CrickVision

Project Team

Tabidah Usmani 22I-2070
Sara Zahid 22I-1861
Amna Javaid 22I-2025

Session 2022-2026

Supervised by

Dr. Muhammad Nouman Noor

Co-Supervised by

Ms. Sadia Saad

Department of Data Science

National University of Computer and Emerging Sciences
Islamabad, Pakistan

June, 2026

Contents

List of Figures

List of Tables

Chapter 1

Introduction [AS PER SCOPE DOCUMENT]

This document serves as a comprehensive proposal for CrickVision. It is a web-based application designed to revolutionize cricket strategy, talent identification, and team man- agement. It outlines the problem facing the Pakistan cricket ecosystem and provides de- tails on how CrickVision provides an innovative data-driven solution. The proposal fo- cuses on its objectives, its core functionalities, technical approach, and the impact that it is expected to have on a large scale in modern cricket. The background of this system lies in the growing complexity and increasing data in contemporary cricket, where traditional intuition-based decision making is increasingly insufficient to maintain competitiveness. CrickVision continues to bridge this gap by integrating advanced data analytics and ma- chine learning to deliver actionable prescriptive insights..

1.1 Existing Solutions

The sphere of the cricket performance analysis has undergone significant development as it switched to the manual scorekeeping to complex, data-driven systems. The solutions that are currently available can be generally divided into three main categories: advanced statistical analysis systems, physical performance tracking systems, and integrated data/video analysis systems. Critical evaluation of these systems shows that they have some strengths and weaknesses, including the lack of one, unified, prescriptive and service such that it covers every aspect of team management, including the identification of talent, all the way up to the financial auction strategy.

1.1.1 State-of-the-Art in Cricket Analytics

CricViz stands as a leading example of an advanced statistical analysis platform. It is renowned for its extensive cricket database and unique predictive models, such as **WinViz** (win predictor) and **PitchViz** (pitch characteristic assessment) (1). CricViz provides deep, context-specific statistical insights, moving beyond traditional averages to quantify player performance in various match situations (2).

Catapult represents the state-of-the-art in physical performance tracking and workload management. Their solutions primarily use GPS and inertial sensors to monitor player movement, speed, and load

during training and matches (3). Catapult's strength lies in its ability to provide objective data on player welfare, helping to prevent injuries and manage fatigue (4).

Stats Perform (Opta) provides a fully-packaged data-driven solution, such as a universal data base of cricket and recruitment profiling solutions (5). Opta specializes in providing detailed and structured event data that can be used to make strategic decisions and recruitment process. It often combines its data with video analysis software and allows coaches to put statistical information into perspective with visual data features of visual evidence contextualization (6).

1.1.2 Gaps

Although these solutions may be powerful, they typically work separately, thus establishing a disconnected ecosystem of team management, which is a fundamental problem CrickVision is supposed to solve. (7).

1. **Lack of Integration and Prescriptive Analytics:** Statistical solutions like CricViz are highly effective at the descriptive and predictive (describing past events and making possible predictions) analysis, but they often lack the prescriptive element the built-in solution that provides specific, actionable advice across all modules (scouting, strategy, welfare, and finance). This means that a coach would have to synthesize data which comes out of a statistical tool, a physical tracking tool and video analysis tool and come up with one decision manually most of the time.

2. **Limited Focus on Financial/Auction Strategy:** There are a number of solutions that offer specialized, integrated franchise/auction management at present, something that is a fundamental functional need of professional T20 leagues. The ability to simulate auctions and maximize financial investments on squads as suggested in the CrickVision Module 3 is a great market gap.

3. **Absence of Integrated Talent Identification:** Other platforms have some recruitment tools, but the feature of having an integrated, data-driven domestic talent identifier based on machine-learning models that analyze lower-league performance data, described in Module 1 of CrickVision, is not a standard, centralized feature of most of the commercial offerings.

Table 1.1: Comparison of Existing Solutions

System Name	System Overview	System Limitations
CricViz (1)	Advanced statistical analysis platform with a vast cricket database and unique predictive models (e.g., WinViz, PitchViz). Provides deep, context-specific statistical insights for tactical planning and broadcast.	Primarily focused on statistical and predictive analysis; lacks integrated physical workload management and dedicated financial/auction simulation tools. Insights are descriptive, not prescriptive.
Catapult (3)	Physical performance tracking system using GPS and inertial sensors to monitor player movement, speed, and workload. Excellent for injury prevention, fatigue management, and objective physical performance metrics.	Primarily focused on physical and physiological data; lacks the deep, context-specific statistical and machine learning models for strategic and tactical decision-making (e.g., role suitability, opponent scouting).
Stats Perform (Opta) (5)	Global sports data provider offering a comprehensive database and profiling tools for recruitment and strategy. Provides detailed, structured event data often integrated with video analysis for contextualization.	Data is often raw or requires significant manual processing by analysts to translate into actionable, prescriptive insights. The focus is on data provision rather than a fully integrated, end-to-end team management platform.

1.2 Problem Statement

Cricket coaches, selectors, and franchise managers currently operate within a broken ecosystem, relying heavily on basic statistical tools, disparate data sources, and subjec- tive intuition for taking high-stake decisions. All of this leads to the problem of inefficient strategies, biased player selections, and missed opportunities for deserving players. The lack of integrated workload management tools contributes to preventable injuries, sidelin- ing key players for extended periods and costing teams valuable performance. There is no single, integrated platform that effectively uses machine learning to translate complex, raw data into clear, actionable insights from talent identification and strategic planning to player welfare and financial management.The absence of these resource makes it to rely on human interpretation of vast datasets, which is slow, prone to cognitive biases and hinders a teams ability to achieve a certain performance.

1.3 Scope

CrickVision is precisely defined to deliver a comprehensive, web-based solution for cricket management, focusing on transforming raw data into meaningful insights. A core com- ponent of the scope involves pre-match opponent analysis, designed to generate tactical blueprints and identify exploitable weaknesses. A significant aspect of the project is the auction simulation tool which will help the franchise managers to optimize squad and financial investments. Explicitly out of scope for this project are live ball-by-ball data streaming, wearable biometric fatigue tracking, direct financial auction transactions, so- cial media integration and video analysis tools. This application does not include captain selection as it just focuses on giving recommendations and justifications.

1.4 Modules

1.4.1 Module 1

This module focuses on player evaluation, discovery and the justification of selection decisions. It utilizes data to identify high potential players and provides valid reasoning for squad composition.

1. ML Data prep: This involves gathering and preprocessing datasets from different sources to prepare them for machine learning models.
2. Automated Scouting Report: It automatically creates comprehensive, data-driven scouting reports, highlighting opponent strengths, weaknesses and key tactical insights thereby saving significant manual effort
3. Data Driven Domestic Talent Identifier: This involves building machine learning models that analyze various performance to identify promising young talent.

1.4.2 Module 2

This module provides tools for pre-match planning and in-game performance analysis, assisting coaching staff in optimizing team strategies and creating Playing XI.

1. Role Suitability Predictor: It involves creating a predictive model that recommends players best suited for specific roles based on their skill sets, historical performance in similar situations and prevailing match conditions.
2. Virtual Team Selector: It is based on user input, generates data-backed pre-match strategies. including an optimal playing XI and tactical game plans.
3. Selection Justification Generator: It focuses on concise and data-supported summaries that provide reasoning for a players inlcusion or exclusion from squad, fostering transparency and accountability in selection processes.

1.4.3 Module 3

It supports long-term investment, team management and financial planning, offering tools for player welfare and strategic auction preparation.

1. Bowling Partnership Synergy Score: It involves developing a unique analytical module that quantifies the combined effectiveness of bowling pairs, assessing their ability to build pressure and take wickets.

2. Form and Fatigue Tracker: It focuses on creating a visual that tracks players recent performance trends alongside their physical workload.

3. Auction Simulator: It allows franchise managers to model various auction scenarios on historical PSL data and optimize bidding strategies to build a balanced team.

1.4.4 Module 4

It is a dashboard that acts as the central brain, integrating insights from all other modules to provide holistic, actionable advice and a single point of access.

1. Executive Dashboard: It is the central brain that synthesizes insights from all other modules, offering integrated, context-aware recommendations and alerts to guide strategic decision making.

2. UI Build and Merge: It involves integrating frontend and backend to ensure seamless data flow between features.

3. System Check and Tuning: It is a comprehensive testing phase including unit tests for ML models, integration tests for interactions between modules and performance evaluation.

1.5 Work Division

Name	**Registration**	**Responsibility / Module / Feature**
Tabidah Usmani	22i-2070	Components Integration, Module 2 - Feat 1, Module 2 - Feat 3, Module 3 - Feat 2
Sara Zahid	22i-1861	System check and tuning, Module 1 - Feat 1, Module 1 - Feat 2, Module 3 - Feat 1
Amna Javaid	22i-2025	ML prep, Module 1 - Feat 3, Module 2 - Feat 2, Module 4 - Feat 1

Table 1.2: Work Division

Chapter 2

Project Requirements [AS PER FYP1 MID REPORT]

This section outlines the necessary requirements for the successful completion of the project. Project requirements can be divided into two main categories: functional requirements, which describe the system's core operations, and non-functional requirements, which specify performance, security, and usability standards.

2.1 Use-case/Event Response Table/Storyboarding

2.1.1 Use Case Diagram

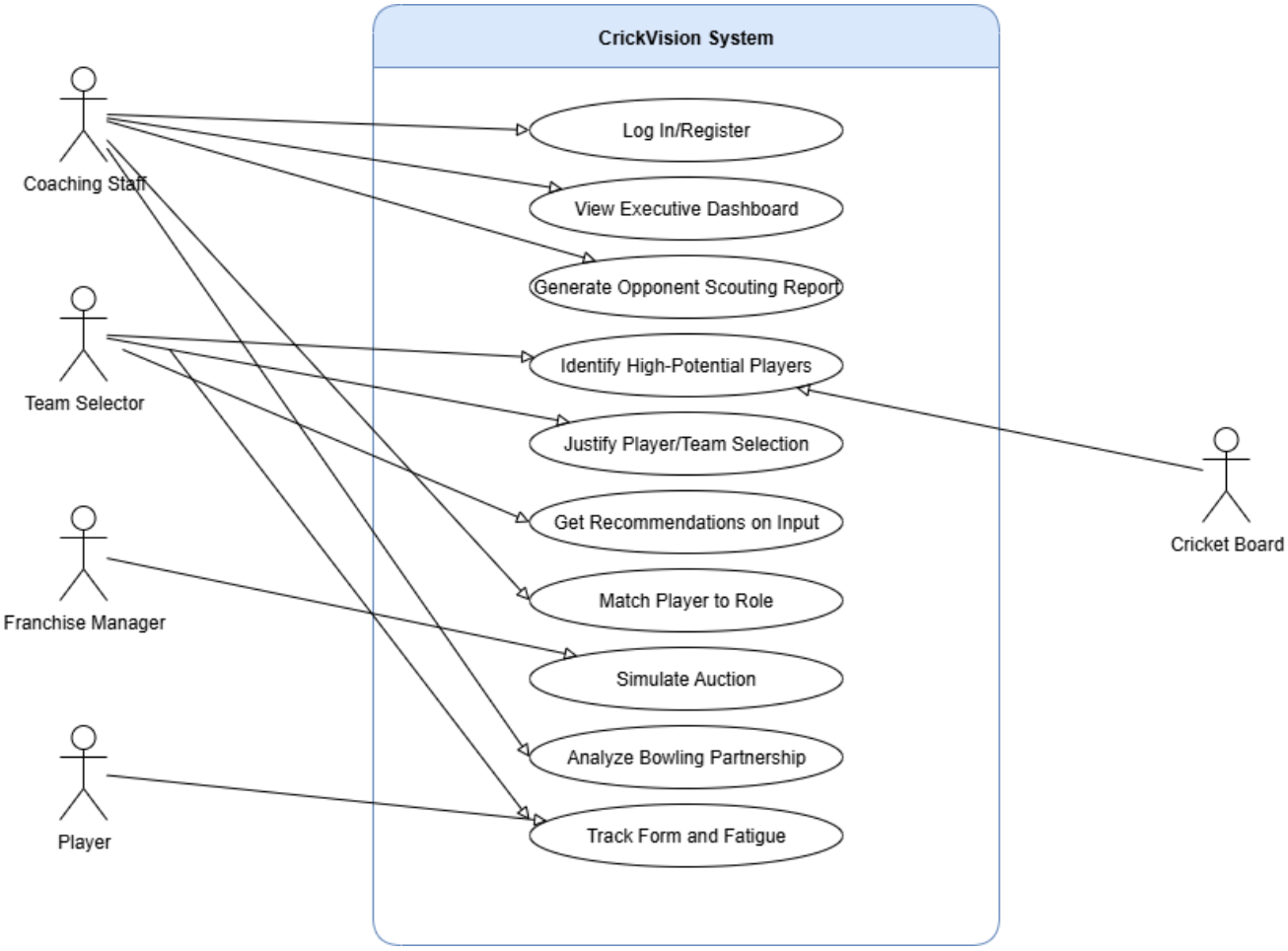

Figure 2.1: Use Case

2.1.2 High-Level Use Cases

Use Case ID	UC1
Use Case	Login/Register
Actors	Coaching Staff, Team Selector, Franchise Manager, Player
Description	It authenticates to use the system or registers for a new account

Table 2.1: UC1

Use Case ID	UC2
Use Case	View Executive Dashboard
Actors	Coaching Staff, Team Selector, Franchise Manager
Description	A high-level user views a consolidated, visual summary of key team and player performance metrics.

Table 2.2: UC2

Use Case ID	UC3
Use Case	Generate Opponent Scouting Report
Actors	Coaching Staff
Description	The coaching staff generates a detailed tactical report on an opposing team, highlighting their strengths and weaknesses.

Table 2.3: UC3

Use Case ID	UC4
Use Case	Identify High-Potential Players
Actors	Team Selector, Cricket Board
Description	They use the system's predictive models to find promising young talent which is then shared with the cricket board.

Table 2.4: UC4

Use Case ID	UC5
Use Case	Justify Player/Team Selection
Actors	Team Selector
Description	Generates a data-driven report that provides a rationaled for their squad choices

Table 2.5: UC5

Use Case ID	UC6
Use Case	Get Recommendations on Input
Actors	Coaching Staff, Team Selector, Franchise Manager
Description	Any priviledged user recevies contextual, AI-driven suggestions from the system to aid in decision-making.

Table 2.6: UC6

Use Case ID	UC7
Use Case	Match Player to Role
Actors	Coaching Staff, Team Selector
Description	The actor finds the best player for a specific role based on data/historical performance.

Table 2.7: UC7

Use Case ID	UC8
Use Case	Simulate Auction
Actors	Franchise Manager
Description	The actor runs a mock player auction to test bidding strategies and practice building a balanced team.

Table 2.8: UC8

Use Case ID	UC9
Use Case	Analyze Bowling Partnership
Actors	Coaching Staff
Description	The actor analyzes the combined performance data of two bowlers

Table 2.9: UC9

Use Case ID	UC10
Use Case	Track Form and Fatigue
Actors	Coaching Staff, Player
Description	The actor monitors a players performance trends and physical workload to manage performance and prevent injury.

Table 2.10: UC10

2.1.3 Extended Use Cases

Use Case ID	UC1
Use Case	Log In/Register
Preconditions	The actor must have a device with access to the CrickVision web app.
Main Flow	• The actor navigates to the login page. • The actor will enter their credentials and system will validate it. • The system establishes a session and directs them to their personalized dashboard. • In case of register, the actor fills out the form with required details. • The system creates a new user account and sends a verification email and verifies it.
Postconditions	Successful, active session with the system.
Alternate Flows	1. If credentials are incorrect, the system displays an 'Invalid username or password' error. 2. If the email is already registered then system displays an 'Email address is already in use' message.

Table 2.11: Extended: UC1

Use Case ID	UC2
Use Case	View Executive Dashboard
Preconditions	The actor must be logged into the system.
Main Flow	• The actor logs in and is directed to the Executive Dashboard. • The system fetches real-time and historical data and aggregates key metrics. • The system displays the information using interactive charts, graphs and summary cards. • The actor can apply basic filters.
Postconditions	The actor is informed of the current strategic and performance status.
Alternate Flows	1. If a data source is lagging, the system shows a message that the data is not real time.

Table 2.12: Extended: UC2

Use Case ID	UC3
Use Case	Generate Opponent Scouting Report
Preconditions	The coach is logged in and historical data for the opponent exists in the database..
Main Flow	• The coach selects the 'Scouting Report' and opponent team. • Then it applies optional filters. • The coach initiates the report generation and the system compiles a report detailing the opponent's strengths, weaknesses, key stats etc. • The report is displayed, which the coach can save or export.
Postconditions	A scouting report is successfully generated and saved.
Alternate Flows	1. If there is not enough data, the system displays a notification of 'Insufficient data'.

Table 2.13: Extended: UC3

Use Case ID	UC4
Use Case	Identify High-Potential Players
Preconditions	the selector is logged in and historical data is present in the system
Main Flow	• The selector opens the feature and defines search criteria. • The system processes the data using ML models to identify players with high potential. • The system presents a ranked list of players with a potential score and key stats. • The selector reviews the list and exports a shortlist to share with the cricket board
Postconditions	A data-backed list of high potential players.
Alternate Flows	1. If filters are too restrictive then system will display that no matching players were found for this.

Table 2.14: Extended: UC4

Use Case ID	UC5
Use Case	Justify Player/Team Selection
Preconditions	A squad is already been created or selected within the system.
Main Flow	• The selector chooses 'Justify Selection' option for a specific squad. • The system automatically compiles relevant data for the selected players. • The system generates a report containing visualizations and text that highlights the reasoning behind each selection and may also included bias analysis. • The selector can add manual notes and export the justification report.
Postconditions	A justification report is created that can be shared with management.
Alternate Flows	1. If a selected player has limited data available then insufficient data message will be displayed.

Table 2.15: Extended: UC5

Use Case ID	UC6
Use Case	Get Recommendations on Input
Preconditions	The actor is performing another task e.g. selecting a team.
Main Flow	• While using a feature, the actor provides input. • The actor clicks a 'get recommendation' button and system provides suggestions. • The system analyzes the input in the current context.
Postconditions	The actor recevies a set of actionable recommendations to improve their decision.
Alternate Flows	1. If the system cannot find a high-confidence recommendation then it will display a message.

Table 2.16: Extended: UC6

Use Case ID	UC7
Use Case	Match Player to Role
Preconditions	The actor is logged in and system contains a comprehensive database of players and their stats.
Main Flow	• The actor selects the feature and defines the desired role. • The actor specifies conditions. • The system analyzes its database, ranking players based on their suitability for the defined role and conditions • The system returns a ranked list of the most suitable players with a compatibility score
Postconditions	A data-driven list of players best matching the specified role is displayed.
Alternate Flows	1. If no players are a strong match for the specified criteria, the system suggests broadening the filters.

Table 2.17: Extended: UC7

Use Case ID	UC8
Use Case	Simulate Auction
Preconditions	The manager is logged in and player auction pool and rules are defined.
Main Flow	• The manager starts the auction simulator. • The manager confirms budget and squad rules. • The bidding starts, competing with other franchises. • When the manager wins a player, the budget and pool is updated accordingly. • The process is repeated until squad is full • The system displays final summary.
Postconditions	A simulated auction result and squad are generated for review.
Alternate Flows	1. If a bid exceeds the available budget, the system displays error message.

Table 2.18: Extended: UC8

Use Case ID	UC9
Use Case	Analyzing Bowling Partnership
Preconditions	The coach is logged in and ball by ball data is available in the system/
Main Flow	• The coach selects the feature. • The coach selects 2 bowlers from their team and applies filters. • The system processes all historical dara where the two bowlers bowled in under the specified conditions. • The system displays a combined stats.
Postconditions	A detailed statistical analysis of the bowling partnership is displayed.
Alternate Flows	1. If the selected bowlers have never bowled together then display that 'no partnership data is available'.

Table 2.19: Extended: UC9

Use Case ID	UC10
Use Case	Track Form and Fatigue
Preconditions	The actor is logged in and player workload and performance data is tracked.
Main Flow	• The coach selects the feature and chooses a player. • The system retrieves and processes the player's historical workload and performance data. • The system displays visualizations showing performance trends and workload metrics.
Postconditions	The actor has a clear, visual understanding of the player's current form and fatigue level.
Alternate Flows	1. If workload data is missing for some matches, the analysis is displayed with a note indicating the data is incomplete.

Table 2.20: Extended: UC10

2.2 Functional Requirements

This section describes the functional requirements for CrickVision. The requirements are grouped by module, which is divided into specific functional requirements that describe the features and functionalities of the system.

2.2.1 Module 1: Talent Identification and Scouting

- FR-1.1: The system shall allow a user to select an opposing team or an individual player to analyze.
- FR-1.2: The system shall generate a comprehensive report detailing the selected entity's statistical strengths, weaknesses and recent performance trends based on historical data.
- FR-1.3: The system shall generate a ranked list of high-potential 'undiscovered' players, justifying their ranking with key performance indicators and predictive models.
- FR-1.4: The system shall process and analyze performance data from specified cricket leagues.

2.2.2 Module 2: Strategic Planning and Team Selection

- FR-2.1: The system shall allow a user to input match conditions and select a squad of available players.
- FR-2.2: The system shall generate a data-backed, optimal Playing XI from the squad tailored to the specified conditions.
- FR-2.3: The system shall allow a user to define a specific tactical role.
- FR-2.4: The system shall recommend the most suitable players from a squad to fill that role.
- FR-2.5: For any player recommended or included in a team, the system shall provide a concise, data-driven summary explaining the reasoning.

2.2.3 Module 3: Performance and Welfare Managements

- FR-3.1: The system shall display a trend graph of a selected player's performance over a specified period.
- FR-3.2: The system shall display a corresponding visualization of the player's recent workload to help coaches and managers assess potential fatigue and injury risk.
- FR-3.3: The system shall allow a user to select two bowlers for comparison.
- FR-3.4: The system shall calculate and display a Synergy score and relevant statistics that quantify their effectiveness.

2.2.4 Module 4: Franchise and Auction Management

- FR-4.1: The system shall allow a user to set auction parameters.
- FR-4.2: The system shall provide an interactive interface to simulate bidding on players from a draft pool.

2.2.5 Module 5: Centralized Intelligence

- FR-5.1: The system shall present a single, consolidated view of the most critical, context-aware insights from all other modules.
- FR-5.2: The dashboard shall include alerts for key events.
- FR-5.3: All dashboard widgets and summaries shall be interactive, allowing users to navigate directly to the detailed feature with a single click.

2.3 Non-Functional Requirements

This section focuses on the non-functional requirements in terms of the quality attributes of CrickVision.

2.3.1 Reliability

- REL-1.1: The primary objective for our project is to achieve a Mean Time Between Failure of at least 120 hours of continuous process. This target ensures that the platform remains consistently available and dependable for its key stakeholders during critical periods like pre-match planning and strategizing.
- REL-1.2: A failure could lead to significant operational setbacks, and such events would undermine user trust and the strategic advantage the platform is designed to provide.
- REL-1.3: In the event of failure, detailed logs will facilitate the root cause analysis. This prioritizes swift recovery through automated process restarts for minor issues.

2.3.2 Usability

- USE-2.1: The system shall allow a new user to successfully generate a scouting report for an opposing team within 10 minutes of their first use without any external support.
- USE-2.2: The user shall be able to access the Executive Dashboard from any other module or view with a single click.

- USE-2.3: The system shall provide clear, descriptive error messages if a user enters invalid data into a field.
- USE-2.4: All data visualizations must use a color-blind accessible palette and include clear labels and legends, ensuring that insights are understandable to all users regardless of visual ability.
- USE 2.5: When a user generates a team, the system shall automatically save it so that even if it is accidentally closed, it can be restored with a single interaction upon returning.

2.3.3 Performance

- PER-3.1: The report generation for a single team or player must be completed and displayed to the user in under 15 seconds from the initial request.
- PER-3.2: The system must support up to 50 concurrent users, accessing and running queries on the platform simultaneously without a degradation in performance.
- PER-3.3: The role suitability predictor shall return player recommendation within 3 seconds after all the parameters have been defined.
- PER-3.4: All database queries retrieving player statistics or historical data for display in the UI must be executed in under 2 seconds to avoid noticeable lag.

2.3.4 Security

- SEC-4.1: The system shall be designed to resist unauthorized access attempts from an attacker with intermediate technical skills and common hacking tools.
- SEC-4.2: The system must ensure that users can only access data and functionality relevant to their own franchise or team.
- SEC-4.3: The system must be robust enough to prevent a malicious actor from altering critical data, such as player performance statistics or historical match results, without detection.
- SEC-4.4: All data must be protected from unauthorized viewing.

Chapter 3

System Overview [AS PER FYP1 MID REPORT]

CrickVision is a comprehensive, web-based analytics platform designed to revolutionize cricket strategy by transforming raw data into actionable intelligence. Built using a Model-View-Controller architecture, it integrates modern technologies including machine learning and data visualization to support team selection, performance evaluation etc.

3.1 Architectural Design

3.1.1 Module 1: Finding and Evaluating Players

- This module transforms ball by ball massive data into meaningful insights using data analytics and machine learning model.
- It does not only generates team of high potential player but also generates reasoning behind selecting every player by generating detail scouting reports.
- Each player's individual performances using past matches data such as statistics of previous matches, performance across different formats is fed into this module to create scouting report of each player.

3.1.2 Module 2: Match Preparation and Team Selection

- This module's purpose is to create a well balanced team with appropriate roles and to ensure that each player's role is also backed up by solid proof.
- It adds contextual match setup including opponent team, venue to make the team selection more efficient.

- This module gives the team combination for playing XI which is backed up by each player's selection justification report to help coaches and selectors to validate their gut feeling and give any alternate options which might have been overlooked.

3.1.3 Module 3: Long-Term Team and Auction Management

- This module aims to keep check and balance on player's workload and form in matches.
- It uses past data to track fatigue among players through their performances.
- This module helps in long term planning in auctions. It helps management to plan ahead in teams auctions and select best players according to their current form and performance.
- The module uses performance data and aligns role appropriateness insights with Module 1 and Module 2. Financial and strategic conclusions are sent to Module 4 for the executive dashboard.

3.1.4 Module 4: Central Dashboard and Integration

- This module controls the entire CrickVision system. Its main function is to aggregate module outputs into a complete, easy-to-use decision-making platform.
- Modules 1, 2, and 3 results are combined into an executive dashboard that delivers coaches, selectors, and franchise managers integrated insights, real-time alerts, and customized suggestions.
- The module ensures the React front-end (responsive and interactive) and Python and machine learning backend function seamlessly together. A seamless and efficient user experience is ensured.
- This module's strongest feature is that it unites all available information, keeps things from getting messy, and empowers stakeholders to make clear and well-informed tactical and strategic decisions.

3.1.5 Module Connections

- Module 1 outputs performance-based assessments that improve role appropriateness predictions and team selection.
- Player performance data from Module 1 can be used for long-term planning, auction simulations, and wise investments in Module 3.
- Module 4 combines, synthesises, and creates a dashboard for viewing and study from Modules 1, 2, and 3.
- Finally, Module 4 provides coaches, selectors, and managers with an easy-to-use interface that presents recommendations and insights in a form they can comprehend and act on.

3.1.6 Modular Decomposition

- Four independent but connected modules make up CrickVision. Because each module handles a different functional area without overlap, it's easy to comprehend and manage. This simplifies software testing, debugging, and updating.
- Live ball-by-ball data streaming, biometric fatigue tracking, and video-based performance analysis can be added as independent modules without modifying the system architecture due to the modular design.
- The modular approach makes teamwork easier because developers can work on multiple components simultaneously.
- Module 4 works as integration layer and combines all results across all modules to make an easy and understandable dashboard to help make efficient decisions

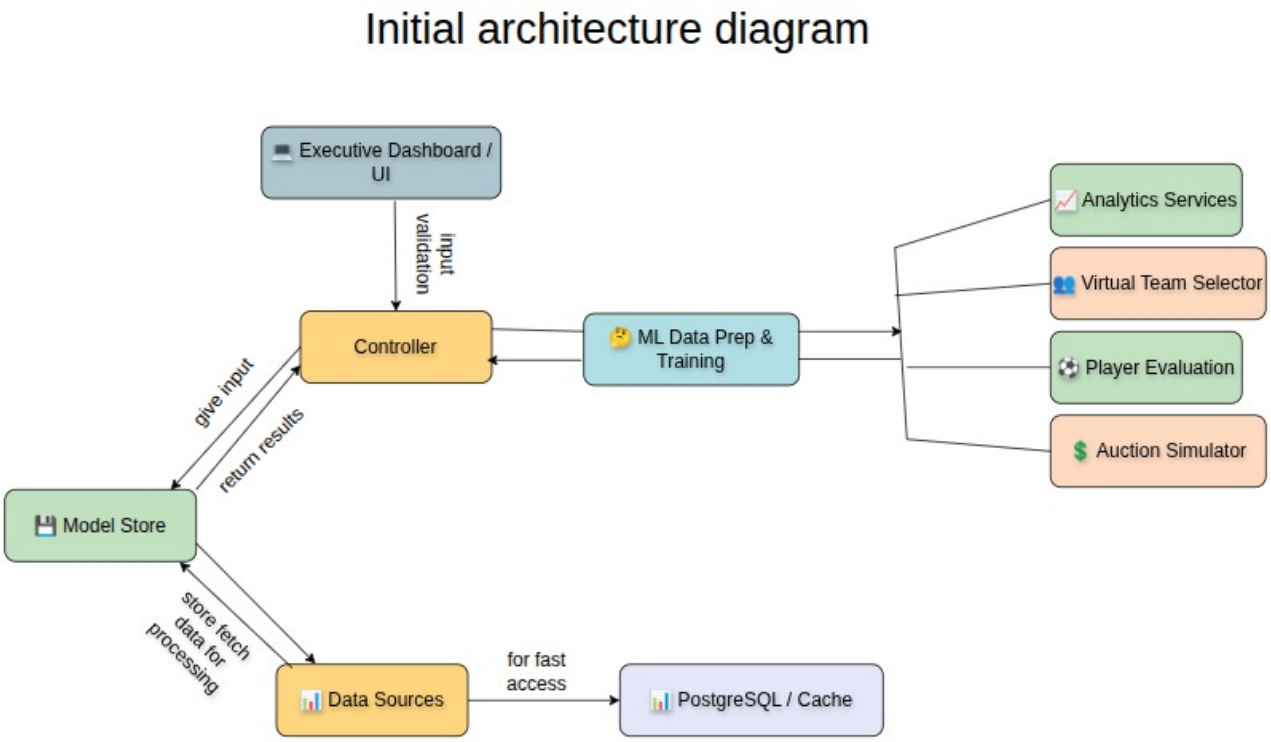

Figure 3.1: Initial Architecture

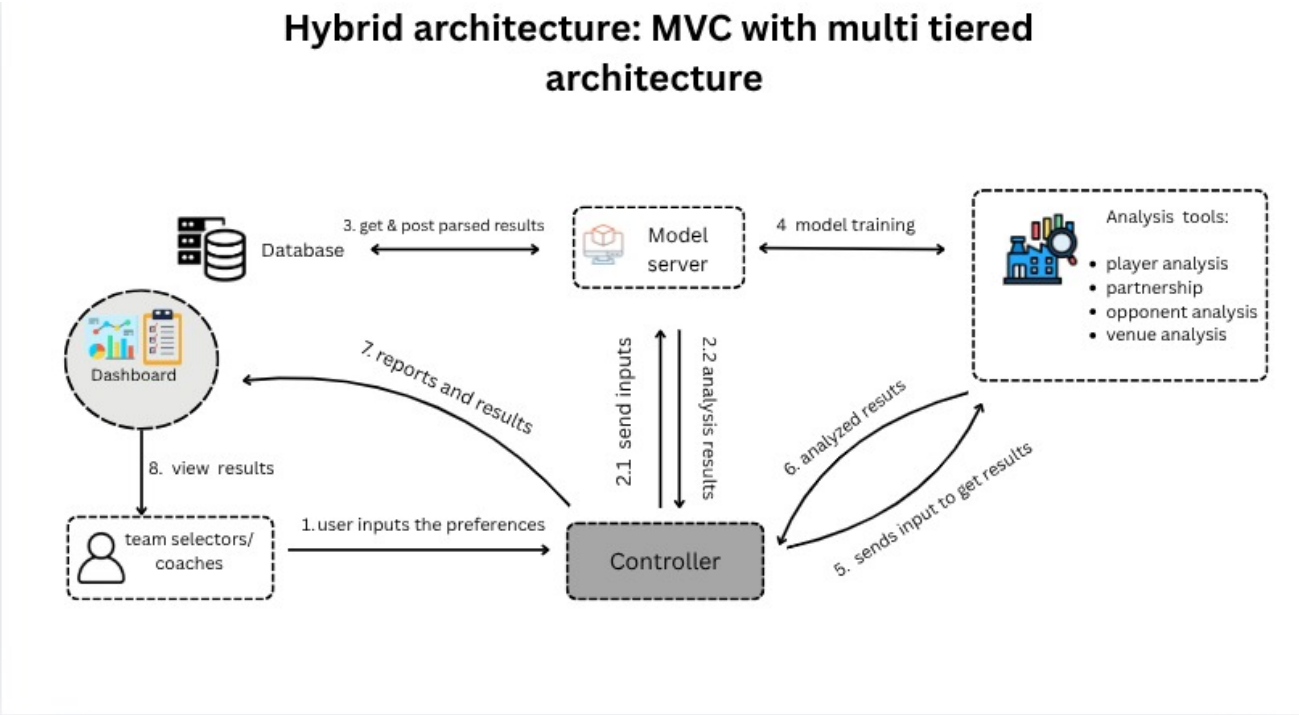

Figure 3.2: Final Architecture

3.2 Data Design

In CrickVision, data design is an essential aspect that ensures proper storage, organization, and retrieval of data. This section outlines how the information domain is transformed into data structures and how various system entities interact with databases and storage mechanisms.

3.2.1 Database Architecture

CrickVision makes use of Supabase as its key database management system which is an implementation based on PostgreSQL. This structure would provide a sturdy, extensible and safe base on which to archive all the data of all kinds that relates to cricket and user information alongside system configurations. The database structure conforms to a relational paradigm model that has well defined entities, relationships and constraints therefore guarantees integrity of the data and maximum query performance.

3.2.2 Data Transformation Pipeline

The transformation from raw cricket data to actionable insights follows a structured ETL (Extract, Transform, Load) pipeline:

1. **Extraction**: Raw ball-by-ball JSON data from Cricsheet is downloaded incrementally using SHA-1 hash comparison and stored in the `raw_matches` table.
2. **Transformation**: Python scripts using Pandas process the raw JSON, calculating match-level and player-level statistics. This includes:
    - Aggregating runs, wickets, and other performance metrics
    - Computing derived statistics (strike rate, economy rate, boundary percentage)
    - Handling missing values and data quality issues
    - Categorizing matches by format, season, and venue
3. **Feature Engineering**: Advanced features are computed and stored in the `features` table:
    - Rolling averages over recent matches
    - Phase-specific performance (powerplay, middle overs, death overs)
    - Contextual statistics (performance vs specific teams/venues)
    - Form scores and fatigue indicators
4. **Loading**: Processed data is upserted into `matches`, `players`, and `player_performances` tables using Supabase's batch insert operations with a chunk size of 1000 rows for optimal performance.

3.2.3 Data Storage and Organization

3.2.3.1 Primary Storage: Supabase PostgreSQL

All structured data resides in Supabase's PostgreSQL database, which provides:

- ACID compliance for data integrity
- Row-level security for multi-tenant access control
- Real-time subscriptions for live dashboard updates
- Automatic backups and point-in-time recovery

3.2.3.2 File Storage: Supabase Storage

Large artifacts such as trained ML models, generated reports (PDFs), and exported visualizations (PNGs) are stored in Supabase Storage buckets:

- **models/**: Serialized ML model files (pickle, joblib, TensorFlow SavedModel)
- **reports/**: Generated PDF reports and HTML exports
- **exports/**: User-requested data exports in CSV/JSON format

3.2.3.3 Caching Layer

To optimize performance for frequently accessed data:

- Redis cache stores aggregated statistics and recently generated reports
- Cache invalidation occurs on data updates via Supabase triggers
- TTL (Time To Live) of 1 hour for real-time dashboards, 24 hours for historical reports

3.2.4 Data Security and Privacy

- **Authentication**: JWT tokens managed by Supabase Auth ensure secure API access
- **Authorization**: Row-level security policies restrict users to their franchise/team data
- **Encryption**: All data at rest is encrypted using AES-256; data in transit uses TLS 1.3
- **Audit Logging**: All data modifications are logged with user context in a separate `audit_logs` table

This data design ensures that CrickVision can efficiently store, process, and retrieve cricket data at scale while maintaining data integrity, security, and performance across all analytical and operational workflows.

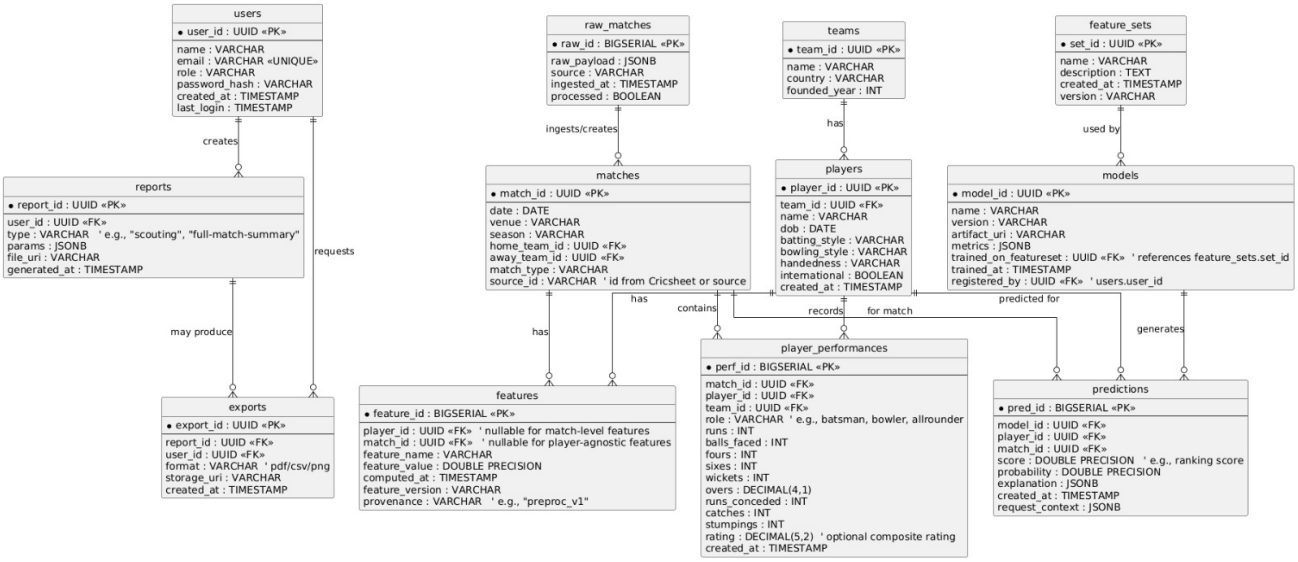

Figure 3.3: Data Design

3.3 Domain Model

The domain model for CrickVision will represent the key entities, relationships, and data flows within the system. The domain model will serve as a blueprint for understanding how the various components of the system relate to each other and how data moves through the system during operations.

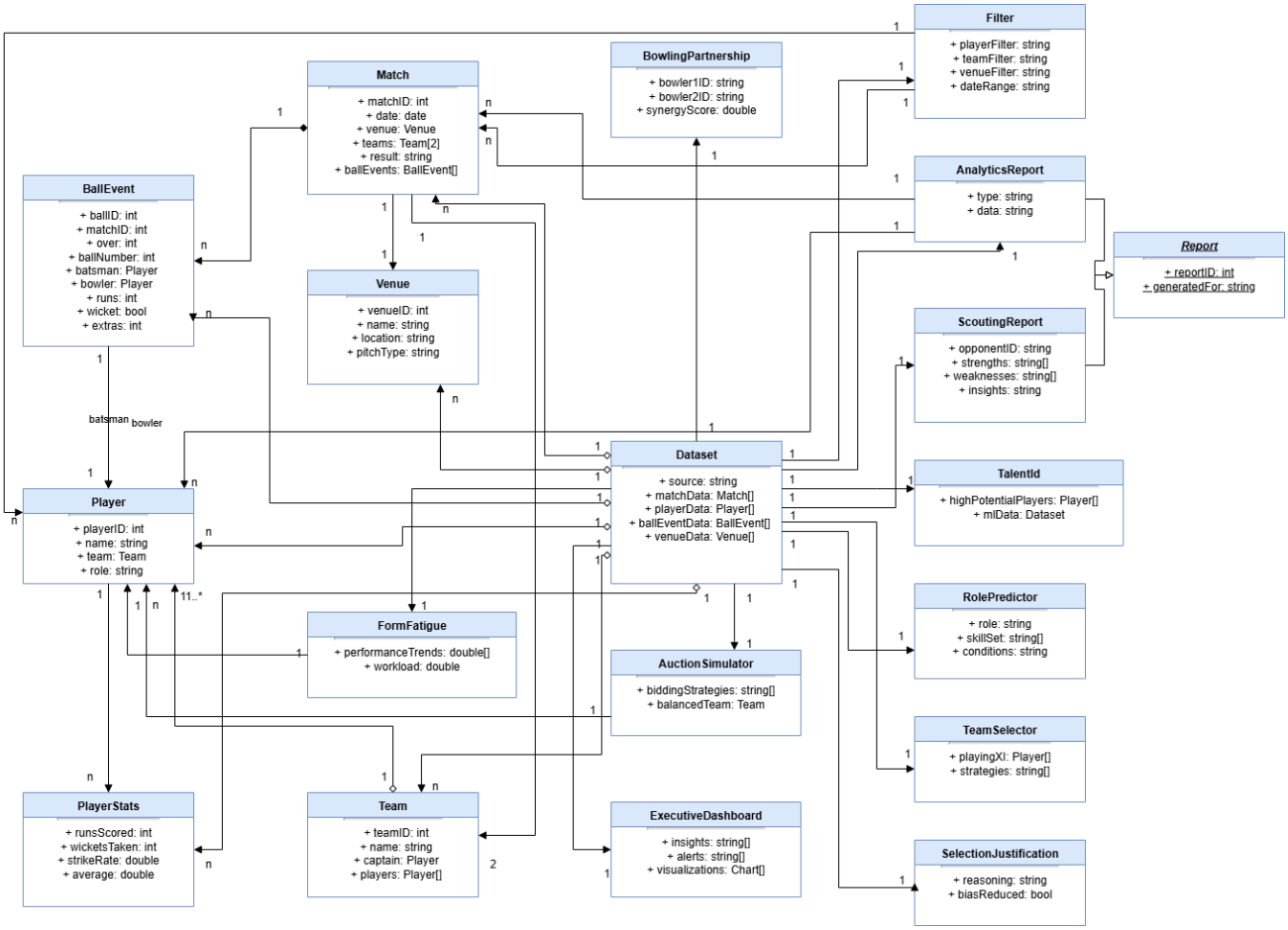

Figure 3.4: Domain Model

3.4 Design Models [UPTO THE CURRENT ITERATION ONLY]

For this section, we created design models to provide a clear understanding of the system's structure and functionality. These models include the Activity Diagram, Class Diagram and Sequence Diagram. Each model serves a specific purpose and helps in visualizing different aspects of the system.

Design Models for Object Oriented Development Approach

3.4.1 Activity Diagram

This MVC architecture effectively separates the user interface, application logic, and data management into three distinct but interconnected components.

Model : This layer handles the preprocessing of data and operations related to machine learning.It begins with raw data from external source which is then cleaned and stored in a feature store.

View : This layer represents the frontend user interface. This final processed information after applying filters is presented through dashboards and visualizations which can be chosen to export or download.

Controller : It handles all user requests and manages the application's logic. It is responsible for checking user credentials and permissions. Upon receiving a valid request from View, it calls the necessary machine learning models in the Model Layer. It sends prepared data back to the View for display.

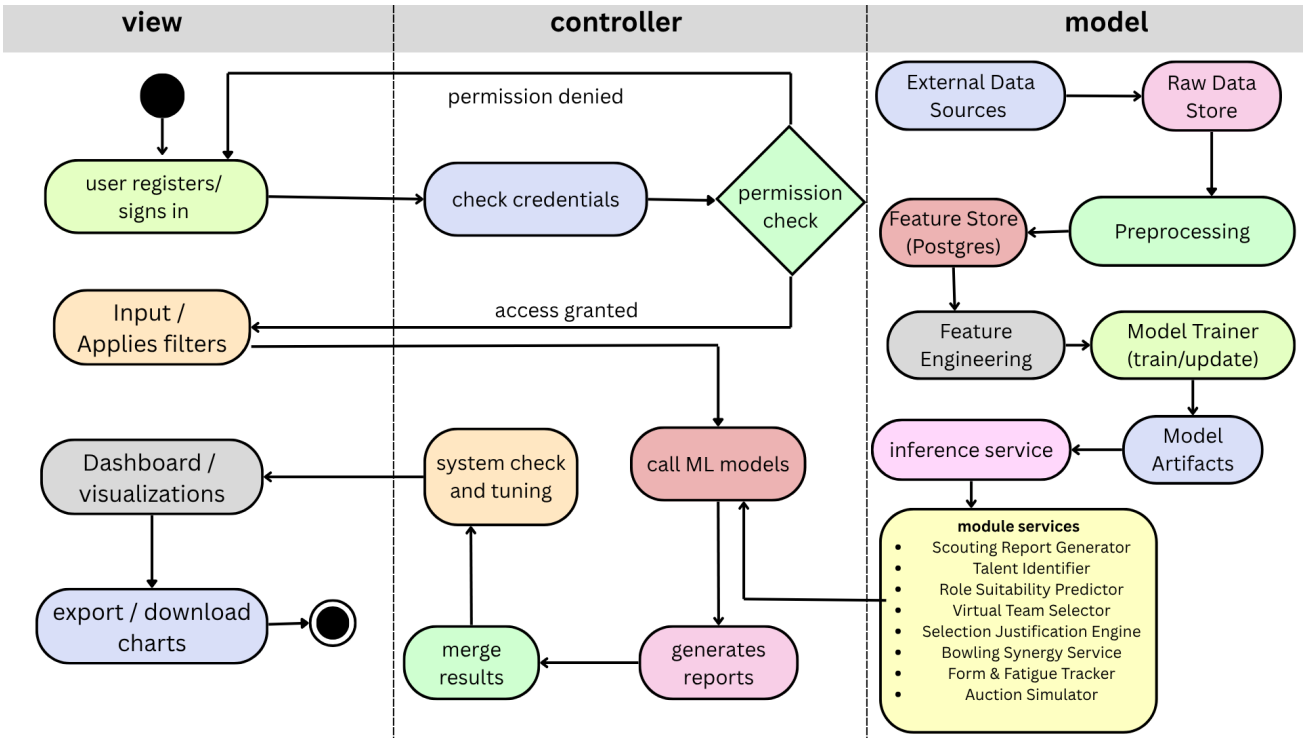

Figure 3.5: Activity Diagram

3.4.2 Class Diagram

Model : This layer contains the application's core logic and data. It includes data entities like Player, Team, and Match and services classes that perform all the main analyses.

View : This layer represents the user interface. Classes like Dashboard View and Scouting Report View are responsible for displaying data and capturing user input.

Controller : This is the middle layer that communicates with both, Model and View. It takes input from View and calls the appropriate function in the Model and then returns back the result to View to be displayed to the user.

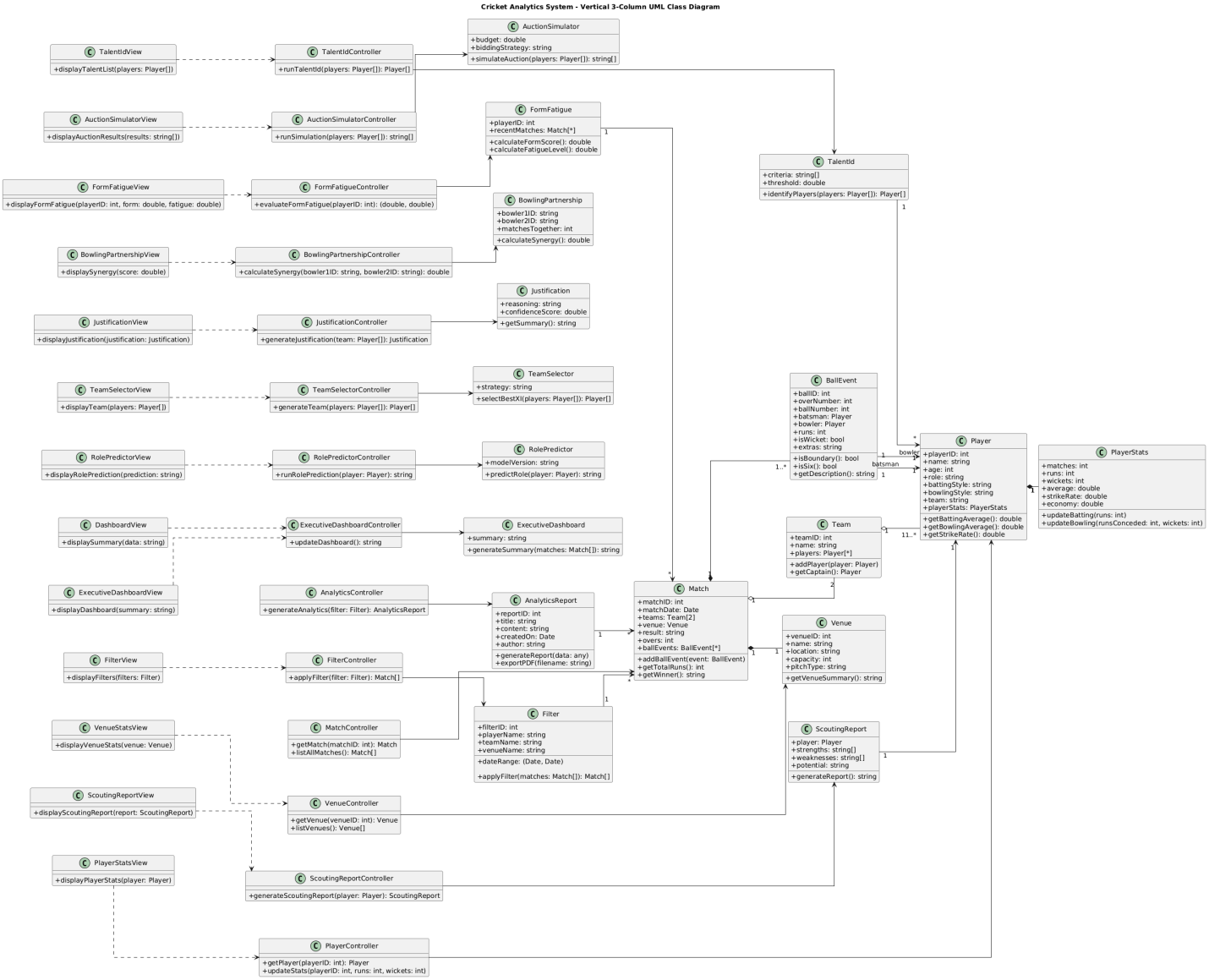

Figure 3.6: Class Diagram

3.4.3 Sequence Diagram

The sequence diagram illustrates the dynamic interactions between the user and the various microservices within the system. It details the step-by-step flow of information for key user actions, showcasing how different components collaborate to deliver functionalities. The diagram is divide into four main scenarios:

Login : The user is given access to the system once their credentials have been verified. This ensures secure access to the system.

Dashboard : Once logged in, the user requests to see the dashboard. Multiple services interact with each other to receive request to build dashboard widgets. It is then sent back to the user's screen.

Apply Filters : When a user applies filters to refine the data, the App Controller processes the query. A new report is generated through Module Services and the Inference Service then predicts the outcomes based on the specific filter criteria.

Export : If charts are to be exported, then the request is handled by the Export Service.

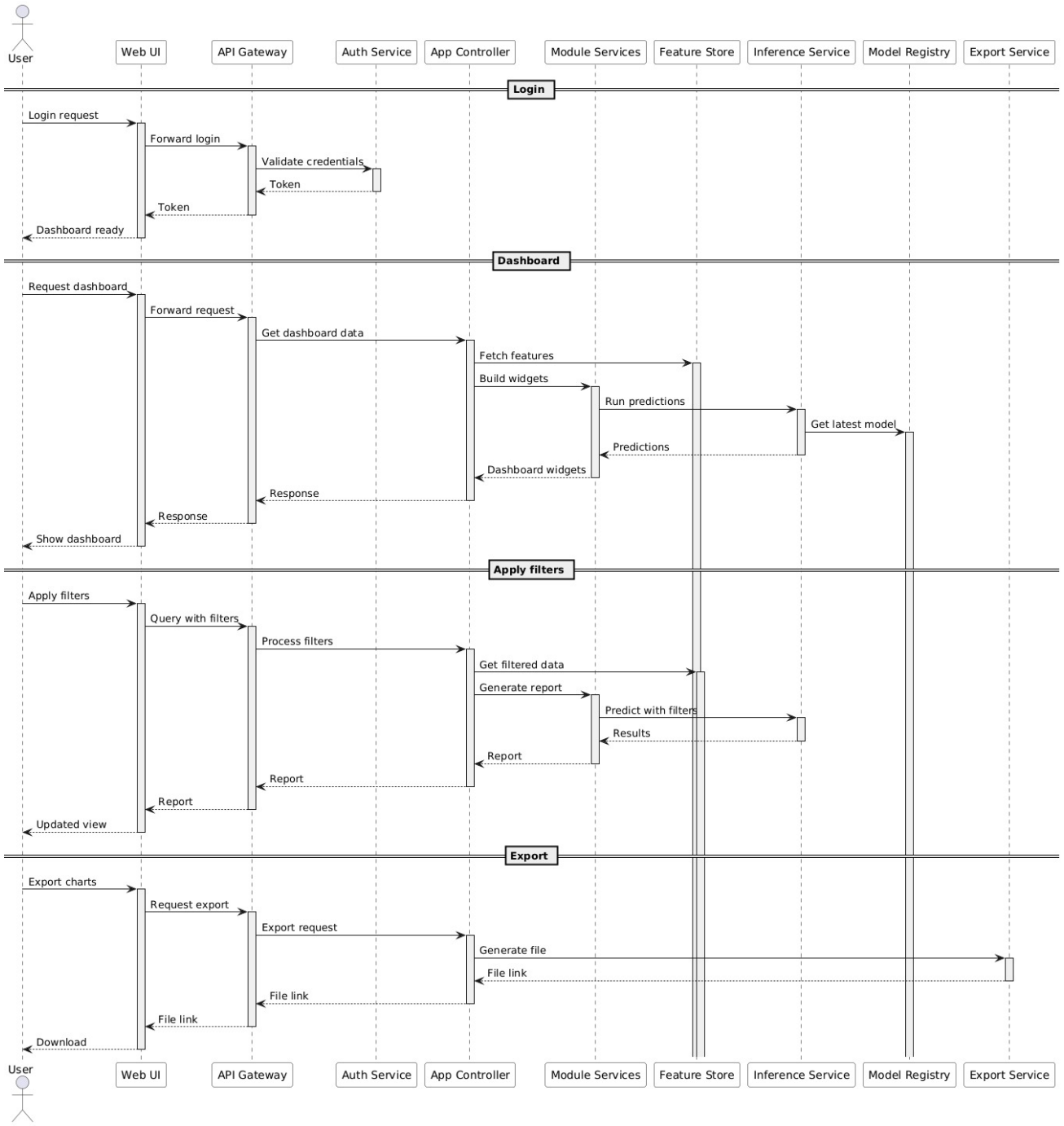

Figure 3.7: Sequence Diagram

3.4.3.1 Log In Sequence Diagram

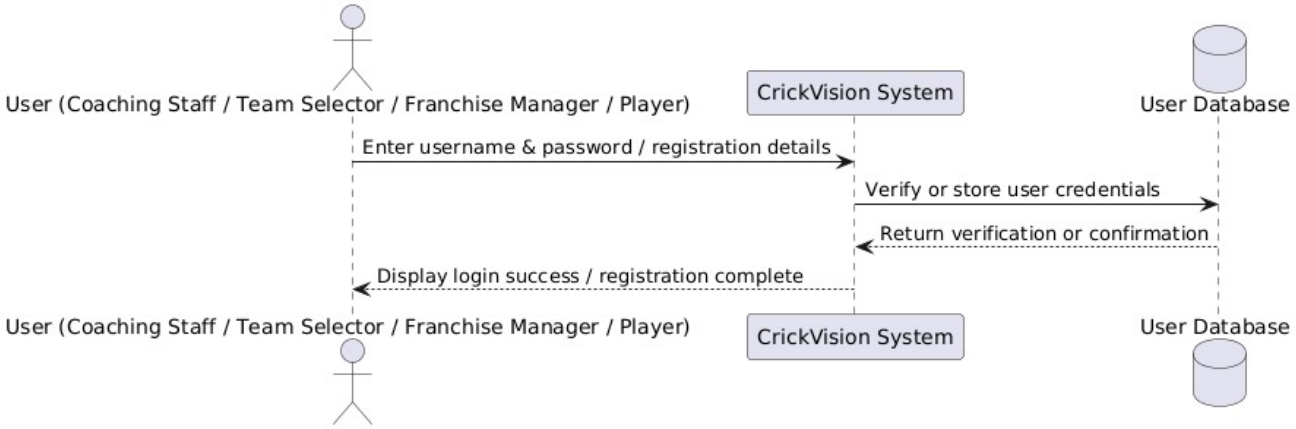

Figure 3.8: Use Case 1

3.4.3.2 Executive Dashboard Sequence Diagram

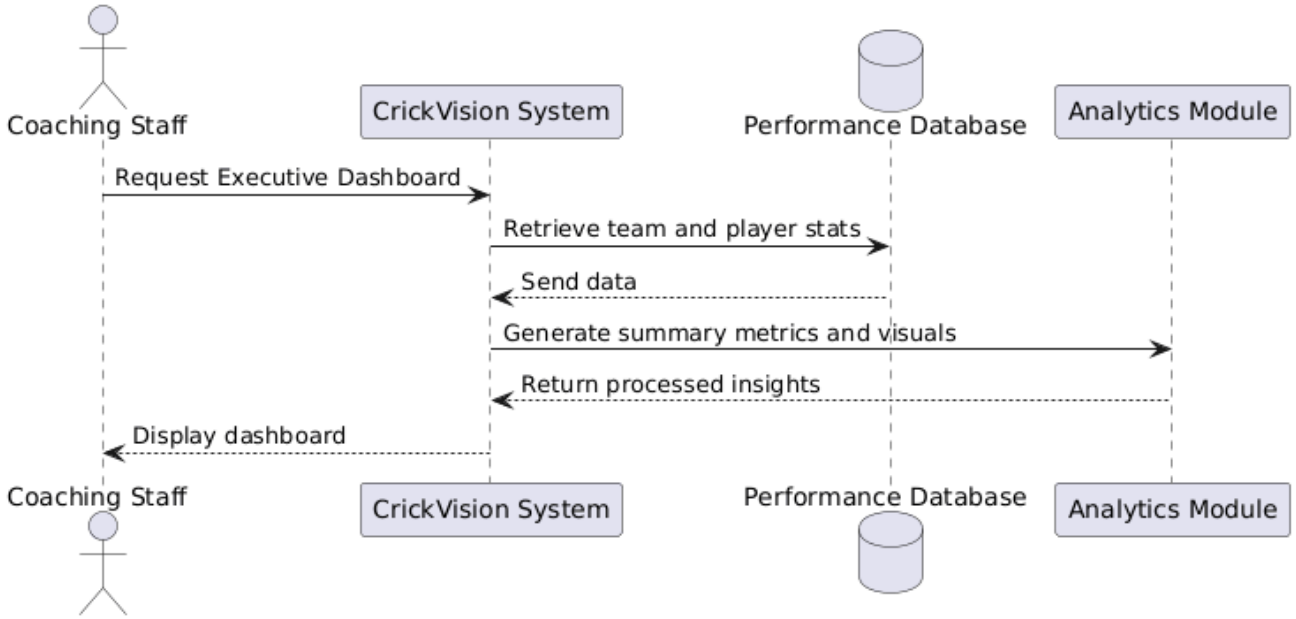

Figure 3.9: Use Case 2

3.4.3.3 Scouting Report Sequence Diagram

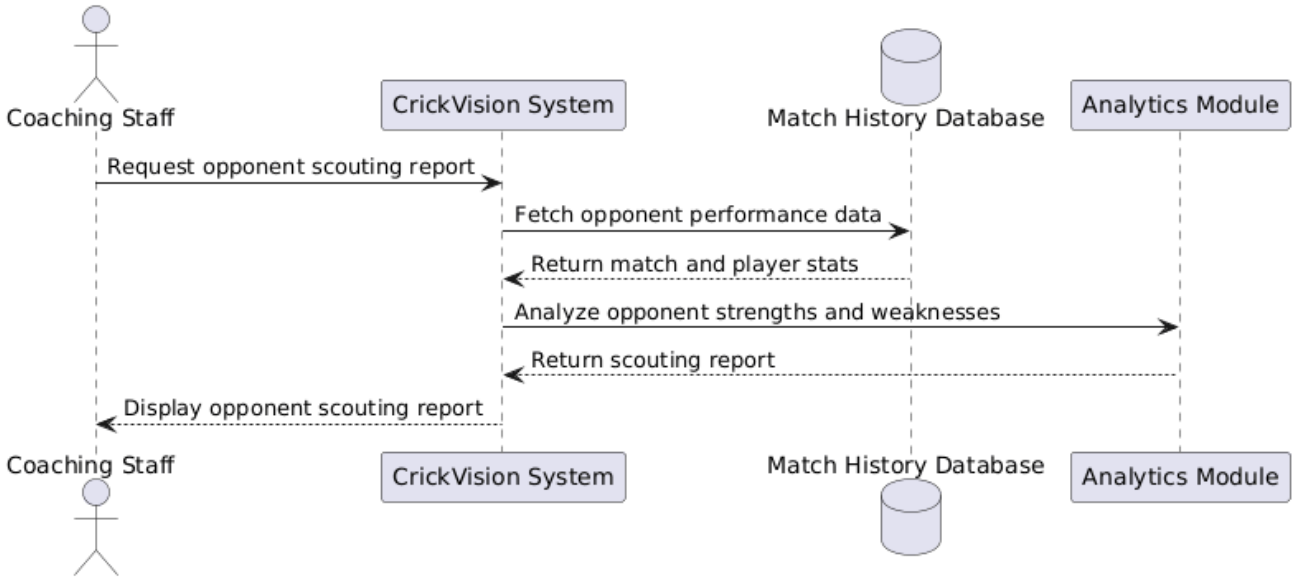

Figure 3.10: Use Case 3

3.4.3.4 High-Potential Players Sequence Diagram

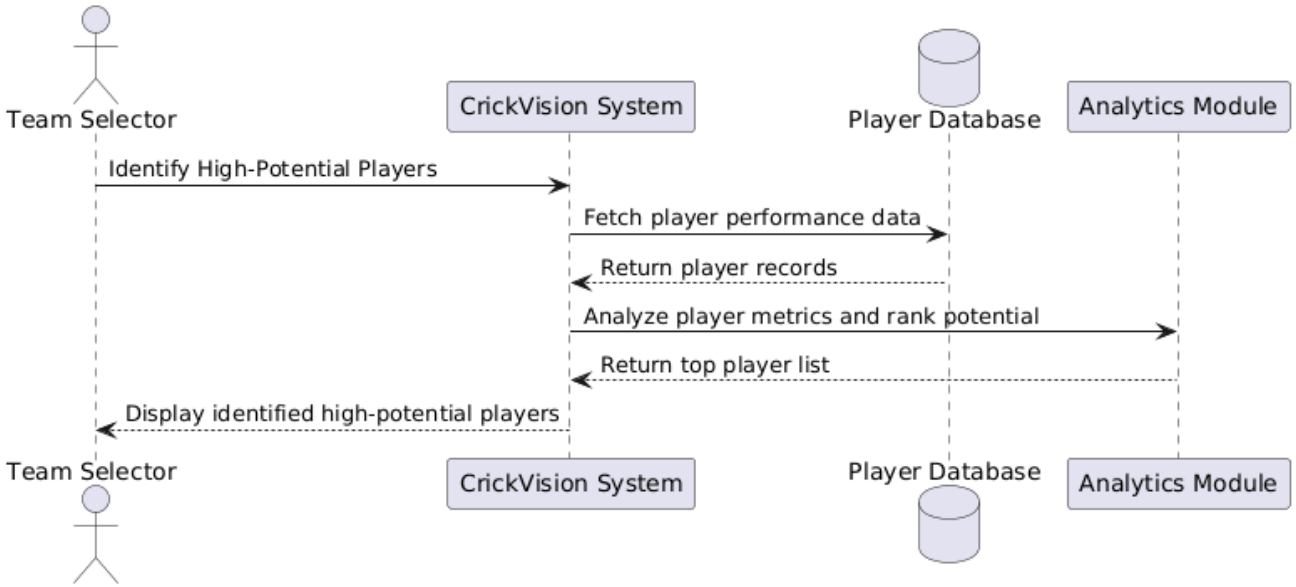

Figure 3.11: Use Case 4

3.4.3.5 Justify Player/Team Selection Sequence Diagram

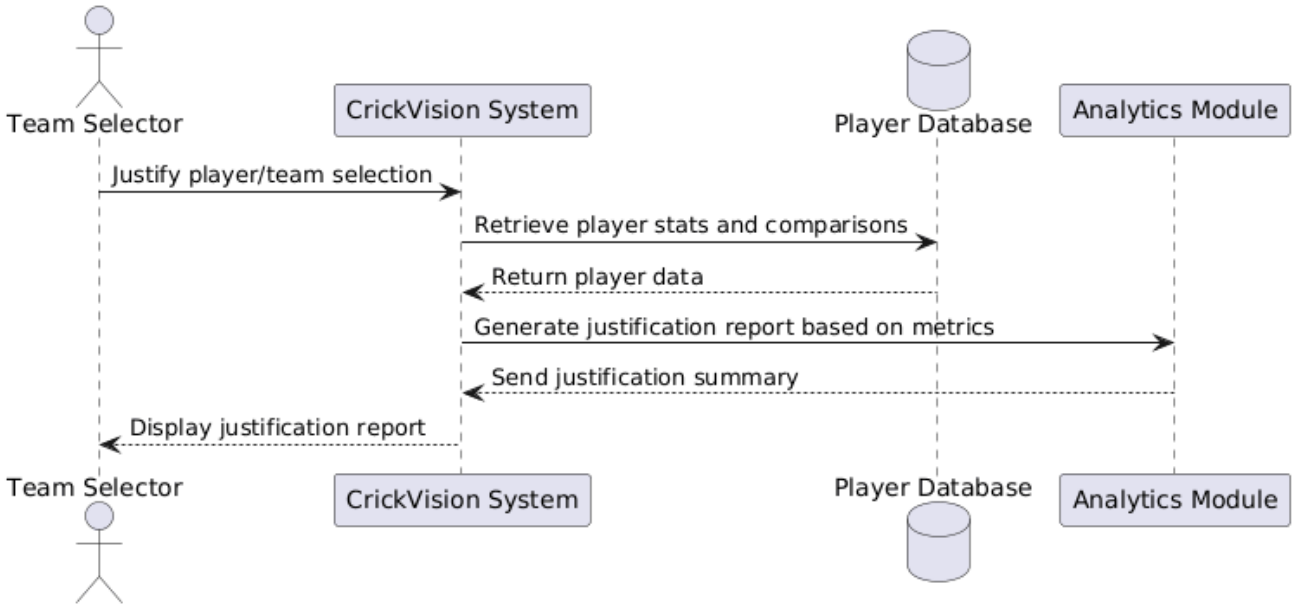

Figure 3.12: Use Case 5

3.4.3.6 Get Recommendations Sequence Diagram

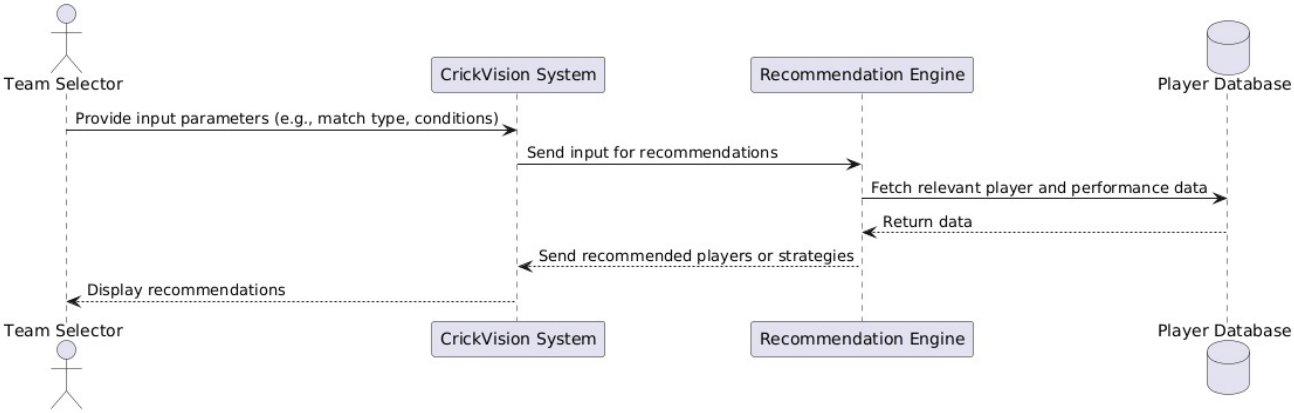

Figure 3.13: Use Case 6

3.4.3.7 Match Player to Role Sequence Diagram

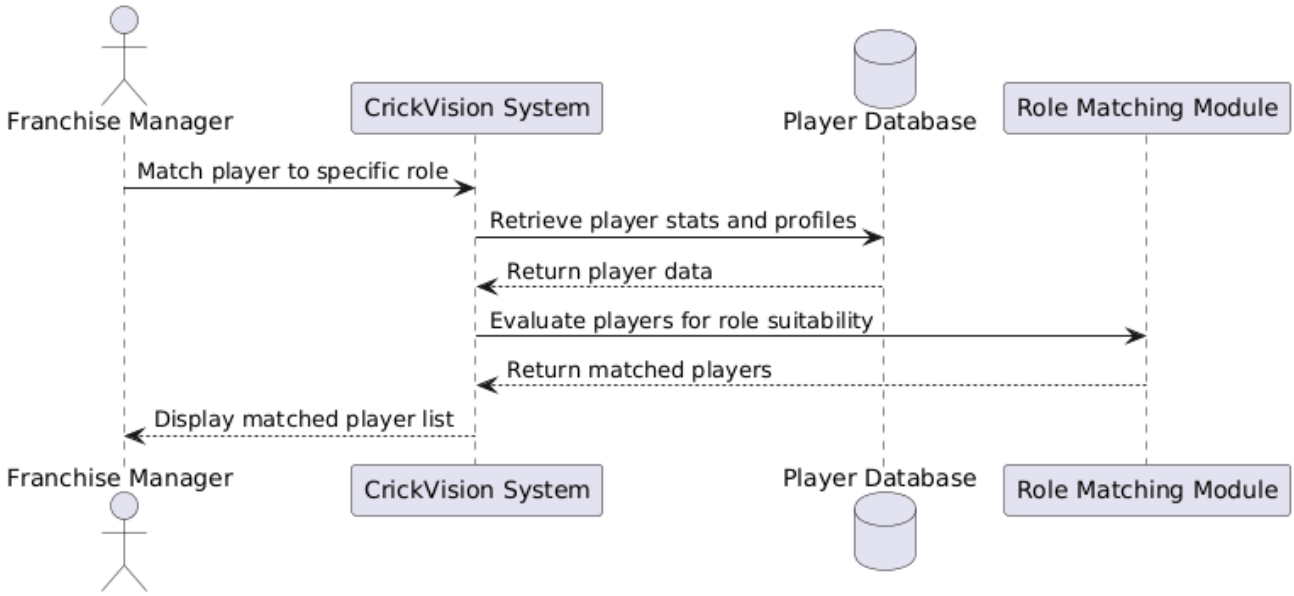

Figure 3.14: Use Case 7

3.4.3.8 Simulate Auction Sequence Diagram

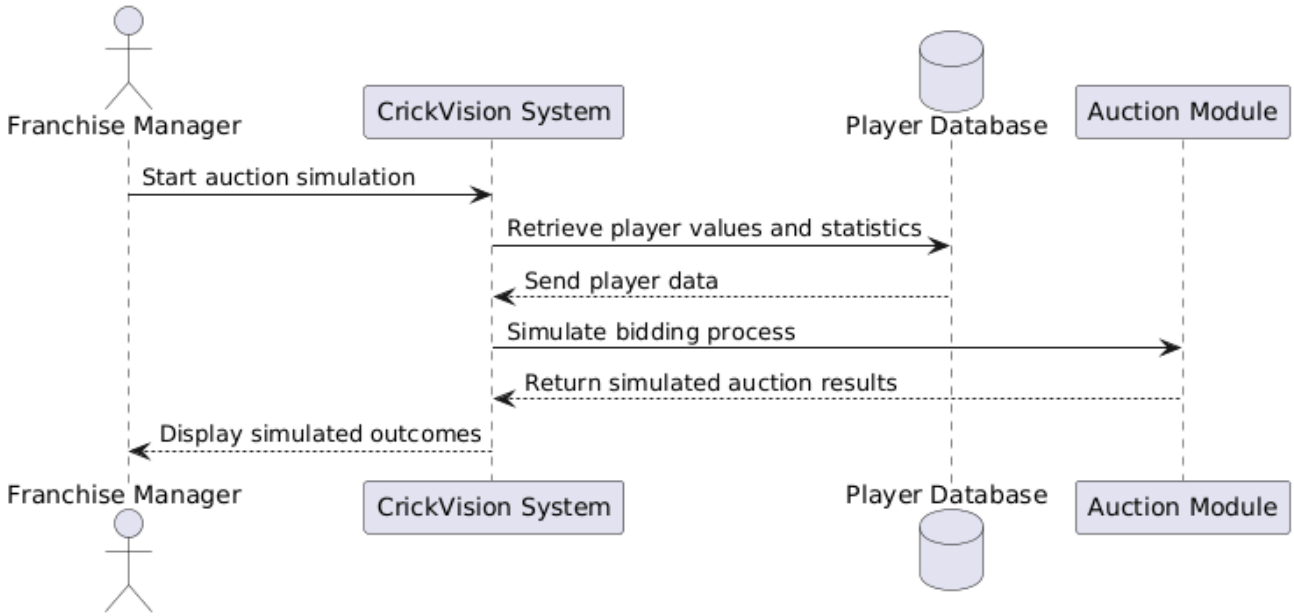

Figure 3.15: Use Case 8

3.4.3.9 Analyze Bowling Partnership Sequence Diagram

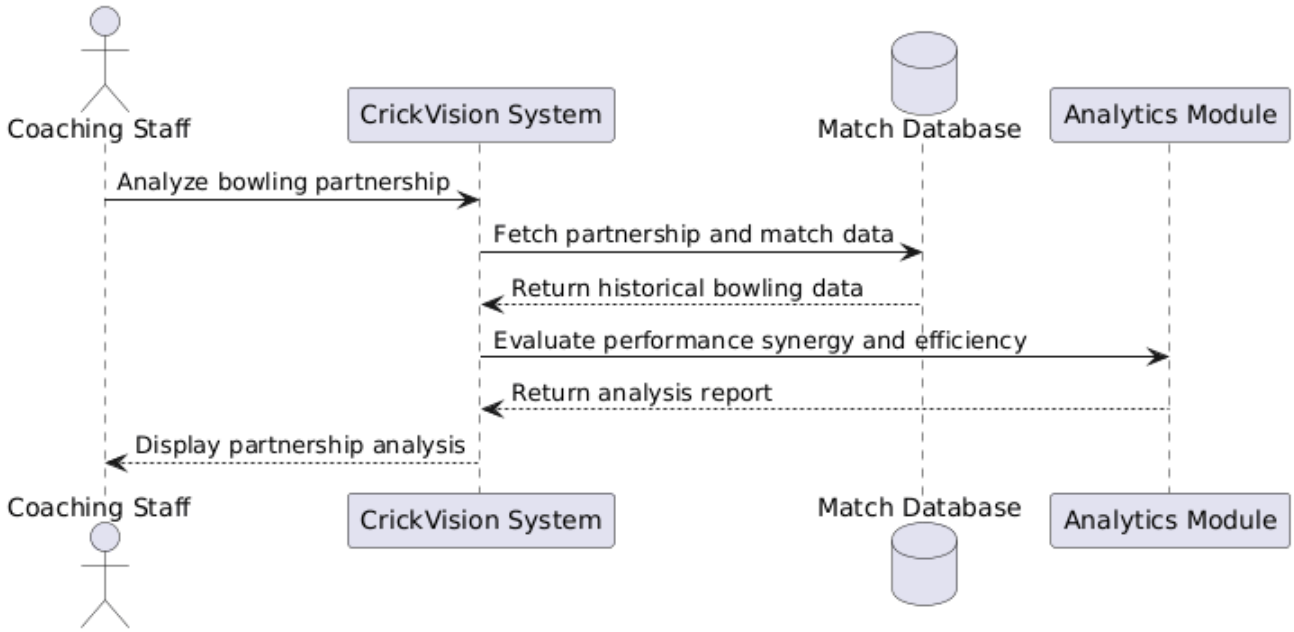

Figure 3.16: Use Case 9

3.4.3.10 Track Form and Fatigue Sequence Diagram

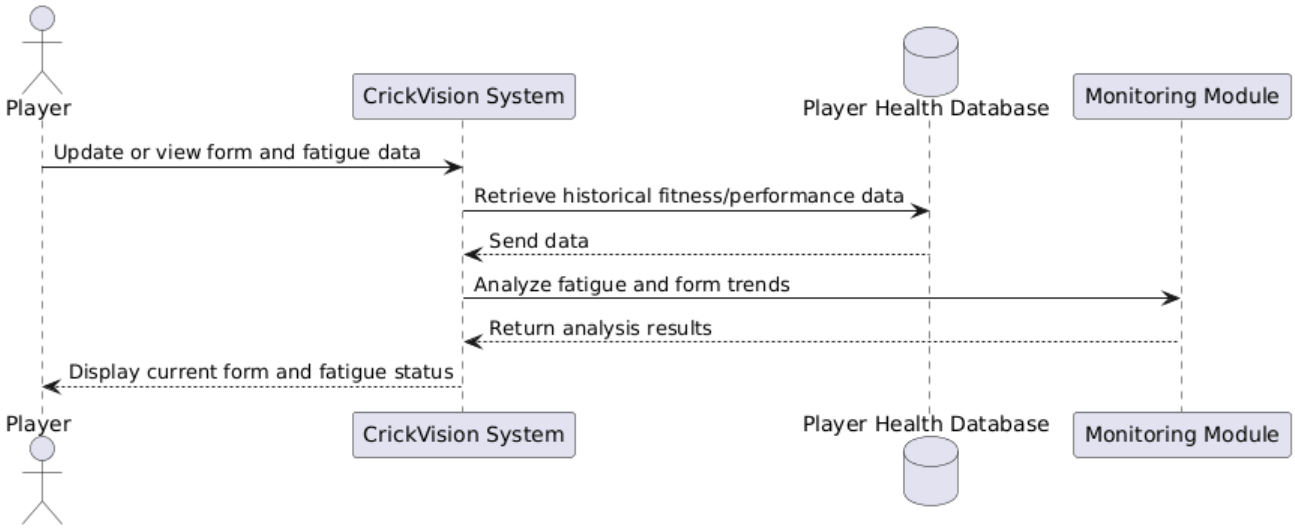

Figure 3.17: Use Case 10

Chapter 4

Implementation and Testing [UPTO THE CURRENT ITERATION ONLY]

CrickVision is a holistic, web-based solution that was developed to transform the management of cricket by replacing hunch decision-making with a combined, objective, and prescriptive knowledge and analytics. The system can execute its main functions and processes on four connected modules, namely: Talent Identification (using machine learning to scout and evaluate a player), Strategic Planning (to make the Playing XI and role fit), Performance and Welfare Management (to monitor form, fatigue, and bowling synergy) and a central Executive Dashboard that can provide all the data in a form of recommendations.

The setting of the project is the disjointed nature of the existing cricket analytics industry, in which coaches and selection teams rely on isolated applications, thus fostering ineffective tactics and preferences in selection. The design of CrickVision provides a single platform to combine machine learning models of cutting-edge player selection, opponent scouting, and financial auction simulation, which has resulted in a comprehensive solution to coaches, selectors, and franchise owners.

4.1 Algorithm Design

A model for interpretability is trained on a historical dataset of established professional players. The features include batting average, strike rate, bowling economy and wicket-taking frequency. The model learns the statistical profile of players who went on to become successful.

Algorithm 1 IdentifyHighPotentialPlayers

Input: A list of *domestic_players_data* (P_d), a pre-trained *classification_model* (M_c).
Output: A ranked list of high-potential players (P_{hp}).

```
'high_potential_list' ← [ ]
for each 'player' in 'domestic_players_data' do
    'player_features' ← CreateFeatureVector('player.stats')
    'prediction_prob' ← 'classification_model'.PredictProbability('player_features')
    if ('prediction_prob' for class "high_potential" > 0.75) then
        'player.potential_score' ← 'prediction_prob'
        Add 'player' to 'high_potential_list'
    end if
end for
Sort 'high_potential_list' by 'potential_score' in descending order
return 'high_potential_list'
```

For a given role, a set of weighted KPI is defined. The algorithm calculates a final suitability score for each player by multiplying their stats by the corresponding weights and summing the results. The players are then ranked by this score.

Algorithm 2 RoleSuitabilityPredictor

Input: A list of *players* (P), a *desired_role* (R), and *match_conditions* (C).
Output: A ranked list of players with their suitability scores (P_s).

```
'role_weights' ← GetWeightsForRole('desired_role')
'player_scores' ← [ ]
for each 'player' in 'players' do
    'suitability_score' ← 0
    'player_stats' ← 'player'.GetStats('match_conditions')
    for each 'stat', 'weight' in 'role_weights' do
        'normalized_stat' ← Normalize('player_stats'['stat'])
        'suitability_score' ← 'suitability_score' + ('normalized_stat' * 'weight')
    end for
    Add 'player', 'score': 'suitability_score' to 'player_scores'
end for
Sort 'player_scores' by 'score' in descending order
return 'player_scores'
```

4.2 External APIs/SDKs

The CrickVision system makes use of various third-party Application Programming interfaces (APIs) and Software Development Kits (SDKs) to enhance its operations, data processing and include specialty services. These are the external elements that will be required to get access to complete cricket data, implement machine-learning models, and provide a powerful and contemporary user experience. The purchasing of already existing third party services allows the development team to focus on the main analytical and strategic aspects of CrickVision. The following table lists the main external APIs and SDKs to be used in the implementation of the project.

Table 4.1: External APIs and SDKs Used in CrickVision

API/SDK Name	**Purpose**	**Usage in System**
Supabase	Database and Backend Services	Primary database for storing player statistics, match data, user profiles, and system configurations. Provides real-time data synchronization and secure authentication.
Cricsheet API	Cricket Match Data	Retrieves historical ball-by-ball match data, including deliveries, runs, wickets, and player performance across multiple formats (T20, ODI, Test).
Scikit-learn	Machine Learning Models	Used for building predictive models including talent identification algorithms, role suitability predictors, and performance trend analysis.
TensorFlow/Keras	Deep Learning	Powers advanced neural network models for complex pattern recognition in player performance data and strategic recommendation systems.
Pandas	Data Processing	Enables efficient data manipulation, transformation, and aggregation of large cricket datasets for analysis and model training.
Plotly	Data Visualization	Creates interactive charts, graphs, and dashboards for the executive dashboard, performance tracking, and tactical reports.
React	Frontend Framework	Builds responsive, component-based user interface for seamless user experience across all modules and dashboards.
Flask/FastAPI	Backend API Framework	Handles HTTP requests, routes user interactions to appropriate services, and manages communication between frontend and ML models.
NumPy	Numerical Computing	Provides efficient array operations and mathematical computations required for statistical analysis and ML preprocessing.
JWT (JSON Web Tokens)	Authentication	Implements secure, stateless authentication system for user login and authorization across different user roles.
Chart.js	Additional Visualization	Supplements Plotly with simple, lightweight charts for real-time performance metrics and quick statistical displays.

4.3 Testing Details

Once the system has been successfully developed, testing has to be performed to ensure that the system working as intended.

4.3.1 Unit Testing

Each unit test is designed to test a specific function or method independently from other components, helping to identify issues directly related to the functionality being tested.

Module 1: Talent Identification and Scouting

Feature 1: ML Data Preparation

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC001	Verify data ingestion from Cricsheet	Data source available	1. Trigger data fetch 2. Validate response	Valid API endpoint	Data fetched successfully with 200 status	Pass
TC002	Validate handling of missing data	Raw data with null values loaded	1. Run preprocessing 2. Check null handling	Dataset with 20% nulls	Missing values imputed or flagged	Pass
TC003	Verify data type conversion	Raw data loaded	1. Execute type casting 2. Validate types	Mixed data types	All fields converted to correct types	Pass
TC004	Verify data normalization	Raw numeric data	1. Apply normalization 2. Check range	Runs: 0-150	Values scaled to 0-1 range	Pass

Table 4.2: ML Data Preparation - Unit Tests

Feature 2: Automated Scouting Report

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC005	Generate report for valid team	Team exists in database	1. Select team 2. Click generate 3. Wait for response	Team: "Pakistan"	Report generated within 15 seconds	Pass
TC006	Generate report for individual player	Player exists in database	1. Select player 2. Generate report	Player: "Babar Azam"	Player-specific report created	Pass
TC007	Verify strengths identification	Valid player/team data	1. Generate report 2. Check strengths section	Player with high SR	Top 3 strengths identified	Pass
TC008	Verify weaknesses identification	Valid player/team data	1. Generate report 2. Check weaknesses	Player struggling vs pace	Weakness categories listed	Pass
TC009	Validate recent performance trends	Player with recent matches	1. Generate report 2. Check trends graph	Last 10 matches data	Line graph shows performance trend	Pass

Table 4.3: Automated Scouting Report - Unit Tests

Feature 3: Data-Driven Domestic Talent Identifier

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC010	Identify high-potential batsmen	Domestic cricket data loaded	1. Run ML model 2. Apply filters: role=batsman	500 domestic players	Top 10 ranked batsmen returned	Pass
TC011	Identify high-potential bowlers	Domestic cricket data loaded	1. Run ML model 2. Apply filters: role=bowler	500 domestic players	Top 10 ranked bowlers returned	Pass
TC012	Verify ranking algorithm	20 identified players	1. Generate list 2. Check ordering	Mixed performance levels	Players ranked highest to lowest	Pass

Table 4.4: Domestic Talent Identifier - Unit Tests

Module 2: Strategic Planning and Team Selection

Feature 1: Role Suitability Predictor

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC013	Predict suitability for opener role	Squad of 15 players available	1. Select role: Opener 2. Add conditions: vs pace 3. Execute	Squad with varied skills	Top 2-3 players ranked	Pass
TC014	Predict suitability for finisher	Squad available	1. Select role: Finisher 2. Set conditions: death overs 3. Run	Squad data	Best finishers identified	Pass
TC015	Test with match conditions: spin-friendly	Squad data	1. Select role: Spinner 2. Set pitch: turning 3. Execute	Venue history	Spinners ranked appropriately	Pass
TC016	Validate prediction confidence scores	Player data available	1. Run prediction 2. Check confidence values	Squad of 20	Each player has 0-1 confidence score	Pass
TC017	Test handling of unavailable players	Squad with injured players	1. Mark 3 players injured 2. Run prediction	Injury list	Injured players excluded from results	Pass
TC018	Test multiple role predictions	Single player	1. Run predictions for 3 roles 2. Compare scores	All-rounder stats	Different scores per role	Pass

Table 4.5: Role Suitability Predictor - Unit Tests

Feature 2: Virtual Team Selector

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC019	Generate Playing XI from squad	Squad of 15 registered	1. Input match conditions 2. Click "Generate XI" 3. Review	Venue: Karachi, Opponent: strong batting	Balanced XI with 11 players	Pass
TC020	Validate team balance: batsmen count	Squad available	1. Generate XI 2. Count batsmen	T20 format	6-7 batsmen in XI	Pass
TC021	Validate team balance: bowlers count	Squad available	1. Generate XI 2. Count bowlers	T20 format	4-5 bowlers in XI	Pass
TC022	Test opponent-specific selection	Opponent data available	1. Select opponent: India 2. Generate XI	India: strong vs pace	More spinners recommended	Pass

Table 4.6: Virtual Team Selector - Unit Tests

Feature 3: Selection Justification Generator

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC023	Generate justification for included player	XI selected with player X	1. Select player 2. Click "Justify Selection"	Player: Shaheen Afridi	Text summary with 3-5 key stats	Pass
TC024	Verify data-backed reasoning	Player with mixed performance	1. Generate justification 2. Check citations	Recent good form + historical stats	Stats referenced in explanation	Pass
TC025	Validate visualization in justification	Player selected	1. View justification 2. Check charts	Player stats over time	Performance graph included	Pass
TC026	Test justification export	Report generated	1. Click export 2. Select format	Complete justification	PDF/DOCX downloaded	Pass

Table 4.7: Selection Justification Generator - Unit Tests

Cross-Module Tests

Test Case ID	Test Objective	Precondition	Steps	Test Data	Expected Result	Pass/Fail
TC027	Test database query performance	Large dataset loaded	1. Execute complex query 2. Measure time	50,000+ player records	Query completes in <2 seconds	Pass

Table 4.8: Cross-Module Unit Tests

Bibliography

[1] CricViz. "CricViz: Cricket data & analytics." *CricViz.com*. [Online]. Available: https://cricviz.com/

[2] CricViz. "CricViz updates Centurion tool with industry-leading features." *CricViz.com*. [Online]. Available: https://cricviz.com/cricviz-updates-centurion-tool-with-industry-leading-fea

[3] Catapult. "Cricket Analysis Software | Performance Solutions." *Catapult.com*. [Online]. Available: https://www.catapult.com/sports/cricket

[4] Catapult. "Cricket Performance Analysis: Optimize Player Potential." *Catapult.com*. [Online]. Available: https://www.catapult.com/blog/cricket-player-performance-analysis

[5] Stats Perform. "Recruitment Solutions for Cricket." *StatsPerform.com*. [Online]. Available: https://www.statsperform.com/team-performance/performance-solutions-for-cricket/recru

[6] Stats Perform. "Opta-powered Performance Solutions for Cricket." *StatsPerform.com*. [Online]. Available: https://www.statsperform.com/team-performance/performance-solutions-for-cric

[7] FYP1-MidReport-F25-246-D-CrickVision.pdf. "1.1 Problem Statement." *CrickVision Project Document*. [Online]. Available: /home/ubuntu/upload/FYP1-MidReport-F25-246-D-CrickVision.pdf