p-ISSN : 2685-9106
e-ISSN : 2686-0384

Analysis and Development of a Football Scouting App based on Flutter: A Case Study of A3N

Andri Zefrinaldi[1], Nungki Selviandro[2], Gia Septiana Wulandari[3]
Faculty of Informatics, Telkom University, Bandung, Indonesia[1,2,3]
e-mail: andrizefrinaldi@student.telkomuniversity.ac.id[1], nselviandro@telkomuniversity.ac.id[2] , giaseptiana@telkomuniversity.ac.id[3]

Author Notification
16 January 2024
Final Revised
31 January 2024
Published
05 February 2024

Zefrinaldi, A., Selviandro, N., & Wulandari, G. S. (2024). Analysis and Development of a Football Scouting App based on Flutter: A Case Study of A3N. ADI Journal on Recent Innovation, 5(2), 181–191.
DOI : https://doi.org/10.34306/ajri.v5i2.1051

Abstract

Scouting is a crucial aspect of football, where the team's success depends heavily on the ability to select the right player. To minimize risk and optimize resources, teams can utilize the football scouting application. Unfortunately, existing applications still have limitations, especially in support of multiple platforms, and are open source. Therefore, this research is focused on developing a football scouting application that supports multiple platforms and is open source, using a single codebase-based Flutter Framework. Using Flutter, the app can be created based on just one code base, which can be compiled directly on Android and iOS platforms. The developed application has several superior features. First, the application allows the admin to insert player data into the database. Another feature is the ability to search for players based on names that users include. In addition, the application is also able to display a list of players based on the position chosen by the user. Finally, the application can display the complete data and statistics of the currently active player. Through the implementation of the selected development method, the implementation of this system can contribute significantly to supporting the scouting process in football teams. Thus, this application can increase the efficiency and effectiveness of decision-making within the scope of football scouting activities.

Keywords: *Application, Flutter, Multiplatform; Open Source, Scouting*

1. Introduction

Scouting is a crucial process in the world of football, where the selection of the right players has a major impact on the success of the team [1]. Football scouts serve as intermediaries between potential players and prestigious clubs, with the primary task of identifying talents who can strengthen the club [2]. In many countries, a major role in identifying young talent lies with coaches and scouts, giving an edge in investing in players who are considered "proper" to be "proper" [3]. A key criterion in scouting is integration in structured systems, by the methods for understanding observed markets [4]. In elite clubs, scouting roles can vary from local to strategic across continents [5]. The growth of the scouting industry is driven by the use of data analysis and statistics in player identification [6]. Scouting is essential in recruiting the best players, especially in Italian football, with new technology providing advantages in match prediction and tactical analysis [7].

There are already many existing scouting applications and analysis has been done by comparing them with several comparator variables, such as usability, open source, and multi-platform. It turns out that very few applications support multiple platforms and have open-source access[8]. This is necessary. Multi-platforms are especially needed today because most of them already use tools and do not have to go through the website alone[9]. Therefore, multi-platform support applications are required. One of them is using Flutter. Flutter is a multi-platform framework targeting the development of high-performance mobile applications [10]. Multiplatform application creation makes it easier for application developers to design and run applications on multiple platforms using a single code base [11]. Developers using Flutter wrote the application in Dart language, and the Flutter engine converted the application into the original ARM code integrated with the Flutter runtime engine. This process allows applications to run uniformly on Android and iOS platforms by rendering the user interface to the app [12]. Thus, in this research, application development was carried out using Flutter. In addition, the application also has open-source access. Because some clubs in Indonesia can have limited funds to purchase these paid applications, it would be better if some were free. In terms of usability, no application displays the full attributes of a sufficient number of players [13]. Therefore, this can be an obstacle for scouts to use the application, even though the function of this application is very good for making decisions in purchasing players. Furthermore, the results of the comparative analysis of applications are in the current comparison of mobile scouting applications[9].

Based on the background outlined, the problem formulation in this research covers two main aspects. First, how to develop a football scouting application that supports multi-platform using Flutter? Second, how to create an application with open-source access to provide ease to football clubs that have limited funds? By addressing this question, the research aims to develop a football scouting application using Flutter that not only supports multi-platform but also open-source applications[14]. The goal is to provide scouting opportunities while providing a limited-budget option for football clubs to utilize the app without the need to charge a fee for the app [15].

Several studies highlight the development of applications in various contexts. Ahmad Zulyaden, Rahma Dewi, and Afri Tantri revealed that Android-based football talent scouting applications with sports search methods are indispensable to football coaches aged 11-14. Their research showed that coaches in 14 football schools in Medan generally have Android phones, but have not used them in football training, so digital applications for football scouting talent tests are considered important [16]. On the other hand, Muhammad Hendriawan and his team managed to develop e-commerce applications at PT. Putra of Sumber Abadi uses Flutter with a waterfall software development method. Although it still needs improvement, it is considered quite easy to use, attractive, and has the potential to increase the company's sales and profits [17]. In addition, research by Muhammad Fauzi, Agus Tedyyana, and Fronti Enda highlighted the mobile disaster response applications developed using the Flutter framework. Evaluation shows that the app can help people cope with disaster problems with good user satisfaction and capabilities, can be run hybrids on Android and iOS platforms, and contributes significantly to preparing people for disasters [18]. Scouting consists of planning games to follow, identifying flaws in the team structure, outlining reports, and conducting observations of the pitch. As such, Scouting can be seen as a means of minimizing losses and monetizing resources, being able to offer information about the team itself, the opposing team, the referee's acting style, and the ecological environment outside the game [19].

Flutter is a cross-platform framework used to develop applications on various platforms such as Android, iOS, Windows, Linux, Mac, and Google Fuchsia. Flutter uses the Dart programming language also developed by Google [20]. Dart is an open-source programming language developed, recognized as a standard by ECMA, and widely used for fast application development on various platforms. The language is object-oriented with the C-style syntax. Dart's advantages include its ability to easily create one-page applications [21]. Flutter, described by Google as a portable UI toolkit, uses Dart to create beautiful native compilation applications for mobile, web, and desktop from a single codebase. With built-in support for iOS and Android mobile platforms, Flutter is defined as a powerful versatile open UI tool by Tim Sneath, Product Manager Group at Google [22]. Flutter brings widgets and renderings from the platform into the application, which allows them to be customized and expanded [23]. The widget is a basic structure box of the Flutter application User Interface (UI). Each widget is an unchanging confirmation of the UI section [24]. For flutter testing, widget testing, or component testing, one widget tests to verify that the behavior, interaction, and UI of the widget match the expected results [25].

2. Research Method

The following system designs were used to build the systems to be built in the research. Here's the plot, that's the process of the system being built.

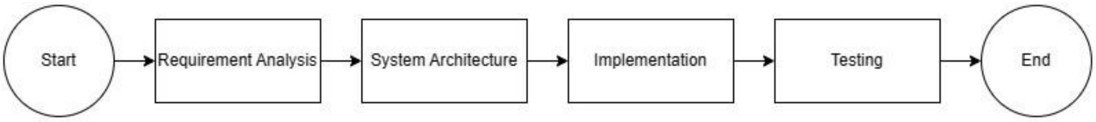


Figure 1. System Design

2.1 Requirement Analysis

Requirement analysis explains the various analyses related to the application to be created. This analysis phase aims to obtain the required information and obtain the concept of the application to be created [26].

2.1.1 Functional Requirements

In the context of application development, functional requirements refer to specific features or functions that must be present in the application to meet user goals and needs [27]. Functional requirements help define what the application should do. The following are the functional requirements for the applications made in this research.

Table 1. Functional Requirements

Function	User	
	Admin	Scout
Register	✓	✓
Login	✓	✓
View player statistics	✓	✓
Search for players by name	✓	✓
View players by position	✓	✓
Add a player	✓	x
Update player data	✓	x
Remove player	✓	x

2.1.2 Non-Functional Requirements

Non-functional requirements in application development refer to aspects that are not directly related to application-specific features or functions but affect user performance, security, and experience. The following are non-functional requirements for the applications made in this research.

Table 2. Non-Functional Requirements

Parameters	Requirements
Availability	This application must be used continuously for 24 hours per day, 7 days per week. Applications use a mobile base and are accessed by users at different times.
Reliability	This application must be run without errors due to the ignorant user by randomly and randomly clicking the application. Tolerable failures are at most 5%.
Agronomy	This application must have a high ergonomic value and can be conveniently used by users with an easy-to-understand and easy-to-operate interface.
Portability	It can be accessed anywhere and anytime as long as it has a mobile device and has Internet access to use the features within it.
Response time	This application must be able to load user-click features for at least 3 seconds. The loading process between these application features can also be affected by external factors such as poor internet access, near-full storage space, or also by small RAM.
Security	This application must be able to use the security system using an encrypted password.
Language	This application must be able to use its native English language to be used universally.

2.1.3 Use Case Diagram

Use case diagrams are a type of UML (Unified Modeling Language) diagram used to describe the interaction between an actor (user or external system) with a system or application. This diagram is helpful in modeling system functionality from a user or stakeholder perspective. The following is a use case diagram of the application made in this research.

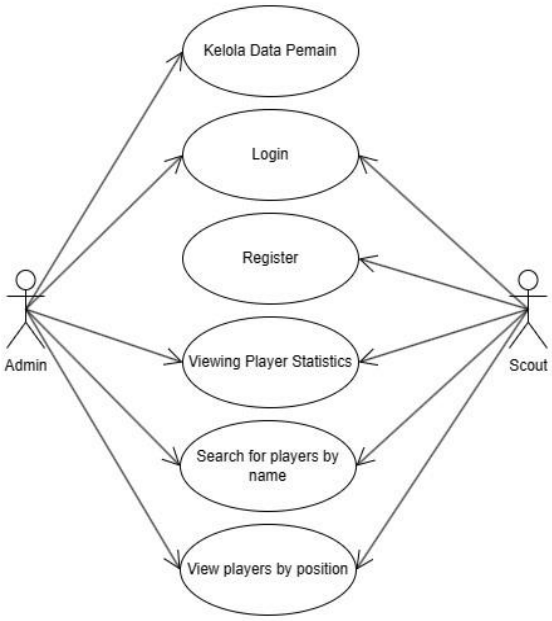


Figure 2. Use Case Diagram

2.1.4 Class Diagram

Class diagrams are a type of UML (Unified Modeling Language) diagram used to model the static structure of a system or application, especially in terms of class, object, attributes, and interclass relationships. This is a class diagram of the applications made in this research.

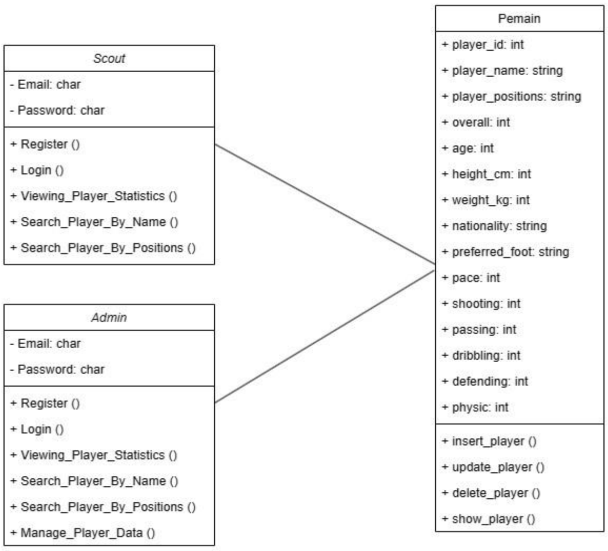


Figure 3. Class Diagram

2.2 System Architecture

The subsequent phase of this research involves the meticulous crafting of a system architecture, a fundamental aspect in the realm of application development. System architecture delineates the organizational structure of key elements and their intricate interactions, strategically devised to efficiently accomplish the specified objectives of the application. Serving as a foundational blueprint, this architectural framework establishes a coherent work plan and extends invaluable guidance to developers throughout the entire application development process. The ensuing paragraphs expound upon the intricacies of the system architecture for the applications developed within the ambit of this research.

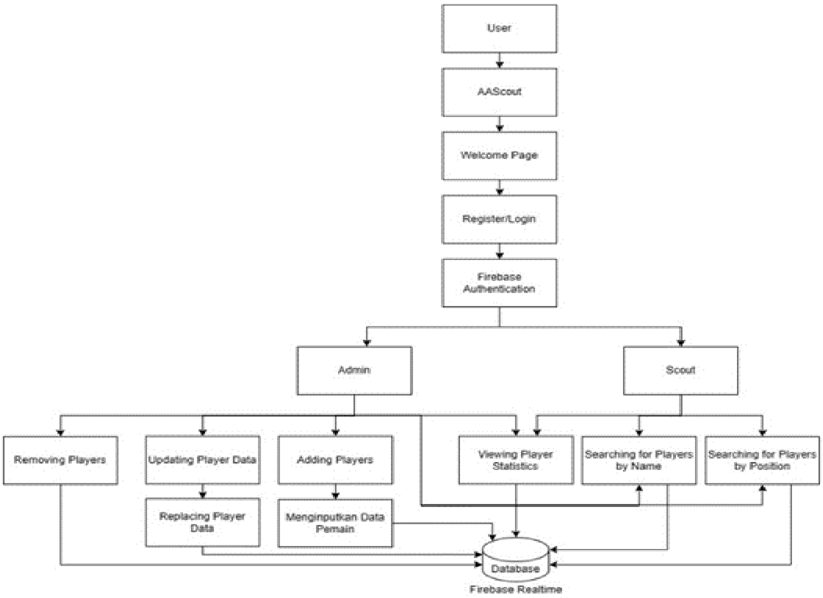


Figure 1. System Architecture

The initial facet of the architecture pertains to user authentication, wherein Firebase Authentication is adeptly leveraged to ensure secure user registration and login. Users are mandated to register or log in using email and password authentication methods, contributing to the overall security and integrity of the application. Subsequently, the architecture incorporates the dynamic retrieval of a curated list of 300 players upon successful user authentication. This player list is intelligently generated based on the highest overall rating procured from the EA Sports FC 24 API, thereby presenting users with a personalized and relevant experience. The cornerstone of the architecture lies in the utilization of the Firebase Realtime Database, functions as the central repository for the application. This pivotal component orchestrates real-time synchronization, ensuring instant updates across diverse devices. Furthermore, it seamlessly scales to accommodate a burgeoning user base and evolving data requirements. The database's versatility is underscored by its provision of REST API access, enabling seamless integration with the application. Integrated authentication safeguards secure access to the database, and the provision of offline support ensures users can access critical data even in the absence of an active internet connection.

Administrator access, a critical facet of the architecture, empowers administrators with comprehensive control to efficiently manage player data. This privileged access is seamlessly facilitated through integration with the Firebase Realtime Database, enhancing administrative efficacy. The integration of Flutter for front-end development and Firebase for back-end services forms the crux of the architecture. Flutter's prowess in delivering a responsive and efficient user interface dovetails seamlessly with Firebase's capabilities in managing data storage, authentication, and real-time updates. In summation, the user experience is a focal point, and the amalgamation of Flutter and Firebase within the architecture ensures the delivery of reliable and efficient applications. Users are greeted with a positive and responsive interface, augmenting overall satisfaction. This meticulously designed system architecture thus harmonizes the strengths of front-end and back-end technologies, culminating in a well-integrated and efficient application ecosystem. Noteworthy Firebase features, including Realtime Database and Authentication, play a pivotal role in ensuring the seamless functionality of the application, ultimately providing users with a reliable and gratifying experience.

2.3 Implementation

The subsequent phase of this research involves the practical implementation, with a primary focus on application development utilizing Flutter and the utilization of the Firebase Realtime Database for data storage. The comprehensive source code is accessible at https://github.com/zefrinaldi/Tugas-Akhir.git.

2.4 Testing

The concluding phase of this research entails testing, specifically employing the black box testing method. Black box testing is a software testing approach where evaluations are conducted without considering the internal structure or logic of an application's code. This method is employed to assess the functionality and behavior of the application, irrespective of its detailed implementation.

3. Findings

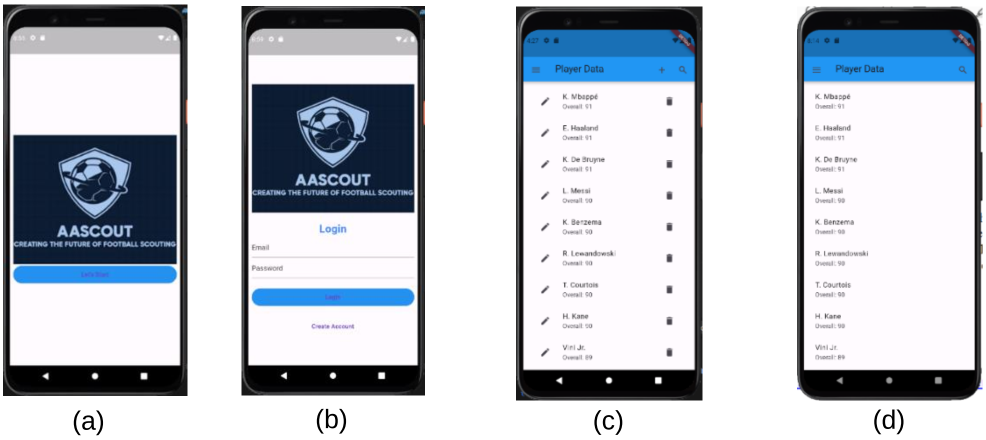
Based on the research conducted, the results are presented in this section, comprising two sub-sections: Problems and Research Implementation. If there are additional findings, additional sub-sections can be added below.

3.1 Database Implementation

The creation of the database is executed using the Firebase Realtime Database. The database encompasses 300 players, ranked based on the highest overall rating. Access to the database is available at https://tugasakhir-41d43-default-rtdb.firebaseio.com/.

3.2 Interface Implementation

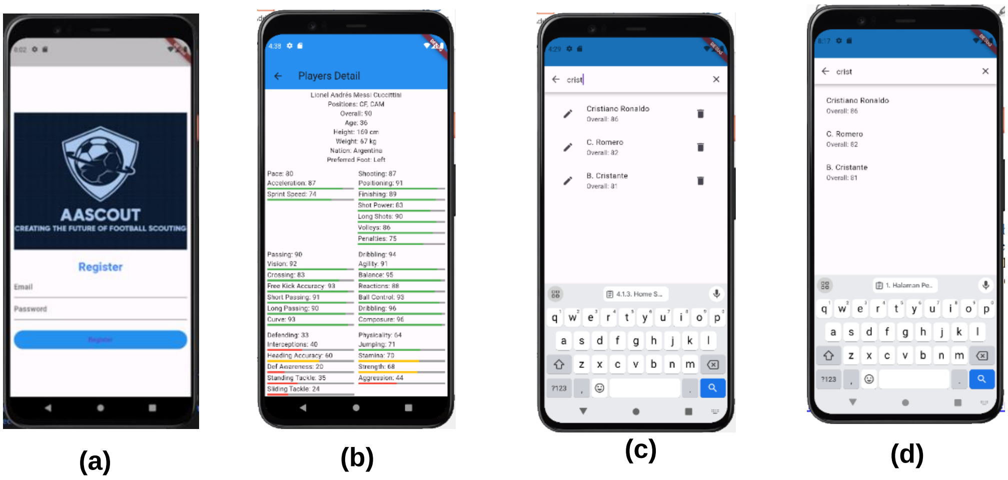
The admin interface contains a Welcome page, Login Page, Home Page, Player Search Page, Player Filter Page, Player Add Page, Player Update Page, Player Delete Page, and Player Details Page. The scout interface includes a Welcome page, Login Page, Register Page, Home Page, Player Search Page, Player Filter Page, and Player Details Page.



K. Mbappé
Overall: 91
E. Haaland
Overall: 91
K. De Bruyne
Overall: 91
L. Messi
Overall: 90
K. Benzema
Overall: 90
R. Lewandowski
Overall: 90
T. Courtois
Overall: 90
H. Kane
Overall: 90
Vini Jr.
Overall: 89

(a) (b) (c) (d)

Figure 6. Welcome page (a), Login Page (b), Admin Home Page (c), and Scout Home Page (d)



Lionel Andrés Messi Cuccittini
Positions: CF, CAM
Overall: 90
Age: 36
Height: 169 cm
Weight: 67 kg
Nation: Argentina
Preferred Foot: Left

Pace: 80
Acceleration: 87
Sprint Speed: 74

Shooting: 87
Positioning: 91
Finishing: 89
Shot Power: 83
Long Shots: 90
Volleys: 86
Penalties: 75

Passing: 90
Vision: 92
Crossing: 83
Free Kick Accuracy: 93
Short Passing: 91
Long Passing: 90
Curve: 93

Dribbling: 94
Agility: 91
Balance: 95
Reactions: 88
Ball Control: 93
Dribbling: 96
Composure: 96

Defending: 33
Interceptions: 40
Heading Accuracy: 60
Def Awareness: 20
Standing Tackle: 35
Sliding Tackle: 24

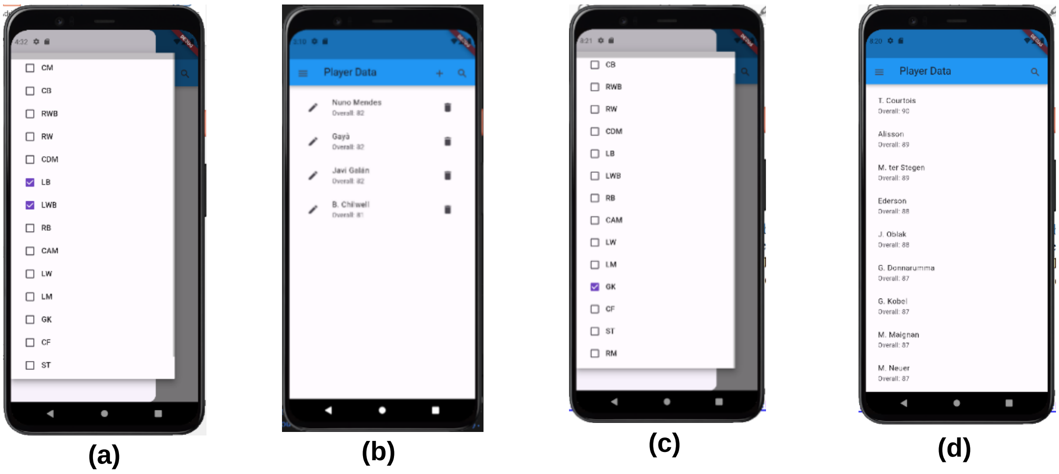
Physicality: 64
Jumping: 71
Stamina: 70
Strength: 68
Aggression: 44

crist

Cristiano Ronaldo
Overall: 86
C. Romero
Overall: 82
B. Cristante
Overall: 81

(a) (b) (c) (d)

Figure 7. Register Page (a), Player Detail Page (b), Admin Player Search Page (c), and Scout Player Search Page (d)



□ CM
□ CB
□ RWB
□ RW
□ CDM
☑ LB
☑ LWB
□ RB
□ CAM
□ LW
□ LM
□ GK
□ CF
□ ST

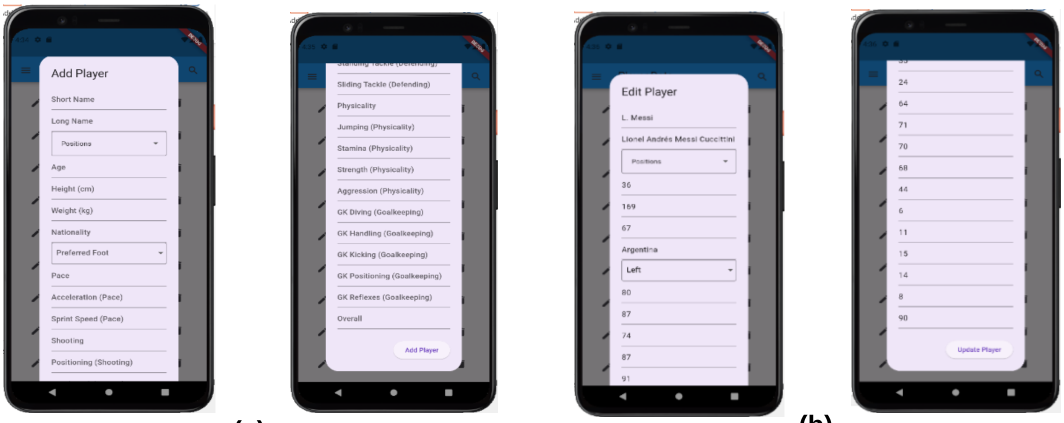
Nuno Mendes
Overall: 82
Gayà
Overall: 82
Javi Galán
Overall: 82
B. Chilwell
Overall: 81

□ CB
□ RWB
□ RW
□ CDM
□ LB
□ LWB
□ RB
□ CAM
□ LW
□ LM
☑ GK
□ CF
□ ST
□ RM

T. Courtois
Overall: 90
Alisson
Overall: 89
M. ter Stegen
Overall: 89
Ederson
Overall: 88
J. Oblak
Overall: 88
G. Donnarumma
Overall: 87
G. Kobel
Overall: 87
M. Maignan
Overall: 87
M. Neuer
Overall: 87

(a) (b) (c) (d)

Figure 8. Admin Player Filter page along with results (a) and Scout Player Filter page along with results (b)



Add Player
Short Name
Long Name
Positions
Age
Height (cm)
Weight (kg)
Nationality
Preferred Foot
Pace
Acceleration (Pace)
Sprint Speed (Pace)
Shooting
Positioning (Shooting)

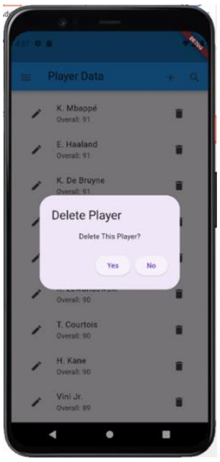
Sliding Tackle (Defending)
Physicality
Jumping (Physicality)
Stamina (Physicality)
Strength (Physicality)
Aggression (Physicality)
GK Diving (Goalkeeping)
GK Handling (Goalkeeping)
GK Kicking (Goalkeeping)
GK Positioning (Goalkeeping)
GK Reflexes (Goalkeeping)
Overall
Add Player

Edit Player
L. Messi
Lionel Andrés Messi Cuccittini
Positions
36
169
67
Argentina
Left
80
87
74
87
91

24
64
71
70
68
44
6
11
15
14
8
90
Update Player

(a) (b)

Figure 9. Add Player Page (a) and Update Player Page (b)



Player Data
K. Mbappé
Overall: 91
E. Haaland
Overall: 91
K. De Bruyne
Overall: 91
Delete Player
Delete This Player?
Yes No
Overall: 90
T. Courtois
Overall: 90
H. Kane
Overall: 90
Vini Jr.
Overall: 89

Figure 10. Remove Player Page

In Figure 6(a), the Welcome page functions as the initial interface upon opening the application. Commonly referred to as the Start page, this section features the application logo, denoted as the "Ascout" app, occupying approximately one screen and visible upon application launch. A "Let's Start" button is provided, enabling users to initiate access to the application, subsequently redirecting them to the login page.

In Figure 6(b), the Login page acts as a gateway to limit user permissions for viewing and interacting with data. Access is restricted to registered users. For administrative login, users must input the designated email and password provided by the researcher to be identified as having an "admin" role. Admins possess the capability to manage player data, including player additions, updates, and deletions. Users without an account are prompted to create one by selecting the "Create Account" option, redirecting them to the register page.

Figure 6(c) showcases the Admin Home Page, the initial interface presented to administrators post-successful login. This page displays a list of 300 players based on the highest overall rating, featuring options for adding players, updating player data, and deleting players through corresponding icons. Additionally, an integrated search feature enables administrators to locate specific players by name. The page also provides a position-based player filtering option accessed through a horizontal three-line icon.

Figure 6(d) depicts the Scout Home Page, differing from the Admin Home Page in player data management. While scouts can view player details, they lack the capability to add, update, or delete players. In Figure 7(a), the Register Page is crucial for user registration, often bundled with the login page as part of the Authority feature. This page necessitates validation, requiring users to complete both the email and password fields. Duplicate emails are disallowed. After successful registration, users are directed to the login page.

Figure 7(b) introduces the Player Details page, offering a comprehensive overview of a player's biography and attributes. In-depth explanations of player attributes can be found at https://community.sigames.com/sigames-manual/football-manager-2024/players-r4958/. Figures 7(c) and 7(d) showcase the Player Search Page on the Admin and Scout interfaces. Triggered by the search icon, users input player names to generate a tailored list displaying relevant player elements based on the search criteria.

In Figures 8(a) and 8(b), the Player Filter Page for Admin and Scout interfaces emerges after selecting the horizontal triple-line icon. This page presents a list of 15 player positions, allowing users to filter players by selecting one or more positions and subsequently displaying a list of players based on the chosen positions.

Figure 9(a) introduces the Add Player Page, featuring biodata and detailed attributes. Admins must input player statistics such as name, position, age, height, weight, and other pertinent information. An "Add Player" button facilitates the insertion of player data into the database. In Figure 9(b), the Player Update Page enables administrators to modify existing player data. An "Update Player" button allows for the replacement of altered player information.

Figure 10 illustrates the Delete Player page, designated for the removal of players from the database. The page offers a "Yes" button to confirm player deletion and a "No" button to cancel the deletion process.

3.2 Result Analysis

The comprehensive system testing results affirm that the utilization of the Flutter framework in developing the Football Scouting App has proven to be effective. This framework successfully facilitates the creation of applications that support multi-platform functionality and adhere to open-source principles. Through the implementation of Flutter, the application has demonstrated its ability to offer solutions that are accessible across various platforms, enhancing its versatility and reach. Moreover, the adoption of Flutter has provided additional advantages in terms of open-source development.

Hence, the decision to select Flutter as the development framework for the Football Scouting App can be deemed appropriate. It not only fulfills the requirements of a multi-platform application but also aligns with the principles of inclusive and open software development. The successful outcomes of the system test underscore the efficacy of Flutter in meeting the project's objectives and requirements.

4. Conclusion

In conclusion, the Scouting implementation system has demonstrated successful functionality in assisting scouts with player filtering, achieving the intended objectives of the application development. The user-centric design enhances the accessibility of comprehensive player data and attributes, with a primary advantage lying in its broad accessibility across platforms, including Android and iOS, and its open-source nature. However, certain limitations have been identified, warranting attention in future development efforts. Notably, the current research faced constraints related to equipment limitations, preventing the display of iOS testing results. It is recommended that future work includes comprehensive testing and presentation of results for the iOS version to ensure a more thorough evaluation of the application's performance on both major platforms.

Furthermore, to enrich the user experience and attract a wider user base, future development should focus on incorporating additional features. These enhancements may include more engaging design elements and the introduction of diverse filters, such as those for female players, countries, player prices, and other relevant parameters. Such augmentations would contribute to a more versatile and appealing mobile app, thereby providing increased value to users and potential customers in the future.

References

[1] S. Ghar, S. Patil, and V. Arunachalam, "Data Driven football scouting assistance with simulated player performance extrapolation," in *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2021, pp. 1160–1167.

[2] F. Pretto, "Development of a Football Analytics Web Application for Player Scouting," 2022.

[3] S. Lazarević, J. Lukić, and V. Mirković, "Role of football scouts in player transformation process: From talented to elite athlete," *Sport. i praksa*, vol. 10, no. 1, pp. 65–79, 2020.

[4] M. Pouyandekia and Z. Memari, "How can clubs play an essential role in talent identification and development programmes? A case study of German football," *Sport Soc.*, vol. 25, no. 9, pp. 1679–1713, 2022.

[5] A. B. Evans *et al.*, "Qualitative research in sports studies: Challenges, possibilities and the current state of play," *European Journal for Sport and Society*, vol. 18, no. 1. Taylor & Francis, pp. 1–17, 2021.

[6] A. Sprake and C. Palmer, "Understanding the interpretive paradigm: a guide for sports students learning through qualitative research," *J. Qual. Res. Sport. Stud.*, vol. 16, no. 1, pp. 45–68, 2022.

[7] E. Radicchi and M. Mozzachiodi, "Social talent scouting: A new opportunity for the identification of football players?," *Phys. Cult. Sport. Stud. Res.*, vol. 70, no. 1, pp. 28–43, 2016.

[8] M. H. R. Chakim, M. Hatta, A. Himki, A. R. A. Zahra, and N. N. Azizah, "The Relationship Between Smart Cities and Smart Tourism: Using a Systematic Review," *ADI J. Recent Innov.*, vol. 5, no. 1Sp, pp. 33–44, 2023.

[9] E. A. Beldiq, B. Callula, N. A. Yusuf, and A. R. A. Zahra, "Unlocking Organizational Potential: Assessing the Impact of Technology through SmartPLS in Advancing Management Excellence," *APTISI Trans. Manag.*, vol. 8, no. 1, pp. 40–48, 2024.

[10] A. Tashildar, N. Shah, R. Gala, T. Giri, and P. Chavhan, "Application development using flutter," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 2, no. 8, pp. 1262–1266, 2020.

[11] M. A. Faiz, D. S. Kusumo, and M. J. Alibasa, "Flutter Framework Code Portability Measurement on Multiplatform Applications with ISO 9126," in *2022 1st International Conference on Software Engineering and Information Technology (ICoSEIT)*, 2022, pp. 36–40.

[12] M. Gonsalves, "Evaluating the mobile development frameworks Apache Cordova and Flutter and their impact on the development process and application characteristics," 2019.

[13] B. Bhima, A. R. A. Zahra, T. Nurtino, and M. Z. Firli, "Enhancing Organizational Efficiency through the Integration of Artificial Intelligence in Management Information Systems," *APTISI Trans. Manag.*, vol. 7, no. 3, pp. 282–289, 2023.

[14] D. Jonas, E. Maria, I. R. Widiasari, U. Rahardja, and T. Wellem, "Design of a TAM Framework with Emotional Variables in the Acceptance of Health-based IoT in Indonesia," *ADI J. Recent Innov.*, vol. 5, no. 2, pp. 146–154, 2024.

[15] I. Handayani, D. Apriani, M. Mulyati, N. A. Yusuf, and A. R. A. Zahra, "A Survey on User Experience of Blockchain Transactions: Security and Adaptability Issues," *Blockchain Front. Technol.*, vol. 3, no. 1, pp. 160–168, 2023.

[16] A. Zulyaden, R. Dewi, and A. Tantri, "Football Talent Scouting Application Development 'Sport Search' Method based on Android," in *Proceedings of the 7th Annual International Seminar on Transformative Education and Educational Leadership, AISTEEL 2022, 20 September 2022, Medan, North Sumatera Province, Indonesia*, 2022.

[17] M. Hendriawan, T. Budiman, V. Yasin, and A. S. Rini, "Pengembangan aplikasi e-

commerce di pt. putra sumber abadi menggunakan flutter," *JISICOM (Journal Inf. Syst. Informatics Comput.*, vol. 5, no. 1, pp. 69–88, 2021.

[18] M. Fauzi, A. Teddyyana, and D. Enda, "Pengembangan Aplikasi Mobile Tanggap Bencana Di Kab. Bengkalis Menggunakan Framework Flutter," *Zo. J. Sist. Inf.*, vol. 3, no. 1, pp. 27–36, 2021.

[19] T. Vilela, F. Portela, and M. F. Santos, "Towards a Pervasive Intelligent System on Football Scouting-A Data Mining Study Case," in *Trends and Advances in Information Systems and Technologies: Volume 3 6*, 2018, pp. 341–351.

[20] H. Hussain, K. Khan, F. Farooqui, Q. A. Arain, and I. F. Siddiqui, "Comparative Study of Android Native and Flutter App Development," *Memory*, vol. 47, pp. 36–37, 2021.

[21] U. U. A. S. Swathiga, P. Vinodhini, and V. Sasikala, "An Interpretation of Dart Programming Language," *DRSR J.*, vol. 11, no. 3.

[22] L. Dagne, "Flutter for cross-platform App and SDK development," 2019.

[23] L. A. B. Pamungkas and M. Imrona, "Analisa Perbandingan Kinerja Cross Platform Mobile Framework React Native dan Flutter," *eProceedings Eng.*, vol. 7, no. 1, 2020.

[24] G. I. Arb and K. Al-Majdi, "A freights status management system based on Dart and Flutter programming language," in *Journal of Physics: Conference Series*, 2020, vol. 1530, no. 1, p. 12020.

[25] D. Andersson and A. Axelsson, "Evaluation of The Software Development Process for A Multi-Platform Solution in Flutter." 2021.

[26] A. Pambudi, N. Lutfiani, M. Hardini, A. R. A. Zahra, and U. Rahardja, "The Digital Revolution of Startup Matchmaking: AI and Computer Science Synergies," in *2023 Eighth International Conference on Informatics and Computing (ICIC)*, 2023, pp. 1–6.

[27] N. N. Rafiana, "Technopreneurship Strategy to Grow Entrepreneurship Career Options for Students in Higher Education," *ADI J. Recent Innov.*, vol. 5, no. 2, pp. 110–126, 2024.