these had already been made however, the whole process took longer than expected overall due to limited prior knowledge and several functionality features requiring more careful planning.

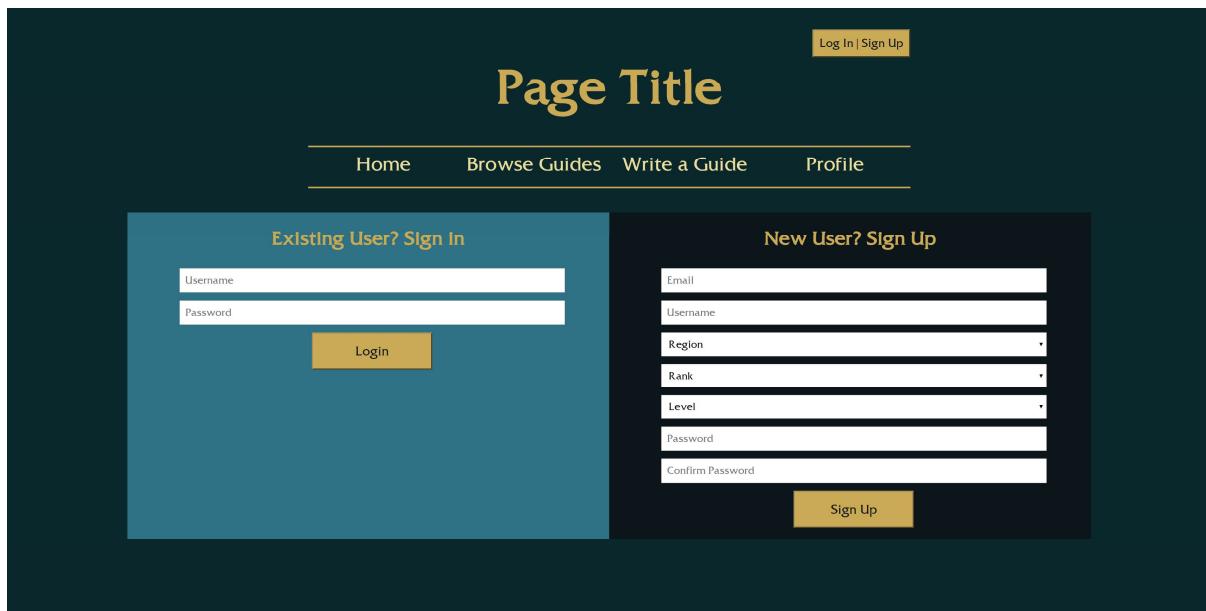
As the first three requirements could be met using front-end programming alone, and previous experience would undoubtedly improve the speed and success in which these could be met, the basic structure and theme of the site was a good place to start development. Since a requirement is that the system must represent the League of Legends aesthetic by using the same colour palette and other design features, development started by copying the colour codes used by the game found in the design process into CSS to make features such as the background and font colour representative of the game. Then, the font used in the game was downloaded and imported into the code ready for use in the site. Even without any content, fans of the game would likely recognise the colour scheme and font and therefore immediately perceive the site as relevant to the game. For this reason, this was an important requirement to meet. Furthermore, an established layout and colour scheme can help to position and style content appropriately in a way that complements the theme and does not clash. The next requirement to meet was to avoid difficult navigation or long wait times by having a minimal

number of pages. So, the next stage in development was to create all the necessary html/php pages and create a navigation feature. As shown in the design, the navigation bar was to remain constant on all pages and contain just four page links; Home, Browse Guides, Write a Guide, and Profile. With this already decided, it was relatively straightforward to implement using HTML and CSS. The third requirement could also be considered at this stage, though required careful consideration throughout the entire implementation process to ensure it was met since it is affected by every element implemented. This requirement was to make the system usable on various screen dimensions/resolutions. This was considered from the beginning by using percentages rather than pixels in sizing in CSS styling. This means that rather than an element always being a specific number of pixels tall and wide, it adjusts to fill a certain percentage of the screen or the div element it is contained within. Appendix 3a Figures 35-37 show the HTML and CSS used to achieve the results shown in the image below.



The next requirement to tackle related to the first main future of functionality; the log-in system. Since parts of the site will only be available to those that sign up and log in such as the ability to write and submit guides and have a customisable profile page, and the guides that those users submit make up the majority of the site's content, implementing this system was a priority. Furthermore, since there are only so many ways to create a login system and it can be relatively straightforward since no JavaScript is required, this was a good starting point to learn the basics of PHP since it is well documented. The first step was to implement the front end – to create a html form with the fields that the users will see and be required to fill in. In order to keep the number of pages to a minimum as required, it was decided that both the sign-up and log-in forms would be displayed on the same page. This technique is used by many popular websites such as Facebook and it makes it easy for both new users and returning users to find what they are looking for in order to access the site as easily as possible. The login form is very simple, with just two fields required; username and password. However, since League of Legends players have many stats relating to the game, these were required at signup since the region they play in and the rank they are would likely to be of interest to others reading guides, since recommendations can differ greatly between regions and ability levels. Furthermore, entering this information at signup means that the user will only have to enter it once which will therefore avoid frustration since they won't have to enter these details on every guide they write individually. The image below shows the result of the login page with all styling completed, the code for the HTML structure can be seen in Appendix 3b Figures 38 and 39. Since signing up is required

in order to be able to login, it was logical to implement the backend of the signup system first. Firstly, a database was made by using SQL code in phpMyAdmin which is part of the local hosting service, XAMPP. Inside the database, a table was created to store users' information which included all fields required at signup, an automatically assigned ID, and a default profile picture. The database diagram created in the design phase was used as reference to create the database table. The SQL code used, and the resulting table can be seen in Appendix 3b Figures 40 and 41. Then, the HTML form needed to be linked to an unseen PHP script which would make necessary checks such as empty fields, invalid email addresses, or unmatching passwords as well as sending the data that the user inputs to the database providing these checks are met. In order for this to work and to make code reusable, another php script, otherwise known as an include, needed to be made which simply connects the site to the database. So, the signup include requires the database connection include and starts by assigning variables based on the names of the fields in the html form. Then, a series of if else statements are made to verify the users input to ensure that erroneous or missing information does not get sent to the database. If and when the user enters valid information, the php script then sends this data to the database. However, in the interest of security, the data is passed using prepared statements which are resilient against SQL Injection attacks since variables are not bound to the statement and therefore are passed indirectly. As well as this, passwords are hashed to be stored in the database so that nobody with access to the database can read what they are. Although no sensitive data is to be stored, it is always beneficial to users to protect their data of any kind in order to maintain trust and build a positive user experience. As well as this, since people may use the same passwords for different services, it is important to protect these. Snippets of the PHP include can be seen in Appendix 3b Figures 42 and 43. Then, since visibility of system status is one of Nielsen's 10 usability heuristics, it was important to make it clear to the user whether their signup attempt was successful or not and what they must do to fix any errors. This was done inserting php code into the webpage containing the forms and displaying various messages such as "success" or "Error: You must fill in all fields" based on the results of the checks performed in the script. The code snippet of implementing these error messages can be seen in Appendix 3b Figure 44. The script for the login system was somewhat similar to the signup and parts could be reused, speeding up development. However, rather than sending information to the database, the information in the database needed to be read and compared with the users input. The only error handling necessary for this part was to ensure that the fields were not empty and that the user input matched the information stored in the database. If these criteria are met, then a session is started for the users' information and they are therefore 'logged in'. The code snippet showing how checks are made for an existing username, correct password, and starting the logged in session can be seen in Appendix 3b Figure 45.



Next, now that users were able to signup and login, and the remainder of the site's content largely relied on user input via guide submissions, the logical step was to implement the core functionality of the website – the system to write and submit guides. This was the largest and most challenging part of the development process as it required methods that had not yet been experienced and required substantial amounts of PHP and JavaScript. Furthermore, since this page was to be highly interactive and design tools have limited interactive capabilities, lots of decision making and experimentation was required in order to decide upon how the features of interactivity were to be implemented. As with the login system, the first step was to create the front end, involving forms. The HTML code to create the structure was very basic, but as there were many champion and item names to list, the code was very long and tedious. Luckily, some time was able to be saved since forms had been styled for the login page so the layout and styling could be reused or copied for the forms used in the guide submission form. However, some fields presented opportunities to improve user experience and interactivity beyond simple forms. Firstly, the five roles in the game have distinct icons which players are most likely to recognise, therefore, rather than requiring the user to type the role the guide relates to, a selectable image feature was implemented instead. Whilst the jQuery code to make the icons selectable was fairly straightforward, as shown in Appendix 3c Figure 46, this feature added complications down the line when it came to sending the data to the database. Another interactive feature implemented was a drag and drop feature. This was implemented so that users could easily drag the items they are recommending to a series of boxes. As there are many items in the game, it would have been very tedious to make the users type the names of them or to scroll through a long list to make a selection. Furthermore, many players may be unaware of some item names since names are rarely seen and used in the game, instead, selections from images are made. Appendix 3c Figure 47 shows an example of the item shop in-game, which shows that users are presented with the image of items along with their price, therefore names may be far less recognised than the icons of the items. For this reason, the aim was to provide users with a similar and recognisable experience when using the website as they would find whilst playing the game. As well as this, this helped in meeting the requirement that 'information must be easy to access and to visualise so the system must use visual cues and avoid large amounts of text where possible'. The jQuery code used to implement this drag and drop feature was rather complex and took longer than anticipated to implement – the code can be seen in Appendix 3c Figure 48. The next feature to implement was very important as it is the feature that makes this project unique and sets this website apart from existing guide sites. This is the feature

to not only recommend a single build, but instead several builds with advice on when to adapt situationally. For example, what a champion might build when they are doing well is very different to what they should build when they are struggling. Building correctly is something both novice and adept players struggle with, so it was important to implement this feature of functionality and uniqueness effectively. This was done by adding a button which could be pressed several times, each time revealing new form fields for the user to fill in to recommend a situational build. The jQuery used to reveal these additional forms at the click of a button can be seen in Appendix 3c Figure 49 and the results in the site can be seen in Figure 50. With the front end complete, it was time to send this data to the database. Firstly, a new table needed to be added to the database to store the guide information. This was based on the diagram made during the design phase and the SQL code used to create the table can be seen in Appendix 3c Figure 51 and the resulting database in phpMyAdmin can be seen in Figure 52. A foreign key was used to link the guides table to the users table so that information about the author of the guide can be obtained and displayed for future use. Then, since many elements were created using JavaScript, AJAX had to be used to send the data to a PHP include for it to be sent to the database as it could not be sent directly. Due to inexperience with AJAX, this was a long and complicated stage during development and therefore took longer than anticipated. The AJAX code used in JavaScript to pass this information can be seen in Appendix 3c Figure 53. Since there were only five role icons, it was simple to convert the image data to word in a string to be stored in the database but since there were many item images, and each build requiring six items, it was necessary to store the image file location as a string to be stored in the database. This was why these elements were ‘stringified’ using JSON. Once this data had been passed to a php include, PHP code and SQL queries could be used to send the data to the database. Since this was already done for the login system and much of the work had been done using AJAX, this was fairly straightforward, and this code can be found in Appendix 3c Figure 54. The resulting user interface can be seen in the image below.

The screenshot displays a user interface for creating a game guide. On the left, there are several input fields and dropdown menus:

- Guide Title:** A text input field with the placeholder "Guide Title".
- Champion:** A text input field with the placeholder "Start typing Champion name...".
- Patch:** A dropdown menu showing "Patch".
- Role:** A dropdown menu showing five icons representing different roles: Support, Marksman, Mage, Tank, and Fighter.
- Core Build:** A row of six empty square input fields for selecting items.
- Description:** A text area with the placeholder "Explain this build...".

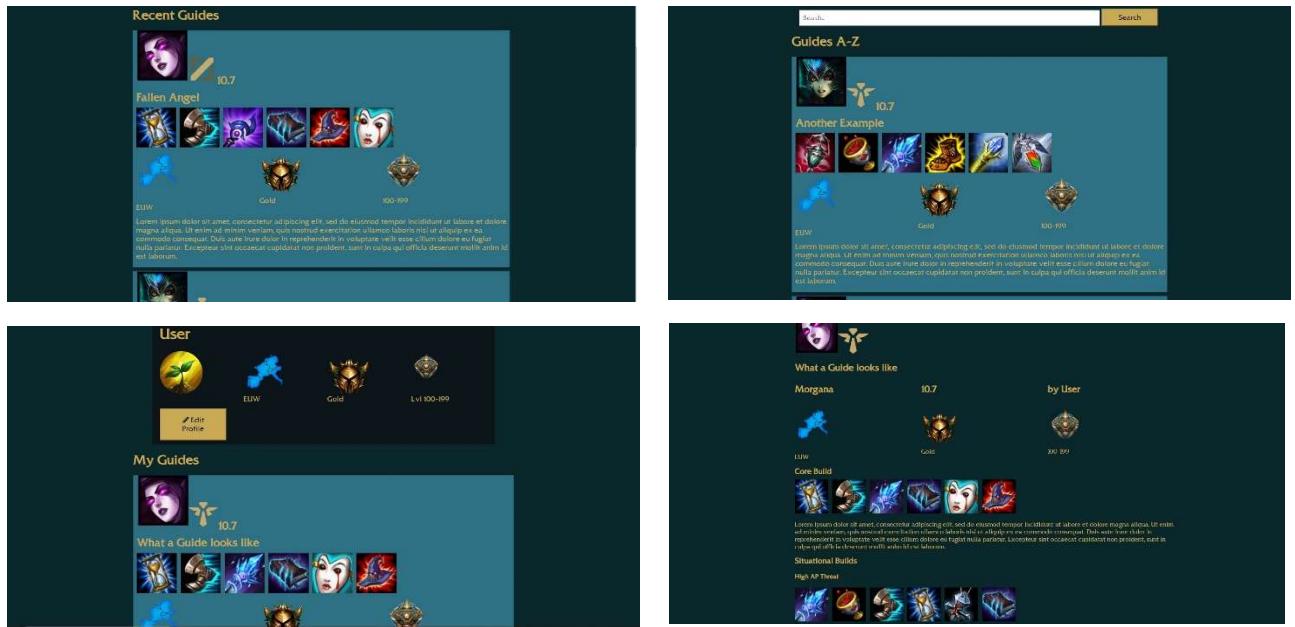
On the right, there is a large grid of 36 item icons, arranged in a 6x6 pattern. The top-left cell contains the text "Drag and drop items onto the build slots...". The items represent various in-game items from a game like League of Legends.

At the bottom of the screen, there are two buttons:

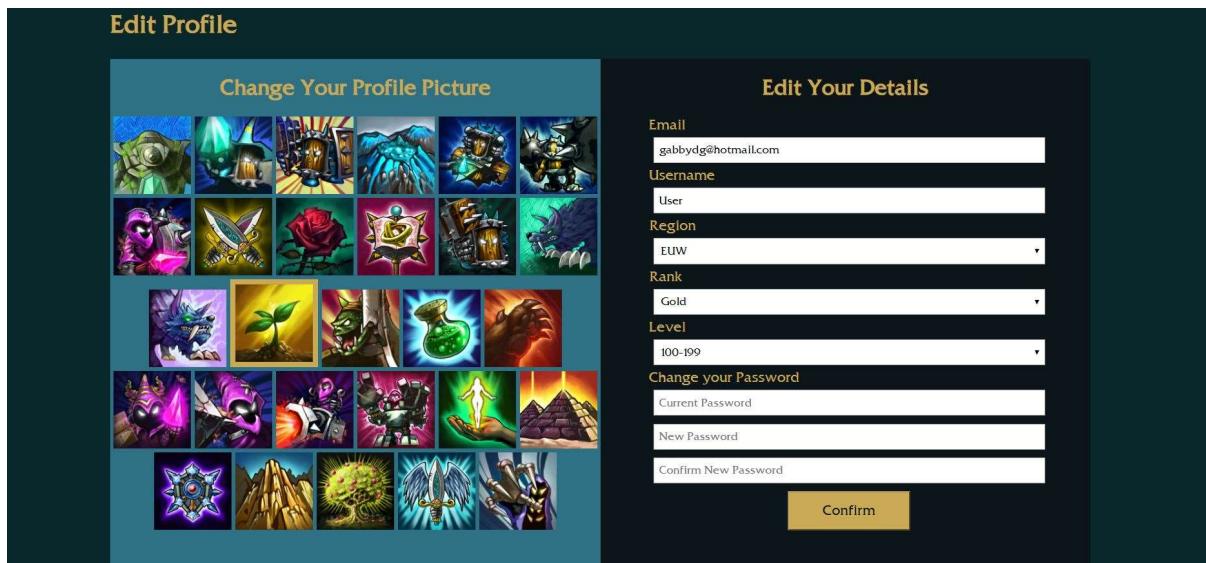
- A blue button labeled "Add a Build Adaptation".
- A yellow button labeled "Submit Guide".

Now that guides were able to be written and stored, it was time to display these guides in order to fully meet the requirement that ‘logged-in users must be able to create and submit guides that will be automatically displayed to all users’. There are three different pages on which guides needed to be

displayed in various ways. Firstly, as ‘the system must allow all users to easily find guides they are looking for and refine their searches by Champion, Role, Level, Rank, Region and Patch’, a page in which users would be able to leisurely browse or search for specific guides needed to be implemented. This was less complex than the previous page as no new background scripts needed to be implemented. The information was retrieved by performing various SQL queries which would get all of the guides in date order by default. The code used to do this can be seen in Appendix 3d Figure 55. In order to avoid overloading the user with information or making the site cluttered, this view only shows previews of the guides, revealing key information such as the Champion, title, and core build. If the user would like to learn more, they will be able to click on these previews to see more information such as situationally adapted builds. The while loop shown at the end of the code snippet then goes on to a lengthy series of if else statements to display the correct Champion image based on the Champion name. Again, this was done because of Champion image often being used in the game as queues rather than text so this was done to ensure that users would be able to easily recognise which Champion a certain guide relates to, which may be particularly useful for newcomers who have not learned the long list of names. Appendix 3d Figure 56 then shows the code snipped used to get and display item images and information about the author, and the further details are not shown in this view. The next step was to implement a search bar so that users could search for what they are looking for. This process was simpler than expected, firstly, a search bar was added by implementing a form similar to the kind that had already been made. Then, a button was added next to the search bar. If this button is pressed, the user is taken another page which displays the result of their search. The code used to return these search results can be seen in Appendix 3d Figure 57. The rest of the code inside and after the while statement is largely identical to the previous page, so this saved some time. A very similar approach was used to display guide previews on the homepage and profile pages. For the homepage, the three latest guides are shown by pulling the first 3 guides in the database in date order in order to grab the user’s attention with the most up-to-date guides. For the profile page, all the guides where the author ID is the same as the current user’s is shown on their profile page. The SQL query used to do this got the guides where the username is the same as the current session, as the user must be logged in to view their profile and therefore the guides they have written. With these previews in place, the further details could then be implemented to display to the user. This was done by making the divs containing the previews clickable links. The user is then taken to a new page that displays all the information about the guide they have clicked on. However, rather than having a different page for every guide, the page content changes depending on the ID of the guide. So, if the user clicks on a guide that has the ID of 2, the details page will get the guide where the ID matches. The code to do this was very similar to before, except WHERE is added to the SQL query to get the guides where the ID matches the ID variable passed when the div was clicked. Then, after reusing some of the code to display the basic information, the code used to display some details such as situational builds can be seen in Appendix 3b Figure 58. This was very similar to the code for displaying the core build but since the name, items, and description of situational builds were stored as one string in one database column, the parts of the string to display needed to be specified. The resulting pages can be seen in the images below.



The final step of development was then to implement a feature in which users could update their details which would display on their profiles and the guides they submit. Whilst users are required to enter their game stats at signup such as their rank and level, these will change the more users play the game and improve their stats. Therefore, a means of updating their information was essential. Firstly, a button was implemented on the profile page which takes user to another page where they can edit their details. To save time, the structure of the login page was reused to create the page where information could be edited. The login form on the left was replaced with a means in which users could change their profile picture based on a selection available but due to time constraints and the low priority of this feature, this was not fully implemented. However, the signup form on the right was kept largely similar, except an additional field was added so that users could update their password by typing it in twice as well as entering their current password for security reasons. These password checks prevent users from mistyping their desired password and prevent others maliciously changing their passwords without knowing their original password, therefore locking the user out of the site. Furthermore, to prevent users from updating their details to blank fields and to save the user time and prevent frustration, the users' current information is filled in by default. This means that they can change only the fields they wish to change, and the others will preserve their current information. A check was performed so that no fields could be left empty, if the user had to fill in all fields even if they do not wish to change that particular field, this would lead to a poor user experience. So, as long as the user does not delete the information in the fields and only changes the ones they wish to update, the process of updating their details should be relatively straightforward and frustration-free. The backend PHP script code used to do this was somewhat similar to the signup include, however SQL queries using UPDATE statements were used instead. A snippet of some of the code used to do this can be seen in Appendix 3e Figure 59. The resulting update details form can be seen in the image below.



## Testing

Testing is the penultimate phase of the waterfall model in which the fully developed system is tested in a variety of ways to assess its functionality, identify bugs, and validate its usefulness to the user. In this project, the finished system was tested in two ways; system testing was performed to identify potential bugs and assess whether it functions as expected (this combined integration and functionality testing), and user testing to validate its usability from the perspective of a potential user.

### Software Testing

If no testing takes place until development is fully completed, it is very likely that there will be many bugs that were not identified whilst coding. However, although the process was planned to be linear, development and testing were carried out iteratively. As an inexperienced and therefore unconfident programmer, tests were performed continuously throughout development to check that a piece of code did what it was expected to do. This was done because it is easier to identify what part of the code is causing the issue if it is tested immediately, rather than having to search through a completed set of code. For this reason, the test carried out in the table below were largely successful as all bugs had been identified and fixed during development. Furthermore, the test that failed was not due to a bug but rather due to unfinished development due to the added time pressures caused by the current pandemic. As this feature of functionality was low priority, it was not fixed as more important matters needed to be prioritised more urgently. As the system testing was performed by the developer, this testing type is known as 'white box testing'. White box testing is a software testing method in which the internal structure/design/implementation of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs, and programming know-how and the implementation knowledge is essential (Software Testing Fundamentals, n.d.). Whilst the first stage of documented testing is usually unit testing, this had largely been completed during development and therefore was not fully documented or done linearly. Unit testing is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software (Software Testing Fundamentals, n.d.). In order to save time and due to the lack of modularity in the system, the first documented testing phase combined both integration tests and functionality/system testing, as well as some unit tests. Integration testing is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units (Software Testing Fundamentals, n.d.). Almost all components of the developed system were tightly integrated and

interact with one another in order to provide a fluid user experience, therefore it was essential to test that all components work together as intended. Functionality testing (also known as system testing) is where a system is tested for its compliance with the requirements and therefore, essential components of functionality defined in the requirements stage were tested. Testing is carried out to ensure product quality, security, and user satisfaction.

Test No.	Priority	Previous Tests Required	Test Description	Input	Desired Output	Actual Output	Status
1	High		Testing that a new user can sign up when valid data is entered into the sign-up form	Email: "gabbydg@hotmail.com" typed Username: "User" typed Region: "EUW" selected Rank: "Gold" selected Level: "100-199" selected Password & Confirm Password: identical password typed in each field	A message informing that the user has been signed up successfully appears. The data entered in each field appears in the database in the relevant columns.	A message informing that the user has been signed up successfully appears. The data entered in each field appears in the database in the relevant columns.	Success
2	Normal		Testing that users cannot sign up if a field is left blank	Email: "test@test.com" typed Username: no input Region: "EUW" selected Rank: "Gold" selected Level: "100-199" selected Password & Confirm Password: identical password typed in each field	An error message appears informing that the user cannot sign up unless they fill in all fields. No data is sent to the database.	An error message appears informing that the user cannot sign up unless they fill in all fields. No data is sent to the database.	Success
3	Normal		Testing that users must enter a matching password twice before being able to signup	Email: "test@test.com" typed Username: "Test" typed Region: "EUW" selected Rank: "Gold" selected Level: "100-199" selected Password: "123" typed Confirm Password: "abc" typed	An error message appears informing that the user cannot sign up unless the passwords match. No data is sent to the database.	An error message appears informing that the user cannot sign up unless the passwords match. No data is sent to the database.	Success
4	Normal		Testing that users that are not logged in cannot see the content of the "Write Guide" and "profile" pages	No login information is entered, the page tabs to the "Write Guide" and "Profile" Pages are clicked	The two pages display a message informing the user that they must login to access them.	The two pages display a message informing the user that they must login to access them.	Success
5	High	1	Testing that a user who has	Username: "User"	User sees a message	User sees a message	Success

			signed up can login to the system using correct information	Password: identical password to the one used to sign up in test 1	confirming that they have successfully logged in, they can now access the 'write a guide' and 'profile' pages	confirming that they have successfully logged in, they can now access the 'write a guide' and 'profile' pages	
6	Normal		Testing that users cannot log in unless they enter valid information	Username: "NoUser"	User is presented with an error message informing them that the username they have entered does not exist, the user is not logged in.	User is presented with an error message informing them that the username they have entered does not exist, the user is not logged in.	Success
7	Normal	1	Testing that users cannot login without the correct password	Username: "User" Password: any password different to the one used at signup in test 1	User is presented with an error message informing them that the password they have entered is incorrect, the user is not logged in.	User is presented with an error message informing them that the password they have entered is incorrect, the user is not logged in.	Success
8	High	5	Testing that a logged in user is able to submit a guide	The "Write a Guide" page is navigated to. Guide Title: "Example Title" typed Champion: "lu" typed, then "Lux" is selected from the drop-down menu of suggestions Patch: "10.7" is selected Role: the support icon is selected Core Build: six images are dragged from the right into the slots Description: "test" typed 'Submit Guide' button clicked	A message appears informing the user that their post has been saved.  The Data is sent to the database in the relevant columns.	A message appears informing the user that their post has been saved.  The Data is sent to the database in the relevant columns.	Success
9	Normal		Testing that a guide cannot be submitted that lacks required information	The "Write a Guide" page is navigated to. Guide Title: "Example Title" typed Champion: no champion is selected Patch: "10.7" is selected	The user is presented with an error message prompting them to fill in all fields.	The user is presented with an error message prompting them to fill in all fields.	Success

				Role: the support icon is selected Core Build: six images are dragged from the right into the slots Description: "test" typed 'Submit Guide' button clicked	No data is sent to the database.	No data is sent to the database.	
10	Normal	8	Testing that the guides on the homepage are sorted in date order	Immediately after performing test 8, the homepage is navigated to	The submitted guide is visible at the top of the homepage as the most recent guide.	The submitted guide is visible at the top of the homepage as the most recent guide.	Success
11	Normal	8	Testing that the guides on the 'Browse Guides' page are sorted alphabetically	Immediately after performing test 9 (after performing test 8), the 'Browse Guides' is navigated to	The submitted guide is visible in the A-Z list in the correct positioning.  The guide submitted about "Lux" is at the top of the list as other guides submitted were about "Morgana", "Nami" and "Sona", all of which are after L in the alphabet	The submitted guide is visible in the A-Z list in the correct positioning.  The guide submitted about "Lux" is at the top of the list as other guides submitted were about "Morgana", "Nami" and "Sona", all of which are after L in the alphabet	Success
12	Normal	8	Testing that a logged in user can see the guides they have written on their profile page.	Whilst logged in as the user who submitted the guide in test 8, the profile page is navigated to	The submitted guide is visible on the user's profile.	The submitted guide is visible on the user's profile.	Success
13	High	8, 10, 11, 12	Testing that clicking on a preview of a submitted guide takes the user to a page displaying further information about the correct guide.	The preview of the guide submitted in test 8 is clicked on all 3 pages it appears	The user is taken to a new page which shows all guide information about the correct guide including what is not shown in the previews such as situational builds.	The user is taken to a new page which shows all guide information about the correct guide including what is not shown in the previews such as situational builds.	Success
14	Normal	5	Testing that a logged-in user is able to update their details	The 'edit profile' button on the profile page is clicked, The Rank field is changed from Silver to Gold, and the	A message appears informing the user that their details have been	A message appears informing the user that their details have been	Success

				'confirm' button is clicked.	successfully updated, the relevant column in the database is updated, and the image and word displayed on the user's profile and the guides they have written are updated accordingly.	successfully updated, the relevant column in the database is updated, and the image and word displayed on the user's profile and the guides they have written are updated accordingly.	
15	Normal	5	Testing that a logged in user is able to change their password	The 'edit profile' button on the profile page is clicked, The current password is correctly entered in the 'current password field' and two identical new passwords are typed in the 'new password' and 'confirm password' field.	A message appears informing the user that their details have been successfully updated, the relevant column in the database is updated (though is hashed), the user is able to log back in using their new password.	A message appears informing the user that their details have been successfully updated, the relevant column in the database is updated (though is hashed), the user is able to log back in using their new password.	Success
16	Normal	5	Testing that a logged in user is not able to change their password unless the password is typed twice and matches.	The 'edit profile' button on the profile page is clicked, The current password is correctly entered in the 'current password field' and two different new passwords are typed in the 'new password' and 'confirm password' field.	An error message appears informing the user the passwords they entered do not match. No database information is updated.	An error message appears informing the user the passwords they entered do not match. No database information is updated.	Success
17	Low	5	Testing that a logged in user is able to change their profile picture.	The 'edit profile' button on the profile page is clicked, and a different profile image is selected.	The image they selected is updated into the database table and displayed on their profile	The image appears selected, but no change is made to the database or their profile	Failure

## Usability Testing

Usability testing refers to evaluating a product or service by testing it with representative users (usability.gov, n.d.) and is done to evaluate how satisfied potential users are with the system as well as identify potential changes that could be made to improve user satisfaction. As discussed in the

Research Methods section, a method of interacting with participants such as focus groups would have been more ideal if not for the Coronavirus outbreak, however the chosen evaluative method is very relevant in the HCI field. The chosen method of evaluating usability was the System Usability Scale (SUS). The SUS was originally created by John Brooke in 1986 and provides a “quick and dirty”, reliable tool for measuring the usability. It consists of a 10-item questionnaire with five response options for respondents; from strongly agree to strongly disagree. SUS has become an industry standard, with references in over 1300 articles and publications. The noted benefits of using SUS include that it: is a very easy scale to administer to participants, can be used on small sample sizes with reliable results, and is valid – it can effectively differentiate between usable and unusable systems (usability.gov, n.d.). The form with the SUS sent to users can be seen in Appendix 4a. These are a predefined set of questions required to calculate the SUS score, along with an optional section for participants to explain their answers and give additional feedback in attempts to substitute the qualitative information usually obtained in more interactive methods such as focus groups. In order to calculate the SUS score, 1 is subtracted from the responses of each odd numbered question (user response – 1) and the responses to each even numbered question is subtracted away from 5 (5 – user response). Then, these are all added together which gives a total score ranging from 0 to 40, with 0 being the lowest usability and 40 being the highest. To get a score out of 100, this result is simply multiplied by 2.5. Whilst users usually fill out a SUS survey after they have interacted with the system personally, time and resource constraints meant that the site could not be hosted online and therefore easily shared with testers. For this reason, testers were instead showed a video approximately 4 minutes in length showcasing the entire functionality of the site, and then asked to rate the usability based on what they saw in the video. Whilst this method was somewhat biased, testers felt that they were given enough insight to fairly rate the site’s usability. Due to time pressures and the situational difficulty of social interaction, just three participants filled out the SUS survey. However, since the SUS is valid and reliable with even small sample sizes and focus groups are usually small, this number was adequate for this study. Furthermore, the small number of responses meant that feedback received in open-ended questions could be more carefully analysed. The first participant gave the website an SUS score of 90, the second resulted in a score of 85 and the third participant gave a rating of 92.5. These scores resulted in a mean SUS score of 89 out of 100, indicating that the developed site was highly usable. The table shown below in Figure 17 shows that any SUS score above 80.3 is equivalent to ‘A Grade’ or ‘excellent’ usability, therefore suggesting that the project was successful in prioritising user experience.

SUS Score	Grade	Adjective Rating
> 80.3	A	Excellent
68 – 80.3	B	Good
68	C	Okay
51 – 68	D	Poor
< 51	F	Awful

Figure 17: SUS Scores Interpretation (Alathas, 2018)

Moreover, this user feedback form did not only allow for a quantitative analysis of a usability score, but also allowed for further insight to be gathered. The first question regarding the frequency of the system resulted in some very positive feedback validating the purpose of the website. “The site looks

very accessible and easy to use; I could see myself bookmarking this for use when playing new champions and/or to adapt builds of champions I already play.” “When learning new champions, I would definitely use this system, it would be beneficial to learning how to play the champion optimally.” These responses gathered from experienced players indicate that they would find reading guides submitted by others useful in helping them play suggesting that the guidance provided would not only help new players, but also those seeking to improve their skills further. Furthermore, the requirement of providing users with a familiar interface to what they would be used to from the game such as colours and affordances was validated with the following response: “the system is quite similar to the current system already in League of Legends, give and take a few extra pieces of information such as the rank of the champion in the current meta, so it seems easy to adjust to”. Another requirement, that the system must be quick and easy to use, was also validated by the following responses: “the system looks easy and simple to use, making it accessible, quick and easy for people search and look for guides, as well as creating and sharing one of their own” and “searching for guides etc. is quick and easy to use. The drag and drop features on the guide builds are simple and easy to use”. These factors are important because players may be seeking to find guidance quickly in the middle of a match. Although this user engagement resulted in an overwhelming amount of positive feedback, some constructive criticism was also provided. One user said: “the only scenario where someone would need to learn a lot of things would be if a new player was to make a guide, as they would require game experience and knowledge. But that is the purpose of the site. These new players will be able to learn more about the game and how to build champions through the existing guides on the site.” Whilst the aim of the site is so that experienced players can be the ones submitting guidance to help others, particularly those less experienced, this response highlights a valid issue. There is nothing stopping a new player from wanting to sign up and create a guide themselves, and this would be difficult to do with limited game knowledge. So, this presents an area where the system could be improved. Potential improvements will be discussed in the following section. All of the responses gathered from the SUS questionnaire can be seen in Appendix 4b.

### Potential Future Operation & Maintenance

Whilst the site is highly usable and successfully met requirements, there are several ways in which the system could be developed further given more time and resources. Firstly, an aspect of functionality was not fully developed in time for the deadline, this was the feature that would allow users to change their profile pictures based on a selection available. More customisability and freedom often results in a better user experience. Furthermore, the selection of profile pictures made available was the same as those available when a new player signs up to the game so this feature would be another way of providing users with an aspect of familiarity from the game, as they would likely recognise these images. Therefore, if development were to continue, fully implementing this feature would be prioritised. A feature that was considered during the early design stages and then scrapped later in development in order to save time and reduce the complexity of the system was the idea of a rating system. This would allow users to rate guides that they see, both so that they can easily find them later and so that the highest rated guides would be suggested to all users on the homepage, promoting the highest quality and most reliable content. Whilst this feature was not essential to the core functionality of the system, I believe it would be beneficial to implement given further development to further improve user engagement and overall experience. As well as simple ratings, comments could be added to guides in which other users could make suggestions if they don't agree with a recommendation or ask questions if they are unsure about why a certain item is being recommended, for example. However, as a commenting system can open doors to malicious and hateful posts appearing on the site, it is ideal to implement some sort of report or moderation system so that inappropriate comments could be removed. Exposing users to unkind words can drastically worsen

user experience which was why the comment system was avoided during this iteration. With further development to ensure it could be appropriately managed, a comment system could eventually be a beneficial addition to the site. Another way of making the homepage more interesting, as well as showing users the latest and most popular guides, could be to show the champions that have the most guides written for them which would suggest that these champions are popular and ‘meta’, which could help players in deciding who to play as well as what to build. To expand on the point raised during user testing, the website could improve in helping new players learn the game quickly so that they are equipped to write guides themselves sooner. An idea of this could be to make an information page in which players can read in-depth, factual information about the champions and the items. Another, simpler, way of doing this could also be to have a brief item description on an image hover as well as just their names. This could help users quickly learn or remember what the different items actually do, and therefore gain a deeper understanding of why certain items are being recommended in certain situations rather than relying on the descriptions provided by other users. Lastly, the final area in which the site could be improved is styling. Due to relative inexperience and ‘fiddly’ nature of CSS styling, it was difficult to get the desired layout as planned in the designs. This led to components being more spread out than desired, resulting in less being visible on the page at one time, which could mean the user needs to scroll more than necessary. Furthermore, whilst some consideration was made into making the site adaptable to different screen sizes such as using percentage measurements rather than pixels, no media queries were implemented which give set styles based on a specific screen size range. Therefore, media queries would be a priority in further development to make the site optimal on all devices and screen resolutions, especially since some user may prefer to access the site on a mobile device as the game may be on their computer monitor at the same time.

## Evaluation & Conclusion

In order to conclude this project, an evaluation of its success must first be carried out. The table below shows all original requirements and a discussion on how each of them has been met. This can help to evaluate the overall success of the project, based on how well the implemented system met requirements.

Requirement	Successfully met?	Discussion
The system must represent the League of Legends aesthetic by using the same colour palette and other design features.	Yes	The themes and colours of the game were carefully analysed during the design phase, and the exact colour codes and fonts were then imported and used on the website. Furthermore, user feedback revealed that users found the system similar to that already in League of Legends and that the themes are consistent and provided familiarity to the game itself.
The system must avoid difficult navigation or long load times by having a minimal number of pages.	Yes	This requirement was kept in mind at all times during development and therefore a minimal number of pages were implemented. Just 4 pages were implemented in the navigation bar as these were essential to be easily accessible from all pages. Furthermore, instead of creating a separate page for each individual guide, query strings were used to change the content of a page accordingly, speeding up response times.
The system must be usable on various devices with different screen resolutions.	Somewhat	As discussed in the previous section, some measures were taken to make the site adaptable to different screen resolutions, namely, using percentages as measurements rather than a specified number of pixels. However, this could be far improved and developed further through the use of media queries to make the site optimal on all screen resolutions, particularly mobile devices.

Users must be able to create an account and log in to access the full capabilities of the system.	Yes	A login system was successfully fully implemented with appropriate security measures and error handling. The content of some pages, the profile and write guide pages, are hidden from users unless they create an account and log in.
Logged-in users must be able to create and submit guides that will be automatically displayed to all users.	Yes	The page in which users can write and submit guides is available to users that are logged in. Once a guide is finished and submitted, it is automatically displayed and relevantly organised on the homepage, browse guides page, and the author's profile page.
The guides must follow a set layout and require mandatory information such as the name and role of the Champion, at least one build recommended for a defined situation, the patch version of the game it is written for, and the rank, level, and region that the guide is tailored to.	Yes	The form in which users write and submit guides has mandatory fields that the writer must fill in before it can be submitted. A check that there are no empty fields is performed before the data is sent to the database and therefore displayed to other users. The mandatory fields are: the guide title, the name of the champion, the role, the patch version, and the core build. The writer is not required to enter information regarding their rank, level and region as these are automatically extracted from their user/profile data.
Information must be easy to access and to visualise so the system must use visual cues and avoid large amounts of text where possible.	Yes	Text is used minimally throughout the site and many visual cues are used that players will be familiar of such as champion images, role icons and item images. These visual cues are consistent with those used in the game. This makes the site accessible to those who may struggle to read.
The system must allow all users to easily find guides they are looking for and refine their searches by Champion, Role, Level, Rank, Region, Patch, and Rating	Somewhat	A search bar was implemented in the 'browse guides' page in which users can search for the names of champions, guides, roles, and even regions and ranks etc. However, it would have been beneficial to implement a filter to allow users to select fields such as ranks, regions, and levels from a drop-down menu.
The system must display useful categories on the home screen such as the most recent guides or most popular Champions	Somewhat	The home screen successfully displays the most recently submitted guides to all users. However, more categories such as the most popular champions or highest rated guides to provide the user with more helpful suggestions and therefore improve their user experience.
Logged-in users must be able to access and customise their profile	Somewhat	Logged in users can successfully update their details including their email, username, and password as well as stats relating to the game that will continuously change as the user plays such as their rank and level. This includes appropriate security measures and error handling. However, the function to allow users to change their profile picture was not fully implemented.

As shown in the table, all requirements were met to at least some extent. No requirements failed to be met, six out of ten were successfully fully met and four out of ten were somewhat met. This indicates that the project was a relative success. The implemented system functions as it was required, and users are happy with the results as shown by the high usability rating. Of course, several things could be done to improve the system, involving both fully meeting all stated requirements as well as adding additional requirements to meet if development was to be taken further. Many areas of potential improvement have been covered in the Operation & Maintenance section and the table above. The process of carrying out this project will be critically reflected upon in the following section which may reveal opportunities for meeting requirements more successfully if certain stages of the process had been done differently. Whilst it is evident that the implemented system sufficiently meets requirements, as this was a HCI project, it is important to reflect upon how well the project followed common practices in the field. An example of one of these practices key to user-centred interaction design is Nielsen's 10 Usability Heuristics (Nielsen, 1994). The table below explores these heuristics and to what extent they were implemented in this project.

Usability Heuristic	Successfully Implemented?	Discussion
Visibility of System Status: The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.	Somewhat	Error messages are presented to the users if a problem occurs due to user error such as not filling in a required field or entering the incorrect password. However, if there were to be a system failure such as the database being inaccessible or a required service being down, the user would not currently be informed of this.
Match between system and the real world: The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.	Yes	In the case of this project, the 'real world' largely refers to League of Legends as this was used to base the interface design upon. The implemented system is highly consistent with the game and provides players with familiar visual cues such as recognisable images. Furthermore, the system matches real world conventions by using simple language in order to avoid users having difficulty understanding or having to interpret meaning. As well as this, cultural conventions were avoided. An example of this could be that some cultures interpret the colour red as being bad, but others interpret it as representing joy or luck.
User control and freedom: Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.	Mostly	The functionality of an emergency exit is not applicable to this system as it is a website, so this is dealt with by the browser, it is not a standalone application. However, the website provides easy navigation so users should not feel 'stuck' on a particular page, text inputs and selections can also be changed easily so the user does not have to commit in case they don't get it right first time. However, this could be improved by more easily allowing users to change the build they are recommending whilst writing a guide in case they accidentally drag and drop the wrong item.
Consistency and Standards: Users should not have to wonder whether different words, situations, or actions mean the same thing.	Yes	Similar to what was explained for the second heuristic, the site uses simple, unambiguous wording and avoids cultural conventions. The same wording and images used in the game have been used in the site, so users should feel a sense of familiarity.
Error Prevention: Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.	Mostly	If invalid data is entered, it is not sent to the database. This prevents users having to undo and redo what they have done by themselves. The users are informed of the reasons why the data has not been accepted such as empty fields or an incorrect password. As well as this, drop-down menus were used where possible to avoid misspelled data being saved to the database. Some Champions have complex names such as "Vel'Koz" so it is important that this name is available for the user to select from rather than having different variants stored in the database and therefore not being categorised as the same champion. However, if much more advanced development was possible, an intuitive responsive system that indicates to users if something seems to be erroneous before they attempt to submit it would be ideal.
Recognition rather than Recall: Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.	Yes	This principle was fundamental throughout development and design with recognition over recall prioritised. Many visual cues are used as they are in the game, so users do not need to remember the complex names of items etc. However, if a user struggles to recognise an image or icon, hovering over them will present the user with the name in case they prefer to recall it. Furthermore, the guide form is in a single location on one page rather than split over multiple subpages so users do not need to remember what they have already suggested as they continue with their guides, as it is all visible at the same time.
Flexibility and Efficiency of Use: Accelerators — unseen by the novice user — may often speed	Mostly	Whilst the functionality of the website remains the same for users with any level of experience, more advanced features are locked unless the users sign up and log in. This discourages

up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.		inexperienced users from attempting to submit advice to others and avoids overwhelming them. Viewing the advice written by others can be seen by everyone without having to sign up or log in.
Aesthetic and minimalist design: Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.	Yes	League of Legends uses an aesthetic and minimalistic design and this style was applied to the website too. Clutter and large sections of text are avoided, and every element serves a clear purpose.
Help users recognize, diagnose, and recover from errors: Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.	Yes	If the user attempts to submit invalid data, they are presented with an error message. These error messages are written in plain and simple English so all users can understand the problem without requiring technical knowledge. These error messages also suggest a solution such as asking the user to fill in all fields.
Help and documentation: Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.	No (N/A)	This heuristic does not seem widely applicable to web-based development. Whilst the introduction of an entirely new application may need explanation, it is very rare for modern websites to offer tutorials as they follow conventions and are often self-explanatory. The site was designed so that users would understand how to use the website immediately, without explanation or a tutorial. Whilst a non-player may have difficulty understanding jargon or recognising images, the product was developed for use by players of League of Legends only. Furthermore, all 3 respondents of the SUS survey said they strongly disagree with the statement that they 'would need the help of a technical person to be able to use this system', suggesting that the site was self-explanatory and they would be confident using the system without technical support.

Based on the table above, it is evident that this project was successful in following all applicable usability heuristics to some degree. Whilst some were not fully followed, the reasoning for this had already been brought to light in previous sections therefore no additional concerns or failings were highlighted. By evaluating the project in this way, it is clear that this project was a relative success in the field of HCI as many relevant teachings were considered and practiced during this process.

Overall, this project was largely successful in many ways including meeting requirements, following HCI principles and creating a product usable and usable to its end users. Furthermore, the chosen methodologies throughout were appropriate though not necessarily ideal. As the waterfall method seems to be being slowly phased out, given normal circumstances and more access to resources and willing participants, an agile approach could be optimal. Furthermore, the development and testing phases were accidentally and naturally carried out iteratively, revealing why it is becoming the industry standard to develop software in this way. Nevertheless, this project resulted in a fully functioning website that functions as intended, whilst it may have felt more complete if a live web link was obtained in time rather than relying on local hosting alone, the aim and requirements of this project have been met.

## Reflective Analysis

Whilst all stages of this project have been critically analysed throughout, I will now take this opportunity to reflect on this process from a personal perspective, as the sole developer on the project. As expected, this project has been a challenge however, as the result of this project is a functioning website that meets the aim of the project, it is clear that it has been successful overall. Furthermore, some shortcomings and ideas for potential future development have already been identified in previous sections. However, this section will explore my opinions on what went well during this process and what could be improved.

Key strengths of mine during this process proved to be determination and perseverance. As an unconfident programmer, coding an entire system from scratch for this solo project felt like an overwhelming task at times. However, I feel like I learned a lot through experience throughout this process and I am now a stronger programmer, better equipped to hopefully begin a career in web development. One success of this project was the very choice of project to begin with. Choosing programming languages that I could see myself using in the future and choosing to make a resource based on a game I am interested in and passionate about helped to motivate me throughout this process. This project was not only completed in order to complete my final year at university and successfully graduate, but to also serve as a useful tool in my future to prove my capabilities to potential employers. Successful completion of this project was made far more difficult due to the coronavirus pandemic as programming support and resources were made more difficult to access. Nevertheless, the quality of this project was not allowed to be significantly reduced due to this crisis. Hard work and commitment ensured that the project was completed successfully despite the unusual situation and remote access to help and resources were taken advantage of where possible. Lastly, I believe that the design process of this project was carried out very successfully. User-centred design is important in the field of HCI and therefore this was prioritised in this project. User involvement was conducted appropriately and successfully, and designs of increasing complexity were created before beginning development to ensure that all aspects of functionality were carefully planned. This led to a product that met its requirements, that provides a sense of familiarity to players of the game, and that implements key features with the convenience and desires of users in mind.

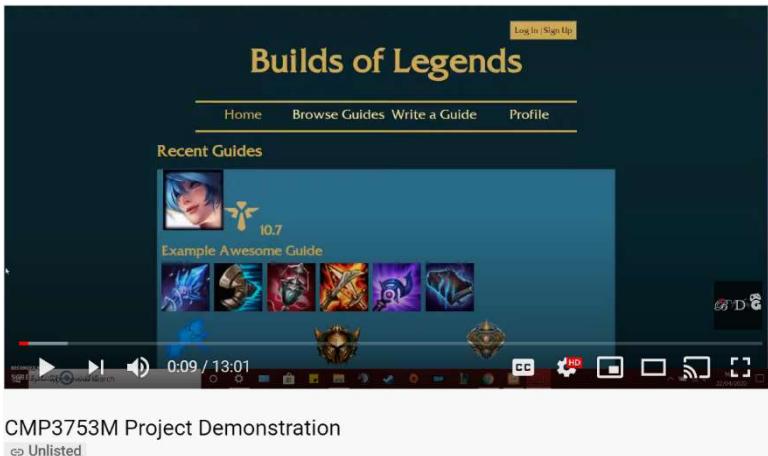
However, the design stage was both a success and the cause of some failings in this project. Whilst every design produced during this process proved to be extremely valuable, too much time was allotted to this stage of the SLDC and therefore insufficient time was dedicated to the development stage. Some ideas during the early design stages had to be phased out due to the lack of time to successfully implement them, an example of one of these features was a rating/liking system. Not only this, but one feature, the feature of allowing users to change their profile picture, was not able to be completed in time given its relatively low priority. For this reason, it is clear that time management was somewhat unsuccessful in this project. Therefore, if I were to do a project like this again, I would spend less time on the planning and design and more time on development. As well as this, more effective use of Project Management Tools may also help with time management. However, another way of preventing the unnecessary time pressures on the development stage could have been by choosing a different methodology, such as an agile process. This would mean that no features that could not be implemented would be designed and no features that had been designed would not be implemented. Nevertheless, given the circumstances of this project, I strongly believe that the Waterfall method was the most appropriate methodology in this case. If I were to carry out a similar project in the future and had more access to resources and user involvement, and perhaps worked as part of a team, I would consider an agile approach. Lastly, if I were to do one more thing differently, it would be that I would have more engagement with my Supervisor. It is very rare in industry for

programming projects to be carried out completely individually, therefore I think access to help and guidance is beneficial. Without the expertise of a staff members in Web Development and HCI, I would have struggled to complete my artefact proving that support during a programming project is important.

Overall, I am satisfied with my completion of this project and the product that I have developed. Although it wasn't done perfectly, the reasons for this have been carefully reflected upon and therefore I have learned valuable lessons through this process. With a year of hard work coming to an end, I can simply hope that this project can help me to achieve the 1<sup>st</sup> class honours degree I have vigilantly worked towards.

## Demonstration Video

Find the link to the video demonstrating the fully functioning completed website below.



<https://youtu.be/KcqMvXBIDQA>

## References

- Cohn, M. (2004). *User Stories Applied: For Agile Software Development*. Pearson Education Inc, Chapter 1 pg.3-15.
- Coyer, C., 2013. *Poll Results: What Javascript Library Do You Use?*. [online] CSS-Tricks. Available at: <<https://css-tricks.com/poll-results-what-javascript-library-do-you-use/>> [Accessed 14 April 2020].
- Datamation.com. 2016. *8 Major Advantages Of Using Mysql*. [online] Available at: <<https://www.datamation.com/storage/8-major-advantages-of-using-mysql.html>> [Accessed 14 April 2020].
- Devault, G., 2019. *Tame Your Focus Groups And Get Useful Insights*. [online] The Balance Small Business. Available at: <<https://www.thebalancesmb.com/what-is-a-market-research-focus-group-2296907>> [Accessed 21 April 2020].
- Digamber, 2020. *XAMPP Vs WAMP – Best Local Server For Web Development In 2020*. [online] Positronx.io. Available at: <<https://www.positronx.io/xampp-vs-wamp/>> [Accessed 14 April 2020].
- Dooley, J., 2017. *Software Development, Design And Coding*. Apress, Chapter 1 & 2 pg.1-29.
- Elgabry, O. (2016). *UX — A quick glance about The 5 Elements of User Experience (Part 2)*. [online] Medium. Available at: <<https://medium.com/omarelgabrys-blog/ux-a-quick-glance-about-the-5-elements-of-user-experience-part-2-a0da8798cd52>> [Accessed 4 Mar. 2020].

Esports Tales. (2019). *League of Legends Rank Distribution in Solo Queue*. [online] Available at: <<https://www.esportstales.com/league-of-legends/rank-distribution-percentage-of-players-by-tier>> [Accessed 27 Feb. 2020].

Gerber, H., Sweeney, K. and Pasquini, E., 2019. Using API Data to Understand Learning in League of Legends: A Mixed Methods Study. *Educational Media International*, [online] 56(2), pp.93-115. Available at: <<https://doi.org/10.1080/09523987.2019.1614250>> [Accessed 16 April 2020].

Gidalevitz, Y., n.d. *Mood Boards In UX Design: Ignite Passion In Your Users*. [online] Usabilitygeek.com. Available at: <<https://usabilitygeek.com/mood-boards-ux-design/>> [Accessed 9 March 2020].

IONOS Digitalguide. 2019. *Web Programming Languages*. [online] Available at: <<https://www.ionos.com/digitalguide/websites/web-development/web-programming-languages/>> [Accessed 14 April 2020].

Kashyap, S., 2019. Top 11 Benefits Of Gantt Charts In Project Management. [online] Proofhub.com. Available at: <<https://www.proofhub.com/articles/benefits-of-gantt-charts>> [Accessed 13 April 2020].

leagueoflegends.com. n.d. *How To Play - League Of Legends*. [online] Available at: <<https://na.leagueoflegends.com/en-us/how-to-play/>> [Accessed 19 April 2020].

League of Legends Wiki. n.d. *Item*. [online] Available at: <<https://leagueoflegends.fandom.com/wiki/Item>> [Accessed 17 April 2020].

Libguides.newcastle.edu.au. 2019. *Libguides: Research Methods: What Are Research Methods?*. [online] Available at: <<https://libguides.newcastle.edu.au/researchmethods>> [Accessed 21 April 2020].

Medium. 2018. *The 5 Most Popular Programming Languages In 2018*. [online] Available at: <<https://medium.com/packt-hub/the-5-most-popular-programming-languages-in-2018-9b43f3b3fa88>> [Accessed 14 April 2020].

Messner, S., 2019. *League Of Legends Draws 8 Million Concurrent Players, Making It The Most Popular Game On PC*. [online] pcgamer. Available at: <<https://www.pcgamer.com/uk/8-million-people-play-league-of-legends-every-day-making-it-the-most-popular-game-on-pc/>> [Accessed 19 April 2020].

Mohapatra, P. (2010). Software engineering. New Delhi: New Age International, Chapter 3 pg.63-92.

Mora-Cantallops, M. and Sicilia, M., 2018. Exploring player experience in ranked League of Legends. *Behaviour & Information Technology*, [online] 37(12), pp.1224-1236. Available at: <<https://doi.org/10.1080/0144929X.2018.1492631>> [Accessed 15 April 2020].

Nascimento Silva, V. and Chaimowicz, L., 2017. A Tutor Agent for MOBA Games. [online] Available at: <[https://www.researchgate.net/publication/317543849\\_A\\_Tutor\\_Agent\\_for\\_MOBA\\_Games](https://www.researchgate.net/publication/317543849_A_Tutor_Agent_for_MOBA_Games)> [Accessed 15 April 2020].

Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. ACM CHI'94: Conference on Human Factors in Computing Systems. Pg.152-158

Ozok, A., 2009. Survey design and implementation in HCI. *Human-Computer Interaction: Development Process*, pg.253-271.

PHP.net. n.d. *PHP: What Is PHP? - Manual*. [online] Available at: <<https://www.php.net/manual/en/intro-whatis.php>> [Accessed 14 April 2020].

Powell-Morse, A., 2016. *Waterfall Model: What Is It And When Should You Use It?*. [online] Airbrake Blog. Available at: <<https://airbrake.io/blog/sdlc/waterfall-model>> [Accessed 13 April 2020].

Rouke, P., 2017. *What Is User Centred Design? A Guide To The Processes Involved*. [online] PRWD. Available at: <<https://www.prwd.co.uk/blog/user-centred-design-process-overview/>> [Accessed 21 April 2020].

Software Testing Fundamentals. n.d. *White Box Testing*. [online] Available at: <<http://softwaretestingfundamentals.com/white-box-testing/>> [Accessed 20 April 2020].

Stackify. 2017. *What Is SDLC? Understand The Software Development Life Cycle*. [online] Available at: <<https://stackify.com/what-is-sdlc>> [Accessed 18 March 2020].

Synotive. 2017. *10 Questions To Ask When Developing Software*. [online] Available at: <<https://www.synotive.com/blog/software-development-client-questionnaire>> [Accessed 13 April 2020].

Usability.gov. n.d. *Usability Testing*. [online] Available at: <<https://www.usability.gov/how-to-and-tools/methods/usability-testing.html>> [Accessed 20 April 2020].

WADIC. n.d. *6 Advantages And Disadvantages Of The Waterfall Model*. [online] Available at: <<https://wadic.net/waterfall-model-advantages-disadvantages>> [Accessed 13 April 2020].

W3schools.com. n.d. *What Is Javascript*. [online] Available at: <[https://www.w3schools.com/whatis/whatis\\_js.asp](https://www.w3schools.com/whatis/whatis_js.asp)> [Accessed 14 April 2020].

## Appendices

### (1) Requirements Elicitation Survey

#### (a) Survey Post in Facebook Group

The screenshot shows a Facebook group post from the 'League of Legends EUW' group. The post is titled 'Dissertation Survey: League of Legends'. It includes a survey form with questions about thoughts and experience of League of Legends, region, and age group. It also links to a Google Doc for the survey. The post has 24 likes and 42 comments. The group page shows 14,605 members.

Figure 18

## (b) Survey Questions and Results

How often do you play League?

200 responses

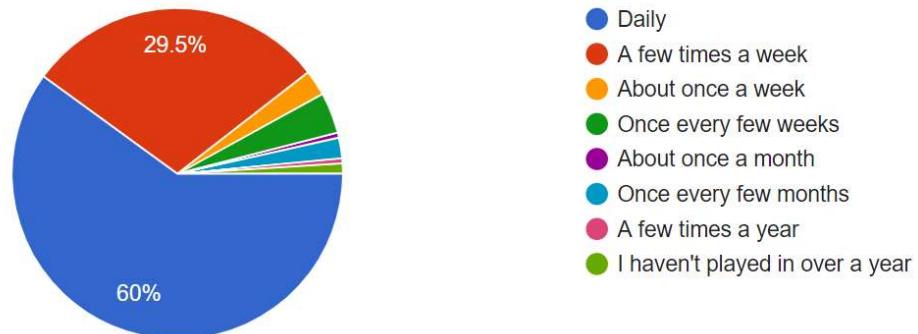


Figure 19

How/why do you play League?

200 responses



Figure 20

What is the Highest Rank you've ever Achieved?

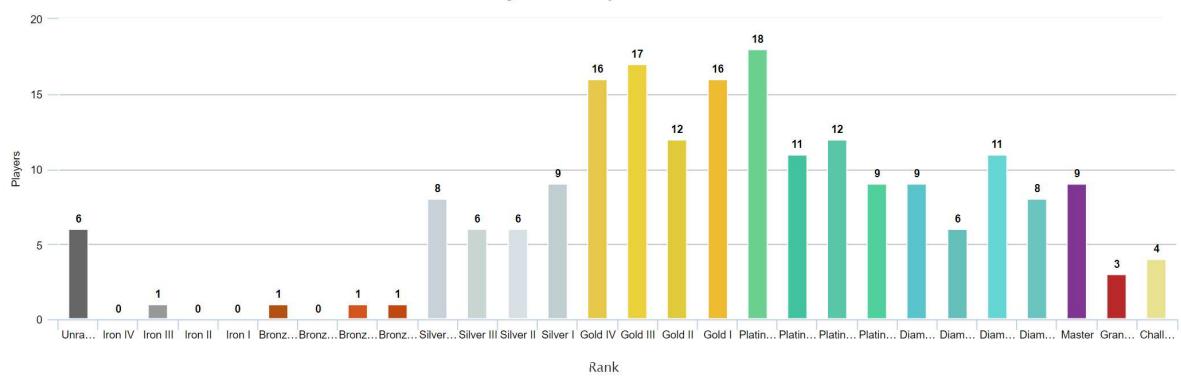


Figure 21

Did you achieve the Rank you wanted or expected to?  
196 responses

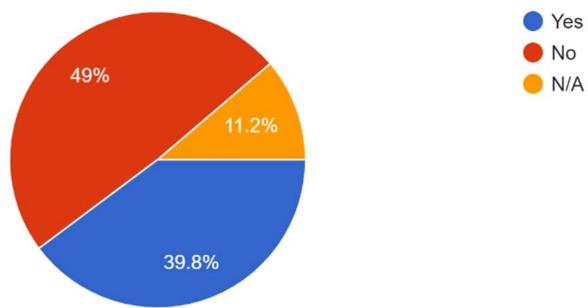


Figure 22

How would you rate the overall difficulty of the game?  
200 responses

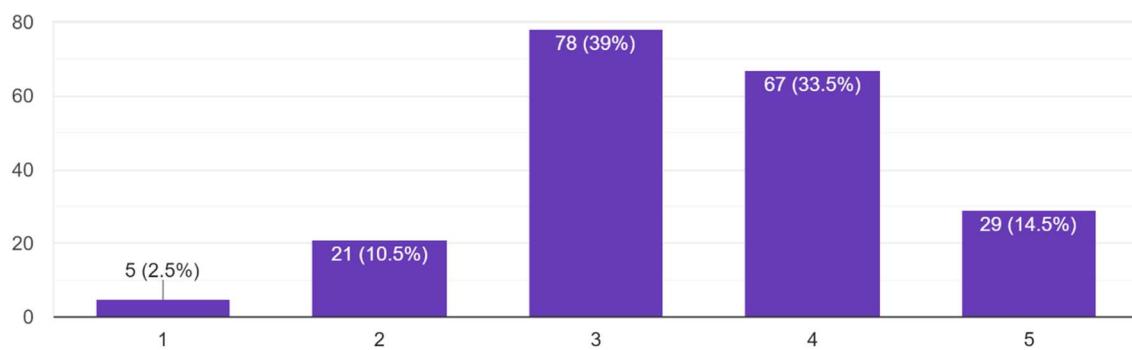


Figure 23

How would you rate the difficulty of the ranking system? (i.e. how fast it is to rank up)  
197 responses

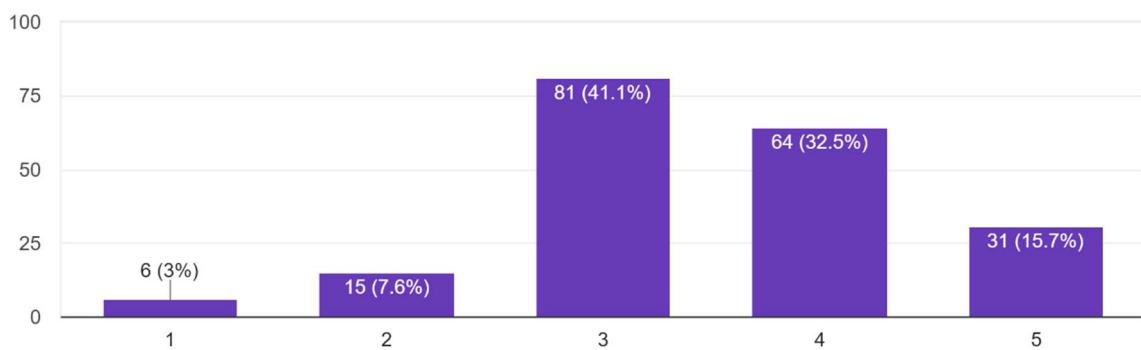


Figure 24

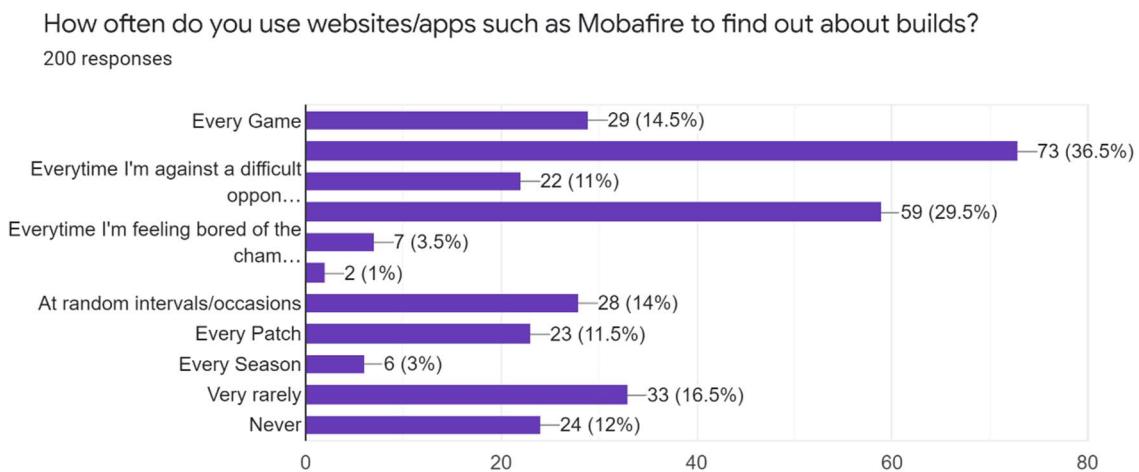


Figure 25

Are you intending/hoping to improve your skill and/or rank in Season 10?  
200 responses

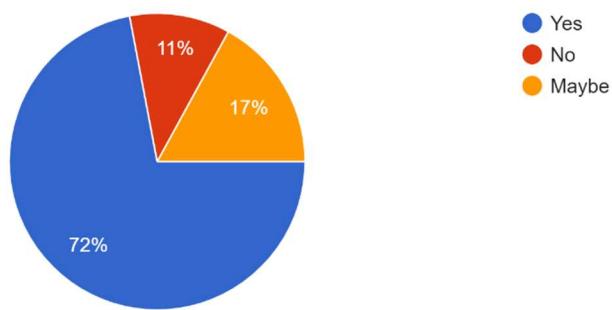


Figure 26

Do you think you would be interested in a new website or app that would combine some of the most important features of various existing sites/apps into one handy place?  
200 responses

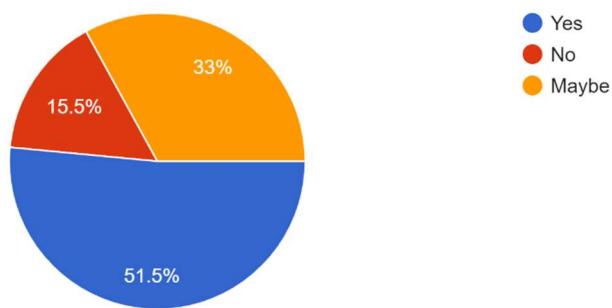


Figure 27

### (c) Summary of Survey Findings

#### **Summary of Results from Surveying 200 League of Legends Players**

- The majority (88%) play in EUW
- The majority (54.5%) are aged 18-22
- Everyone is under the age of 40
- The majority (60%) play daily
- 34.5% play just for fun, 25% play seriously/competitively, 19% play casually
- Support (26%) and Bot/ADC (25.5%) are the most mained roles
- Jungle (11%) is the least mained role
- Marksman (27.5%) is the most popular playstyle
- Tank (5.5%) is the least popular role
- More players are between levels 101-200 than any other level ranges
- Only 1.5% are under level 30, only 1% are over level 500
- More people finished in Gold IV (14.5%) in Season 9 than any other rank
- More people have peaked at Platinum IV (9%) than any other rank
- 8.5% were unranked at the end of Season 9 and 3% have never been ranked before
- Almost half (49%) of people said they did not achieve the rank they wanted/expected.  
Excluding the 11.2% N/A, 55% were unsatisfied
- More people rated the overall difficulty of the game as neither easy or difficult (39%) than any other category of difficulty
- Almost half, 48%, of people rated the overall difficulty of the game as either difficult or very difficult
- More people rated the overall difficulty of the ranking system as neither easy or difficult (41.1%) than any other category of difficulty
- Almost half, 48.2%, of people rated the overall difficulty of the ranking system as either difficult or very difficult
- Most popular responses to how players think is the main way they can improve their rank/winrate: being good at one role (15%), communicating with the team (14.5%), playing with a partner (13%), solo carrying (11.5%), focusing on objectives (10.5%)
- More people (33%) said that uncooperating/unskilled teammates was their main cause of losses than any other reason. The next highest proportion of responses was just 8%
- 8% of people blame poor game knowledge for their main cause of loses and 8% said it was due to poor objective priority/focus
- 72% of players intend/hope to improve their skill and/or rank in Season 10. 89% answered yes OR maybe
- The skills that most people want to work on are: map awareness (40%), objective control (35.5%), learning when to freeze and when to push (35%), learning more champions (30%), improving CS (27.5%), warding (24%), dying less (22.5%), mastering a role (21.5%), adapting builds appropriately to various situations (21.5%), mastering the kits of a small champion pool (20%)
- Only 3.5% of people said they do not want to work on any skills
- 75.5% watch eSports or competitive play
- The most popular use cases for websites/apps relating to League are: every time they play a champion they haven't played in a while (36.5%) and every time they play a new champion (29.5%)

---

- The most important feature of sites/apps relating to League is easy and quick access to the desired information (61%)
- Other important features: being able to see counterpicks (25.5%), having a variety of build options for each champion (25.5%), being able to see how to adapt your build to various scenarios (22%), having guides updated to each patch (21%)
- 51.5% would be interested in a new website. 84.5% answered yes OR maybe
- Excluding 21.5% of N/A inexperienced players who answered, 63% of experienced players would be willing to offer their guidance to newer players. 92% answered yes OR maybe
- 55.5% said they want/wish there was more guidance available to help learn the game better.
- Only 11% said that the game alone offers enough guidance to be able to learn
- 75% are willing to answer another survey to give feedback on the new website. 92.5% said yes OR maybe

(d) User Stories

**Using 20 Randomly Selected Survey Responses to Generate User Stories**

As a serious/competitive Jungle player, I want easy and quick access to adaptive build guides updated to each patch so I can learn how to itemize effectively in varying situations.

As a player in low Silver, I want to learn more roles, champions, and builds so that I can improve my rank so I can get away from uncooperating and unskilled teammates.

As an experienced player at level 201-300 and Diamond rank, I would be willing to offer my advice to newer players because I understand that game knowledge is very hard to learn.

As a casual mid-laner, I want to be able to see a variety of build options for each champion because I think not being able to itemize effectively/correctly is my main cause of losses.

As a regular player, I want easy and quick access to information that is regularly updated so I can get help when I am against a difficult opponent.

As a player who finished an entire league below their highest record, I want to learn how to adapt builds appropriately so I can solo carry more effectively

As a streamer, I want to learn more roles and champions so I can make new and exciting content

As a daily player, I want help with mastering a role because I think my main cause of losses is that I am ok at all roles but talented at none.

As a support main, I want to learn advice I can give to all roles on my team to keep team morale positive.

As a player with somewhat poor game knowledge, I want to check a help site every game to see counterpicks so that I can enjoy the game more.

As a competitive League of Legends player with poor knowledge of how to build correctly, I would like a service that recommends what to build on my character based on what the enemy team is comprised of. This is so I can spend less time studying the boring build side of the game and focus on improving my gameplay.

As someone that only plays because my friends do, I want guides that are updated to each patch so that I can keep up with the game easily without it being a passion of mine.

As someone that takes the game seriously, I want a site that can help me improve as many skills as possible all in one place so that I can take responsibility and improve my own mistakes.

As someone that has been playing for a very long time, I want guides that can help me every time I play a new champion so that I don't under-perform in my division and I maintain my passion.

---

As a player on my 2nd account after failing to learn enough the first time I levelled to 30, I still get stuck in low-gold, so I want a resource that will help me learn more in depth knowledge so that I can work on skills that my opponents might also lack.

As a support main, I lack knowledge about ADC builds so I want to learn more about builds and items that I do not use in my limited playstyles so that I can learn how to complement and counter more effectively.

As a player with only one main champion, I want a resource that I can check every time I am against a difficult opponent that has in-depth explanations so that I can apply what I learn to future games.

As a player that usually plays the same champion, I want a guide that I can check every time I play a new champion that can help me learn their kit and build so that I can try different things to avoid boredom.

As a player who has reached Grandmaster, I still feel that there are several skills that a resource could help me with such as mastering roles and adapting builds so that I can train to become a professional.

As a someone who has not played in a while, I want to be able to readily see what the meta item builds are for champions I used to play, so I can quickly get up to speed and start performing as well as I used to.

## (2) Designs

### (a) Surface Plane: Designs made in Adobe XD

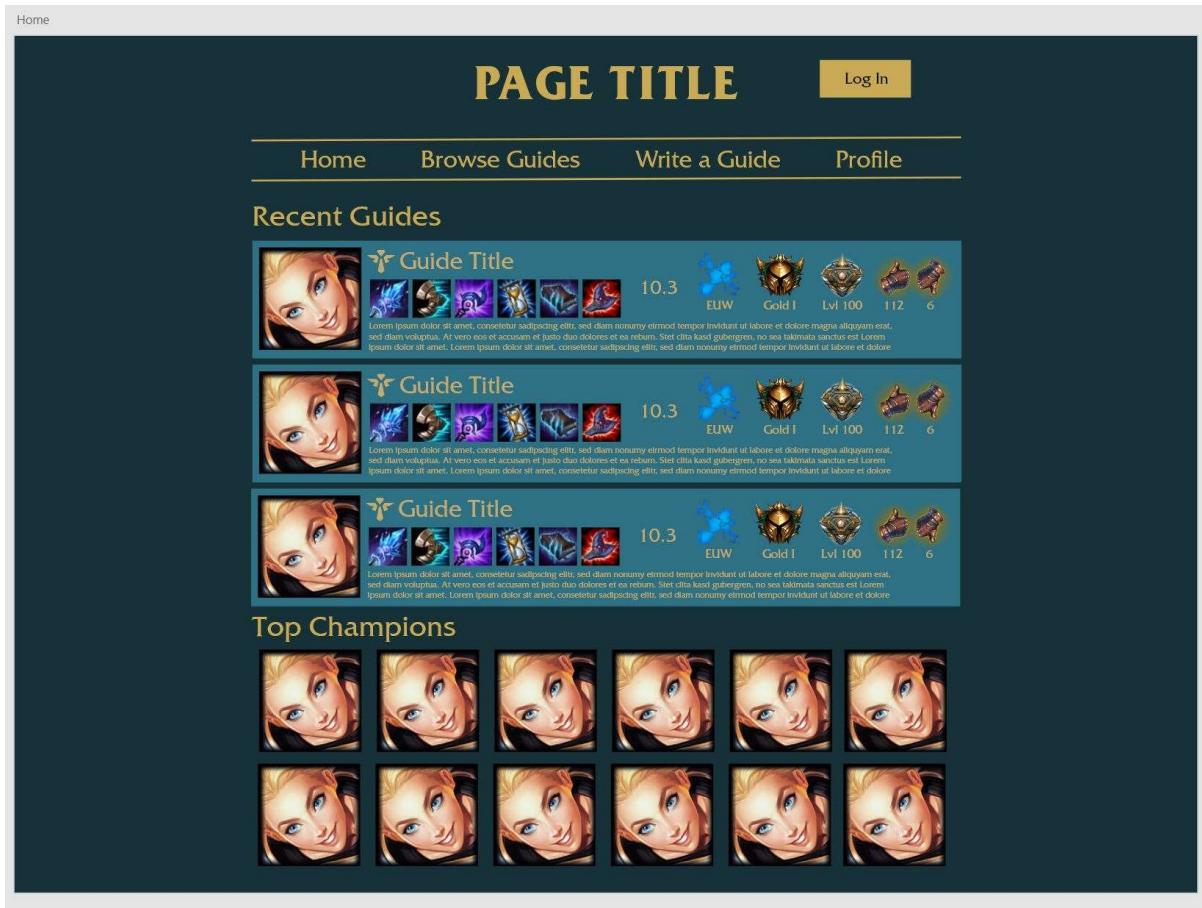


Figure 28: Homepage

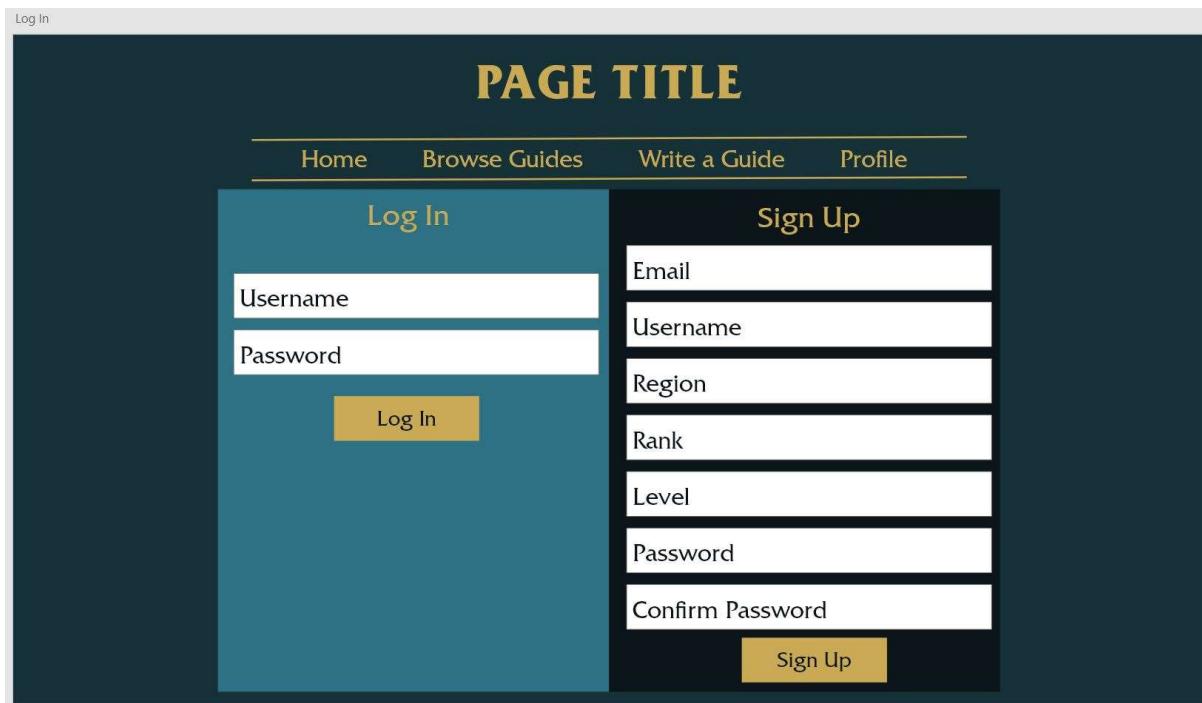


Figure 29: Login

Write a Guide

# PAGE TITLE

[Log In](#)

---

[Home](#)   [Browse Guides](#)   [Write a Guide](#)   [Profile](#)

**Guide Title**

Guide Title...

**Champion**

Champion...

**Patch**

Patch...

**Role**

**Core Build**

**Description**

Guide Description...

[Add a Build](#)

[Submit Guide](#)

Figure 30: Write a Guide

Browse Guides

# PAGE TITLE

[Log In](#)

---

[Home](#)   [Browse Guides](#)   [Write a Guide](#)   [Profile](#)

Search Results for 'search'...

**Guide Title**  
10.3 EUW Gold I Lvl 100 112 6  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy elitmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

**Guide Title**  
10.3 EUW Gold I Lvl 100 112 6  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy elitmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

**Guide Title**  
10.3 EUW Gold I Lvl 100 112 6  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy elitmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Figure 31: Browse Guides

View a Guide

# PAGE TITLE

[Log In](#)

---

Home    Browse Guides    Write a Guide    Profile



**Guide Title**  
**Lux**  
 by Author
 

**10.3**

 EUW
 

 Gold I
 

 Lvl 100

**Core Build**



Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum

**Situational Build**



Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum

Figure 32: Read a Guide

Profile

# PAGE TITLE

[Log In](#)

---

Home    Browse Guides    Write a Guide    Profile



**Username**

 EUW
 

 Gold I
 

 Lvl 100
 

**12**

**Guides Written**

[Edit Profile](#)

**My Guides**



**Guide Title**



**10.3**

 EUW
 

 Gold I
 

 Lvl 100
 

 112
 

 6



**Guide Title**



**10.3**

 EUW
 

 Gold I
 

 Lvl 100
 

 112
 

 6



**Guide Title**



**10.3**

 EUW
 

 Gold I
 

 Lvl 100
 

 112
 

 6

Figure 33: User Profile

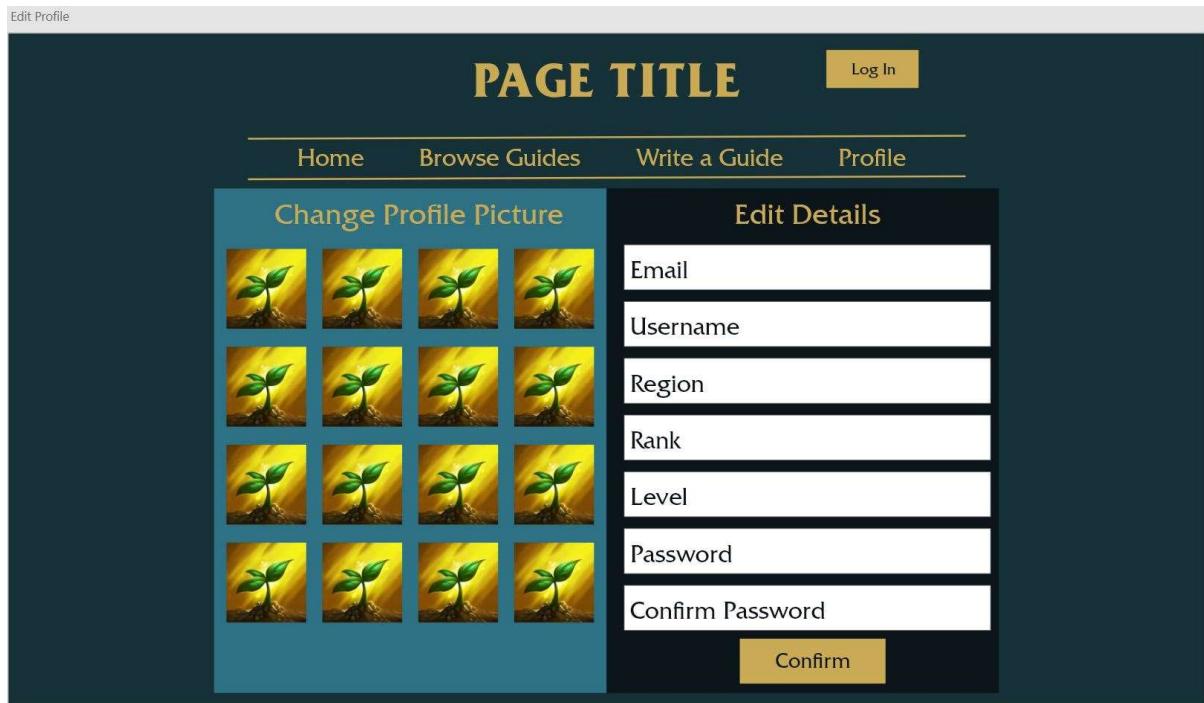


Figure 34: Edit Profile

### (3) Implementation Code

#### (a) Basic layout and themes

```
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="UTF-8">
        <link rel="stylesheet" type="text/css" href="CSS/styles.css">
        <script src="JS/main.js"></script>
        <script src="https://kit.fontawesome.com/a118157f67.js" crossorigin="anonymous"></script>
        <title>LoL</title>
    </head>

    <body>
        <header>
            <div id="loginBar">
            </div>

            <h1 id="Title">Page Title</h1>

            <div id="navigationBar">
                <ul id="navigation">
                    <li class="navigationList"><a id="selected" class="nav" href="Home.php">Home</a></li>
                    <li class="navigationList"><a class="nav" href="Guides.php">Browse Guides</a></li>
                    <li class="navigationList"><a class="nav" href="WriteGuide.php">Write a Guide</a></li>
                    <li class="navigationList"><a class="nav" href="Profile.php">Profile</a></li>
                </ul>
            </div>
        </header>

        <footer class="footer">
        </footer>
    </body>
</html>
```

Figure 35: HTML used for Header

```

@font-face
{
    font-family: 'FrizQuadrataRegular';
    src: url("../Assets/Font/FrizQuadrataRegular.eot");
    src: local('FrizQuadrataRegular'), url("../Assets/Font/FrizQuadrataRegular.woff") format('woff'), url("../Assets/Font/FrizQuadrataRegular.ttf") format('truetype');
}
/*COLOURS: blue1:#0C151A,blue2:#0A272B,blue3:#163138,blue4:#2E7184,gold1:#79613B,gold2:#CAAA55,gold3:#A6925F,gold4:#EDDF9E*/
body
{
    background-color: #0A272B;
    text-align: center;
    font-family: 'FrizQuadrataRegular';
    margin: 0;
}

```

Figure 36: CSS Used to Import and Use Font

```

#Title
{
    font-weight: bold;
    font-size: 75px;
    color: #CAAA55;
}
#navigationBar
{
    border-top: thick solid #CAAA55;
    border-bottom: thick solid #CAAA55;
    display: table;
    text-align: center;
    align-content: center;
    font-size: 30px;
    margin: auto;
    width: 50%
}
#navigation
{
    width: 100%;
    text-align: center;
    list-style-type: none;
    margin: 0;
    padding: 0;
    display: inline-block;
}
.navigationList
{
    width: 25%;
    float: left;
    text-align: center;
    padding-top: 10px;
    padding-bottom: 10px;
}
.nav
{
    color: #EDDF9E;
    text-decoration: none;
}
.nav:hover
{
    color: #2E7184;
    text-decoration: none;
}
#selected
{
    color: #CAAA55;
    text-decoration: none;
}

```

Figure 37: CSS used to style Title and Navigation Bar

## (b) Signup/Login System

```
<div id="signin-signup">
    <div id="signin">
        <h2 class="signin">Existing User? Sign in</h2>
        <form action="includes/signin.php" method="post">
            <div class="form-field">
                <input class="field" type="text" name="Username" placeholder="Username">
            </div>
            <div class="form-field">
                <input class="field" type="password" name="pwd" placeholder="Password">
            </div>
            <div class="form-field">
                <button class="button" type="submit" name="login-sumbit">Login</button>
            </div>
        </form>
        <div class="form-field">
```

Figure 38: HTML for Login Form

```
<div id="signup">
    <h2 class="signin">New User? Sign Up</h2>
    <form action="includes/signup.php" method="post">
        <div class="form-field">
            <input class="field" type="text" name="email" placeholder="Email">
        </div>
        <div class="form-field">
            <input class="field" type="text" name="Username" placeholder="Username">
        </div>
        <div class="form-field">
            <select class="field" name="Region">
                <option value="" disabled selected>Region</option>
                <option value="BR">BR</option>
                <option value="CN">CN</option>
                <option value="EUNE">EUNE</option>
                <option value="EUW">EUW</option>
                <option value="JP">JP</option>
                <option value="KR">KR</option>
                <option value="LAN">LAN</option>
                <option value="LAS">LAS</option>
                <option value="NA">NA</option>
                <option value="OCE">OCE</option>
                <option value="PH">PH</option>
                <option value="RU">RU</option>
                <option value="SG">SG</option>
                <option value="TH">TH</option>
                <option value="TR">TR</option>
                <option value="TW">TW</option>
                <option value="VN">VN</option>
            </select>
        </div>
        <div class="form-field">
            <select class="field" name="Rank">
                <option value="" disabled selected>Rank</option>
                <option value="Unranked">Unranked</option>
                <option value="Iron">Iron</option>
                <option value="Bronze">Bronze</option>
                <option value="Silver">Silver</option>
                <option value="Gold">Gold</option>
                <option value="Platinum">Platinum</option>
                <option value="Diamond">Diamond</option>
                <option value="Master">Master</option>
                <option value="Grandmaster">Grandmaster</option>
                <option value="Challenger">Challenger</option>
            </select>
        </div>
        <div class="form-field">
            <select class="field" name="Lvl">
                <option value="" disabled selected>Level</option>
                <option value="Under 30">Under 30</option>
                <option value="30-99">30-99</option>
                <option value="100-199">100-199</option>
                <option value="200-299">200-299</option>
                <option value="300-399">300-399</option>
                <option value="400-499">400-499</option>
                <option value="500+>">500+</option>
            </select>
        </div>
        <div class="form-field">
            <input class="field" type="password" name="pwd" placeholder="Password">
        </div>
        <div class="form-field">
            <input class="field" type="password" name="pwd-confirm" placeholder="Confirm Password">
        </div>
        <div class="form-field">
            <button class="button" type="submit" name="signup-sumbit">Sign Up</button>
        </div>
    </form>
```

Figure 39: HTML for Signup Form

```
CREATE TABLE Users (
    ID int NOT NULL PRIMARY KEY AUTO_INCREMENT,
    Username varchar(30) NOT NULL,
    Email varchar(30) NOT NULL UNIQUE,
    Region varchar(4),
    Division varchar(12),
    Lvl varchar(9),
    ProfilePic varchar(255) DEFAULT 'Lil_Sprout',
    Pwd longtext NOT NULL
);
```

Figure 40: SQL code to create a database table to store user information

ID	1	Username	Email	Region	Division	Lvl	ProfilePic	Pwd
1	Test	gabbydg@hotmail.com	EUW	Platinum	100-199	Lil_Sprout	\$2y\$10\$XAW8Sx4mMKbl.rEBJmtayuCYIQoS9Sba0w0fXONbJ64...	
4	Test1	gabby2805@gmail.com	EUW	Gold	100-199	Lil_Sprout	\$2y\$10\$ntA/FUhs4EadlztH71/M./z8nVbcLOg.kAwq5bofVw...	
5	Test2	test@test.com	EUW	Silver	100-199	Lil_Sprout	\$2y\$10\$OEfsUuc7nB2PExFYZ7/oeBiSxYvle6Amof9tOAxPFQ...	
7	PasswordChecker	testtwo@test.com	EUW	Silver	100-199	Lil_Sprout	\$2y\$10\$qC4EURGS4OnsZF0qqX/55OCg.fZZS6PL0d8rQtYVmDP...	

Figure 41: The User Table of the Database in phpMyAdmin

```

if(empty($email) || empty($username) || empty($password) || empty($passwordRepeat))
{
    header("Location: ../login.php?error=emptyfields&username=".$username."&email=".$email."&region=".$region."&rank=".$rank."&lvl=".$lvl);
    exit();
}
else if(!filter_var($email, FILTER_VALIDATE_EMAIL))
{
    header("Location: ../login.php?error=invalidemail&username=".$username."&region=".$region."&rank=".$rank."&lvl=".$lvl);
    exit();
}
else if($region == '' || $rank == '' || $lvl == '')
{
    header("Location: ../login.php?error=unselectedfields&username=".$username."&email=".$email."&region=".$region."&rank=".$rank."&lvl=".$lvl);
    exit();
}
else if($password != $passwordRepeat)
{
    header("Location: ../login.php?error=passwordcheck&username=".$username."&email=".$email."&region=".$region."&rank=".$rank."&lvl=".$lvl);
    exit();
}
else
{
    $sql = "SELECT Username FROM users WHERE Username=? AND Region=?";
    $statement = mysqli_stmt_init($connection);
    if(!mysqli_stmt_prepare($statement, $sql))
    {
        header("Location: ../login.php?error=mysqlerror");
        exit();
    }
    else
    {
        mysqli_stmt_bind_param($statement, "ss", $username, $region);
        mysqli_stmt_execute($statement);
        mysqli_stmt_store_result($statement);
        $resultCheck = mysqli_stmt_num_rows($statement);
        if($resultCheck > 0)
        {
            header("Location: ../login.php?error=usernametaken&email=".$email."&region=".$region."&rank=".$rank."&lvl=".$lvl);
            exit();
        }
    }
}

```

Figure 42: Error handling performed in the signup process

```

else
{
    mysqli_stmt_bind_param($statement, "ss", $username, $region);
    mysqli_stmt_execute($statement);
    mysqli_stmt_store_result($statement);
    $resultCheck = mysqli_stmt_num_rows($statement);
    if($resultCheck > 0)
    {
        header("Location: ../login.php?error=usernametaken&email=".$email."&region=".$region."&rank=".$rank."&lvl=".$lvl);
        exit();
    }
    else
    {
        $sql = "INSERT INTO users (Username, email, Region, Division, Lvl, Pwd) VALUES (?, ?, ?, ?, ?, ?)";
        $statement = mysqli_stmt_init($connection);
        if(!mysqli_stmt_prepare($statement, $sql))
        {
            header("Location: ../login.php?error=mysqlerror");
            exit();
        }
        else
        {
            $hashedPwd = password_hash($password, PASSWORD_DEFAULT);

            mysqli_stmt_bind_param($statement, "ssssss", $username, $email, $region, $rank, $lvl, $hashedPwd);
            mysqli_stmt_execute($statement);
            header("Location: ../login.php?signup=success");
            exit();
        }
    }
}

```

Figure 43: PHP code and SQL statements to send user input to the database

```

<?php
if (isset($_GET["error"]))
{
    if ($_GET["error"] == "emptyfields")
    {
        echo '<p class="errorMsg">Error: You must fill in all fields</p>';
    }
    else if($_GET["error"] == "invalidemail")
    {
        echo '<p class="errorMsg">Error: Email address is not valid</p>';
    }
    else if($_GET["error"] == "uselectedfields")
    {
        echo '<p class="errorMsg">Error: You must select an option from all drop-down menus</p>';
    }
    else if($_GET["error"] == "passwordcheck")
    {
        echo '<p class="errorMsg">Error: Passwords do not match</p>';
    }
    else if($_GET["error"] == "usertaken")
    {
        echo '<p class="errorMsg">Error: That Username is already taken</p>';
    }
}
else if (isset($_GET["signup"]))
{
    if ($_GET["signup"] == "success")
    {
        echo '<p class="errorMsg">Signup Successful!</p>';
    }
}
}

```

Figure 44: PHP code inserted to display error messages to the user

```

$sql = "SELECT * FROM users WHERE Username=?";
$statement = mysqli_stmt_init($connection);
if(!mysqli_stmt_prepare($statement, $sql))
{
    header("Location: ../login.php?error=sqlerror");
    exit();
}
else
{
    mysqli_stmt_bind_param($statement, "s", $username);
    mysqli_stmt_execute($statement);
    $result = mysqli_stmt_get_result($statement);
    if($row = mysqli_fetch_assoc($result))
    {
        $passwordCheck = password_verify($password, $row['Pwd']);
        if($passwordCheck == false)
        {
            header("Location: ../login.php?error=wrongpassword");
            exit();
        }
        else if($passwordCheck == true)
        {
            session_start();
            $_SESSION['UserID'] = $row['ID'];
            $_SESSION['Username'] = $row['Username'];
            $_SESSION['Region'] = $row['Region'];
            $_SESSION['Rank'] = $row['Division'];
            $_SESSION['Level'] = $row['Lvl'];

            header("Location: ../login.php?login=success");
            exit();
        }
        else
        {
            header("Location: ../login.php?error=wrongpassword");
            exit();
        }
    }
    else
    {
        header("Location: ../login.php?error=nouser");
        exit();
    }
}

```

Figure 45: PHP code to check the validity of input login data and start a login session

(c) Write/Submit Guide System

```
$( "img.roleIcon" ).click(function()
{
    console.log( $(this).attr('id') );
    $(".roleIcon").removeClass('selected')
    || $(this).addClass('selected');
    // pass the id to db,
    role = $(this).attr('id');
});
```

Figure 46: jQuery code used to make images selectable



Figure 47: In-Game Item Shop (League of Legends Wiki, n.d.)

```

function adddroppable() {
    $(".draggable").draggable({
        revert: "valid", // when not dropped, the item will revert back to its initial position
        containment: "document",
        helper: "clone",
        cursor: "move"
    });

    $(".droppable").droppable({
        classes: {
            "ui-droppable-active": "ui-state-active",
            "ui-droppable-hover": "ui-state-hover"
        },
        drop: function (event, ui) {
            var pic = ui.draggable.attr("src");
            //console.log( 'dropped',pic,imageList );
            var droppeditem = $(event.target).parent().prop('class')
            if (droppeditem == "core") {

                imageList.push(pic);
                $(this).css('background-image', 'url(' + pic + ')');
                $(this).attr("data-asset", pic);

            }
            if (droppeditem == "extra") {

                extraList.push(pic);
                $(this).css('background-image', 'url(' + pic + ')');
                $(this).attr("data-asset", pic);

            }
        }
    });
}

```

Figure 48: jQuery code used to implement the drag and drop feature

```

$("#addBuildbtn").click(function () {
    var str = `<h4 class="subheading">Situational Build</h4>
      <div id="build" class="extra">
        <div class="form-field">
          <input class="field" type="text" name="title" placeholder="Situation/Adaptation Name"/>
        </div>
        <div data-asset="" class="itemSlot droppable"></div>
        <div class="form-field">
          <textarea class="description" name="desc" placeholder="Explain this build, . . .></textarea>
        </div>
      </div>`;
    $('#Buildcontainer').append(str);
    adddroppable();
});

```

Figure 49: jQuery code used to reveal elements on button click

```

var allextraitemsarray = [];

$("#guide-button").click(function () {
    //RESET ARRAY
    var row = null;
    $('.extra').each(function (i, obj) {
        //GET TITLE
        var title = $(obj).find('input').val();
        var desc = $(obj).find('textarea').val();
        console.log(title)
        //CREATE OBJ
        row = {
            "name": title,
            "items": [],
            "desc": desc
        };
        //LOOP THROUGH CHILDREN
        $(obj).children('.itemSlot').each(function (i, obj2) {
            //GET DATA-ASSET FROM DROP
            //UPDATES DROPPABLE TO $(this).attr("data-asset", pic);
            var pic = $(obj2).data('asset');
            console.log(pic)
            row.items.push(pic);
        });
        //pass to ajax
        allextraitemsarray.push(row);
        console.log(JSON.stringify(allextraitemsarray));
    });
})

```

**Situational Build**

Situation/Adaptation Name

Explain this build...

**Situational Build**

Situation/Adaptation Name

Explain this build...

+ Add a Build Adaptation

Figure 50: Results in website after clicking 'Add a Build Adaptation' twice to reveal additional forms

```
CREATE TABLE Guides (
    ID int NOT NULL PRIMARY KEY AUTO_INCREMENT,
    AuthorID int NOT NULL,
    DateUploaded DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
    GuideTitle varchar(50) NOT NULL,
    Champion varchar(20) NOT NULL,
    Role varchar(7) NOT NULL,
    Patch varchar(5) NOT NULL,
    CoreBuild longtext NOT NULL,
    Description longtext,
    Situations longtext,
    FOREIGN KEY (AuthorID) REFERENCES Users(ID)
);
```

Figure 51: SQL code to create the database table to store guide information

ID	AuthorID	DateUploaded	GuideTitle	Champion	Role	Patch	CoreBuild	Description	Situations
1	1	2020-03-31 16:58:12	Lux Test	Lux	Supp	10.6	[Assets/Images/Items/Ardent_Censer.png,"Assets/I...]	Test Description	[{"name": "Sample Situation", "items": ["Assets/Images/Items/Lux_Echo.png"]}]
2	1	2020-03-31 16:59:19	Fallen Angel	Morgana	Mid	10.6	[Assets/Images/Items/Ludenis_Echo.png,"Assets/I...]	Morg Mid	[{"name": "Situation X", "items": ["Assets/Images/Items/Morgana_Echo.png"]}]
3	1	2020-04-03 15:16:20	What a Guide looks like	Aurelion Sol	Mid	10.7	[Assets/Images/Items/Ardent_Censer.png,"Assets/I...]	Lorem ipsum dolor sit amet, consectetur adipiscing...	[{"name": "Situation One", "items": ["Assets/Images/Items/Aurelion_Sol_Echo.png"]}]

Figure 52: Guides Table in Database in phpMyAdmin

```

var guideTitle = $('#guide-title').val();
var champ = $('#champion').val();
var patch = $('#patch').val();
var desc = $('#core-desc').val();
var build = JSON.stringify(imageList);
var extra = JSON.stringify(allextraitemsarray);

if(guideTitle === "" || champ === "" || role === "" || patch === ""){
    alert("please fill in all fields");
    return
}
console.log("clicked")

//AJAX DATA IS KEY, VALUE PAIRS I.E DATA: {NAME:VALUE}
//IN PHP GET POST VALUE BY $_POST['NAME']

//ADDED DATATYPE JSON AND RETURNED JSON IN PHP
$.ajax({
    url: './Includes/submitGuide.php',
    data: {GuideTitle: guideTitle, Champion: champ, Role: role, Patch: patch, CoreBuild: build, Description: desc, Situations: extra},
    type: 'post',
    async: 'true',
    dataType: 'json',
    beforeSend: function() {
        // This callback function will trigger before data is sent
        console.log("pre-send")
    },
    complete: function(data) {
        // This callback function will trigger on data sent/received complete
        console.log("data sent")
    },
    success: function (result) {
        console.log("Success", result);
        if(result)
        {
            | | alert('post saved');
        }

        //inserted
    },
    error: function (request,error) {
        // This callback function will trigger on unsuccessful action
        console.log("error")
    }
});

```

Figure 53: AJAX JavaScript code used to send the data to a php file in the form of JSON strings

```

<?php

session_start();

require '../includes/dbh.php';

// you passed wrong info from js you need get the post key to ket value
// effectively ajax is giving a query string like .../submitguide.php/?guideTitle=jsvar&Champion=jsvar...

// LOOK UP https://www.w3schools.com/js/js\_json\_php.asp

$guidetitle = $_POST['GuideTitle'];// these are the js
$champion = $_POST['Champion'];
$role = $_POST['Role'];
$patch = $_POST['Patch'];
$author = $_SESSION['UserID'];
$corebuild = $_POST['CoreBuild'];
$description = $_POST['Description'];
$situational = $_POST['Situations'];

$myObj = new StdClass;

$sql = "INSERT INTO `guides` (`ID`, `DateUploaded`, `GuideTitle`, `Champion`, `Role`, `Patch`, `AuthorID`, `CoreBuild`, `Description`, `Situations`)
VALUES (NULL, NOW(), '$guidetitle', '$champion', '$role', '$patch', '$author', '$corebuild', '$description', '$situational')";
// $statement = mysqli_stmt_init($connection);

if (mysqli_query($connection, $sql)) {
    $myObj->success = true;
}

else {
    //echo "Error" . $sql . "<br>" . $connection->error;
    $myObj->success = false;
}
$json = json_encode($myObj);
echo $json;

mysqli_close($connection);
?>

```

Figure 54: PHP code used to send the guide data to the database using SQL queries

#### (d) Displaying Guides stored in the Database to Users

```
//Here is an example where we use the connection to take data from our database, and show it in the browser
$build = [];

//Here we mix PHP and SQL in order to have a statement ready that we can refer to later on
$sql = "SELECT * FROM guides ORDER BY DateUploaded DESC;";
//Here we "query" the SQL statement in the database using our connection variable
$result = mysqli_query($connection, $sql);
//Here we get the number of results the query returned from the database
$resultCheck = mysqli_num_rows($result);

//We then check if we had atleast 1 result from the database
if ($resultCheck > 0) {
    //If we had a result, then we use a while loop to spit out our rows of data, one by one
    //At the same time we also assign the database data to a variable named $row
    while ($row = mysqli_fetch_assoc($result)) {
        //We can spit out the data by referring to our database column names
        $authorID = $row['AuthorID'];
    }
}
```

Figure 55: Code used to get the guides stored in the Database

```
echo'<h2 class="guideTitle">' . $row['GuideTitle'] . '</h2>';

$build = json_decode($row['CoreBuild']);
$build = $build;
foreach ($build as $item)
{
    $item = (array)$item;
    foreach($item as $pic)
    {
        echo ' 0) {
    while ($rowAuthor = mysqli_fetch_assoc($resultAuthor)) {
        echo'<p>' . $rowAuthor['Region'] . '</p>
        <p>' . $rowAuthor['Division'] . '</p>
        <p>' . $rowAuthor['Lvl'] . '</p>';
    }
}
echo'<p>' . $row['Description'] . '</p>';
echo'</div>
</a>';
```

Figure 56: Code used to get and display various pieces of guide information

```
<?php
if (isset($_POST['search-submit'])) {
    //Here we get the search input
    $search = mysqli_real_escape_string($connection, $_POST['search']);
    $build = [];
    echo'<form action="searchResults.php" method="post">
        <input class="field" id="searchbar" type="text" name="search" placeholder="Search...">
        <button class="button" id="searchButton" type="submit" name="search-submit">Search</button>
    </form>';
    //Here we use the keyword "LIKE" in our query and we wrap the $search variable in "%", instead of setting the column name equal to our variable like we usually do.
    $sql = "SELECT * FROM guides WHERE GuideTitle LIKE '%$search%' OR Champion LIKE '%$search%' OR Patch LIKE '%$search%'";

    //Here we get the number of results we got from the search!
    $result = mysqli_query($connection, $sql);
    $resultCount = mysqli_num_rows($result);

    if ($resultCount > 0) {
        echo'<h2>Search Results for "' . $search . '"</h2>';
        while ($row = mysqli_fetch_assoc($result)) {
```

Figure 57: Code used to return search results

```

echo'<h4 class="subheading">Situational Builds</h4>';
$situations = json_decode($row['Situations']);
$situations = $situations;
foreach ($situations as $item)
{
    $item = (array)$item;
    echo'<h4>' . $item['name'] . '</h4>';
    foreach($item['items'] as $pic)
    {
        echo ' | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| strongly disagree     |                       |                       |                       | strongly agree        |

1b) Please explain your answer to the question above

Your answer

2a) I think the system is unnecessarily complex \*

|                       |                       |                       |                       |                       |
|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 1                     | 2                     | 3                     | 4                     | 5                     |
| <input type="radio"/> |
| strongly disagree     |                       |                       |                       | strongly agree        |

2b) Please explain your answer to the question above

Your answer

3a) I think the system is easy to use \*

|                       |                       |                       |                       |                       |
|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| 1                     | 2                     | 3                     | 4                     | 5                     |
| <input type="radio"/> |
| strongly disagree     |                       |                       |                       | strongly agree        |

3b) Please explain your answer to the question above

Your answer

4a) I think that I would need the support of a technical person to be able to use this system \*

1      2      3      4      5

strongly disagree

strongly agree

4b) Please explain your answer to the question above

Your answer

---

5a) I think the various functions in the system are well integrated \*

1      2      3      4      5

strongly disagree

strongly agree

5b) Please explain your answer to the question above

Your answer

---

6a) I think that there is too much inconsistency in the system \*

1      2      3      4      5

strongly disagree

strongly agree

6b) Please explain your answer to the question above

Your answer

---

7a) I would imagine that most people would learn to use the system very quickly

\*

1      2      3      4      5

strongly disagree



strongly agree

7b) Please explain your answer to the question above

Your answer

---

8a) I think the system seems very cumbersome to use \*

1      2      3      4      5

strongly disagree



strongly agree

8b) Please explain your answer to the question above

Your answer

---

9a) I would feel very confident using the system \*

1      2      3      4      5

strongly disagree



strongly agree

9b) Please explain your answer to the question above

Your answer

---

10a) I would need to learn a lot of things before I could get going with the system

\*

1      2      3      4      5



strongly disagree

strongly agree

10b) Please explain your answer to the question above

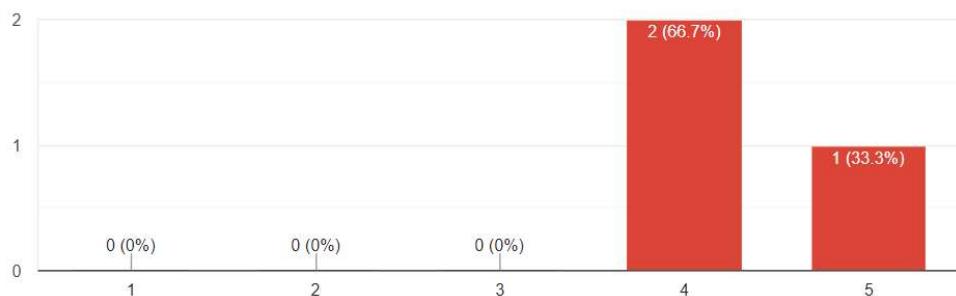
Your answer

**Submit**

## (b) User Responses to the SUS Form

1a) I think that I would like to use this system frequently

3 responses



1b) Please explain your answer to the question above

3 responses

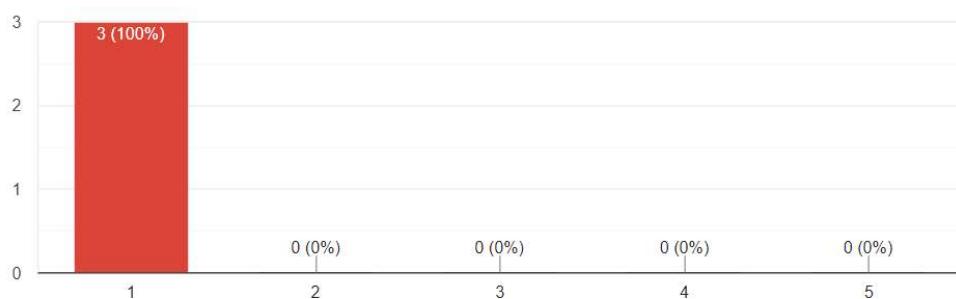
Yes, the site looks very accessible and easy to use, I could see myself bookmarking this for use when playing neww champions and/or to adapt builds of champions I already play.

When learning new champions I would definitely use this system, it would be beneficial to learning how to play the champion optimally.

Looks line a clean and responsive website without the clutter on other similar websites

2a) I think the system is unnecessarily complex

3 responses



2b) Please explain your answer to the question above

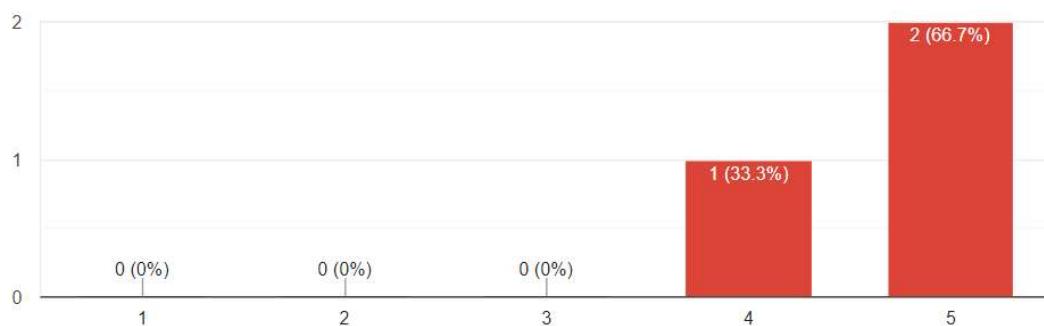
2 responses

No it's not overly complex, it's what it needs to be. Nothing more, nothing less.

the system is quite similar to the current system already in League of Legends, give and take a few extra information such as the rank of the champ in the current meta, so it seems easy to adjust to.

**3a) I think the system is easy to use**

3 responses



**3b) Please explain your answer to the question above**

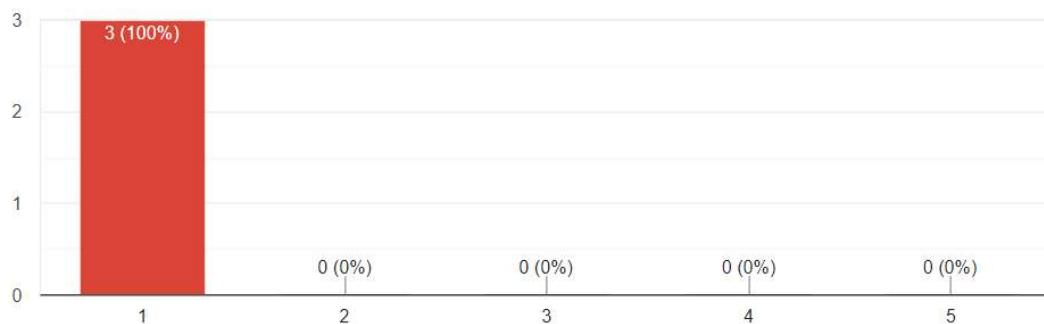
2 responses

Yes, the system looks easy and simple to use, making it accessible, quick and easy for people search and look for guides, aswell as creating and sharing one of their own.

the system is similar to the current system in League of Legends

**4a) I think that I would need the support of a technical person to be able to use this system**

3 responses



**4b) Please explain your answer to the question above**

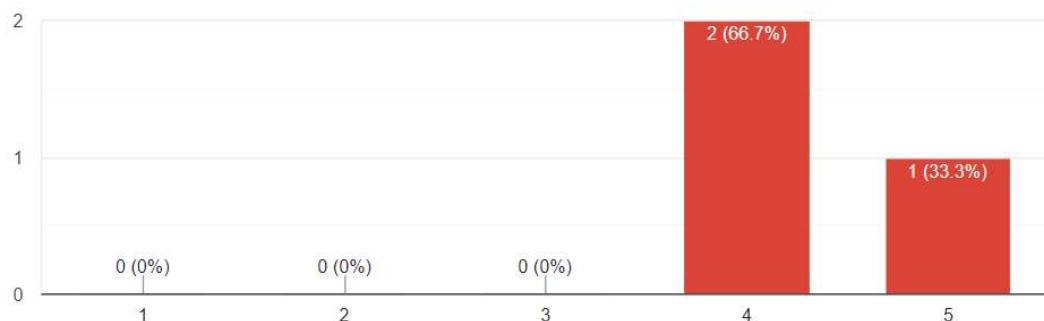
2 responses

Not at all.

it seems easy to use and figure out

5a) I think the various functions in the system are well integrated

3 responses



5b) Please explain your answer to the question above

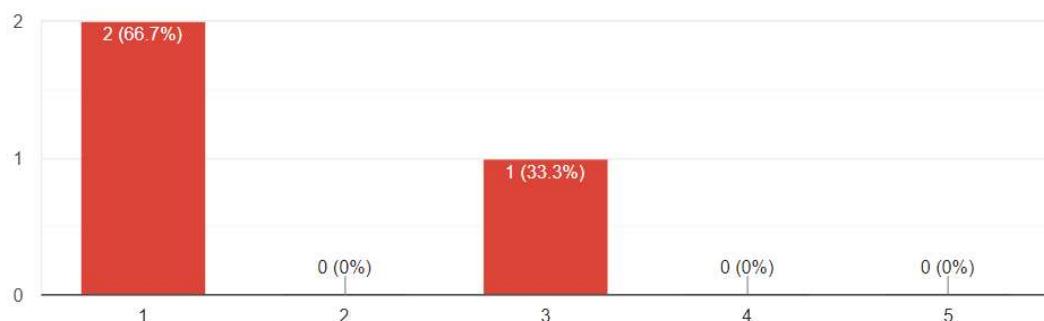
2 responses

Yes, searching for guides ect is quick and easy to use. The drag and drop features on the guide builds are simple and easy to use.

I don't know much about programming etc but it seems to work well!

6a) I think that there is too much inconsistency in the system

3 responses



6b) Please explain your answer to the question above

2 responses

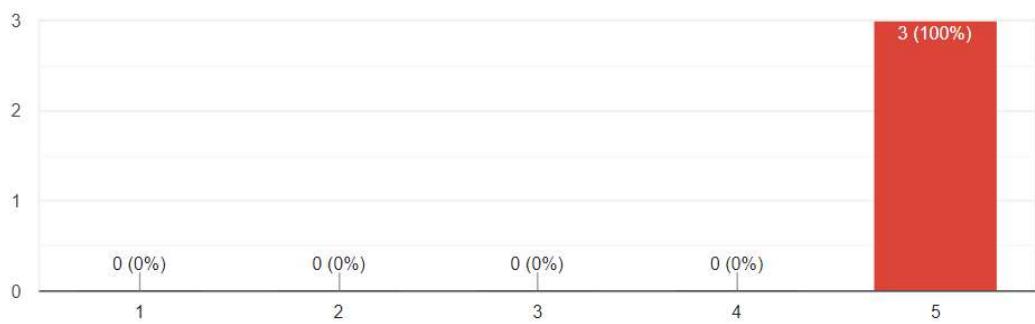
No, everything seems to be consistant, in the same theme. making looking for things on different sections easy.

no, the system seems consistent, it looks similar to League itself which is helpful regarding familiarity, it looks impressive

7a) I would imagine that most people would learn to use the system very quickly



3 responses



7b) Please explain your answer to the question above

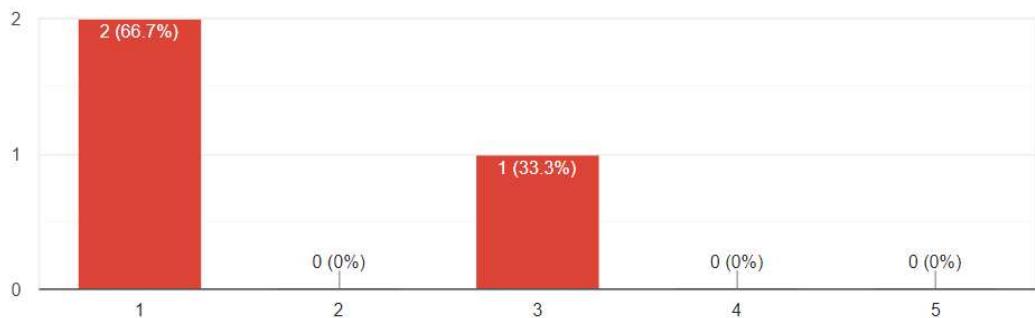
2 responses

Yes, it seems very easy. simple search functions and clear sections make it very obvious where to go, and what to do. Making guides is pretty self explanatory.

it seems very user friendly

8a) I think the system seems very cumbersome to use

3 responses



8b) Please explain your answer to the question above

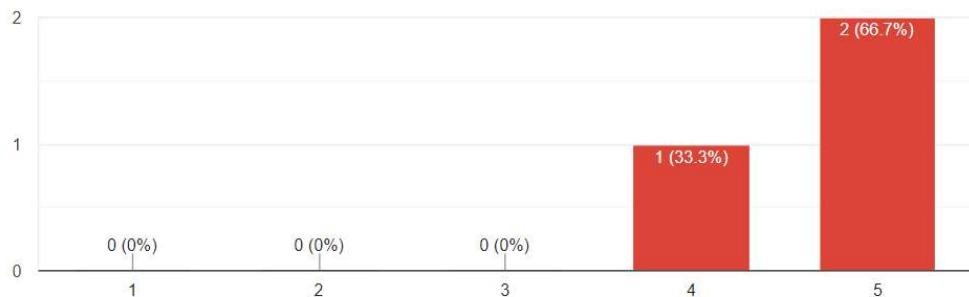
2 responses

No, it is simple and to the point. Making an account is simple are efficient. Guides are easy to find. There is no unnecessary information on display.

some people may not want to use a system such as this as they might already know the builds of the champs they currently play, but those learning to play new champs would find this easy to use

9a) I would feel very confident using the system

3 responses



9b) Please explain your answer to the question above

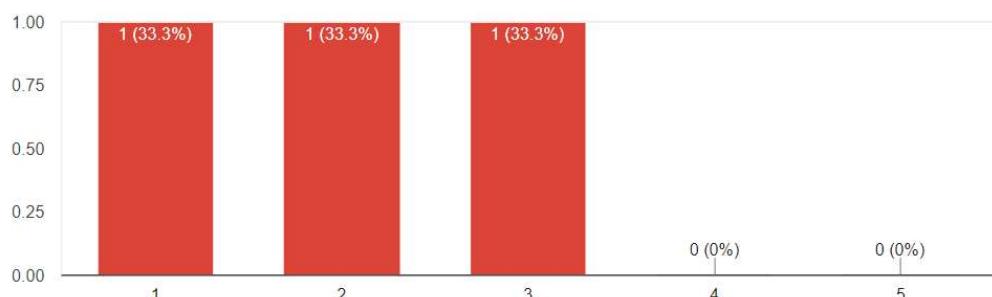
2 responses

I could see myself using this system often to find guides and build options before playing a match.

sometimes builds work for a persons play style, sometimes they don't, finding a build which works for your style would be important

10a) I would need to learn a lot of things before I could get going with the system

3 responses



10b) Please explain your answer to the question above

2 responses

No, the system is self explanatory and easy to access. It seems to be designed with quick accessibility in mind, meaning even someone is able to jump on very quickly and find a guide, even if they haven't used the site before. That speed is important for people looking up guides, as often they are quickly searching before their game starts.

The only scenario where someone would need to learn a lot of things, would be if a new player was to make a guide, as they would require game experience and knowledge. But that is the purpose of the site. These new players will be able to learn more about the game and how to build champions through the existing guides on the site.

it is informative and would be easy to start using