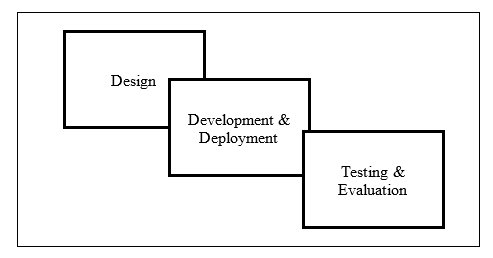
# CHAPTER THREE

# 3.0 RESEARCH METHODOLOGY /SYSTEM ANALYSIS AND DESIGN

# 3.1 Overview

The research methodology outlines the processes and techniques used for developing a digital platform to connect team managers and talented athletes, specifically in football and basketball. It outlines the system design, software requirements, platform development, testing and evaluation processes, and ethical considerations to ensure a systematic approach to building and deploying the platform. The methodology procedure is shown in Figure 3.1.



**Figure 3.1: Block Diagram of Platform Development Procedure**

# 3.2 System Design

The system design involves a thorough analysis and interpretation of the problem to create a solution that effectively addresses the identified needs. This process requires aligning the development requirements with the objectives of the platform to ensure it meets user expectations and functional goals. To achieve this, a comprehensive analysis of the necessary software, tools, and architecture is essential This section provides a detailed overview of the software requirements, system architecture, and the tools utilized for the development of the platform.

## 3.2.1 The Software Requirements

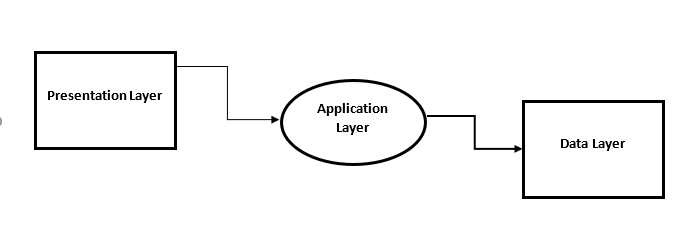
This section identifies the necessary software, libraries, and tools for the platform's development. The platform requires selecting the appropriate programming languages, integrated development environments (IDEs), framework and libraries that align with its objectives, database management system, development tools, Hosting, and the application programming interface.

1. **Programming Language**
2. **HTML (Hyper-Text Markup Language):** Forms the backbone of the platform’s web pages by defining the layout, structure, and interactive elements like forms, profiles, and navigation menus that users engage with.
3. **CSS (Cascading Style Sheets):** Enhances the visual appeal and usability of the platform by controlling design aspects such as colors, fonts, and layout, ensuring a responsive and consistent user experience across all devices.
4. **Java**: Handles server-side operations and backend processing, including managing user sessions, handling input, and ensuring smooth communication between the server and database.
5. **Python:** Used for backend development and integrating advanced features like machine learning algorithms, enabling talent analytics, and matching scouts with athletes.
6. **SQL (Structured Query Language):** Operates as the database language to store, manage, and retrieve user data, maintaining data integrity, security, and supporting complex queries for efficient information management.
7. **Frameworks and Libraries**
8. React: A JavaScript library for building dynamic and interactive user interfaces, particularly the front-end components of the platform.
9. Materialize CSS: A responsive front-end framework based on Material Design principles to provide a modern and clean look to the platform.
10. **Hosting:** A cloud platform Vercel was employed for static sites and serverless functions, chosen for its performance, scalability, and easy integration with React applications.
11. **APIs:** Pocketbase, is a backend API service used for managing the database, user authentication, and handling real-time data connections between athletes, scouts, and the platform administrator.
12. **Database Management Systems (DBMS):**
13. PocketBase is an open-source backend that manages real-time data and user authentication for web and mobile applications. It simplifies athlete and scout registration, logins, and permissions, and provides a REST API for easy integration with the React front-end.
14. MySQL; is a scalable, high-performance relational database system used to store and manage structured data like user profiles and activity logs. It supports complex queries and data analysis, ensuring reliable and efficient data handling as the platform grows.

The combination of PocketBase and MySQL, the platform benefits from both real-time data capabilities and the reliability of a mature relational database system, ensuring efficient, secure, and scalable data management.

## 3.2.2 System Architecture

This section provides a detailed explanation of the system architecture, including a block diagram that outlines the platform’s components and their interactions. The system is designed using a three-tier architecture that includes the layers represented in Figure 3.2:



**Figure 3.3: Block Diagram of Platform Architecture**

1. **Presentation Layer**

This layer consists of the user interface, developed using HTML, CSS, and React. It provides web pages for different user types (athletes, scouts, and admins), allowing them to register, log in, search, and interact. The use of Materialize CSS ensures the interface is responsive, visually appealing, and easy to navigate.

1. **Application Layer**

The core logic of the platform is handled here and built using Java and Python. This layer processes user requests, manages user sessions, handles data processing, and ensures secure communication between different parts of the system. It also integrates with APIs like Pocketbase for real-time data management, user authentication, and notifications.

1. **Data Layer**

Utilizes SQL for managing and querying the relational database. Stores user profiles, connection requests, communication logs, and other relevant data. Pocketbase API acts as a data management layer, ensuring data consistency and providing an interface for CRUD (Create, Read, Update, Delete) operations.

1. **Block Diagram Description**

The block diagram illustrates the flow of data and interaction between various components. Figure 3.4 illustrates the component's interaction, facilitating the information flow between the registered athletes and team owners.

Athlete Portal

Scout Portal

User Interface

Application Layer

Database

**Figure 3.4: Dataflow and User Interaction**

1. Users (Athletes, Scouts, Admins) access the platform through the User Interface (UI), developed with HTML, CSS, and React. The users can register and login as athletes or team owners. Their data is then stored in the connected database.
2. The User-Interface sends requests to the Application Layer, where backend processing (Java and Python) handles data processing, logic implementation, and interaction with the database. It provides an interactive page for users to send requests, either to register as a new user or request for an athlete of need.
3. The Application Layer communicates with the Pocketbase API for user authentication, data management, and notifications. It plays back and front. It serves as an intermediary by receiving user’s request and providing feedback based on the request made.
4. The Data Layer (SQL Database) stores all critical data, including user information, connection requests, and logs.
5. Admins manage connections and oversee platform activities using an admin panel, which provides them with various tools and insights into user interactions.

# 3.3 System Development and Deployment

This section outlines the development and deployment phases of the platform.

1. **Development Phases**
2. **User Interface**: This phase involves the creation of wireframes and prototypes for different user interfaces, ensuring clarity and usability. Develop a technical design document outlining the system architecture, data flow, and module interactions. The interface has a portion for Team and Athlete with fields for users to provide their information.
3. **Admin Panel:** The admin panel is developed for platform administrative management, to coordinate connections and monitor activities on the platform.
4. **Coding Phase**
5. **Front-end Development**: Use of HTML, CSS, React, and Materialize CSS to build responsive web pages.
6. **Back-end Development**: Implement server-side functionalities using Java and Python, including data processing and user authentication.
7. **Integration:** Integrate the Pocketbase API for real-time data management and user authentication processes. This thus provides optimal security of user’s information on the platform.
8. **Deployment Phase**

The platform is deployed on Vercel, leveraging its capabilities for serverless deployment and static site hosting. Configuration of the production environment, including security settings, load balancing, and monitoring services. Post-deployment monitoring will be provided to ensure the platform functions as intended and adjustments are made as necessary.

# 3.4 Testing and Evaluation

This section describes the methods and criteria used for testing and evaluating the platform. The platform is tested with 20 different individuals where 5 of them registered as team owners and 15 as an athlete. The registered users then try login access

## 3.4.1 Functional Testing

Individual functionalities, such as user registration, login, search, and data submission, were tested to ensure they perform as expected. The system's operation relies on the interaction between interconnected components (UI, API, Database) that work together seamlessly. End-to-end testing of all use cases and scenarios was conducted to ensure the overall functionality of the platform.

## 3.4.2 Usability Testing

User User testing sessions were conducted to evaluate ease of use, navigation, and overall user experience. Feedback on accessibility was collected through surveys and interviews with different user types (athletes, scouts, and admins). This information was used for a heuristic evaluation to identify and rectify usability issues.

**3.4.4 Evaluation Metrics**

To assess the platform's performance and identify areas for improvement, the following key performance indicators (KPIs) are defined and measured

1. System response time
2. Users Satisfaction
3. Error Rate
4. Task Completion Rate

## 3.5 Ethical Considerations

* 1. **Informed Consent:** Participants are provided with detailed information about the study, including its purpose, procedures, and potential risks. Informed consent is obtained in writing for surveys and interviews, with an option to withdraw at any stage without any penalty.
  2. **Confidentiality and Anonymity:** All personal information and responses are anonymized to protect participants’ identities. Data is stored securely in encrypted formats and is only accessible to authorized researchers.
  3. **Voluntary Participation:** Participation is entirely voluntary, and participants are informed that they can withdraw from the study at any time. No incentives are offered that might coerce participation, ensuring that consent is genuinely voluntary.
  4. **Ethical Review:** The research proposal is reviewed and approved by an Institutional Review Board (IRB) or an equivalent ethics committee to ensure that it adheres to ethical standards in research.
  5. **Bias Mitigation:** To mitigate potential bias, multiple researchers are involved in the data analysis process. For qualitative analysis, intercoder reliability is ensured by having multiple coders independently analyse the data before comparing results.