# Building a Sports Data Software: A Requirement Engineering perspective

M. ATIF SIDDIQUI
*The University of Kansas*

Over the last decade, the fusion of analytics and software has activated a high-stakes data revolution across competitive sports. Advanced metrics, video feeds, wearable devices, and interactive systems now provide invaluable intelligence for improving performance, aiding decision-making, and optimizing workflows. However, enormous potential still lies largely untapped without the right technical approach tailored to the ever-evolving and variable needs of different coaches, teams, and sports. Many organizations adopt solutions that are misaligned with their practical workflow. As new ideas emerge and requirements evolve amidst rotating personnel, existing systems often lack the ability to adapt. A clear understanding of available technology, its implications, and utilization is crucial to overcome these challenges.

With the increasing importance of data-driven decision-making, more professional teams are building in-house software solutions to cater to their specific needs. However, a literature gap exists concerning the unique requirements for developing software solutions in sports, limiting the effective design, development, and deployment of custom software solutions.

Software requirements engineering (SRE) is a crucial bridge for connecting innovative capabilities to real impact in software development. SRE can provide sports organizations with an effective means to comprehend their needs, guiding them in adopting and building customized data software for optimal utilization and insight generation.

This paper provides an approach that adapts and combines SRE techniques to comprehend end-user needs in building effective sports data software. A study is conducted to understand the sports domain through role-specific surveys to discover current challenges and wish lists. Subsequently, an industry survey is conducted to grasp the current market landscape, providing examples for different types of data softwares. The paper then presents academic SRE methods that can be adapted for the sports domain. These methods aid in understanding end-user needs for utilizing existing products and constructing stakeholder-satisfaction-oriented software solutions. Finally, a sports data software is modeled by combining SRE techniques, illustrating how SRE can elevate software development and use.

The focal point of this paper emphasizes the importance of understanding end-users expectations and integrating them with technical insights through a systematic SRE approach. This approach aims to enable organizations to unlock the full potential of technology in the ever-changing world of sports.

General terms: Sports data software, software requirement engineering, software development, software requirement specification.

---

Authors' address: M. A. Siddiqui, Department of Electrical Engineering and Computer Science, University of Kansas, 1520 W 15th St, Lawrence, KS 66045, USA. Correspondence email: atifsiddiqui10@ku.edu.

# 1. INTRODUCTION

## 1.1 Data softwares in sports

Technological advancement has led to a significant increase in data across the sporting landscape. Data collected through wearable devices, advanced tracking, computer vision capabilities, and video tagging provide a wealth of information. The massive amount of available data has given rise to different data softwares catering to various tasks and workflows within the sporting domain.

Sports data software is defined as a software solution that empowers end users with the capabilities to extract meaningful insights from the vast amounts of available data. These software solutions play a pivotal role in complementing the workflows of various roles within the sports domain, functioning as invaluable tools for coaches, trainers, and other stakeholders. Moreover, these software solutions connect raw data and actionable insights, allowing teams and coaches to harness the full potential of their data. By providing users with intuitive interfaces and analytical tools, these software solutions facilitate a seamless transition from raw data to informed decision-making. For example, using data software, a team can analyze player performance, manage training loads, prevent injuries with greater precision, and strategically plan and adapt game tactics based on detailed opponent and team analyses. Data software solutions act as enablers, empowering sports professionals to derive strategic advantages from their data.

In conclusion, the definition of data softwares in sports is intricately tied to its role in facilitating the extraction of insights from the expansive data landscape. These solutions act as essential companions to the various roles within the sports domain, providing them with the means to navigate and leverage the vast sea of data at their disposal.

## 1.2 Software requirement engineering

Software Requirement Engineering is a fundamental process in software development involving the systematic elicitation, analysis, documentation, and validation of requirements to develop  software systems that meet end user needs. The importance of SRE becomes increasingly evident when considering its role in establishing a structured foundation for the development process. Through rigorous methodologies and frameworks, SRE ensures a comprehensive understanding of user requirements and lays the groundwork for scalable, customizable, and actionable software solutions. It serves as a guiding force throughout the software development lifecycle, facilitating communication between stakeholders, developers, and end-users.

## 1.3 Problem statement

Despite the rapid integration of data in sports, a noticeable need exists for comprehensive literature addressing the requirements for developing robust and tailored sports data softwares. Existing studies lack a systematic approach or a defined model that can serve as a foundation for developing independent and scalable software solutions. This knowledge gap emphasizes the critical need for a dedicated focus on the study of requirement engineering and its role in building and utilizing sports data softwares.

## 1.4 Contribution

This paper aims to significantly contribute to sports software development by introducing a systematic requirement engineering model. This contribution is expected to directly and positively impact software development efforts, fostering the creation of scalable, effective, and adaptable solutions. By adaptations SRE techniques to the sports domain, the research aims to empower sports organizations to navigate the complexities of understanding end-user requirements. Thus enhancing their technological capabilities and providing a competitive edge against their competition.

**Empowering In-house analytics departments**

A noteworthy illustration of the paper's significance lies in its potential to empower professional sports teams that are increasingly establishing in-house analytics departments and proprietary software solutions. While these teams recognize the transformative potential of data-driven insights, there needs to be a documented study to guide the formulation of software requirements. This paper aims to bridge this gap by providing a framework for understanding the software requirements, offering a blueprint for constructing software systems that align with the distinctive needs of sports teams. By furnishing practical insights, real-world examples, and strategic considerations, the paper seeks to empower sports organizations to embark on their technological journeys with confidence and clarity.

# 2. UNDERSTANDING THE SPORTS DOMAIN

Domain understanding has significant impacts on software development. Before we dive into the details of creating and utilizing sports data software it is essential to understand the unique aspects of the sports domain. Sports industry is a complex network involving various stakeholders and a wide range of data types, all crucial factors to consider when building software applications. The key players in this ecosystem include teams, coaches, players, league officials, media broadcasters, fans, and sports data companies. The data in sports spans different aspects such as video content, physical and GPS data, injury databases, scouting reports, contracts, statistics, and financial records. Any software designed for the sports industry needs to navigate this diverse landscape with precision.

This paper centers on the actors like coaches, analysts and performance staff, directly related to teams' on-field performance and the software designed to offer technical and performance-centric insights. The following subsections outline the key aspects of the sports domain, including roles, data types, and existing software solutions. It prepares the ground for a closer look at the requirements engineering process in the following sections. A grasp of the unique aspects of sports is crucial for developing software that meets stakeholder needs and fully leverages the wealth of sports data.

## 2.1. Identifying the actors

As we dive into the sports domain, understanding the various actors involved is important. It is a crucial step in building software solutions that truly cater to the specific needs of coaches, players, analysts, and other staff. This awareness guides us in adopting and building softwares that seamlessly aligns with the workflows of different roles. The following actors play a crucial role by utilizing and contributing to the development of sports data softwares:

- **Coaches:** Coaches utilize sports data software to make informed decisions, optimize player performance, and strategize effectively by leveraging comprehensive analytics on team performance, opponent scouting, player evaluations, and workout tracking.
- **Performance staff:** Performance coaches engage with sports data software to design and monitor customized workout plans, ensuring athletes' physical condition aligns with training protocols for overarching performance goals.
- **Data & video analysts:** Analysts specialize in extracting insights from data and video. Beside video analysis, they utilize data softwares for data processing, visualization, modeling, and report creation to highlight key performance indicators and refine strategies based on empirical data.
- **Players:** Players have not widely embraced the use of sports software as significant users. They tend to interact with sports data software to access personalized insights into performance metrics, training routines, and health data, enabling them to monitor progress, identify areas for improvement, and collaborate with coaches and trainers for individualized development.
- **Sport scientists:** Sport scientists leverage sports data software for in-depth analysis of physiological and biomechanical data, utilizing it to prevent injuries, optimize recovery, and support overall player well-being through evidence-based interventions.
- **Business analysts:** Business analysts use sports data software to conduct comprehensive analyses, providing insights into market trends, fan engagement, and revenue optimization, facilitating informed business decisions within the sports industry.
- **Data providers:** Data providers play a central role in supplying the necessary data infrastructure. They contribute to the ecosystem by ensuring the availability of accurate and relevant data that fuels the functionalities of sports data software.

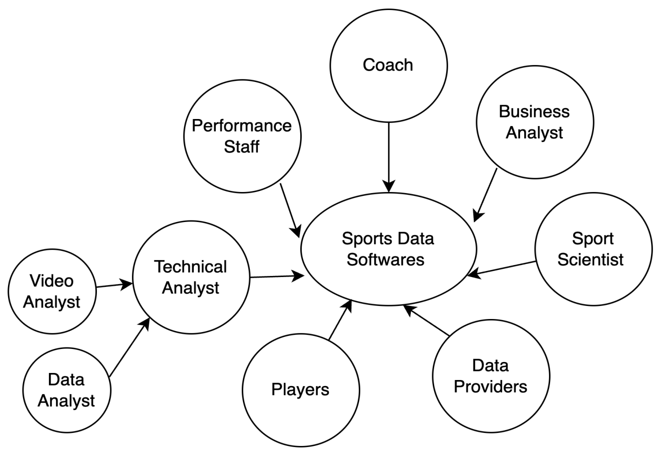


***Figure 1: Types of actors***

In conclusion, coaches, performance coaches, data and video analysts, players, sport scientists, and business analysts actively engage with sports data software, forming an interconnected ecosystem. Although this ongoing collaboration produces challenges for developers, it also propels continuous advancements, creating an environment where data-driven insights positively influence every facet of the sports industry.

## 2.2 Data collection in sports

The systematic collection and analysis of data plays a pivotal role in shaping strategic decisions and optimizing performance in sports. This section describes the various types of data that form the backbone of informed decision-making in the ever-evolving world of sports.

- **Team and player statistics:** This category encompasses a broad range of statistical information related to both individual players and entire teams. Metrics like points scored, goals, assists, and field goal percentages provide quantitative insights crucial for evaluating and comparing the overall performance of players and teams.

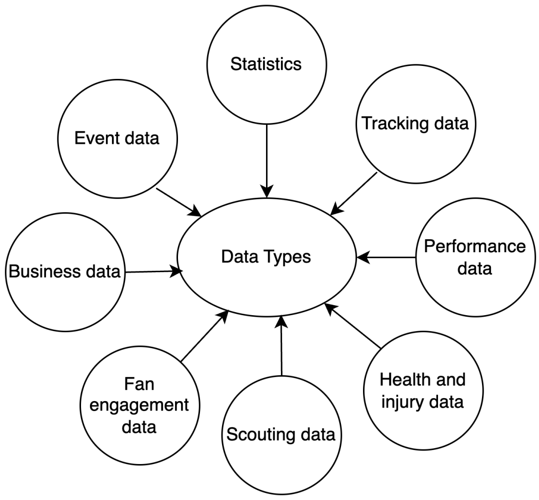


***Figure 2 : Data types in sports***

- **Match event data:** Capturing crucial moments during a game, match event data includes goals scored, fouls committed, substitutions made, passes executed, and other events shaping the course of a game. It provides coordinates for each event and compliments it with qualifiers. Analyzing this data provides a granular understanding of game dynamics, aiding strategic planning for future games.
- **Player tracking data:** Utilizing advanced sensor technologies, player tracking data involves continuous monitoring of player movements and positions on the field. This provides insights into spatial awareness, speed, distance covered, and play patterns—crucial for refining strategies and optimizing team coordination.
- **Team and player performance data:** This category of data provides comprehensive assessment of team and individual player performances. It encompasses various physical attributes including training load, heart rate, accelerations and speed specific data points. This data contributes to the ongoing refinement of training regimens and game strategies.
- **Health and injury data:** Focusing on the physical well-being of athletes, health and injury data include information on injuries sustained, recovery timelines, and overall health assessments. Monitoring this data allows sports organizations to implement preventive measures and tailor training programs for optimal athlete health and fitness.
- **Fan engagement data:** In the era of heightened connectivity, fan engagement data is crucial. It includes information related to fan interactions, social media engagement, ticket sales, and viewership statistics. Understanding fan behavior contributes to

enhanced fan experiences, targeted marketing strategies, and the commercial success of sports organizations.
- **Business data:** Adding another layer, business-related data encompasses insights into market trends, revenue optimization, and overall business performance. This data aids informed decision-making for sports organizations, ensuring a balanced approach between sporting excellence and commercial success.
- **Scouting Data:** Critical for talent identification, scouting data involves the analysis of players' potential and suitability for a team. This data includes performance indicators, skill assessments, and other criteria crucial for building competitive and well-rounded teams.

Understanding the different types of data collected in sports is crucial for making smart decisions in sports technology. It can help guide sports organizations to make informed decisions when using or creating data software.

## 2.3 Current market products and in-house solutions

Creating effective sports software begins with a clear understanding of the current technology landscape. This involves examining platforms related to performance, engagement, and betting to identify essential functionalities. Evaluating data handling, interface options, and access constraints helps pinpoint alignment gaps between common tools and specific organizational needs. This understanding serves as a foundation for tailored innovation that meets particular roles and use cases.

Taking a thoughtful approach for choosing the right software solution helps in avoiding missteps and obstacles in the future. By thoroughly characterizing existing spaces, the development of customized technical solutions can follow well-informed paths that tightly address user realities for maximum impact. Table 1 provided different industry software types, key features and some example solutions that currently exist in the market.

***Table 1: Data software market***

| **Software Type** | **Key Features** | **Software Examples** |
|---|---|---|
| Player Performance Analysis | Video analysis, performance metrics, sports science application | Hudl Sportscode, Hudl Studio Tableau, PowerBi, Catapult |
| Team Strategy and Planning | Statistics, match reports, player reports, opponent analysis resorts, data analysis and visualization dashboards | Wyscout, Justplay Sports solution, Statsbomb, Hudl Insight, Statsperform |
| Scouting and Recruitment | Player profiles, recruitment events | Wyscout, Hudl, Driblab, TransferMarket |
| Fan Engagement | Personalized content delivery, fan analytics | Jump Trade, Greenfly, FanThreeSixty, Dugout |
| Injury Prevention and Health Monitoring | Injury risk analysis, training load and workout analysis. | Athletigen Technology, Catapult |
| Betting and Fantasy Sports | Odds calculation, predictive analysis | Bet365, Draftkings, Fanduel |

| | | |
|---|---|---|
| Advance Data Science and Machine Learning | Advance data models and insights | Statsbomb IQ, Statsperform's Opta vision, Gemini sports analytics |

# 3. SURVEY OF EXISTING DATA AND SOFTWARE PRACTICES IN SPORTS

Building on our initial domain understanding from sections 2, a comprehensive survey was conducted to gain insights into the current software and data practices in sports. The survey aimed to understand how end users interact with available softwares and the challenges associated with them.

## 3.1 Methodology

The survey targeted three key roles central to sports software - coaches, analysts, and software owners. Dedicated question subsets were designed for each role, probing their specific workflows, pain points, and capability wishlists. This enabled gathering well-rounded inputs on sports software usage and development from both business and technical perspectives.

## 3.2 Role specific questionnaire

### 3.2.1 Coaches

The coach's survey aimed to understand how existing data and software solutions integrate within natural coaching environments. The questionnaire asked the coaches to list areas in their role where they most utilize software, and to point out the most significant limitations in current tools related to their role. They were asked how comfortable they are interpreting data and their primary analysis methods. Lastly, they were prompted to provide three features they consider essential for data softwares.

### 3.2.2 Software Developers and Data Professionals

On the technical side, developers and data professionals were asked about their challenges while developing sports-focused software. They were asked about their role in their organization's decision-making process for software use and development. They were given a list of software development traits and asked to rate their priorities.

### 3.2.3 Product Owners

The survey focused on broader software considerations for product owners, like getting practical requirements, handling user feedback, and rating priorities for the software roadmap. Product owners also shared the challenges they face and their vision for the future of a sports-specific software. This part of the survey aimed to capture the big-picture vision guiding the development and growth of softwares in the sports domain.

### 3.3 Discussion of survey results

The data collected through the survey pointed to the following conclusions:

- Coaching level showed an inverse relationship with using and understanding data softwares. This was associated with the age of the user and time availability. Several coaches marked video as their primary mode of analysis. Time limitations and understanding complex existing tools dominated the struggles faced with usage and adoption of softwares.
- Development roles listed aligning software development with sports specific needs as one of their significant challenges. A lack of a clear development plan and changing requirements were also dominant. These roles showed a lack of involvement in the decision-making process.
- Market research and brainstorming sessions were chosen as software owners' primary sources of requirement gathering. Alignment with company goals and importance to end users were selected as top priorities.

The challenges highlighted through the survey underline the need for a well-defined requirement engineering process. It will allow software providers and developers to better understand their end users and provide solutions that align with their needs and changing requirements. Table 2 shows the role specific challenges that were collected through the survey.

***Table 2. Challenges associated with roles***

| **Roles** | **Softwares Challenges** |
|---|---|
| Coaches | Time limitations |
| | Applying data insights to coaching |
| | Making sense of complex softwares |
| Developers | Alignment to sport specific needs |
| | Changing requirements |
| | Balancing performance with user experience |
| Software Owners | Securing access to sports data |
| | Identifying and implementing appropriate technologies |
| | Scalability of large volume of data |

## 4. A REQUIREMENT ENGINEERING MODEL FOR SPORTS DATA SOFTWARES

Requirement engineering is the process that involves activities for exploring, evaluating, documenting, and revising the capabilities, qualities, assumptions, and constraints for a software system. It allows us to address the following questions about a software system

1. Why is the system needed?
2. What needs will be addressed by it?

3. Who in this system will take part in fulfilling the needs?

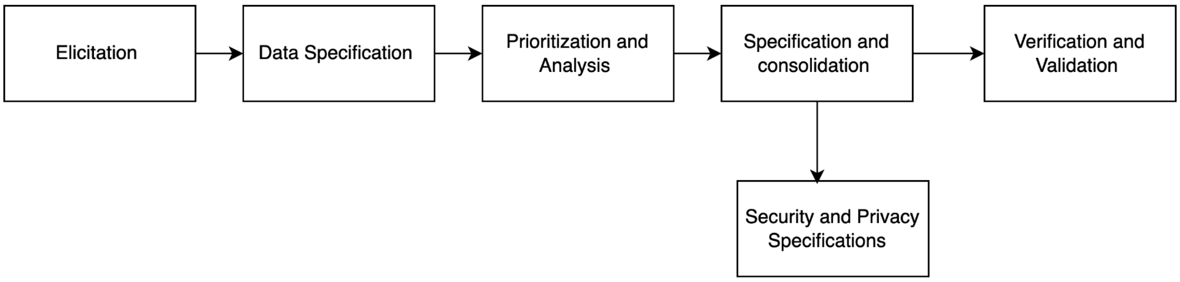


***Figure 3: RE Process flow***

Building on our domain understanding from section 2 and responding to the challenges presented in section 3, a SRE process is reviewed. This SRE process, shown in figure 3, consists of elicitation, analysis, specification and validation steps that can be adapted to sports specific needs. In addition to these steps data capture and collection specification are presented. Steps are also incorporated to tackle the security and privacy concerns related to softwares. Various techniques, shown in table 3, sourced from a multitude of academic papers are examined for each step in the SRE process. These techniques are assessed with the purpose to either combine them or selectively utilize them for the sports domain. This paper focuses on performance and technical data softwares, examples and insight are provided for these two types. The techniques and methods presented in this section are implemented in the development of Soccertact, a validation use-case for our SRE model developed in section 4.7.

***Table 3 : Papers and Techniques for each step***

| **RE Activity** | **Paper** | **Contribution** |
|---|---|---|
| Elicitation | Lau (2014) | A conceptual architecture for requirement elicitation |
| Data Specifications | Al-Najran (2015) | A requirement specification framework for data collection |
| | Narayanan (2015) | Two templates to specify data requirements |
| Analysis and Prioritization | Schmid (2021) | Techniques for analyzing and prioritizing requirement & handling changing requirements |
| Specification and Consolidation | Wieringa (1998) | Graphical techniques for requirement specifications |
| | Noorwali (2016) | An approach for integrating Big Data characteristics & quality requirements |
| Security and Privacy Specifications | Jutla (2013) | UML extension for privacy requirements analysis and specifications |
| Verification and Validation | Suratno (2015) | Techniques for verification and validation |
| | Staple (2016) | Techniques for verification and validation |

## 4.1 Requirement elicitation

Requirement elicitation is the process of gathering, analyzing, and documenting user needs and developing from these system functionality expected from a software solution. The output of this step are user stories and initial system requirements. Effective elicitation can be vital to building scalable and customized platforms aligned to end-user workflows in sports organizations. Lau's (2014) paper provides a model-driven elicitation methodology for big data systems. The model combines scenario-based elicitation technique and sensemaking models that can be utilized when building softwares for the sport domain. This model can help gain a better understanding of the coach's needs and also incorporates input from technical staff. It helps in developing new ideas and building softwares that are actionable. The paper outlines a three-step approach: scenario analysis to understand current workflows and pain points, exploiting sense-making models to refine user needs through collaborative modeling, and developing a conceptual architecture mapping technical components to user requirements. The model provided in this paper can be adapted to the sports use case, and sports data software can capture niche domain needs more effectively during elicitation phases.

### 4.1.1 Technique 1: Eliciting requirements from scenarios

This first technique extracts requirements by observing and capturing real-world workflows and processes relevant to the problem domain. For sports software, ethnographic analysis of how coaches and staff use current tools can provide insights into limitations, desired capabilities, and the need for a software solution. Scenario analysis can help identify existing pain points across coaching activities. Structured interviews can further elicit user stories prioritizing features of potential future systems.

Specifically, this technique should include extracting functional needs from real-world coaching scenarios. A scenario-driven approach can help capture end-user perspectives on current sports software practices. Analyzing different tools in place can uncover issues tackled and gaps that exist. The goal is to understand contexts for needing new customized platforms.

Primary activities should include:

- Understanding coaching and technical staff workflow, software and data use and current challenges experienced
- Collecting information about involved users and communities to inform later data and collaboration requirements
- Discussing existing data sources and formats as they relate to various workflows
- Having end users describe an ideal future system vision and eliciting user stories centering on key features

However, this technique still needs more critical input from technical partners, which is essential for converting visions into well-architected software systems.

### 4.1.2 Technique 2: Exploiting Sense-Making models:

The second technique in Lau's requirements elicitation framework leverages sensemaking models to interpret and refine user needs obtained from initial scenario analyses. Sensemaking is an iterative cognitive process that humans perform in order to build up a representation of an information space that is useful to achieve their goal. Integrating individual and collaborative sensemaking, Lau's (2014) framework provides triggers and characteristics that lead to new understandings and ideas.

In the sports landscape, there is an inherent gap between coaches' on-the-ground domain expertise and technical staff's understanding of software and data implications. Utilizing sense making models can allow coaches and analysts to share knowledge, make collective sense of software gaps and uncover more thorough functionality needs. The key focus of this technique should be bringing different perspectives together and letting roles jointly make sense of needs and possibilities. Coaches can contribute to the realities of coaching needs, while technical staff advise viable technologies—the blending yields creative ideas and requirements.

Lau's (2014) paper combines these two techniques to create a conceptual generic architecture for big data analytics elicitation. The paper provides triggers for moving into collaborative sensemaking and the characteristics that should be involved, these characteristics and triggers are shown in table 4.

***Table 4: Triggers and characteristics for sensemaking model***

| | **Description in model** | **Example in domain** |
|---|---|---|
| Triggers | Ambiguous Information | An analyst needs support from a coach on whether his/her interpretation of data is correct. |
| | Role based information sharing | |
| | Lack of expertise | |
| Characteristics | Prioritizing relevant information | Coaches provide support through feedback.<br>Collaboratively designed reports.<br>Analysis is prioritized based on how actionable it is. |
| | Sense making trajectory | |
| | Activity awareness | |

## 4.2 Data specific requirements

Following requirement elicitation the next step for sports SRE is to gather data specific requirements. At its core, impactful sports software stands on processed data providing insights. It is crucial to understand the data availability and requirements for developing actionable sports data softwares. Sports teams have access to an extensive range of data, spanning from general box score statistics to granular player tracking data. Additionally, they can tap into historical and predictive datasets, which are valuable resources for machine learning applications. To optimally harness the available data, a profound comprehension of the data sources and their capabilities is required. The following techniques can be employed to systematically make sense of the existing data, develop software that enables its full utilization, and discover if additional data is required.

### 4.2.1 Technique 1: Backward analysis for capturing data

Al-Najran's (2015) paper introduces the concept of "big data scenarios" to improve the process of big data collection. It argues that current ad hoc data gathering processes are inadequate in capturing relevant information from large-scale data. As a result, users are often unable to obtain valuable insights from the collected data.

The paper proposes a framework for big data collection and capture based on "backward analysis". The key ideas are:

- Data analytics can only be successful if the underlying data collection processes capture relevant data for the specific scenario or purpose.
- Backward analysis involves defining the properties of the required input data based on the context and properties of the desired output. This helps eliminate irrelevant data.
- The framework models the requirements for scenario-based data collection using a conceptual model called W*H which was originally developed for specifying service systems.
- The W*H model comprises primary, secondary and additional questions that help characterize the ends, means, sources and value of the data collection process.
- Applying this framework can accelerate analysis and decision making by focusing the data collection on what is relevant for the specific scenario.

To implement this methodology for sports specific data collection and capture, the key is to clearly define the scenario and purpose of data collection. The backward analysis framework can then be used to determine what data needs to be collected or acquired to serve that purpose. The W*H model provides a construct for asking the right questions (Table 5) about ends, means, sources and value to characterize data needs for the scenario. By linking data collection to intended use cases in this manner, sports organizations can vastly reduce time spent in capturing and managing useful data and filtering out irrelevant data.

***Table 5: W*H model***

| **W*H Model Questions** | | **Sport scenario specific sample questions** |
|---|---|---|
| Ends or Purpose | Why?<br>Where to?<br>For when?<br>For which reason? | Why is the data needed for optimizing player performance in team sports?<br>Where is this data intended to be used?<br>For when and for which reasons is the data crucial in sports analytics? |
| Supporting means | Wherein?<br>Wherefrom?<br>For what?<br>Where?<br>How?<br>What? | Wherein and wherefrom can analytics techniques be applied to analyze sports data effectively?<br>For what purpose will this data be used, and where does the capturing technique come into play? - How will this data be utilized for optimizing team strategies?<br>What analytics techniques can analyze this data? |
| Sources | By whom?<br>To whom?<br>Whichever?<br>What in?<br>What out? | By whom and to whom is the data collected and processed in sports analytics?<br>Whichever data sources are utilized for capturing sports data?<br>What goes into and out of the data collection process? |

| | | |
|---|---|---|
| Value | Where at?<br>Where about?<br>When? | Where at and where about will the values from this data enhance the team's performance?<br>When will the data values be most impactful for decision-making in sports analytics?<br>What specific values will this data provide to contribute to the team's success? |

### 4.2.2 Technique 2: Data templates

Narayanan's (2016) provides a two-template that can be effectively employed for sports-specific data management. The “Data Acquisition Template” can streamline collecting relevant information. It involves defining specific requirements and the data required to fulfill these requirements. This template can be tailored to the unique needs within sports for efficiently gathering data from various sources. Systematically implementing this template establishes a robust process for acquiring and handling data.
The “Matching Data to Business Problem" template is a strategic tool for data organizations. It precisely aligns data with specific business challenges by defining problems, specifying required data, identifying sources and owners, determining collection methods, categorizing data types, ensuring security, and setting refresh frequencies. This systematic approach ensures targeted and actionable insights, streamlining data collection for informed decision-making.
Incorporating these two templates can help sport organizations and software developers get better understanding of sports data and helps unlock maximal business value and actionability.

### 4.2.3 Data specification

Furthermore, Noorwali’s (2016) work emphasizes the need to integrate traditional quality attributes around performance, availability, and security with emerging considerations around volume, variety, velocity, and veracity in the context of extensive data systems. Unified specifications tying these characteristics allow the alignment of critical data properties directly with business objectives.

In sports software, data specifications can play an essential role in framing customized, scalable platforms matching data innovations and the rapid pace of workflow.

## 4.3 Requirement analysis and prioritization

Effective requirements prioritization is essential to the success of any software development project. Structured evaluation techniques enable categorizing and ranking collected system functionality, properties, and data representations based on a combination of user value and implementation effort. These analytical methods ensure that development teams focus on delivering the most critical capabilities first by business objectives and resource availability. A well-defined, collaboratively agreed-upon requirements hierarchy further facilitates shared understanding across stakeholders for the system to be.

In dynamic sports environments with frequent shifts in critical data sources, workflows, and innovation opportunities, analytical requirements planning techniques are especially essential in developing adaptable and actionable software that responds well to changes. Schmid’s (2021) paper provides techniques for requirement evaluation and prioritizing. These methods are stakeholder satisfaction-oriented thus can be valuable to the sports domain.

### 4.3.1 Technique 1: Numerical assignment

This technique involves scoring each specified requirement on simple numerical scales indicating relative importance, business value, urgency, or other weighted attributes. Typically, a 1-10 scale is employed for ease of ranking. The quantitative assignments can be grouped into segmented buckets like essential, conditional/optional, or customizable to distinguish higher priority features versus nice-to-have capabilities.
For sports software, this method provides an intuitive starting point for coaches and staff to assess needs, such as assigning a 10 for video tagging automation to save staff effort or a 7 for customized fitness dashboard views. The grouped bucketing into must-have, value-add, or luxury tiers then separates core functionality needs from additional functionalities that can be added in the future. This enables focusing on critical workflows even amidst a wide wishlist.

### 4.3.2 Technique 2: Planning game

This complementary method combines user-driven prioritization of specifications with developer feasibility analysis and effort estimation by collaborative, iterative planning. Sports coaches and analysts prioritize desired features on product roadmaps, while technical developers estimate the feasibility and effort needed to build those features. Working together in this way allows users and developers to make joint decisions on tradeoffs and determine what can realistically get built in each product phase based on priorities and constraints.
For example, coaches may prioritize a computer vision service for automated video highlight generation, but technical staff cautions on model accuracy limitations needing more data. Joint adjustment defers that wishlist item while prioritizing more mature athlete tracking analytics. This game approach ultimately aids agile responses to the inevitable new datasets, algorithms, and tools constantly emerging in the sports landscape.

### 4.3.3 Requirement Management

In Schmid's (2021) paper, a straightforward approach to handling evolving requirements is outlined. The process involves breaking down and categorizing requirements, assessing their business value, cost, and risk, and then prioritizing and ranking them. The next steps include negotiation and decision-making to effectively manage changing project needs.

## 4.4 Requirement specification and consolidation

Requirements specifications refer to the detailed documentation of functional capabilities, system properties, workflows, components, and constraints the software must support. This step enhances the quality of initial requirements collected in the elicitation step. Consolidation involves organizing these descriptions into a blueprint. Thorough specifications act as a contract between users and developers on expected system behavior and business objectives. They provide critical contexts for technical teams to integrate software platforms with niche operational environments. IEEE Standard 830-1998 outlines industry best practices that drive effective development.

Various techniques exist for crafting high-quality specifications. Wieringa (1998) paper put forth specification methods spanning graphical visualizations of relationships, decomposition of requirements into modular units, communication flow diagrams, use case models, and finite state representations. that allow capturing nuanced system behaviors and user workflows. Some of the techniques that can be adopted for sport data software are described below. The accompanying sports specifics use cases are presented in table 6.

**Decomposition specification techniques**

- **Entity relationship diagrams:** ERD's are primarily used to show data structure in a database but it can also be used to highlight connections between key entities. A relationship node in an ERD can represent any kind of relationship. It may represent a communication link but it may also represent a visibility link, a permission, a part-of link.
- **UML class diagrams:** Visualize logical structure with boxed classes, attributes, and relationships. They can capture modular software components.

**Communication-specific techniques**

Illustrate sequences and flows between system elements:

- **Data flow diagrams:** Visualize data inputs, processes/transformations, storage, and outputs. They can help in showing data pipelines and interfaces.
- **Context diagrams:** A context diagram shows the external systems with which the system under development communicates.
- **Sequence diagrams:** Represent timeline ordering of interactive messages across software objects/roles reacting to events. These diagrams can be useful for user story mapping.
- **Collaboration diagram:** A collaboration diagram is a directed graph in which the nodes represent communicating entities and the edges represent communications. The edges are numbered to represent the ordering of communications in time

**Function-specific diagrams**

- **Use case diagrams:** Map out high-level functionalities provided by the system to end-users/actors. These diagrams can help guide user story refinement.
- **Event-response specification diagrams:** These diagrams provide a simple way of specifying the functions of a system by making a list of events to which the system must respond and writing down the desired response for each event.

**Behavior-specific diagrams**

Dynamically model different states and transitions:

- **Finite state machine diagrams:** These diagrams explicitly show transition triggers between important system states. Each state represents a condition or mode of the system, and transitions represent the events or actions that cause the system to move from one state to another.
- **Extended FSM diagrams:** Enhance basic FSM diagrams with additional variables. They include additional information about actions, events, and conditions associated with each state transition.
- **State chart diagrams:** Statechart diagrams provide a more expressive and feature-rich modeling approach compared to basic STDs and even ESTDs. They are particularly well-suited for modeling complex, concurrent, and hierarchical systems where a more detailed representation of state-based behavior is necessary.

*Table 6 : Graphical Techniques for Sports data softwares*

| Graphical technique | Sport data software use case examples |
|---|---|
| Entity Relationship Diagrams | Illustrate the relationships between various actors like coaches, analysts and players. Highlighting how data is structured and interconnected in the database. |
| UML Class Diagrams | Visualize the logical structure of software components, such as modules for player statistics, team performance, and game outcomes. |
| Data Flow Diagrams | Shows the flow of data inputs, processes, storage, and outputs, helping depict the interfaces and pipelines in a sports data analytics system. |
| Context Diagrams | Outline external systems that communicate with the software, providing a high-level view of the software's interactions. |
| Sequence Diagrams | Represent the timeline of interactive messages between system objects, useful for mapping out user interactions in sports data analysis. |
| Use Case Diagrams | Visualizes the actors,such as coaches, analysts, and players, interacting with use cases, and the distinct components of the system involved in each specific use case. |
| Event-Response Specification Diagrams | List events (e.g., match outcomes) and specify the system's desired responses, aiding in defining functions related to event handling. |
| Finite State Machine Diagrams | Represent states related to specific functionalities or modes within the software.Transitions between these states can be illustrated based on events or actions, offering a dynamic visualization of the software's behavior over time. |
| State Chart Diagrams | Provide a detailed representation of complex system behaviors, allowing for modeling intricate scenarios in sports data software development. |

In sports software, requirement specifications can play an important role in framing customized, scalable requirements matching the rapid pace of workflow and data innovations. Comprehensive yet adaptable specifications can help outline system capabilities for varied end user personas while clarifying complex processes. It can allow coaches and other non-technical staff to make sense of data software. Well-documented specifications make software intricacies navigable for coaches, athletes, analysts, and administrators seeking to match diverse performance data to customizable interfaces, reports, and workflow interventions under accelerated development timelines.

## 4.5 Security and privacy requirements

With data at the core of sports data systems, security, and privacy represent crucial non-functional requirements needing specialized specification techniques. As these platforms handle athlete medical records, workout statistics, biometric sensor readings,

video footage, and other sensitive information flows, implementing rigorous access controls and usage governance is of utter importance.
However, traditional requirements analysis often overlooks fine-grained security properties in favor of features and functionality. Jutla's (2013) research bridges this gap by introducing extended UML modeling, specifically incorporating security and privacy aspects.

### 4.5.1 Technique: Extended UML diagrams

Building on standard use case diagrams commonly used to describe software behavior, new specialized extensions such as lock icons, permission notes, and encrypted data overlays allow for specifying granular security properties.
For example, video analytics use cases can define allowed data attributes, permitted model training usages, localized storage constraints, etc. Sensor data ingest use cases can encapsulate encryption protocols, access policies, and monitoring needs. These extensions accelerate the analysis of critical privacy requirements in data pipelines, ensuring they are captured earlier during specification rather than later in implementation.

Applying such extended UML techniques ensures sports data systems implement security controls aligned to unique organizational considerations like league data sharing policies, regional privacy legislations, and athlete consent limitations that traditional specifications often miss.

## 4.6 Requirement verification and validation

Requirements validation is a critical step post-specification to confirm that the documented functionalities, properties, and constraints accurately reflect the desired system behaviors and services envisioned by stakeholders. Detecting misunderstandings early on is paramount to avoiding costly rectifications later in the development process. This phase allows stakeholders and developers to revisit the specification document produced in section 4.4. Suratno's (2015) research introduces a validation and verification technique known as prototyping, which seamlessly aligns with the iterative nature of sports data software development.

### 4.6.1 Validation technique 1: Validation checks

Suratno's (2015) paper outlines five essential checks for requirements validation:

- Validity Checks: Ensuring specifications align with actual needs.
- Consistency Checks: Verifying coherence across interrelated requirements.
- Completeness Checks: Identifying all necessary capabilities.
- Realism Assessments: Evaluating the achievability of the specified scope.
- Verifiability Analysis: Assessing the testability of requirements.

### 4.6.2 Validation technique 2: Online discursive feedback

Additionally, Staple's (2016) work suggests using continuous validation frameworks tailored for adaptable systems like sports software. Online discursive feedback methods can collect annotations and dialogues around prototypes, obtaining feedback on existing system features and enabling users to communicate challenges.

### 4.6.3 Verification techniques

**Low-Fidelity Prototypes:** Sports organizations can utilize low-fidelity prototypes early in the process, consisting of simple user interface sketches and workflow diagrams. These convey the general idea to users for initial feedback.

**High-Fidelity Prototypes:** Employing higher-fidelity interactive prototypes with real graphics, content, and basic analysis functionality can rigorously demonstrate intended capabilities.

By incorporating these techniques—ranging from reviews and prototyping to feedback analysis—teams can validate requirements and identify gaps early in the development cycle when modifications are more cost-effective. These specialized methods align with the dynamic nature of sports analytics.

## 4.7 SRE model for sports data softwares

Integrating insights from the aforementioned SRE steps, a comprehensive Sports SRE Model is presented. This model prioritizes stakeholder satisfaction, emphasizing continuous end-user involvement throughout the entire process. The sequential steps are outlined below:

**Step 1:** Initiate the process by extracting insights from scenarios and construct a user story board.
**Step 2:** Progress to collaborative sensemaking using triggers outlined in Section 4.1, Technique 2. This introduces a technology perspective into the model.
**Step 3:** Employ backward analysis techniques detailed in Section 4.2 to capture data-specific scenarios. Make end-users aware of available and required data through the utilization of data templates.
**Step 4:** Once initial requirements are gathered, apply the Numerical Assessment technique from Section 4.3 to rank and categorize these requirements. Incorporate the Planning Game technique, enabling technical staff to comprehend business, risk, and cost factors associated with the requirements.
**Step 5:** Document and specify requirements using the IEEE-830 format. Utilize graphical techniques to derive use cases from functional requirements. This step ensures agreement between end-users and the software's functionalities. Any necessary changes can be made here at a low cost.
**Step 6:** Implement prototyping techniques as outlined in Section 4.6. This enables end-users to comprehend the user interface and understand all features. Conduct validation checks at this stage to ensure accuracy and alignment with expectations.

# 5. CASE STUDY OVERVIEW: SOCCERTACT

## 5.1 Overview of the Soccertact

Soccertact serves as a practical use case study in SRE. It also provides a verification and validation criteria for the RE model developed in section 4.7. It is a software solution designed specifically for collegiate-level soccer programs with ambitious data analytics goals but limited budgets and technological resources. Developed in collaboration with Travis Smith, assistant coach at The University of Kansas, Soccertact is an in-house solution initially tailored for a single program but with potential for expansion into a multi-club model. Emphasizing adaptability and customization, the software addresses evolving needs, roles, and market dynamics. Its primary objective is to offer NCAA programs a cost-effective means to leverage data capabilities, enhancing both on-field and off-field performance. Soccertact is motivated by current industry challenges, aiming to empower coaches, analysts, and staff to effectively manage, visualize, and gain insights from their data.

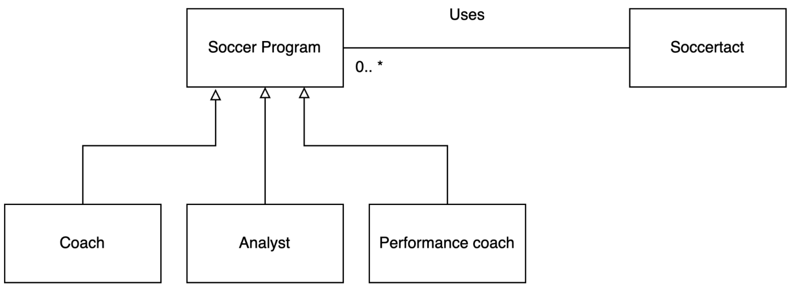


*Figure 4: Domain model for Soccertact*

## 5.2 A SRE model implementation

### 5.2.1 Step 1: Requirement elicitation from scenario

To shape Soccertact's vision, Coach Smith played a central role in employing elicitation techniques to capture coaching workflows and understand software use cases. The insights gained were then used to gather user stories, forming the core of a user story board shown in table 7. Following this foundational step, we transitioned to implementing the sensemaking model from section 4.1.2, injecting a data analytics perspective. Guided by a deepened understanding of data capabilities and technology, user stories were subsequently refined, updated, and removed. The collection of primary requirements are  rooted in user stories, and is an important step to ensure Soccertact aligns with the needs of the end user.

*Table 7: Soccertact User storyboard*

| **User storyboard** |
|---|
| As a coach I want be able to build and manage my teams schedule<br>**Acceptance criteria:**<br>Should be able to create, modify and delete events from schedule.<br>Should be able to share my schedule |
| As a coach I want to be able to get minimal and visual team and opposition reports<br>**Acceptance criteria:**<br>Provide a means to create custom reports by selecting components.<br>Reports should be visual based to help understand data. |
| As a coach I want to be able to use data to build KP'sI.<br>**Acceptance criteria:**<br>Should be able to combine stats to build KPI's |
| As a coach I want to be able to get custom timestamps from video footage<br>**Acceptance criteria:**<br>Should be able to get accurate timestamps for custom events. |

### 5.2.2 Step 2: Sense making model:

By leveraging elicitation techniques, we laid the groundwork for an understanding of coaching dynamics. The implementation of the sensemaking model further enriched our approach by infusing a crucial data analytics perspective. The user stories, refined through this process, became not just narratives but the foundations for Soccertact's primary requirements. This user-centric evolution ensures that Soccertact is not just a response to current needs but a dynamic solution that adapts to the evolving requirements of coaches and the data landscape. The initial requirements are shown in table 8.

***Table 8: Initial requirements***

| # | **Functional Requirements** | **Roles associated** |
|---|---|---|
| 1 | The system should allow to coaches to build and manage team schedules | Coach |
| 2 | The system should allow coaches and analysts to build custom video tags and extract important timestamps | Coach, Data |
| 3 | The system should allow data management capabilities | Data |
| 4 | The system should allow the development of custom reports | Coach, Data |
| 5 | The system should provide visual data representation | Coach, Data |
| 6 | The system should allow creating custom KPI | Data, Coach |
| 7 | The system should assist in maintaining a stats profile | Data |
| 8 | The system should allow data upload capabilities | Data |
| 9 | The system should assist in gaining insight from advance analysis methods and machine learning models | Data |
| 10 | The system should allow content sharing | Coach, Data, Players |
| 11 | The system should be able to utilize tracking data capabilities and analysis | Data, Coach |
| # | **Non Functional Requirements** | |
| 12 | The system should be user friendly and allow easy quick access to features | Coach |
| 13 | The system should maintain high security in regards to athletes data and private informations | Data |
| 14 | The system should provide accurate and updated data | Data |
| 15 | The system should comply to budget constrictions | Data |

### 5.2.3 Step 3: Data specific requirements

Adapting data templates and techniques from section 4.2 to the sports domain involved a backward analysis approach. Key questions about data granularity and value across sources were posed, linking requirements to diverse data types. Drawing inspiration from Narayana's (2016) template (I), shown in table 9, we tailored it for Soccertact, providing insights into data categories, sources, and security. A modified version of template (II), shown in table 10, informed end-users about data types, including cost considerations. Adjustments were made to requirements based on data costs, especially for high-cost tracking data.

***Table 9 : Data requirements***

| Req # | Data required | Data collection | Data Owners | Types of data | Security Required |
|---|---|---|---|---|---|
| 1 | Team schedule | Input, web scraping | Self | Manual | Low |
| 2 | Video/broadcast data | Video, time series data | Hudl, Spideo | Event time series data | Mid |
| 4 | Statistical data, visualization data | API's, databases, sensors | Hudl, | Event data, tracking data, performance data | High |
| 6 | Event specific data, player positioning data | API's, databases, sensors | Hudl | Event dat, tracking data | High |
| 7 | Stats data | Web scraping, API's | Hudl | Statistics and Metrics | Low |
| 8 | Files or documents | File upload | Self | CSV, XML, JSON | Mid |
| 9 | Historical data, match data, event specific data, positioning and team data | API's, sensors, databases | Hudl, VXSports, Skill corner | Event data, Statistics, Tracking data, Scouting data | High |
| 11 | Player positioning | API, sensors | Skill corner, VX | Tracking data | High |

***Table 10: Data types and sources***

| Data Type | Source | Available | Cost |
|---|---|---|---|
| Statistics | NCAA, Wyscout platform | Yes | Low |
| Event/Time Series | Wyscout API | Yes | Mid |
| Tracking | Skill corner, Wyscout API | No | High |
| Performance | Sensors, VXSports platform | Yes | Mid |
| Historical | Wyscout API | No | High |

### 5.2.4 Step 4: Requirement analysis and prioritization

Considering data and cost, some requirements were omitted or postponed, and the remaining were analyzed and prioritized. Prioritization techniques described in section 4.3 were used to group initial requirements into must-have, desired, and luxury buckets, as shown in table 11.

A planning game approach was utilized by integrating a data and development perspective. Notably, Andy Wachter, CTO, and Salman Hasni, Senior Developer, at Just Play Sports Solutions played integral roles. They were tasked with ranking and prioritizing requirements based on cost and implementation considerations. The analysis of these ratings provided valuable insights, and the end-users were informed about the associated time and cost aspects for each requirement. Functional requirements were specified and combined into initial use cases, with end-user input on UI functionality and section grouping for initial interface design. This resulted in critical use cases designed to meet role-specific tasks, combating information overload and complex interfaces.

***Table 11: Requirement ranking and priority***

| Req # | Bucket | Cost Rating | Time/Implementation rating |
|---|---|---|---|
| 1 | Desired | 3 | 3 |
| 2 | Must have | 8 | 8 |
| 3 | Desired | 5 | 3 |
| 4 | Must have | 6 | 7 |
| 5 | Must have | 5 | 2 |
| 6 | Luxury | 10 | 5 |
| 7 | Desired | 3 | 3 |
| 8 | Luxury | 4 | 2 |
| 9 | Luxury | 10 | 8 |
| 10 | Desired | 3 | 2 |
| 11 | Luxy | 10 | 4 |
| 12 | Must have | 3 | N/A |
| 13 | Desired | 5 | N/A |
| 14 | Must have | 7 | N/A |
| 15 | Must have | N/A | N/A |

### 5.2.5 Step 5: Specification and consolidation:

Functional requirements were specified and grouped to develop initial use cases for Soccetact. The end user was consulted regarding the user interface functionality and to group requirements into sections for initial interface design. These sections were developed into proper use cases and are shown in figure 5. Attention was given into grouping requirements together that meet role specific tasks. Sections were critically designed to meet workflow dedicated to roles and providing only necessary functionalities. This enables in combating challenges regarding importation bombardment and complex user interfaces. The functional requirements were grouped amongst the following use cases :

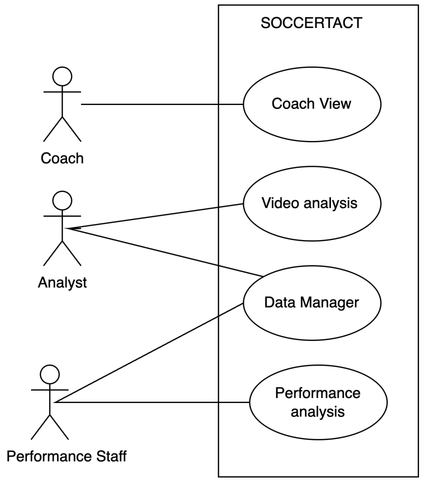


*Figure 5: User level UML*

**Coaches View:** This section should allow the coaches to manage schedules and view data reports. The coach should have the option to request and provide feedback on data reports.

**Data Manager:** This section should allow analysts to manage their data syncs, build and distribute custom reports, maintain a stats profile and process data files uploads.
**Video Analysis:** This section should allow the analyst to develop custom tags through the process event section and combination. It should provide a list of timestamps for these tags to assist in video analysis purposes.
**Performance Analysis:** This section should allow the performance analyst to upload performance specific data and develop custom reports.
`

For the scope of this paper only UML diagrams were utilized. Other diagram classes can also be used to help end-users understand about the additional functions, communication and behavior of the system. UML diagrams were utilized to help make the end user understand various features and the functionalities for each use case. The UML diagrams were extended to show security components. A SRS document can be developed that will be helpful in the specifying, tracing and changing requirements for a system. IEEE-830 can be referenced to build the document.

**Use Case 1: Coach View**
The use case outlines the functionalities and features associated with the "Coach View" scenario in figure 6. It ensures a user-friendly experience for coaches to manage events, access reports, and engage in the analysis process.

1. Showcase a list of available schedules.
2. Allow coaches to create, delete, and update events by interacting with events on the schedule.
3. Provide a user-friendly interface for coaches to choose between creating, deleting, or updating events.
4. Display a dedicated "Analysis" tab to present a list of available reports.
5. Allow coaches to select a specific report from the list.
6. Provide options for coaches to provide feedback on selected reports.
7. Provide options for coaches to request specific reports from analysts.

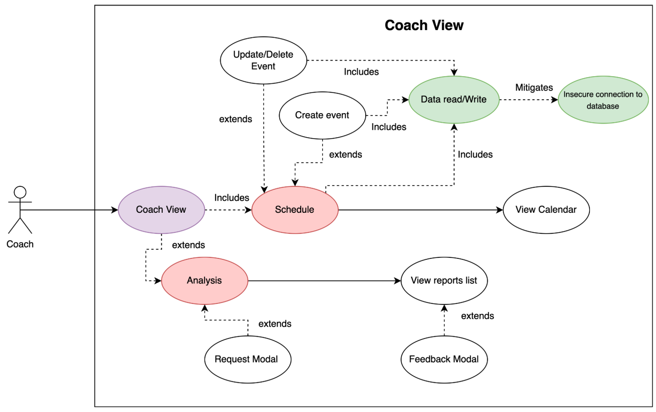


***Figure 6: Coach View - UML Use Case Diagram***

## Use case 2: Video Analysis

The use case outlines the functionalities and features associated with the "Video Analysis" scenario in figure 7**.** The use case streamlines video analysis for coaches with a dedicated tag-building section for creating new tags. Analysts can easily apply and customize these tags, retrieve associated video timestamps, and pinpoint specific moments, enhancing the overall analysis experience.

1. Display a list of available video footage events.
2. Video analysts can apply custom tags to events, allowing for personalized categorization and annotation of key moments within the video.
3. Retrieve video timestamps associated with applied tags, enabling analysts to pinpoint and revisit specific moments during the video analysis.
4. Dedicated section within the system for coaches to build custom tags.
5. Build Tags by Selecting Zone, Action Type, Possession Type, Phase Type
6. Display list of all available tags within the system

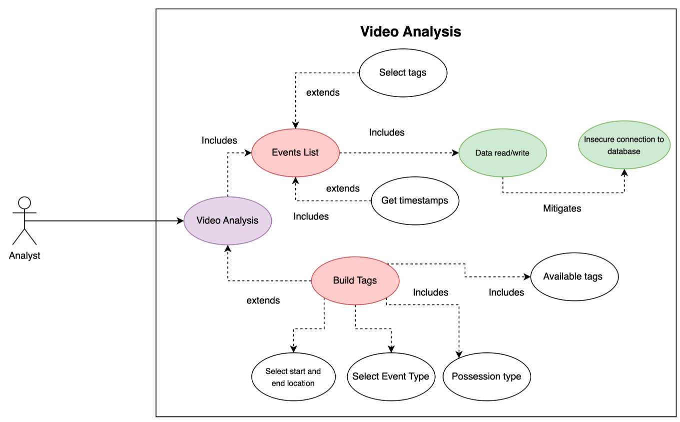


***Figure 7: Video Analysis: UML Use Case diagram***

**Use case 3: Performance Analysis**

The use case outlines the functionalities and features associated with the "Performance Analysis View" scenario in figure 8. The use case empowers analysts to build, view, and assign performance-related reports, and it also extends report builder modal.

1. Access a list of all matches and training sessions for a holistic overview.
2. Allows performance analysts to view and assign reports.
3. Dedicated tab integrated with the report builder for streamlined and efficient report creation.
4. Enable analysts to effortlessly upload data files by selecting specific data type and provider type.

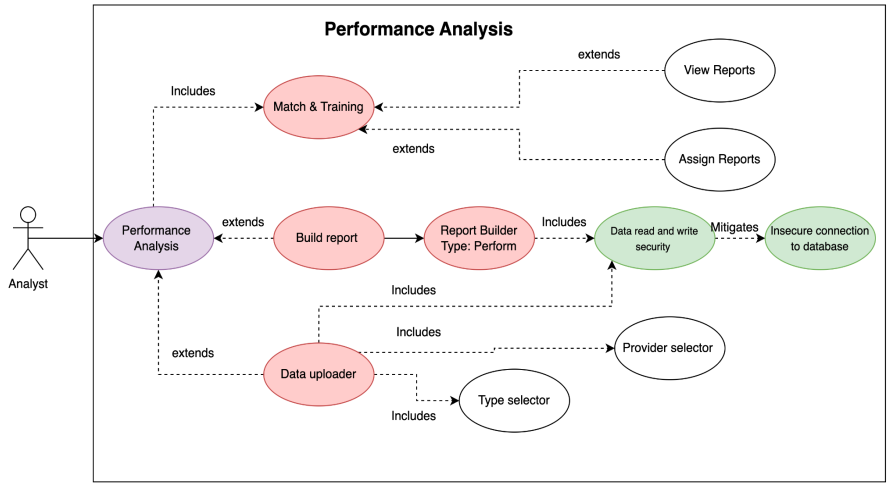


*Figure 8: Performance Analysis UML Use case*

**Use Case 4: Data Manager**

The use case outlines the functionalities and features associated with the "Data Manager" scenario in figure 9. The use case lays the foundation for a robust and versatile data management system. The modular design allows for flexibility, ensuring that current functionalities are user-friendly while also providing a framework for future expansions into advanced analytics and data modeling.
Provides analyst five modes for data insight and management:

1. **Stats profile management:**
   Analysts efficiently manage and customize stats profiles.
   Select from available statistical parameters to tailor profiles.
2. **Report viewing and building:**
   Analysts view existing reports and create new ones.
   Choose between templates or utilize the report builder modal.
3. **Data upload:**
   Seamless upload of data in various formats.
   Enhances accessibility and usability of diverse datasets.
4. **Data sync management:**
   Features for overseeing data synchronization.

Ensures consistency across different system components.

5. **Data model:**
   Anticipates future integration of data modeling and advanced analysis.
   Provides a foundation for machine learning capabilities.

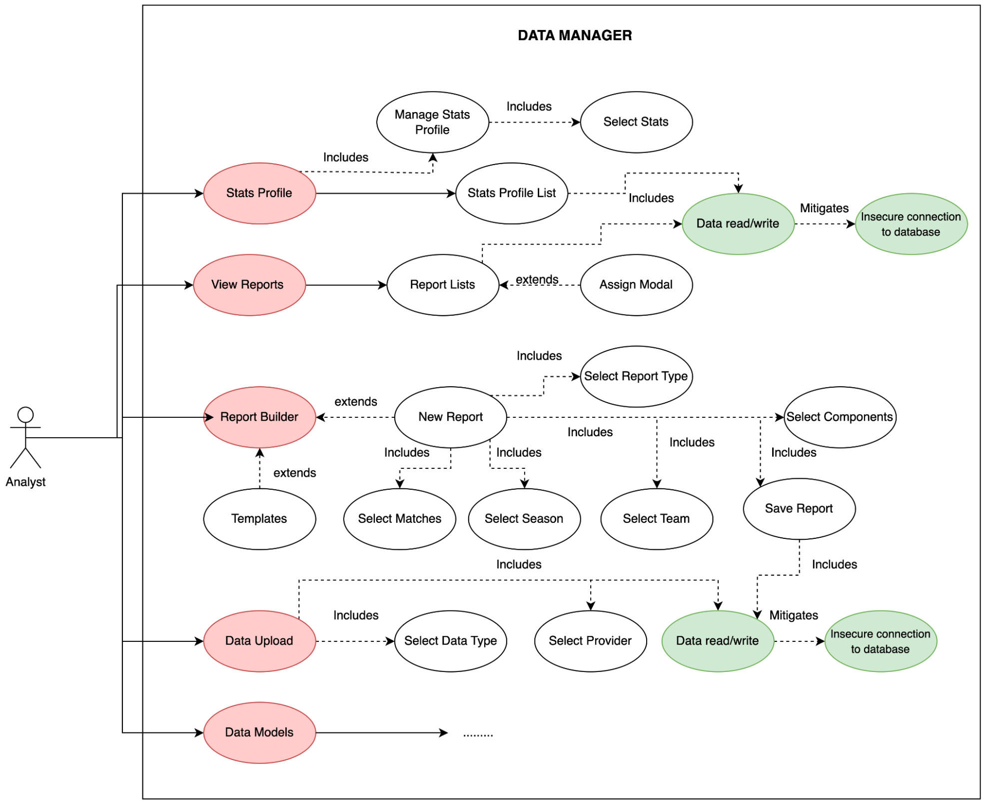


*Figure 9: Data Manager UML Use case*

### 5.2.6 Step 6: Validation and verification

Following the specification and consolidation phase, a verification process is used, involving stakeholders to ensure that the outlined requirements are aligned with the envisioned goals. This verification step played a crucial role in refining the direction of Soccertact's development, ensuring that the proposed functionalities resonated effectively with the needs of all stakeholders. To further enhance the understanding of the system's intricacies, low-fidelity prototypes were employed. These prototypes provided a visual representation of the flow between features and functionalities. Prototypes for “Coach View” (figure 10) and “Video Analysis” (figure11) tabs were presented to the end-user.

Use Case 1 : Coach View

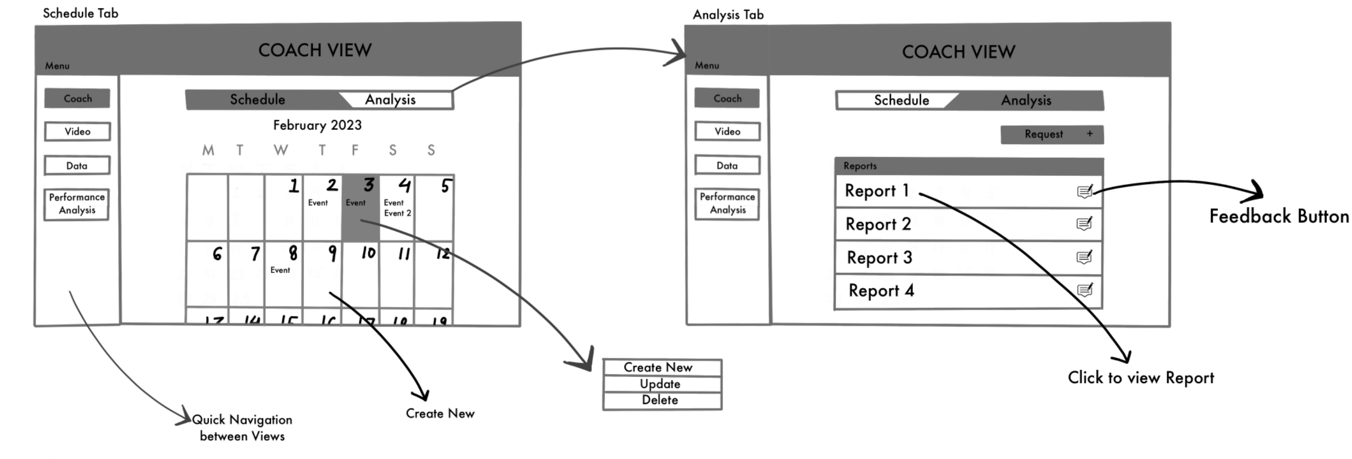


*Figure 10: Low fidelity prototype for Coach View*

Use Case 2 : Video Analysis

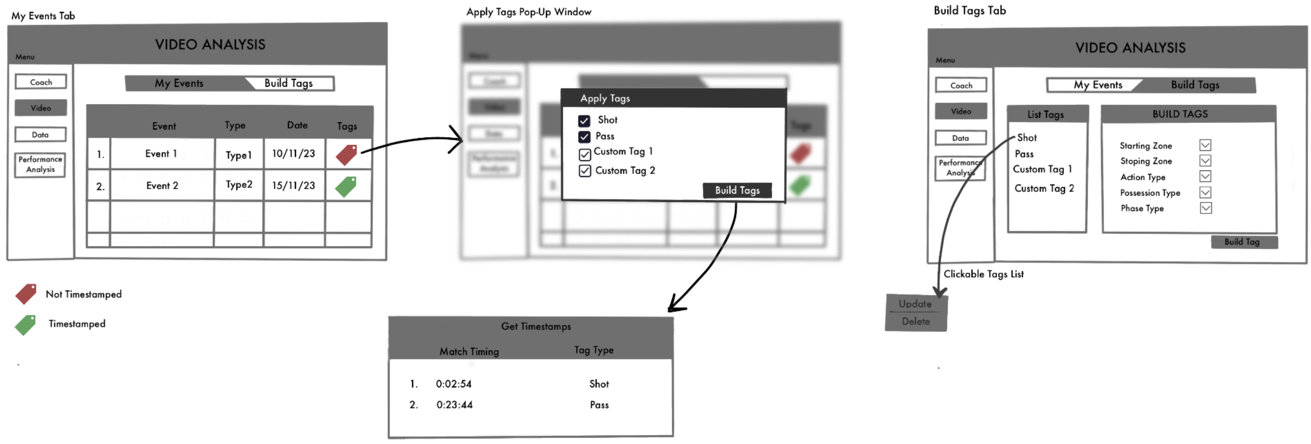


*Figure 11: Low fidelity prototype for Video Analysis*

As part of the user-centric approach, end-users were actively engaged in the consultation process, specifically regarding the functionality of the user interface. Each section was thoughtfully designed to align with distinct workflows dedicated to specific roles, offering only the functionalities deemed necessary. This approach addressed challenges related to information overload and complex user interfaces, ensuring that Soccertact provides a streamlined and user-friendly experience.

## 5.3 Evaluation and insights

Applying the proposed SRE model to formulate requirements for Soccertact has proven effective based on low modification made to initial requirements and high user satisfaction. Tracking requirement changes showed that the process accurately converted needs to requirements across the iterative phases. Soccertact stakeholders confirmed that the platform met its core objectives and demonstrated proper customization to their existing workflows and processes. Initial outcomes display the potential to transform desired capabilities elicited early on into actual aligned software implementations.

However, opportunities exist to refine this model by utilizing high fidelity prototypes, other graphical techniques for specifications, and more active participation from different roles. In scenarios involving multiple participants and diverse roles, model validation could be achieved by utilizing the Online Discursive Feedback technique described in section 4.6. This approach can serve as valuable testing data and can be implemented across all phases to collect feedback consistently. This feedback proves beneficial for both model verification and addressing requirement change requests, contributing to the efficient updating of requirements.

# 6. CONCLUSION

## 6.1 Requirement engineering for sports software

This paper pioneers a dedicated exploration of software requirement engineering in sports, providing a unique contribution to the field. The paper establishes domain understanding by studying the diverse actors, data types, and software products currently in the market. The role-specific survey served as a foundation for comprehending challenges, practices and expectations associated with software products within this domain. Existing industry software analysis further enriched our insight into the current industry. We formulated a tailored Software Requirement Engineering (SRE) model for the sports domain by synthesizing techniques from various academic papers. This model places the end-user at its core, fostering end-user contributions at each step. Finally, we applied this model to conceptualize Soccertact, a use case that validates our proposed model.

## 6.2 Future Work

Although this study is a significant step forward, there is still much room for further exploration. The inclusion of more participants could enhance model validation and contribute to refining the current techniques. Emphasis should be placed on selecting techniques that prioritize end-user centrality. Once functional requirements are modeled and confirmed, the scope of this paper can be expanded to incorporate non-functional requirements and delve into more technical aspects. This ongoing research provides a foundation for continuous enhancement and adaptation within the dynamic landscape of sports software development.

# ACRONYM LIST

RE: Requirement Engineering
SRE: Software Requirement Engineering
UML: Unified Modeling Language
FSM: Finite State Machines

## ACKNOWLEDGEMENT

I express my sincere gratitude to Travis Smith, Assistant Coach at KU Women's Soccer, for his valuable insights, contributions to the understanding of the practical needs and challenges in the sports domain and as an end-user for Soccertact.
Special thanks to Andy Watcher, Chief Technology Officer at Just Play Sports Solutions, and Salman Hasni, Senior Developer at Just Play Sports Solutions, for their expert feedback. They contributed to the requirements analysis and consolidation process by bringing a technical perspective.
I would also like to acknowledge Mohd. Asim Siddiqui, Prototype Designer, for his creative input and dedication to crafting effective prototypes that greatly enriched the validation and verification process.
Their collective efforts and expertise have significantly contributed to the success of this research project.

## REFERENCES

Al-Jaroodi, J., & Mohamed, N. (2016). Characteristics and Requirements of Big Data Analytics Applications. *2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC)*, 426-432. https://doi.org/10.1109/cic

Altarturi, H.H, Ng, K.-Y., Ninggal, M. I. H., Nazri, A. S. A., & Ghani, A. A. A. (2017). A requirement engineering model for big data software. *2017 IEEE Conference on Big Data and Analytics (ICBDA)*, 111–117. https://doi.org/10.1109/ICBDAA.2017.8284116

Arrud, D., & Madhavji, N. H. (2017). Towards a requirements engineering artefact model in the context of big data software development projects: Research in progress. *2017 IEEE International Conference on Big Data (Big Data)*, 2314–2319. https://doi.org/10.1109/BigData.2017.8258185

Chen, H.M., Kazman, R., & Haziyev, S. (2016). Agile Big Data Analytics Development: An Architecture-Centric Approach. *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 5378–5387. https://doi.org/10.1109/HICSS.2016.665

Jutla, D. N., Bodorik, P., & Ali, S. (2013). Engineering privacy for Big Data Apps with the Unified Modeling Language. *2013 IEEE International Congress on Big Data*. https://doi.org/10.1109/bigdata.congress.2013.15

Lau, L., Yang-Turner, F., & Karacapilidis, N. (2014a). Requirements for Big Data Analytics Supporting Decision Making: A sensemaking perspective. Studies in Big Data, 49–70. https://doi.org/10.1007/978-3-319-02612-1

Kuengjai, Y., & Ramingwong, L. (2021). A pilot study of requirement prioritization techniques in agile software development. *2021 4th International Conference on Computer Science and Software Engineering (CSSE 2021)*. https://doi.org/10.1145/3494885.3494912

Noorwali, I., Arruda, D., & Madhavji, N. H. (2016). Understanding Quality Requirements in the Context of Big Data Systems. *2016 IEEE/ACM 2nd International Workshop on Big Data Software Engineering (BIGDSE)*, 76–79. https://doi.org/10.1145/2896825.2896838

Otero, C. E., & Peter, A. (2015). Research Directions for Engineering Big Data Analytics Software. *IEEE Intelligent Systems, 30(1)*, 13–19. https://doi.org/10.1109/MIS.2014.76

Schmid, & Fricker, S. (2021). Evaluation of a Stakeholder Satisfaction-oriented Method for Prioritising Change Requests. 2021 *IEEE 29th International Requirements Engineering Conference (RE)*, 291–301. https://doi.org/10.1109/RE51729.2021.00033

Staples, M., Liming Zhu, & Grundy, J. (2016). Continuous Validation for Data Analytics Systems. *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*, 769–772.https://doi.org/10.1145/2889160.2889207

Suranto, B. (2015). Software prototypes: Enhancing the quality of requirements engineering process. 2015 *International Symposium on Technology Management and Emerging Technologies (ISTMET),* 148-153. https://doi.org/10.1109/ISTMET.2015.7359019

Wieringa, R. (1998). A survey of structured and object-oriented software specification methods and Techniques. *ACM Computing Surveys, 30(4)*, 459–527. https://doi.org/10.1145/299917.299919

Woodcock, Larsen, P., Bicarregui, J., & Fitzgerald, J. (2009). Formal methods: Practice and experience. *ACM Computing Surveys*, 41(4), 1–36. https://doi.org/10.1145/1592434.1592436