CA400 Functional Specification

Lower Limb Sports Injury Prediction and Prevention

By:

Dylan Bagnall - 20413066

Lorcan Dunne - 19311511

Functional Specification Contents:

Table of contents Page Number

1. Introduction

1. **Overview**
1. **Purpose**

Lower limb injuries are a major and pressing problem in the high intensity sport. These injuries can have a significant impact on players' careers and lives. There is a growing interest in using machine learning to predict and prevent injuries. Machine learning models can be trained on historical data to identify patterns that are associated with injuries. These patterns can then be used to predict which players are at increased risk of injury. This information can then be used to develop prevention strategies.

The implementation of machine learning in injury prevention strategies has the potential to revolutionise the landscape of high-intensity sports. By giving athletes and healthcare professionals actionable insights, machine learning can foster a proactive approach to injury prevention, safeguarding athletes' well-being and optimising their performance.

2. **Scope**

   This project aims to develop a lower limb injury prediction model and its accompanying user interface. The project's scope excludes real-time injury monitoring or medical diagnosis. Instead, the model will provide insights into potential risk factors and trends related to lower limb injuries, serving as a complementary tool to professional medical judgement rather than a replacement.

   The primary objectives of this project are to:

1. **Develop a lower limb injury prediction model:** This model will utilise machine learning algorithms to analyse historical data and identify patterns associated with lower limb injuries. The model will be trained to predict which individuals are at an increased risk of injury based on various factors, such as previous injury occurrences, what turf these injuries took place on and what the weather
2. **Design a user-friendly interface:** Using Flask we aim for the interface to provide a seamless and intuitive platform for users to interact with the injury prediction model.

2. **Business Context**

Our aspiration for this model is to extend its reach and provide assistance to as many NFL players as possible. To achieve this goal, we envision our model being freely accessible to all NFL teams and their physiotherapists. This widespread adoption would undoubtedly benefit the NFL in several ways. Firstly, by optimising player health and performance, our model would contribute to extending the careers of star players, allowing them to remain on the field for a longer duration. This, in turn, would translate into increased viewership and, consequently, higher profits for the league. Additionally, the model's ability to effectively manage and prevent injuries could lead to a reduction in the number of games missed by key players, further enhancing the quality of gameplay and fan engagement.

Although the NFL is our primary emphasis right now, we hope to eventually include Irish sports like rugby and Gaelic games in the project's scope. However, this growth is not feasible at this time due to the absence of relevant data. However, we are adamant in our commitment to look at ways to broaden the model's use and have a beneficial influence on sports medicine outside of the NFL.

3. **Glossary**

Decision Trees	A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute,each branch represents the outcome of the test, and each leaf node represents a class label. The paths from root to leaf represent classification rules. Commonly used in machine learning tasks.
Random Forest	A random forest is an ensemble learning method that consists of a collection of decision trees. Each decision tree is trained on a random subset of the training data, and the predictions of the individual trees are then combined to produce a final prediction.
XGBoost	Stands for Extreme Gradient Boosting. It is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library.

NFL	National Football League (American Football)
Flask	A web framework written in Python.
Data Cleaning	Identifying and correcting errors, inconsistencies and inaccuracies in a dataset
Kaggle	A website for open sourced, published datasets.

2. General Description

2.1 Product / System Functions

The AI prediction model will be able to predict the likelihood of a lower limb injury in a high intensity sport such as the NFL, given a set of input features. The model will be trained on a dataset of historical lower limb injuries and player performance data that can be found as a competition on Kaggle. The model will be able to be used by players, coaches, and other stakeholders to identify players who are at increased risk of injury and to develop prevention strategies.

2.2 User Characteristics and Objectives

General User

The management and coaching staff of an NFL club, which may include data analysts, physiotherapists, and sports scientists, are the target audience for the prediction model. These people should be moderately to highly proficient with Flask and other software systems that will be used to communicate with the model. They might not be deeply versed in machine learning algorithms, but they should be able to operate with most data analytics software. With this degree of expertise, they will be able to use the model's predictions to make informed decisions about the performance and health of their athletes.:

- **Sports Scientists:** Sports scientists are responsible for analysing and interpreting player data to optimise performance and reduce the risk of injury. They will utilise the predictive model to identify potential injury risks and develop personalised training and rehabilitation programs for players.

- **Physiotherapists:** Physiotherapists play a crucial role in the prevention, diagnosis, and treatment of player injuries. They will utilise the predictive model to assess injury risks and tailor their treatment plans accordingly.
- **Data Analysts:** Data analysts are responsible for collecting, cleaning, and analysing large datasets of player data. They will use the predictive model to identify trends and patterns in injury data, which can inform future injury prevention strategies.

In addition to the technical skills mentioned above, users of the predictive model are also expected to have a strong understanding of human movement and biomechanics. This knowledge will be essential in interpreting the model's predictions and applying them to real-world scenarios.

By providing the management and coaching staff of NFL teams with a user-friendly and informative predictive model, we aim to empower them to make informed decisions that promote player health, optimise performance, and ultimately contribute to the success of their teams.

User Objectives

The primary objective of the predictive model is to provide NFL teams with a comprehensive tool for injury risk assessment. By identifying players at an elevated risk of injury, the model will enable management staff to make informed decisions regarding player well-being and game time participation. This data-driven approach is particularly valuable for teams seeking to optimise player performance and minimise injury-related setbacks.

The model's user-friendly interface will cater to a diverse range of users with varying levels of technical expertise. This accessibility ensures that sports scientists, physiotherapists, data analysts, and management staff can effectively utilise the model's predictions to inform their decision-making processes. The system will cater for teams that are interested in data-driven decision-making, when it comes to managing their roster and planning for upcoming games. The teams need a system that has a user-friendly interface that is accessible to staff with varying levels of technical expertise.

2.3 Operational Scenarios

Use case 1

Name:		Data Cleaning
Pre-Conditions:	1	Accessing the data from the Kaggle files
Description:	1 2 3 4	Open the data files Remove irrelevant/duplicate data Deal with missing data and outliers Structure data
Post-Conditions:	1	Dataset has no outliers or missing data

Use case 2

Name:		Data Input
Pre-conditions:	1.	Accessing the data from the Kaggle files
Description:	1 3 4	Collecting the data from the files Ensuring the data is clean and accurate The data is then inputted into the model
Post-Conditions:	1	The model analyses the data

Use case 3

Name:		Data Training
Pre-conditions:	1	The data has been pre-processed
Description:	1 2 3 4	The data has been separated into training and validation sets Categorise variables to suit machine learning algorithms Algorithm is selected and learns patterns between input features and target variables Training model is validated using a separate dataset
Post-Conditions:	1	Model is successfully trained and capable of making predictions

Use case 4

Name:		Displaying models results on Flask
Pre-Conditions:	1 2	The machine learning model has been trained and evaluated Flask web app is setup and ready to integrate the model
Description:	1	The trained model is integrated into the Flask web app - may need API endpoint for

		handling predictions
	2	Input data is sent to the model endpoint for prediction
	3	Flask endpoint processes the data and uses the model to make predictions
	4	The predicted results are rendered on the Flask web page showing the probability of a lower limb injury for a given player
Post-Conditions:	1	The Flask web app displays the predicted results for lower limb injury risks based on the machine learning model

Use case 5

Name:		Data evaluation
Pre-Conditions:	1	Training and validation sets have been completed
Description:	1	The evaluation dataset is prepared separate to the training set but with the same features
	2	Trained model is deployed to make predictions on the evaluation dataset

	3	Performance is analysed using measures such as accuracy, precision, recall.
Post-Conditions:	1	Models performance is analysed and documented

The following operational scenarios illustrate how the AI prediction model will be used by different stakeholders:

- **Player:** A player can use the model to identify their own risk factors for injury and to develop a personalised prevention plan.
- **Coach:** A coach can use the model to identify players on their team who are at increased risk of injury. The coach can then develop personalised training plans for these players to help reduce their risk of injury.
- **Sports medicine professional:** A sports medicine professional can use the model to develop general injury prevention strategies for high intensity sports teams.
- **Sport clubs and board members:** When a club is looking into the possibility of building a synthetic turf pitch. The board members can use the model to identify if there is an injury risk related to constant training and matches on a synthetic turf pitch.

2.4 Constraints

When developing this AI prediction modal, there will be multiple constraints that we will inevitably encounter. Listed below are a few constraints that we may encounter.

- The computational resources available. Training and running the model requires significant computational resources.
- The complexity of the problem. Accurately predicting lower limb injuries requires a comprehensive understanding of the intricate interactions between factors such as biomechanical, physiological, and environmental elements. Conventional methods of injury prediction often rely on expert knowledge and statistical models, which may not be able to capture the full complexity of the problem. As a result, these methods may struggle to identify individuals at high risk of injury, leading to potential missed opportunities for prevention.

3. Functional Requirements

Requirement: Data Collection and Integration

Description	The system must be capable of gathering and integrating historical player performance data and lower limb injury records from various sources, including NFL databases and Kaggle datasets. This requirement involves data retrieval, data cleaning, and ensuring that the collected data is accurate and up to date.
Criticality	This requirement is of utmost criticality to the overall system, as it forms the foundation for all subsequent processes. Without reliable and comprehensive data, the AI model's predictions and injury risk assessments will be unreliable and potentially harmful.
Technical issues	Implementing Data Collection and Integration may involve challenges related to data quality, format compatibility, and data source connectivity. Technical issues may include data cleansing, handling missing values, and establishing a data pipeline that can efficiently retrieve, process, and integrate data from multiple sources
Dependencies with other requirements	Data Collection and Integration is closely tied to Feature Selection and Preprocessing, Model Training and Validation, Real-time Prediction, and Reporting and Analytics. Without proper data collection and integration, the other requirements cannot be effectively implemented.

Others as appropriate	The requirement may involve establishing data version control, data backup strategies, and data source documentation for transparency and accountability.

Requirement: Feature Selection and Preprocessing

Description	The system should include techniques for selecting relevant input features from the collected data, such as player statistics, physical characteristics, and game-related factors. Additionally, it must handle data preprocessing tasks, including addressing missing values, outliers, and normalising the data to ensure its quality and suitability for predictive modelling.
Criticality	This requirement is critical to the overall system as it directly impacts the quality and effectiveness of the AI model. Selecting the right features and preparing the data correctly are essential for accurate injury risk predictions.
Technical issues	Feature Selection and Preprocessing may involve challenges related to feature engineering, such as identifying which player attributes are most relevant for predicting lower limb injuries. Additionally, dealing with missing or noisy data, outlier detection, and selecting appropriate data scaling and normalisation techniques are important technical aspects.

Dependencies with other requirements	Feature Selection and Preprocessing is closely intertwined with Data Collection and Integration as the data used for prediction must first be processed. It also impacts Model Training and Validation, as the quality of the preprocessed data affects the model's performance.
Others as appropriate	Considerations may include automated feature selection algorithms, data transformation techniques, and the development of procedures for handling different types of data (numerical, categorical, text, etc.). Ensuring transparency in the feature selection process is important for building trust in the model's predictions. Additionally, documenting feature selection and preprocessing methods can help with model reproducibility and compliance.

Requirement: Machine Learning Model

Description	Analysing historical data—such as player injury records, training volume, playing circumstances, and other significant variables—requires the machine learning model. By recognizing patterns and trends related to injuries to the lower limbs, the model will be able predict the risk of injuries to specific athletes. It needs to be something that's always learning and changing to accommodate new information.

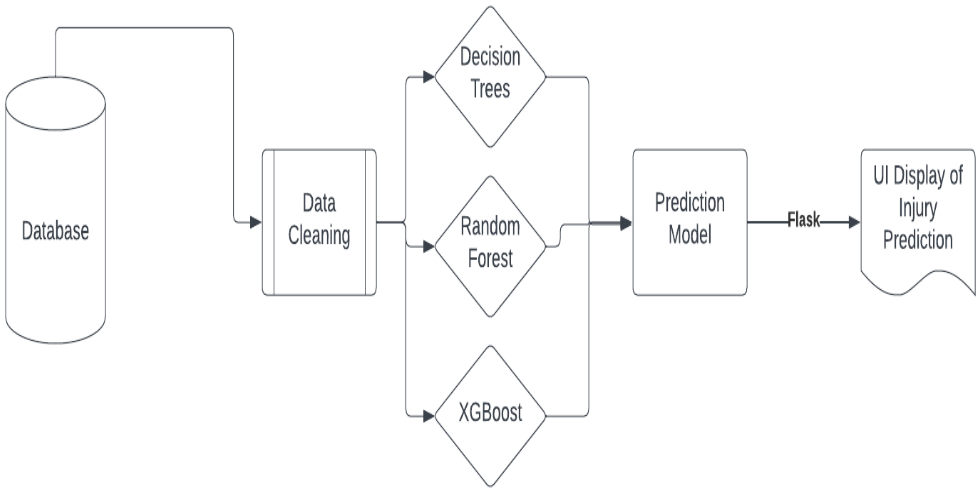
Criticality	This set of requirements is important for the system as a whole because the machine learning model's accuracy and dependability play a major role in how well the injury prediction and preventive methods work. The effectiveness of this component directly affects the project's ability to reduce lower limb injuries.
Technical issues	When creating a machine learning model for injury prediction, the right algorithms must be chosen, features must be created, and performance must be optimised by adjusting hyperparameters. To prevent bias in predictions, handling imbalanced datasets—where the number of non-injury instances may greatly outnumber the number of injury instances—requires careful thought.
Dependencies with other requirements	The completeness and quality of the historical data gathered determine how accurate the machine learning model is. As a result, the need for data integration and collecting is directly dependent on it. The machine learning model and the user interface must be able to communicate with each other efficiently so that the model can display its insights and receive prediction inputs.
Others as appropriate	

Requirement: Displaying the models prediction on Flask

Description	The system should include a functionality to display the machine learning model's predictions on a user interface built using Flask. This involves creating a web page or interface where users can input relevant data, trigger the prediction process, and view the model's predictions regarding the likelihood of lower limb injuries for individual players
Criticality	This requirement is crucial to the overall system as it directly interfaces with users and provides them with insights into the likelihood of players sustaining an injury. The effectiveness of injury prevention strategies relies on the timely and accurate communication of these predictions to relevant stakeholders, making it critical to the success of the project.
Technical issues	Designing and implementing a responsive and user-friendly web interface using Flask. Establishing a secure connection between the user interface and the backend system to ensure the confidentiality and integrity of the prediction data. Integrating the Flask application with the machine learning model, allowing seamless communication and data exchange. Implementing real-time updates on the user interface to reflect the latest predictions from the machine learning model.
Dependencies with other requirements	The functionality to display model predictions depends on the successful implementation and performance of

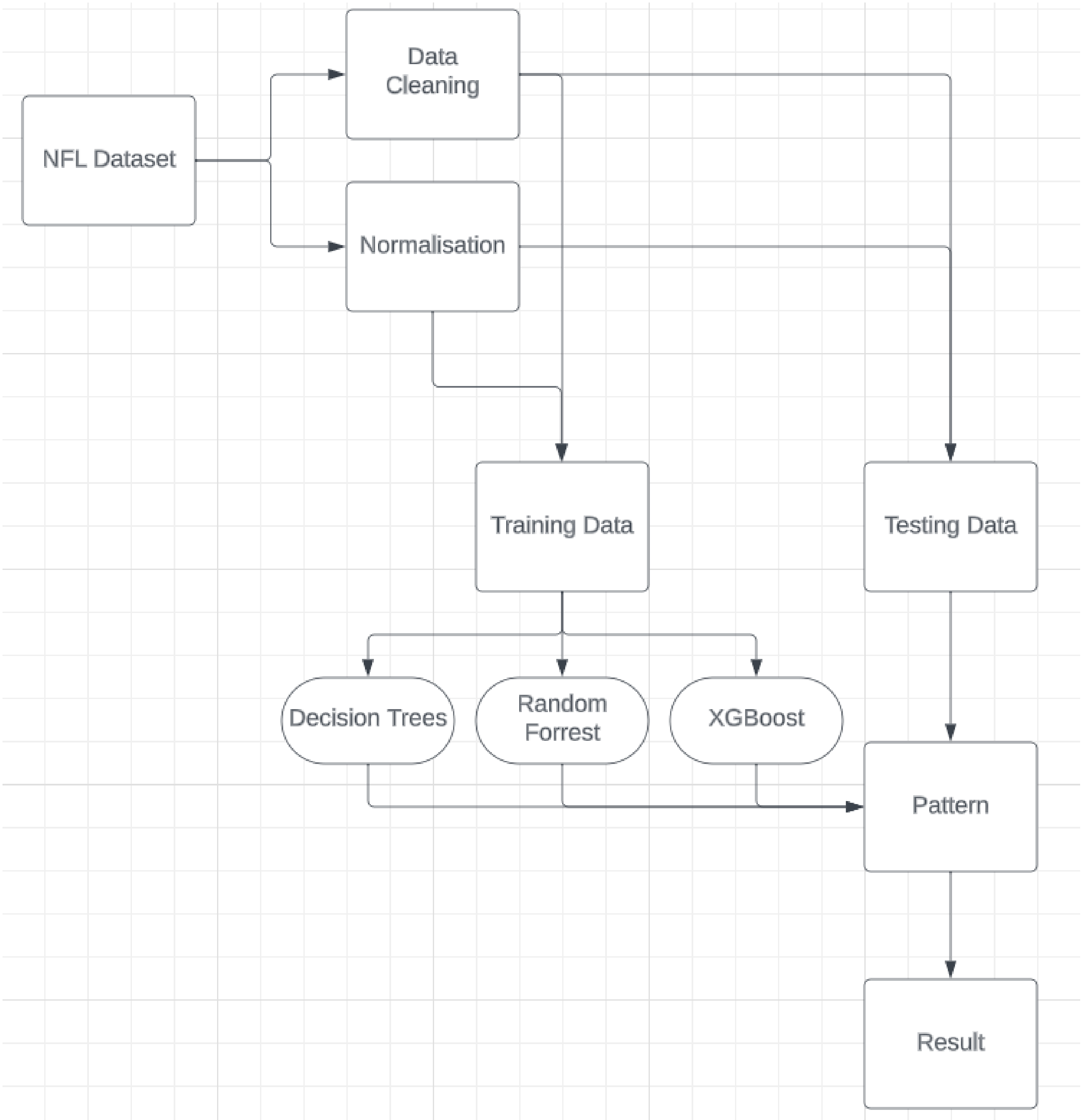
	the machine learning model requirement. Accurate and reliable predictions are essential for providing meaningful insights to users through the Flask interface. Interaction with the data collection and integration requirement is necessary to ensure that the user interface has access to the relevant historical data needed for making predictions.
Others as appropriate	Implementing visualisation elements on the Flask interface to help users interpret and understand the model's predictions effectively.

4. System Architecture

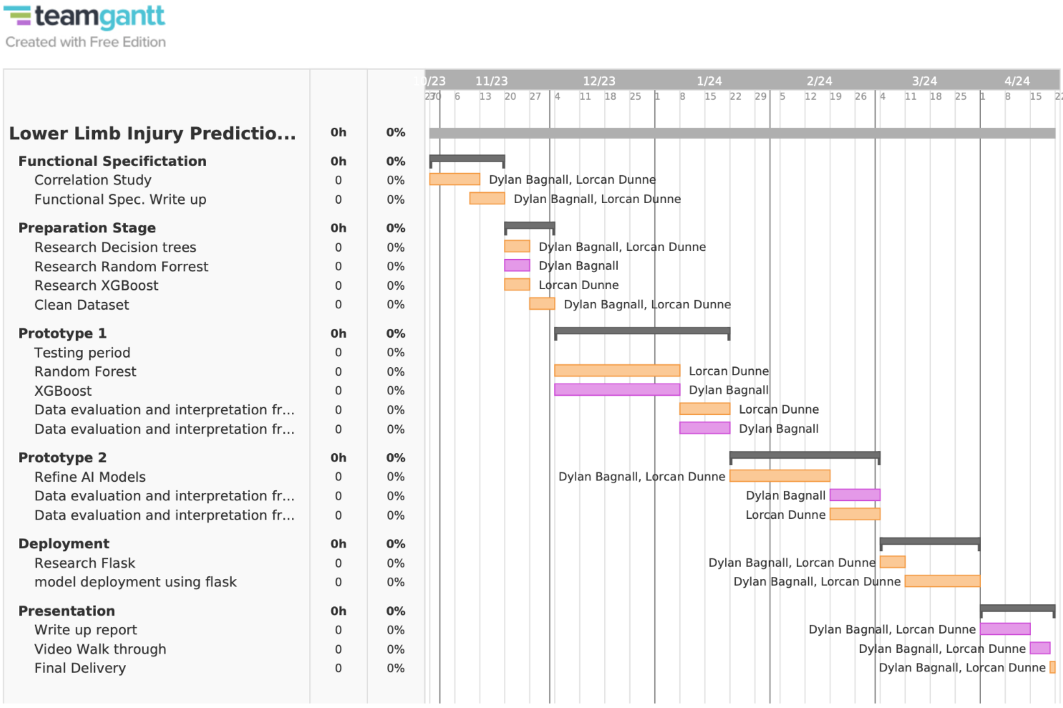


5. High-Level Design

Data Flow Diagram:



6. Preliminary Schedule



7. Appendices

- This is the dataset from kaggle that we will be using for to train this model:
  - https://www.kaggle.com/code/aleksandradeis/nfl-injury-analysis

- Some links we found useful in learning and understanding about predictive modelling:

  Basics on predictive modelling:
  - https://www.techtarget.com/searchenterpriseai/definition/predictive-modeling#:~:text=Predictive%20modeling%20is%20a%20mathematical,forecast%20activity%2C%20behavior%20and%20trends.

  Learning about predictive model algorithms:
  - https://medium.com/@ghanshyamsavaliya/11-most-popular-data-prediction-algorithms-that-help-for-decision-making-d6c73b796db9

  Preparing the data for training:
  - https://www.projectpro.io/article/data-preparation-for-machine-learning/595