Vietname National University HCMC

International University

PROJECT REPORT

Sports Facility Booking Web Application

Member list:

Nguyễn Nhật Trường – ITITIU20137

Table of contents

Abstract

The development of the "EGGO" sports facility rental web application represents a significant advancement in enhancing community engagement through sports. This report delves into the design, development, and deployment of "EGGO," an intuitive platform that simplifies the process of booking sports facilities. Targeting two primary user groups—general users and hosts—the application provides tailored functionalities to each. General users can view facilities, make reservations, and add favorite venues for future reference, all facilitated by a user-friendly interface. Hosts, on the other hand, can list new sports facilities and manage existing listings, ensuring they have full control over their offerings.

The system architecture incorporates modern web technologies, including React for the frontend and MongoDB with Prisma for backend operations. Authentication is robustly handled via NextAuth, ensuring secure access and data integrity. The application also features comprehensive testing phases, ensuring reliability and performance stability before deployment on a cloud platform.

This report outlines the project's phases, technologies used, and the systematic approach towards a scalable and secure application, aiming to foster a vibrant community around sports and recreational activities.

CHAPTER 1 INTRODUCTION

1.1 Background

In recent years, the integration of digital solutions into everyday activities has become ubiquitous, significantly altering how services are accessed and delivered. The sports industry, particularly in the context of community engagement and facility management, is no exception. Traditional methods of booking and managing sports facilities often involve cumbersome processes, lacking in flexibility and accessibility. The advent of web applications dedicated to service facilitation presents an opportunity to streamline these processes, enhancing user experience and operational efficiency.

"EGGO" is a web application designed to bridge the gap between sports facility owners (hosts) and individuals seeking to rent these spaces (users). By providing a centralized platform, EGGO aims to simplify the interaction between these parties, making sports more accessible to the community and providing facility owners with a tool to effectively manage and promote their offerings.

1.2 Problem statement

The process of renting sports facilities is often fraught with challenges, including inefficient booking systems, lack of real-time availability updates, and the absence of a centralized platform for comparing different facilities. These issues deter participation and complicate the management for facility owners. Additionally, potential renters face difficulties in finding suitable locations for their activities, leading to decreased physical activity and community engagement.

1.3 Objectives and Scope

1.3.1 Objectives

The primary objective of the EGGO web application is to create an intuitive, accessible, and efficient platform that:

- Facilitates easy and quick booking of sports facilities.

- Provides comprehensive details about facilities, including real-time availability, to aid decision-making.

- Enhances the visibility of sports facilities to potential renters, thereby increasing booking rates and revenue for facility owners.

- Improves community engagement by simplifying access to sports facilities.

1.3.2 Scope

- User Interface Development: Designing a user-friendly interface that caters to both general users and facility hosts, incorporating functionalities such as facility viewing, booking, and management.
- Backend Development: Establishing a robust backend architecture to handle data management, real-time booking updates, and user authentication.
- Security Measures: Implementing comprehensive security protocols to protect user data and ensure secure transactions.
- Testing and Deployment: Conducting extensive testing to ensure the application's reliability and deploying it on a scalable cloud platform to handle potential growth in user numbers.

The application targets a diverse audience, including casual athletes, sports enthusiasts, professional trainers, and facility owners, aiming to foster a vibrant, active community.

CHAPTER 2 REQUIREMENTS

2.1 Stakeholders

The success of the "EGGO" sports facility rental web application hinges on the active participation and support of several key stakeholders. Each group has specific needs and expectations that the application must meet to ensure widespread adoption and sustainability. Understanding these stakeholders is crucial for tailoring features and functionalities that address their concerns and requirements.

2.1.1 General Users

Description: General users include individual athletes, sports enthusiasts, and team organizers who seek to find and book sports facilities for personal or group activities.
Needs:
A comprehensive search tool to find facilities based on location, availability, type of sport, and other criteria.
A streamlined booking process with real-time availability updates.
Access to detailed information about facilities, including amenities, pricing, and user reviews.

2.1.2 Facility Hosts

Description: Facility hosts are owners or managers of sports facilities who list their spaces on the platform to attract bookings.
Needs:

An intuitive interface to list and manage their facilities, including updating availability, pricing, and special offers.
Analytics tools to monitor bookings, financial transactions, and customer feedback to optimize their services.
Marketing features to enhance the visibility of their facilities to potential users.

2.1.3 System Administrators

Description: System administrators are responsible for overseeing the platform's overall functioning, including managing user activity, content moderation, and technical support.
Needs:
Powerful administrative tools to manage user accounts, resolve disputes, and ensure compliance with platform policies.
Robust security features to protect against data breaches and maintain user trust.
Scalable infrastructure management capabilities to ensure smooth operation during peak loads.

2.2 Functional and Non-functional requirements

Table 2.1: Functional and Non-functional Requirements

Req.ID	Requirement Name	Detailed Description	Type
01	Register/Login/Logout	Customers may establish their own accounts to record personal usage to book or rent fields.	Functional requirement
02	Facility Searching and Viewing	The application shall provide detailed views of each facility, including photos, descriptions, amenities, pricing, and user reviews	Functional requirement
03	Booking and Reservation Management	The application shall enable users to book facilities directly through the platform	Functional requirement

04	Facility Listing and Management	Facility hosts shall be able to list new facilities, including setting and updating details such as description, availability, and pricing.	Functional requirement
05	Usability	The application shall be user-friendly, with an intuitive interface that minimizes the learning curve for new users.	Non-functional requirement
06	Security	The application shall implement robust security measures, including secure data transmission, encrypted storage of sensitive data, and regular security audits.	Non-functional requirement
07	Reliability and Availability	The application shall have a system uptime of 99.9%, excluding scheduled maintenance.	Non-functional requirement
08	Scalability	The application architecture shall support scaling operations to accommodate growing numbers of users and data volume without significant changes to the system.	Non-functional requirement

CHAPTER 3 USE CASE DIAGRAM

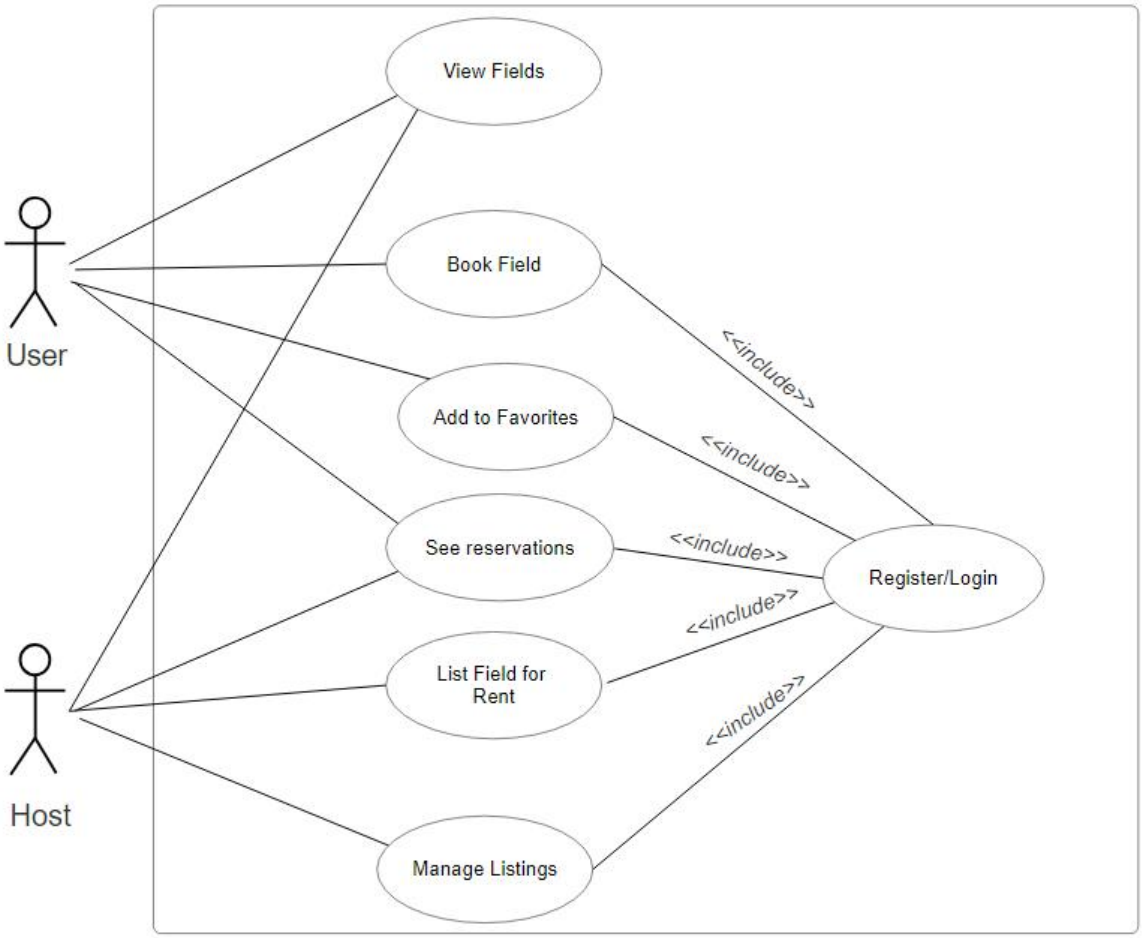


Figure 3.1: Use case Diagram

3.1 Brief explanation

The provided Use Case diagram offers a structured representation of the interactions between two types of users—general users ("User") and field hosts or owners ("Host")—with a sports facility booking system.

3.1.1 Actors

User: Represents general users who interact with the system to view fields, book fields, add fields to favorites, and see their reservations.

Host: Represents users who have permissions to list fields for rent and manage these listings, including editing or deleting them.

3.1.2 Use Cases for User

View Fields: Users can browse and view details of available sports fields without needing to log in.

Book Field: To book a field, users must first register or log in, highlighting the dependency on the authentication process.

Add to Favorites: This allows users to mark fields they like for easy access later, also requiring them to be logged in.

See Reservations: Users can view all their current and past field reservations, a feature that requires user authentication.

3.1.3Use Cases for Host

List Field for Rent: Hosts can add new sports fields to the system for other users to book, necessitating authentication to ensure only authorized hosts can perform this action.

Manage Listings: This allows hosts to modify or remove their field listings, with necessary authentication to maintain security and integrity.

Relationships

Inclusion of Register/Login: The diagram incorporates "Register/Login" as a critical part of use cases where user identity verification is necessary. This includes booking fields, adding favorites, listing fields for rent, and managing listings. The <<include>> relationship is used to denote that authentication is not just a prerequisite but a part of these processes.

CHAPTER 3 ERD:

3.1 Entities:

User:

1. **Attributes**: Includes ID, name, email, image, hashed password, creation and update timestamps, and a list of favorite IDs.
2. **Relationships**:

    Accounts: A one-to-many relationship, indicating that a user can have multiple associated accounts.

    Listings: A one-to-many relationship, showing that a user can list multiple properties.

    Reservations: A one-to-many relationship, meaning a user can make multiple bookings or reservations.

Account:

1. **Attributes**: Includes ID, user ID, type, provider, provider account ID, and various tokens related to authentication.
2. **Relationships**:

    User: A many-to-one relationship linked via user ID, indicating that each account is associated with one user.

Listing:

1. **Attributes**: Includes ID, title, description, image source, creation time, category, room count, bathroom count, guest count, location value, user ID, and price.

2. **Relationships**:

   User: A many-to-one relationship showing that each listing is owned by a user.

   Reservations: A one-to-many relationship, indicating that each listing can have multiple reservations.

Reservation:

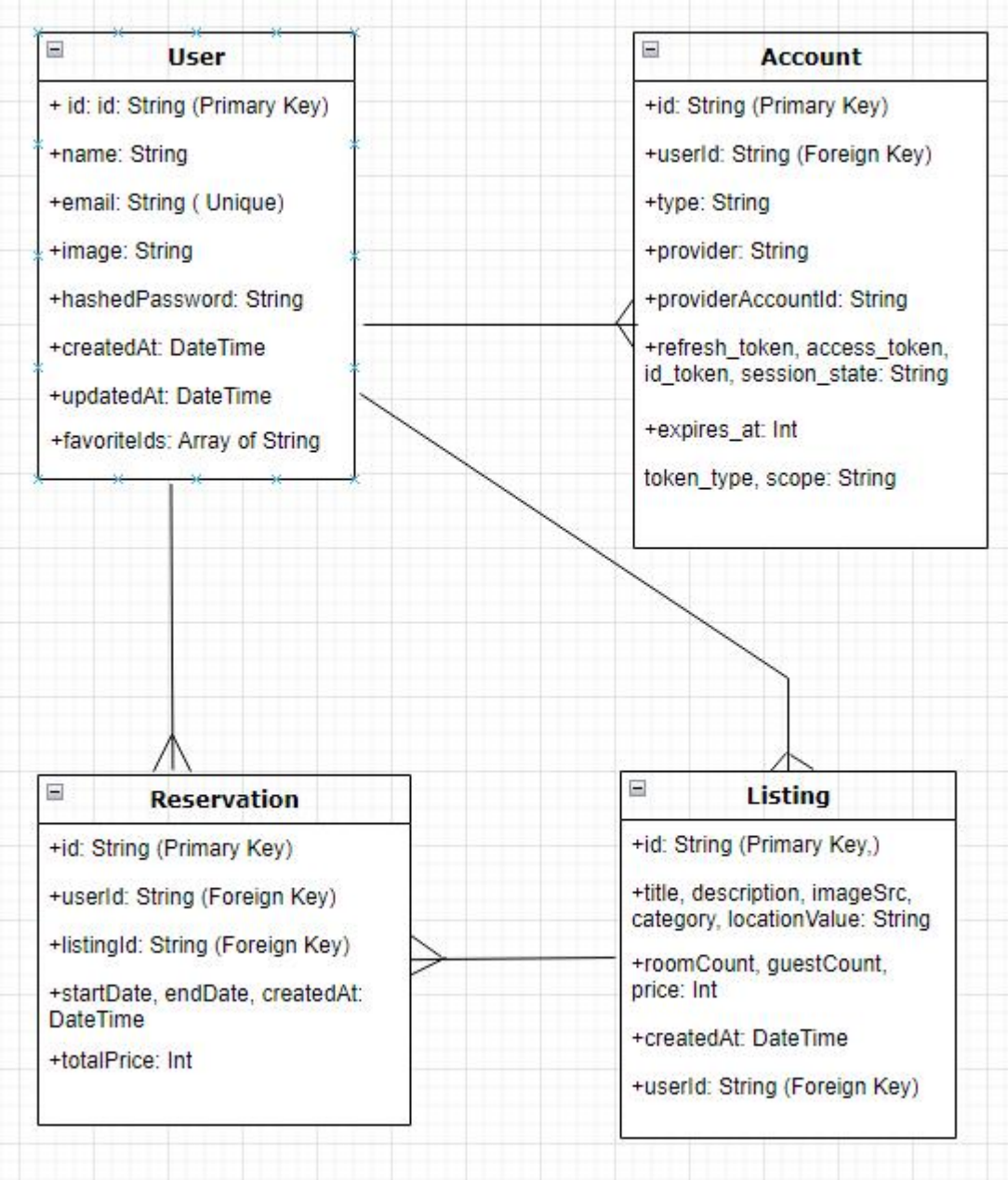
1. **Attributes**: Includes ID, user ID, listing ID, start date, end date, total price, and creation time.
2. **Relationships**:

   User: A many-to-one relationship, indicating that each reservation is made by a user.

   Listing: A many-to-one relationship, showing that each reservation is associated with a particular listing.

Key Points:

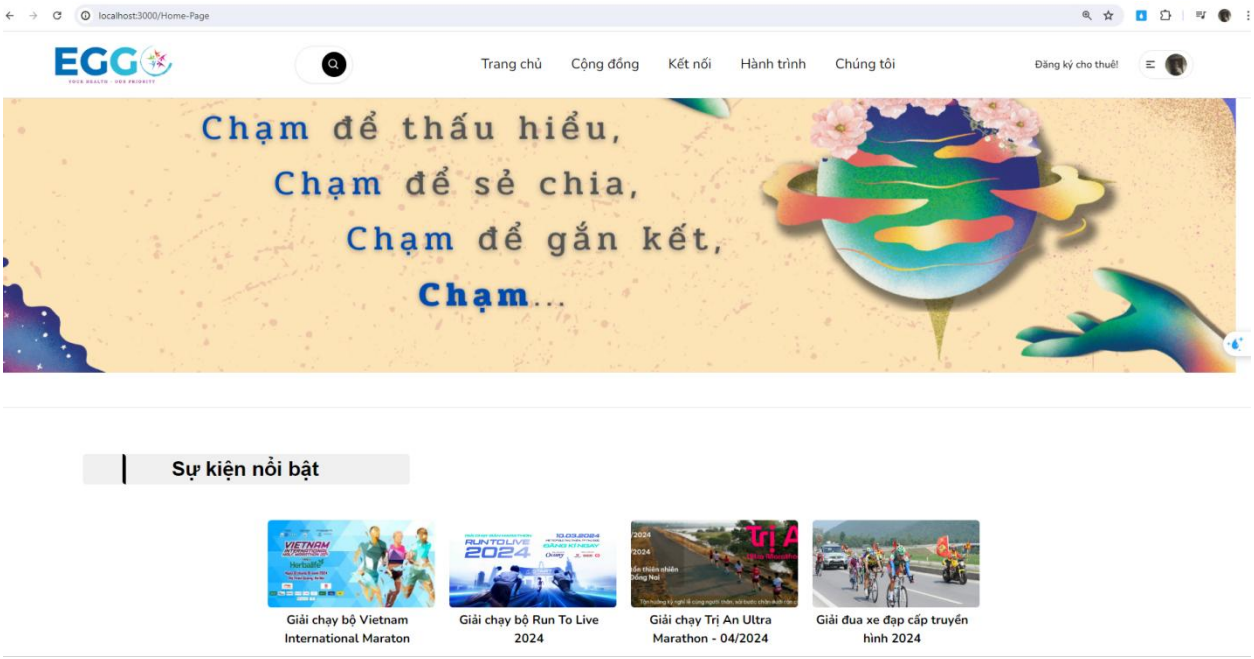
- **Primary Keys**: Each entity has an ID that uniquely identifies it within the database.
- **Foreign Keys**: Used to establish and enforce links between tables, e.g., userId in Account, Listing, and Reservation links back to the User table.
- **Indexes**: Unique constraints on email in the User model and a combined unique index on provider and providerAccountId in the Account model ensure data integrity and speed up query processing.



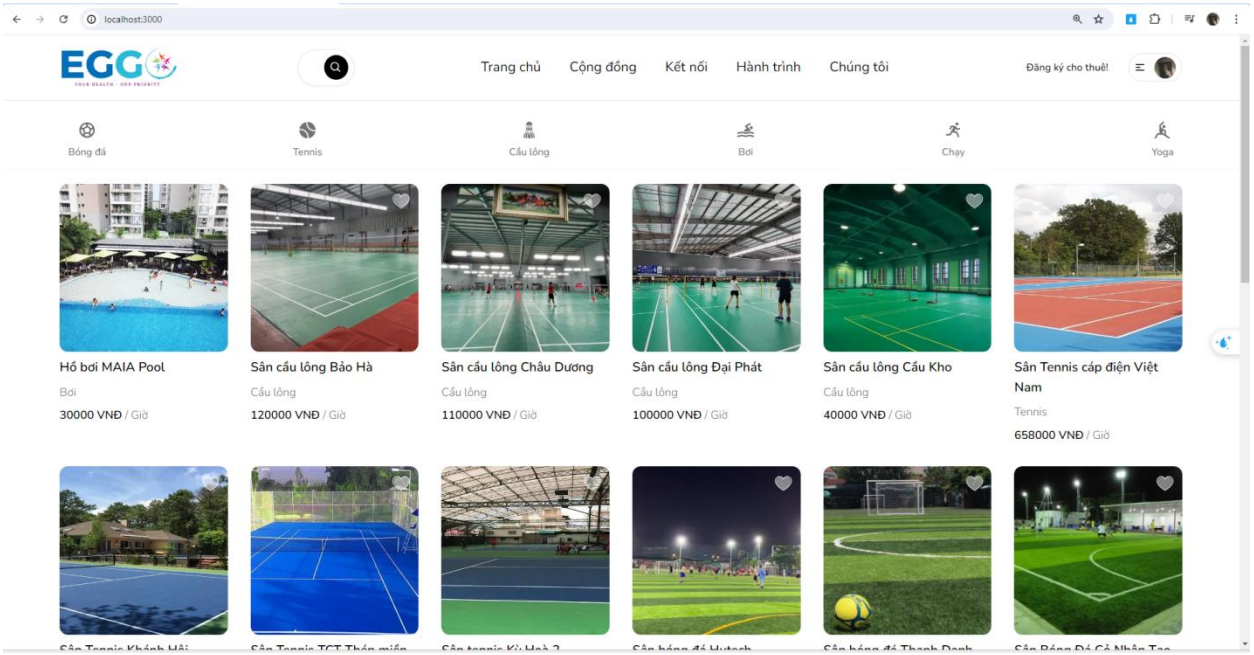
CHAPTER 4:

USER INTERFACE:

The homepage provides a prominent search bar allowing users to easily find events or sports facilities. Additionally, the page highlights major events like marathons and bike races, encouraging user participation or booking facilities for these events.

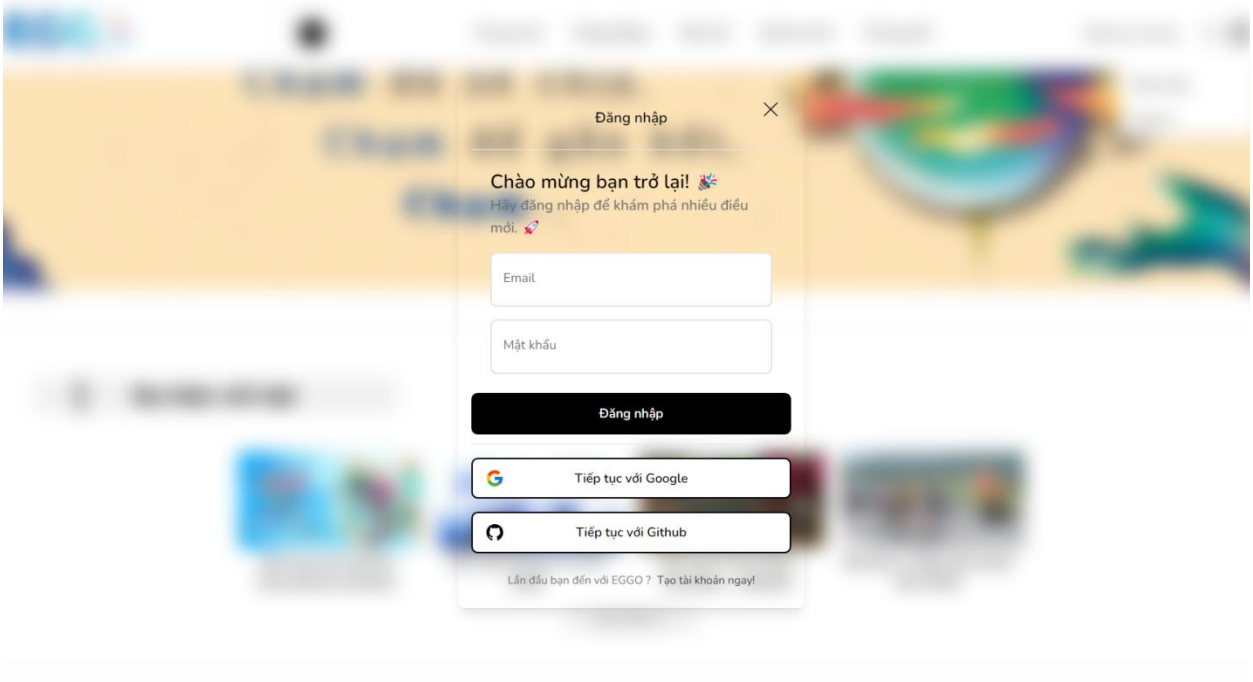


Figures 4.1: Home page

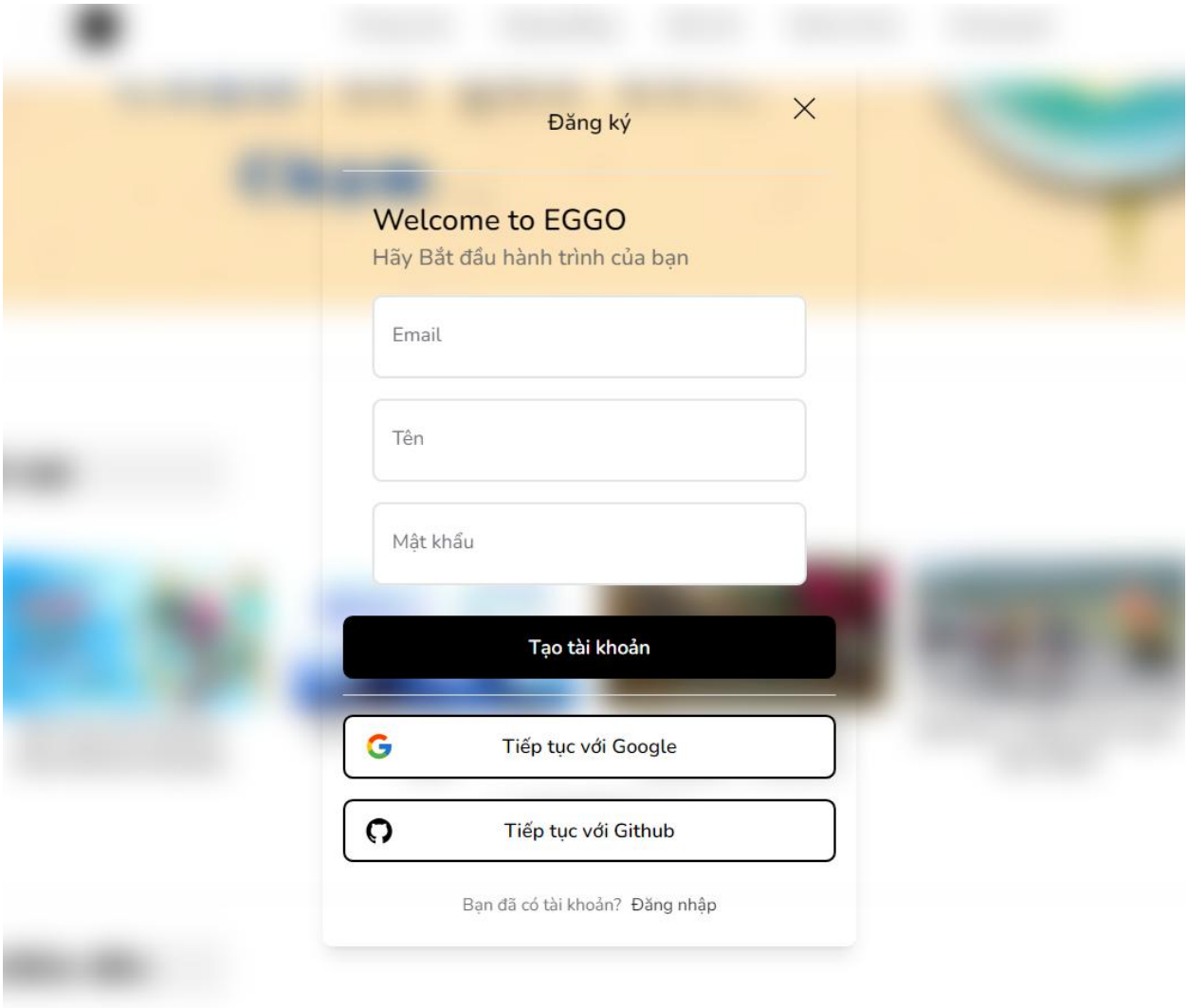


Figures 4.2: View fields

This allows users to create an account or use existing ones to access advanced features like booking, managing favorites, and viewing booking history.

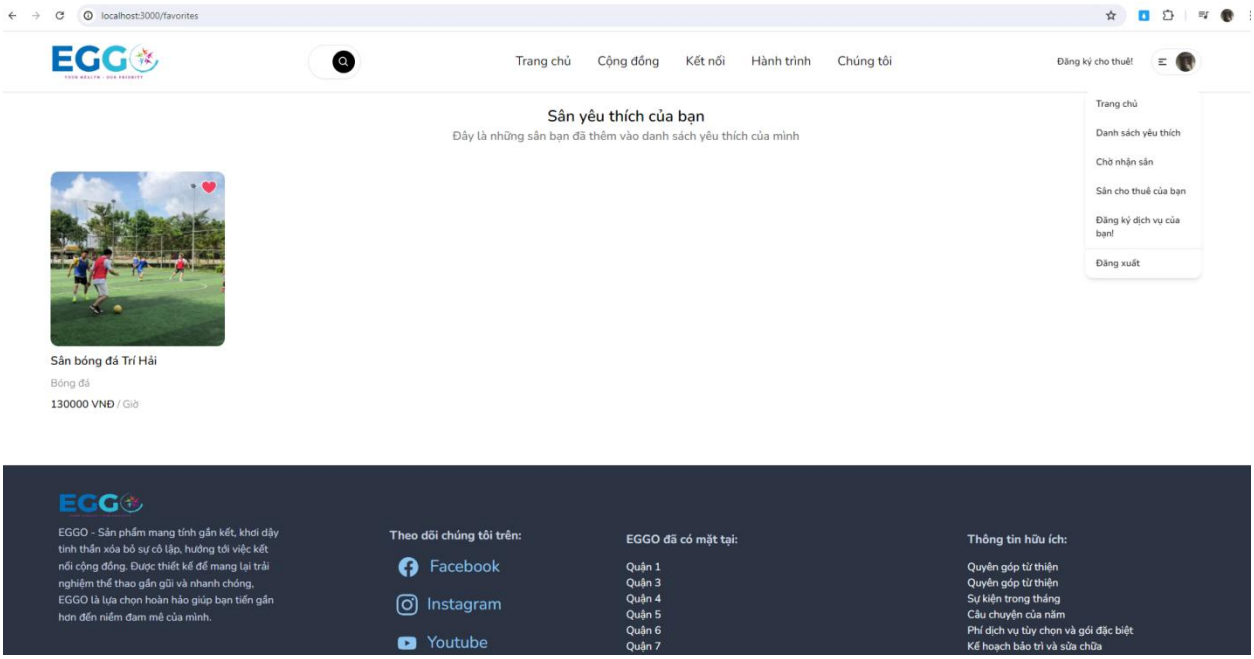


Figures 4.3: Login

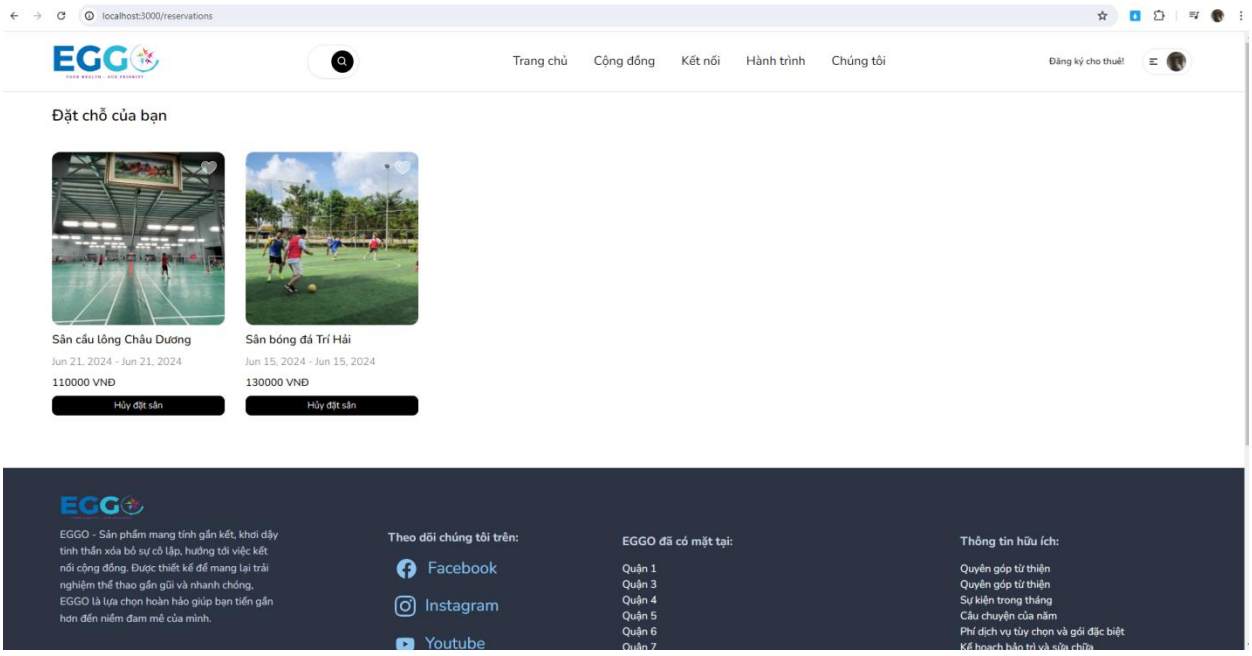


Figures 4.4: Register

Users can view the facilities they've saved in their favorites list and proceed to book quickly and conveniently.

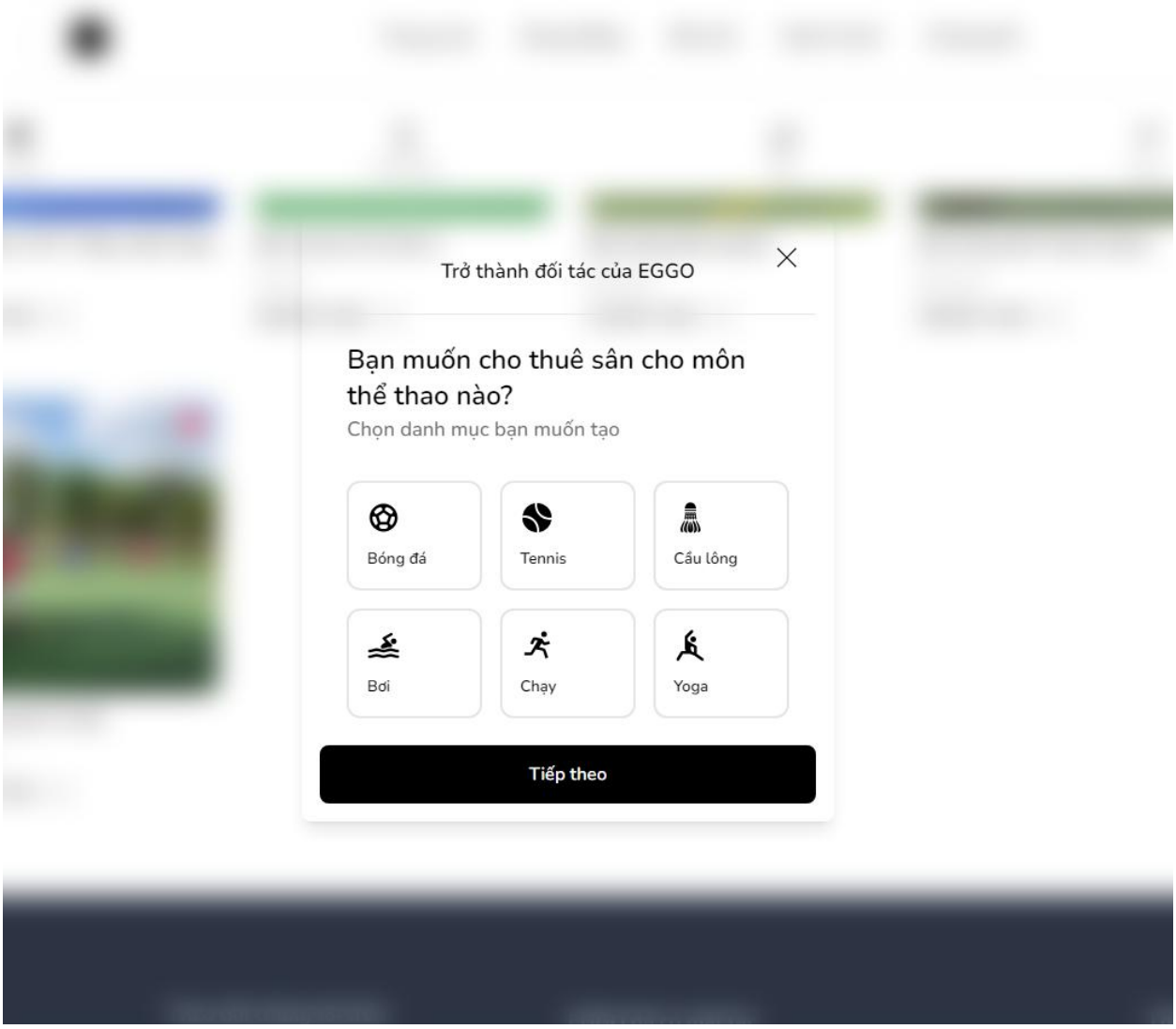


Figures 4.5: Favorites



Figures 4.6: Reservations

Facility owners can easily register their venues on the platform. They need to provide the name and address of the facility, then set the rental price. This process enhances accessibility and management for facility owners.



Figures 4.7: Register Facility

REFERENCE