FUTSAL FIELD RESERVATION SYSTEM

CK-FA21-110734 - Ehtisham Naveed

CK-FA21-110420 - Saadullah

Supervised By

Prof. Nazish Bashir

Submitted for the partial fulfillment of the requirements for the degree of

Bachelor of Science in Computer Science

DEPARTMENT OF COMPUTING & TECHNOLOGY

IQRA UNIVERSITY CHAK SHEHZAD, ISLAMABAD

February, 2025

ABSTRACT

Due to the lack of centralised football booking platform in our country, both team players and ground owners have difficulty obtaining slot bookings. It's an altogether tiresome an time-consuming process both of the owners and the players with the owners having to wait for people to contact them in order to get their grounds booked, which is neither profitable not feasible from a business standpoint. And for the teams who have to travel from one venue to another in order to find a suitable pitch.

As a result of numerous challenges that our players face, we have aimed to design an online web-service that will help both teams obtain ground bookings and facility providers manage their facilities. We provide a simple solution by providing a centralised platform for both teams and ground owners, where ground owners can register their grounds on our website so that local teams can easily view and book grounds for their matches based on their preferred time and location. It was built using a structure design methodology, technologies like ReactJs for the frontend, NodeJs and Express for the backend, and MongoDB for data management and file storage. This software product will significantly improve the local football atmosphere of our twin cities.

Test cases include functionalities such as user registration, booking, donations. The system passed all test cases successfully. It was tested on Windows OS Opera-mini, and the overall results show that the application meets the functional and non-functional requirements. Testing was conducted using Black-box, White-Box, and GUI techniques.

The current scope of the project is limited to just twin cities but in the future we wish to expand it to a country-wide level and include national football leagues too and also include the mobile application support. These improvements will help increase the reach of the platform and make it more scalable.

CERTIFICATE

Dated: _____

Final Approval

It is certified that project report titled **" Fustal Field Reservation System"** submitted by **Ehtisham Naveed** and **Saadullah** for the partial fulfillment of the requirement of **"Bachelor's Degree in Computer Science"** is approved.

COMMITTEE

Dr. Tahir Ejaz
Associate Dean: Signature: _____

Dr. Imran Ali Qureshi
HoD C&T: Signature: _____

Ms. Tayyaba Rashid
Project Coordinator: Signature: _____

Ms. Nazish Bashir
Supervisor: Signature: _____

DECLARATION

We hereby declare that our dissertation is entirely our work and genuine / original. We understand that in case of discovery of any PLAGIARISM at any stage, our group will be assigned an F(FAIL) grade and it may result in withdrawal of our Bachelor;s degree.

Group members:

Name	**Signature**
Ehtisham Naveed	_____
Saadullah	_____

PLAIGRISM CERTIFICATE

This is to certify that the project entitled "Futsal Field Reservation System", Which is being submitted here with for the award of the " Degree of Bachelor's in Computer Science". This is the result of the original work by Ehtisham Naveed and Saadullah under my supervision and guidance. The work embodied in this project has not been done earlier for the basis of award of any degree or compatible certificate or similar tile of this for any other dimploma/examining body or university to the best of my knowledge and belief.

Turnitin Originality Report

Processed on
ID:
Word Count:

Similarity Index
10%
Similarity by Source
Internet Sources:
Publications:
Student Papers:

Date: _____

Prof. Nazish Bashir

TURNITIN ORIGINALITY REPORT

Group Based Online Streaming Application using Android Mobiles [BSCS] by
Ehtisham Naveed and **Saadullah**
From **Prof. Nazish Bashir**

Processed on
ID:
Word Count:

Similarity Index
10%
Similarity by Source
Internet Sources:
6%
Publications:
0%
Student Papers:
8%

SOURCES:

1.
2.
3.

ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to our project guides for providing all of the necessary materials, making valuable suggestions, and encouraging us throughout the project's duration. We would also like to thank IQRA UNIVERSITY CHAK SHAHZAD for providing us with a conducive environment in which to complete our project work.

TABLE OF CONTENTS

Chapter   **Page**

LIST OF FIGURES

Figure	Caption	Page No.

LIST OF TABLES

Chapter 1
INTRODUCTION

There are many futsal grounds in Rawalpindi and Islamabad but if a local team wants to book a slot for their football game in any of these grounds, first they will have to go through each and every ground owner's contact information to inquire if a slot is available or not. If the slot is already booked then they will have to repeat the entire process again which can prove to be quite the tedious process.Moreover, if they do find a slot they will still need to find players to play against. In this way such hurdles deter people in trying to play a simple game of football. **Futsal Field Reservation system** is a service that will allow local football teams, to reserve a pitch for their football games while simultaneously helping ground owners manage bookings for their facilities. Our platform offers a solution that streamlines facility management for providers. While also providing for the general public who want to book, practice and play on sports facilities. Our mission is to provide the local football by minimizing the ground booking hurdles that most of the players residing in our country face. And help turn ground renting into a lucrative business for the ground owners.

Our System will also offer an academy option, for people who are looking to learn, practice and hone their football skills, they can easily register through our website for academies of whichever ground is offering. We also aim to setup a donation system for impoverished players, who wish to play the game but are unable to due to various financial issues. So that everyone has equal chance. We are committed to increasing public access to football sports facilities.

1.1. Project Domain:

The category of our project is Web Application Development and Online Reservation Systems. The project will help tackle the real-life issue experienced by local footballers and futsal ground owners in Pakistan, especially in the twin cities of Rawalpindi and Islamabad. Our system fills this void by creating a web based platform through which teams can see, choose, and book existing slots online, and ground owners can add and maintain their venues. By eliminating this process through automation, the project maximizes convenience, minimizes time, and facilities a better-structured sports infrastructure.

1.2. Problem Identification:

No such system currently exist in Pakistan that serves the local football community .As a result Football players have to face so many issues when they were trying to book the ground. It's a time consuming process on the owner's and player's, for the owner's having to wait for people to contact them to get their grounds booked which from a business perspective isn't profitable. And for the players having to contact each and every ground owner that they know of, and then trying to find if a team which will be available at the same time as them, which all in all proves to be quite the hassle.

So to provide the solution of these time consuming hurdles, we will offer our services both to the teams and ground owners, in the form of our website, **"Futsal Field Reservation System"** which will provide a friendly interface for both the footballers and ground owners to communicate. Our system will provide updates about slot is available or not , other teams can also chat with each other and discuss about the match should be held and other matters related to futsal.

1.2.1.Proposed Solution:

The proposed system will help footballers to make ground reservations for their games and at the same time facilitate ground owners to register their grounds with our system and get booking in an efficient and timely manner. The solution will come to play

by providing a web-based service for both footballers and ground owners alike, where football players will have the option of viewing lists of grounds and instantly getting to know if a slot is available for booking or not. The system will offer features like user registration and login, ground and team profile management, real time chat for communication, academy registration, and a donation module.This project aims to help local football players of twin cities to book their matches, and professional manner rather than trying to contact each ground owner first for booking a slot and then rushing to find opponents and grounds.The project will be deployed as a web-based service with tools such as ReactJs, NodeJs, and NoSQL.This project will surely help in improving the futsal field reservation system in our twin cities.

1.2.2. Objectives:

The intended objectives of this product will be as follows:

• To create a centralized web system that makes it easy for local football clubs to Book futsal grounds in the twin cities.
• To create a platform where facility owners can register, schedule, and market their facilities effectively.
• To give ground owners complete control over their profiles, such as booking calenders, facility information, and media uploads.
• To enable a communication through which teams can communicate with each Other and with ground owners for coordination and scheduling purposes.
• To incorporate an academy registration functionality for users interested in participating in football training schemes.
• To add a donations feature for sponsorship of underprivileged players that do not have the financial capability for football kits, training, or match charges.

1.2.3. Scope of the Project

The Futsal Field Reservation system is a service that allows local football teams to reserve a pitch for their games while also assisting ground owners in managing bookings for their facilities. Our platform provides a solution that greatly simplifies facility management for providers while also allowing the general public to book, pay for, and use sports facilities.

Our mission is to provide local football players in the twin cities with the opportunity to play a proper and professional game of football by reducing the ground booking obstacles that most players in our country face. And assist ground owners in turning ground renting into a profitable business. We also want to create a donation system for poor players, who want to play but cannot due to various financial issues.So everyone has the same chances.

1.3. Effectiveness/ Usefulness of the System

The Futsal Ground Booking System makes a valuable contribution when compared to other available systems, especially in the Pakistani scenario where no special platform is available today for online booking of futsal grounds. Unlike international sites like Pitch-booking or GWSports, our system is specifically designed for the twin cities Rawalpindi and Islamabad for local footballers and ground Owners. The website allow users to browse grounds available, browse booking slots, and book with ease sparing the time consuming and laborious exercise of calling ground owners one by one. Ground owners gain a professional website to advertise their grounds, administer bookings, and capture more clients. Other features such as live chat, academy

registration, and donation facilities enhance inclusiveness and community nature of the system.

1.4. Resource Requirement

This section provides the necessary resources for the effective development and implementation of the project. It encompasses both the hardware and software resources, including development tools, programming languages, framework, and hosting environments.

1.4.1. Hardware Requirement

The creation and implementation of the this project need a computer with average specifications that can be used in web development. The system was developed and tested on devices that can support development tools such as Visual Studio Code, Nods.Js, and MongoDB. The Tabulation of the suggested hardware specifications are below:

Table 1.1. Hardware Requirement

COMPONENT	SPECIFICATIONS
Processor	Intel Core i5 or Higher
RAM	8 GB or higher
Storage	256 SSD
OS	Windows 10, 11
Internet	Stable Broadband

1.4.2. Software Requirement

The creation of the Futsal Field Reservation System involved several software tools, frameworks, and programming languages in the implementation of both the front-end and back-end of the application. The tabulation of the major software and tools are mention below:

Table 1.2. Software Requirements

Tool / Software	Purpose
ReactJs	Frontend Development
NodeJs & ExpressJs	Backend Server
MongoDB	Database Management

		Table
Firebase	Image and File Storage	
Visual Studio Code	Code editing and Development	
Google Chrome	Web application Testing / Browser	
Google Ai Studio	Gemini Api Key	

1.4.3. Data Requirement

There are no machine learning models in this project, so no training or test datasets are necessary. All data, including user profiles, booking details, ground listings, and donations history, are stored and administrated dynamically via the MongoDB database as people use the site.

1.5. Report Organization

This project report is systematically organized into six chapters to provide a comprehensive overview of the Futsal Field Reservation System, from its conception to its final evaluation. Chapter 1 serves as the introduction, outlining the problem statement, project scope, and a review of existing systems. Chapter 2 provides a detailed description of the proposed system, clarifying its objectives, features, and overall project overview. Chapter 3 focuses on the requirements specifications, detailing the functional and non-functional requirements along with the graphical user interface design. Chapter 4 is dedicated to system modeling and design, presenting the system's architecture through various UML diagrams such as Use Case, Activity, Sequence, and Class diagrams. Chapter 5 covers the critical phase of system testing and validation, explaining the testing techniques employed and documenting the specific test cases used to verify the system's functionality. Finally, Chapter 6 concludes the report by summarizing the project's achievements, acknowledging its limitations, and recommending potential directions for future work.

CHAPTER 2
BACKGROUND AND EXISTING SYSTEMS

The Futsal Field Reservation System caters to the increasing demand for a centralized and organized way to reserve futsal grounds in city towns, especially in the twin cities of Rawalpindi and Islamabad. Presently, ground owners and players are struggling with major obstacles since there is no digital platform available for ground management and booking.

In this chapter, we have talked about the available systems like Pitchbooking(UK), Gwsports(india), and other existing similar systems providing ground booking facilities in various countries. We have pointed out their features, usage, and major differences. We have determined the weakness of these systems, particularly their inability to support localized requirements and community based functionalities like academies and donations that our suggested system will address.

Table: (Bench Marking)

System	Chat Box	Ground Owner's Dash-Board	Teams Dash-board	Academy	Listings	Map Based	Donations
Pitchbooking	☑	☑	☒	☒	☑	☑	☒
Gwsports	☒	☒	☒	☒	☑	☒	☒
PlaySpots	☑	☒	☒	☒	☑	☒	☒
Futsal Field Reservation System	☑	☑	☑	☑	☑	☑	☑

2.1. Related Literature Review

Paper 1: Pitchbooking(UK)

Overview: Pitchbooking is a UK-built online booking system where users can discover, observe, and reserve sports pitches such as football pitches and tennis courts within different cities.

Findings: It provides an easy-to-use interface for viewing real-time availability, automatic confirmation of booking, and facility management software for ground owners. The system enhanced operational effectiveness of the customer for private sports providers and local councils.

Limitations: Pitchbooking is primarily directed towards developed nations and lacks community features such as team matchmaking, academy integration, or donation support for needy players.

Paper 2: Gwsports(India)

Overview: Gwsports is an Indian website that brings players into contact with sports grounds for the likes of cricket, badminton, and football. It allows users to book pitches according to location and desired time using a web and mobile platform.

Findings: The system enhances access to sporting facilities and aids event management and sports coaching. It has a storing presence in tier-2 and tier-1 Indian cities, hence applicable for new markets in sports.

Limitations: Gwsports does not have any social interaction feature-teams, academy cannot converse directly. Nor does it offer any provision to assist cash-strapped players.

Paper 3: PlaySpots(India)

Overview: PlaySpots is yet another Indian platform that aims to make sports complexes accessible by enabling real-time bookings of courts and grounds for different sports such as futsal, tennis, and basketball.

Findings: It provides facilities such as location based venue search, time-slot reservation, and user reviews of venues. Its mobile application allows click booking and reminders.

Limitations: PlaySpots, although effective at booking facilities, lacks attributes such as infrastructure for donations, team profiles, and academy sign-ups--restricting the opportunities for community creation.

Table 2.1: Summary of Reviewed Literature

Year	Authors	Contribution	Techniques	Limitations
2018	PitchBooking (UK)	Provided a centralized platform for booking sports fields	Web & mobile platform with dashboard	No team interaction, donations, or academy features
2019	GwSports	Connected users with sports grounds and coaching centers	Web & mobile Based booking system	No team interaction, and donations features
2020	PlaySpots	Allowed real-time booking of Spots venues via mobile app	Location-based Facility search app	Lacks team features, no donations or academy management system

2.2. Related Systems / Applications

Some systems already exists with the aim of streamlining sports ground booking, each one with special characteristics. Below are two dominant systems that motivated our solution.

System 1: Pitchbooking(UK)
Pitchbooking is an online booking platform for football and other sports pitches. It features live availability, automated confirmation of bookings, and facility management tools. It does not have team profile creation, chatting options, or support for community-

based sports development, which makes it less useful in developing nations such as Pakistan.

System 2: Gwsports(India)
Gwsports provides user with the opportunity to reserve sports centers in different Indian cities. It provides event organizing and training services. It is widely available, it lacks the facility of real-time chat between users as well as donation options for needy players, so it is less friendly to community-oriented objectives.

<div align="center">

Table 2.2: Summary of Existing System

</div>

Year	System	Contribution	Tool / Technologies	Limitations	Application
2018	Pitchbooking	Online sports pitch booking and management	Web & mobile platform	No team chat, no donations, lacks communities features	Sports facility booking
2019	Gwsports	Sports ground booking & coaching connection	Web & mobile app	No team profiles, no chat, no donation support	Event & venue booking

2.3. Identified Problem from Work

The currently available systems like Pitchbooking(UK) and Gwsports(india) offer limited functionality for sports ground booking and scheduling management. These platforms, come with some major limitations. Technologically, the majority of them do not offer real-time chat support, and thus teams ad ground owners cannot communicate in real-time. Functionality, they do not support features such as team profile setup, which are integral in developing an integrated football community.

From a non-functional point of view, most systems are not localized or adapted to developing nations and do not feature donation support options for less privileged players. In order to enhance these limitations, our project aims at a centralized web-based system that combines communication tools, academy features, and donation modules to develop a more accessible and efficient booking platform for ground owners and local football players.

2.4. Selected Boundary For Proposed Solution

The Futsal Reservation System proposed in this project will incorporate major functionalities like user registration, ground listing and reservation, profile management, real-time chat, and academy registration. These functionalities are required to solve the issues recognized in Section 2.3, which involve in sufficient communication, centralized booking, and opportunities for skill development. The system will also incorporate a donation feature to help unprivileged players so that the system is more inclusive and community oriented.

However, some advanced features will not be supported because of time and resource limitations. They are mobile application creation, and support for sports other than futsal. These features are useful, they are beyond the initial scope and can be taken into consideration at future upgrades. By explicitly specifying what is included and excluded, we define a realistic and workable scope for our project development.

CHAPTER 3
SYSTEM REQUIREMENT AND SPECIFICATIONS

This chapter presents the detailed functional and no-functional specifications of the Futsal Field Reservation System, encompassing a complete technical view and system requirements for the development of the system. The requirements are divided to maintain clarity, feasibility, and accordance with the project goals. Functional requirements specify the system primary functions, including user registration, ground booking, and communications features, whereas non-functional requirements cover performance, usability, security, and compatibility.

The chapter divided into sections for orderly presentation. Section one presents the document structure and convention. The next sections cover the functional requirements with descriptions of the system's features and interactions, followed by the non-functional which encompass operational constraints and quality attributes. Further, User interface design is shown to give a visual description of the interaction points of the system. This systematic methodology provides a clear system of understanding the requirements of the system and forms the basis of its design and implementation.

3.1. System Specification

This section provides the main technical and operational specifications of the Futsal Field Reservation System, providing a clear picture of its form and functions. Here, we explain the system's functional requirements like user sign-up, pitch-booking, team messaging, AI chat bot, and donation tracking which are the essential features users will be working with. We also discuss non-functional requirements, security features, usability requirements, and cross platform compatibility. These requirements form the backbone for development, informing the design and implementation stages to build a smooth, effective, and user-friendly system for football teams and ground owners. By tackling these issues, we ensure the system is viable in real world scenarios as well as ensure reliability and scalability for future advances.

3.2. System Modules

System modules are standalone, clearly defined elements of a software system, each performing one particular functionality. By segmenting the system into modules, we achieve improved organization, maintenance simplicity, and scalability. The Futsal Filed Reservation System is segmented in to the following main modules:

User Management: Manages registration, authentication, and profile handling for teams and ground owners.

Ground Booking: manages slot booking, checking availability, and scheduling.

Team Communication: Facilities communication between teams and ground owners through chat.

Academy Registration: Allows singing up of users with football training academies.

Donation System: Controls donations on behalf of poor players.

Admin Dashboard: offers administration control for user and ground administration.

This modularity enhances flexibility every component can be written, tested, and upgraded separately while maintaining perfectly coordinated integration.

3.2.1. User Management

This Module manages user sign-up, login, and ground owner/teams profile customization for two main roles: ground owners and teams. Ground owners list their venues, prices, and availability, whereas teams set up profiles with players names and levels of skill. Email/password secure authentication maintains data confidentially. The modularity makes role based access easy to implement and expand on in the future.

3.2.2. Ground Booking

The central component of the system, this module enables teams to search, browse, and reserve available slots on registered grounds. There are real time availability updates, location/price filtering, as well as automated confirmation messages. Through separate booking logic, we achieve scalability new functionality can be implemented without affecting other modules.

3.2.3. Team Communication

A dedicated in app chat feature allows teams to communicate with their opponents and negotiate with land owners. The module supports real time messaging without intermingling with booking workflows to keep things simple.

3.2.4. Academy Registration

Owners of grounds with training programs can register their academies, while users register through organized forms. This module is independent and supports integration with payments processing and class scheduling, with the potential to link to the practice slot booking system.

3.2.5. Donation System

Secure payment gateway enables users to donate money for underprivileged players. Separation module ensures compliance with financial laws and ease of auditing. Transparency features may be implemented subsequently.

3.3. Functional Requirements/Software Features

The Futsal Field Reservation System is engineered to give flawless experience to both football teams and ground owners by providing core functionalities that relate to the goals of the project. The major features of the system involve user registration and authentication, management of grounds booking, team communication, registration of the academy, and a donation system. All the functional requirements are precisely designed to meet particular user needs while maintaining efficiency, reliability, and ease of use. Here is a detailed analysis of these features.

3.3.1. User Registration and Authentication

The system shall allow users to register as either teams or ground owners by providing essential details such as name, email, and password. Registered users can login securely using their credentials. This feature ensures that only authenticated users can access the system's functionalities, maintaining data privacy and security.

3.3.2. Ground Booking Management

Teams shall be able to search for available futsal grounds based on location, date and time. The system will display real time availability and allow teams to book slots instantly. Ground owners can update their slot availability and manage bookings through their profiles. This feature streamlines the booking process, eliminating the need for manual inquires.

3.3.3. Team Communication

The system shall include an in app chat feature that enables teams to communicate with each other and with ground owners. This functionality facilities match coordination, dispute resolution, and general inquires, enhancing user interaction and convenience.

3.3.4. Academy Registration

Ground owners offering football training academies shall be able to visit to list their programs, including details like schedule, fees, and coach information. Users can browse and register for these academies through the system. This feature promotes accessibility to training opportunists and supports grassroots football development.

3.3.5. Donation System

The system shall provide a secure platform for users to donate funds to support underprivileged players. Integrated payment gateways will ensure smooth transactions, and donors will receive confirmation receipts. This feature ensures efficient system maintenance and provides oversight to prevent misuse.

3.4. Non-Functional Requirements

Non-functional requirements specify the qualities of a system in operation, but not its particular behaviour. NFRS are what guarantee the Futsal Field Reservation System provides a secure, dependable, and convenient user experience. Our NFRS include performance, security, usability, compatibility, and scalability each chosen with great care to ensure the long term sustainability and user satisfaction of the system. The following are the main non-functional requirements for our application.

3.4.1. Performance

The system shall:

1. Load all interactive pages within 2 seconds undernormal network conditions (<5Mbps)
2. Support 100 concurrent users without degradation in response time
3. Process booking transactions

Rationale: Response times are essential for user retention, particularly when viewing real time ground availability. These performance metrics were compared to similar booking sites.

3.4.2. Security

The system shall:

1. Encrypt all senstive data(passwords, payments)
2. Maintain audit logs of all financial transactions

Rationale: As the system stores personal data and takes payments, strong security controls are necessary for user confidence and regulatory compliance.

3.4.3. Usability
The system shall:

1. Obtain a System Usability Scale (SUS) score > 75 on first round user testing
2. Make all critical functionality available within 3 clicks
3. Provide tool help icons fro complex features like donation processing

Rationale: our users are non-technical ground owners and players, so intuitive design supported by usability testing is critical.

3.4.4. Compatibility
The system shall:

1. Support Chrome, FireFox, Opera, Safari , Edge
2. Maintain responsiveness at screen resolutions from 360x640px to 1920x1080px

Rationale: Field Booking often occur on mobile devices, necessitating cross platform compatibility.

3.4.5. Scalability
The system architecture shall:

1. Handle 3x present user load without the need for infrastructure modifications
2. Support addition of new sports facilities
3. Have performance while scaling to new cities

Rationale: The platform is intended for future expansion beyond futsal and the twin cities metro area.

3.4.6. Reliability
The system should:

1. Have 99.5% uptime excluding planned maintenance
2. Automatically recover from system crashes within 30 seconds
3. Have all transactions retained during failures

Rationale: High availability is needed for booking systems, particularly evening/weekend peak hours.

3.4.7. Maintainability
The codebase should:

1. Contain dosumentaion for each API endpoint
2. Have test coverage>80% for important modules
3. Support feature updates with < 2 hours of developer onboarding

Rationale: Guarantees long term sustainability and effective team collaboration.

Chapter 4
SYSTEM MODELING AND DESIGN

System modeling and design are critical stages in software development, serving as the foundation for building reliable, scalable, and user-oriented applications. System analysis helps identify the functional and non-functional requirements of the system, while design translates those requirements into technical blueprints that guide implementation. These processes reduce ambiguity, improve communication among stakeholders, and ensure the system is developed in a structured and efficient way.

In this project, the system under discussion is the Futsal Field Reservation System, a web-based platform that simplifies the ground booking process for football teams and enables ground owners to manage their facilities. To support its design, we used various modeling techniques, including Context Diagrams to represent interactions with external entities, Data Flow Diagrams (DFD Level 0 and 1) to show how data moves through the system, a System Architecture diagram to outline the technical structure, and a High-Level Design table to highlight major components and constraints.

Each diagram was selected based on the nature and scope of the system, ensuring clarity and completeness in understanding how different parts interact. These models play a key role in visualizing system behavior, enhancing the development process, and ensuring the final product meets user needs effectively.

4.1. System Design and Analysis

In the development of the Futsal Field Reservation System, various analysis and design models were employed to ensure a structured and comprehensive understanding of the system's functionality. A Context Diagram was used to represent the system as a single process and to illustrate its interaction with external entities such as teams, ground owners, and administrators. To further detail the internal processes, Data Flow Diagrams (DFD Level 0 and Level 1) were created. The Level 0 DFD provided an overall view of the major processes and data flows within the system, while Level 1 broke these down into more detailed sub-processes.

A System Architecture Diagram was also developed to depict the technical structure of the application, showcasing the relationships between the front-end, back-end, database, and integrated services. Additionally, a High-Level Design Table was prepared to define the system's main modules, the operating environment, selected tools, and constraints. These models were chosen based on the specific needs and complexity of the system, supporting clear visualization and efficient system development.

4.2. Use Case Diagrams

Use case diagrams are an essential part of system analysis as they depict the functionalities of a system from an end-user's perspective. They are crucial for understanding how different users, known as actors, will interact with the system and what functions they can perform. A use case diagram is mandatory for any system that involves user interaction, as it helps to clearly define the scope and requirements before development begins. It can only be skipped in the rare instance that a system has no direct user interaction.

The Use Case Diagram for the Futsal Field Reservation System is shown below (as shown in Figure 4.1). It identifies the primary actors and their interactions with the system's core functionalities. The two main actors are the Team and the Ground Owner. The diagram illustrates how a Team can perform actions such as registering, searching for grounds, booking a slot, and managing their profile. Similarly, a Ground Owner can register, manage their ground's details, and view bookings. Both actors share common

functionalities like logging into the system, communicating via the chat feature, and getting assistance AI-powered Futsal Assistant. This model provides a clear and comprehensive view of the system's intended behavior from the perspective of its end-users.
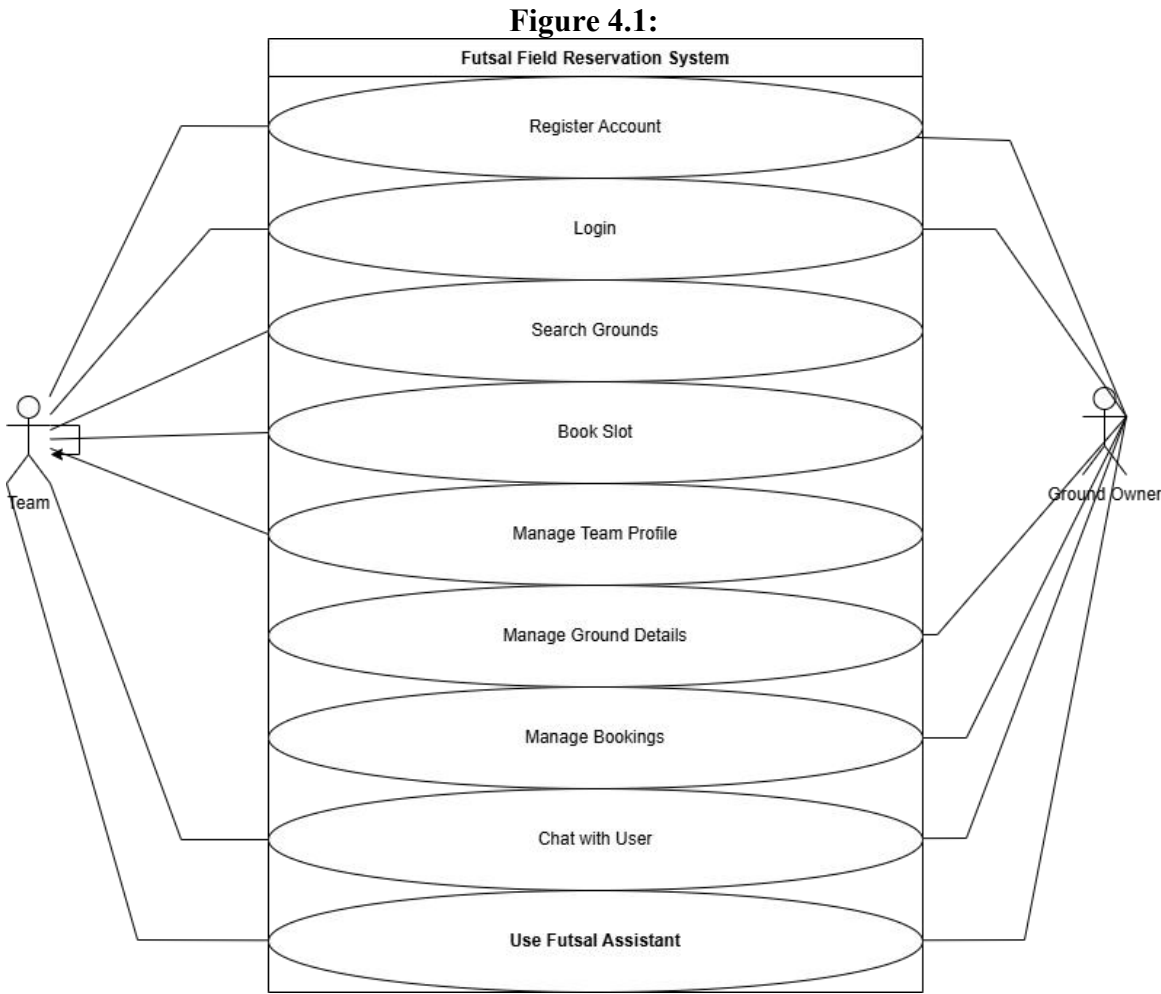
Figure 4.1:

Figure 4.1: Use Case Diagram

4.3. Full Dress Use Case/Detailed Use Case

While a use case diagram provides a high-level overview of system functionalities, a full dress or detailed use case is a textual description that elaborates on the interaction between an actor and the system. Its importance lies in its ability to capture the intricate details, flows, and exceptions of a specific function, which is crucial for complex processes. Full dress use cases are not developed for every function they are reserved for those that are core to the system's purpose or involve multiple steps and potential alternative paths.Full dress use case is as follows:

4.3.1. Full Dress Use Case for Book Slot

Table 4.1: Full Dress use case for book slot

Use Case Selection	Comment
Use Case Name	Book Slot

Scope	Futsal Field Reservation System
Level	User goal
Primary Actor	Team
Stakeholders and Interest	**Team:** Wants to find and book an available ground easily. **Ground Owner:** Wants their available slots to be booked to generate revenue.
Pre-conditions	The Team must be logged into the System.
Success Guarantee	The selected time slot is reserved for the Team, an entry is made in the system's booking record, and the ground Owner is notified.
Main Success Scenario	1. Team navigates to the Booking or Search Grounds page. 2. System displays a list of a available grounds. 3. Team Selects a specific ground. 4. System displays the ground's details and a calender of available slots. 5. Teams selects a desired date and slot. 6. Team clicks 'Confirm Booking'. 7. System validates the slot is still available, reserves it and displays a confirmation message to the Team.
Extensions	**No slots available:** If the selected ground has no available slots, the system displays a message indicating this. **Slot just became Unavailable:** If another teams books the slot during the process, the system displays an error message and prompts the user to select another slot.
Special Requirements	The system must provide real-time to prevent double booking. The interface must be responsive and work on mobile devices.

4.3.2. Full Dress Use Case for Manage Ground Details

Table 4.2. Use case For manage ground details

Use Case Selection	Comment
Use Case Name	Manage Ground Details
Scope	Futsal Field Reservation System
Level	User goal
Primary Actor	Ground Owner
Stakeholders and Interest	**Ground Owner:** Wants to present their facility attractively and accurately to attract bookings. **Team:** Wants clear, accurate information to choose a suitable ground.
Pre-conditions	The Ground Owner must be logged into their account.
Success Guarantee	The ground's profile is successfully created or updated in the system database and is visible to all teams using the platform.
Main Success Scenario	1. Ground Owner logs in and navigates to their dashboard/profile. 2. Ground Owner selects the option to 'Add' or 'Edit' a ground. 3. System presents a form for ground details (name, location,

	price, etc.).
	4. Ground Owner fills in the details. 5. Ground Owner uploads photos of the facility. 6. Ground Owner sets the available time slots and any specific rules. 7. Ground Owner saves the changes. 8. System validates the data, saves it to the database, and displays a 'Successfully Updated' message.
Extensions	**Invalid file format:** If the user tries to upload a file that is not a supported image format (e.g., JPG, PNG), the system shows an error message. **Missing required information:** If a mandatory field (e.g., ground name) is left blank, the system prevents saving and highlights the required fields.
Special Requirements	Image uploads must be optimized for web display. The location should be linkable to a map service.

4.3.3. Full Dress Use Case for Futsal Assistant

Table.4.3. Use case for Futsal Assistant

Use Case Selection	Comment
Use Case Name	Use Futsal Assistant
Scope	Futsal Field Reservation System
Level	Subfunction
Primary Actor	Team, Ground Owner
Stakeholders and Interest	**User (Team/Ground Owner):** Wants quick answers to common questions without having to search the site or contact support. **System Owner:** Wants to reduce the support load and improve user satisfaction.
Pre-conditions	The user is on any page of the website where the chatbot icon is visible.
Success Guarantee	The user receives a relevant and helpful answer to their query or is guided to the correct section of the website.
Main Success Scenario	1. User clicks on the 'Futsal Assistant' chat icon. 2. The chatbot interface opens with a welcome message and suggested topics. 3. User types a question into the chat box (e.g., "How do I book a ground?"). 4. The Futsal Assistant processes the natural language query. 5. The Assistant provides a direct answer and may include a link to the relevant page (e.g., the bookings page). 6. The user's query is resolved.
Extensions	**Query not understood:** If the Assistant cannot understand the query, it responds with a message like, "I'm sorry, I don't understand. Could you rephrase, or would you like to see a list of

	common topics?" **Request for human support:** If the user's query is too complex or they explicitly ask to speak to a person, the Assistant provides contact information for human support.
Special Requirements	The chatbot must be able to process natural language. Responses should be fast (under 2 seconds).

4.4. Activity diagram:

An activity diagram is a behavioral diagram that graphically represents workflows of stepwise activities and actions. It is essentially a sophisticated flowchart that can be used for modeling the flow of control from one activity to another, investigating business requirements, and gaining a high-level understanding of the system's functionalities. Activity diagrams are particularly useful for illustrating complex processes that involve multiple steps, conditional logic, and parallel operations.

For this project, the activity diagram is drawn from the perspective of a Team user & Ground Owner performing the core function. This workflow is central to the system's purpose and involves several decisions and steps that are ideal for representation in an activity diagram.
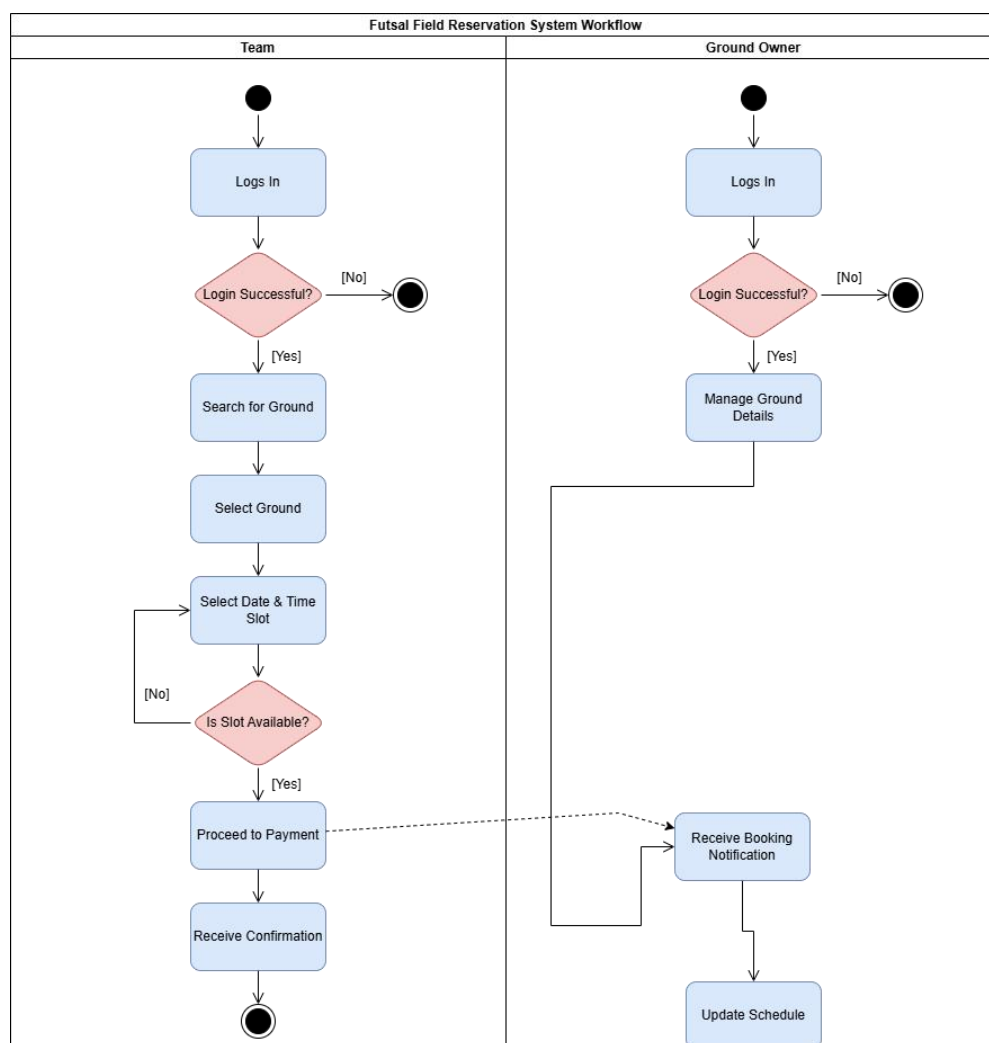

Figure 4.2. Activity Diagram

4.5. Data Flow Diagram:

A Data Flow Diagram (DFD) is a graphical technique that illustrates the flow of data through a system. It provides a clear picture of the system's inputs, outputs, the processes that transform the data, and the data stores where information is held. Unlike an activity diagram, a DFD has no control flow, decision rules, or loops; its sole purpose is to trace the path of data. Developing DFDs for the Futsal Field Reservation System is significant because the system is data-driven; it receives input from users (like registration details and booking requests), processes this data, and produces outputs (like confirmations and updated schedules). For this report, Level 0 and Level 1 DFDs are essential to model the system from a high-level overview down to a more detailed functional breakdown.

4.5.1. DFD Level 0

It represents the entire system as a single process and highlights its interaction with external entities. As shown in Figure 4.3, the Futsal Field Reservation System is at the center. It interacts with two primary external entities: the Team and the Ground Owner. The Team provides Booking Requests and Profile Info and receives Ground Details and Booking Confirmations. The Ground Owner provides Ground Info and Availability and receives Booking Notifications. Both entities can also provide Chat Messages and Assistant Queries and receive responses from the system.
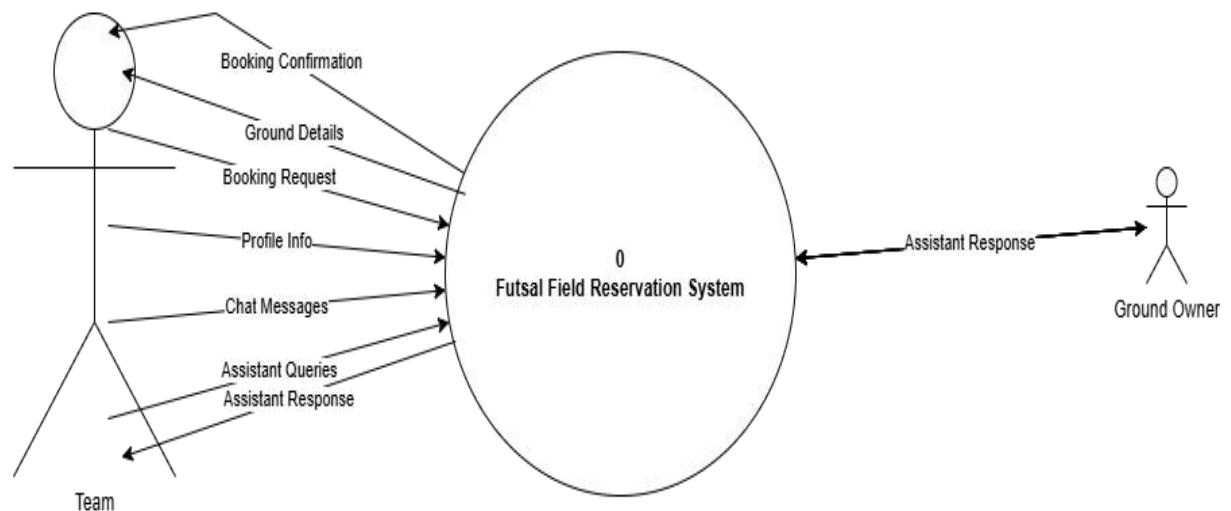

Figure 4.3. DFD Level 0

4.5.2. DFD Level 1

It shows how data flows between these internal processes and connects to data stores. As seen in Figure 4.4, the system is broken down into key processes like Manage User Profiles, Manage Grounds, Process Bookings, and Provide Assistance. The diagram illustrates, for example, that the Manage User Profiles process interacts with the Users data store, while Process Bookings accesses both the Grounds and Bookings data stores to function. This level clearly maps out the primary data transformations and storage points within the system.
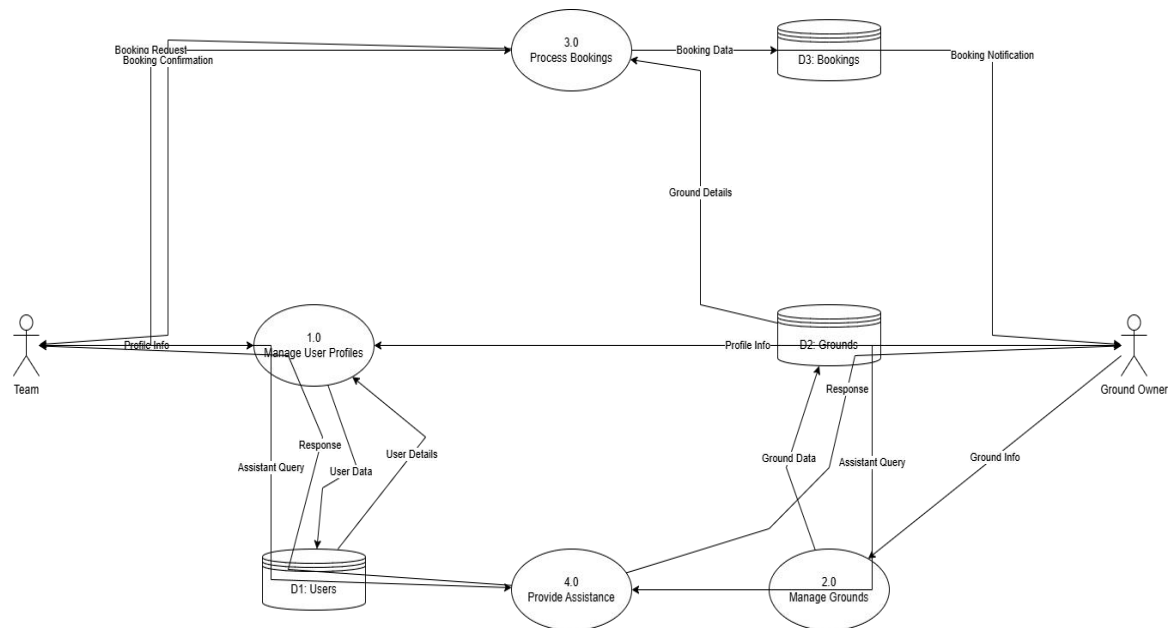
Figure 4.4. DFD Level 1

4.6. System Sequence Diagram

A System Sequence Diagram (SSD) is a crucial artifact in software design that illustrates the sequence of interactions between an external actor and the system for a specific scenario. It visualizes the input and output events as messages, treating the entire system as a single black box. The urge to draw an SSD for this project stems from the need to translate the narrative descriptions of the full dress use cases into a more formal, diagrammatic view. This helps developers understand the exact order of operations and the method calls required to realize a use case.

System Sequence Diagrams are derived directly from use cases and are essential for bridging the gap between requirements analysis and detailed design. For this project, we will develop an SSD for the Book Slot use case. This scenario is chosen because it represents the most significant and complex interaction a user has with the system, involving a clear sequence of requests and responses that are critical to the system's core purpose.
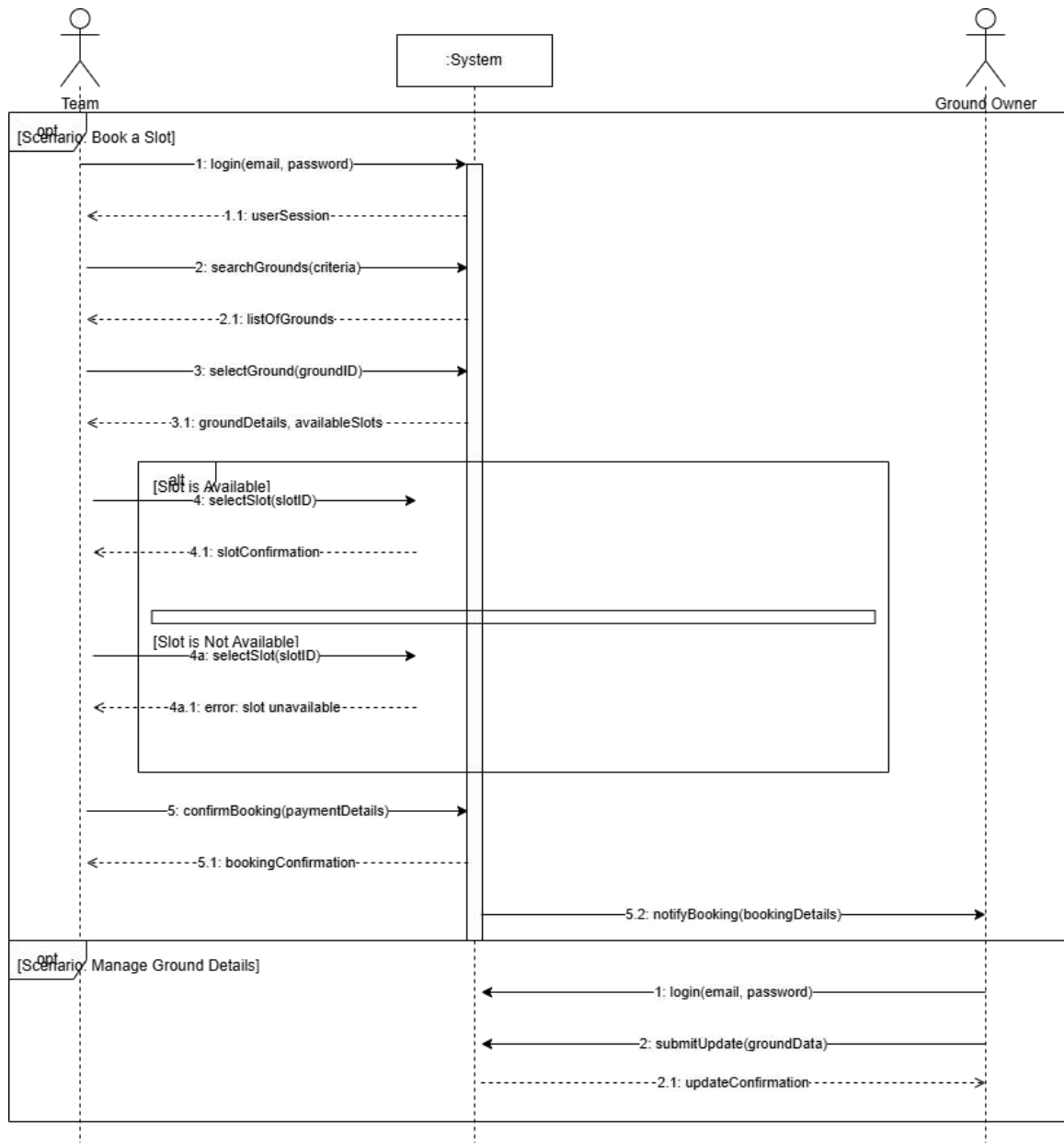
Figure 4.5. System Sequence Diagram

4.7. Sequence Diagram

A Sequence Diagram (SD) is a type of interaction diagram that details how operations are carried out by showing the time-ordered sequence of messages exchanged between different objects or components within a system. Unlike an SSD, which hides internal details, an SD provides a logical view of how a use case is realized by illustrating the collaboration between internal software components.

We will develop a Sequence Diagram for the most complex and critical event in the project: A Team successfully booking a ground. This scenario is chosen because it involves multiple internal components working together, from the user interface to the business logic controller and the database. The SD will show how these components pass messages to one another to search for a ground, check its availability, process the booking, save the data, and notify the relevant parties.
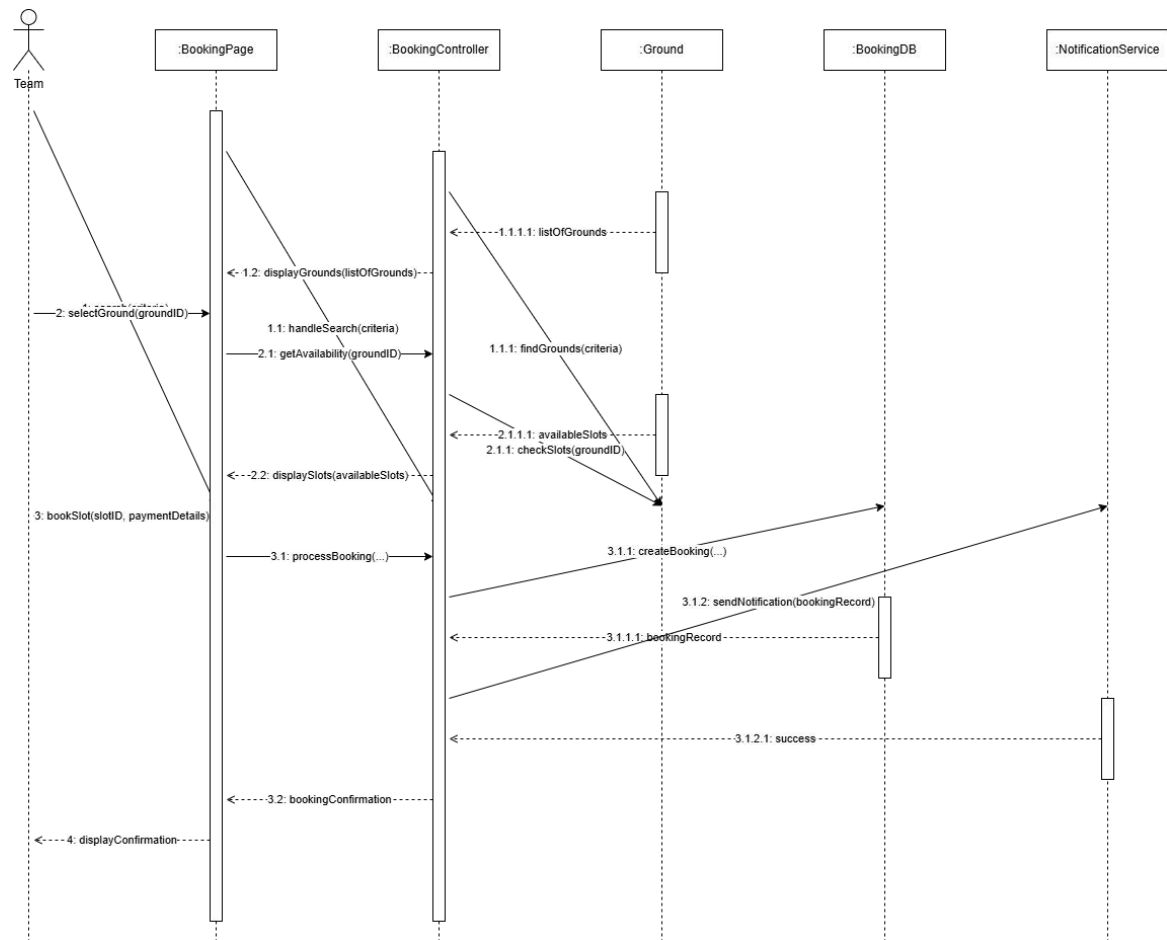
Figure 4.6. Sequence Diagram

4.8. Design Class Diagram

A Design Class Diagram (DCD) is a static, structural UML diagram that serves as the blueprint for a software system's architecture. It is a mandatory component of a Final Year Project report for any object-oriented system because it describes the structure by showing the system's classes, their attributes, their operations (or methods), and the relationships between them. The DCD provides a clear, high-level map of the codebase, helping developers to understand how different components are interconnected, which is essential for building a logical, scalable, and maintainable application.

The important classes for the Futsal Field Reservation System are identified based on the core entities of the project. The main classes include User, which acts as a base class, with Team and Ground Owner inheriting from it. The Ground class holds all information about a facility, and it is composed of multiple Time Slot objects. The central class that connects these entities is the Booking class, which is associated with a Team, a Ground, and a Payment. Finally, utility classes like Futsal Assistant and Notification Service are included to handle specific functionalities like the AI chat bot and sending alerts.

The Design Class Diagram for this project is shown in Figure 4.7. It illustrates the relationships between all these key classes. The inheritance hierarchy is clearly visible with Team and Ground Owner extending User. The diagram shows that a Ground Owner can own multiple Grounds, and a Team can make multiple Bookings. The Booking class acts as a central hub, linking a single Team to a single Ground for a specific reservation.

This diagram provides a complete and detailed overview of the system's static structure, which is invaluable for the development and future maintenance of the software.
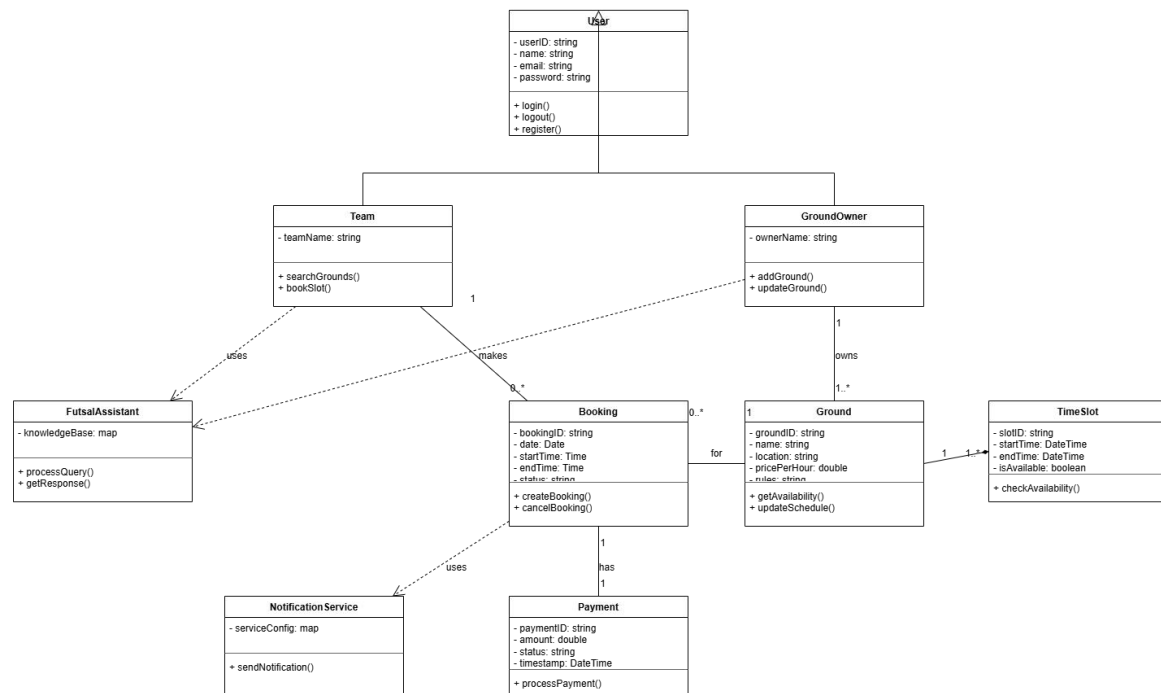

Figure 4.7. Design Class Diagram

4.9. Architectural diagrams

An architectural diagram serves as a high-level blueprint for a software system. Its primary purpose is to provide an abstraction that manages the system's complexity and establishes a clear communication and coordination mechanism among its various components. These diagrams are essential for ensuring that the system is built in a structured, scalable, and maintainable way. They are typically created when a diagrammatic illustration is necessary to convey the overall structure and strategy of the system's design.

4.9.1. Interface Design

The Interface Design diagram illustrates the high-level communication paths between the major logical layers of the system. It shows how the user's client (the browser) interfaces with the back end server, and how the server, in turn, interfaces with the various data and cloud services. This diagram is crucial for understanding the system's boundaries and the primary channels through which data flows.

The interface design shows the React Web App (running in the user's browser) making API calls to the Back end Server (Node.js/Express). The back end server then communicates with the MongoDB Database for core data storage (users, grounds, bookings) and with Firebase Services for specialized tasks like file storage (ground photos) and potentially real-time features. This separation ensures a clean and organized flow of requests and data.
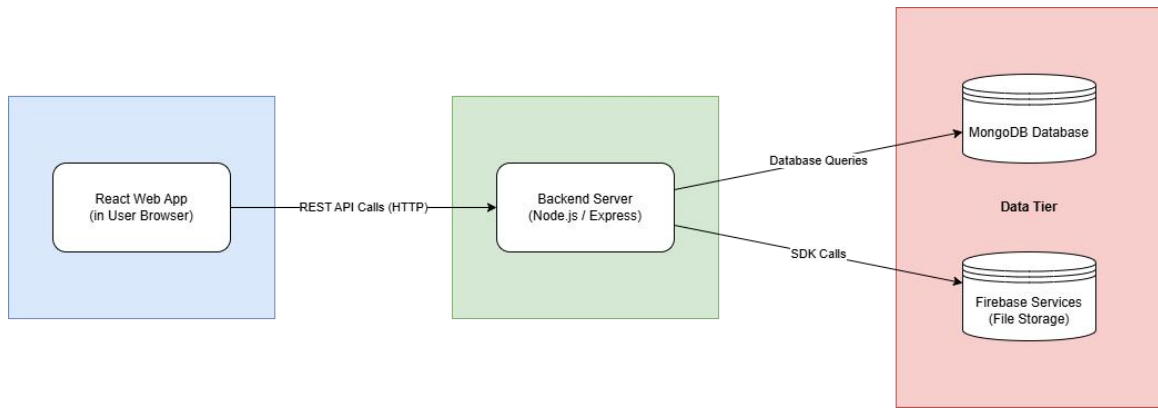
Figure 4.8. Interface Design

4.9.2. Component Level Design

A Component Diagram provides a "white-box" view of the system's logical architecture. It breaks down the application into its major functional components and illustrates the dependencies and interfaces between them. This diagram is necessary to show how the system is modularized. A well-componentized system is easier to understand, test, and maintain.

The diagram below shows the key components. The Web App is the main client-side component, which itself contains smaller UI components like Booking UI and Profile UI. This client communicates with the Futsal API provided by the server. The server is composed of a Web Server that routes requests to various service components like Booking Service, User Service, and Futsal Assistant, each responsible for a specific domain of business logic. These services then depend on a Database Interface to interact with the data stores.
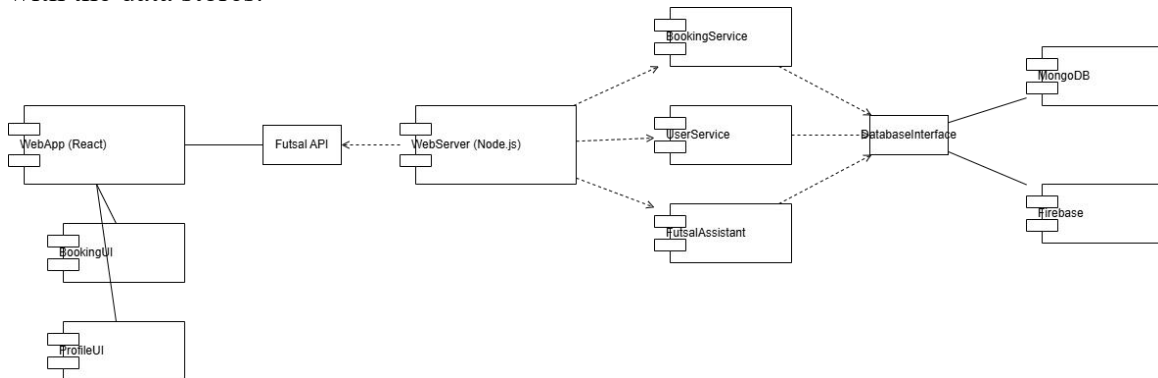

Figure 4.9. Component Level Design

4.9.3. Deployment

A Deployment Diagram illustrates the physical deployment of the system's software artifacts on hardware nodes. It provides a static view of the runtime configuration of a system, showing how software components are distributed across the physical infrastructure. This diagram is necessary to plan for hardware requirements, network configurations, and the physical distribution of the application.

The deployment diagram for this project shows three main nodes. The User's Device node runs a web browser, which hosts the React Web App artifact. This communicates over the internet with a Cloud Web Server (e.g., a virtual machine on AWS or Heroku). This server node hosts the Node.js Application artifact. The web server, in turn, connects to the MongoDB Atlas Cluster and Firebase Cloud Services, which are external, managed service nodes that host the databases and file storage respectively.

CHAPTER 5
SYSTEM TESTING AND VALIDATION

This chapter focuses on the critical phase of system testing and validation, which is performed to ensure that the Futsal Field Reservation System meets its specified requirements and is free of defects. The primary goal of this stage is to verify and validate the software, confirming that it functions as expected and provides a reliable, high-quality experience for its users. This involves executing a series of planned tests designed to uncover errors in logic, functionality, and the user interface, thereby ensuring the system's overall integrity and correctness before deployment.

This chapter is organized into several sections to provide a comprehensive overview of the testing process. It will begin with a general discussion of the Test Design, outlining the overall strategy and approach. This will be followed by detailed Test Cases for each of the core functionalities of the system. The chapter will then delve into specific testing methodologies, presenting the results from White-Box Testing, which examines the internal code structure, and Black-Box Testing, which focuses on functional requirements from a user's perspective without knowledge of the internal workings. Finally, it will cover Graphical User Interface (GUI) Testing to ensure that the visual elements of the application are functional, user-friendly, and aligned with the design specifications.

5.1. System testing

System testing is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements. Testing a software is critically important because it verifies that the product meets the business and technical requirements that guided its design and development. It is the primary process for ensuring quality, reliability, and performance, helping to identify and fix defects before the software is released to end-users, which ultimately saves costs and protects the developer's reputation. For this project, a combination of testing techniques was applied. White-Box Testing was used to inspect the internal code structure and logic to ensure the internal operations were performing correctly. Black-Box Testing was conducted from a user's perspective to verify that the system's functionalities, such as registration, booking, and profile management, met the specified requirements without any knowledge of the internal implementation. Finally, GUI Testing was performed to ensure that all visual elements of the web application were correctly displayed, functional, and provided a user-friendly experience.

The overall analysis of the testing results indicates that the Futsal Field Reservation System successfully meets its core objectives. The successful "Pass" status across all major test cases for booking, user registration, and profile management confirms that the primary goal of providing local football players with a streamlined and professional platform for reserving grounds has been achieved. The tests validated that the system effectively minimizes the booking hurdles previously faced by players. Similarly, the successful tests for the ground owner's portal demonstrate that the objective of helping owners turn ground renting into a more lucrative and manageable business has been met. The positive outcomes of the GUI testing further confirm that the system provides a user-friendly and intuitive interface, which aligns with the overall objective of creating an efficient and easy-to-use centralized platform for both teams and facility providers.

5.2. Testing Techniques

Software testing involves various techniques to evaluate a system, each offering a different perspective on the application's quality and correctness. These techniques provide a systematic approach to finding defects and verifying that the software meets its requirements. For the Futsal Field Reservation System, a combination of testing

methodologies was employed to ensure comprehensive coverage of both the internal logic and the external functionality. This section details the two primary testing techniques used: White-Box Testing and Black-Box Testing.

5.2.1. White box Testing

White-Box Testing, also known as clear-box, glass-box, or structural testing, is a software testing method in which the internal structure, design, and coding of the software are tested. The tester has access to the source code and uses their knowledge of the implementation to design test cases. This technique focuses on verifying the flow of inputs and outputs through the application, improving design and usability, and strengthening security by examining the code itself.

The rationale for using White-Box Testing in this project was to ensure the internal integrity and logical correctness of the system's core backend functions. Since processes like user registration, payment processing for donations, and slot booking involve critical data manipulation and business logic, it was essential to verify that the underlying code was sound. This technique allowed for the examination of specific conditional paths, loops, and data handling within the code to confirm that they behaved as expected under various conditions, which is not possible with black-box methods alone.

The tests were carried out by conducting thorough code reviews of the key modules and functions within the Node.js backend. Developers executed specific functions with predefined inputs to trace their execution paths and validate the outputs. For example, in testing the user registration function (as documented in WBTC #:W1), the code responsible for hashing passwords and inserting user data into the MongoDB database was directly inspected and executed to ensure it performed these operations securely and correctly. Similarly, other critical code segments for booking and payment were tested to ensure they handled data transactions properly.

The results of the White-Box Testing were highly successful. As indicated by the "Pass" status for all white-box test cases in the project documentation, the internal logic of the application's most critical components is robust and free from logical errors. This confirms that the data flows correctly through the system, calculations are accurate, and the security measures, such as password handling, are implemented as intended.

5.2.2. Black Box Testing

Black-Box Testing is a software testing method in which the functionality of an application is tested without any knowledge of its internal code structure, implementation details, or internal paths. This technique focuses solely on the inputs and outputs of the software, treating the system as a "black box." The primary goal is to verify that the system meets the functional requirements and behaves as expected from an end-user's perspective.

This testing technique was fundamental to the project because it directly validates the user experience and ensures that the system fulfills its intended purpose for its target audience (Teams and Ground Owners). The rationale for its extensive use was to simulate real-world user interactions and confirm that all features are intuitive, functional, and deliver the expected results. It ensures that a user can seamlessly navigate the website, register an account, search for a ground, and complete a booking without encountering functional errors, regardless of the underlying complexity of the code.

Black-Box testing was carried out by testers who interacted with the live web application through its graphical user interface. Following the detailed test cases outlined in the documentation (e.g., TC1 through TC7), testers performed actions that a typical user would. This included filling out registration forms with both valid and invalid data to check validation, logging in, using the search functionality, navigating the booking calendar, and completing the booking process. Each step was performed to verify that the system's response matched the expected output defined in the test case.

The results of the Black-Box Testing were overwhelmingly positive, with all test cases receiving a "Pass" status. This demonstrates that the Futsal Field Reservation System is functionally complete and correct from the end-user's viewpoint. It confirms that users can successfully accomplish their goals on the platform, from creating an account to booking a pitch, and that the system correctly handles all user inputs as per the specified requirements. This success is a strong indicator that the software is ready for user acceptance and deployment.

5.3. Test Cases

Test cases are a set of actions executed to verify a particular feature or functionality of a software application. They are a fundamental component of the testing process, providing a structured and systematic way to ensure that the system behaves as expected and meets its prioritized requirements. It is not necessary to test every single minor functionality; instead, test cases are designed to cover the most critical and high-risk features of the system.

Test Case 1: User Registration

Table 5.1: Test Case 1

GENERAL INFORMATION			
Test Stage:	☐Unit ☑Functionality ☐Integration ☐System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot		
Test Date:		**System Date, if applicable:**	
Tester:	Saadullah	**Test Case Number:**	TC-01
Test Case Description:	To verify that a new user (Team or Ground Owner) can successfully create an account on the platform.		
Results:	☐Pass ☐Fail	**Incident Number, if applicable:**	
INTRODUCTION			
Requirement(s) to be tested:	01: The System shall allow users to set up their respective accounts. 02: The System shall allow the user to set email for sign up.		

	FR003: The System shall allow the user to set password for sign up.
Roles and Responsibilities:	The Tester is responsible for executing the test steps and recording the outcome.
Set Up Procedures:	1. Open the Futsal Field Reservation System website in a compatible browser. 2. Ensure no user is currently logged in. 3. Navigate to the "Sign Up" page.
Stop Procedures:	Close the browser tab after the test is complete.
ENVIRONMENTAL NEEDS	
Hardware:	Standard desktop or laptop computer with an internet connection.
Software:	Google Chrome web browser.
Procedural Requirements:	None.
TEST	
Test Items and Features:	User Registration feature, including input fields for name, email, password, and user type selection.
Input Specifications:	Valid and unique user email, a strong password, and other required profile information.
Procedural Steps:	1. Fill in all required fields in the registration form. 2. Click the "Submit" button. 3. Observe the system's response.
Expected Results of Case:	The system should validate the inputs, create a new user account in the database, and redirect the user to the login page or their new dashboard, displaying a success message.
ACTUAL RESULTS	
Output Specifications:	The system successfully created the new user account. The user was redirected to their dashboard, and a confirmation was implicitly received. The database correctly reflected the new user entry. The test passed.

Test Case 2: Accessing Ground Owner Profile

Table 5.2: Test Case 2

GENERAL INFORMATION			
Test Stage:	☐Unit ☑Functionality ☐Integration ☑System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot		
Test Date:	mm/dd/yy	**System Date, if applicable:**	mm/dd/yy
Tester:	Saadullah	**Test Case Number:**	TC-02
Test Case Description:	To verify that a registered Ground Owner can securely log in and access their specific profile dashboard.		
Results:	☑Pass ☐Fail	**Incident Number, if applicable:**	N/A
INTRODUCTION			
Requirement(s) to be tested:	1: The System shall allow the user to login with correct credentials. 2: The System shall allow the ground owners to provide all theirs and ground details.		
Roles and Responsibilities:	The Tester is responsible for executing the test and verifying access to the correct dashboard.		
Set Up Procedures:	1. Ensure a Ground Owner account exists in the system. 2. Navigate to the login page.		
Stop Procedures:	Log out of the account and close the browser.		
ENVIRONMENTAL NEEDS			
Hardware:	Standard desktop or laptop computer with an internet connection.		
Software:	Google Chrome web browser.		
Procedural Requirements:	Valid login credentials for a Ground Owner are required.		
TEST			

Test Items and Features:	Login functionality, user authentication, and role-based redirection.
Input Specifications:	A registered Ground Owner's email and password.
Procedural Steps:	1. Enter the Ground Owner's email and password into the login form. 2. Click the "Login" button. 3. Verify that the page redirects to the Ground Owner's dashboard.
Expected Results of Case:	The system should authenticate the user and grant access to the Ground Owner dashboard, where they can manage their facilities. Access to other user-type dashboards (e.g., Team) should be denied.
ACTUAL RESULTS	
Output Specifications:	The user was successfully authenticated and redirected to the correct Ground Owner dashboard. All profile management features were visible and accessible. The test passed.

Test Case 3: Booking a Ground Slot

Table 5.3: Test Case 3

GENERAL INFORMATION			
Test Stage:	☐Unit ☑Functionality ☐Integration ☑System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot Specify the testing stage for this test case.			
Test Date:	mm/dd/yy	System Date, if applicable:	mm/dd/yy
Tester:	Saadullah	Test Case Number:	TC-03
Test Case Description:	To verify the complete, end-to-end workflow of a Team user successfully booking a ground slot.		
Results:	☑Pass ☐Fail	Incident Number, if applicable:	N/A
INTRODUCTION			

Requirement(s) to be tested:	1: The System will provide a view of grounds for booking. 2: The System will allow teams to view grounds and click on whichever they want to book.. 3: Teams can book a slot if available at their desired time.
Roles and Responsibilities:	The Tester is responsible for simulating the Team user's actions and verifying the booking confirmation.
Set Up Procedures:	1. Log in as a registered Team user. 2. Ensure there is at least one ground listed by a Ground Owner with available slots. 3. Navigate to the "Bookings" or "Search Grounds" page.
Stop Procedures:	Log out of the account. The booking can be manually cancelled in the database if needed.
ENVIRONMENTAL NEEDS	
Hardware:	Standard desktop or laptop computer with an internet connection.
Software:	Google Chrome web browser.
Procedural Requirements:	A logged-in Team user session.
TEST	
Test Items and Features:	Ground search, ground details view, availability calendar/slot selection, and booking confirmation process.
Input Specifications:	Selection of a ground, a date, and an available time slot.
Procedural Steps:	1. Select a ground from the list. 2. On the ground's detail page, select an available date and time slot. 3. Click the "Book Now" button. 4. Proceed through any confirmation steps.
Expected Results of Case:	The system should reserve the selected slot, make it unavailable for other users, record the booking in the database, and display a success confirmation to the Team user.
ACTUAL RESULTS	
Output Specifications:	The slot was successfully booked. A confirmation message was displayed, and the slot was no longer shown as available on the ground's schedule. The test passed.

Test Case 4: Accessing Teams Profile

Table 5.4: Test Case 4

GENERAL INFORMATION			
Test Stage:	☐Unit ☑Functionality ☐Integration ☑System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot Specify the testing stage for this test case.		
Test Date:	mm/dd/yy	**System Date, if applicable:**	mm/dd/yy
Tester:	Ehtisham Naveed	**Test Case Number:**	TC-04
Test Case Description:	To verify that a registered Team can securely log in and access their specific profile dashboard.		
Results:	☑Pass ☐Fail	**Incident Number, if applicable:**	N/A
INTRODUCTION			
Requirement(s) to be tested:	1: The System shall allow the user to login with correct credentials. 2: The System shall provide teams with dedicated profiles. 3: The System shall allow the teams to update their profiles.		
Roles and Responsibilities:	The Tester is responsible for executing the test and verifying access to the correct dashboard.		
Set Up Procedures:	1. Ensure a Team account exists in the system. 2. Navigate to the login page.		
Stop Procedures:	Log out of the account and close the browser.		
ENVIRONMENTAL NEEDS			
Hardware:	Standard desktop or laptop computer with an internet connection.		
Software:	Google Chrome web browser.		
Procedural Requirements:	Valid login credentials for a Team are required.		
TEST			

Test Items and Features:	Login functionality, user authentication, and role-based redirection to the Team dashboard.
Input Specifications:	A registered Team's email and password.
Procedural Steps:	1. Enter the Team's email and password into the login form. 2. Click the "Login" button. 3. Verify that the page redirects to the Team's dashboard.
Expected Results of Case:	The system should authenticate the user and grant access to the Team dashboard, where they can manage their profile and view bookings.
ACTUAL RESULTS	
Output Specifications:	The user was successfully authenticated and redirected to the correct Team dashboard. All profile management features were visible and accessible. The test passed.

Test Case 5: Making a Donation

Table 5.5: Test Case 5

	GENERAL INFORMATION		
Test Stage:	☐Unit ☑Functionality ☐Integration ☑System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot Specify the testing stage for this test case.		
Test Date:	mm/dd/yy	**System Date, if applicable:**	mm/dd/yy
Tester:	Ehtisham Naveed	**Test Case Number:**	TC-05
Test Case Description:	To verify that a user can successfully make a monetary contribution through the donation feature.		
Results:	☑Pass ☐Fail	**Incident Number, if applicable:**	N/A
	INTRODUCTION		
Requirement(s) to be tested:	1: The System will provide a donation feature. 2: The System will allow users to make the amount of payment they		

	desire.
Roles and Responsibilities:	The Tester is responsible for navigating to the donation page and completing a test transaction.
Set Up Procedures:	1. Navigate to the website's "Donation" page. 2. Ensure test payment credentials are available if using a sandbox environment.
Stop Procedures:	Close the browser tab.
ENVIRONMENTAL NEEDS	
Hardware:	Standard desktop or laptop computer with an internet connection.
Software:	Google Chrome web browser.
Procedural Requirements:	None.
TEST	
Test Items and Features:	Donation page UI, payment form integration (e.g., Stripe), and transaction processing.
Input Specifications:	A monetary amount and valid (test) payment details.
Procedural Steps:	1. Click the "Donate" button on the donation page. 2. Enter a donation amount and payment information into the form. 3. Click the "Pay" or "Submit" button.
Expected Results of Case:	The system should securely process the payment and display a "Thank You" or success confirmation message.
ACTUAL RESULTS	
Output Specifications:	The test transaction was processed successfully, and a confirmation message was displayed to the user. The test passed.

Test Case 6: Chat-Box

Table 5.6: Test Case 6

GENERAL INFORMATION			
Test Stage:	☐Unit ☑Functionality ☑Integration ☑System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot Specify the testing stage for this test case.		
Test Date:	mm/dd/yy	**System Date, if applicable:**	mm/dd/yy
Tester:	Ehtisham Naveed	**Test Case Number:**	TC-06
Test Case Description:	To verify that users can communicate with each other in real-time using the integrated chat feature.		
Results:	☑Pass ☐Fail	**Incident Number, if applicable:**	N/A
INTRODUCTION			
Requirement(s) to be tested:	1: The System shall allow the teams to communicate with other teams. 2: The System will allow communication between users.		
Roles and Responsibilities:	Two testers are required, one for each user account, to test the real-time message exchange.		
Set Up Procedures:	1. Log in with two different user accounts (e.g., Team A and Ground Owner B) in two separate browser windows. 2. Navigate to a page where the chat functionality is accessible.		
Stop Procedures:	Log out of both accounts.		
ENVIRONMENTAL NEEDS			
Hardware:	Two computers or two separate browser sessions on one computer.		
Software:	Google Chrome web browser.		
Procedural Requirements:	Two active, logged-in user sessions.		
TEST			

Test Items and Features:	Chat interface, message input field, send button, and real-time message display.
Input Specifications:	Text messages from both users.
Procedural Steps:	1. User A initiates a chat with User B. 2. User A types and sends a message. 3. Verify that User B receives the message instantly. 4. User B types and sends a reply. 5. Verify that User A receives the reply instantly.
Expected Results of Case:	For common questions, the chatbot should provide a direct, accurate answer or a link to the relevant page. For unrecognized queries, it should provide a polite message indicating it doesn't understand and offer alternative help options.
ACTUAL RESULTS	
Output Specifications:	Messages should be sent and received between the two users in real-time without significant delay. The chat history should be correctly displayed for both users.

Test Case 7: Using the Futsal Assistant

Table 5.7: Test Case 7

GENERAL INFORMATION		
Test Stage:	☐Unit ☑Functionality ☑Integration ☑System ☐Interface ☐Performance ☐Regression ☐Acceptance ☐Pilot Specify the testing stage for this test case.		
Test Date:	mm/dd/yy	**System Date, if applicable:**	mm/dd/yy
Tester:	Ehtisham Naveed	**Test Case Number:**	TC-07
Test Case Description:	To verify that a user can interact with the Futsal Assistant chatbot and receive relevant, helpful responses.		
Results:	☑Pass ☐Fail	**Incident Number, if applicable:**	N/A

INTRODUCTION	INTRODUCTION
Requirement(s) to be tested:	The system shall provide an AI chatbot ("Futsal Assistant") to answer user queries..
Roles and Responsibilities:	The Tester is responsible for interacting with the chatbot and evaluating the relevance and accuracy of its responses.
Set Up Procedures:	1. Navigate to any page on the website where the Futsal Assistant icon is present.
Stop Procedures:	Close the chat interface.
ENVIRONMENTAL NEEDS	ENVIRONMENTAL NEEDS
Hardware:	Standard desktop or laptop computer with an internet connection.
Software:	Google Chrome web browser.
Procedural Requirements:	None.
TEST	TEST
Test Items and Features:	Chatbot UI, natural language processing, and response generation.
Input Specifications:	A set of common user queries, such as "How to book a ground?", "What are the prices?", and "I forgot my password."
Procedural Steps:	1. Click the Futsal Assistant icon to open the chat window. 2. Type a common question into the input field and press Enter. 3. Observe the chatbot's response. 4. Type a more complex or nonsensical query to test its fallback response.
Expected Results of Case:	For common questions, the chatbot should provide a direct, accurate answer or a link to the relevant page. For unrecognized queries, it should provide a polite message indicating it doesn't understand and offer alternative help options.
ACTUAL RESULTS	ACTUAL RESULTS
Output Specifications:	The chatbot correctly answered all predefined common questions. When presented with an unknown query, it responded with a helpful fallback message. The test passed.

5.4. Non-Functional Requirements

Non-functional requirements (NFRs) are a critical aspect of system specification that define the quality attributes of a software system. Unlike functional requirements, which describe what the system does, NFRs describe how the system performs a certain function. They are constraints on the system's behavior and are essential for ensuring a good user experience, system reliability, and maintainability. It is crucial to define NFRs using objective and measurable metrics rather than subjective statements like "the software is fast," to allow for proper verification and validation. This section details the key non-functional requirements for the Futsal Field Reservation System. Each requirement is defined and evaluated against specific, measurable criteria to provide a clear and objective assessment of the system's quality. The purpose of the table provided below is to list common NFRs and their corresponding measurement criteria, which will be used as a basis for evaluating this project.

5.4.1. Compatibility

Compatibility refers to the ability of the software to run on specific hardware, operating systems, applications, or network environments. For a web-based application, browser compatibility is a primary concern.

Property	Measure	Target
Compatibility	Number of target systems (browsers).	The system must be fully functional on the latest versions of Google Chrome, Mozilla Firefox, and Microsoft Edge.

Table 5.8. Compatibility

5.4.2. Usabality

Usability, or ease of use, measures how easy it is for a user to learn, operate, and get satisfactory results from the system. It is a critical factor for user adoption and satisfaction.

Property	Measure	Target
Ease of use	Training time	A new user should be able to successfully register and book a ground in under 5 minutes without any external training or help documentation.
Ease of use	User/event response time	All pages and primary interactive elements (e.g., booking calendar) must load and respond to user input in under 3 seconds.

Table 5.9. Usability

5.4.3. Safety and Security

Safety and Security requirements are constraints that protect the system and its data from accidental or malicious harm. This includes protecting user data and ensuring that only authorized users can access specific functionalities.

Property	Measure	Target
Security	Password-based authentication	All user accounts must be protected by passwords that are hashed before being stored in the database.
Security	Access control	Only authenticated users can access their respective dashboards. A user with a 'Team' role cannot access the 'Ground Owner' dashboard, and vice versa.
Robustness	Probability of data corruption on failure	The system must ensure that in the event of a failed booking transaction (e.g., payment failure), no partial or corrupt booking data is saved to the database. The probability should be less than 0.1%.

Table 5.10. Safety and Security

CHAPTER 6
CONCLUSION

This final chapter provides a summary of the entire project, reflecting on its achievements and outlining potential avenues for future development. The chapter is organized into two main sections. The first section offers concluding remarks, summarizing the project's success in meeting its initial objectives and the contributions it makes. The second section discusses future work, identifying the limitations of the current system and suggesting enhancements that could be implemented in subsequent versions to expand its scope and improve its functionality.

The Futsal Field Reservation System was developed to address a significant gap in the local sports community: the lack of a centralized, efficient platform for booking futsal grounds. The project successfully created a web-based service that streamlines the entire process for both football teams and ground owners. By providing a single platform where owners can list their facilities and teams can easily view availability and make reservations, the system effectively eliminates the time-consuming and frustrating hurdles that previously defined this process. The successful implementation and testing of core features—such as user registration, profile management, a real-time booking calendar, and secure login—confirm that the primary objectives have been met. The system not only simplifies ground booking but also fosters a more connected community through integrated communication features and supports with its donation functionality. Ultimately, the project delivers a practical and valuable solution that enhances the futsal experience in the twin cities.

While the current system provides a robust and functional platform, there are several avenues for future enhancement that could significantly increase its impact and utility. The current scope is limited to futsal grounds within the twin cities; a major future goal would be to expand the system nationwide, incorporating facilities from other cities and potentially including other sports to cater to a wider audience. Another critical enhancement would be the development of a dedicated mobile application for both Android and iOS platforms. This would offer users greater convenience and on-the-go access to the system's features. Furthermore, future versions could focus on improving system efficiency and accuracy, potentially by integrating more advanced analytic for ground owners and refining the user interface based on long-term user feedback. Finally, the inclusion of national football leagues and more structured tournament management features could elevate the platform from a simple booking tool to a comprehensive hub for the national football community.

6.1. Conclusion

The Futsal Field Reservation System was conceived and developed to resolve the disconnected and inefficient process of booking futsal grounds in the twin cities. This project successfully delivers a centralized, web-based platform that directly connects football teams with facility owners, fundamentally streamlining the entire reservation workflow. Through the synthesis of key features—including dedicated user portals, a dynamic ground search and booking engine, and an integrated communication system capped by an intelligent Futsal Assistant—the project has achieved its primary goal. The successful testing and validation of these functionalities serve as clear evidence that the system provides a robust and reliable solution, effectively eliminating the booking hurdles that previously hindered the local football community and creating a more accessible and organized sporting environment.

6.2. Limitations and Future work

While the Futsal Field Reservation System successfully meets its primary objectives, its current implementation has several inherent limitations that provide a clear road map for future enhancements. The most significant limitation is its geographical and sport-specific scope; the service is currently restricted to futsal grounds within the Rawalpindi and Islamabad twin cities. This naturally excludes a large number of potential users across the country and limits the system's overall market reach. Furthermore, the system is exclusively a web-based application, which, while accessible, lacks the convenience and rich user experience provided by native mobile applications.

Based on these findings, several key actions are recommended for future work to overcome these limitations. Firstly, it is strongly recommended that the platform be expanded to a national level. This would involve a phased roll out to include major cities across Pakistan, requiring a scalable infrastructure and partnerships with facility owners in new regions. Secondly, to broaden the user base, it is recommended that the system be enhanced to include other popular sports, such as cricket, badminton, and basketball, transforming it into a comprehensive multi-sport booking platform. Finally, the development of dedicated mobile applications for both Android and iOS is a critical recommendation. A mobile app would not only improve accessibility but also enable powerful features like push notifications for booking reminders and real-time match alerts, significantly enhancing user engagement and solidifying the system's position as an indispensable tool for the local sports community.

APPENDIX-I
User Manual: Futsal Field Reservation System

1. Introduction

This manual provides a comprehensive guide for users of the Futsal Field Reservation System. It covers all the necessary steps for registering an account, logging in, and utilizing the platform's core features. The manual is divided into sections based on user type: Team and Ground Owner, and also includes instructions for using the Futsal Assistant.

2. Getting Started: Account Registration

To access the system, all users must first create an account.

1. Navigate to the Website: Open your web browser and go to the homepage of the Futsal Field Reservation System.
2. Select User Type: Click on the "Sign Up" button. You will be prompted to register as either a 'Team' or a 'Ground Owner'.
3. Fill in the Details:
- For Teams: Provide your Team Name, Captain's Name, Email, Password, Mobile Number, and Address.
- For Ground Owners: Provide the Ground's Name, Owner's Name, Email, Password, Mobile Number, and the Ground's Address.
4. Submit the Form: After filling in all the required information, click the "Submit" button. Your account will be created, and you will be redirected to the login page.

3. Logging In

Once you have a registered account, you can log in to access your dashboard.

1.Click the "Login" button on the homepage.
2. Enter the email and password you used during registration.
3. Click "Login". You will be redirected to your personalized dashboard based on your user type.

4. For Team Users

As a Team user, you can search for grounds, book slots, and manage your profile.

4.1 Searching for and Booking a Ground

1. Navigate to Bookings: From your dashboard or the main navigation bar, click on "Bookings" or "Grounds".

2. View Grounds: A list of all available futsal grounds will be displayed. You can use the search bar to find a specific ground by name.

3. Select a Ground: Click on the ground you are interested in to view its details, including location, price, and photos.

4. Check Availability: The ground's detail page will show a calendar with available time slots.

5. Book a Slot: Click on a desired date and time slot, then click the "Book Now" button.

6. Confirm Booking: Follow the on-screen instructions to confirm your booking. This may involve a payment step. Once completed, you will receive a booking confirmation.

5. For Ground Owner Users

As a Ground Owner, you can manage your facility's listing, view bookings, and update your schedule.

5.1 Managing Your Ground Details

1. Access Your Dashboard: After logging in, you will land on your Ground Owner dashboard.

2.Add/Edit Ground: You will have options to "Add a New Ground" or "Edit" an existing one.

3.Update Information: Fill in or update the form with your ground's name, location, price per hour, rules, and upload new photos.

4.Set Availability: Use the calendar tool to set or block out the time slots that are available for booking.

5.Save Changes: Click "Save" or "Update" to make your changes live on the platform.

5.2. Viewing Bookings

Your dashboard will display a list or calendar view of all upcoming and past bookings for your facility. You will receive a notification when a new booking is made.

6. Using the Futsal Assistant

The Futsal Assistant is an AI-powered chatbot available to all users for instant help.

1. Open the Assistant: Click on the chat icon, usually located in the bottom-right corner of the screen.

2. Ask a Question: The chat window will open. Type your question into the text box (e.g., "How do I cancel a booking?", "What are the payment methods?").

3. Receive an Answer: The assistant will process your question and provide an immediate response. If it cannot answer, it will provide guidance on how to contact human support.

REFERENCES

[1] Deploying A Free Cluster, [Online] Available: https://www.mongodb.com/docs/atlas/tutorial/deploy-free-tier-cluster/#deploy-a-free-cluster. [Accessed October 3, 2021]

[2] Uploading Images & Files To Firebase Storage In React JS | Node JS | MongoDB | Express, March 5, 2022, [Online] Available: https://dev.to/cyberwolve/uploading-images-files-to-firebase-storage-in-react-js-node-js-mongodb-express-ngi [Accessed December 9, 2021]

[3] Building a Real-Time Chat App with React and Firebase, 25 March, 2020, [Online] Available:https://css-tricks.com/building-a-real-time-chat-app-with-react-and-firebase/#top-of-site [Accessed January 12, 2022]

[4] Blocked by CORS policy: The 'Access-Control-Allow-Origin' - Mean Stack, January 3,2019,[Online]Available:https://www.digitalocean.com/community/questions/blocked-by-cors-policy-the-access-control-allow-origin-mean-stack . [Accessed February 3, 2022]