Backswing Golf Application

Logan Wongbangchuad
Department of Computer Science
Angelo State University
San Angelo, Texas
ldellwong1@gmail.com
GitHub: LoganWongbangchuad/BackswingMobileApp

***Abstract*—This application provides the user a space for them to upload and compare golf scores with friends over various different courses. By utilizing frameworks, such as .NET MAUI with Blazer Hybrid and Bootstrap and using a local database SQLite, I created a fully functional golf application, with a nice looking user interface, where golf enthusiasts can keep track of scores through different courses and compare them to friends. Through the use of this application users will be able to awaken their competitive side of golf.**

***Index Terms*—.NET MAUI with Blazer Hybrid, SQLite, Bootstrap.**

I. Introduction

Over the course of this semester, I was presented with the opportunity to design, and develop a mobile application of my own choosing, with little restrictions. This project allowed me to explore software development, test my creativity, technical skills, and problem solving abilities. The result of this effort is a golf score keeping application that aims to better the experience of golfers by offering a social and interactive platform.

In this report, I will detail the various aspects of the development process, including the software tools, design methods, implementation, and problem solving strategies that I used. Additionally, I will discuss the challenges encountered during development and discuss how I overcame them, highlighting the key lessons learned along the way. This document serves not only as a record of the work I have completed but also as an analysis of the mistakes and struggles that helped shape the final product.

The primary objective of this application is to provide users with a way to record and track their golf scores across a different golf courses. Other than score keeping, the app also has a social component, giving users the ability to compare their scores with friends and engage in competitions. To include a personalization aspect, the app also includes user account functions, which allow users to create, sign in, and modify their account.

This project brings together software engineering and user focused design to influence a social aspect for the golfing community. By integrating functionality, usability, and a competitive social element, the app aims to provide golfers with a great tool that boost their enjoyment of the sport. Through this report, I hope to convey the technical achievements, creative decisions, and valuable insights gained during the development of this application.

II. Related Work

One application that significantly inspired the development of my golf app is Leaderboard, available on the iOS App Store. Leaderboard has been an app that me and my friends use constantly. It offers many features tailored to golf players. These features include the ability to create and compare golf scores with friends, create tournaments, send direct messages, access course maps, track scores, and calculate handicaps. Its design and focus on creating a sense of community among golfers make it a standout application in the golf community.

While I recognized early on that creating an app as big and with as many featurs as Leaderboard would be more work that could be done in a single semester, it provided a significant source of inspiration for my project. I was particularly drawn to the social and competitive aspects of the app, especially the leaderboard feature. This component allows users to compare scores across different courses, creating a sense of friendly competition. For my friends and me, this functionality has added more competitiveness to our golf outings, turning them into not just a sport but a shared experience of competition.

Having used Leaderboard many times, I developed an appreciation for the value it adds to the golfing community. I started to see ways that a more focused or simplified app could better fit certain needs or preferences. This realization motivated me to attempt creating my own version of a golf app. I wanted to create one that would incorporate some of the features I enjoyed most while allowing me to implement my own ideas.

The development of my golf application represents both a challenge and an opportunity to apply my technical skills to a project inspired by my own interests. While I knew my app would be less intricate than Leaderboard, I aimed to capture the essence of its biggest features, particularly the leaderboard, and adapt them to my project throughout the semester. The creation process has not only deepened my appreciation for mobile app development but also given me the oppotunity to contribute something meaningful to the golfing community.

While the Leaderboard app served as my primary inspiration, I also drew significant influence from the work of Andrew Tiger, the Dean of the Norris-Vincent College of Business. His research on golf, particularly studies such as ”Improving

Golf Pace of Play Using Time Study Analysis: Influencing Factors on the Green and Tee Box" and "Which Type of Golfers are Interested in Playing a Reversible Golf Course Design: A Survey Study", provided me with a fascinating perspective on the sport. Although my project is distinct from his research in terms of scope and objectives, his findings offered valuable insights into the complexity of golf and inspired me to think about how data and analysis could enhance the user experience.

One study that particularly captured my interest was his analysis of golf's pace of play. Through meticulous data collection and statistical analysis, Tiger examined various factors that influence the time it takes to complete a golf course. By employing tools such as Microsoft Excel and Monte Carlo simulations, he evaluated variables such as green speeds, hole design, distance between holes, and par ratings. This research allowed him to calculate average times for completing individual holes and entire courses, offering insights into how course design and player behavior can optimize pace of play. His work not only earned him recognition from the golfing community but also gained attention from the United States Golf Association (USGA), which has directly published his findings.

As both a golfer and a computer science student with an interest in data science and statistical analysis, I found Tiger's research deeply inspiring. His ability to blend analysis with real world applications in golf resonated with me, encouraging me to consider how data driven approaches might influence my future works and journey learning AI and data science. While the scope of my project does not include detailed time studies or course design considerations, his work demonstrated the potential for applying computational techniques to solve real world problems in golf.

In reflecting on Tiger's contributions, I found myself particularly inspired by his methodical approach to understanding the sport. His focus on analyzing variables that influence player experience aligns with my own goal of creating an app that not only tracks scores but also improves the overall enjoyment of the game. By incorporating features like leaderboards and creating friendly competition, my app aims to address a different aspect of the golfing experience, rooted in creating connections, social aspects, and influencing improvement. Tiger's work ultimately reaffirmed the value of combining technical skills with a passion for golf, a principle that I was influenced to apply throughout my project.

III. Design

The design process for this application was not as structured or intricate as it could have been. Being my first major project, I lacked experience in software design and jumped directly into programming without fully outlining a comprehensive plan. While doing this allowed me to begin implementation quickly, it also would create several challenges down the road. For instance, modifying certain parts of the code often resulted in unintended problems elsewhere, as dependencies and interactions between components were not fully accounted for in the initial stages.

This experience underscored the importance of following the best practices in software design. Despite not following a formal design method, I gained a deeper understanding of why careful planning and structured design are critical for successful software development. One key insight came from discussions with Dr. Dogdu, who highlighted that nearly 70 percent of software companies fail due to poor design practices. This statistic resonated with me as I encountered firsthand the difficulties that arise when good design methods are overlooked. While my approach lacked structure to the design, it reinforced the necessity of good design practices to ensure scalability, maintainability, and overall project success.

Although my design methods were informal, I did set some foundational goals to guide the development process. Before writing any code, I outlined the key features I wanted the application to have, such as the specific pages to be implemented and the integration of a database to manage user and course data. These goals served as a roadmap for the project, even if the other details of the design were corrected through trial and error during development.

Moving forward, I recognize the value of following structured design practices, such as creating detailed specifications, wireframes, and flowcharts before beginning development. Such practices not only help identify potential challenges early but also create a framework that makes the code more resilient to changes. While this project may not have followed standard design principles, it provided a valuable learning experience that will shape how I approach software development in the future.

IV. Methods

The development of this application was accomplished using the .NET MAUI Blazor Hybrid framework. This framework combines the power of three key technologies: C#, HTML, and CSS (with Bootstrap integration). Each technology played a distinct role in the design and function of the application, enabling implementation of backend logic, user interface design, and database interactions.

A. Backend Development

The backend functionality was primarily implemented using C#, which served as the foundation for creating services, models, and managing the database. Within the services layer, I developed the database integration and implemented CRUD operations (Create, Read, Update, Delete) to handle essential tasks such as adding, deleting, and modifying data. These operations were crucial for ensuring ideal interaction between the user interface and the underlying data storage. The models, which were also written in C#, provided a structure for handling data throughout the application and reducing the chance of errors.

B. Razor Pages and User Interface

The frontend of the application used Razor Pages, which enabled the integration of C# code within HTML. This allowed me to generate content and use features like variables, loops, and functions directly within the HTML structure. Bootstrap was employed for styling and layout, which made the user interface design process much easier. Instead of writing custom CSS from scratch, I used Bootstrap's prebuilt components and grid system to create a visually appealing and mobile/desktop friendly design. This combination of softwares allowed for an efficient development process and a great looking and functional user interface.

C. Navigation

Efficient navigation was achieved through the use of Blazor's routing capabilities. The navigation setup was configured in the App.razor file using the ¡Router¿ component, and each Razor page was defined with the @page directive. This approach allowed for clear mapping of URLs to components and provided a user friendly navigation experience. Below are the key navigation elements implemented:

Navbar: The navigation bar, located at the top right of the mobile interface, features a dropdown menu with four primary options: Home, Courses, Sign In, and Profile. This intuitive design ensures that users can easily access the app's main features.

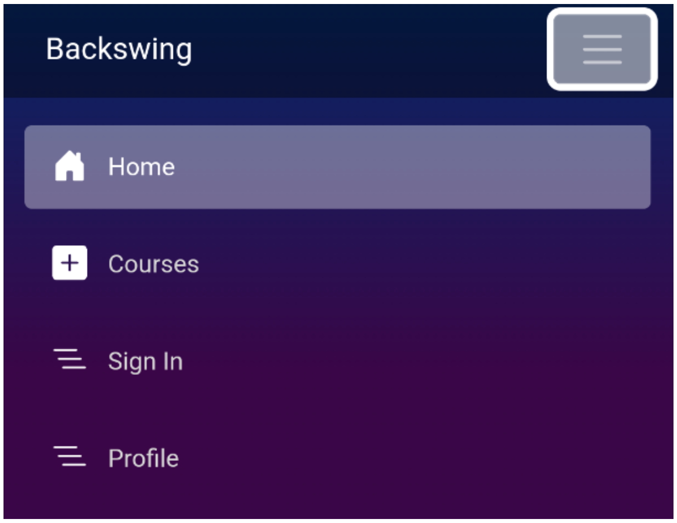


Fig. 1. Login Page for Backswing

D. Razor Pages

Home Page: The Home Page greets users with a personalized welcome message, displaying their name alongside a brief introduction to the app's purpose. A prominent button on this page directs users to the Courses Page, encouraging exploration of the app's core features.

Courses Page: The Courses Page dynamically displays a list of golf courses using a loop. Each course name is wrapped in a NavLink, making it clickable and allowing users to navigate to detailed course information. The Course Details Page uses the @page directive with route parameters, enabling the display of content based on the selected course's ID. This structure also allows users to navigate directly to specific course pages via URL.

Leaderboard Page: The Leaderboard Page presents users with a list of players and their scores for a selected course. This data is retrieved from the database and displayed in a clean, organized manner. Users can navigate to this page from the courses section and interact with the leaderboard to view and compare scores.

Profile Page: The Profile Page displays the user's personal information, including their address, phone number, email, and username. This page also includes an Edit Profile button, allowing users to update their details. Changes are reflected across the application, ensuring consistency.

Create Account Page: If a user does not have an account, they can navigate to the Create Account Page by clicking "Join Now" on the Sign In Page. Here, they are prompted to enter their username, password, phone number, email, and address. After successfully registering, users are redirected back to the Sign In Page, where they can log in with their new credentials.

Sign In Page: The Sign In Page validates the user's login credentials. Upon successful authentication, users are redirected to the Home Page. This page ensures secure access and a smooth onboarding experience for returning users.

Add Course Page: The Add Course Page allows users to contribute to the golf course database by entering a course name and the number of holes it offers. Once submitted, the course is added to the database, and the user is redirected back to the Courses Page, where the new course is visible.

Add Score Page: Similar to the Add Course Page, the Add Score Page enables users to submit their name and score for a particular game. This feature is accessible from the leaderboard and ensures that scores are easily added and updated.

Edit Profile Page: The Edit Profile Page allows users to update their account information, including their username, password, email, address, and phone number. Upon saving changes, users are redirected back to their profile, where the updates are displayed immediately.

E. Database Methods

Database Structure: The database was structured with multiple tables, including Leaderboard, Course, and User. These tables were designed with primary keys, foreign keys, and constraints to ensure data integrity and establish relationships between entities.

SQLite Integration: I used SQLite as the local database solution due to its data management capabilities and since a local database was all that was needed instead of a datbase like mySQL or SQLServer. This integration enabled storage and retrieval of user, course, and leaderboard data, increasing app performance.

CRUD Operations: I implemented CRUD operations using Entity Framework Core, which provided a clean and intuitive interface for interacting with the SQLite database. These operations allowed for efficient data management and ensured a consistent user experience.

F. Services and Repositories

AuthService: The AuthService handled user authentication, including validating credentials during login, checking for duplicate usernames or emails during registration, and creating new users. This service ensured secure and reliable user management.

SessionService: The SessionService maintained session data for the currently logged in user, allowing for consistent tracking across pages. This service played a role in preserving user context and the experience.

CourseService and PlayerService: Both the CourseService and PlayerService managed database interactions for courses and players. They included CRUD operations and methods for retrieving data, such as fetching a list of players for the leaderboard or list of courses for the courses page. It also includes methods to add and delete course and players

G. Models

Player Model: The Player Model defined the structure of user data, including fields such as username, email, and password. It provided a consistent format for representing users throughout the application.

Course Model: The Course Model specified the structure of course data, including fields for course ID, name, image URL, description, and the number of holes. Although only course ID, name, and number of holes were used, the model ensured data consistency across components.

V. RESULTS

By implementing the diverse methods and approaches detailed above, I successfully developed a fully functioning golf application that not only meets its intended purpose but also features a visually appealing and user friendly interface. The integration of the backend logic, efficient navigation, database management, and UI design elements contributed to the creation of a app for tracking and comparing golf scores.

This app shows the result of thoughtful planning, ongoing development, and problem-solving to create something that improves the user experience. Its features, such as course listings, leaderboards, user profiles, and account management, highlight the development process.

Below, you will find images showcasing the design of the application. These visuals show how the methods and technologies discussed above were applied to achieve the final results, reflecting both the technical and visual accomplishments of this project.

VI. APPLICATION NAVIGATION AND VISUALS

When first launching the Backswing application, users are greeted with the login page, where they can sign in to access the app's features. For those without an account, a "Join Now" button redirects them to the registration page.

On the registration page, users are prompted to provide their username, email, phone number, address, and password. Upon completing the form and selecting "Create Account," they are redirected back to the login page, ready to log in with their newly created credentials.

After logging in successfully, users are directed to the home page, which features a personalized welcome message, displays their username, and provides a brief overview of what Backswing offers.

From the home page, users can navigate to various sections of the app, including the profile page, accessible through the dropdown navigation menu. On the profile page, users can view their profile picture along with key account details. Additionally, an "Edit Profile" button allows users to make changes to their information.

The "Edit Profile" page displays all user information in editable fields. After making updates and clicking "Save Changes," users are redirected back to the profile page, where the revised information is immediately reflected.

The courses page, accessible through the navigation menu or through the home page's "Courses" button, presents a list of golf courses, each accompanied by the number of holes, and two actionable buttons: "Leaderboards" and "Delete." From this page, users can either add new courses to the list or view leaderboards for existing courses.

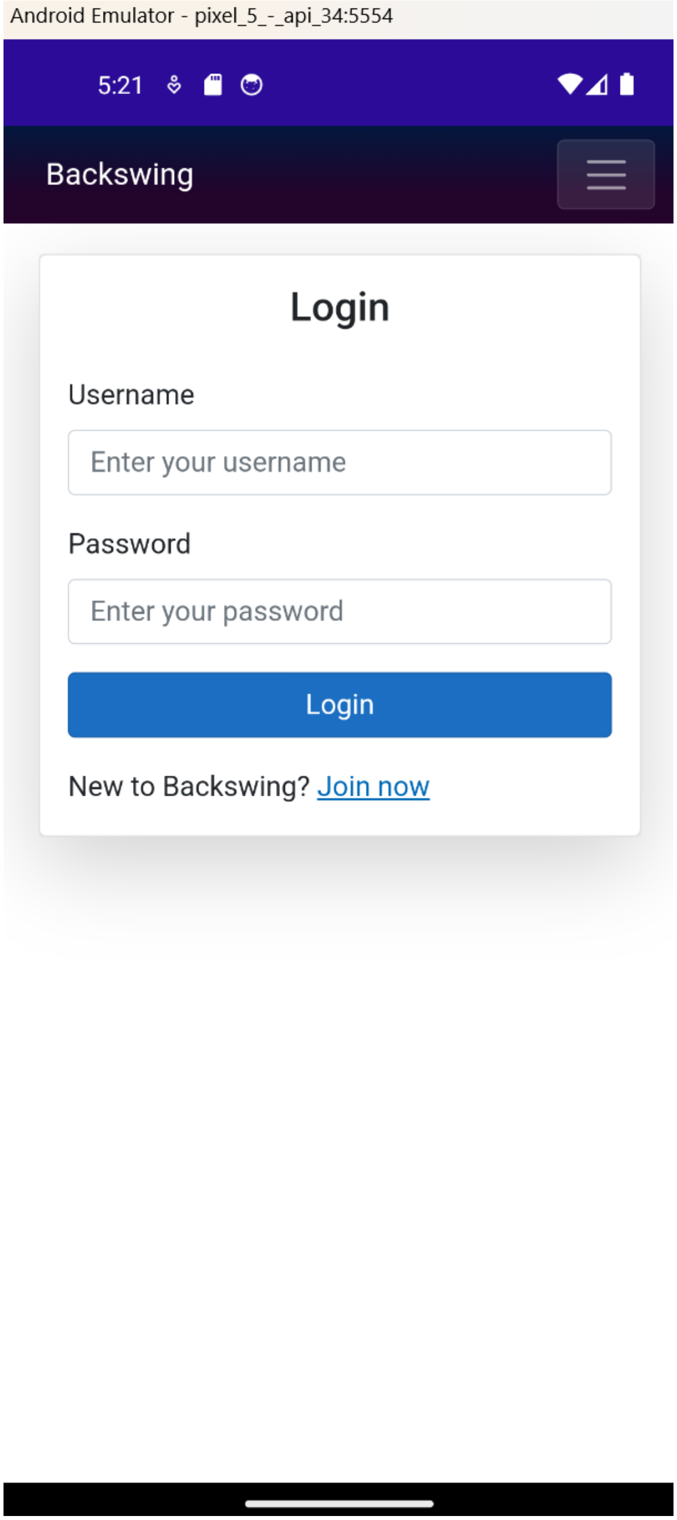
To add a new course, users can select the "Add Course" button, which directs them to a page where they are prompted to input the course name and the number of holes. Once submitted, they are redirected back to the courses page, where the new course is displayed.

To view a course's leaderboard, users can click the "Leaderboards" button next to the desired course. The leaderboard page provides an overview of scores associated with that course, displaying each player's name and their respective score. At the top of the page, users will find an "Add Score" button, which allows them to record new scores for the course.

On the "Add Score" page, users are prompted to input their name and score for the selected course. After submitting the score, they are redirected back to the leaderboard, where the new score is reflected in real-time.

If the leaderboard is empty, a message informs the user that no scores are currently available and encourages them to add a score to get started.

This flow ensures users can seamlessly navigate the application, manage their profiles, and interact with the course and leaderboard functionalities with ease.



Android Emulator - pixel_5_-_api_34:5554

5:21

Backswing

Login

Username

Enter your username

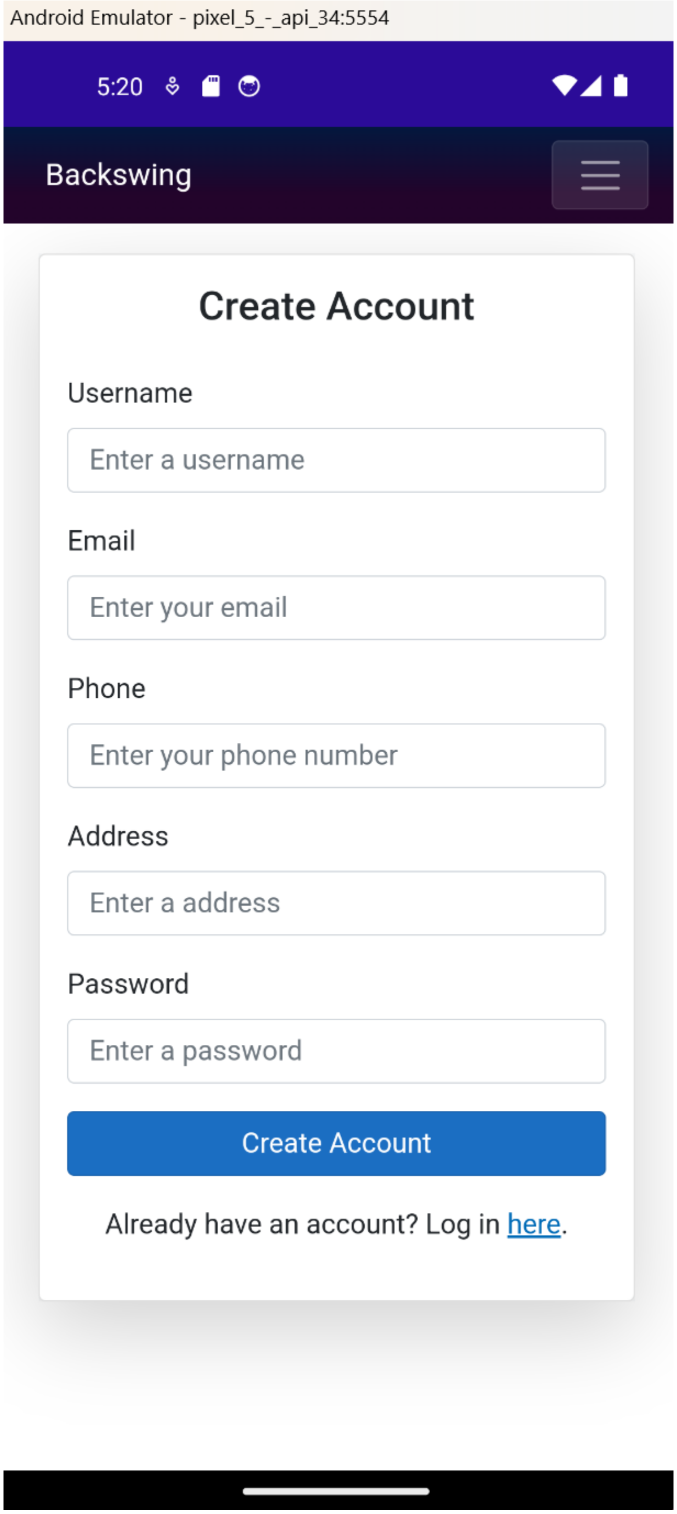
Password

Enter your password

Login

New to Backswing? Join now

Fig. 2. Login Page for Backswing



Android Emulator - pixel_5_-_api_34:5554

5:20

Backswing

Create Account

Username

Enter a username

Email

Enter your email

Phone

Enter your phone number

Address

Enter a address

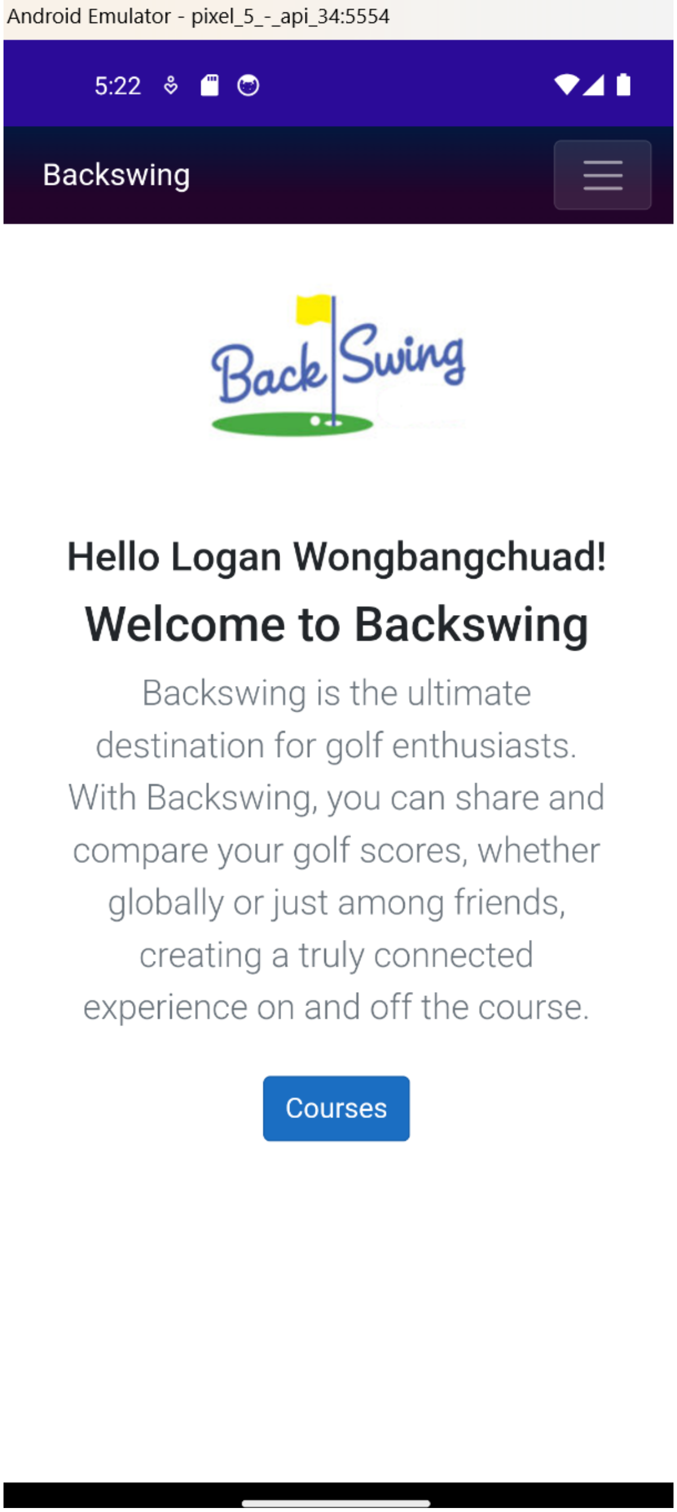
Password

Enter a password

Create Account

Already have an account? Log in here.

Fig. 3. Sign up page for Backswing

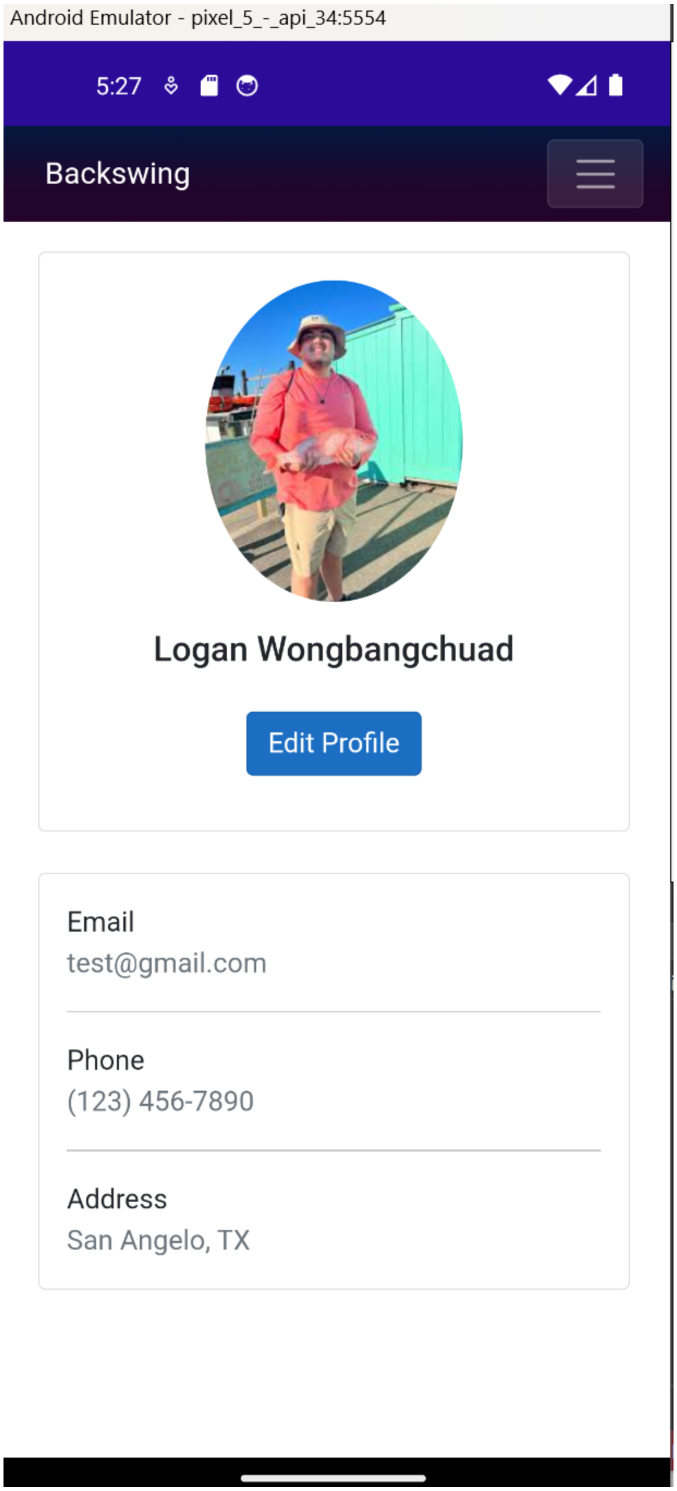


Hello Logan Wongbangchuad!

Welcome to Backswing

Backswing is the ultimate destination for golf enthusiasts. With Backswing, you can share and compare your golf scores, whether globally or just among friends, creating a truly connected experience on and off the course.

Fig. 4. Home page for Backswing



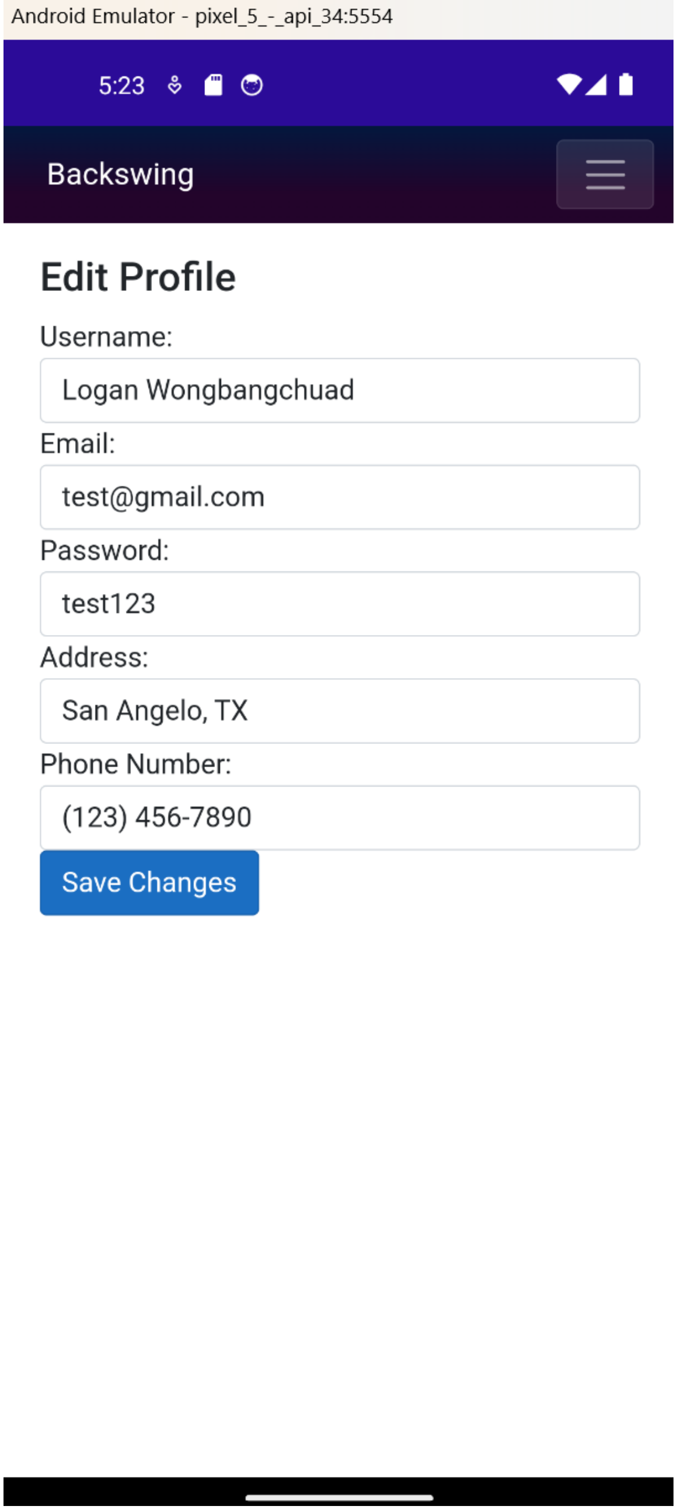
Logan Wongbangchuad

Email
test@gmail.com

Phone
(123) 456-7890

Address
San Angelo, TX

Fig. 5. Profile page for Backswing



Edit Profile

Username: Logan Wongbangchuad

Email: test@gmail.com

Password: test123

Address: San Angelo, TX

Phone Number: (123) 456-7890

Fig. 6. Edit Profile page for Backswing

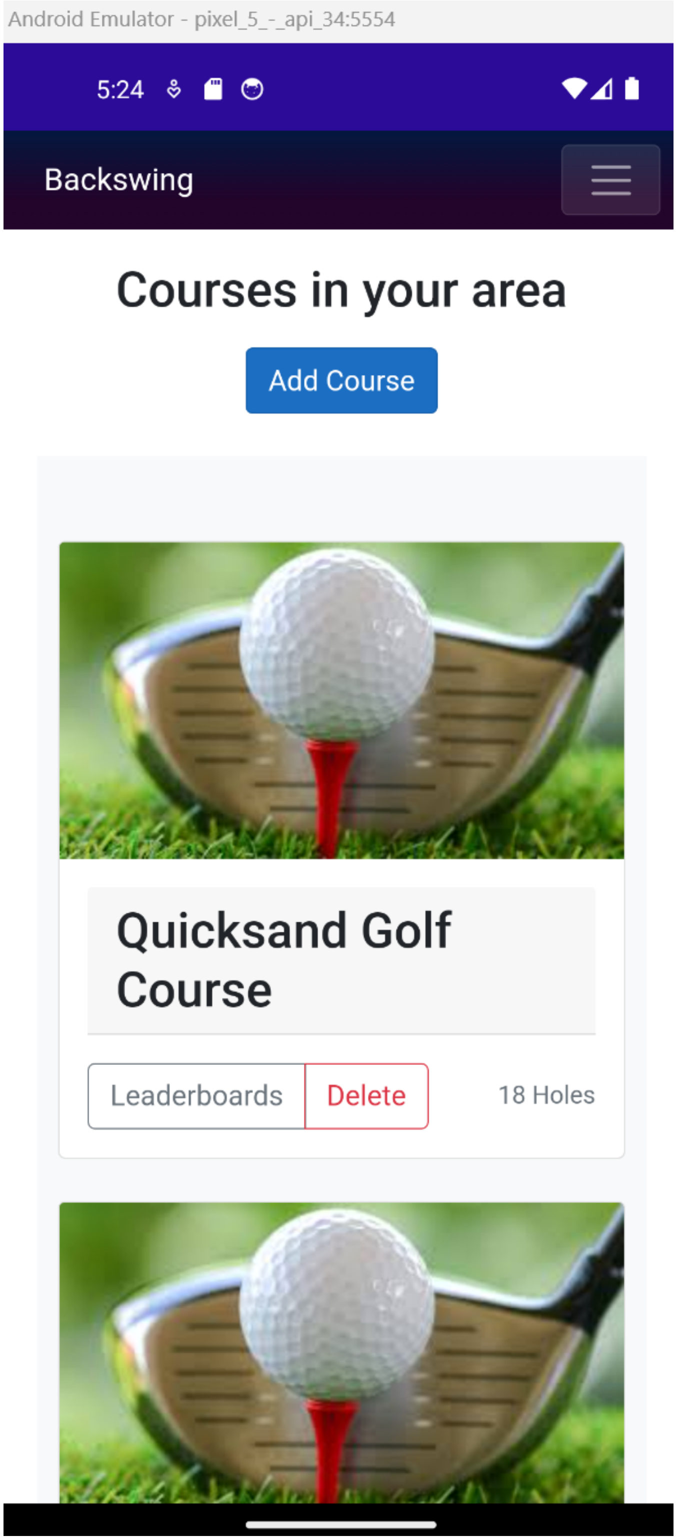
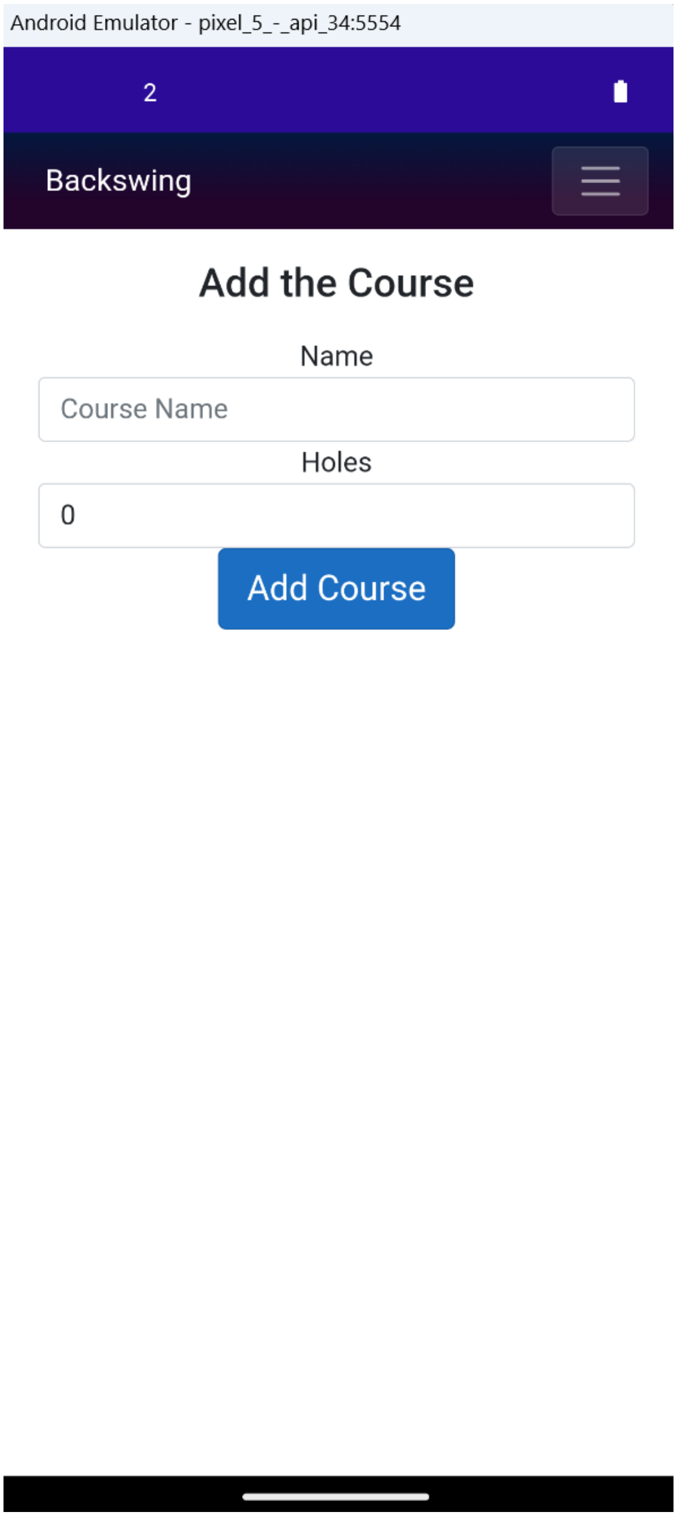


Fig. 7. List of all courses

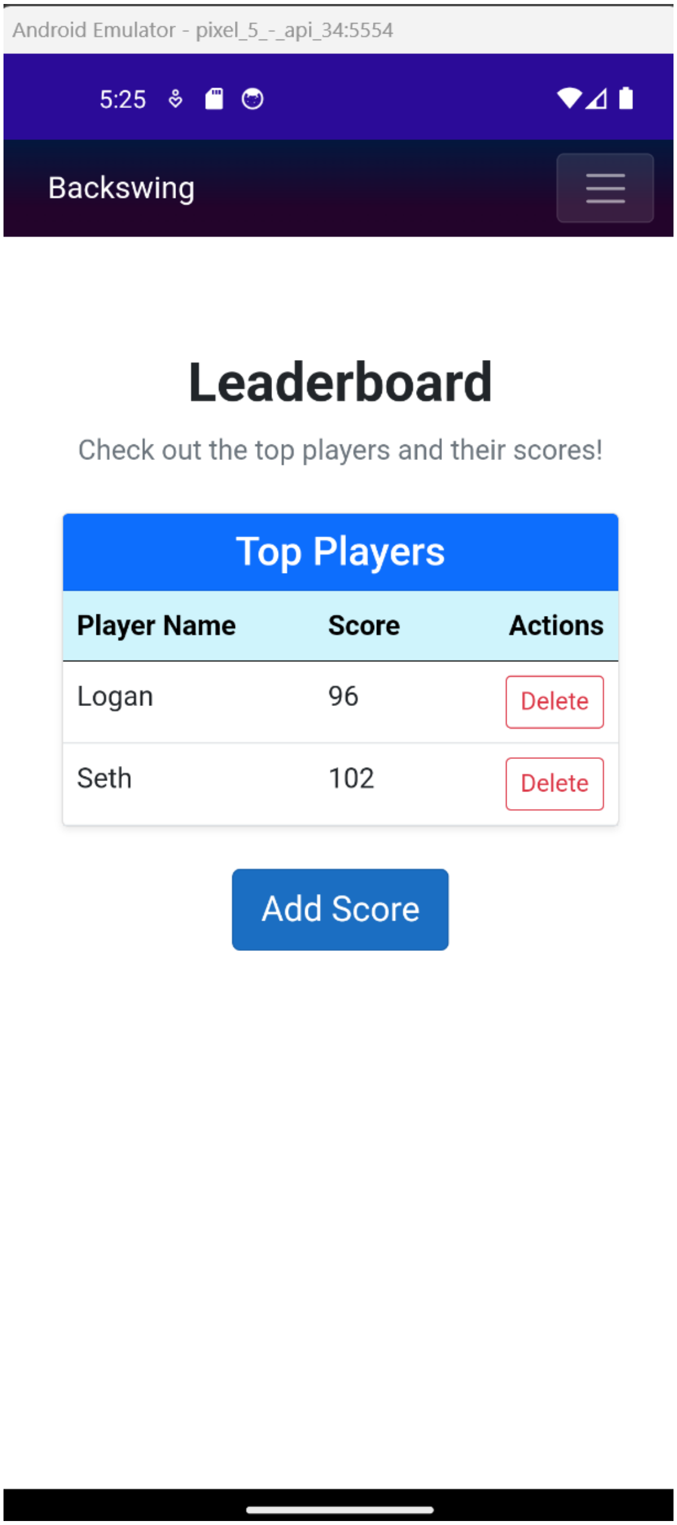


Add the Course

Name: Course Name

Holes: 0

Fig. 8. Where to add a course to the list



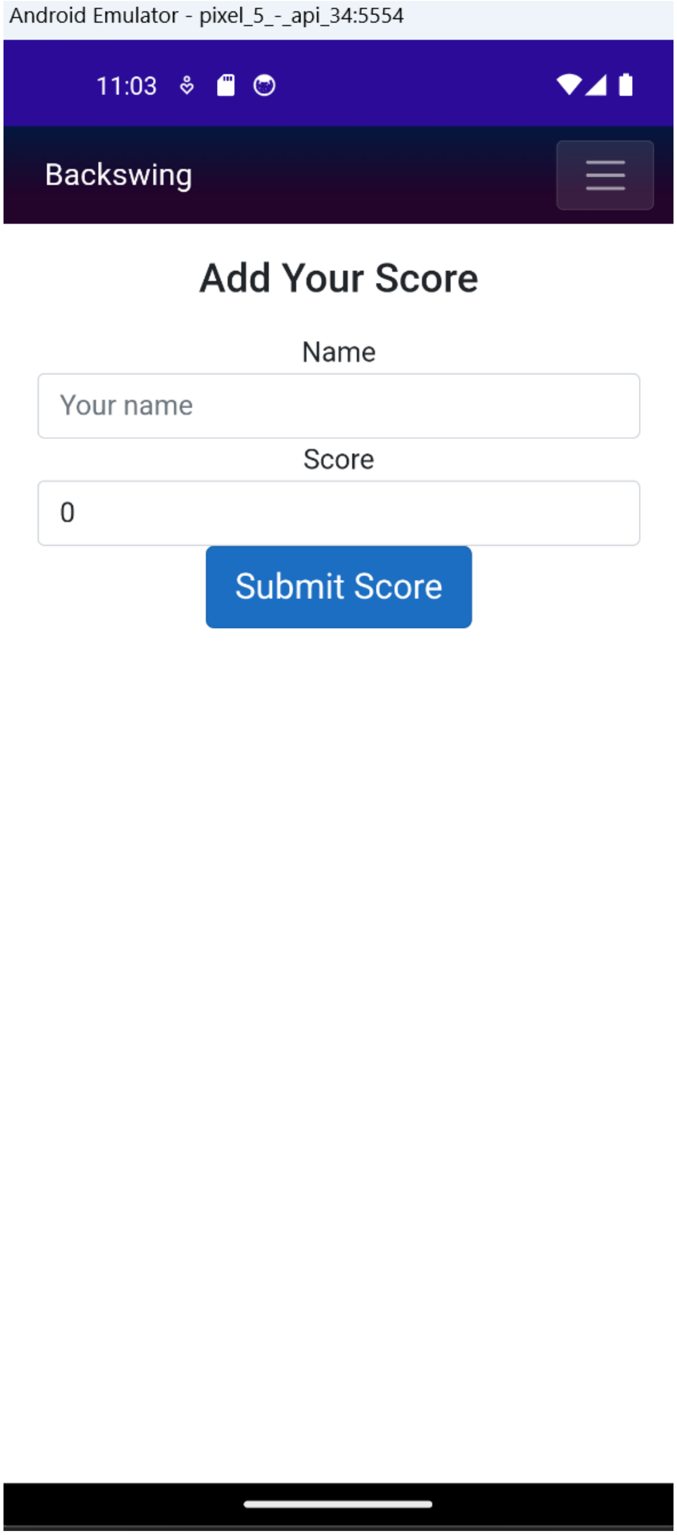
Leaderboard

Check out the top players and their scores!

Top Players

Player Name	Score	Actions
Logan	96	Delete
Seth	102	Delete

Fig. 9. The course leaderboard



Add Your Score

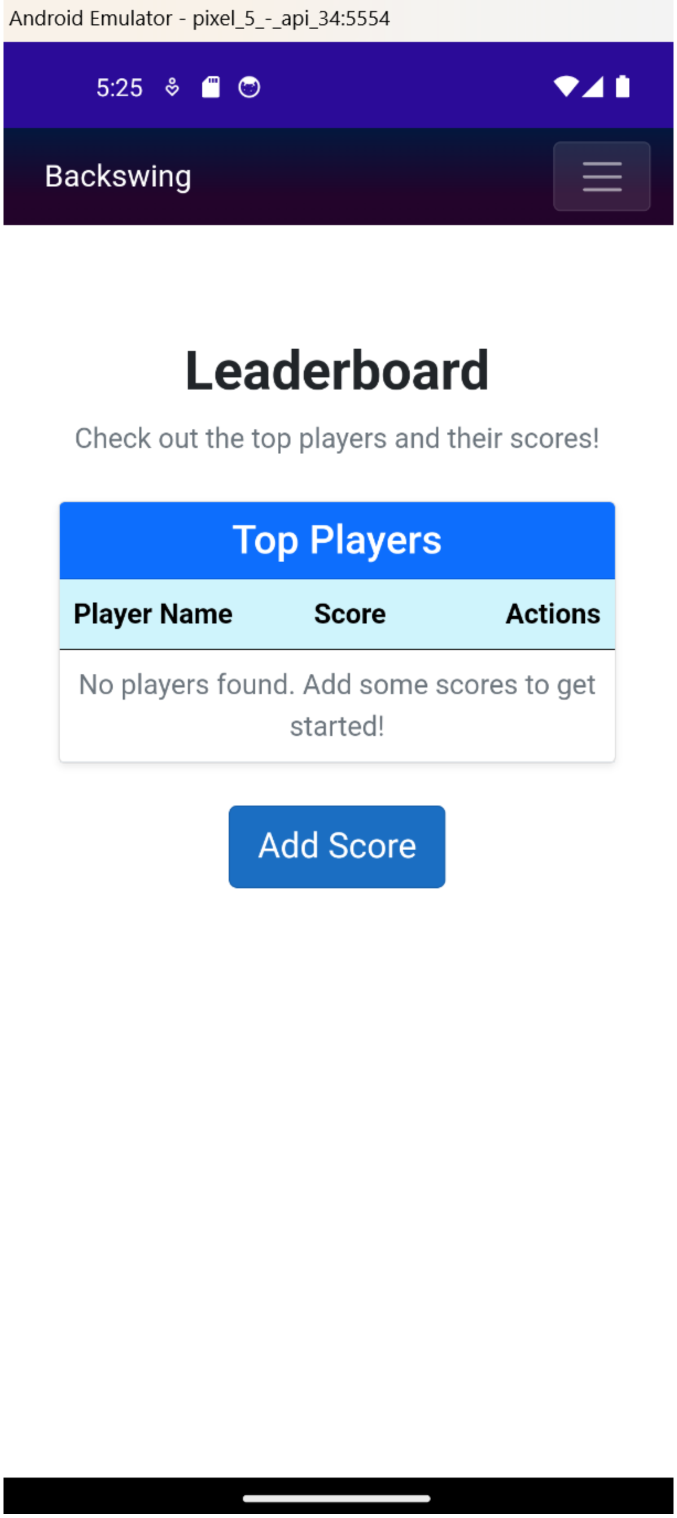
Name

Your name

Score

0

Fig. 10. Where to add a score to the leaderboard



Leaderboard

Check out the top players and their scores!

Top Players

Player Name	Score	Actions
No players found. Add some scores to get started!		

Fig. 11. Leaderboard page without any scores

VII. Conclusions and Future Works

In conclusion, Backswing is fully functional, allowing users to create profiles, view courses, and add their scores to the leaderboard. This project demonstrates a personal growth in mobile app development. However, given additional time, there are several improvements I would have made to improve the functionality and overall experience. These additions are still features I plan implement after the class to better the application.

One key improvement would be the implementing permissions. Incorporating role based controls, such as the difference between regular users and admin users, would have been a key feature for enhancing security and management. For example, an admin role could be given rights to perform sensitive actions such as deleting courses or modifying app settings, while regular users would have restricted access. This separation of roles is important for maintaining control over changes and ensuring the safety of the apps data.

Another feature I would have liked to add is photo uploading. Although static images are used throughout the current application, users do not have the ability to upload their own photos. Integrating a photo upload feature would better the experience, allowing them to customize their profiles or add images to courses. I would likely implement this function using a service such as Cloudinary, which makes the backend processes of image uploading and storage easier. This feature would make the application more engaging.

Additionally, password hashing is an area of key importance that I would have prioritized for a polished version of the app. Currently, passwords are stored as plain text in the database, which is acceptable for a local application used in very limited settings, but it is highly insecure for any software deployed for public use. Implementing password hashing using algorithms such as bcrypt or Argon2 would significantly enhance security, ensuring that user credentials are protected against potential data breaches or unauthorized access.

Lastly, to enhance the social aspect of the app, I would explore features such as allowing users to follow and be followed by others. This function could create a more community environment where users can connect with friends, view each other's progress, and participate in competitions. Direct messaging would also be an addition added, enabling users to communicate directly within the app. These features would transform Backswing into a more interactive platform and would follow other designs seen in other social apps.

Overall, while the application achieves its primary goals and provides users with a nice leaderboard, these improvements could take it to a stronger and more polished level. This would not only improve the app's function but also expand its appeal and usability, influencing future development and scalability.

VIII. References

I designed this app based on my own ideas, shaping it to match my preferences and vision for how it should work and look. While golf leaderboard applications already exist in the market, I aimed to create a custom solution that reflects my approach to both the technical and visually pleasing aspects of the project. Although I drew inspiration from the golf app Leaderboard my application was built entirely from scratch to ensure it aligns with my goals and desired features.

Throughout the development process, I used various resources to address challenges and improve my understanding of the tools and frameworks used. Online platforms such as YouTube were very valuable for providing tutorials and guidance on implementing databases and working with the .NET MAUI framework. These resources offered insights that helped me overcome technical challenges and improve different aspects of the app.

In addition to online tutorials, collaboration with others played a significant role in the project. I frequently discussed ideas and technical challenges with Seth Boggess. These conversations provided a similar perspective and helped refine the design and implementation of specific features. The exchange of knowledge contributed to the overall progress and quality of the application.

To speed up the coding process and address issues, I leveraged GitHub Copilot, an AI coding tool that provided suggestions and solutions. GitHub Copilot proved to be a very effective assistant in finding errors, optimizing code, and implementing good practices. Its ability to generate code snippets and recommend improvements greatly reduced development time and ensured better coding practices.

The references from tutorials, collaboration with Seth Boggess, and GitHub Copilot enabled me to tackle the challenges of this project with more confidence and efficiency. By combining these resources with my own creativity and determination, I successfully developed an application that helped me grow as a developer and strengthen my ability to translate a concept into a functional and user friendly product.