

## Chapter1. Introduction

### 1.1 Company Profile



### InfoWeb Solution

InfoWeb Solution is the leading IT Solutions company based in Surat, India. Since 2012, InfoWeb Solution is a IT Firm which combines beautiful interactive design with intelligent technology such as .NET and REACT, FLUTTER and other JavaScript Framework. Here at Speed Limit, InfoWeb Solution understand that having a great website, a print piece, or even a beautiful logo is just not enough. You need results. InfoWeb Solution is a result- focused company. InfoWeb Solution love's tying in creative marketing campaigns to our great work.

**Address:-** 25, Vatsalya Bunglows, Near S.D Jain Modern School, Vesu, Surat - 395007

**Contact No.:-** +91 99092 84838

**Email:-** sunnychevli@gmail.com

## 1.2 Customer Profile

### 1.1.1 Customer Detail

The Live Score project provides an engaging platform for viewers to follow Taekwondo tournaments in real-time through a Flutter-based mobile application. Viewers can access live match details and scores, ensuring they stay updated on the latest events and outcomes. The system leverages SignalR for real-time updates, offering an immersive experience with up-to-the-minute information on ongoing matches. This user-friendly Application interface allows viewers to effortlessly track their favorite athletes and matches without any management responsibilities.

Viewers benefit from a seamless and intuitive interface that prioritizes ease of access and real-time information. The platform's design ensures that all live scores, match progress, and outcomes are displayed promptly, keeping viewers engaged and informed. With the integration of Flutter Appliication, updates are instantaneous, providing a dynamic experience that mirrors the excitement of watching the matches live. Whether following individual athletes or entire tournaments, viewers can enjoy a comprehensive overview of all the action,making the Live Score project an indispensable tool for Taekwondo enthusiasts.

## 1.3 Current System

The Live Score project is tailored for various stakeholders involved in Taekwondo tournaments:

- Admins:

Responsible for managing categories, tournaments, coordinators, matches, and historical data. They oversee the entire system ensuring smooth operations.

- Coordinators:

Handle athlete registrations, match scheduling, referee assignments, coach management, and scoring details. They act as the backbone of the tournament organization.

- Referees:

Focus on live scoring during matches and ensure fair play. They have the ability to update their personal profiles.

- Viewers:

Public users who access the platform to view real-time match details and scores. They do not have any management responsibilities but benefit from live updates facilitated by Flutter Application with SignalR integration.

The system ensures secure authentication using JWT tokens and supports essential user functions such as login, logout, registration, and password recovery via email services.

## Chapter2. Proposed System

### 2.1 Scope

The Live Score project aims to streamline Taekwondo tournament management and viewing experiences. It encompasses functionalities for Admins, Coordinators, Referees, and Viewers, ensuring efficient operations and real-time updates. Utilizing .NET Core API, React with Redux and Flutter, the system offers secure authentication, seamless communication via SignalR, and robust database management. Admins oversee categories, tournaments, and Coordinator verification, while Coordinators manage athletes, matches, and scoring. Referees handle real-time scoring, and Viewers access live match details effortlessly. Overall, the project optimizes tournament management and enhances the viewing experience for all stakeholders.

### 2.2 Objective

The primary objective of the “Live Score” project is to provide a comprehensive and efficient solution for managing and viewing Taekwondo tournaments. This entails:

- Streamlining tournament logistics through functionalities tailored for Admins and Coordinators, including category and tournament management, athlete registration, match scheduling, and scoring oversight.
- Facilitating real-time communication via Flutter Application with SignalR to ensure viewers receive live match details and scores instantly, enhancing their engagement and experience.
- Offering a user-friendly interface for both administrators and viewers, allowing easy access to relevant information and functionalities without compromising on security.
- Providing viewers with a dynamic platform to access live match details seamlessly, promoting a richer viewing experience.

### 2.3 Constraints :

The Live Score project operates within several constraints that shape its development and implementation. Time constraints present a challenge, requiring efficient

management of development, testing, and deployment phases to meet project deadlines. Resource limitations, including human, financial, and technological resources, may restrict the project's ability to scale or implement advanced features.

Additionally, technical constraints such as dependencies on external APIs and regulatory compliance requirements impose further limitations on development efforts. Despite these challenges, proactive management and prioritization of stakeholder expectations can help mitigate constraints and ensure the successful delivery of a streamlined and engaging Taekwondo tournament management and viewing platform.

### 2.3.1 H/w Constraints:

#### I. Server Specification:

- Limited hardware resources may impact the performance and scalability of the application. The server specification, including CPU, RAM, and disk space, must be sufficient to handle concurrent user requests and real-time updates effectively.

#### II. Network Infrastructure:

- The network infrastructure, including bandwidth capacity and network latency, influences the responsiveness and reliability of real-time communication features such as SignalR. Constraints in network bandwidth or high latency may result in delays or interruptions in live score updates.

#### III. Storage System:

- Constraints in storage capacity and performance of the storage system, whether it's a local disk or a cloud-based solution, may affect the system's ability to store and retrieve data efficiently.

#### IV. Security Hardware:

- Constraints in security hardware, such as firewalls, intrusion detection systems, and encryption accelerators, may impact the system's ability to protect sensitive user data and prevent unauthorized access or attacks.

## V. Cost Constraints:

- Consider budgetary constraints when selecting hardware. Optimize the hardware configuration to meet project requirements within budgetary limits.

### 2.3.2 S/W Constraints:

#### I. Database Management System:

- The Live Score project's software constraints revolve around its Database Management System (DBMS). Compatibility with SQL Server Management Studio (SSMS), scalability for handling large datasets, and security measures are key considerations.

#### II. Operating System:

- Compatibility with various operating systems like Windows, Linux, and macOS is essential for broad accessibility.

#### III. Framework and Dependency Limitations:

- Compatibility issues between different versions of frameworks or dependencies may impact development and deployment flexibility.

#### IV. Licensing Restrictions:

- Compliance with software licensing agreements is necessary to avoid legal complications and ensure sustainability.

#### V. Performance Constraints:

- Performance limitations of software components could affect overall system performance and scalability.

#### VI. Security Vulnerabilities:

- Risks associated with software security vulnerabilities require proactive measures to ensure data integrity and confidentiality.

## 2.4 Advantages

- SQL Server Management Studio (SSMS) is a robust and scalable platform for managing relational databases, offering advanced features like stored procedures and triggers.
- SSMS Integration with other Microsoft technologies provides seamless compatibility and support.
- React and Redux Component-based architecture of React allows for modular and reusable code, enhancing development efficiency.
- Redux facilitates centralized state management, simplifying data flow and improving application scalability..
- .NET Core API is a Cross-platform compatibility of .NET Core allows for deployment on various operating systems, enhancing flexibility.
- Strong integration with other Microsoft technologies and libraries streamlines development and reduces compatibility issues.
- Flutter provides a unified framework for building robust applications across multiple platforms, ensuring consistent functionality and user experience on iOS, Android, web, and desktop environments, thereby simplifying cross-platform development efforts.

## 2.5 Limitation

- The database SSMS Licensing costs associated with SQL Server may be prohibitive for smaller projects or organizations.
- .NET Core AP Limited support for certain third-party libraries and frameworks compared to other languages like Node.js or Python.

## Chapter3. Environment Specification

### 3.1 Hardware & Software Requirements

#### HARDWARE SPECIFICATION

- ✓ Processor: 12<sup>th</sup> Gen Intel Core i3 1.20 GHz
- ✓ Ram : 8 GB

#### SOFTWARE SPECIFICATION

- ✓ Operating System: - Windows 10 ,11
- ✓ Front S/W: React, Redux, Flutter
- ✓ Back S/W: .Net Core API
- ✓ Database: SQL Server Management Studio (SSMS)

### 3.2 Development Description

The development of the Live Score project is a comprehensive process involving careful planning, design, implementation, testing, and deployment. By adhering to a structured approach and utilizing modern tools and technologies, the project aims to deliver a robust and efficient system for managing Taekwondo tournaments with real-time score updates and seamless user experience for all roles involved.

The development team will follow a systematic approach:

#### I. Requirement Analysis:

The initial phase involved gathering requirements from stakeholders, understanding the intricacies of Taekwondo scoring, and outlining the roles and functionalities needed for Admin, Coordinator, Referee, and Viewer.

#### II. System Design:

The architecture of the system was designed. This included defining the database schema, setting up the backend and frontend architectures, and designing the APIs. The system was structured to ensure scalability, reliability, and real-time capabilities. The use of SQL Server Management Studio (SSMS) for the database, .NET Core for the backend,

Flutter for the mobile application and React with Redux for the frontend were finalized.

### **III. Database Development:**

The database was developed using SQL Server Management Studio (SSMS). Tables were created to manage athletes, matches, referees, coaches, scores, and rounds. Stored procedures were implemented to handle complex business logic efficiently.

### **IV. Backend Development:**

The backend was developed using .NET Core API. This included creating APIs for all functionalities such as user management, match management, score management, and more. JWT tokens were implemented for secure API authentication. SignalR was integrated to handle real-time updates for live scores and round management. Temporary memory management was also implemented to optimize the handling of transient data during live operations.

### **V. Frontend Development:**

The frontend was developed using React and Redux, and Flutter was used for the mobile application. Components were created for various roles such as Admin, Coordinator, Referee, and Viewer. The frontend communicated with the backend APIs to fetch and display data dynamically. Real-time updates were managed using SignalR to ensure viewers received live scores without needing to refresh the page in application.

### **VI. Integration and Testing:**

Extensive testing was conducted to identify and fix bugs. Unit tests, integration tests, and end-to-end tests were performed to ensure the system met all requirements and functioned correctly under various scenarios.

### **VII. Deployment Configuration:**

The system was deployed on a server with a suitable operating system (Windows Server or a compatible Linux distribution). IIS or Nginx/Apache was configured as the web server.

## Chapter4. System Planning

### 4.1 Feasibility Study

The Live Score project is feasible across all key components, promising a valuable tool for Taekwondo event management and live scoring.

#### Key components:

##### I. Project Scope and Objectives:

- The system supports four user roles—Admin, Coordinator, Referee, and Viewer—each with distinct functionalities. Key objectives include real-time score updates, efficient match management, secure user authentication, and seamless user experience across roles.

##### II. Market Analysis:

- The demand for real-time sports scoring and management systems is growing, driven by increasing digital engagement in sports. Target users include sports organizations, Taekwondo clubs, event coordinators, referees, athletes, and spectators.

##### III. Technical Feasibility:

- The project leverages robust and scalable technologies such as .NET Core for backend, React and Redux for the frontend, Flutter for the mobile application and SSMS for database management. Real-time communication is facilitated by SignalR and use of JWT Tokens and email services ensures secure and reliable user authentication and account management.

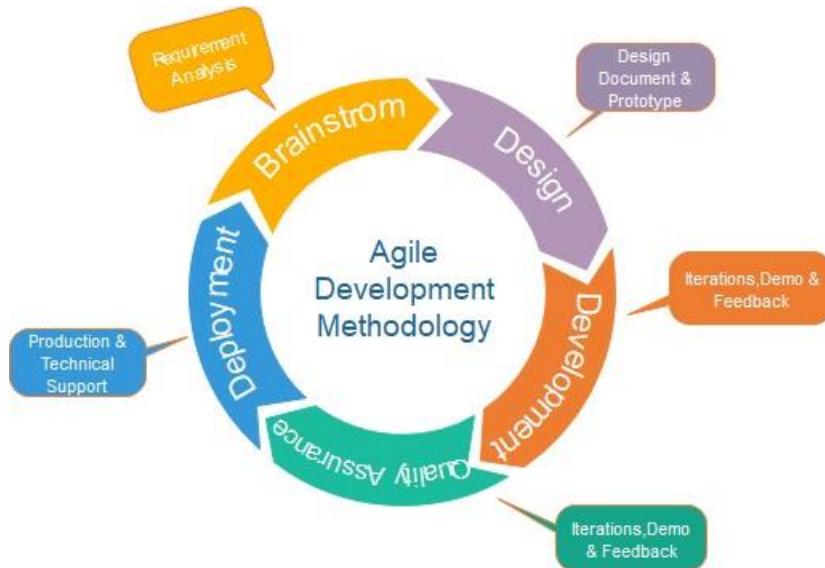
##### IV. Legal and Ethical Feasibility:

- The project complies with data protection regulations such as GDPR and CCPA, ensuring user data privacy and security. Ethical considerations include fair play, transparency in scoring, and non-discriminatory practices.

##### V. Operational Feasibility:

- The operational workflow integrates seamlessly with existing sports event management practices. Admins and coordinators can easily manage categories, tournaments, and personnel, while referees can efficiently handle scoring during matches. Viewers benefit from intuitive access to live.

## 4.2 Software Engineering Model



*Fig. Agile Model*

- ✓ The meaning of Agile is swift or versatile. "**Agile process model**" refers to a software development approach based on iterative development.
- ✓ Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning.
- ✓ The project scope and requirements are laid down at the beginning of the development process.
- ✓ Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.
- ✓ Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks.
- ✓ The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.

## 4.3 Risk Analysis

While the Live Score project presents several risks, these can be effectively managed through careful planning, robust security measures, thorough testing, and proactive mitigation strategies.

### I. Technical Risks:

- System Downtime, Scalability Issues and Integration Challenges various technologies (e.g., .NET Core, SignalR, Flutter, React, SQL Server) may result in compatibility issues
- Mitigation: Implement redundant servers and robust backup systems to ensure high availability and reliability and use well-documented integration practices to ensure seamless interaction between components.

### II. Project Management Risks:

- Timeline Delays such as Unforeseen issues may delay the project timeline, impacting delivery schedules, and Team Coordination.
- Mitigation: Use agile project management practices, regular progress reviews, and contingency planning to stay on track

### III. Operational Risks:

- User Adoption and Data Accuracy like Inaccurate data entry by referees or coordinators could lead to incorrect scores and match outcomes.
- Mitigation: Implement validation checks and provide user training to ensure accurate data entry.

### IV. Security Risk:

- Data Breaches Unauthorized access to sensitive data and Authentication Failures weak authentication mechanisms could lead to unauthorized access.
- Mitigation: Implement strong encryption, secure authentication (JWT Tokens)

### V. Legal and Compliance:

- Regulatory Compliance like Non-compliance with data protection laws old result in legal penalties and Intellectual Property
- Mitigation: Regularly review and update compliance practices to ensure adherence to relevant regulations.

## 4.4 Project Schedule

### 4.4.1 Task Dependency:

### 4.4.2 Timeline Chart

Task	Jan-Feb	Mar-April	May	June	July	August	September	October	November	December
<b>1) Initial Planning</b>	5 22	7 23	3 24	9 25	10 26	11 27	12 28	13 29	14 30	15 31
Identify Project and their requirements										
Develop project timeline	40 39	41 42	43 44	45 46	47 48	49 50	51 52	53 54	55 56	56 57
Assign roles and responsibilities to team members	58 56	59 59	60 60	61 61	62 63	64 65	66 67	68 69	70 71	71 72
<b>Milestone: Define the project scope &amp; objectives</b>		76 74	77 75	78 75	79 80	81 81	82 83	83 84	85 85	86 87
<b>2) Planning</b>	93 92	94 92	95 93	96 97	97 98	99 99	100 101	101 102	102 103	103 104
Identify project risks & develop mitigation strategies	109 125	110 126	111 127	112 128	113 129	114 130	115 131	116 132	117 133	118 134
Define project deliverables and timelines										
Develop a detailed project schedule	141 141	142 143	143 144	144 145	145 146	146 147	147 148	148 149	150 150	151 152
<b>Milestone: Develop a high-level project plan</b>	157 157	158 158	159 159	160 160	161 161	162 162	163 163	164 164	165 165	166 167

<b>3) Requirements Analysis &amp; Design</b>	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188
<b>Designing</b>																
Conduct user research and identify requirements	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204
Develop user stories and acceptance criteria	205	206	207	208	209	210	211	212	213	214	215	216	217	218	219	220
Develop system architecture and design	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236
<b>Milestone: Designing completed</b>	237	238	239	240	241	242	243	244	245	246	247	248	249	250	251	252
<b>4) Implementation</b>	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268
Develop code for core functionality	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284
Conduct code reviews and testing	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300
Document system design and implementation	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316
<b>Milestone: Coding completed</b>	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332
<b>5) Development</b>	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348
Develop code for additional features	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364
Conduct code reviews and testing	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380
Document system design and implementation	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396
<b>Milestone: Development completed</b>	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412
<b>6) Testing</b>	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428
<b>Develop test plans and cases</b>	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444
<b>Conduct unit, integration, and system testing</b>	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460
<b>Document testing results and defect</b>	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476
<b>Milestone: Testing completed</b>	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492

## Chapter5. System Analysis

### 5.1 Detailed SRS

Software Requirements Specification (SRS) for your LiveScore project involves documenting various aspects such as functional requirements, non-functional requirements, and constraints. Below is a structured table-based format for the SRS, incorporating key details from the tables and project description provided:

#### I. Functional Requirements:

- Admin Management:

<b>Requirement ID</b>	<b>Requirement Description</b>
<b>FR-ADM-01</b>	Admins can manage categories including adding, editing, and deleting categories.
<b>FR-ADM-02</b>	Admins can manage tournaments including scheduling, updating details, and assigning coordinators.
<b>FR-ADM-03</b>	Admins can verify coordinators by reviewing their credentials and approving/rejecting their requests.
<b>FR-ADM-04</b>	Admins can view match details including participants, scores, and match history.

- Coordinator Management:

<b>Requirement ID</b>	<b>Requirement Description</b>
<b>FR-COORD-01</b>	Coordinators can manage athletes including registration, updating details, and assigning to categories.
<b>FR-COORD-02</b>	Coordinators can manage matches including scheduling, updating match details, and assigning referees.

<b>FR-COORD-03</b>	Coordinators can manage referees including assigning matches and reviewing performance.
<b>FR-COORD-04</b>	Coordinators can manage coaches including adding, updating, and assigning to athletes.
<b>FR-COORD-05</b>	Coordinators can manage scoring during matches including entering points and penalties.
<b>FR-COORD-06</b>	Coordinators can manage rounds including scheduling, timing, and recording scores.

➤ Referee Management

<b>Requirement ID</b>	<b>Requirement Description</b>
<b>FR-REF-01</b>	Referees can manage scoring during matches including entering points and penalties.
<b>FR-REF-02</b>	Referees can manage rounds including recording round results and winner.

➤ Viewer Management

<b>Requirement ID</b>	<b>Requirement Description</b>
<b>FR- VIEW-01</b>	Viewers can access live match scores and updates in real-time using SignalR..
<b>FR- VIEW-02</b>	Viewers can view past match results and historical data.

## II. Non-Functional Requirements

Requirement	Requirement Description
ID	
NFR -01	The system shall ensure high availability and performance during peak usage times.
NFR -02	The system shall maintain data consistency and integrity across all transactions.
NFR -03	The system shall be secure with data encryption and user authentication using JWT tokens.
NFR -04	The system shall provide a responsive and user-friendly interface using React and Redux.
NFR -05	The system shall comply with GDPR regulations for user data privacy and protection.

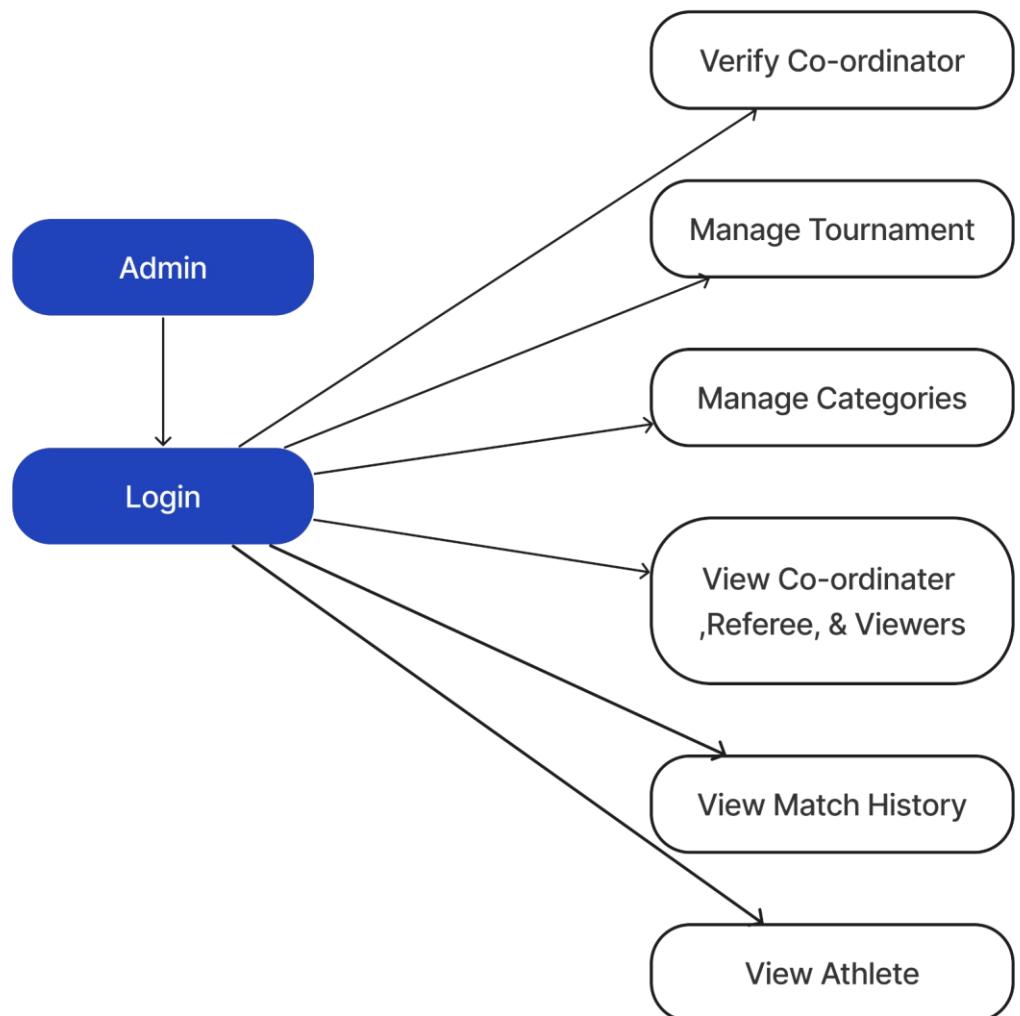
## III. Constraints

Requirement	Requirement Description
ID	
CON-01	The system shall be developed using .NET Core for backend APIs.
CON-02	The system shall use SQL Server Management Studio (SSMS) for database management.
CON-03	The system shall utilize SignalR for real-time score updates and round management.
CON-04	The system shall implement stored procedures for business logic and data manipulation.

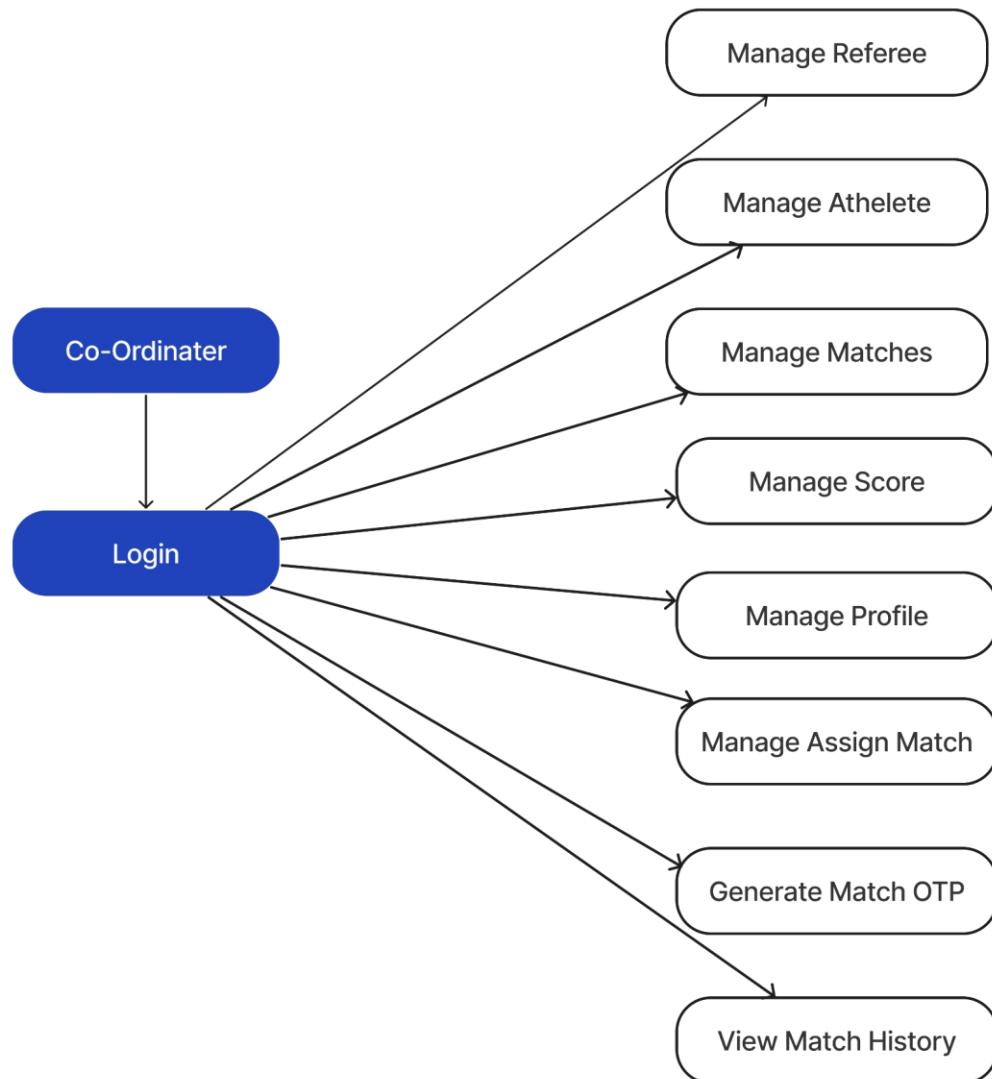
## 5.2 UML Diagram

### 5.2.1 Use Case Diagram

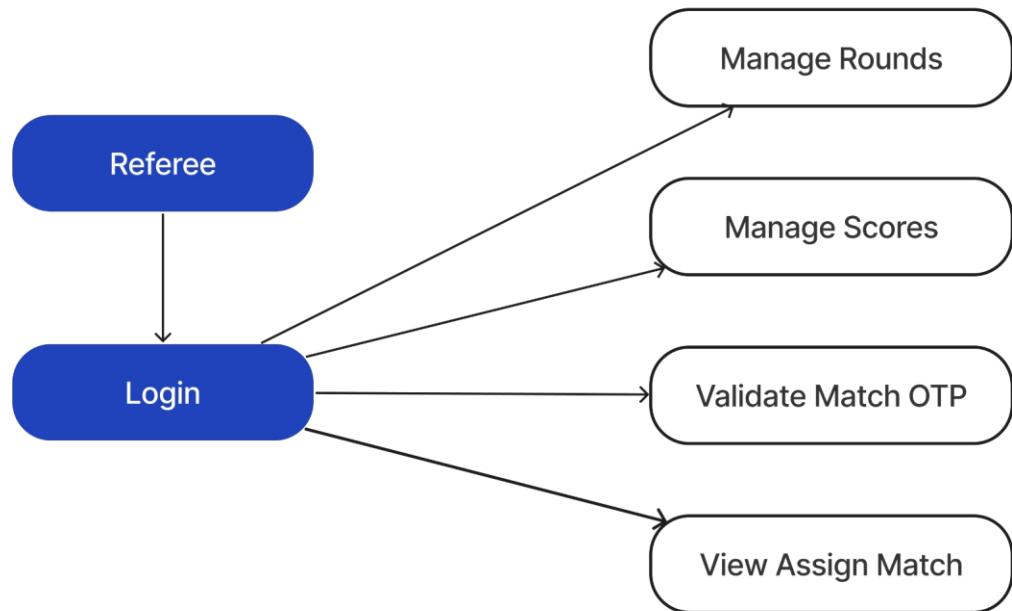
→ Admin UML:



→ Coordinator UML:

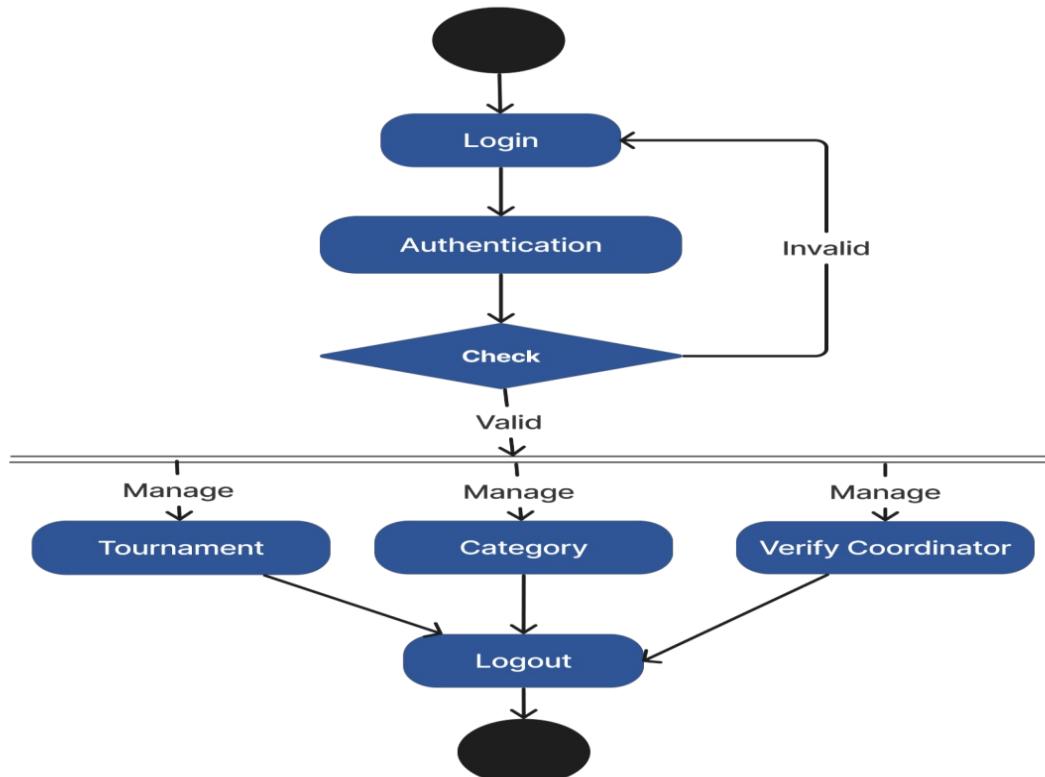


➔ Referee UML:

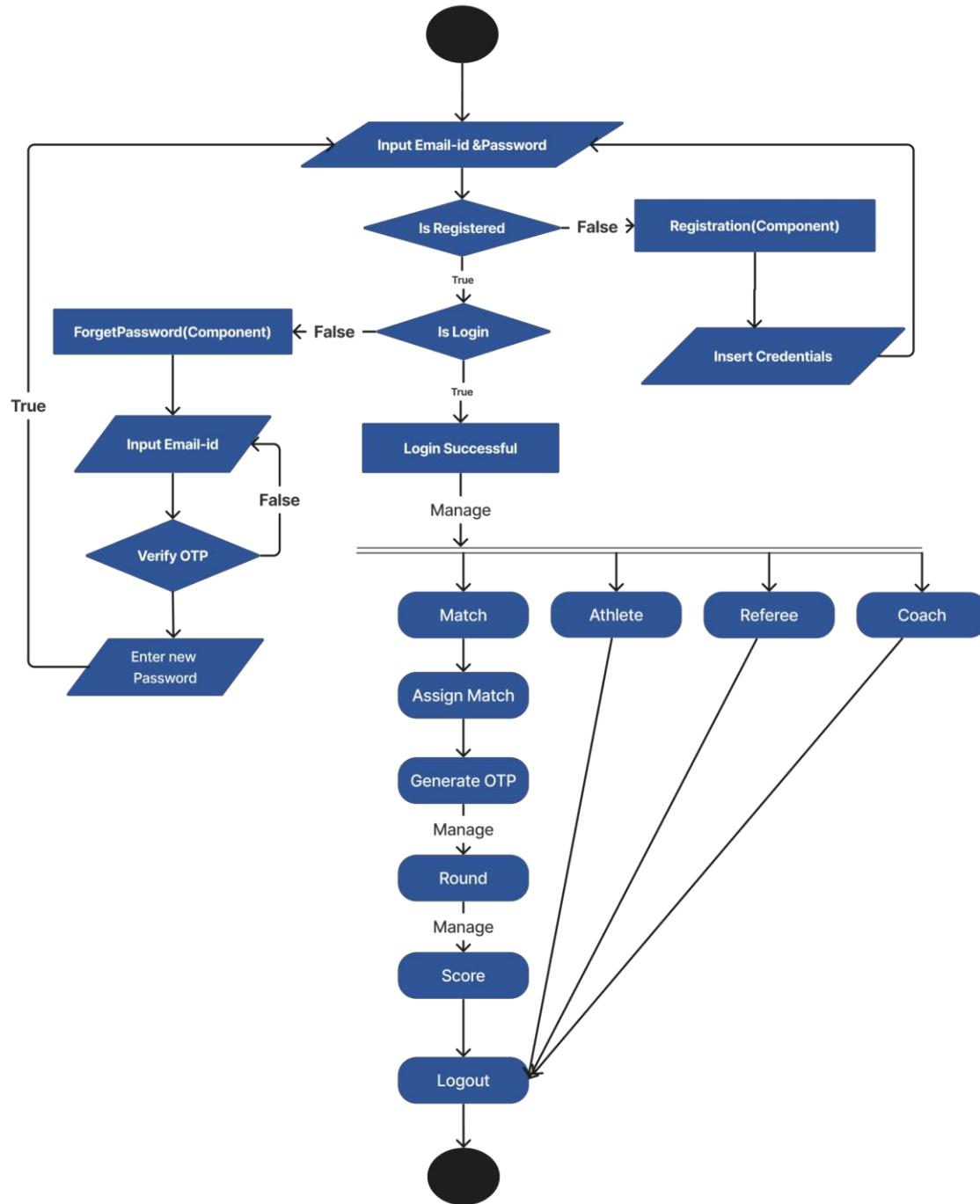


### 5.2.2 Activity Diagram

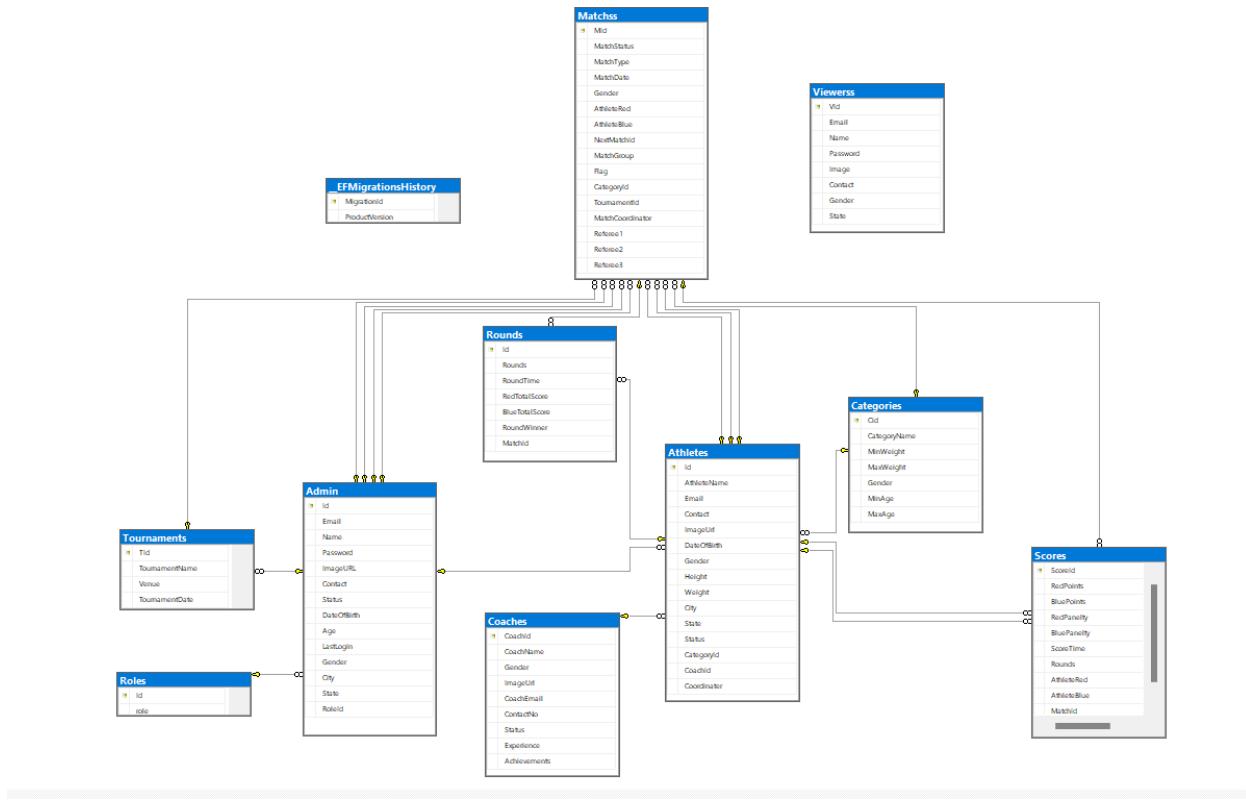
➔ Admin Activity:



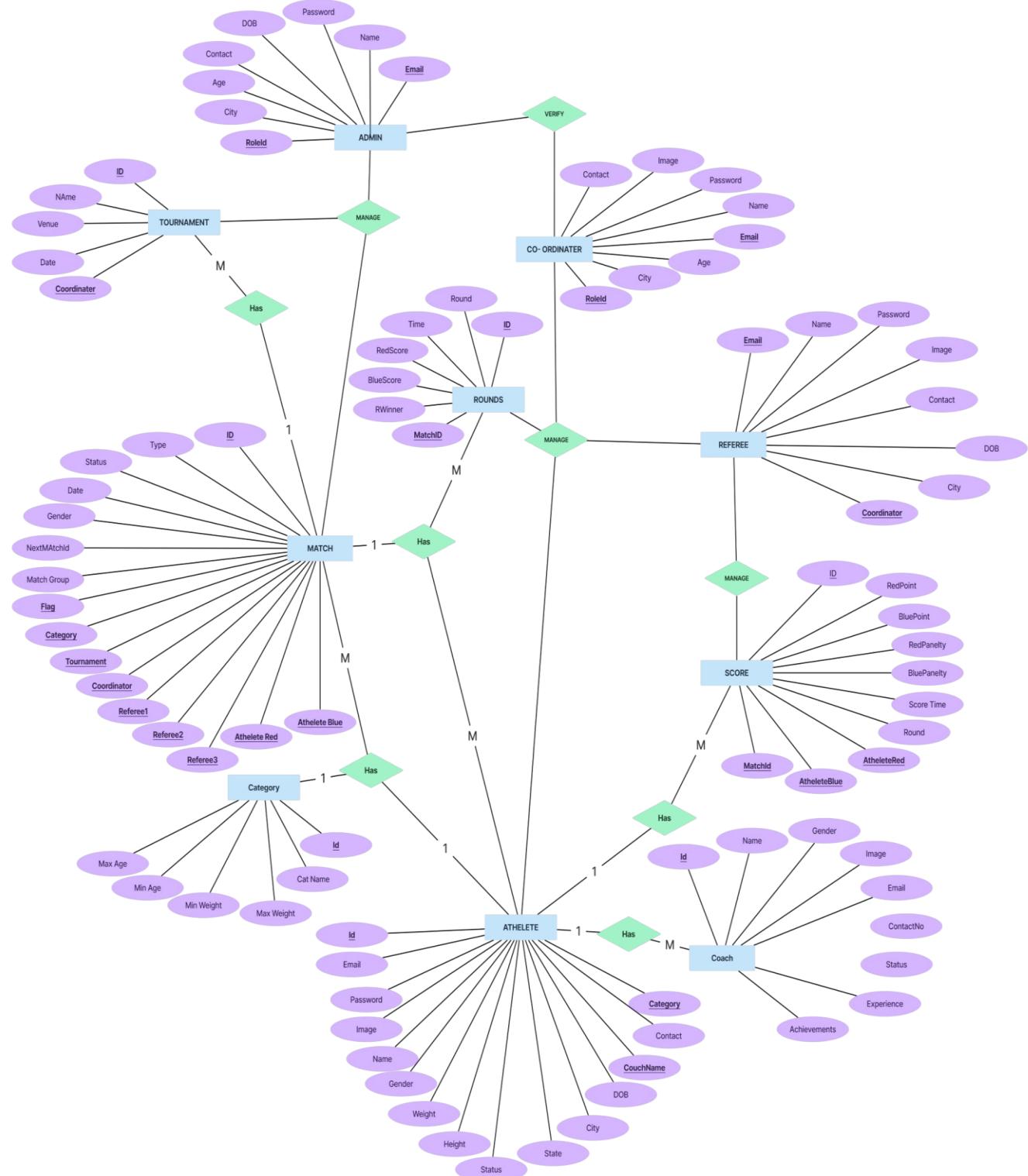
→ Coordinator Activity:



## 5.3 Class Diagram



## 5.4 E-R Diagram



## Chapter6. Software Design

### 6.1 Database Design

➤ ACR Table : (Admin, Coordinator and Referee)

Field	Type	Description	Constraint
Id	int	Primary key	Primary key, auto-generated
Email	nvarchar(101)	Email address	Not null
Name	nvarchar(101)	Name	Not null
Password	nvarchar(101)	Password	Not null
ImageURL	nvarchar(max)	URL to profile image	Not null
Contact	nvarchar(101)	contact number	Not null
Status	nvarchar(max)	status (e.g., Pending, Live etc)	Not null
DateOfBirth	datetime2(7)	Date of birth	Not null
Age	int	Age	Not null
LastLogin	datetime	Date and time of last login	Nullable
Gender	nvarchar(101)	Gender	Nullable
City	nvarchar(101)	City	Nullable
State	nvarchar(101)	State	Nullable
RoleId	int	Foreign key to Roles table	Foreign key to Roles(Id)

➤ Role Table:

<b>Field</b>	<b>Type</b>	<b>Description</b>	<b>Constraint</b>
Id	int	Primary key	Primary key, auto-generated
role	nvarchar(101)	Role name	Nullable

➤ Category Table:

<b>Field</b>	<b>Type</b>	<b>Description</b>	<b>Constraint</b>
Cid	int	Primary key	Primary key, auto-generated
CategoryName	nvarchar(101)	Name of the category	Not null
MinWeight	int	Minimum weight for the category	Not null
MaxWeight	int	Maximum weight for the category	Not null
Gender	nvarchar(10)	Gender associated with the category	Nullable
MinAge	int	Minimum age for the category	Not null
MaxAge	int	Maximum age for the category	Not null

➤ Coach Table:

<b>Field</b>	<b>Type</b>	<b>Description</b>	<b>Constraint</b>
CoachId	int	Primary key	Primary key, auto-generated
CoachName	nvarchar(101)	Name of the coach	Not null
Gender	nvarchar(101)	Gender of the coach	Not null
ImageUrl	nvarchar(max)	URL to coach's profile image	Not null
CoachEmail	nvarchar(101)	Email address of the coach	Not null
ContactNo	nvarchar(101)	Contact number of the coach	Not null
Status	nvarchar(101)	Status of the coach (e.g., active)	Nullable
Experience	nvarchar(101)	Coach's experience details	Not null
Achievements	nvarchar(101)	Coach's achievements	Not null

➤ Athlete Table:

<b>Field</b>	<b>Type</b>	<b>Description</b>	<b>Constraint</b>
Id	int	Primary key	Primary key, auto-generated
AthleteName	nvarchar(101)	Name of the athlete	Not null
Email	nvarchar(101)	Email address of the athlete	Not null
Contact	nvarchar(101)	Contact number of the athlete	Not null

ImageUrl	nvarchar(max)	URL to athlete's profile image	Not null
DateOfBirth	datetime2(7)	Date of birth of the athlete	Not null
Gender	nvarchar(101)	Gender of the athlete	Not null
Height	float	Height of the athlete (in meters)	Not null
Weight	float	Weight of the athlete (in kilograms)	Nullable
City	nvarchar(101)	City where the athlete resides	Not null
State	nvarchar(101)	State where the athlete resides	Not null
Status	nvarchar(101)	Status of the athlete (e.g., active)	Nullable
CategoryId	int	Foreign key to Categories table	Foreign key to Categories(Cid)
CoachId	int	Foreign key to Coaches table	Foreign key to Coaches(CoachId)
Coordinator	int	Foreign key to Admin table (Coordinator)	Foreign key to Admin(Id)

➤ Tournament Table:

Field	Type	Description	Constraint
TId	int	Primary key	Primary key, auto-generated
TournamentName	nvarchar(101)	Name of the tournament	Not null
Venue	nvarchar(101)	Venue where the tournament is held	Not null
TournamentDate	datetime2(7)	Date and time of the tournament	Not null
TournamentCoordinator	int	Foreign key to Admin table (Coordinator)	Nullable

➤ Viewers Table:

Field	Type	Description	Constraint
VId	int	Primary key	Primary key, auto-generated
Email	nvarchar(101)	Email of the viewer	Not null
Name	nvarchar(101)	Name of the viewer	Not null
Password	nvarchar(101)	Password of the viewer	Not null
Image	nvarchar(max)	URL or path to viewer's image	Not null
Contact	nvarchar(10)	Contact number of the viewer	Not null
Gender	nvarchar(101)	Gender of the viewer	Nullable
State	nvarchar(101)	State of the viewer	Nullable

➤ Match Table:

Field	Type	Description	Constraint
MId	int	Primary key	Primary key, auto-generated
MatchStatus	nvarchar(101)	Status of the match (e.g., ongoing, completed)	Nullable
MatchType	nvarchar(101)	Type of the match (e.g., sparring, forms)	Nullable
MatchDate	datetime2(7)	Date and time of the match	Not null
Gender	nvarchar(max)	Gender category of the match	Nullable

AthleteRed	int	Foreign key to Athletes table for Red athlete	Nullable
AthleteBlue	int	Foreign key to Athletes table for Blue athlete	Nullable
NextMatchId	int	ID of the next match in sequence	Nullable
MatchGroup	int	Group or category of the match	Not null
Flag	int	Foreign key to Athletes table for flag	Nullable
CategoryId	int	Foreign key to Categories table	Nullable
TournamentId	int	Foreign key to Tournaments table	Not null
MatchCoordinator	int	Foreign key to Admin table (Match coordinator)	Nullable
Referee1	int	Foreign key to Admin table (Referee 1)	Nullable
Referee2	int	Foreign key to Admin table (Referee 2)	Nullable
Referee3	int	Foreign key to Admin table (Referee 3)	Nullable

➤ Scores Table:

Field	Type	Description	Constraint
ScoreId	int	Primary key	Primary key, auto-generated
RedPoints	int	Points scored by the Red athlete	Not null
BluePoints	int	Points scored by the Blue athlete	Not null
RedPanelty	int	Penalty points for the Red athlete	Nullable

BluePanelty	int	Penalty points for the Blue athlete	Nullable
ScoreTime	datetime2(7)	Date and time of the score entry	Not null, default value set to current timestamp
Rounds	int	Round number	Nullable
AthleteRed	int	Foreign key to Athletes table (Red athlete)	Nullable
AthleteBlue	int	Foreign key to Athletes table (Blue athlete)	Nullable
MatchId	int	Foreign key to Matchss table	Nullable

➤ Rounds Table:

Field	Type	Description	Constraint
Id	int	Primary key	Primary key, auto-generated
Rounds	int	Round number	Not null
RoundTime	datetime2(7)	Date and time of the round	Not null, default value set to current timestamp
RedTotalScore	int	Total score for the Red athlete	Nullable
BlueTotalScore	int	Total score for the Blue athlete	Nullable
RoundWinner	int	Foreign key to Athletes table (winner)	Nullable
MatchId	int	Foreign key to Matchss table	Nullable

## 6.2 Interface Design sitemap followed with page snapshots

### Login Page:



### Sign in

Email  Enter Your Email

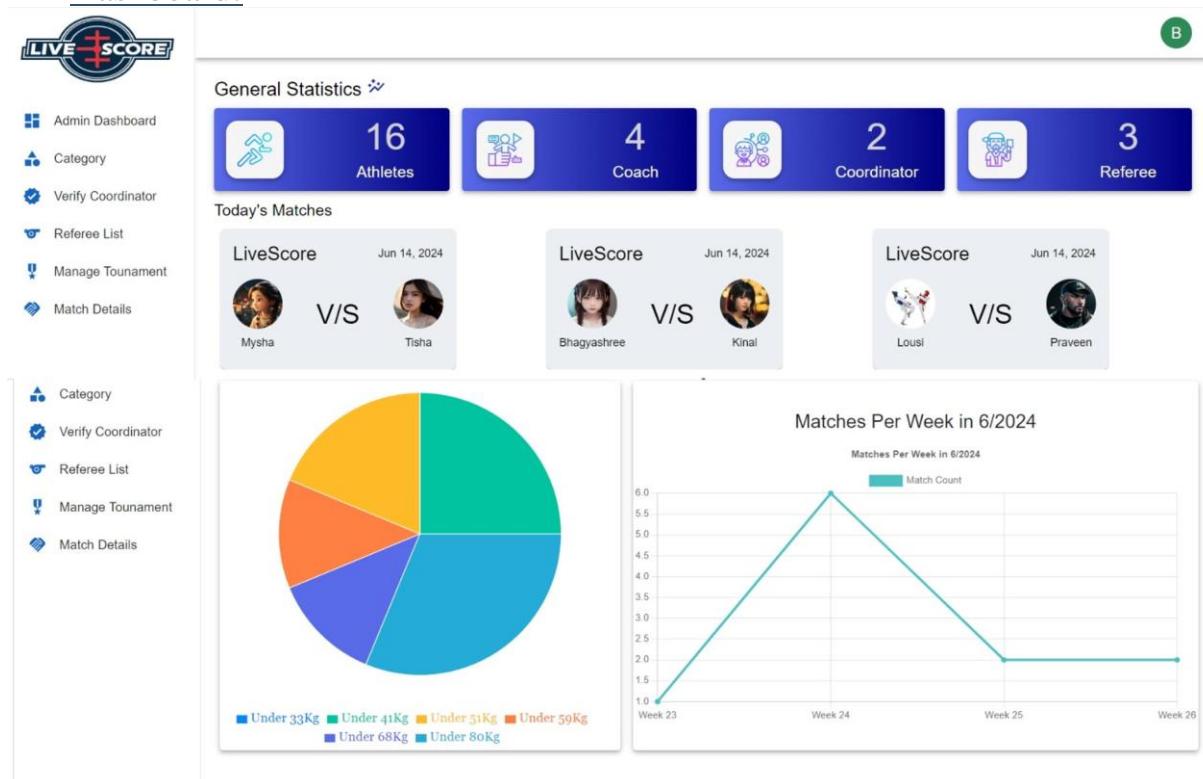
Password  Enter Your Password

**SUBMIT**

[Forgot password?](#) [Register Coordinator](#)

### Admin Side:

#### Dashboard:



Category:

## ➤ List:

Name	Minimum Weight	Maximum Weight	Minimum Age	Maximum Age	Actions
Under 33Kg	29 Kg	33 Kg	12	14	
Under 41Kg	34 Kg	49 Kg	14	17	
Under 51Kg	42 Kg	51 Kg	14	28	
Under 59Kg	52 Kg	59 Kg	17	34	
Under 68Kg	60 Kg	68 Kg	17	35	
Under 80Kg	69 Kg	80 Kg	17	35	

## ➤ Insert:

## ➤ Edit:

Coordinator Verification's :

➤ List:

LIVE SCORE Admin Dashboard

**Coordinator Verifications**

Avatar	Name	Email	Contact	DateOfBirth	Gender	Age	City	state	Status	Actions
	Coordinator	Coordinator@gmail.com	9643256781	19/05/2003	female	21	Jaisalmeir	Rajasthan	<span>Verified</span>	<span>BLOCK</span>
	Coneeva	Coneeva@gmail.com	9876543456	09/12/2000	female	23	Surat	Gujarat	<span>Not Verified</span>	<span>ACCEPT</span>

Rows per page: 100 ▾ 1–2 of 2 < >

Tournament :

➤ List:

LIVE SCORE Admin Dashboard

**Tournament Management**

Tournament Name	Venue	Date	Coordinator	Actions
Khel Maha Khumb	Surat	09/06/2024	Coordinator	<span>EDIT</span>
GamersGame	Surat	14/06/2024	Coordinator	<span>EDIT</span>

ADD TOURNAMENT

Rows per page: 100 ▾ 1–2 of 2 < >

➤ Insert:

LIVE SCORE Admin Dashboard

**Tournament Management**

Add Tournament

Tournament Name	<input type="text" value="Khel Maha Khumb"/>
Venue	<input type="text" value="Surat"/>
Tournament Date	<input type="text" value="mm/dd/yyyy"/>
Coordinator	<input type="text"/>

SUBMIT

ADD TOURNAMENT

Rows per page: 100 ▾ 1–2 of 2 < >

## ➤ Edit:

**Edit Tournament**

Tournament Name: GamersGame

Venue: Surat

Tournament Date: 06/14/2024

Coordinator: Coordinator

**SUBMIT**

Match :

## ➤ List:

**Match Management**

GId	Tournament	Match Type	Match Status	Match Date	Category	Gender	Athlete Red	Athlete Blue	Coordinator	Referee 1	Referee 2	Bh
6	Khel Maha Khumb	Quarter-Final	Live	14/06/2024	Under 80Kg	Male	Lousi	Praveen	Coordinator	Referee1	Referee2	Bh
7	Khel Maha Khumb	Quarter-Final	Upcoming	22/06/2024	Under 59Kg	Female	Alia	Jiya	-----	-----	-----	
8	Khel Maha Khumb	Semi-Final	Live	25/06/2024	Under 51Kg	Female	Bhagyashree	-----	-----	-----	-----	
9	Khel Maha Khumb	Semi-Final	Live	25/06/2024	Under 80Kg	Male	Meet	Sujeet	-----	-----	-----	
10	Khel Maha Khumb	Semi-Final	Upcoming	15/06/2024	Under 41Kg	Female	Tisha	Mysha	-----	-----	-----	
11	Khel Maha Khumb			13/06/2024	Under 80Kg	Male	Bhuwan	Bhuwan	-----	-----	-----	
12	GamersGame	Semi-Final	Upcoming	14/06/2024	Under 41Kg	Female	Mysha	Tisha	-----	-----	-----	
13	GamersGame	Final	Upcoming	14/06/2024	Under 51Kg	Female	Bhagyashree	Kinal	-----	-----	-----	

Rows per page: 100 ▾ 1–8 of 8 < >

## ➤ Match History:



- [Admin Dashboard](#)
- [Category](#)
- [Verify Coordinator](#)
- [Referee List](#)
- [Manage Tournament](#)
- [Match Details](#)

Match Management

History

	COLUMNS	FILTERS	EXPORT	DENSITY									
Rounds	Gid	NextId	Winner	RedImage	Athlete Red	BlueImage	Athlete Blue	Tournament	Match Type	Match Date	Category	Gender	Co
1	8	Bhagyashree			Kinal			Khel Maha Khumb	Quarter-Final	08/06/2024	Under 51Kg	Female	C
2	9	Sujeet			Meet			Khel Maha Khumb	Quarter-Final	20/06/2024	Under 80Kg	Male	C
3	11	Bhuwan			Viduyt			Khel Maha Khumb	Quarter-Final	13/06/2024	Under 80Kg	Male	C

➤ Round History:

LIVE SCORE

- Admin Dashboard
- Category
- Verify Coordinator
- Referee List
- Manage Tournament
- Match Details

B

Rounds

- Round 1
- Round 2
- Round 3

➤ Score History:

LIVE SCORE

- Admin Dashboard
- Category
- Verify Coordinator
- Referee List
- Manage Tournament
- Match Details

B

Rounds

ROUND WINNER : BHAGYASHREE

RED TOTAL SCORE : 8

BLUE TOTAL SCORE : 7

RedPoints	BluePoints	RedPenalty	BluePenalty	ScoreTime
1	0	0	0	08/06/2024 15:14:52
0	2	0	0	08/06/2024 15:15:06
0	0	0	1	08/06/2024 15:15:15
3	0	0	0	08/06/2024 15:15:33
0	0	1	0	08/06/2024 15:15:44
0	0	1	0	08/06/2024 15:15:48
2	0	0	0	08/06/2024 15:16:00
0	2	0	0	08/06/2024 15:16:11

## Coordinator Side:

### Dashboard:



- Coordinator Dashboard
- Athlete
- Referee
- Coach
- Matches

Match Assign

LiveScore Jun 14, 2024



Mysha

VIEW

V/S



Tisha

OTP

LiveScore Jun 14, 2024



Lousi

VIEW

V/S



Praveen

OTP

Today's Matches

LiveScore Jun 14, 2024



Mysha

V/S



Tisha

LiveScore Jun 14, 2024



Bhagyashree

V/S



Kinal

LiveScore ● Live

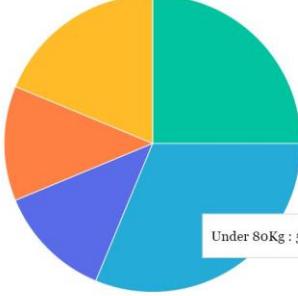


Lousi

V/S



Praveen



Under 33Kg : 5
Under 41Kg : 5
Under 51Kg : 5
Under 59Kg : 5
Under 68Kg : 5
Under 80Kg : 5

Matches Per Week in 6/2024



The graph shows the number of matches per week in June 2024. The count peaks at Week 24 (approximately 6 matches) and then drops to a steady state of about 2 matches per week for Weeks 25 and 26.

Week	Match Count
Week 23	1.0
Week 24	6.0
Week 25	2.0
Week 26	2.0

### Registration:

#### Register

Name ●

Email ●  ✉

Contact ●

Date of birth ●

Upload Image ●  No file chosen

Gender ●  Male  Female  Other

state ●  city ●

REGISTER

Page 37

Profile:
Athlete:

## ➤ List:

## ➤ Insert:

## ➤ Update:

LIVE SCORE

Coordinator Dashboard

Athlete

Referee

Coach

Matches

Edit Athlete

Name: Tisha

Email: @ tisha@gmail.com

Contact: 8765457866

Date of Birth: 12/15/2009

Gender: Female

Height (cm): 140

Weight (kg): 40

State: Gujarat

City: Surat

Coach: Prerna

Category: Under 41Kg

SUBMIT

## ➤ Block:

LIVE SCORE

Coordinator Dashboard

Athlete

Referee

Coach

Matches

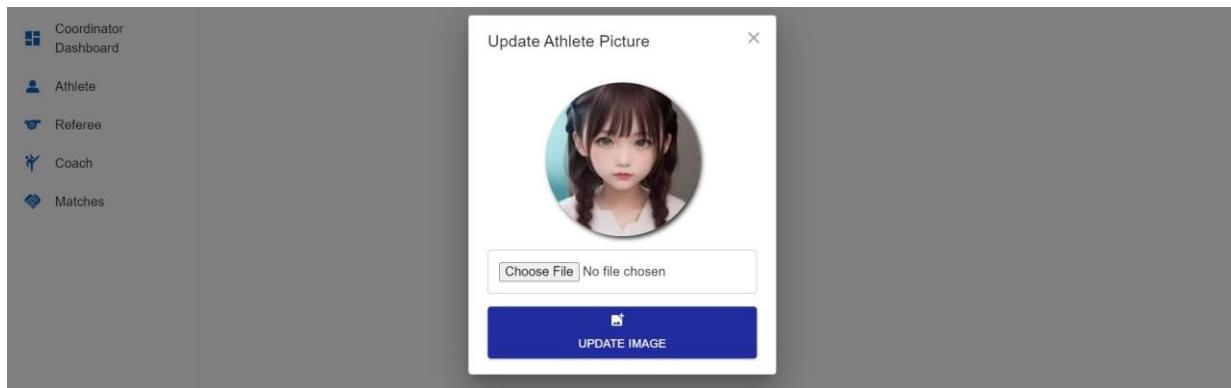
Athlete Management

COLUMNS FILTERS EXPORT DENSITY ADD ATHLETE

Avatar	Name	Email	Contact	Age	Gender	City	State	Category	Coach	Coordinator	Action
	Bhagyashree	bhagyashree@gmail.com	9876543878	21	Female	Surat	Gujarat	Under 51Kg	Jainul	Coordinator	UNBLOCK
	Kinal	kinal@gmail.com	9876543579	21	Female	Valsad	Gujarat	Under 51Kg	Jainul	Coordinator	EDIT  BLOCK
	Meet	meet@gmail.com	9864392322	20	Male	Bilimora	Gujarat	Under 80Kg	Meet	Coordinator	EDIT  BLOCK
	Sujeet	sujeet@gmail.com	7897897896	21	Male	JaunPur	Up	Under 80Kg	Meet	Coordinator	EDIT  BLOCK

Athlete Is Successfully Blocked

## ➤ Update Image:



Referee:

➤ List:

Avatar	Name	Email	Contact	DateOfBirth	Gender	Age	LastLogin	City	State	Status
	Referee1	referee@gmail.com	9876543215	1999-10-17T...	female	24	14/06/2024 10:18	Surat	Gujarat	<span>Verified</span>
	Referee2	referee1@gmail.com	8907897892	1997-05-22...	male	27	-----	Jaisal...	Rajasthan	<span>Verified</span>
	Bhagvash...	2023023061@vns...	string	2024-06-14...	string	0	14/06/2024 11:42	string	string	<span>Verified</span>

Rows per page: 100 ▾ 1–3 of 3 < >

➤ Insert:

**Add Referee**

Name

Email

Password

Contact

Date of birth

Upload Image

Gender   
  Male  Female  Other

state

city

ADD REFEREE

Coach:

➤ List:

Avatar	Name	Email	Contact	Gender	Experience	Achievements	Actions
	Jainul	jainul@gmail.com	9876543456	male	5 years	2 silver	
	Meena	meena@gmail.com	9876543456	female	25 years	4 Diamond	
	Meet	meet@gmail.com	7895267888	male	5 years	2 silver	
	Prerna	prerna@gmail.com	7643235733	female	15 years	3 platinum	

## ➤ Insert:

**Coach Management**

**Add Coach**

Avatar	Name
	Jainul
	Meena
	Meet
	Prerna

Name:

Email:

Upload Image:  No file chosen

Contact:

Gender:  Male  Female  Other

Achievements: 2 silver

Experiences: Diamond

**SUBMIT**

## ➤ Edit:

**Edit Coach**

Name
Meena

Name:

Email:

Contact:

Gender:  Male  Female  Other

Achievements: 4 Diamond

Experiences: 25 years

**SUBMIT**

## ➤ Block:

**Coach Management**

**ADD COACH**

Avatar	Name	Email	Contact	Gender	Experience	Achievements	Actions
	Jainul	jainul@gmail.com	9876543456	male	5 years	2 silver	UNBLOCK
	Meena	meena@gmail.com	9876543456	female	25 years	4 Diamond	BLOCK
	Meet	meet@gmail.com	7895267888	male	5 years	2 silver	BLOCK
	Prerna	prerna@gmail.com	7643235733	female	15 years	3 platinum	BLOCK

**coach Is Successfully Blocked**

Match:

## List:



- Coordinator Dashboard
- Athlete
- Referee
- Coach
- Matches

Match Management

Match
ADD MATCH

GId	Action	Tournament	Match Type	Match Status	Match Date	Category	Gender	Athlete Red	Athlete Blue	Coordinator	F
6	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	Khel Maha Khumb	Quarter-Final	Live	14/06/2024	Under 80Kg	Male	Lousi	Praveen	Coordinator	<span style="color: red;">B</span>
7	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	Khel Maha Khumb	Quarter-Final	Upcoming	22/06/2024	Under 59Kg	Female	Alia	Jiya	-----	
8	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	Khel Maha Khumb	Semi-Final	Live	25/06/2024	Under 51Kg	Female	Bhagvashree	-----	-----	
9	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	Khel Maha Khumb	Semi-Final	Live	25/06/2024	Under 80Kg	Male	Meet	Sujeet	-----	
10	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	Khel Maha Khumb	Semi-Final	Upcoming	15/06/2024	Under 41Kg	Female	Tisha	Mysha	-----	
11	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	Khel Maha Khumb			13/06/2024	Under 80Kg	Male	Bhuwan	Bhuwan	-----	
12	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	GamersGame	Semi-Final	Upcoming	14/06/2024	Under 41Kg	Female	Mysha	Tisha	Coordinator	<span style="color: red;">B</span>
13	<span style="color: orange;">EDIT</span> <span style="color: green;">ASSIGN</span>	GamersGame	Final	Upcoming	14/06/2024	Under 51Kg	Female	Bhagvashree	Kinal	-----	

## ➤ Insert:

Match Management

Match
ADD MATCH

GId	Athlete Red	Athlete Blue	Coordinator	F
6	Lousi	Praveen	Coordinator	<span style="color: red;">B</span>
7	Alia	Jiya	-----	
8	Bhagvashree	-----	-----	
9	Meet	Sujeet	-----	
10	Tisha	Mysha	-----	
11	Bhuwan	Bhuwan	-----	

Add Match

C
Select Data
Select Athlete
Scheduling

Tournament: GamersGame

Category: Under 41Kg

Gender: Male  Female  Other

BACK
NEXT

Match Management

Match
ADD MATCH

GId	Athlete Red	Athlete Blue	Coordinator	F
6	Lousi	Praveen	Coordinator	<span style="color: red;">B</span>
7	Alia	Jiya	-----	
8	Bhagvashree	-----	-----	
9	Meet	Sujeet	-----	
10	Tisha	Mysha	-----	
11	Bhuwan	Bhuwan	-----	

Add Match

C
Select Data
Select Athlete
Scheduling

Athlete Red: -----

Athlete Blue: -----

BACK
NEXT

## Live Score

Info Web Solution

Match Management

Gid	Athlete Red	Athlete Blue	Coordinator
6	Lousi	Praveen	Coordinator
7	Alia	Jiya	-----
8	Bhagyashree	-----	-----
9	Meet	Sujet	-----
10	Tisha	Mysha	-----
11	Bhuwan	Bhuwan	-----

Add Match

Select Data      Select Athlete      Scheduling

Match Date: mm/dd/yyyy  
Match Type: BACK

SUBMIT

### ➤ Edit:

Edit Match

matchStatus: Live  
Match Type: Quarter-Final  
Match Date: 06/14/2024

SUBMIT

### ➤ Assign:

Assign Match

Coordinator: Coordinator  
Referee 1: Referee1  
Referee 2: Referee2  
Referee 3: Bhagyashree

SUBMIT

### ➤ Assign Match :(View)



- Coordinator Dashboard
- Athlete
- Referee
- Coach
- Matches

#### ← Match Details

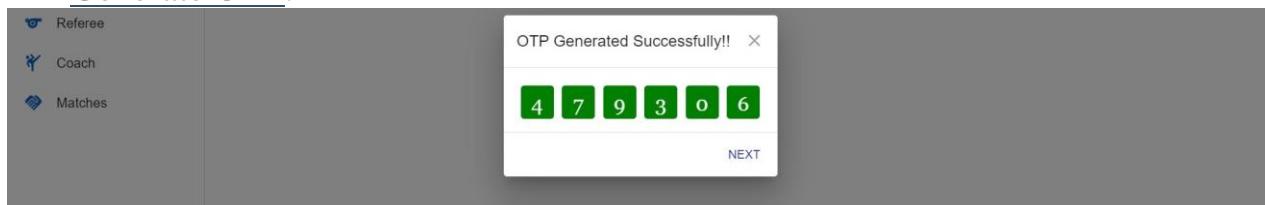
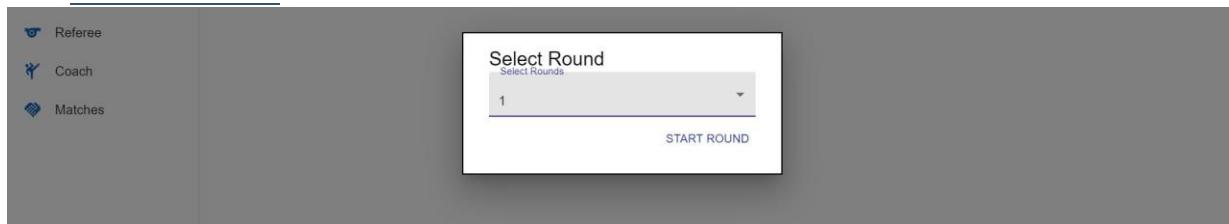


Mysha  
Gujarat  
Meet



Tisha  
Gujarat  
Prerna

Tournament	GamersGame
Category	Under 41Kg
Match Type	Semi-Final
Match Date	6/14/2024
Coordinators	Coordinator
Referee 1	Referee1
Referee 2	Referee2
Referee 3	Bhagyashree

Generate OTP:Select Round:Match Start:

Referees	Red Score	Blue Score	Red Penalty	Blue Penalty
referee1	3	0	0	0

Review Match:

Edit	RedPoints	BluePoints	RedPenalty	BluePenalty	ScoreTime	Timestamp
1	2	0	0	0	19/06/2024 20:47:35	19/06/2024 20:47:35
2	0	2	0	0	19/06/2024 20:47:36	19/06/2024 20:47:36
3	0	3	0	0	19/06/2024 20:47:37	19/06/2024 20:47:37
4	0	0	0	1	19/06/2024 20:47:38	19/06/2024 20:47:38
5	3	0	0	0	19/06/2024 20:47:39	19/06/2024 20:47:39
6	1	0	0	0	19/06/2024 20:47:41	19/06/2024 20:47:41
7	0	0	1	0	19/06/2024 20:47:42	19/06/2024 20:47:42
8	0	0	1	0	19/06/2024 20:47:43	19/06/2024 20:47:43
9	0	1	0	0	19/06/2024 20:47:44	19/06/2024 20:47:44

Edit Score:

The screenshot shows a modal window titled "Edit Score". It contains fields for "Red Points" (value 2) and "Blue Points" (value 0). Below these are "redPenalty" (value 0) and "Blue Penalty" (value 0). A "SUBMIT" button is at the bottom. To the right of the modal, a player card for "Praveen" is visible with the number 8.

Match End:

The screenshot shows a modal window titled "End Round". It displays "Red Total Score" (10) and "Blue Total Score" (11). The "Round Winner" section shows "Mysha" selected. Buttons for "END ROUND" and "Athlete" are at the bottom.

Round End:

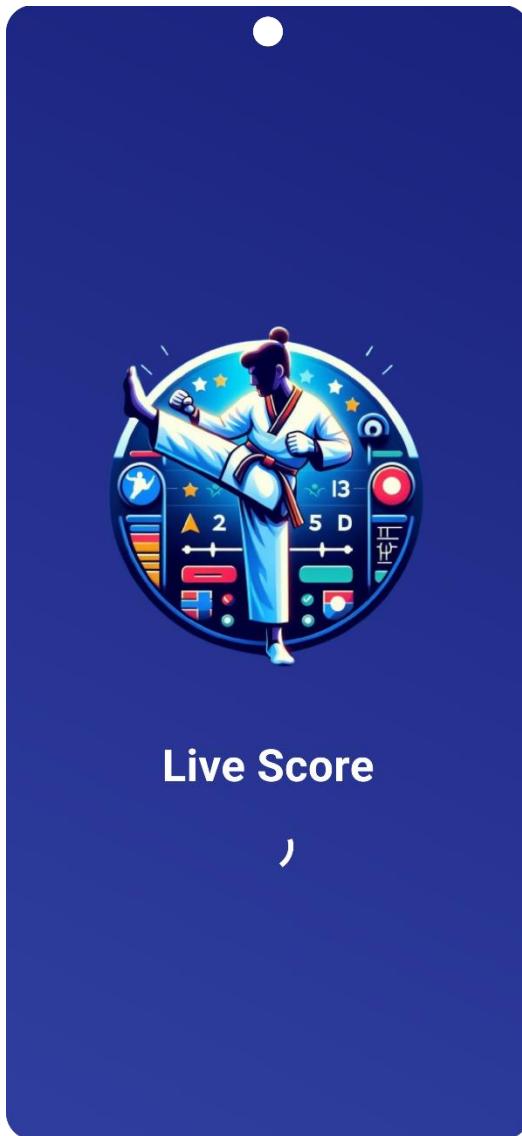
The screenshot shows a modal window titled "End Round". It displays "Round 1 Winner is Mysha". Buttons for "NEXT ROUND" and "END MATCH" are at the bottom.

Match End:

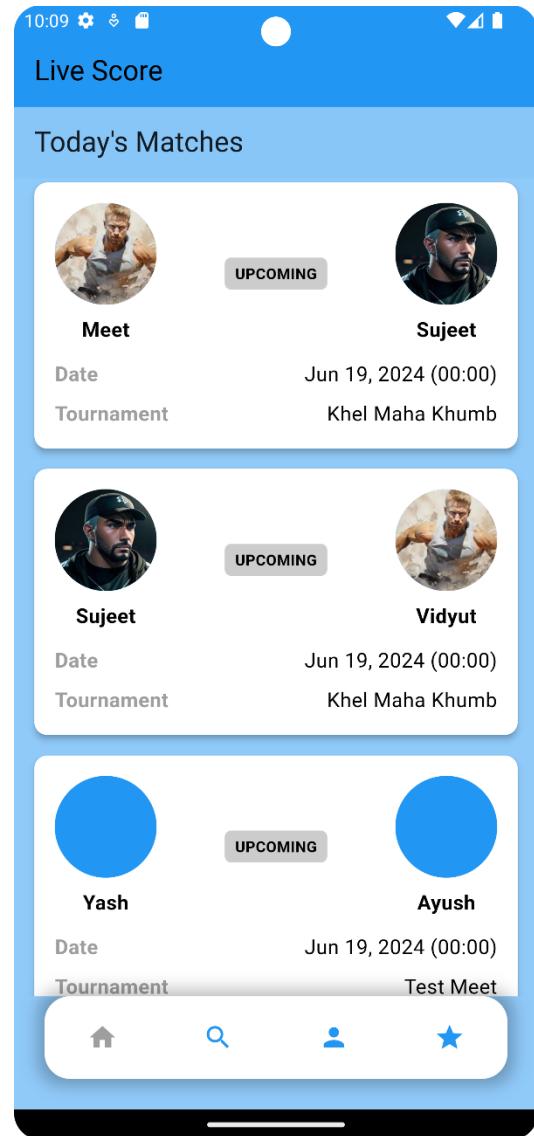
The screenshot shows a modal window titled "End Match". It displays "Select Winner" with "Mysha" selected. A "END MATCH" button is at the bottom.

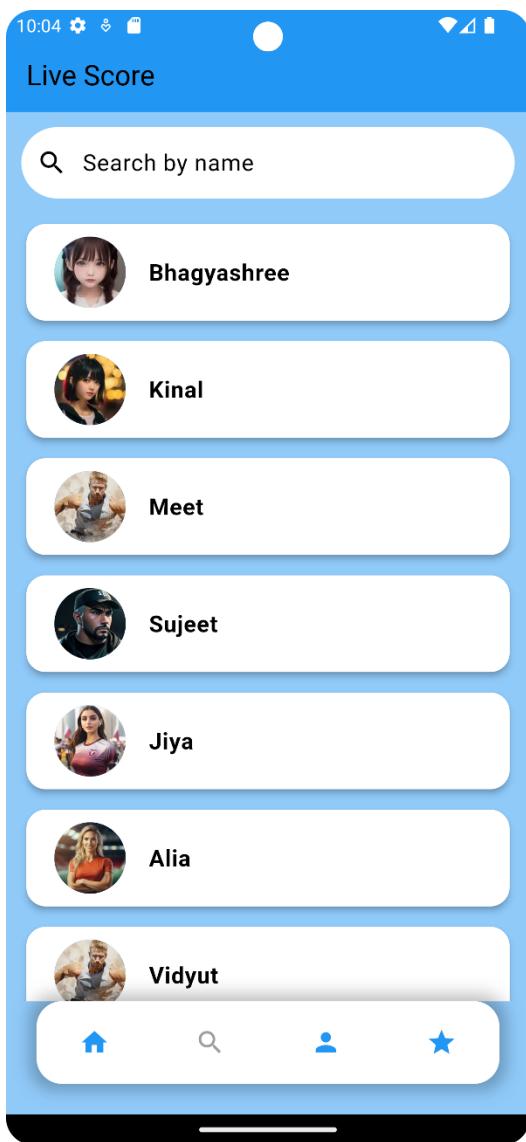
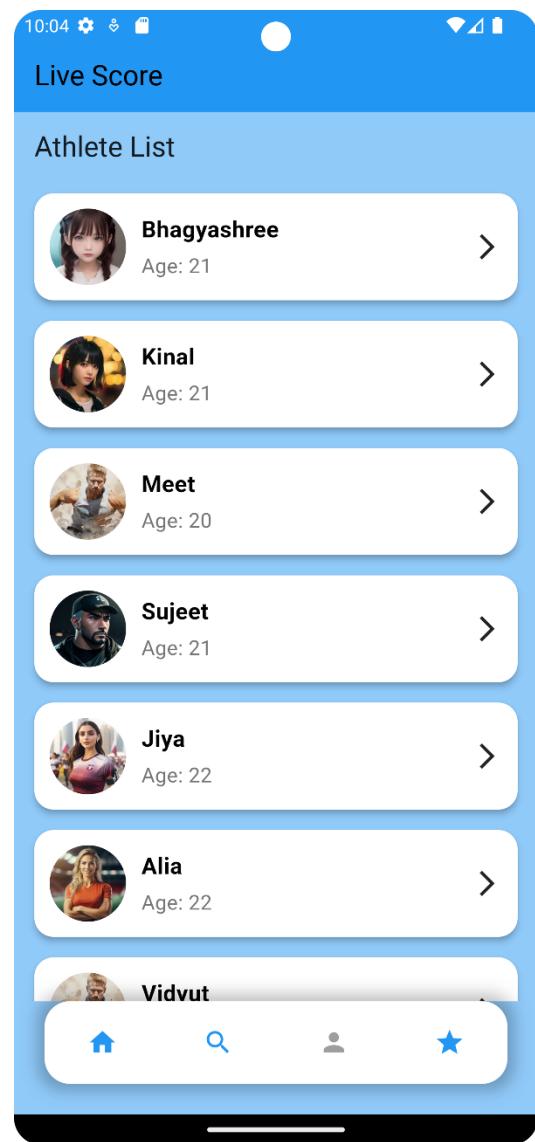
## View Side

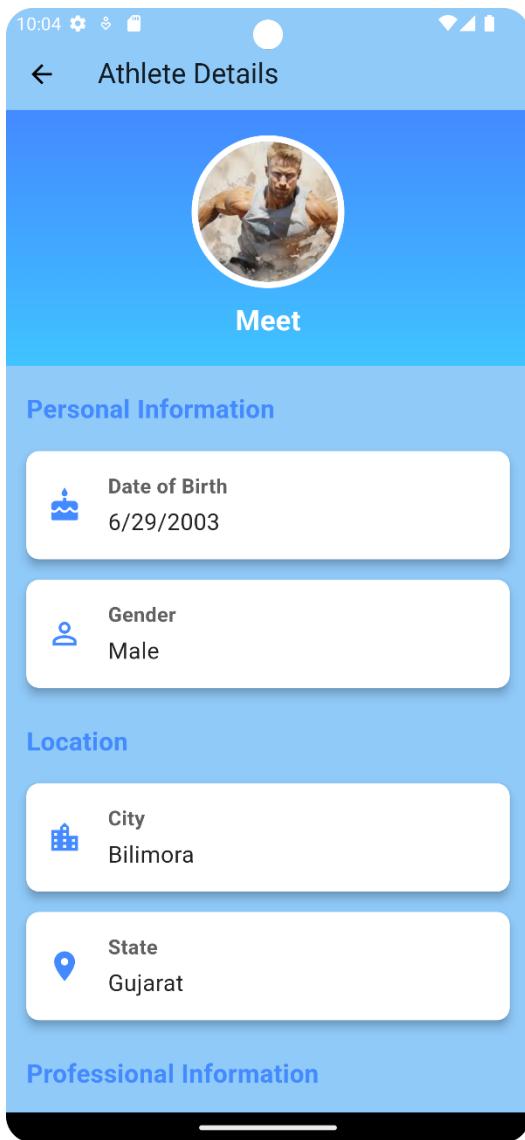
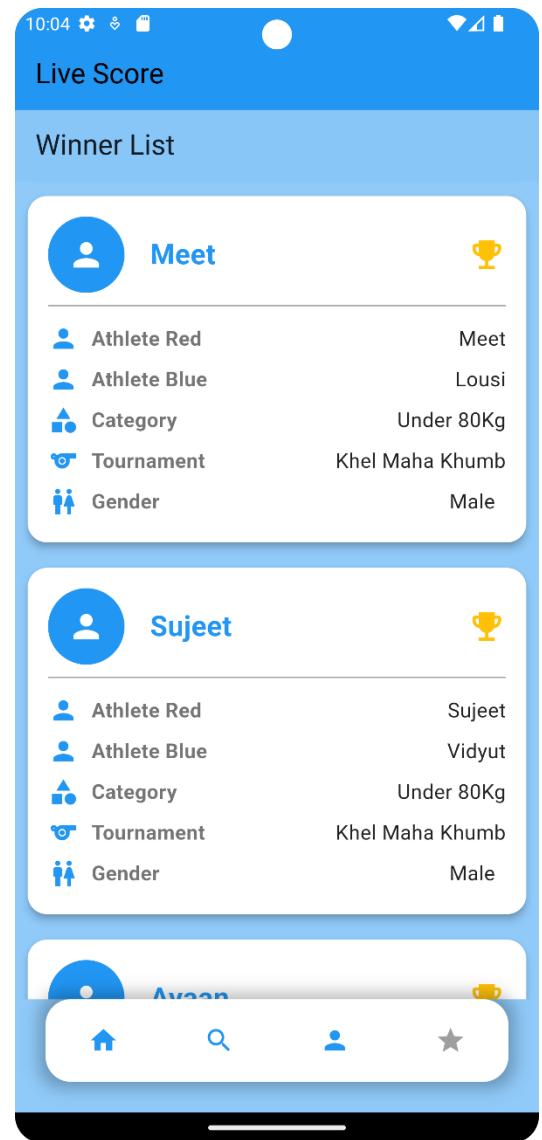
Splash Screen:



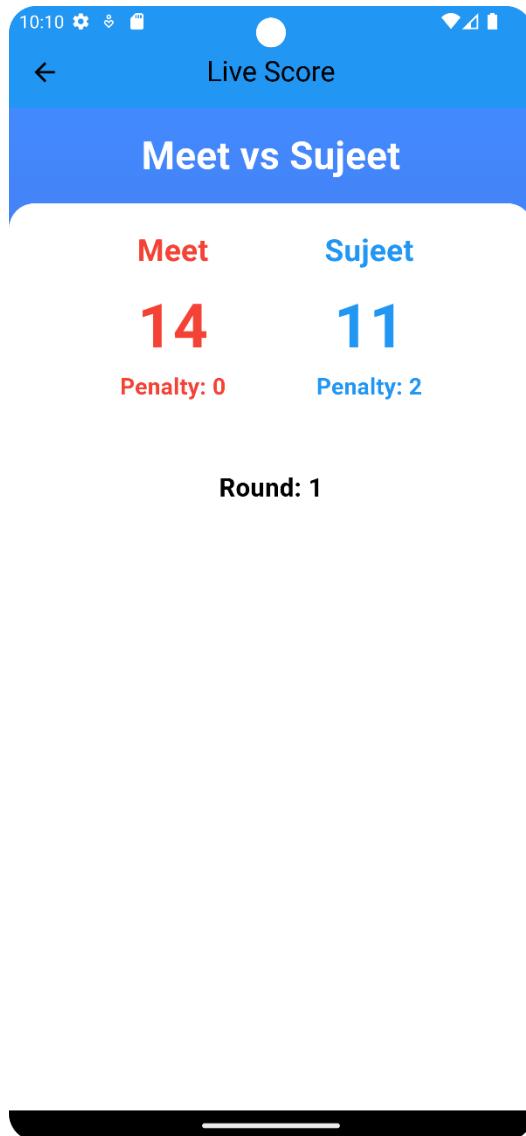
Home Screen:



Search Screen:Athlete Screen:

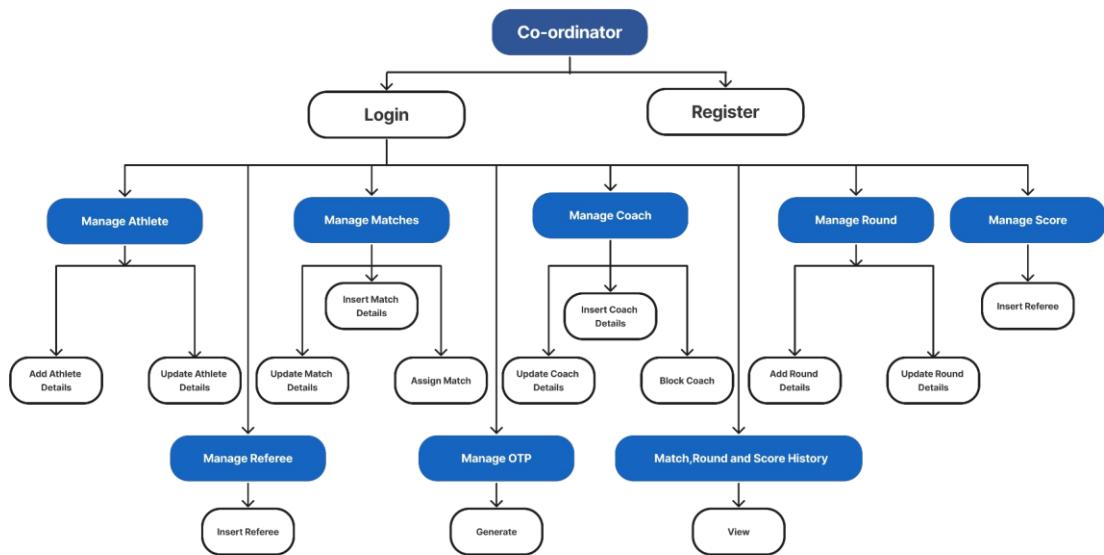
Profile Screen:Winner Screen:

Score board:



### 6.3 Architecture Design

Coordinator:



## Chapter7. Testing

### 7.1 Unit Testing

✓ Test Case For Login:

<b>Test Case ID</b>	<b>Test Field</b>	<b>Condition</b>	<b>Expected Result</b>
TC-LINK-01	Email	Valid email and correct password	Successful login
TC-LINK-02	Email	Valid email but incorrect password	Error message "Invalid credentials"
TC-LINK-03	Email	Invalid email format	Error message "Invalid email format"
TC-LINK-04	Password	Valid email and password, password encrypted	Successful login
TC-LINK-05	Password	Password too short (less than 6 characters)	Error message "Password too short"

✓ Test Case For Register:

<b>Test Case ID</b>	<b>Test Field</b>	<b>Condition</b>	<b>Expected Result</b>
TC-LINK-01	Email	Valid email and correct password	Successful login
TC-LINK-02	Email	Valid email but incorrect password	Error message "Invalid credentials"
TC-LINK-03	Email	Invalid email format	Error message "Invalid email format"
TC-LINK-04	Password	Valid email and password, password encrypted	Successful login
TC-LINK-05	Password	Password too short (less than 6 characters)	Error message "Password too short"

✓ Test Case For Forget Password:

<b>Test Case ID</b>	<b>Test Field</b>	<b>Condition</b>	<b>Expected Result</b>
TC-FORGET-01	Email	Valid email registered in the system	Password reset email sent
TC-FORGET-02	Email	Email not registered in the system	Error message "Email not found"
TC-FORGET-03	Email	Invalid email format	Error message "Invalid email format"

✓ Test Case For Entry in ACR:

<b>Test Case ID</b>	<b>Test Field</b>	<b>Condition</b>	<b>Expected Result</b>
TC-ADMIN-01	Email	Valid email format	Accepts valid email format
TC-ADMIN-02	Name	Valid name format	Accepts valid name format
TC-ADMIN-03	Password	Valid password format	Accepts valid password format
TC-ADMIN-04	ImageURL	Valid URL format	Accepts valid URL format
TC-ADMIN-05	Contact	Valid contact format	Accepts valid contact format
TC-ADMIN-06	Status	Valid status format	Accepts valid status format
TC-ADMIN-07	DateOfBirth	Valid date format	Accepts valid date format
TC-ADMIN-08	Age	Valid age	Accepts valid age
TC-ADMIN-09	LastLogin	Valid date format	Accepts valid date format
TC-ADMIN-10	Gender	Valid gender format	Accepts valid gender format

TC-ADMIN-11	City	Valid city format	Accepts valid city format
TC-ADMIN-12	State	Valid state format	Accepts valid state format
TC-ADMIN-13	RoleId	Valid role ID	Accepts valid role ID
12	Product Name	Valid	Order Details Created Successfully
13	Tax	Valid	Order Details Created Successfully
14	Meter	Valid	Order Details Created Successfully

✓ Test Case For Tournament:

Test Case ID	Test Field	Condition	Expected Result
TC-TOURN-01	TournamentName	Valid tournament name format	Accepts valid tournament name format
TC-TOURN-02	Venue	Valid venue format	Accepts valid venue format
TC-TOURN-03	TournamentDate	Valid date format	Accepts valid date format
TC-TOURN-04	TournamentCoordinator	Valid coordinator ID	Accepts valid coordinator ID

✓ Test Case For Match:

Test Case ID	Test Field	Condition	Expected Result
TC-MATCH-01	MatchStatus	Valid match status format	Accepts valid match status format
TC-MATCH-02	MatchType	Valid match type format	Accepts valid match type format

TC-MATCH-03	MatchDate	Valid date format	Accepts valid date format
TC-MATCH-04	Gender	Valid gender format	Accepts valid gender format
TC-MATCH-05	AthleteRed	Valid athlete ID	Accepts valid athlete ID
TC-MATCH-06	AthleteBlue	Valid athlete ID	Accepts valid athlete ID
TC-MATCH-07	NextMatchId	Valid next match ID	Accepts valid next match ID
TC-MATCH-08	MatchGroup	Valid match group format	Accepts valid match group format
TC-MATCH-09	Flag	Valid flag format	Accepts valid flag format
TC-MATCH-10	CategoryId	Valid category ID	Accepts valid category ID
TC-MATCH-11	TournamentId	Valid tournament ID	Accepts valid tournament ID
TC-MATCH-12	MatchCoordinator	Valid coordinator ID	Accepts valid coordinator ID
TC-MATCH-13	Referee1	Valid referee ID	Accepts valid referee ID
TC-MATCH-14	Referee2	Valid referee ID	Accepts valid referee ID
TC-MATCH-15	Referee3	Valid referee ID	Accepts valid referee ID

## 7.2 Integration Testing

### ➤ Integration between Backend and Frontend

<b>Test Case ID</b>	<b>Description</b>	<b>Steps</b>	<b>Expected Outcome</b>
INTG-BEFE-01	Verify login functionality	1. Enter valid credentials in frontend 2. Submit login request to backend 3. Backend verifies credentials and responds 4. Frontend receives response and updates UI	User should be authenticated and logged in
INTG-BEFE-02	Registration of new viewer	1. Fill registration form in frontend 2. Submit registration request to backend 3. Backend responds with success message 4. Frontend confirms successful registration	New viewer details sent to backend  Viewer account created in database
INTG-BEFE-03	Forgot password	1. Request password reset in frontend 2. Frontend sends request to backend 3. User receives password reset instructions	Email sent to registered email address  Backend processes request and sends email

## Chapter8. Future Enhancement

- Advanced Analytics and Reporting.
- Improved User Management.
- Expanded Features for Coordinators and Coaches
- Enhanced Scoring and Refereeing Tools.
- Enable social media logins and sharing
- Implement tools for data analytics.
- Support multiple languages.

## Chapter9. Glossary

<b>Term</b>	<b>Definition</b>
<b>Admin</b>	A user role responsible for managing categories, tournaments, verifying coordinators, viewing matches and history, and overall administrative tasks within the system.
<b>Athlete</b>	A participant in the Taekwondo tournament. Athletes are managed by coordinators and can be assigned to matches.
<b>Backend</b>	The server-side part of the application that handles business logic, database interactions, and API endpoints. For this project, it's implemented using Dotnet Core API.
<b>Category</b>	A classification used to group athletes based on criteria like weight, age, and gender.
<b>Coach</b>	A person responsible for training and mentoring athletes. Coaches are managed by coordinators.
<b>Coordinator</b>	A user role responsible for managing athletes, matches, referees, coaches, scores, and rounds.
<b>Database</b>	A structured collection of data stored electronically. This project uses SQL Server Management Studio (SSMS) for database management.
<b>Frontend</b>	The client-side part of the application that interacts with users, displaying data and capturing user inputs. For this project, it's implemented using Flutter Application.
<b>JWT Token</b>	JSON Web Token, a compact, URL-safe means of representing claims to be transferred between two parties. Used for API authentication in the project.
<b>Live Score</b>	Real-time display of match scores and updates. Implemented using SignalR for live score and round management.
<b>Match</b>	A competitive event between athletes. Matches are managed by coordinators and can have various statuses and types.
<b>Referee</b>	A user role responsible for managing scoring during matches.
<b>Round</b>	A subdivision of a match where athletes compete, and scores are recorded. Rounds are managed by referees and coordinators.
<b>SignalR</b>	A library for ASP.NET that enables real-time web functionality, allowing server code to push content to connected clients instantly as it becomes available.

<b>SSMS</b>	SQL Server Management Studio, a software application for configuring, managing, and administering all components within Microsoft SQL Server.
<b>Stored Procedure</b>	A prepared SQL code that can be saved and reused. In this project, logic is implemented in stored procedures for database operations.
<b>Tournament</b>	A series of matches held between athletes to determine a champion. Managed by coordinators and administrators.
<b>Viewer</b>	A user role with the ability to view match scores and updates but without the ability to manage any entities within the system.
<b>Email Services</b>	Functionality to send emails, used in the project for actions like registration, password reset, and notifications.
<b>Temporary Memory</b>	In-memory data storage used in the application for performance optimization, caching, or session management.
<b>React</b>	A JavaScript library for building user interfaces, particularly single-page applications where data can change without a page reload.
<b>Redux</b>	A predictable state container for JavaScript apps, often used with React to manage application state.
<b>API Authentication</b>	Process of verifying the identity of a user or process attempting to access the application's API endpoints, typically using tokens like JWT.
<b>Registration</b>	The process by which a new user creates an account in the system.
<b>Login</b>	The process by which a registered user gains access to the system by providing credentials.
<b>Logout</b>	The process by which a user ends their session in the system.
<b>Forgot Password</b>	A feature allowing users to reset their password via email if they have forgotten it.
<b>ID</b>	A unique identifier for records in the database, often set as a primary key.
<b>Primary Key</b>	A unique identifier for each record in a database table, ensuring that each entry is unique.
<b>Foreign Key</b>	A field in a database table that creates a link between two tables, enforcing referential integrity.

## Chapter10. Reference

- ✓ <https://www.figma.com/>
- ✓ <https://mui.com/>
- ✓ [https://learn.microsoft.com/en-us/sql/ssms/tutorials/ssms-configuration?  
view=sql-server-ver16](https://learn.microsoft.com/en-us/sql/ssms/tutorials/ssms-configuration?view=sql-server-ver16)
- ✓ <https://jwt.io/>
- ✓ <https://medium.com/thesignalgroup>
- ✓ [https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-  
8.0&WT.mc\\_id=dotnet-35129-website](https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-8.0&WT.mc_id=dotnet-35129-website)
- ✓ <https://www.npmjs.com/>
- ✓ <https://formik.org/docs/overview>
- ✓ <https://docs.flutter.dev/>