

### Project Report

on

**Optimal Real Time Substitution in Football**

Submitted by

**Project Members**

Malay Jignesh Shah (1032212062)

Sayal Goyal (1032212395)

Kanish Singhania (1032211295)

**Under the Internal Guidance of**

Prof. Rashmi Rane

## Department of Computer Engineering and Technology MIT World Peace University, Kothrud,

**Pune 411 038, Maharashtra - India**

## 2024-2025



### DEPARTMENT OF COMPUTER ENGINEERING AND TECHNOLOGY

**C E R T I F I C A T E**

This is to certify that,

Malay Jignesh Shah (1032212062)

Sayal Goyal (1032212395)

Kanish Singhania (1032211295)

of BTech. (Computer Science & Engineering) have completed their project titled “*Optimal Real Time Substitution in Football”* and have submitted this Capstone Project Report towards fulfillment of the requirement for the Degree-Bachelor of Computer Science & Engineering (BTech-CSE) for the academic year 2023-2024.

**[Prof. Rashmi Rane] [Dr. Balaji M Patil]**

Project Guide Program Coordinator

Department of CET Department of CET

MIT World Peace University, Pune MIT World Peace University, Pune

Internal Examiner:

External Examiner:

**Date:**

# Acknowledgement

We would like to express our deepest appreciation to all those who provided us with the possibility to complete the Seminar research. A special gratitude we give to our Capstone guide, Prof. Rashmi Rane from MIT-WPU whose contribution to stimulating suggestions and encouragement, helped us to coordinate our project as well as in writing this report.

Furthermore, we would also like to acknowledge with much appreciation the crucial role of the staff of MIT World Peace University, who permitted us to use all required equipment and the necessary materials to complete this report “Optimal Real Time Substitution in Football”. Last but not least, many thanks to our professors, who have invested their full effort in guiding and achieving this goal. We have to appreciate the guidance given by other supervisors that improved our presentation skills due to their comments and advice.

Malay Jignesh Shah (1032212062)

Sayal Goyal (1032212395)

Kanish Singhania (1032211295)

# Abstract

This study investigates the impact of risk-taking on football match outcomes, focusing on player substitutions. The analysis reveals that risk-taking propensity peaks when a team is trailing by 2-3 goals and diminishes when leading by the same margin. Younger managers outperform middle-aged ones in risky decisions, while older managers excel in later substitutions. Additionally, a manager's tenure with the team increases the effectiveness of risk-taking, particularly in earlier substitutions and stronger teams. This study also emphasizes the importance of mental state in player performance, proposing a framework combining Match Score Analysis (Kaplan-Meier Fitter) and Score Analysis to evaluate players' mental stability and survival rates during the game. By integrating these models, teams can make better-informed decisions regarding substitutions, considering both past performance and mental health, ultimately enhancing match outcomes. This research underscores the synergistic potential of combining black-box causal machine learning with interpretable models, offering valuable insights for football management and beyond.

List of Figures

* 1. Web application 3
  2. System Architecture[… 2](#_gjdgxs)0

5.5.1 Team Manager Use Case Diagram[… 2](#_30j0zll)1

5.5.2 Player Use Case Diagram[… 2](#_30j0zll)1

5.6 Activity Diagram[… 2](#_1fob9te)2

5.7 Class Diagram… 23

5.8 Module Diagram………………………………………………………………24

6.1 Project Timeline… 32

* 1. Code snippet… 32
  2. Web Application glimpse… 32

9.1 Result of Web Application… 37

9.2 Result of Web Application… 37

# List of Tables

* 1. Journal and Conference Papers ……………………………………………… 4
  2. Our Approach to Earlier Work ………………………………………………. 6
  3. Requirements Rationale …………………………………………………….. 13
  4. Risk Management …………………………………………………………… 14

# Contents

# Abstract …………………………………………………………………………………... I

List of Figures …………………………………………………………………………... II

List of Tables …………………………………………………………………………… III

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | Introduction | | | 1 |
|  | 1.1 | Project Statement | | 1 |
|  | 1.2 | Area | | 1 |
|  | 1.3 | Project Introduction and Aim | | 2 |
| 2 | Literature Survey | | | 4 |
|  | 2.1 | Journal and Conference Papers | | 4 |
|  | 2.2 | Our Approach to Earlier Work | | 6 |
| 3 | Problem Statement | | | 7 |
|  | 3.1 | Project Scope | | 7 |
|  | 3.2 | Project Assumptions | | 8 |
|  | 3.3 | Project Limitations | | 9 |
|  | 3.4 | Project Objectives | | 9 |
| 4 | Project Requirements | | | 10 |
|  | 4.1 | Resources | | 10 |
|  |  | 4.1.1 | Human Resources | 10 |
|  |  | 4.1.2 | Software Requirements | 10 |
|  |  | 4.1.3 | Hardware Requirements | 11 |
|  | 4.2 | Requirements Rationale | | 13 |
|  | 4.3 | Risk Management | | 14 |
|  | 4.4 | Functional Requirements from Organization | | 15 |
|  |  | 4.4.1 | Technical Requirements | 15 |
|  |  | 4.4.2 | Non-Technical Requirements | 16 |

|  |
| --- |
| Team Manager Use Case Diagram |
| Player Use Case Diagram |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 5 | System Analysis & Proposed Architecture | | | 17 |
|  | 5.1 | Design Considerations | | 17 |
|  | 5.2 | Assumptions and Dependencies | | 18 |
|  | 5.3 | General Constraints | | 19 |
|  | 5.4 | System Architecture | | 20 |
|  | 5.5 | Use Case Diagram | | 21 |
|  |  | 5.5.1 | | 21 |
|  |  | 5.5.2 | | 21 |
|  | 5.6 | Activity Diagram | | 22 |
|  | 5.7 | Class Diagram | | 23 |
|  | 5.8 | Modules of Project | | 24 |
|  |  | 5.8.1 | Module:1 Defining Goals and Requirements | 24 |
|  |  | 5.8.2 | Module 2: Algorithm Development | 24 |
|  |  | 5.8.3 | Module 3: Model Training and Testing | 25 |
|  |  | 5.8.4 | Module 4: Integration and Deployment | 25 |
| 6 | Project Timeline | | | 26 |
| 7. | Implementation | | | 30 |
|  | 7.1 | Methodology | | 30 |
| 8 | Performance Evaluation and Testing | | | 33 |
|  | 8.1 | Performance Evaluation | | 33 |
|  | 8.2 | Execution Time | | 34 |
| 9 | Results and Analysis | | | 35 |
|  | 9.1 | Test Results Discussion | | 35 |
|  | 9.2 | Test Results Analysis | | 36 |
| 10 | Deployment Strategies | | | 38 |
| 11 | Applications | | | 39 |
| 12 | Conclusion | | | 41 |
| 13 | Future Aspects | | | 42 |
| 14 | References | | | 44 |

Chapter 1

Introduction

## Problem Statement:

The study seeks to examine how risk-taking, particularly through player substitutions, impacts football match outcomes. By integrating causal machine learning with interpretable models, the study seeks to optimize substitution strategies for improved match outcomes, offering valuable insights for football management.

**1.2 Area**

The study seeks to examine how risk-taking, particularly through player substitutions, impacts football match outcomes. By integrating causal machine learning with interpretable models, the study seeks to optimize substitution strategies for improved match outcomes, offering valuable insights for football management.

**ReactJs:** ReactJS is a popular JavaScript library for building user interfaces. It simplifies the creation of interactive and dynamic web applications by allowing developers to design reusable components. React efficiently updates and render components as data changes, enhancing the user experience by ensuring seamless and responsive web applications.

**Tailwind CSS** is a highly customizable, utility-first CSS framework that enables developers to style applications directly within their HTML by using predefined classes. It eliminates the need to write custom CSS, promoting rapid development while maintaining design consistency.

**Visual Studio Code:** Visual Studio Code (VS Code) is a free, lightweight, crossplatform source code editor developed by Microsoft. It is widely used by developers for various programming languages and provides a rich set of features and extensions that enhance productivity and streamline the development workflow.

**Jupyter Notebook:** Itis an open-source web application that allows developers to create and share documents containing live code, equations, visualizations, and narrative text. It's widely used for data analysis, machine learning, and collaborative research.

**Excel**: It is a versatile spreadsheet application by Microsoft that enables users to organize, analyze, and visualize data using features like formulas, charts, and pivot tables. It's widely used for tasks ranging from simple data entry to complex financial modeling.

## 1.3 Project Introduction:

Football is a dynamic sport where real-time decision-making plays a crucial role in determining match outcomes. Among these decisions, player substitutions are pivotal for adapting to the ebb and flow of a game. Substitutions enable teams to respond to player fatigue, injuries, tactical shifts, or adverse game conditions. Traditionally, these decisions have been based on the intuition and experience of coaches, who rely on their understanding of players’ capabilities and the unfolding game context. However, in the modern era of sports, where data analytics and technology have become integral to performance optimization, a more structured and scientific approach is needed to make substitution decisions more effective and impactful.

The capstone project titled "Optimal Real-Time Substitution in Football" is designed to address this need by utilizing advanced data analytics and machine learning techniques. The project focuses on developing a system that can recommend optimal player substitutions during a match based on real-time data and contextual game information. By analyzing key factors such as player performance metrics, historical trends, team formations, and situational game dynamics, the system provides actionable insights to guide substitution strategies. The objective is to improve team performance and increase the likelihood of achieving desired match outcomes.

The core of the project lies in leveraging machine learning models, survival analysis, and sentiment analysis to predict the effectiveness of potential substitutions. These models consider both quantitative and qualitative aspects of the game, including a player’s fitness level, past performance under similar conditions, and even emotional stability during high-pressure situations. This integration of multiple data streams allows the system to generate comprehensive and accurate recommendations, offering a significant advantage over traditional, experience-based substitution methods.



### 

### Figure 1.1 Web Application

This project delivers a web application designed to assist football managers in making real-time substitution decisions. The platform analyzes live match data and predicts the top three substitutes based on performance metrics, enabling managers to optimize their strategies seamlessly. Powered by robust machine learning algorithms and authenticated data sources, the application ensures accurate and reliable recommendations with an intuitive user interface for quick decision-making.

By integrating advanced sports analytics, the project automates and optimizes substitution strategies, overcoming limitations of traditional methods. It highlights the transformative potential of combining technology and sports, paving the way for future innovations in predictive analytics to enhance team performance and strategic decision-making.

Moreover, the project has broader implications for sports analytics and management. As the use of big data continues to grow across various sports, systems like the one developed in this project highlight the potential of combining technology and sports to achieve significant performance improvements. By offering a practical, data-driven solution, this project sets the stage for future developments in predictive sports analytics, empowering managers, players, and teams to achieve excellence on and off the field.

# Chapter 2

# Literature Survey

## Journal and Conference Papers:

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date of Publication** | **Discussion** | **Gaps** |
| When do Risky Substitutions Improve Match Outcomes? Interpreting Causal Machine Learning Models for Insights *[Journal]* | 2023 | Risky substitutions generally have a non-negative impact on match outcomes, with manager characteristics like age and tenure influencing their effectiveness. The impact also depends on the team's strength and the timing of the substitution. | Research gaps include the focus on the substitutions, neglecting other risk-taking behaviors, and limited data scope. The study also assumes all relevant factors are included, suggesting the potential missing variables. |
| Football Player Substitution Analysis using NLP and Survival Analysis *[Conference]* | 2023 | The paper offers an innovative framework that combines Natural Language Processing and survival analysis to assess player substitutions, emphasizing both performance and mental health. Its practical implications for coaches and the potential for future research make it a valuable contribution to sports analytics. | Expanding sentiment data sources and including physiological metrics would also strengthen the model's effectiveness. |
| In-game behavior analysis of football players using machine learning techniques based on player statistics. *[Journal]* | 2021 | The study used machine learning techniques, including t-SNE, UMAP, and the RIPPER algorithm, to identify and differentiate football player positions based on technical-tactical statistics alone. It found behavioral differences between specific positions, such as strikers vs. wingers and defensive midfielders vs. central defenders, while also identifying and excluding data outliers. | Analyzing how player styles of play differ based on match results can aid scouting departments, while examining positional adjustments in response to managerial tactics provides insights into their impact on player performance. This dual approach enhances understanding of individual contributions within team dynamics. |

|  |  |  |  |
| --- | --- | --- | --- |
| A Data Science Approach to Football Team Player Selection *[Conference]* | 2020 | The paper discusses a data science approach to player selection, emphasizing the importance of considering both player ratings and age for team success. It also highlights how player nationality can influence a club's reputation and business profitability. | Utilizing the analytical approaches to support businesses in selecting players as brand ambassadors & forming new clubs, with a focus on cost reduction by selecting players with the right skill set and performance at nominal wages |
| Predictive analysis and modelling football results using machine learning approach for English Premier League *[Journal]* | 2019 | The gradient boosting model outperformed random forest, SVM, and naive Bayes, excelling in predicting draws, the least likely outcome. Its predictions were well-calibrated, with probabilities closely matching actual results. | The research suggests incorporating features like injury data, key player presence, and psychological factors, along with exploring social media data for predictions. It also recommends developing regression models to predict team goal outcomes. |
| Who Will Win It? An In-game Win Probability Model for Football *[Journal]* | 2019 | Existing win probability models from other sports struggle in football, prompting the introduction of a Bayesian statistical model that effectively predicts win, tie, and loss probabilities for the home team using eight variables. This model not only offers well-calibrated probabilities but also enhances fan experiences and aids in evaluating performance during critical moments. | Better justification of feature selection and handling rare events would improve the methodology. Additionally, practical real-time implementation and overfitting concerns need more attention. |

### Table 2.1 A list of Research Papers in the Domain

## Our Approach to Earlier Work:

|  |  |
| --- | --- |
| **Gaps in alternative applications** | **Our solutions to that** |
| Traditional substitution strategies lack real-time frameworks that consider dynamic match conditions, player fatigue, and tactical needs, limiting their effectiveness in optimizing outcomes. | Developed a real-time prediction system that evaluates substitution risks and benefits by integrating live match data, player performance metrics, and contextual game information. This enables managers to make informed decisions. |
| Player substitution strategies often neglect the psychological state of players and fail to utilize sentiment analysis tools to assess performance readiness, missing an opportunity for more nuanced decisions. | Integrated sentiment analysis using NLP models to analyze player interviews, social media data, and match reports. This helps incorporate emotional and psychological insights into substitution recommendations for better outcomes. |
| Current models for player behavior analysis do not adapt to fluctuating performance patterns during live matches, leading to outdated insights that fail to reflect real-time dynamics.  . | Implemented real-time tracking systems that analyze and update player behavioral patterns dynamically during matches, ensuring accurate substitution and tactical adjustments. |
| Existing approaches to player selection overemphasize historical performance data without adequately accounting for real-time statistics or situational context, reducing flexibility in decision-making. | Developed a data-driven system that integrates real-time player statistics and situational game contexts to provide dynamic and flexible player selection recommendations tailored to specific match scenarios. |
| Many predictive models focus only on pre-match data and fail to incorporate live match dynamics like ongoing events or tactical shifts, limiting their accuracy in forecasting outcomes. | Designed predictive models that continuously incorporate live match data, including team formations, live scores, and match events, ensuring real-time updates and improved accuracy in predictions. |
| Win probability models often overlook the influence of in-game tactical shifts, such as substitutions or changes in team formation, on outcome probabilities. | Developed a dynamic probability model that incorporates tactical shifts and substitution effects, continuously updating win probabilities based on live game events to support strategic decisions. |

**Table 2.2 Research Gaps and their corresponding solutions**

# Chapter 3 Problem Statement

## Project Scope

The scope of the project *"Optimal Real-Time Substitution in Football"* revolves around leveraging advanced analytics to transform player substitution strategies in football matches. The project aims to develop a robust system that integrates real-time data extraction, machine learning algorithms, and a user-friendly software platform to provide actionable insights for coaches and team managers. By analyzing key factors such as player performance, game context, and opposition strategies, the system ensures that substitution decisions are data-driven rather than relying solely on intuition or experience.

This project goes beyond traditional approaches by addressing critical gaps in current research. It incorporates real-time data processing, advanced modeling techniques, and a focus on practical, live-match applications. The scope also includes enhancing substitution strategies by factoring in variables like fatigue, game conditions, and player history, offering a comprehensive and interpretable decision-making tool. The project emphasizes reliability, scalability, and ethical use of data, ensuring robust performance across diverse scenarios.

In the long term, the project envisions further advancements, such as integrating training data for a holistic assessment of player potential and automating decision-making processes for seamless real-time application. By deploying this system on scalable platforms, it holds the potential to revolutionize sports management, making substitution strategies more effective and efficient across all levels of football.

### Features and Functionality:

* + 1. **Substitution Predictions:** Predicts the top three substitutes based on real-time performance metrics.
    2. **Live Data Integration**: Incorporates live match statistics for accurate and timely insights.
    3. **Interactive Interface**: Allows easy team and player selection with clear performance visuals.

## Project Assumptions:

During the planning and development of the **Optimal Real-Time Substitution in Football** web application, certain assumptions are made to streamline the project scope and ensure a smooth development process. These assumptions include:

* + 1. **Platform Support:** It will support major operating systems, including Windows, macOS, iOS, Android, and Linux. These platforms are chosen to ensure broad accessibility for football team managers, coaches, and players.
    2. **Internet Connectivity:** It is assumed that users will have access to stable internet connectivity, essential for real-time predictions and data synchronization. The application will utilize cloud services for storing player performance metrics and supporting real-time data analysis.
    3. **Security Requirements:** Users will have an expectation of privacy and security regarding their data, especially the performance metrics of players and match data. The application will implement standard security measures such as data encryption, secure authentication, and access controls to protect user data during analysis and storage.
    4. **Scalability Considerations:** The application will be designed with scalability in mind to support large datasets (e.g., multiple teams and players) and accommodate potential growth in user demand. The infrastructure and database management system will be designed to handle an increasing amount of real-time data and user interactions over time.

These assumptions will guide the development and planning phases of the project but should be validated and adjusted throughout the project lifecycle based on user feedback, evolving requirements, and technological advancements.

## Project Limitations:

* + 1. **Timeframe:** The project timeline should consider development, testing, and deployment phases, accounting for potential delays and contingencies.
    2. **Compatibility:** The web application should be compatible with a wide range of devices, operating systems, and versions.
    3. **Scalability:** The web application architecture should be designed to accommodate future growth and increasing user demand.

## Project Objectives

The project objectives for the Analysis and Predicting substitution are defined to ensure a clear direction and successful implementation. These objectives encompass the desired outcomes and goals for the application's development and deployment. The project objectives include:

* + 1. **Develop a user friendly web application:** The main objective of the project is to develop an easy to use web application using React, TailWind CSS, which would perform mathematical calculation and visualization.
    2. **Enhance User Experience:** Design an intuitive and user-friendly interface that simplifies using the website, ensuring that users, regardless of their technical expertise, can easily navigate the web application.
    3. **Implement Authentication Measures:** Employ an authentication system to ensure that no one other than company professionals access the website.
    4. **Data Visualisation:** The result of the mathematical calculations would be visualized using syncfusion react chart library. thus making it easier for users to analyze the output.
    5. **Ensure Scalability and Performance**: Design and develop the application with scalability in mind, ensuring it can handle increasing user demands..
    6. **Deliver Comprehensive Documentation and Support:** Provide user documentation, tutorials, and support resources that guide users on how to effectively use the application's features and make the most of its capabilities.

# Chapter 4 Project Requirements

## Resources :

* + 1. **Human resources:**

Our team, comprising three dedicated members, has collaborated seamlessly throughout the project to ensure its successful completion. Each team member has contributed their expertise and skills in their respective areas of specialization, collectively working towards the development of Scientific injection moulding web application.Additionally, we were fortunate to receive invaluable support and guidance from Prof. Rashmi Phalnikar, a renowned expert from MIT World Peace University. Her expertise and insights significantly contributed to the successful completion of the project. Moreover, Mr. Ajay Ghatpande, a seasoned professional serving as our Technical and Project Lead at SymphonyTech, provided valuable guidance and leadership, ensuring the project progressed smoothly and met all the desired objectives. Their mentorship greatly enhanced the quality and effectiveness of our work.

## Software Requirements:

* + - * ReactJs
      * Tailwind CSS
      * Jupyter Notebook
      * Excel
      * Xcode, or Visual Studio Code, depending on the operating system
      * An active internet connection for updates and storage services
      * A Wi-Fi network with a working WLAN method

**ReactJs:** ReactJS is a popular JavaScript library for building user interfaces. It simplifies the creation of interactive and dynamic web applications by allowing developers to design reusable components. React efficiently updates and renders

components as data changes, enhancing the user experience by ensuring seamless and responsive web applications.

**Tailwind CSS** is a highly customizable, utility-first CSS framework that enables developers to style applications directly within their HTML by using predefined classes. It eliminates the need to write custom CSS, promoting rapid development while maintaining design consistency.

**Visual Studio Code:** Visual Studio Code (VS Code) is a free, lightweight, cross platform source code editor developed by Microsoft. It is widely used by developers for various programming languages and provides a rich set of features and extensions that enhance productivity and streamline the development workflow.

**Jupyter Notebook:** Itis an open-source web application that allows developers to create and share documents containing live code, equations, visualizations, and narrative text. It's widely used for data analysis, machine learning, and collaborative research.

**Excel**: It is a versatile spreadsheet application by Microsoft that enables users to organize, analyze, and visualize data using features like formulas, charts, and pivot tables. It's widely used for tasks ranging from simple data entry to complex financial modeling.

## Hardware Requirements:

* + - * **Device Type:** The web application should be compatible with a range of devices, including desktop computers, laptops, smartphones, and tablets. Users should be able to access the web application from their preferred device, regardless of the operating system (e.g., Windows, macOS, iOS, Android) they are using.
      * **Processor and Memory:** The device should have a capable processor and sufficient memory to run the application smoothly. For example, PC’s have a decent processor and a recommended minimum of 8 GB RAM or more for optimal performance.
      * **Display:** The device's display should provide a satisfactory user experience for interacting with the application's user interface. It should have a suitable resolution, size.
      * **Network Connectivity:** The device should have access to a reliable internet connection to facilitate data sharing and synchronization. This can be through Wi-Fi, mobile data, or any other means of connecting to the internet. The connection should have sufficient bandwidth to support file transfers without significant latency or interruptions.
      * **Security Measures:** From an end-user perspective, the device should have basic security measures in place, such as password protection, screen lock, and up-to-date antivirus software. These measures help safeguard the user's data and protect against unauthorized access.
      * **Operating System:** To ensure compatibility with the application, it is essential to meet the minimum system requirements for different operating systems. For desktop devices, Windows 7 or higher is required for Windows operating systems, while Ubuntu devices should have version 16 or higher.

**Note**: These are minimum requirements, and a better hardware and software setup may result in a better user experience. Additionally, certain features may not be available on older operating systems or devices with limited hardware.

## Requirements Rationale:

|  |  |  |
| --- | --- | --- |
| **Requirements** | **Rationale** | **Benefit** |
| **Cross-Platform Compatibility:** | Users often work with multiple devices  and operating systems. By ensuring cross platform compatibility, the application  allows users to access the website | Users can easily access website without restrictions, enhancing productivity and  convenience. |
|  | seamlessly across different devices, |  |
|  | irrespective of their preferred platform. |  |
| **Security Measures:** | Data security and privacy are paramount | Users can quickly understand and use the  application, reducing the learning curve and improving overall user  satisfaction. |
|  | when sharing files. Implementing robust |
|  | security measures, such as encryption |
|  | protocols, secure authentication, and access |
|  | controls, helps protect user data and |
|  | ensures confidential and secure file |
|  | transfers. |
| **Synchronization in Real-Time** | synchronization features allow users to  access and work on data which is already stored and can be fetched on real-time . Users can access previous  calculations and stored data. | Users can access previous calculations and stored data. |

**Table 4.1 The Requirements, Rationale and Benefits**

## Risk Management :

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk Factor** | **Description** | **Impact** | **Mitigation Strategy** |
| **Platform Compatibility Issues** | Compatibility issues may arise when  accessing on different operating systems/platforms. | **High**  Delays in loading of the website. | Conduct thorough testing on various platforms. |
| **Security Vulnerabilities** | Inadequate security measures may  expose user data to unauthorized access or breaches. | **High**  Compromised user data or privacy | Implement strong  authentication mechanisms, and regular security audits. Stay updated with the latest security practices and promptly address any identified vulnerabilities. |
| **User Interface Usability** | Poor user interface design may lead to confusion or difficulties in navigating the  application. | **High**  User frustration and lower  adoption | Conduct user testing, incorporate user feedback, and follow established  UI/UX design principles to create an intuitive and userfriendly interface.  Provide user documentation and tutorials to guide users through the application. |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Integration Challenges** | Integrating with  external services or APIs (e.g., cloud  storage providers) may pose technical challenges or dependencies. | **High**  Limited functionality or delays in implementation | Thoroughly research and plan integration points, maintain clear  communication with  external service providers,  and have backup solutions or alternatives in case of integration difficulties. |

**Table 4.2 Risk Management**

## Functional Requirements from Organization :

* + 1. **Technical Requirements:**

1. **User interface:** The application should provide a user-friendly and intuitive interface that allows users to easily perform the tasks described above. The interface should be designed for mobile devices, such as smartphones and tablets.
2. **Security:** The application should provide adequate security mechanisms to protect the privacy and integrity. This could include authentication mechanism and network traffic, and protection against common attacks such as eavesdropping, tampering, and denial-of-service.
3. **Compatibility:** The application should be compatible with a wide range of devices and operating systems, to ensure that users can easily connect and share files with each other regardless of their hardware or software platform.
4. **Performance Analysis:** The application should be optimized for performance and efficiency, to minimize the network latency, and battery consumption while able to fetch the real-time data and compare it with data in dataset. This could be achieved through techniques such as data compression, caching, and adaptive bitrate streaming.

**4.4.2 Non-Technical Requirements:**

1. The system should deliver fast and accurate result with reduced latency
2. The platform should require user authentication before granting access to services to prevent unauthorized usage
3. The platform should provide users with an option for using stored data and see previous results.
4. The platform should be user-friendly and intuitive, allowing for easy navigation, it should be accessible across different devices and platforms
5. The platform should offer customers the ability to provide feedback and raise issues if they encounter challenges while using the systems.
6. Regular updates and maintenance should be provided to ensure the platform remains stable, secure, and reliable at all times.

## Chapter 5

System Analysis Proposed Architecture

## Design Considerations:

Several crucial design considerations were taken into account during the development of the Scientific Injection Moulding web application. These considerations were essential to ensure the application's functionality, usability, security, and scalability.

* + 1. **User Experience (UX):** The user interface (UI) was designed with a focus on simplicity, intuitiveness, and ease of use. Attention was given to creating a seamless and visually appealing user experience, allowing users to navigate through the application effortlessly. Clear and concise instructions were provided to guide users through the entry process, ensuring a positive and efficient interaction.
    2. **Security and Privacy:** React Js Framework provides with a robust security framework to safeguard sensitive user information.User authentication mechanisms were integrated to prevent unauthorized access and ensure the authenticity of users connected to the network.
    3. **Performance and Scalability:** The application was designed to handle high network traffic efficiently. Considerations were made to optimize network bandwidth and minimize latency. Scalability was prioritized, enabling the application to accommodate a growing number of users and without compromising performance.
    4. **Compatibility:** Compatibility across different devices and operating systems was a significant consideration. The application was developed to be compatible with Android and iOS mobile devices, as well as Windows and Ubuntu desktop platforms, meeting the diverse needs of users.
    5. **Error Handling and Recovery:** Robust error handling and recovery mechanisms were implemented to handle various exceptions and ensure graceful recovery from failures. Adequate error messages were provided to assist users in troubleshooting and resolving issues effectively.

By considering these design considerations throughout the development process, the resulting Football Substitution web application was equipped with a user-friendly interface,

stringent security measures, optimal performance, compatibility with multiple platforms, and a supportive environment for users.

## Assumptions and Dependencies:

It is assumed that the users of the Football Substitution web application possess basic knowledge of Football Substitution concepts and are familiar with using mobile and desktop devices. Additionally, it is assumed that the users have access to stable internet connectivity to enable seamless file transfers and network communication.

### Operating System Dependencies:

The application's compatibility and functionality are dependent on the specific operating systems it is designed for. For Android devices, the application relies on the Android operating system (version 5 or higher), while for iPhone devices, it requires the iOS operating system (version 11 or higher). Similarly, the application requires Windows 7 or higher for Windows devices and Ubuntu version 16 or higher for Ubuntu devices.

### Network Connectivity:

The application is dependent on a stable and reliable internet connection for the seamless transfer of files between connected users. The availability of network connectivity, along with sufficient bandwidth, is crucial for the application's proper functioning.

### Security Libraries and Frameworks:

The implementation of security features, such as authentication mechanisms, relies on the availability and proper integration of security libraries and frameworks. Dependencies on these libraries and frameworks must be satisfied to ensure the desired security measures are effectively implemented.

### Compliance and Regulations:

The application's adherence to applicable legal and regulatory requirements, such as data privacy and security regulations, may be dependent on external factors. Compliance with these regulations may require adherence to specific protocols or guidelines, which should be taken into consideration during development and deployment.

## General Constraints:

### Time Constraints:

The project development and completion are subject to time constraints. The project timeline, spanning from August to October, imposes limitations on the available time for each phase, including planning, development, testing, and release. Efficient time management and adherence to the project schedule are crucial to ensure timely delivery.

### Resource Constraints:

The availability of resources, including human resources, software, hardware, and infrastructure, may impose constraints on the project. Limited resources may impact the development process, requiring effective resource allocation and management to meet project objectives within the available resources.

### Technology Constraints:

The selection of technologies and frameworks is subject to certain constraints. The project must work within the limitations and capabilities of the chosen technologies, ensuring compatibility and interoperability across different platforms and devices. Compatibility with specific operating systems, hardware requirements, and security considerations may impose constraints on the choice of technologies.

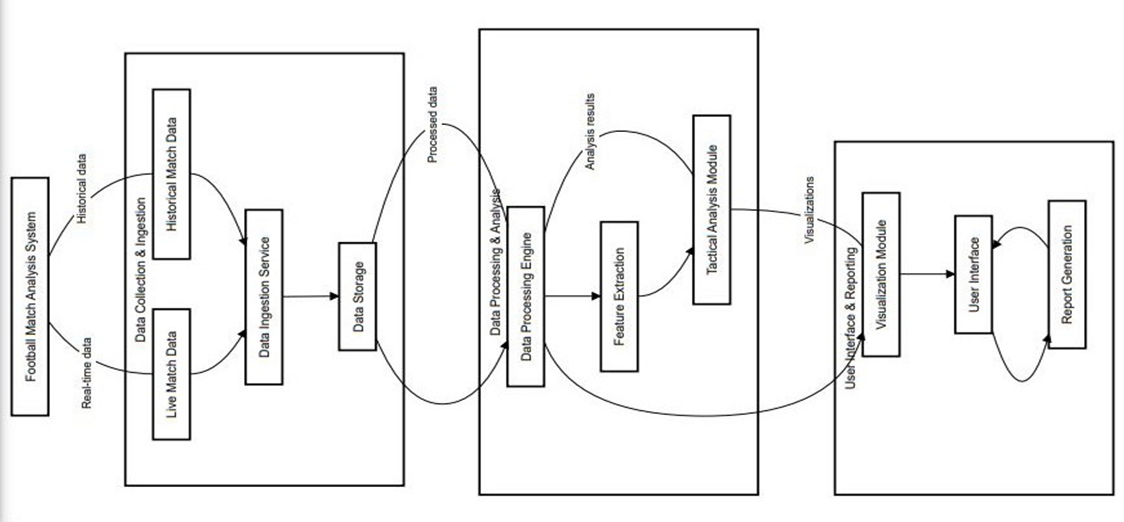
### Scalability Constraints:

The application must be designed with scalability in mind to accommodate future growth and increased user demands. Consideration should be given to the scalability of the infrastructure, network, and database systems, allowing the application to handle a growing number of users, larger file sizes, and increased data traffic effectively.

### User Acceptance Constraints:

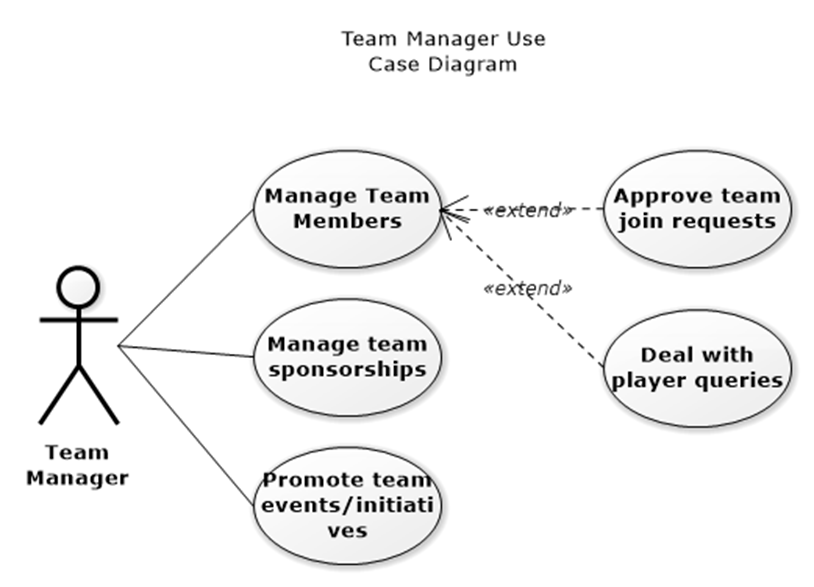
The success of the project relies on user acceptance and adoption of the web application. Meeting user expectations, providing an intuitive user interface, and delivering a seamless user experience is essential to ensure user satisfaction and acceptance. Gathering feedback, conducting usability testing, and iterating based on user input are necessary to address any constraints related to user acceptance.

## System Architecture

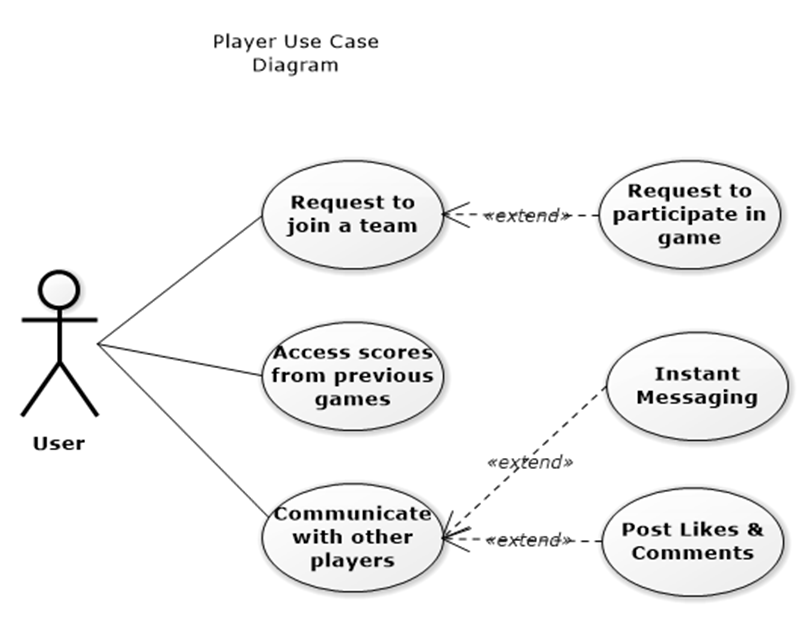


## Use Case Diagram:

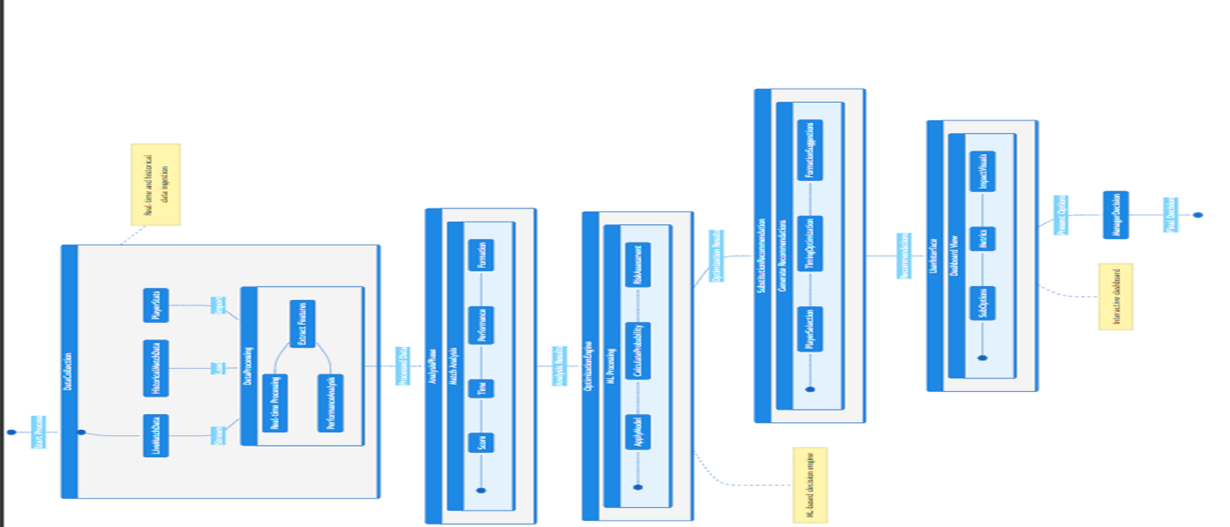
**5.5.1 Team Manager Use Case Diagram**

****

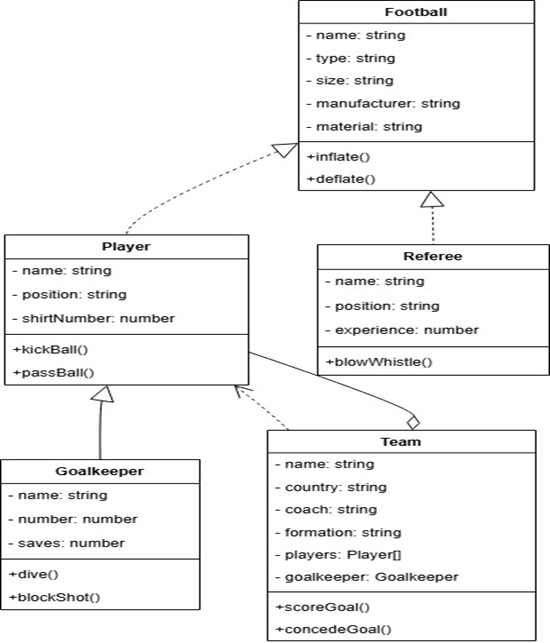
**5.5.2 Player Use Case Diagram**

****

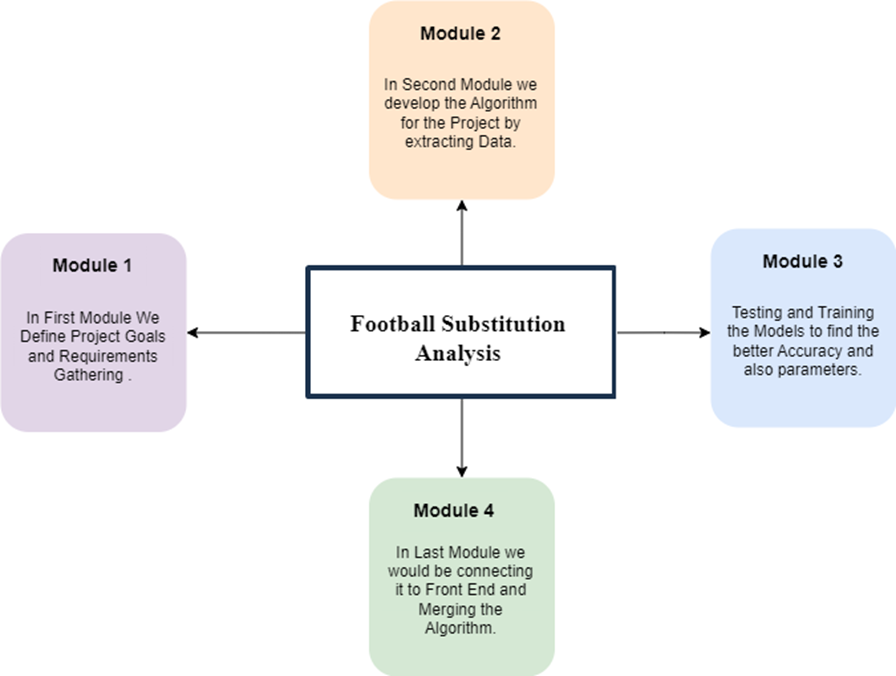
## Activity Diagram:

****

## Class Diagram:

****

* 1. **Modules of the Project**



The project was structured into four main modules, each serving a specific purpose.

### Module 1: Defining Goals and Requirements

In the first module, the project begins with clearly defining the goals and objectives of the Football Substitution Analysis. This stage includes gathering requirements, understanding the scope, and outlining the steps to ensure the project aligns with the stakeholders' expectations.

### Module 2: Algorithm Development

The second module focuses on extracting relevant data and developing the algorithm. This includes data cleaning, preprocessing, and initial implementation of models that serve as the foundation for substitution analysis.

### Module 3: Model Training and Testing

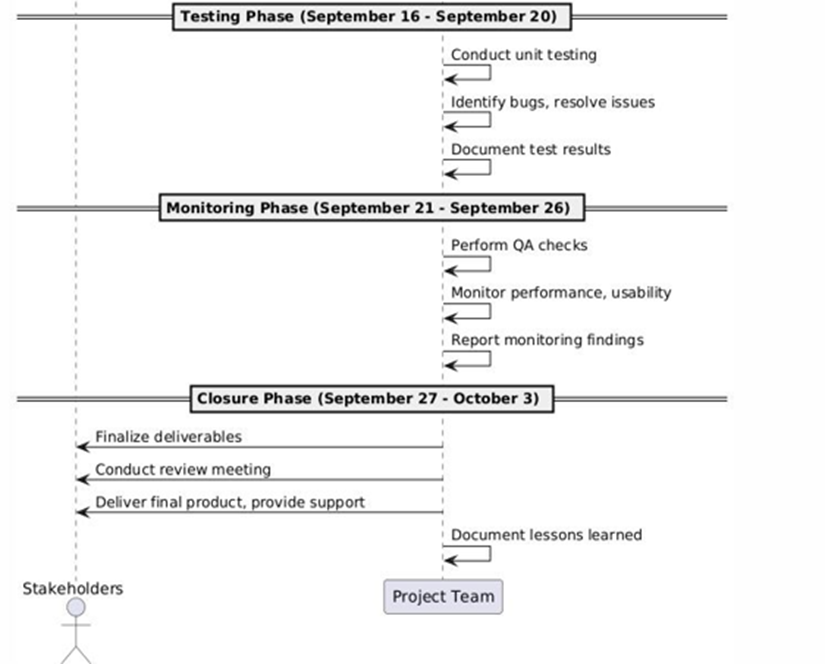
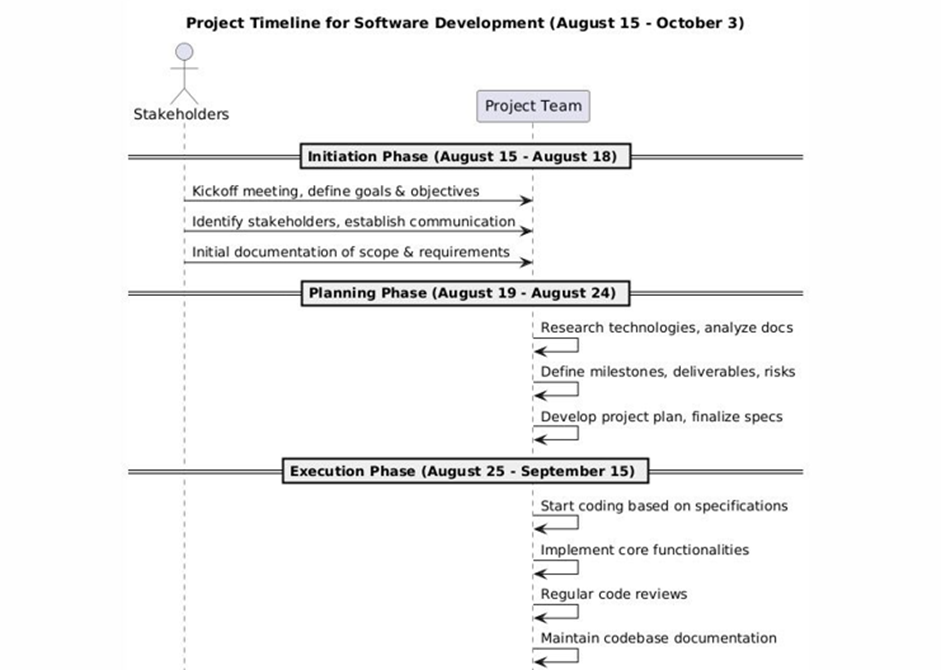
In third module module, the models are rigorously tested and trained to achieve higher accuracy and optimized performance. The goal is to refine the parameters and improve the reliability of predictions related to football substitutions.

### Module 4: Integration and Deployment

The final module involves connecting the developed algorithm to the front-end interface. This includes merging the backend logic with the user interface, ensuring seamless integration, and preparing the solution for end-user deployment.

# Chapter 6

# Project Timeline



### Fig 6.1 Project Timeline

**Different Phases:**

**Phase P1 : 15th August-18th August**

**Phase P2 : 19th August-24th August**

**Phase P3 : 25th August-15th September**

**Phase P4 : 16th September-20th September**

**Phase P5: 21st September- 26th September**

**Phase P6: 27th September- 03rd October**

The project is divided into six phases, spanning from August to October, each serving a specific purpose to ensure a systematic and successful completion of the project.

### Phase 1: Initiation Phase (August 15 - August 18):

This phase begins with a kickoff meeting to define project goals and objectives. Stakeholders are identified, communication channels are established, and the initial documentation of scope and requirements is prepared.

### Phase 2: Planning Phase (August 19 - August 24):

During this phase, the team researches relevant technologies and analyzes documentation. Milestones, deliverables, and risks are defined. The project plan is developed, and the specifications are finalized for the next phase.

### Phase 3: Execution Phase (August 25 - September 15):

In this phase, coding begins based on the finalized specifications. The core functionalities of the software are implemented. Regular code reviews are conducted, and the team ensures the codebase is well-documented for consistency and maintenance.

### Phase 4: Testing Phase (September 16 - September 20):

In the testing phase, the team conducts unit testing to verify that individual components and modules function as expected. Bugs and issues identified during testing are resolved systematically. Test results are meticulously documented to ensure transparency and provide insights for further improvements. This phase focuses on quality assurance to ensure a stable and reliable product.

### Phase 5: Monitoring Phase (September 21 - September 26):

The monitoring phase emphasizes comprehensive quality assurance and performance checks. The team evaluates system performance, usability, and compatibility under different conditions. Monitoring tools and manual testing methods are employed to ensure that the software adheres to predefined standards. Findings are documented and communicated to stakeholders for necessary adjustments before final delivery.

### Phase 6: Closure Phase (September 27 - October 3) :

This Final phase marks the conclusion of the project. The team finalizes and delivers the product to the stakeholders, ensuring all deliverables are met. A formal review meeting is conducted to assess the outcomes, including project successes and challenges. Support for the transition phase is provided if required. Lastly, the lessons learned throughout the project are documented to guide future initiatives and improve processes.

By following this structured timeline and progressing through each phase diligently, the project team successfully delivered a Optimal Real Time Substitution in Footbal**l** Web application by October.

# Chapter 7 Implementation

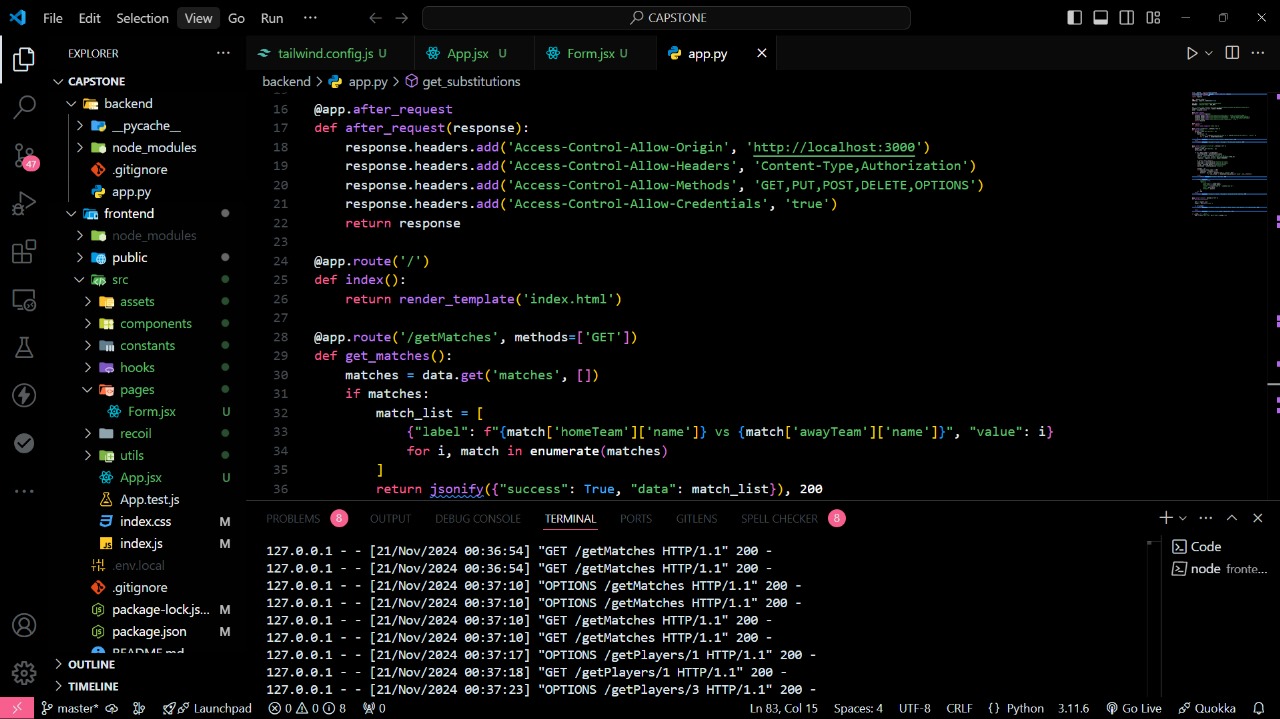
## 7.1 Methodology:

In modern football, player substitutions play a crucial role in influencing match outcomes and team performance. To analyze and optimize substitution strategies, this project delves into a data-driven approach combining data collection, statistical analysis, and visualization techniques. Historical match data and player statistics were collected using Python libraries such as BeautifulSoup and Scrapy. The dataset underwent extensive cleaning to ensure accuracy and consistency, enabling meaningful insights into player performance.

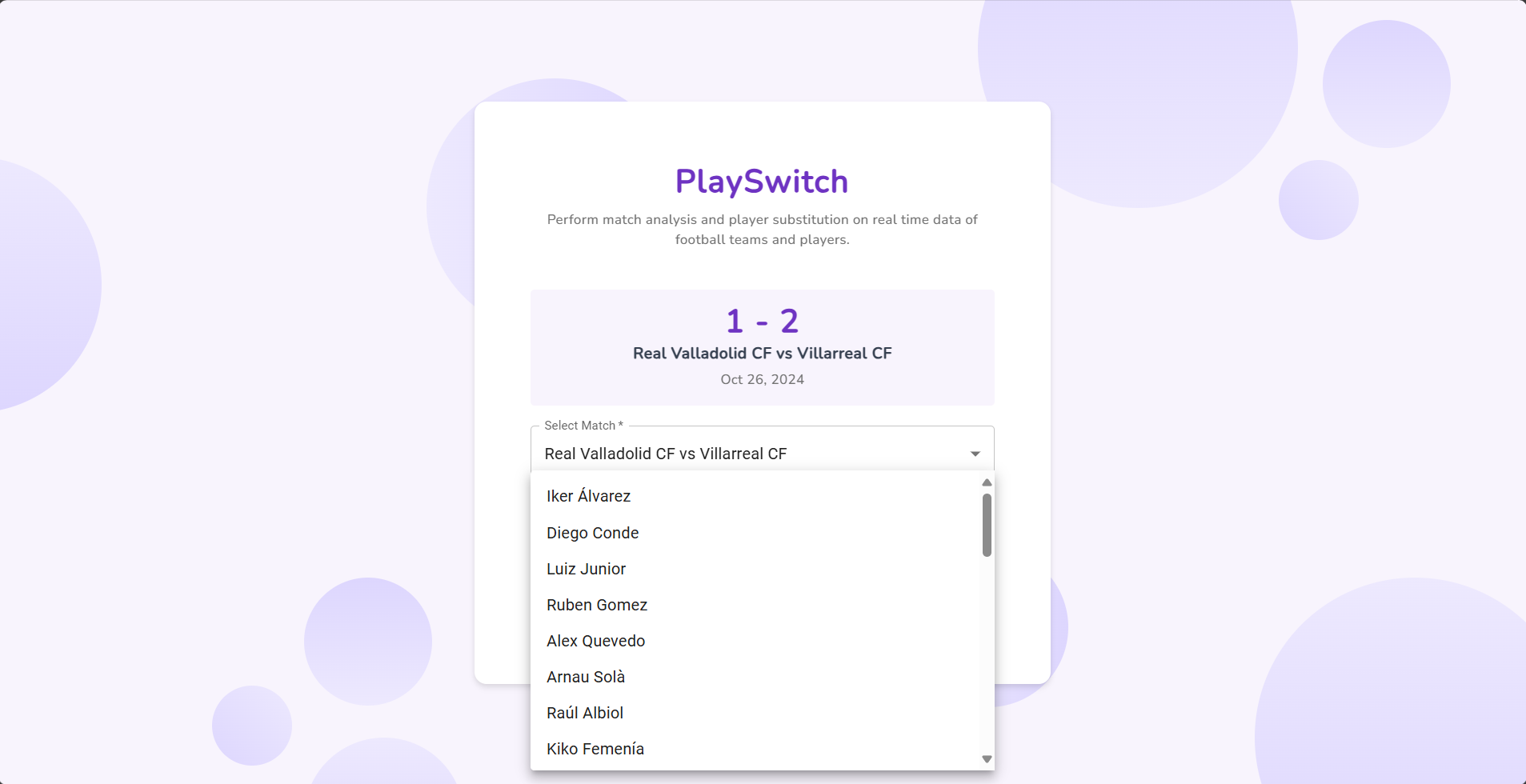
The analysis involved statistical comparisons between starting players and substitutes to evaluate performance differences. A specialized framework was designed to assess risks associated with substitutions based on match contexts and player metrics. Furthermore, a recommendation system was implemented, leveraging player statistics and situational factors to suggest optimal substitution strategies.

To enhance interpretability, visualizations created using Matplotlib and Seaborn highlighted trends in player performance, substitution impacts, and match outcomes, offering actionable insights for decision-making in competitive football scenarios. This project aims to provide data-backed solutions to optimize substitution strategies and improve team success rates. The concise methodology is :

1. **Project Setup:** Start by setting up your project with React for the frontend, Flask for the backend, and integrate real-time data fetching from football.org for live match data. Ensure proper installation of dependencies and establish the project structure.
2. **Frontend Development (React)**: Develop the React components to incorporate real-time data fetching and implement the player substitution analysis interface. Allow users to input team configurations and game scenarios, then display real-time substitution recommendations based on player performance metrics and match context.
3. **Backend Development (Flask):** Create API endpoints using Flask to handle data from the front end. The backend should be designed to fetch real-time match data from trusted sources, process the data, run predictive algorithms, and generate the best substitution recommendations.
4. **Data Collection:** Utilize Python libraries like BeautifulSoup and Scrapy to scrape historical match data, player profiles, and team performance from trusted sources such as football.org. The collected data includes player statistics, match reports, and performance data across multiple games, all of which are crucial for supporting the player substitution analysis and making informed decisions based on past performance.
5. **Perform Backend Calculations:** Utilize Flask's capabilities to implement calculations on the backend. Fetch real-time data, process it with predictive algorithms, and generate output data used to suggest optimal substitutions based on player performance, match context, and statistical analysis.
6. **Recommendation System:** Implement an algorithm that suggests the best substitute players based on historical performance data, average skills, and performance under match pressure, thereby helping managers make data-driven decisions in real time.



### Fig 7.1 Code snippet

****

**Fig 7.2 Web Application glimpse**

# Chapter 8

# Performance Evaluation and Testing

## Performance Evaluation

The performance of our predictive model was rigorously evaluated using a suite of statistical metrics that measure accuracy and error in predicting player scores. These metrics provide a comprehensive assessment of the model's effectiveness, as summarized below:

### R-squared (R²):

Achieving an R² score of 0.9519 indicates that approximately 95.19% of the variance in the player scores is explained by our model, which underscores a very strong fit to the data.

* + 1. **Mean Squared Error (MSE) :**

The MSE of 0.0509 suggests that the average squared difference between the predicted and actual values is minimal. This low value reflects the model's effectiveness in minimizing large errors, which is crucial for accurate score predictions.

### Mean Absolute Error (MAE):

The MAE of 0.1699 indicates that, on average, predictions deviate from actual values by a margin of only 0.17 points, demonstrating the model's precision in making close-to-accurate predictions

**8.2 Execution Time**

With an execution time of 0.354 seconds, the model demonstrates both efficiency and scalability, allowing for near real-time predictions on new data.

It is important that the recommendation model successfully highlights players who may require substitution due to lower scores in certain attributes, like goalkeeping for field players or attacking for defenders. The system, therefore, offers a granular approach to tactical decisions, providing managers with insights to optimize team performance dynamically during a match.

# Chapter 9

# Results and Analysis

## Test Results Discussion

In analyzing the specialized web application tailored for football substitution analysis, the integration of React, Flask, and real-time data fetching APIs from trusted sources like football.org has resulted in a robust platform capable of providing actionable insights during live matches. The React frontend, designed with a dynamic and user-friendly interface, seamlessly integrates a predictive model to recommend optimal substitutions in real-time based on player performance metrics.

The Flask backend plays a critical role in processing these inputs, leveraging its capabilities to fetch, analyze, and process live data. It interfaces with APIs to collect real-time match information and historical statistics, which are then utilized by the predictive model to generate substitution recommendations. This backend-driven architecture ensures precise and timely insights, fostering a smooth collaboration between the frontend and backend components.

Using the platform, football managers can input team configurations and game scenarios, enabling the system to compute and recommend the top three substitutes. The frontend dynamically displays these recommendations, allowing managers to visualize performance trends and substitution impacts. The integration of real-time fetching APIs and advanced data analysis ensures both accuracy in decision-making and ease of use.

The collaborative features of Postman further enhance the application's reliability by facilitating the seamless testing of API endpoints. This ensures efficient communication between the frontend and backend, maintaining the smooth flow of data and computations.

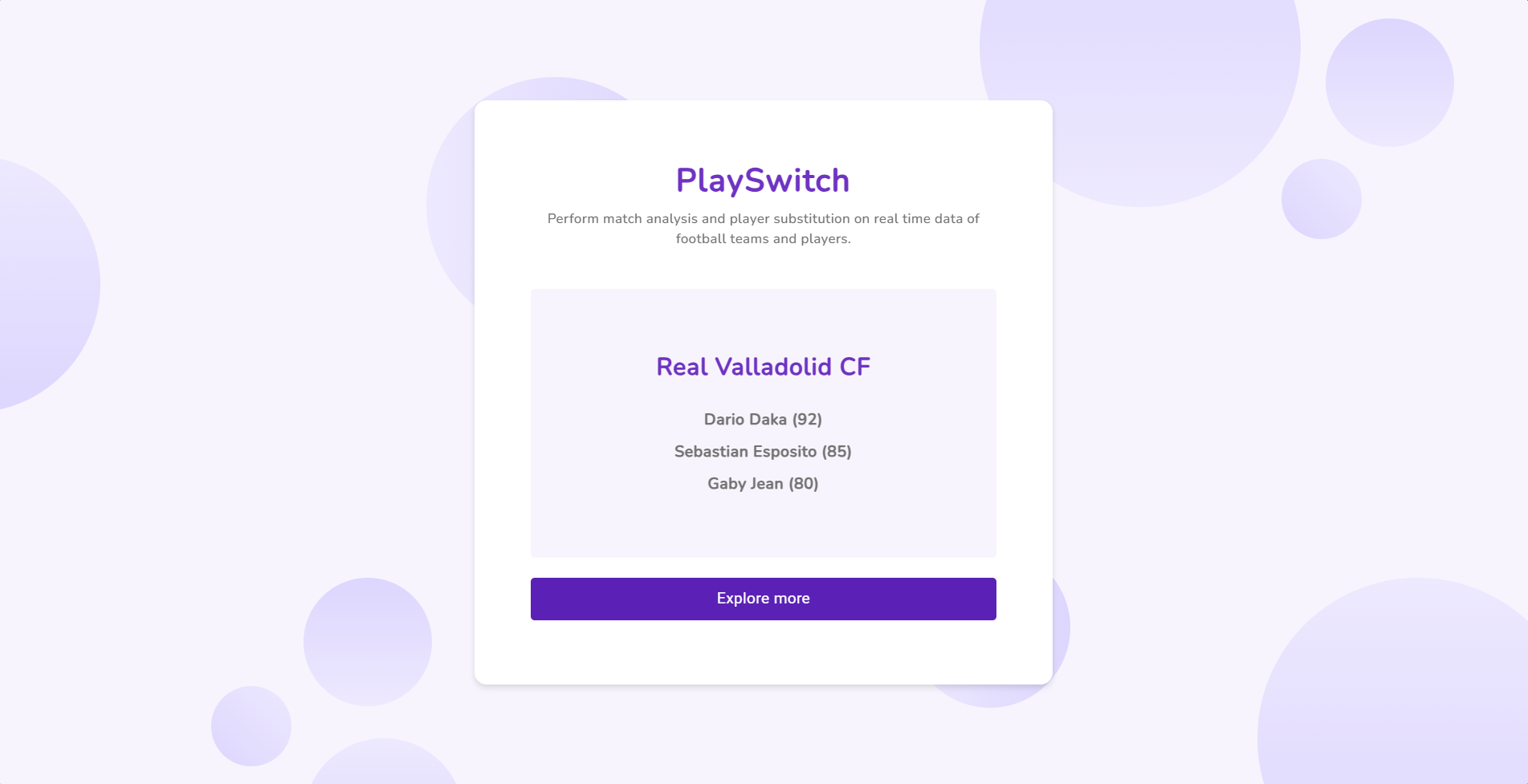
In summary, the web application effectively harnesses React, Flask, predictive analytics, and real-time data fetching to create a cutting-edge platform for football substitution analysis. Its precision, real-time capabilities, and user-centric design position it as a valuable tool for football managers and analysts, revolutionizing decision-making in live matches.

## Test Results Analysis

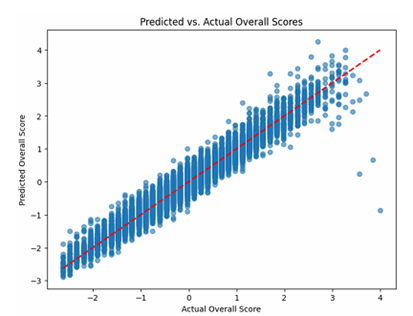
The evaluation of the test results for the football substitution analysis web application highlights its impressive performance across various aspects. Thorough testing was conducted to validate the functionality and reliability of the integrated technologies, encompassing React, Flask, real-time data fetching APIs, and Postman. The React frontend showcased a user-friendly interface, adept at managing team configurations and game scenarios, ensuring a seamless and intuitive user experience. The integration of real-time data fetching APIs confirmed the system’s accuracy in retrieving and processing live match information, while the backend effectively analyzed and processed data to generate optimal substitution recommendations.

By leveraging player attributes, performance metrics, and substitution recommendations, we provided actionable insights into optimizing team performance. The developed models achieved high accuracy (R² = 0.95), underscoring their reliability in predicting player performance and guiding tactical decisions.

This research underscores the value of integrating analytical models into sports management, enhancing decision-making processes and supporting teams in achieving better outcomes on and off the field. Future enhancements could include further real-time data integration and deeper exploration of causal relationships to refine predictions and recommendations even further.



**Fig 9.1 Result of the Web Application**

****

**Fig 9.2 Accuracy of the Model Trained Graph**

# Chapter 10 Deployment Strategies

The deployment phase of the football substitution analysis web application is a pivotal step in making the platform accessible to users. The process seamlessly integrates React for the frontend, Flask for the backend, and APIs to fetch real-time data from trusted sources such as football.org. This ensures the application functions efficiently in a live environment while delivering accurate substitution recommendations.

For the frontend, React was used to develop a dynamic and user-friendly interface. The React application undergoes bundling and optimization for production, generating static files suitable for deployment. These files are then hosted on a platform to ensure seamless accessibility. The frontend also integrates a predictive model, prepared using historical comparisons and real-time data, to recommend optimal football substitutions during live matches.

On the backend, the Flask application is configured for production, with settings optimized for security, performance, and scalability. APIs are employed to fetch real-time data from football.org, a trusted source widely used across the football community. The backend processes this data and interfaces with the predictive model to generate actionable insights for substitution decisions.

The deployment process ensures smooth communication between the frontend and backend through well-structured API endpoints. Continuous monitoring tools are integrated to track the application's performance, identify potential issues, and optimize responsiveness. This robust deployment strategy guarantees that football managers and analysts can rely on the platform for real-time, data-driven substitution recommendations during live matches.

In conclusion, the deployment process entails harmonizing the frontend and backend components, ensuring the seamless integration of technologies in the live environment. This holistic approach guarantees the stability and efficiency of the scientific injection molding website, poised to serve the intended user base with precision and reliability.

# Applications

The Football Substitution Recommendation website, developed with React, Flask Framework, with real-time data-fetching showcases remarkable versatility for applications in the domains of Sports.

### Football Team Management:

The website excels in executing intricate mathematical calculations tailored for substitution . Assisting coaches and managers during live matches to make optimal substitution decisions. Identifying players who can bring the most value based on current game conditions and historical data. Using player analytics to develop strategies and prepare for specific opponents.

### Player Development and Talent Scouting:

Researchers benefit from the website's capacity for experimentation and data analysis. The system analyzes individual performance trends to identify areas where players can improve, aiding in skill development and overall growth. Simultaneously, it evaluates the potential of substitute players using performance metrics, helping scouts identify promising new talent for future opportunities. This dual approach supports both player development and effective talent acquisition.

### Commercial and Sponsorship Opportunities :

The application can be offered as a service to football clubs, providing custom insights to optimize team performance and decision-making. Additionally, it presents opportunities for brand partnerships, where collaborations with analytics companies or sports brands can help integrate and promote the platform, enhancing its reach and value

### Quality Control and Assurance:

The website's capability to conduct calculations and generate visual representations is pivotal for quality control in manufacturing. Manufacturers leverage the platform to scrutinize and verify molding parameters, ensuring compliance with rigorous quality standards and maintaining consistent product quality.

### Efficient Data-driven Decision Making:

Empowering users with data-driven decision-making capabilities, the website enables engineers and manufacturers to make informed choices. Real-time visualizations and calculations provided by the platform contribute to more efficient molding processes, reducing waste, and optimizing resource utilization.

In essence, By addressing these diverse applications, the web application showcases its versatility in improving decision-making, optimizing performance, and transforming football analytics and management practices. Its integration of cutting-edge technologies positions it as an invaluable asset for professionals and researchers striving to advance injection molding and enhance overall manufacturing efficiency.

# Conclusion

In conclusion, This project successfully demonstrates the potential of data-driven decision-making in football management through advanced statistical analysis and machine learning techniques. By leveraging player attributes, performance metrics, and substitution recommendations.

Furthermore, the substitution recommendation system highlights underperforming players based on multiple key attributes, offering a systematic approach to improving team efficiency during matches. The methodologies employed, including feature importance analysis, correlation exploration, and substitution logic, provide a robust framework that can be adapted for various sports analytics applications.

This research underscores the value of integrating analytical models into sports management, enhancing decision-making processes and supporting teams in achieving better outcomes on and off the field. Future enhancements could include real-time data integration and deeper exploration of causal relationships to further refine predictions and recommendations.

Looking ahead, the success of this project serves as a foundation for future innovations and improvements. Sustained collaboration, ongoing testing, and refinements will be crucial to align the platform with evolving industry requirements and technological progress. The scientific injection molding website not only meets current industry demands but also anticipates future needs, driving progress in both academic and industrial spheres of injection molding through its thoughtful integration of cutting-edge technologies.

# Future Prospects of the Project

The future prospects of the Football substitution Analysis, developed are highly promising and diverse, encompassing technological advancements, industry transformation, and ongoing growth.

1. **Integration with Real-Time Match Data**: By linking the Kaplan-Meier analysis with live match data such as distance covered, heart rate, and current game conditions, the system can predict when a player’s performance might decline. This allows the coach to proactively make substitutions to maintain team performance.
2. **Efficiency in Substitutions**: The system can suggest the optimal substitution timing, reducing delays in decision-making, and minimizing the risk of keeping underperforming or fatigued players on the field for too long
3. **Integration with Sentiment Analysis**: By combining Kaplan-Meier data with sentiment analysis, the system can factor in psychological and social influences on player performance. For instance, a dip in sentiment might correlate with poor performance or lower confidence, influencing substitution decisions.
4. **Data-Driven Bench Management:** Coaches can rely on automated insights to manage the bench effectively, ensuring that substitutions not only address performance dips but also align with tactical goals based on player-specific endurance predictions.
5. **Predictive Risk Alerts**: For injury prevention, Kaplan-Meier automation can assess risk thresholds, providing alerts if a player is approaching a critical point where their likelihood of injury increases significantly.

In conclusion, the future outlook is characterized by its commitment to technological evolution, significant industry impact, continuous contributions to research and innovation, expanded educational offerings, and the facilitation of global collaboration.

# References

[1] O. G. Ali and E. Yilmaz, "When do Risky Substitutions Improve Match Outcomes? Interpreting Causal Machine Learning Models for Insights," in IEEE Access, vol. 11, pp. 131351-131361, 2023, doi: 10.1109/ACCESS.2023.3333878.

[2] A. Bera, S. Joshi and A. M. Nair, "Football Player Substitution Analysis using NLP and Survival Analysis," 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2023, pp. 753-757, doi: 10.1109/ICSSIT55814.2023.10061118.

[3] García-Aliaga, A., Marquina, M., Coterón, J., Rodríguez-González, A., & Luengo-Sánchez, S. (2021). In-game behaviour analysis of football players using machine learning techniques based on player statistics. International Journal of Sports Science & Coaching, 16(1), 148-157. https://doi.org/10.1177/1747954120959762

[4] P. Rajesh, Bharadwaj, M. Alam and M. Tahernezhadi, "A Data Science Approach to Football Team Player Selection," 2020 IEEE International Conference on Electro Information Technology (EIT), Chicago, IL, USA, 2020, pp. 175-183, doi: 10.1109/EIT48999.2020.9208331.

[5] Baboota, R., & Kaur, H. (2019). Predictive analysis and modelling football results using machine learning approach for English Premier League. International Journal of Forecasting, 35(2), 741-755.

[6] Robberechts, P., Van Haaren, J., & Davis, J. (2019). Who will win it? An in-game win probability model for football. arXiv preprint arXiv:1906.05029, 4.

[7] Bransen, L., Van Haaren, J., & van de Velden, M. (2019). Measuring soccer players’ contributions to chance creation by valuing their passes. Journal of Quantitative Analysis in Sports, 15(2), 97-116.

[8] Barnes, C., Archer, D. T., Hogg, B., Bush, M., & Bradley, P. (2014). The evolution of physical and technical performance parameters in the English Premier League. International journal of sports medicine, 35(13), 1095-1100.

[9] Del Corral, J., Barros, C. P., & Prieto-Rodriguez, J. (2008). The determinants of soccer player substitutions: A survival analysis of the Spanish soccer league. Journal of Sports Economics, 9(2), 160-172.

[10] Gregson, W., Drust, B., Atkinson, G., & Salvo, V. D. (2010). Match-to-match variability of high-speed activities in premier league soccer. International journal of sports medicine, 237-242.

[11] Hughes, M., & Franks, I. (2005). Analysis of passing sequences, shots and goals in soccer. Journal of sports sciences, 23(5), 509-514.

[12] Mackenzie, R., & Cushion, C. (2013). Performance analysis in football: A critical review and implications for future research. Journal of sports sciences, 31(6), 639-676.

[13] Liu, H., Gomez, M. Á., Lago-Peñas, C., & Sampaio, J. (2015). Match statistics related to winning in the group stage of 2014 Brazil FIFA World Cup. Journal of sports sciences, 33(12), 1205-1213.

### PART-B

INDIVIDUAL CONTRIBUTION

**Problem Statement:** Optimal Real Time Substitution in Football

**Name of Student:** Malay Jignesh Shah

**Module Contribution:** Web-Scrapping and Fetching Real-time Data using API

**Module Objective:** Web scraping is a powerful technique used to extract data from websites that do not provide an API or direct access to their data. In this project, Python libraries like BeautifulSoup and Scrapy are used to scrape historical match data, player profiles, and team performance statistics from trusted football data sources such as football.org. BeautifulSoup is utilized for parsing HTML data and extracting relevant match reports, player statistics, and performance metrics, while Scrapy is employed for more advanced and large-scale data scraping, allowing for faster and more efficient extraction of structured data from web pages. The data collected includes details such as player performance metrics, match results, substitution timings, and more, which form the foundation for analyzing player substitutions and team performance.

An API is developed to provide actionable insights during live football matches, the system needs access to real-time data. This is accomplished by developing an API that fetches live match data from trusted sources like football.org, which provides up-to-date statistics such as live match scores, player stats, and substitution events. Using Flask Framework on the backend, API endpoints are created to interact with real-time data, ensuring that live game data is retrieved, processed, and used to offer immediate substitution recommendations. The API handles the retrieval of player statistics and match conditions in real time, ensuring seamless integration with the frontend. These APIs are essential for making quick, data-driven decisions, providing football managers with optimal player substitution suggestions based on live match data and performance metrics.

**Problem Statement:** Optimal Real Time Substitution in Football

**Name of Student:** Sayal Goyal

**Module Contribution:** Created Front-end and Visualized the Data

**Module Objective:** The front end of the scientific injection molding project, made with React and boosted by the Football Substitution React library, provides a user-friendly and advanced interface. React, a powerful JavaScript library, is the main tool for creating dynamic and responsive interfaces. The project's front end is carefully designed to handle scientific inputs easily, making it straightforward for users to interact with. Using React's component-based approach, the interface is well-organized and easy to maintain, allowing for updates and changes without any hassle.

The Football Substitution React library adds various user interface components and modern data visualization tools, including advanced charts that simplify scientific data. The user interface is made to be clear and simple, ensuring easy navigation for professionals, researchers, and students using the platform. Essentially, the front end of the project highlights the smooth blend of React and Model, providing an attractive, interactive, and effective interface tailored for users involved in scientific injection molding processes.

These visualizations help in understanding complex molding parameters, improving decision-making processes. The backend ensures that the data is not only processed and analyzed accurately but also delivered in an intuitive format that enhances the user experience.

**Problem Statement:** Optimal Real Time Substitution in Football

**Name of Student:** Kanishk Singhania

**Module Contribution:** Create the Back-end and Trained the Model

**Module Objective:** The backend of the scientific injection molding website, built with Flask Framework, efficiently handles, processes, and analyzes the input data. Leveraging Flask robust capabilities, it manages the reception of API requests, processes incoming scientific parameters, and executes the complex mathematical computations essential for injection molding calculations. By utilizing Flask Framework, the backend ensures smooth interaction between the frontend and backend, enabling seamless data exchange for accurate results.

Once the model is trained and deployed on the backend, it processes real-time inputs and generates predictions based on the user’s scientific parameters. The backend then sends the processed results to the frontend for visualization.

With a focus on security and scalability, the backend architecture guarantees the confidentiality and reliability of sensitive data. Moreover, it plays a crucial role in maintaining data consistency through strategic database migrations, ensuring accurate storage and retrieval of scientific parameters input by users. The collaboration between the backend and frontend components is pivotal to the website's success, with the Flask backend serving as the bedrock for the platform's efficiency, reliability, and overall performance in handling complex scientific computations.

**Project to Outcome Mapping**

1. Web Scrapping , Developing API and Fetching the Real Time Data
2. Create the web pages using react , Data Visualization
3. Data Handling, processing and analyzing , Model Training

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.No | PRN | Student Name | Individual Project Student  Specific Objective | Learning  Outcomes mapped  ( To be filled by Guide ) |
| 1. | 1032212062 | Malay Jignesh Shah | Web Scrapping the Data & Developing API and Fetching the Real Time Data. |  |
| 2. | 1032212395 | Sayal Goyal | Designing the UI of the website and then implementing the design. |  |
| 3. | 1032211295 | Kanishk Singhania | Handling the Data using a database. Managing the input and output. |  |