OWL Predict AP2

Jude Dillon – B00737902  
BEng Software Engineering

# Abstract (500 Words)

The purpose of the abstract is to give a summary of the overall project, enabling the reader to gain an impression of the origins, aims, nature and final results of the work, without having to read the detail of later chapters. The abstract should not exceed 500 words.

# Acknowledgements/Dedication - (Optional)

Contents

[Abstract (500 Words) 2](#_Toc70364041)

[Acknowledgements/Dedication - (Optional) 2](#_Toc70364042)

[1 Requirement Control Document & Modification of the Project Plan 4](#_Toc70364043)

[1.1 Final List of Requirements 4](#_Toc70364044)

[1.1.1 Functional Requirements 4](#_Toc70364045)

[1.1.2 Non-Functional Requirements 4](#_Toc70364046)

[1.2 Requirements Evolution 5](#_Toc70364047)

[1.3 Modifications done to the project plan 5](#_Toc70364048)

[2 System Design 6](#_Toc70364049)

[2.1 System Architecture Diagram 6](#_Toc70364050)

[2.2 Interface Design 7](#_Toc70364051)

[2.3 Data Support Design 7](#_Toc70364052)

[2.3.1 Consideration of Security and Data Validation 7](#_Toc70364053)

[2.3.2 ER Diagram 7](#_Toc70364054)

[2.4 User Interaction Design 8](#_Toc70364055)

[2.5 Additional Design Artefacts 8](#_Toc70364056)

[3 System Implementation 8](#_Toc70364057)

[4 System Verification 8](#_Toc70364058)

[5 System Validation 8](#_Toc70364059)

[6 Conclusion and Reflection 8](#_Toc70364060)

[7 References 9](#_Toc70364061)

[8 Appendices 9](#_Toc70364062)

# 1 Requirement Control Document & Modification of the Project Plan

## Final List of Requirements

### 1.1.1 Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Requirement | Priority | Risk Level |
| F1 | User can select two different Overwatch League teams | Must Have | Low |
| F2 | System will predict team that will win using a Machine Learning Algorithm | Must Have | High |
| F3 | System will output team that it predicts to win | Must Have | Medium |
| F4 | System needs to be able to extract data used for predictions from the dataset | Must Have | Medium |
| F5 | System will provide accurate predictions (above 60% accuracy) | Should Have | Low |
| F6 | System will output a percentage stating how sure it is of its prediction | Should Have | Low |
| F7 | Users can tune the range of closest data points that the system uses to make decisions | Should Have | Low |
| F8 | Users can choose which season of Overwatch League will be used to make predictions in the system | Should Have | Medium |
| F9 | System will make prediction within 2 seconds [1] | Should Have | Medium |
| F10 | User can make predictions through an API | Could Have | Low |
| F11 | System will predict upcoming matches in advance | Won’t Have | Low |

Figure Functional Requirements

### 1.1.2 Non-Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Requirement | Priority | Risk Level |
| NF1 | System will be robust | Must Have | Medium |
| NF2 | System will be intuitive | Must Have | Low |
| NF3 | System will look visually appealing | Must Have | Low |
| NF4 | System must work on majority of browsers | Must Have | Medium |

Figure Non-Functional Requirements

## Requirements Evolution

During the development process some changes were made to the initial requirements that were created during the initial project planning.

F8 was changed from “Users can choose which data will be used to make predictions in the system” to “Users can choose which season of Overwatch League will be used to make predictions in the system”. The justification for this change was that although the initial plan was for the user to be able to change both the seasons used for predictions and be able to choose different predictors for the system to use for predictions the developer found that changing the predictors used for each prediction was much more difficult to implement than expected. As a result of this the Project Manager made the decision to instead just implement the selection of which season the data used for predictions would be gathered from so users would still be able to have a more specific prediction to their liking.

No other changes were made to the requirements because from discussions between the project manager and a focus group of target users the remaining requirements were all deemed sufficient for what members of the focus group expected from the product.

## Modifications done to the project plan

# 2 System Design

The approach to the design of the architecture of the system was to make it all flow in a way where it moved the data from the frontend, to the backend via an API call which would access the database to then calculate the prediction with the K Nearest Neighbour algorithm [2] which would then return the prediction from the original API call to the frontend.  
The overall plan for the design of the system from a user standpoint was to make it as simple and intuitive for the users as possible so it was easy for them to use, this was achieved with the use of abstraction [3].  
The visuals, in particular the colours of the system were inspired by Overwatch League [4] and the colour scheme used in it because it would be familiar to users of the system.

## 2.1 System Architecture Diagram

Diagram

Description automatically generated

Figure System Architecture Diagram

The system architecture diagram in Figure 3 shows an overall view of OWL Predict, how the data flows through the system and how each section of the system communicates with each of the other sections.  
The user opens the website and the webpage which is created via angular JavaScript will display the values the user needs to input to make a prediction.   
When the user inputs and submits the required values the frontend will make a call containing these values to the API which is created in python.  
The API will then convert these values to a format ready for the K Nearest Neighbour Algorithm and then send them to the K Nearest Neighbour algorithm.  
The K Nearest Neighbour algorithm will send these values to the mongo DB database to calculate the Predictors for the inputted data as well as calculate the predictors for stored data when using the users selected season.  
When these values are all returned from the database the K Nearest Neighbour Algorithm will create a list of all the K nearest neighbours to the input data where K is the number of neighbours selected and submitted by the user.   
The algorithm will then gather the responses from this list of nearest neighbours and if the outcomes of this game were a win or a loss for team 1, if more of the nearest neighbours predict the outcome as a win then the algorithm will return to the API that it predicts a win for team 1 and its percentage confidence in it which is calculated as the percentage of nearest neighbours where the outcome was a win and vice versa for a win for team 2.  
The API will then return the result of this prediction to the frontend where it will be displayed to the user.

## 2.2 Interface Design

Provide your interface storyboards and wireframes, explain each figure (5 Pages Maximum)

Provide a narrative establishing your consideration for HCI and Usability/Accessibility of the User Interface.

## 2.3 Data Support Design

### 2.3.1 Consideration of Security and Data Validation

### 2.3.2 ER Diagram

A screenshot of a computer

Description automatically generated with medium confidence

## 2.4 User Interaction Design

A picture containing diagram

Description automatically generated

## 2.5 Additional Design Artefacts

# 3 System Implementation

# 4 System Verification

# 5 System Validation

# 6 Conclusion and Reflection

# 7 References

|  |  |
| --- | --- |
| [1] | F. F.-H. Nah, A Study on Tolerable Waiting Time: How Long Are Web Users Willing to Wait?, Association for Information Systems, 2003, p. 285. |
| [2] | N. S. Altman, “An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression,” *The American Statistician,* vol. 46, no. 3, pp. 175-185, 1992. |
| [3] | Sharpened Productions, “Abstraction Definition,” TechTerms, 19 April 2019. [Online]. Available: https://techterms.com/definition/abstraction. [Accessed 26 04 2021]. |
| [4] | Wikipedia, “Overwatch League,” Wikipedia, 26 04 2021. [Online]. Available: https://en.wikipedia.org/wiki/Overwatch\_League. [Accessed 26 04 2021]. |

# 8 Appendices