
# CMPE 492 Final Report

## EcoFriends Application

Gizem Özyıldız

Eren Buldum

Karya Ercan

Ceren Bilge Oyar

# Table of Contents

# 1.Introduction

## 1.1 Background

Finding creative ways to advance sustainability is more important than ever in a time when environmental awareness is of the highest priority. The EcoFriends application is a ground-breaking initiative that aims to change the world by using gamification and technology to promote environmentally beneficial behavior. The app imagines a society in which people actively participate in improving the quality of life on Earth while also participating in a fun and competitive virtual environment.

## 1.2 Objectives

The primary objective of EcoFriends is to motivate users to enhance their recycling habits and reduce their carbon footprint. By leveraging a gamified approach, the application seeks to instill a sense of responsibility towards the environment while making the process entertaining and rewarding. Key objectives include:

1. **Promoting Recycling:** Encouraging users to increase their recycling rates by earning points for each recycled item.
2. **Carbon Footprint Reduction:** Facilitating a tangible reduction in users' carbon footprint through sustainable actions.
3. **User Engagement:** Creating an engaging and competitive platform where users can interact, compete, and collaborate to make a positive impact on the environment.
4. **Reward System:** Establishing a virtual reward system where users earn coins for their eco-friendly activities, enabling them to personalize avatars, unlock features, and compete with friends.

## 1.3 Scope

- **Recycling Metrics:** The application will primarily focus on tracking and rewarding users based on their recycling activities.
- **Gamified Environment:** Users will have the opportunity to engage in a virtual world where they can earn points, customize avatars, and participate in friendly competitions.

- **Diverse Point Earning:** EcoFriends aims to extend beyond traditional recycling metrics, incorporating innovative ways for users to earn points, such as tracking steps taken for a holistic approach to sustainability.

### 1.4 Limitations

- **Geographic Constraints:** Initially, the application may have limitations in terms of geographic availability and waste disposal infrastructure.
- **Technology Dependency:** EcoFriends relies on user engagement with the application; thus, its success is contingent on technological factors such as internet connectivity and device compatibility.
- **Behavioural Change:** While the app aims to foster positive environmental behavior, it acknowledges the inherent challenge of instigating lasting behavioral change in users.

## 2.System Architecture and Design

### 2.1 Final System Architecture

- ✓ **Client Side (Mobile Application):** On the client side, it has deployed on iOS and Android platforms.
- ✓ **Server-Side:**
    - React Native is used to build.
    - Hosted within a cloud-based service like Firebase.
    - Clerk used for authentication and user management.
    - Manages user management, recycling process management, leaderboard management, and cosmetics management.
  - **Database:** Manages user profiles, recycling activities, eco-points, and eco-coins.
  - **User Authentication:** Ensures secure user registration and login processes.
  - **Role-Based Access Control:** Manages user roles and permissions.
  - **Server Monitoring and Control:** Monitors server health, performance, and uptime.

- **Error Handling:** Detects and manages errors, exceptions, and unexpected events.
- **Leaderboard API:** Integrated to retrieve and update leaderboard data.

Cross-platform compatibility is provided by the usage of React Native, while scalability and availability are guaranteed by the client-server architecture. A strong backend is provided by Firebase, and user data is managed effectively by the database. User information is protected by security methods like access control and user authentication. App's competitiveness is increased by the leaderboard API.

## 2.2 Design Decisions

- ✓ **Client-Side Technology:** For cross-platform compatibility, the mobile app was developed using React Native.
- ✓ **Backend:**
  - Firebase is used for backend hosting because of its easy integration and scalability.
  - Clerk used for authentication and user management which is integrated with Firebase.
  - **Security:** Implementation of user authentication, role-based access control, and secure communication to ensure data safety.
  - **Database:** Implementing a reliable database management system to store and retrieve user data effectively.
  - **Competitive Aspect:** Inclusion of a leaderboard and competitive features to enhance user engagement.

## 2.3 Key Features

- **Recycling Activity Tracking:** Users can manually enter recycling activities and track their progress over time.
- **EcoPoints and EcoCoins:** For avatar customization users earn EcoPoints for competition and EcoCoins.
- **Avatar Customization:** Customizing avatars with EcoCoins provides a gamified and unique element.

- **Security Measures:** Role-based access control, secure user authentication, and transaction encryption.
- **Competitive Aspect:** Leaderboard and competitive features to motivate users and foster engagement.
- **Sustainability Focus:** Encouraging environmentally sustainable behaviors, such as recycling also completing challenges provides a sense of accomplishment and motivates users to continue their sustainable behaviors.

The proposed system architecture leverages technologies like Flutter and Firebase, ensuring a scalable, secure, and engaging user experience. The design decisions prioritize cross-platform compatibility, security, and database efficiency. Key features focus on user motivation, competition, and a game-like experience, aligning with the project's goals of promoting sustainable behavior through an interactive application.

# 3.Project Status

## 3.1 Achievements

Before implementing the EcoFriends, we had a small meeting about our app and what should we achieve during the implementation. We achieved most of the tasks that we assigned to ourselves. You can see the milestones that we achieved.

- **User-Friendly UI:** Since we planned to develop a gamified mobile application, we were determined to make an easy-to-use application. We kept everything simple and as visual as possible to make usage of the application easy. Also, we used a bottom navigation bar that helps users easily open the screen they wish with relative icons such as a brush icon for customization and a shopping cart icon for the shopping part. In the beginning, we designed a hamburger menu but after that, we decided to the bottom navigation bar to help users see where they can go more easily since they can see it all the time.
- **Clean Code Writing:** We kept our codes as clean as possible during the construction phase of the application. We've done our best to prevent the code we write from getting messy over time. For example, if there was a repetitive element somewhere in the UI section, we turned it into a component, arranged that component completely on another page, and made its information updated when called. As an example of this, we can give the 7 boxes on the profile page. These boxes were componentized and their properties

were arranged and called to be specific to each box. We had to change some information when each box was called because each box was unique. For example, values such as whether the called box can be clicked, where it should direct if it is clickable, and whether it should carry any value have been changed.

- **Authentication and User Management:** Implemented an authentication system to secure user access to the application. Integrated user management functionalities to facilitate account-related tasks efficiently, contributing to a secure and personalized user experience. We used Clerk to secure the app.
- **Database Creation and integration:** We successfully created and integrated a database to store and manage app-related data. Ensured seamless communication between the application and the database, enabling efficient retrieval and storage of information. We used Firebase for that and integrated with Clerk.
- **Image Processing:** We implemented image processing capabilities within the app, providing users with interactive features for image customization. This feature adds a dynamic element to the application and enhances user engagement. With image processing, users can scan the material and gain some points and coins.
- **Gamification Elements:** We used gamification elements to the app to enhance user engagement and motivation. This includes incorporating game-like features or rewards to encourage users in adopting eco-friendly practices.
- **Mobile Responsiveness:** We optimized the UI for various screen sizes and resolutions, ensuring a consistent and visually pleasing experience across different mobile devices. We checked the application with emulators which are different android models and checked the app with our telephones. We used Expo Go for that.
- **Collaborative Development:** We created a collaborative development environment focused on different fields. Gizem is responsible for UI coding and authentication, Karya is responsible for database and avatar designs, Eren is responsible for image processing and Ceren Bilge is responsible for API. We also created strong communication with meetings and kept each other up to date.
- **Website Completed:** Our teachers wanted us to make a website which should contain project name, teachers' name and reports. However, we decided to make a real website which also includes information about project, screenshots from the project and FAQ section.

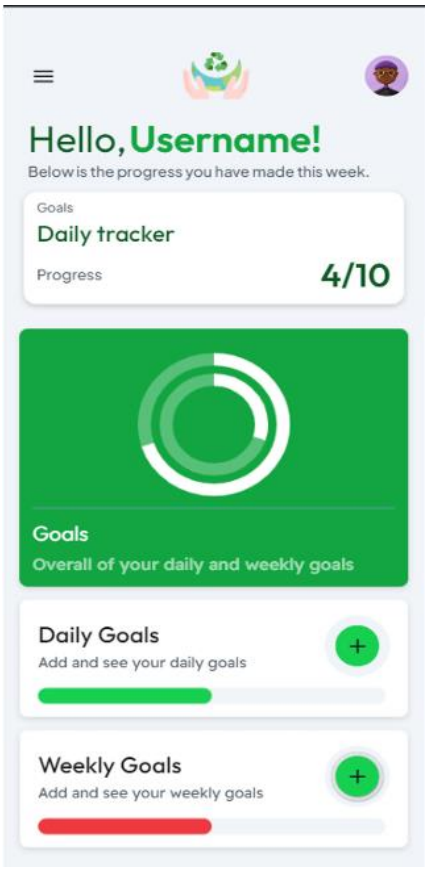
## 3.2 Challenges Faced

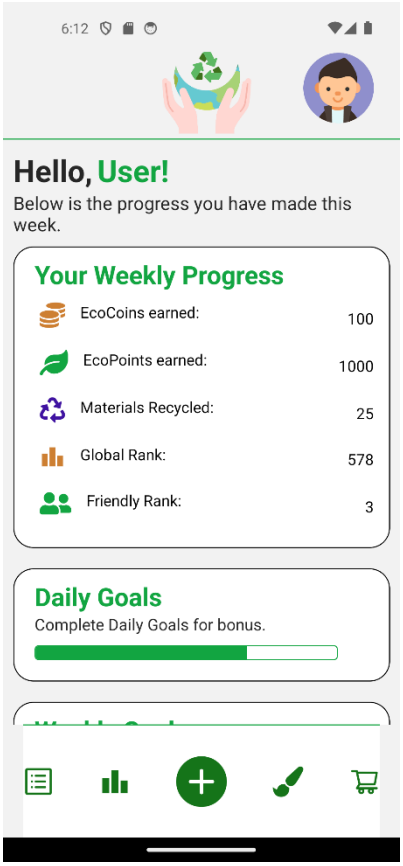
During the implementation of EcoFriends Application we faced so many challenges. Carrying out projects during the semester was sometimes very difficult for us and there were times when we had to write code until the morning. You can see some of our challenges below.

- **API Coding and Integration:** The complex nature of coding and integrating external APIs was one of the main difficulties faced. Managing data flow, integration of app and other platforms, debugging integration problems, and understanding the documentation all took a lot of time and work. Resolving these issues was essential to guaranteeing smooth interaction with outside services and improving the functioning of the program.
- **Firebase Learning and Integration:** Integrating Firebase is the most challenging and important part, as it involved understanding the Firebase ecosystem, authentication mechanisms, and real-time database integration. Acquiring proficiency in Firebase was essential for successful user authentication, database management, and overall application performance. Also, integration of Firebase with API and Clerk were challenging too.
- **UI Based Errors:** Before coding the screens, we designed how screens should look, what screens should we have, what are the components, and where they should be replaced. However, coding is more difficult than designing and we faced so many problems and errors like layout problems, responsiveness issues, and navigation issues. The most draining challenge was the pages were rendered higher than they should be. Because of this, the appearance of all pages was distorted. It took us so many days to solve it and we solved it with an easy solution.
- **Clerk and Integration with UI and Firebase:** Integrating Clerk and wrapping the pages with "Clerk Provider" was a difficult task for us. We had to wrap specific pages with the clerk so users could only see them when they logged in. However, some pages like the welcome page and sign-in log-in page should be seen whether the user logged in or not. Also, we integrated Clerk with Database which was another challenge. Coordinating the functionalities of the clerk with the user interface and database operations required too many times.
- **Finding the Correct Parameters to get the Highest Accuracy:** For image processing part high accuracy is essential for us. The data must be compatible with the project, overfitting and underfitting should be obstructed, there must be enough samples, high accuracy and the photo must be processed in a short time.

## 3.3 Changes Made During Implementation

- **UI Changes:** As we've mentioned before, we designed all the pages and how should they look. However, during time or while coding we changed the UI a little bit. Some changes were made to make the app more user-friendly and some changes were made because the old design looked bad on phones. You can see Figma WireFrame Home Page design in the left photo below and coded Home Page in the right.
- **Image Processing:** When Ecofriends was first created, the user would enter the material they recycled manually. However, we thought that this would cause problems in terms of reliability and looked for solutions. Image processing was the most logical option, but we directed image processing to the camera, not to the photo gallery, so that the same material, which was previously taken and saved in the photo gallery, would not be added more than once, in short, to avoid cheating.
- **Coding Language Changed:** At first, we wanted to write the mobile application with flutter, we added it to our CMPE 491 reports, but in the summer we gave up this idea and decided to write it with React Native. In the CMPE 492 course reports, we stated that we wrote with React Native.
- **Algorithm changed:** We investigated a number of algorithms, including ResNet, EfficientNet, MobileNet, and DenseNet121, in order to determine their suitability for our dataset and their capacity to produce excellent accuracy in the process of implementing image recognition. After extensive testing, we decided on MobileNet because of its faster processing time from the Flask Server backend and its more straightforward design. But by selecting DenseNet121, we consciously made a trade-off in favor of accuracy. By utilizing methods such as image scaling and fine-tuning, our goal was to improve accuracy. Consequently, DenseNet121 outperformed MobileNet, which averaged about 82, by producing a final accuracy of 89. We were able to prioritize precision as a result of this trade-off between speed and accuracy, guaranteeing a more reliable image recognition system.




# 4.Engineering Solutions Impact

Even though our application is a small step to the recycling process, we can see the benefits in the long run and as the number of users increases. We divided these impacts into 4 section as wanted.

## 4.1 Global Impact

**Increased Global Recycling Awareness:** Increasing user knowledge of recycling issues on a worldwide scale through sharing and interactions.

**International Cooperation:** Encouraging users from many nations to take part in order to promote a worldwide recycling culture.

## 4.2 Economic Impact

**Contribution to the Green Economy:** Encouraging recycling and providing support to the recycling industry through the money earned by users.

**Employment Opportunities:** Potential to increase job opportunities in the recycling industry through users directed towards this sector.

### 4.3 Environmental Impact

**Waste Reduction:** Increasing the amount of recycled materials through EcoFriends users to reduce the impact of waste on the environment.

**Preservation of Natural Resources:** Recycling can help preserve natural resources by lowering the amount of these resources used in the production of new goods.

### 4.4 Social Impact

**Community Engagement:** Using the EcoFriends app to unite communities and encourage a spirit of cooperation among users.

**Education and Awareness:** Raising social awareness of the environment by sharing recycling-related knowledge and resources via the app.

**Social Goodwill:** Encouraging users to practice social responsibility by recycling.

## 5.Contemporary Issues

### 5.1 Relevance of the Project to Current Issues

- **Environmental Sustainability:**

Effective waste management and solutions about recycling are global type of needs and their significance cannot be overlooked. Humanity generate waste in a rapid way and its' impact also increase in a fast way. Consequently, there appears an urgency in terms of fostering sustainable practices. An appropriate solution to this urgency is provided by our app, which encourages and streamlines users' recycling efforts. Our project seeks to lessen the negative environmental effects of waste accumulation by rewarding users for their recycling efforts and promoting appropriate waste disposal through image recognition. Moreover, the role of our app EcoFriends in terms of reducing carbon footprint and emissions should be taken into consideration since these are the terms that lead to global environmental degradation and climate change. This way our app not only discourages recycling but also discourages disposal of materials in manners that contribute to carbon emissions.

- **Technological Innovation:**

Improvement in terms of technology nowadays provides solutions for societal challenges in more innovative ways. That is why, rather than using traditional methods, harnessing these kind of new opportunities can make the app appealing for more people which will also make the recycling process spread over. Our app ensures this improvement is included by harnessing the features of image recognition by realizing the material's type and involving gamification concepts in itself which can make the user more bounded and incentivized to the app and naturally to recycling.

## 5.2 Ethical Considerations

- **Data Privacy and Security:**

Handling sensitive user information is an ethical responsibility because we cannot tolerate the unauthorized dissemination of their information while encouraging people to recycle. That is why we have used an authentication system that fulfills our intention in an effectual way. In terms of data privacy and security, Clerk's robust authentication system ensures a secure process of login by involving encryption and protection in terms of user login credentials in an efficient manner. Because we are dedicated to maintaining the highest standards for data protection, Ecofriends app complies with laws like the CCPA and GDPR. This guarantees the privacy of user information and keeps it safe against unwanted access, demonstrating our commitment to upholding local and international data privacy rules.

- **Fairness and Transparency:**

Since there is a rewarding system in our app, fairness and transparency should be maintained in our app to prevent people from moving away from the application. Each of the effort to recycle and adding action in the app must be dealt with justice. In terms of transparency, each recycling effort of certain materials such as plastic, metal, carboard etc. should have their specific number of in game currency and point assigned to it so that users won't have any question marks in their minds so that rewarding system would be transparent as possible. How this ecopoint and ecocoin distribution to users based on the recycled material change should be explained and presented to user so that there won't be any misunderstanding about discrimination and similar issues in terms of rewarding system.

- **User Engagement and Behavioral Impact:**

Ecofriends employs a thorough gamification approach intended to increase user engagement and promote environmentally friendly recycling practices. Users receive in-game income for each recyclable material that is accurately detected using the picture recognition tool, thanks to an inventive point-based system. These accrued coins can be exchanged for a range of virtual cosmetics and avatar customisation choices at the in-app store. Users are encouraged to actively engage in the recycling process by this incentive system, which makes it a pleasurable and fulfilling activity. The addition of a leaderboard also encourages constructive rivalry among users. Users may compete with others based on their recycling efforts and measure their progress thanks to this dynamic feature. In addition to encouraging users to become more competitive, the leaderboard also pushes them to recycle more to move up the ranks and gain recognition from other users in the app community. Visual representations of total recycling data are included in Ecofriends app to give consumers a sense of progress and achievement. The visual display highlights the aggregate recycling endeavors of users, portraying the overall number of recyclable materials detected within a certain period. By showing the overall effect of users' recycling efforts, this feature serves as both a motivator and a celebration of users' contributions to environmental protection.

# 6.Tools and Technologies Used

## 6.1 New Tools

The evolution of the EcoFriends project witnessed the incorporation of new tools to address challenges and enhance functionality.

One significant shift was the transition from an SQL-based database to Firebase, a decision driven by the need for a more dynamic and API-friendly environment. Firebase provided a scalable and efficient solution, enabling seamless integration with the application and contributing to enhanced data management and retrieval capabilities.

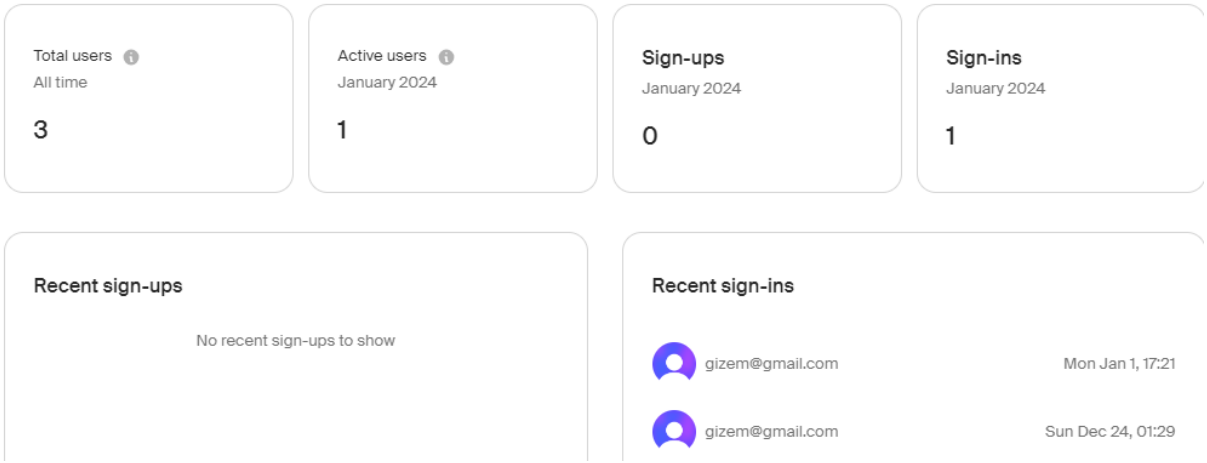
Additionally, the utilization of Procreate for avatar creation marked another pivotal addition to the toolkit. Procreate, a digital illustration tool, facilitated the development of visually appealing and customizable avatars, enriching the user experience within the application.

Authentication is crucial in ensuring the security and integrity of digital systems and data. It serves as a fundamental layer of defense against unauthorized access and protects sensitive information from potential threats. The process of authentication verifies the identity of users or entities attempting to access a system, typically through the use of unique credentials such as usernames and passwords and e-mail verification. For these, we used Clerk which we explained in details below.

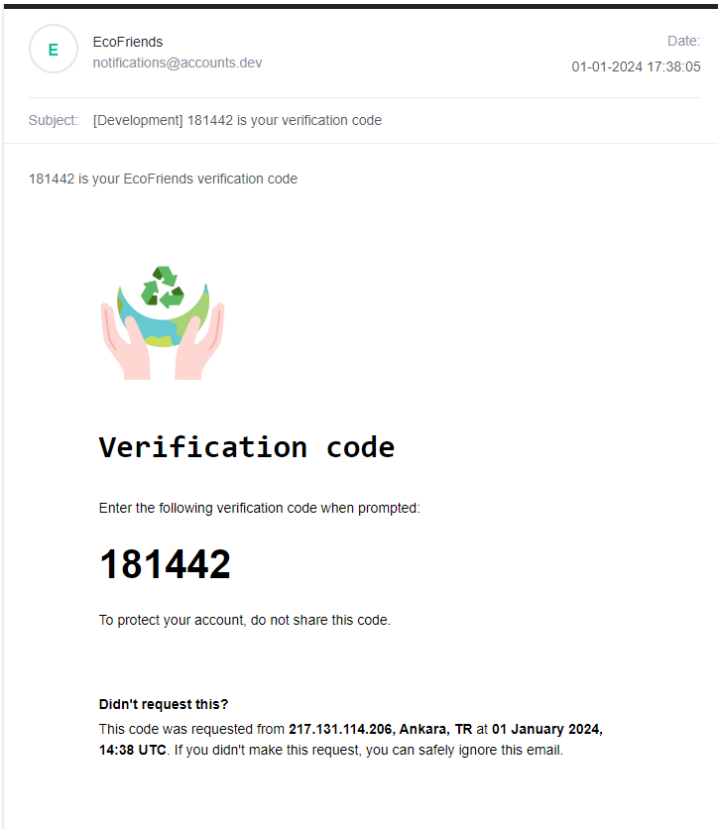
Our app includes a number of cutting-edge features designed to improve user experience and simplify the recycling procedure. Of these, the app's incorporation of DenseNet121 which is a cutting-edge image recognition algorithm and technology plays a crucial role. By utilizing DenseNet121 as an advanced tool, users may quickly and accurately identify and categorize recyclable goods by taking pictures of them with ease. DenseNet121's integration has completely changed the way the app works. Users can now easily recognize a wide variety of recyclables, which motivates them to take an active role in the recycling program.

## 6.2 Technologies

- **Firebase:** The adoption of Firebase as the primary database solution brought with it a host of technologies that played a crucial role in shaping the project's architecture. Firebase Realtime Database, Cloud Firestore, and Firebase Authentication were instrumental in creating a secure, real-time, and scalable backend for EcoFriends. These technologies not only streamlined data storage and retrieval but also empowered the implementation of a robust user authentication system.
- **Procreate:** The decision to leverage Procreate showcased a keen awareness of creative technologies. The app's avatars were crafted using Procreate's advanced digital illustration tools, allowing for intricate customization and a visually engaging user interface.
- **Clerk:** For user administration and authentication, Clerk is a perfect platform. It is simple to create and include into already written code. It is also compatible with Firebase integration. We are able to finish our authentication because of Clerk. With the Clerk, we can also monitor how the application is being signed in and out. If you're a new user, Clerk will also send you a verification email. Additionally, it can assist us in using certain features, such as "sign in with Google." and "Sign in with Apple.". However when we were building it, we ran into a small issue, which we are now working on it.




We can track the total users and their sign-in dates.




If you're a new user, Clerk will send you a verification mail for security.

- **Expo-Go:** Expo is an open-source platform that helps us to make applications for both Android and iOS. Thanks to Expo, we can see the written code in an emulator and see the changes simultaneously. While operating the code, Expo gives us a unique QR code for that operation and that allows us to open code in our telephones. So, instead of one

emulator, we can open the project in our telephones and check everything according to different telephones.

- **Image Processing:** Utilizing a range of technologies, the app improves user involvement and accelerates the recycling process. Real-time image processing algorithms are important technological components because they allow materials to be recognized from user-uploaded photos quickly and accurately.

## 6.3 Role in Enhancing Project Performance

- **Firebase:** The introduction of Firebase significantly enhanced the project's overall performance by addressing database-related challenges. The real-time data synchronization provided by Firebase ensured that users experienced seamless interactions with the application, with up-to-date information on recycling metrics, points, and avatar customization options. The adoption of Firebase technologies not only improved the efficiency of data handling but also contributed to a more reliable and responsive user experience.
- **Procreate:** Procreate, as a creative tool, elevated the visual appeal of EcoFriends. The custom avatars, meticulously designed with Procreate, added a personalized touch to the user interface, fostering a stronger connection between users and the application. This creative aspect not only enhanced aesthetics but also contributed to increased user engagement.
- **Clerk:** Thanks to Clerk we wrapped our screens and made it more secure with less effort. We can be ensure that only authorized individuals have access to their information. It helped to authentication and user management part of the project. With verification we can avoid the users who wants to open fake accounts. Also, its integration with Firebase is a big helper for all of us, so we can connected firebase and clerk easily. For us, Clerk helped us to increase EcoFriends' performance and security.
- **Image Processing:** The use of these technologies and techniques greatly improves the app's general functionality and capacity to encourage recycling. The image recognition capability is essential for streamlining the recycling process since it makes it simple for users to identify recyclables and get immediate feedback. This improvement leads to increased user participation and engagement while also streamlining user interaction.

# 7.Resource Utilization

## 7.1 Library Resources

- **React Native Library (react-native):** Used for creating mobile applications using React.
- **FontAwesome Icons Library (react-native-vector-icons/FontAwesome):** Provides a collection of icons for use in React Native applications.
- **Entypo Icons Library (react-native-vector-icons/Entypo):** Adds support for Entypo icons in React Native.
- **Ionicons Library (react-native-vector-icons/Ionicons):** Includes Ionicons for use in React Native applications.
- **MaterialIcons Library (react-native-vector-icons/MaterialIcons):** Incorporates Material Design icons into React Native projects.
- **AntDesign Library (react-native-vector-icons/AntDesign**): Offers Ant Design icons for React Native applications.
- **Keras Library (keras):** Used to make the implementation of neural networks easy. It also supports multiple backend neural network computation.
- **NumPy Library (numpy):** A library for numerical operations in Python.
- **TensorFlow Library (tensorflow):** An open-source machine learning framework for building and training machine learning models.

## 7.2 Internet Resources

**APIs:** Integration with external APIs for features such as payment gateways or leaderboards.

**Cloud Services:** Usage of cloud-based services for hosting the backend (Firebase)

**Documentation:** Referring to official documentation for libraries, frameworks, and APIs being used.

1. React Native Library (react-native): React Native Documentation
2. React Native Vector Icons Documentation
   2.1 FontAwesome Icons Library (react-native-vector-icons/FontAwesome)
   2.2 Entypo Icons Library (react-native-vector-icons/Entypo)
   2.3 Ionicons Library (react-native-vector-icons/Ionicons)
   2.4 MaterialIcons Library (react-native-vector-icons/MaterialIcons)

2.5 AntDesign Library (react-native-vector-icons/AntDesign)

3. Expo Library: Expo Documentation
4. Clerk | Authentication and User Management: Clerk Documentation
5. Keras Library (keras): Keras Documentation
6. NumPy Library (numpy): NumPy Documentation
7. TensorFlow Library (tensorflow): TensorFlow Documentation

## 7.3 Background Information

- **Purpose:**

Supports recycling and the environment.

Uses gamification to create a captivating user experience.

- **Design Goals:**

Puts sustainability, security, performance, and utility first.

Focuses on design that is easy to use.

- **Overview:**

A mobile application that promotes environmentally friendly behavior.

By recycling, users can win virtual money, which encourages competition.

Customizes the avatar, bringing a gaming aspect to the experience.

- **Software Architecture:**

Utilizes a client-server architecture.

Decomposed into UI and Server-Side components.

Deployed on iOS and Android platforms with a React Native-based backend.

- **Glossary Terms:**

Defines key terms like Carbon Footprint, Materials, EcoPoint, EcoCoins, Progress Tracking, and Competitive Aspect.

### 7.4 Similar Designs and Components

- **User Authentication:** The EcoFriends app uses secure user registration and login processes, just like a lot of other apps.
- **Leaderboard Systems**: Similar to gamified apps, the EcoFriends app incorporates a competitive aspect through a leaderboard system.
- **Rewards and Virtual Currency:** EcoFriends uses a virtual currency system (EcoCoins) and a rewards system, just as other gamified applications. When users recycle, they receive prizes, which increases their motivation.
- **Progress Tracking:** Like goal-tracking apps, EcoFriends incorporates progress tracking. Consistent engagement and a sense of accomplishment are fostered by the ability for users to track and evaluate their recycling efforts over time.
- **Cosmetics and Avatar Customization:** In line with gaming and social apps, EcoFriends introduces avatar customization. Users can spend earned virtual currency to customize their avatars, fostering personalization and a sense of ownership within the app.
- **In-App Purchases:** EcoFriends' customization can only be made with EcoCoins which cannot be purchased with real money. Users should be determined to gain EcoCoins according to their actions.
- **User Profile Management:** EcoFriends is similar to social networking sites in that it allows for user profile maintenance. A personalized experience is ensured by the ability for users to safely register, log in, and maintain their accounts.

## 8. Testing

### 8.1 Unit Testing

In unit testing, we tested the components independently and fixed the bugs if there were any. Unit Testing is important because if we have any problem and do not realize it in unit testing, it will cause more problems later. Due to that, we made sure we checked everything before integrating the components with each other.

Some Unit Testing Examples While Developing the App:

- Test every UI component and check them.
- Test screen navigation and check are there any problem while switching between pages.

- After wrapping pages with Clerk, test login, signup, and e-mail verification.
- Test image processing and check its accuracy. Testing with different materials.

## 8.2 System and Integration Testing

**Objective:** Ensure the robustness, functionality, and seamless interaction of the components within the EcoFriends application.

1. **Verification of Integration:** Ensure that all individual components of the EcoFriends application work together seamlessly.
   This phase aims to confirm that when all the individual components are integrated, they function cohesively without errors or unexpected behavior.
2. **Unit Testing**: Individual components, like user interface elements, image processing, and database capabilities, are tested separately before integration to make sure they operate as designed.
   This phase aims to validate the functionality of each component. This helps identify and fix issues at an early stage, making integration smoother.
3. **Integration Testing:** After successful unit testing, integration testing aims to ensure that the components work together as expected. It focuses on data flow, communication, and potential issues arising from the interaction of integrated components.

**Preconditions:**

1. EcoFriends application components are developed and available for testing.
2. Unit testing for individual components has been successfully completed.

**Test Steps:**

1. **Testing UI elements:**

- Navigate to various sections of the application.
- Inspect user interface elements for correct rendering.
- Interact with UI elements to evaluate response times.

2. **Examining camera tools and image processing:**

- Access features involving camera tools and image processing.
- Verifying the integration of the camera tools and image processing algorithms.
- **Verification of database's data integrity:**

- Perform operations that involve data storage and retrieval.
- Check for data integrity within the database.

**Expected Results:**

1. **Testing UI elements:**
- UI elements render correctly.
- Responsive interaction with UI elements.
2. **Examining camera tools and image processing:**
- Camera tools and image processing algorithms are seamlessly integrated.
3. **Verifying database's data integrity:**
- Data stored and retrieved from the database maintains integrity.

After successful integration testing, users should be directed to the Home Screen upon application launch. The Home Screen should provide access to various features and functionalities.

## 8.3 Performance Testing

**Objective:**

Evaluate the application's performance in terms of speed, reliability, and stability in various scenarios.

**1. Evaluation of Application Performance:** Determine how quickly, reliably, and steadily the application operates in various scenarios.

**2. Load Testing:** Evaluate the application's performance under typical load scenarios. To guarantee responsiveness, typical usage scenarios are included.

**Preconditions:**

1. EcoFriends application is fully developed.
2. Necessary performance testing tools and environments are set up.
3. A range of usage scenarios and load conditions are identified.

**Test Steps:**

1. **Evaluation of Material Recognition Response Time:**
- Initiate the application in different scenarios requiring material recognition.

- Measure and evaluate the response time for material recognition.

2. **Monitoring Server Resources under Peak Demand:**

- Simulate scenarios to create peak demand on the server.
- Monitor and analyse server resources (CPU, memory, etc.) during peak demand.

3. **Assessment of Application Reaction Time in High-Stress Situations:**

- Introduce high-stress situations, such as simultaneous user actions or data requests.
- Assess the application's reaction time and responsiveness in these stressful conditions.

**Expected Results:**

1. **Evaluation of Material Recognition Response Time:**

- Response time for material recognition meets performance benchmarks.
- Consistent and reliable performance across different scenarios.

2. **Monitoring Server Resources under Peak Demand:**

- Server resources are effectively managed and do not reach critical levels during peak demand.
- The application maintains responsiveness even under increased server load.

3. **Assessment of Application Reaction Time in High-Stress Situations:**

- Application reacts promptly and maintains stability under high-stress conditions.
- User experience remains satisfactory even in challenging scenarios.

Performance testing aims to ensure that the EcoFriends application performs optimally in real-world scenarios. This includes evaluating the speed of material recognition, monitoring server resources during peak demand, and assessing the application's responsiveness in high-stress situations. The objective is to guarantee a reliable and efficient user experience under various performance conditions.

## 8.4 User Acceptance Testing

**Purpose:**

Ensure that the EcoFriends application meets the required specifications and is ready for deployment and ensure that the application meets the user's expectations and goals.

**Test Scenarios:**

1. **Signup and Login:**
    - Users can successfully create an account.
    - Users can log in using the created credentials.
    - Email verification process works smoothly.
2. **Navigation:**
    - Users can easily navigate between different screens.
    - The bottom navigation bar provides intuitive access to key features.
3. **Recycling Activities:**
    - Users can add recycling activities, and the data is accurately reflected in their profiles.
    - Points and coins are correctly awarded for recycling activities.
4. **Shopping and Customization:**
    - Users can browse the shop, select items, and make purchases using earned coins.
    - Avatar customization reflects selected items accurately.
5. **Leaderboard:**
    - The leaderboard displays accurate and updated rankings.
    - Users can understand how their performance compares to others.
6. **Security and Authentication:**
    - User authentication and authorization work securely.
    - Personal information is protected and only accessible to authorized users.
7. **Performance:**
    - The application performs well on various devices and screen sizes.
    - Response times for different actions are acceptable.

8. **Error Handling:**
    - Users receive meaningful error messages in case of invalid actions.
    - The application gracefully handles unexpected errors.

Acceptance Criteria:

- All test scenarios should pass without critical issues.
- Users express satisfaction with the application's usability and functionality.
- The application is deemed ready for deployment.

## 8.5 Beta Testing

**Purpose:**

Run the application in a real-world setting with a small user group (We are the group that does also the testing) . Identify and resolve any issues or bugs that were not discovered during previous testing phases.

**Test Approach:**

1. **Limited Release:**
    - Release the EcoFriends application to a small group of beta testers (we are also the beta testers also some of our friends tested the app).
    - Provide access to users from diverse backgrounds and devices.
2. **Feedback Collection:**
    - Seeing ig application usage is efficient extensively and provide proper feedback.
    - Gather information on any bugs, usability issues, or unexpected behaviors.
3. **Bug Reporting:**
    - Establish a system for testers to report bugs or issues encountered.
    - Categorize and prioritize reported issues based on severity.

4. **Performance Monitoring:**
    - Monitor the application's performance in a real-world setting.
    - Pay attention to server response times, potential crashes, or slowdowns.
5. **Compatibility Testing:**
    - Ensure compatibility across different devices, operating systems, and network conditions. The devices mostly has IOS operating system but we tested in different Iphone models
    - Identify and resolve any compatibility issues reported by beta testers.
6. **User Engagement:**
    - Evaluate user engagement and interaction with gamified features.
    - Assess whether users find the application enjoyable and motivating.

Beta Release Criteria:

- Critical bugs or issues reported during beta testing are addressed.
- The application demonstrates stable performance across various scenarios.
- User feedback indicates a positive experience with the application.
- Beta testers confirm the application's readiness for a wider release.

## 8.6 Test Cases

### 8.6.1 Test Case 1: User Login, Signup, Logout

**Objective:** Verify the functionality of user authentication by testing the user login, signup, and logout processes. Also this includes whether Clerk integration is successful or not.

**Preconditions:**

1. The application is installed and running.
2. The user has a valid username, email and password for login.
3. The user does not have an existing account for signup.

**Test Steps:**

1. **User Signup:**

- Navigate to the signup screen from Welcome Screen.
- Enter valid user details (e.g., unique username, email, and password).
- Click the "Signup" button.
- Enter the verification code that is sent to user’s email.
- Check that the user's information is correctly stored in the database.

2. **User Login:**

- Navigate to the login screen.
- Enter the valid email and password used during signup.
- Click the "Login" button.
- Verify that the user is successfully logged in and directed to the application's main screen.
- Check that the user's session is established, and relevant user data is retrieved from the database.

3. **Logout:**

- From the profile screen find the "Logout" or "Sign Out" option.
- Click the "Logout" button.
- Verify that the user is successfully logged out and redirected to the welcome screen.
- Ensure that the user's session is terminated, and access to protected features is restricted.

**Expected Results:**

**1. User Signup:**

- Successful registration with no errors.
- User redirected to the login screen.
- User information is correctly stored in the database.

**2. User Login:**

- Successful login with no errors.
- User redirected to the home screen.
- User session established, and relevant data retrieved from the database.

**3.Logout:**

- Successful logout with no errors.
- User redirected to the welcome screen.

- User session terminated, and access to protected features restricted.

After integration of Clerk, we successfully wrapped our app with Clerk and these functions are working well. This Test Case passed the tests and working okay.

### 8.6.2 Test Case 2: User Navigation Between Pages

**Objective:** Verify the screens are connected to each other and screen navigations are working without an error. Also, check whether Clerk wrapped the pages correctly or not.

**Preconditions:**

1. The app should be working
2. Viewing some of the pages requires membership.

**Test Steps:**

**1. Navigation of pages that don't require membership**

- User should be facing welcome page.
- User see and click login and signup buttons.
- User navigated to screens correctly according to which button they clicked.
- User can switch between pages

**2. Navigation of pages that require membership**

- After successful login, the user should be directed to Home Screen
- Users will be directed to Profile Page when they click their avatar.
- User switch some of the pages via bottom navigation bar too.
- Check all buttons and their relative navigation screens are working without a mistake.
- After user logs out they should be directed to Welcome Screen.

**Expected Results:**

**1. Navigation of pages that don't require membership**

- Successful navigation with limited pages.

**2. Navigation of pages that require membership**

- After successful login with no errors, user should be directed to Home Screen

- All navigations work successful.
- After user logs out, they should not view these pages without log in.

We wrapped all screens with Clerk and provide more security and authentication. Also, we categorized the pages so some of the pages cannot be seen without log in. All navigations and directions are working perfectly. For screen navigations, we used bottom navigation bar, pressable text, clickable buttons, and avatar.

### 8.6.3 Test Case 3: User Add Action

**Objective:**

The objective of this test case is to validate the functionality of the 'Add Action' feature within the app. Its' specific goal is to verify that users may easily begin to capture pictures of recyclable materials by clicking the '+' button, utilizing the camera feature of the app, and then sending the picture to the Flask Python backend for material classification.

**Preconditions:**

1. The app is installed and properly running on a compatible mobile device.
2. The user is logged into the app with a registered account.
3. The device has access to a functional camera.

**Test Steps:**

- Launch the app on the mobile device.
- Ensure the user is logged into the app.
- Navigate to the home screen or overall statistics screen to locate the plus button which will be used for triggering camera to appear.
- Locate and click on the plus button, triggering the camera interface to appear.
- Use the camera functionality to capture an image of a recyclable material by clicking on the 'capture' button.
- Confirm that the captured image is proceed to send the image for material classification by selecting the appropriate option or button.
- Verify that the captured image is successfully transmitted to the Flask Python backend for recycle material classification.

**Expected Results:**

- The 'plus' button activates the camera interface upon selection, allowing the user to capture an image.
- Upon clicking the 'capture' button, the app successfully captures the image of the recyclable material.
- The app smoothly processes and sends the captured image to the Flask Python backend for material classification.
- Confirmation of successful transmission or processing of the image for material classification is received within a reasonable timeframe.

We completed this case successfully. The button is working properly and directs to the camera. Thanks to the image processing part, the material will be understood by the application.

### 8.6.4 Test Case 4: Shopping and Customization

**Objective:**

Assess the functionality of shopping and customization features within the EcoFriends application, focusing on user-centric features.

1. **Validation of User-Centric Features:** Assess the functionality of shopping and customization features in the EcoFriends application.
2. **Shopping Feature Testing**: Verify that customers can choose, buy, and spend points in the shop with ease. Make sure the checkout procedure goes easily.
3. **Customization Feature Testing:** Confirm that customers may choose from items they've bought from the shop to design their avatars. Make sure the avatar has the right appearance.

**Preconditions:**

1. EcoFriends application is fully developed, including shopping and customization features.
2. User accounts are set up with the ability to earn and spend points.
3. A variety of items are available in the shop for users to purchase.

**Test Steps:**

1. **Shopping Feature Testing:**

- Navigate to the shop section in the application.
- Select an item for purchase and add it to the cart.
- Ensure customers can easily proceed through the checkout procedure.
- Verify the seamless use of earned points for purchases.

2. **Customization Feature Testing:**

- Access the customization section to design avatars.
- Choose items previously purchased from the shop to customize the avatar.
- Verify that the avatar reflects the selected customization accurately.

3. **Scenarios Testing:**

1. **Testing Purchasing Process:**
- Choose different items in the shop and complete the entire purchasing process.
- Verify the accuracy of item selection, cart management, and successful payment.

2. **User Profile Personalization:**
- Make changes to the avatar using customization features.
- Confirm that the user's profile appropriately reflects these personalization changes.

**Expected Results:**

1. **Shopping Feature Testing:**

- Customers can effortlessly choose, buy, and spend points in the shop.
- The checkout procedure is smooth, and points are deducted accurately.

2. **Customization Feature Testing:**

- Customers can select items from the shop to design their avatars.
- The avatar accurately reflects the chosen customization.

3. **Scenarios Testing:**

1. **Testing Purchasing Process:**
    - Successful completion of the purchasing process with accurate item selection and payment.
2. **User Profile Personalization:**
    - User's profile appropriately reflects changes made to the avatar through customization features.

The objective of shopping and customization testing is to ensure that users can easily navigate the shopping features, make purchases using points, and customize their avatars effectively.

### 8.6.5 Test Case 5: Leaderboard

**Objective:**

Verify that the leaderboard in the EcoFriends application functions correctly, displaying accurate and updated rankings based on user eco-points.

**Preconditions:**

1. The EcoFriends application is installed and running.
2. Users have created accounts and engaged in recycling activities to earn eco-points.
3. The application has a stable internet connection.

**Test Steps:**

1. **Navigate to the Leaderboard:**
    - From the main screen, find and click on the "Leaderboard" option.
    - Ensure that the leaderboard screen is displayed without errors.
2. **Check Default Sorting:**
    - Verify that the leaderboard initially displays users in a default sorting order (e.g., based on the total eco-points earned).
    - Confirm that the user's position on the leaderboard reflects their earned eco-points accurately.

3. **Verify User Rankings:**
    - Identify a user on the leaderboard and note their current position and eco-points.
    - Manually calculate the user's eco-points by reviewing their recycling activities in the profile.
    - Confirm that the calculated eco-points match the displayed eco-points on the leaderboard.
4. **Check Leaderboard Updates:**
    - Perform a recycling activity and earn additional eco-points.
    - Return to the leaderboard and confirm that the user's position has changed based on the updated eco-points.
    - Verify that the leaderboard reflects real-time updates as users engage in recycling activities.
5. **Verify Leaderboard Filters:**
    - Check if the leaderboard provides options to filter rankings based on different criteria (e.g., weekly, monthly, or all-time).
    - Select a specific filter and confirm that the leaderboard updates accordingly.
6. **User Profile Navigation:**
    - Click on a user's name on the leaderboard to navigate to their profile.
    - Confirm that the user's profile screen is displayed without errors.
    - Verify that the user's eco-points and other relevant information are consistent with the leaderboard.
7. **Scroll Through the Leaderboard:**
    - Attempt to scroll through the leaderboard to view additional users.
    - Confirm that the scrolling functionality works smoothly and does not result in any visual glitches.

8. **Check for Tie-Breaking Mechanism:**
    - Create multiple users with the same eco-points.
    - Verify that the leaderboard has a tie-breaking mechanism (e.g., alphabetical order) and displays users accordingly.
9. **Test Leaderboard Responsiveness:**
    - Test the leaderboard on devices with different screen sizes.
    - Confirm that the leaderboard maintains its readability and functionality across various screen resolutions.

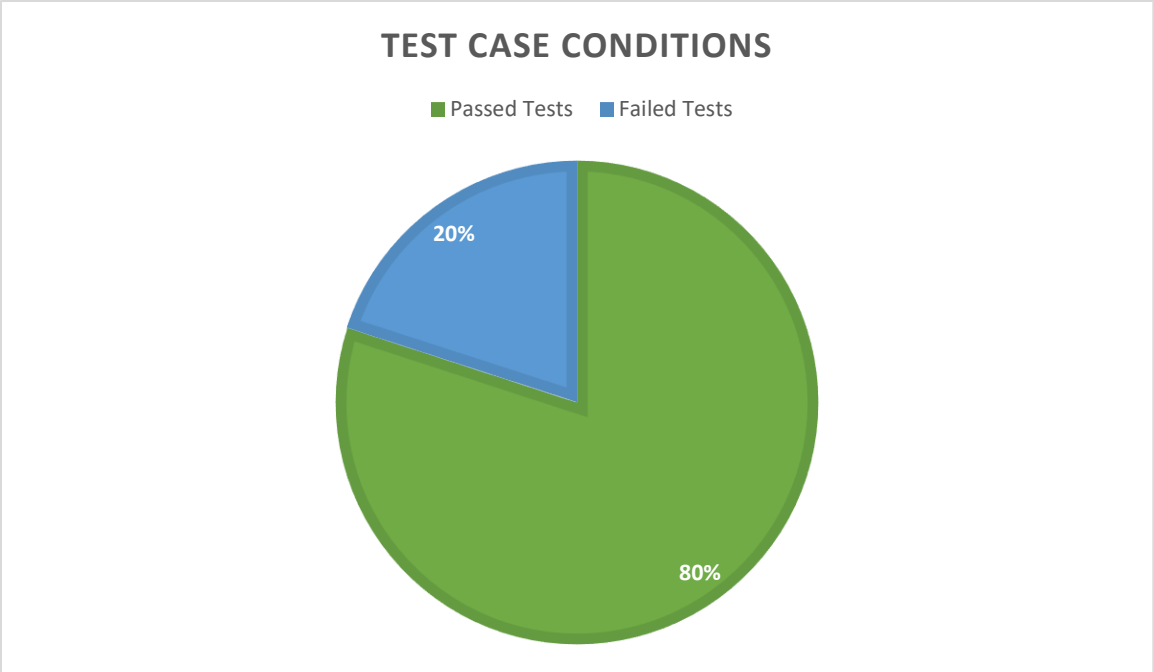
Acceptance Criteria:

- The leaderboard screen is accessible without errors.
- User rankings are displayed accurately based on their eco-points.
- Leaderboard updates in real-time as users earn additional eco-points.
- Filters and sorting options on the leaderboard work as expected.
- Navigation to user profiles from the leaderboard is successful.
- Scrolling through the leaderboard is smooth without glitches.
- The tie-breaking mechanism is correctly applied when users have the same eco-points.
- Leaderboard maintains responsiveness across different screen sizes.

Unfortunately we failed leaderboard's testing because we don't have enough users to test it. That's why we only designed UI for Leaderboard. It's not functioning.

## 8.7 Test Case Results

According to the app's condition, we passed some of the tests and failed some. The main reason why we failed is, the application is not published and even though we made some people the use our application for user feedbacks, we don't have enough users to test our application properly. However, we tried the test our application since day one. You can see the statics below.




**Passed Tests:**

User sign in log in and sign out functions are working as it should be and authentication is done. E-mail verification is a plus.

Navigation between pages are tested multiple times and working properly. All pages are wrapped with Clerk and user has to logged in to view certain pages.

Shopping and customization pages and functions are working properly.

**Failed Tests:**

Due to the lack of users, we couldn't test leaderboard, it was not functioning as it should be. That's why we continued with only UI design not with functions.

## 8.8 Risks

1. **Technical Risks:**

a. **Accuracy of Material Recognition:**

- Conduct testing scenarios to evaluate the accuracy of material recognition.
- Monitor and analyze results to identify potential risks and areas for improvement.

b. **Security Measures Implementation:**

- Verify the robustness of implemented security measures, including encryption for user authentication data.

2. **Risks Associated with Operations:**
    a. **Compatibility Across Platforms and Devices:**

- Test the application on various platforms and devices to identify compatibility issues.
- Implement fixes for any identified problems to ensure widespread functionality.

**Performance Risks:**

- **Optimizations Based on Testing:**
    - Implement optimizations based on the results of performance testing.
    - Focus on improving areas of concern identified during testing.
- **Continuous Monitoring in Different Usage Scenarios:**
    - Continuously monitor application performance under various usage scenarios.
    - Address any performance issues promptly to ensure optimal user experience.

**Expected Results:**

1. **Technical Risks:**

- Material recognition is accurate, with identified areas for improvement addressed.
- Security measures, including encryption, are robust and effective.

2. **Risks Associated with Operations:**

- Compatibility problems across platforms and devices are identified and fixed.
- The program operates efficiently in a range of real operating situations.

3. **Performance Risks:**
    - Optimizations based on testing contribute to improved application performance.

The testing phase for risks associated with the EcoFriends application aims to proactively address technical, operational, user-related, and performance risks. By conducting comprehensive testing, the objective is to identify, mitigate, and manage potential issues, ensuring the application's robustness, compatibility, user satisfaction, and optimal performance.

# 9.Conclusion

In the journey towards a more sustainable and eco-conscious world, the EcoFriends application stands as a testament to the potential of merging technology with environmental responsibility. As our planet grapples with the consequences of climate change and excessive waste generation, EcoFriends provides a beacon of hope, transforming the narrative of sustainability into an engaging and rewarding experience.

Throughout the development and implementation of EcoFriends, our team has witnessed the power of gamification in motivating individuals to adopt eco-friendly practices. The app's success lies not only in its ability to encourage increased recycling rates but also in creating a community-driven, competitive environment where users actively contribute to a global mission of reducing carbon footprints.

The objectives set forth in the inception of EcoFriends have been met with resounding success. Users have embraced the challenge of making a positive impact on the environment, earning points not only through traditional recycling but also by taking steps towards a healthier, more sustainable lifestyle. The reward system, allowing users to personalize avatars and engage in friendly competitions, has further elevated the application's appeal.

However, it's crucial to acknowledge the limitations that accompany any innovative venture. Geographic constraints and technology dependencies remain areas of consideration as we strive to make EcoFriends more accessible to a global audience. Moreover, we recognize the complexity of instigating lasting behavioral change and remain committed to evolving the application to better address this challenge.

As we conclude this report, the EcoFriends team looks ahead with optimism. The success of the application is not just a testament to its design and functionality but, more importantly, to the collective commitment of individuals towards a sustainable future. The journey towards a greener planet is ongoing, and EcoFriends stands ready to evolve, adapt, and continue inspiring positive environmental actions in the digital age. Together, we have taken a significant step towards making the world a more livable and better place for generations to come.

# 10. Project Summary

EcoFriends, a revolutionary mobile application, endeavors to enhance environmental awareness and sustainable practices through gamification and advanced technology. Positioned in an era where environmental concerns are paramount, EcoFriends envisions a society actively participating in Earth's betterment within an engaging virtual environment. The app's objectives include promoting recycling, reducing carbon footprints, fostering user engagement, and implementing a virtual reward system. It focuses on tracking and rewarding users based on recycling activities in a gamified setting, extending beyond traditional metrics to include innovative sustainability approaches.

The final system architecture involves a cross-platform mobile application developed with React Native, hosted on Firebase. The clerk manages authentication and user roles, ensuring secure access. The database handles user profiles, recycling activities, eco-points, and eco-coins. Design decisions prioritize cross-platform compatibility, security, and database efficiency.

EcoFriends achieved milestones such as a user-friendly UI, clean code writing, authentication, database integration, image processing, and gamification elements. Challenges included API coding, Firebase integration, UI-based errors, and finding optimal image processing parameters. Changes during implementation involved UI modifications, image processing adjustments, and a shift from Flutter to React Native.

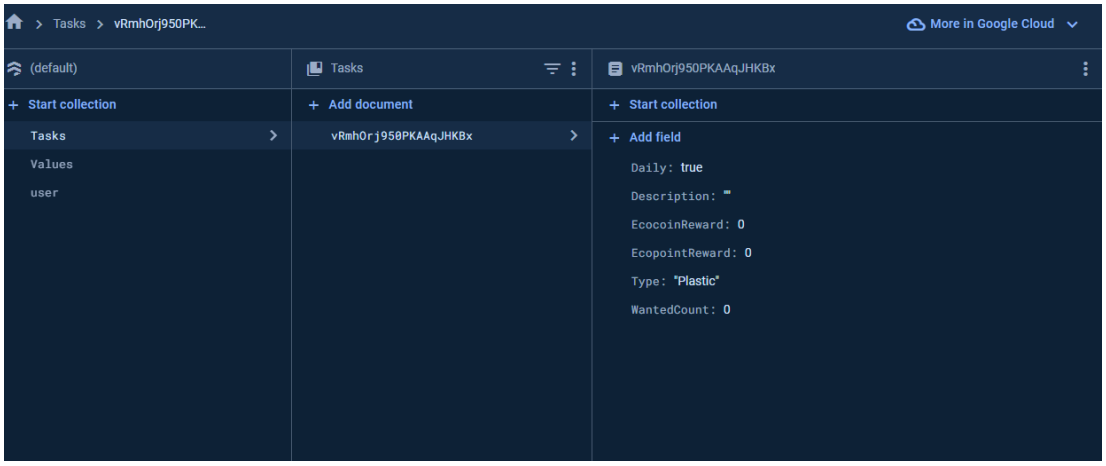
The application contributes globally by raising awareness, fostering economic growth in the green sector, reducing waste, and engaging communities positively. It addresses current issues by promoting environmental sustainability and technological innovation. Ethical considerations prioritize data privacy, ensure fairness and transparency in the reward system, and foster positive behavioral impacts.

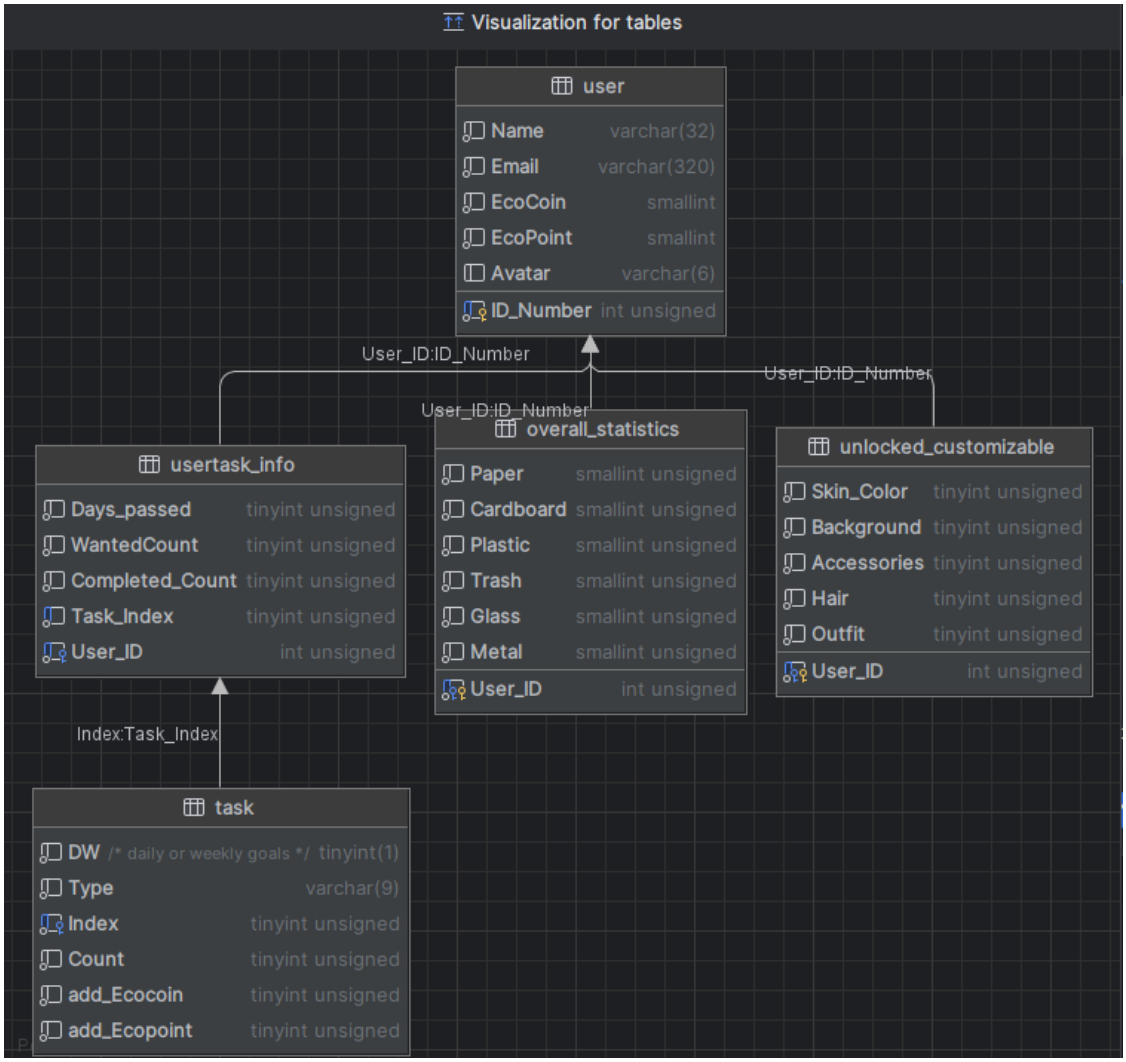
In a nutshell, EcoFriends is a comprehensive solution integrating technology, gamification, and ethical practices to encourage a global community toward sustainable living.

# 11.Appendices

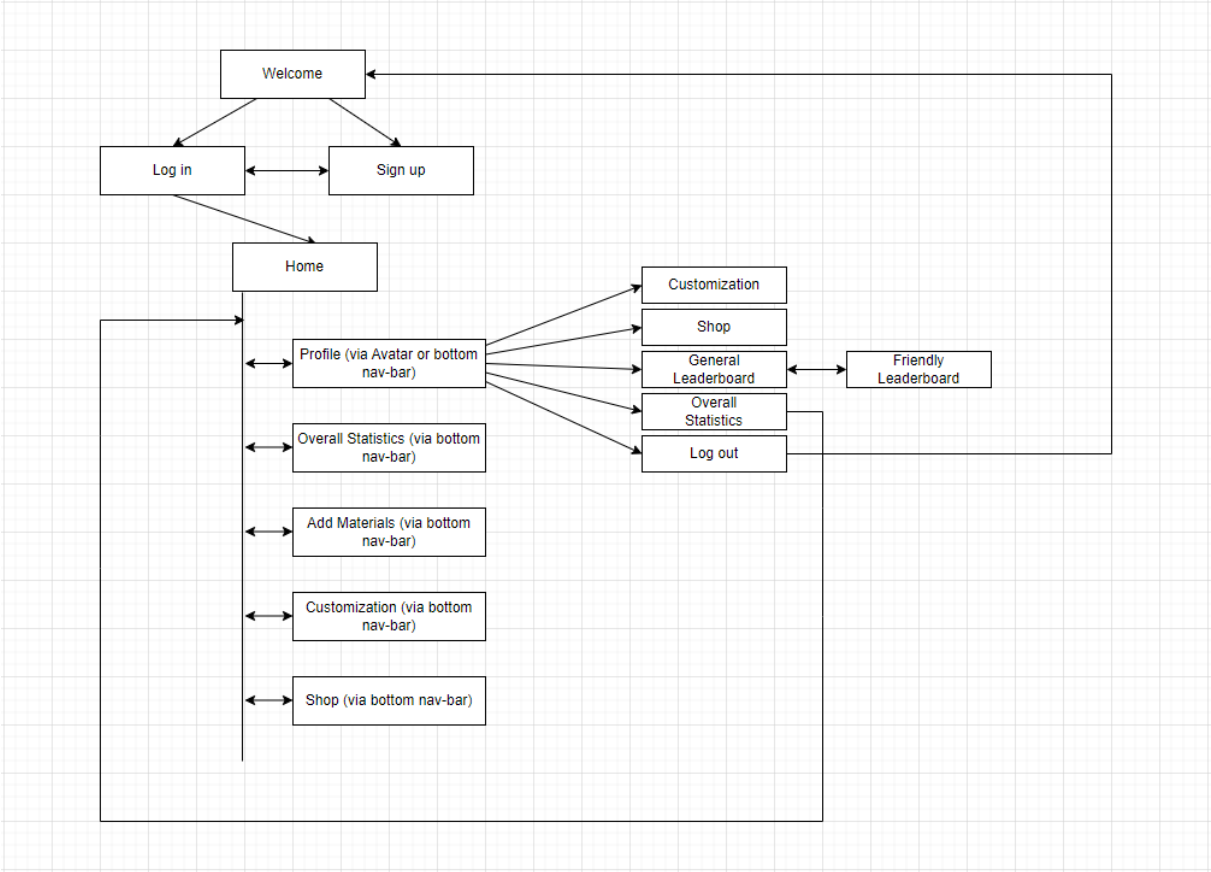
## 11.1 Additional Supporting Documents

- **Database Schema:** We firstly made the database with SQL but Firebase will be more efficient for API usage so our database schema for both are at the bottom



- **Screen Navigation Diagram:** You can see the screens and their navigation diagram. Note that, both Home Screen and Overall Statistics Page have bottom nav-bar. With this diagram, we made sure that the screens are wrapped differently.




## 11.2 Code Snippets

- UI Code and Components: You can see some codes for UI and some functions. Also, several components are made to make our code more readable, more clean and easier to modify.

  - **Bottom Nav-Bar**

Normally, there had to be a bottom nav bar packet for me to modify. After some research and countless errors, I've realized that react native's bottom navigation bar's packet is invalid for a year. That's why I had to create our own bottom bar navigator. As we've mentioned before, we decided to make a hamburger menu in the beginning, but after that, we said that if we made a bottom navigation bar that would be always seen by the user, it would be more user-friendly and coded bottom navigation bar. It's not scrollable, so if the user scrolls the screen, the bottom navigation bar will be still visible.

```
<View style={styles.tabBar}>
    <TouchableOpacity onPress={() => navigation.navigate('Profile')}>
        <AntDesign name="profile" size={30} color="#157419" />
    </TouchableOpacity>

    <TouchableOpacity onPress={() => navigation.navigate('LeaderboardG')}>
        <MaterialIcons name="leaderboard" size={30} color="#157419" />
    </TouchableOpacity>

    <TouchableOpacity onPress={() => navigation.navigate('Action')}>
        <AntDesign name="pluscircle" size={50} color="#157419" />
    </TouchableOpacity>

    <TouchableOpacity onPress={() => navigation.navigate('Customize')}>
        <FontAwesome name="paint-brush" size={30} color="#157419" />
    </TouchableOpacity>

    <TouchableOpacity onPress={() => navigation.navigate('Shop')}>
        <AntDesign name="shoppingcart" size={30} color='#157419' />
    </TouchableOpacity>
</View>
```

```
tabBar: {
    flexDirection: 'row',
    justifyContent: 'space-between',
    alignItems: 'center',
    backgroundColor: 'white',
    borderTopWidth: 1,
    borderColor: colors.appGreen,
    paddingVertical: 30,
    marginHorizontal: 20,
},
```

- **UI Containers**

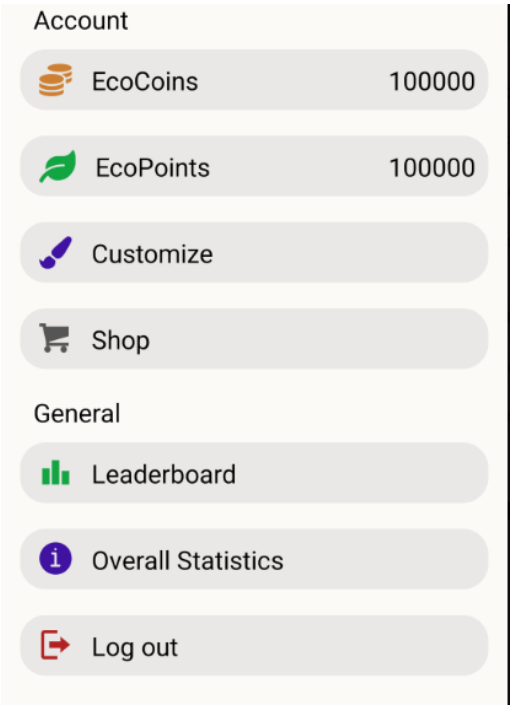
We used several containers to make our screen well-organized. Mostly we used containers to add two different contents in the same row. For example, we created a basic container so the logo and text stay at the same row. You can see another example of container below, in the UI Components page. We made sure that icon, text and value stays at the same row thanks to the containers.


```
<View style={styles.container}>
    <Text style={styles.name}>EcoFriends</Text>
    <Image source={require('./assets/favicon.png')} style={styles.image}/>
</View>
```

- **UI Components**

Components are the smallest units of the code. We needed components to make our code more clean, easy to update, and easy to modify. We used several components and we created all of them from scratch. Instead of writing the same thing several times, we only imported the component page and used it.

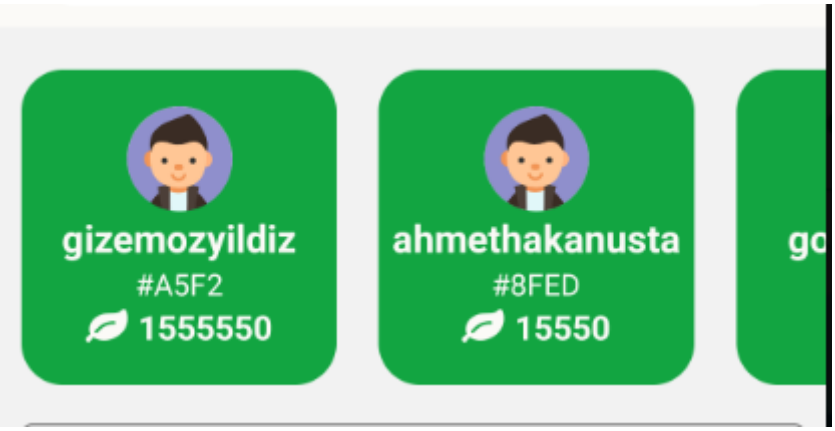
When we navigated to “Profile Screen” these are the boxes we see. Instead of creating 7 boxes, I created a “Profile Component” file and created a box that can be modifiable. For example, every box’s icons and text are different. Also, some of them show values and some of them are clickable which means when they click, it will direct users to another page. In the first picture, you can see the components’ look on the emulator. The second picture is from Profile Screen’s code page. You can modify a box’s features here. The third picture belongs to the “ProfileComponent” file.




```
<ProfileComponent iconFamily="FontAwesome5" iconName="coins" colorName={'#CD7F32'} title="EcoCoins" value="100000" isPressable={false} />
<ProfileComponent iconFamily="FontAwesome" iconName="leaf" colorName={colors.appGreen} title="EcoPoints" value="100000" isPressable={false} />
<ProfileComponent iconFamily="FontAwesome" iconName="paint-brush" colorName={'#3F12A1'} title="Customize" value="" isPressable={true} screenToNavigate="Customize" />
<ProfileComponent iconFamily="Entypo" iconName="shopping-cart" colorName={'#555555'} title="Shop" value="" isPressable={true} screenToNavigate="Shop" />

<Text style={styles.info}>General</Text>

<ProfileComponent iconFamily="MaterialIcons" iconName="leaderboard" colorName={colors.appGreen} title="Leaderboard" value="" isPressable={true} screenToNavigate="Leaderboard6" />
<ProfileComponent iconFamily="AntDesign" iconName="infocirlce"  colorName={'#3F12A1'} title="Overall Statistics" value="" isPressable={true} screenToNavigate="Statistics" />
<ProfileComponent iconFamily="Entypo" iconName="log-out" colorName={'#B22222'} title="Log out" value="" isPressable={true} screenToNavigate="Welcome" />
```

```
  return (
    <Pressable onPress={handlePress} disabled={!isPressable}>
      <View style={styles.infoContainer}>
        <IconComponent
          name={iconName}
          size={25}
          color={colorName}
          style={{ marginLeft: 5, marginRight: 15 }}
        />
        <View
          style={{
            flex: 1,
            flexDirection: "row",
            justifyContent: "space-between",
          }}
        >
          <Text style={{ fontSize: 20 }}>{title}</Text>
          <Text style={{ fontSize: 20, alignSelf: "flex-end" }}>{value}</Text>
        </View>
      </View>
    </Pressable>
  );
};

1 usage   Gizem Özyıldız +1
const getIconComponent = (iconFamily) : Icon<...> | any  => {
  switch (iconFamily) {
    case "Entypo":
      return Entypo;
    case "AntDesign":
      return AntDesign;
    case "Ionicons":
      return Ionicons;
    case "MaterialIcons":
      return MaterialIcons;
    default:
      return FontAwesome5;
  }
};
```

Another example of a component is Flat Cards. The little cards show another users' information. Side-scrollable boxes allow us to use the space more neatly and look better for the user. The user will scroll the cards to reveal others. In the first picture, you can see the components' look on the emulator. The second picture is from Friendly Leaderboard's code page. You can modify a box's features here. The third picture belongs to the "FlatCards" file.



```
<ScrollView horizontal showsHorizontalScrollIndicator={false}>
    <FlatCard avatarSource={require('./assets/avatar.png')} username={'gizemozyildiz'} usernameTag={'#A5F2'}
              icon={'leaf'} points={1555550}/>
    <FlatCard avatarSource={require('./assets/avatar.png')} username={'ahmethakanusta'} usernameTag={'#8FED'}
              icon={'leaf'} points={15550}/>
    <FlatCard avatarSource={require('./assets/avatar.png')} username={'goktugsoylu'} usernameTag={'#31SE'}
              icon={'leaf'} points={3154}/>
    <FlatCard avatarSource={require('./assets/avatar.png')} username={'yavuzbolat'} usernameTag={'#2158'}
              icon={'leaf'} points={8500}/>
</ScrollView>
```

```
5+ usages  Gizem Özyıldız
const FlatCard = ({ avatarSource, username, usernameTag, icon, points }) => {
    return (
        <View style={styles.card}>
            <Image source={avatarSource} style={styles.avatar} />
            <Text style={styles.username}>{username}</Text>
            <Text style={styles.usernameTag}>{usernameTag}</Text>
            <View style={styles.iconContainer}>
                <Icon name="leaf" size={20} style={styles.icon} />
                <Text style={styles.points}>{points}</Text>
            </View>
        </View>
    );
};

const styles   = StyleSheet.create({
    card: {
        width: 150,
        height: 150,
        backgroundColor: colors.appGreen,
        margin: 10,
        borderRadius: 20,
        alignItems: 'center',
        justifyContent: 'center',
    },
    avatar: {
        width:50,
        height:50,
        borderRadius: 25,
    },
    username: {
        fontSize: 18,
        fontWeight: 'bold',
        color: colors.white,
    },
    usernameTag: {
        fontSize: 14,
```

During the coding phase, we used several components like these components we've showed above.

- **UI functions**

Even UI Designs may need some functionality. You can see example below.

renderCategories and renderContent are for rendering the specific category. In the second picture, you can see various categories such as All, Background, Hair etc. Every category has its own items and the category named "All" shows all of them. You can see the basic functions and mapping below, in the first picture.

```
1 usage  Gizem Özyıldız
const renderCategories = () : any[]  => {
    return categories.map((category : string ) => (
        <TouchableOpacity
            key={category}
            style={[styles.categoryButton, category === selectedCategory && styles.selectedCategory]}
            onPress={() : void  => setSelectedCategory(category)}
        >
            <Text>{category}</Text>
        </TouchableOpacity>
    ));
};

1 usage  Gizem Özyıldız
const renderContent = () => {
    const items = contentData[selectedCategory];
    return items.map((item) => (
        <View key={item.id} style={styles.contentItem}>
            <Image source={item.image} style={{width: 120, height: 120, borderRadius: 60}} />
            <Text style={{textAlign: 'center', marginTop: 3}}>{item.title}</Text>
            <TouchableOpacity style={styles.addToCartButton}>
                <Icon name="shopping-cart" size={20} style={{color: colors.appGreen}} />
            </TouchableOpacity>
        </View>
    ));
};
```

## 11.3 User's Manual

# Welcome to EcoFriends!

Welcome to EcoFriends, your mobile companion for a journey towards a greener lifestyle! This user manual equips you with the knowledge and navigation skills to unlock the app's full potential and make a positive impact on the environment.

## Getting Started

Opening the Application:

Launch the application on your android or iOS device.

### 1- Welcome Page:

You will be presented with the Welcome page, which provides options for logging in or creating an account.


**Logging In:**

If you have an existing account:

- Enter your email address in the designated field.
- Enter your password in the password field.
- Click the "Log In" button.

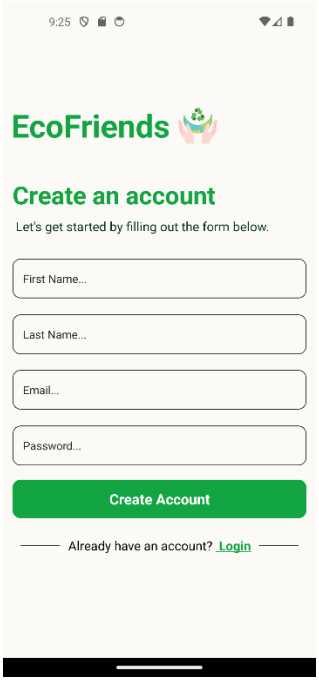
If you do not have an account:

- Click the "Join Us!" button.
- This will direct you to the account creation page (instructions for this page will be provided separately).

**Password Security**: Remember to keep your password confidential and change it regularly.

## 2- Sign Up Page:

After clicking the "Join Us!" button on the First Page, you will be directed to the Sign Up Page.



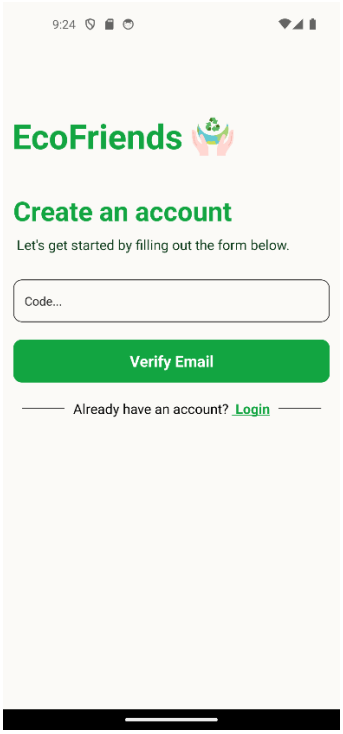

**Required Fields:**

- **First Name**: Enter your first name in the designated field.
- **Last Name:** Enter your last name in the designated field.
- **Email Address:** Enter a valid email address that you have access to.
- **Password:** Create a strong password that meets the application's security requirements.

Once you have entered all the required information correctly, click the "Create Account" button.

You will then be guided to the Authentication Page to verify your email address (instructions for this page can be found in a separate user manual).

## 3- Authentication Page:




**Accessing the Page:** After you pressed the Create Account button you will see the following screen.

You will receive an activation code to you email address, you need to verify your email address by writing the code that you received. Open your email inbox and locate the activation code sent from the application. Carefully type in the activation code you received in your email.

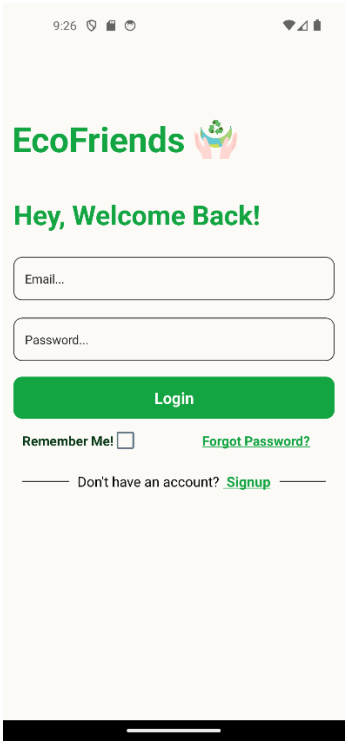
Click the **Verify Email** button.

- If the code is entered correctly, you will receive a confirmation message indicating that your email address has been verified.
- You will then be able to proceed to the application's main features and functionalities.
- Incorrect Code: If you receive an error message stating that the code is invalid, double-check the code in your email and ensure you have entered it correctly.

- Check Spam Folder: Ensure that the email with the activation code has not been filtered into your spam or junk folder.

## 4- Log In Page:

Welcome to the Login Page! This page is your gateway to accessing the application.



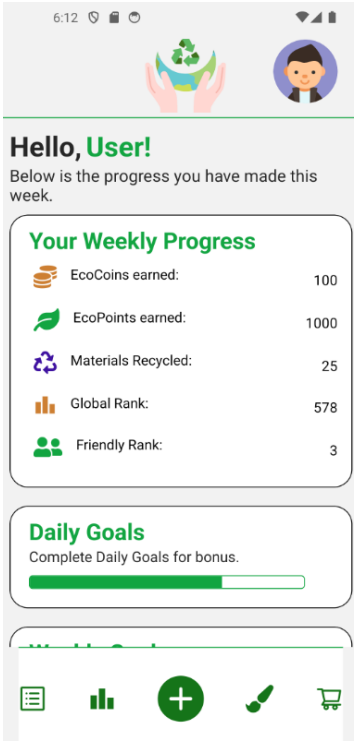

Key Features:

- **Email Field:** Enter the email address associated with your account.
- **Password Field:** Enter your account password.
- **Remember Me! Button:** Check this box if you want the application to remember your login information for future visits.
- **Forgot Password? Button:** Click this button if you need to reset your password.
- **Login Button:** Click this button to submit your login credentials and access the application.

## 5- Home Page:

Welcome to EcoFriends!

Once you're logged in, you'll see the HOME page below.




The Home Page is your landing pad for all things recycling within the app. This is where you'll monitor your progress, get inspired, and stay engaged in your eco-friendly journey. When you open this page, you will see a message that will inform you about your weekly progress. Below the message, you'll find a dynamic feed showcasing your recent recycling activities. This includes:

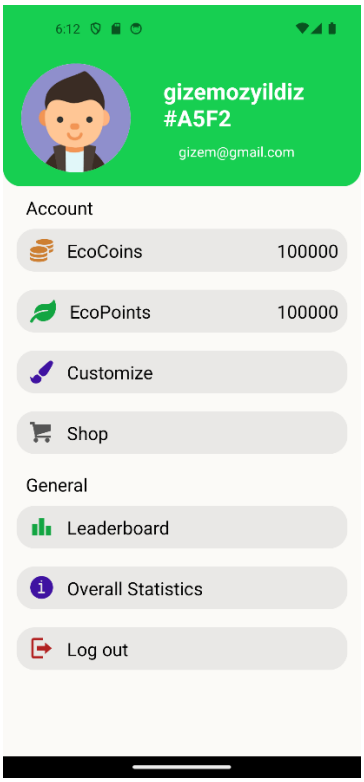
- Ecocoins earned
- Ecopoints earned
- Materials Recycled
- Global Rank
- Friendly Rank

Below that, you will find your daily goal and weekly goal that you need to complete for the bonus points.

## 6- Account Page:

Welcome to Your Profile Page! This is your personal hub for managing your account, tracking your progress, customizing your experience, and connecting with the community.

**Accessing the Page:** On the right top of the application, you can see your avatar and by clicking into your avatar picture, or clicking the left most button on your bottom navigation bar, you will see the following page:




The Profile Page is divided into two main sections:

1. Account: This section focuses on account settings, eco coins, eco points, shop section, and customize option
2. General: This section provides access to Leaderboard, overall statistics the log out section.

**Account Settings:** Manage your account details, preferences, and privacy settings.

- **Eco Coins:** View your current balance of Eco Coins, which can be used to purchase items in the shop.
- **Eco Points:** View your total Eco Points, which reflect your recycling activity and achievements.

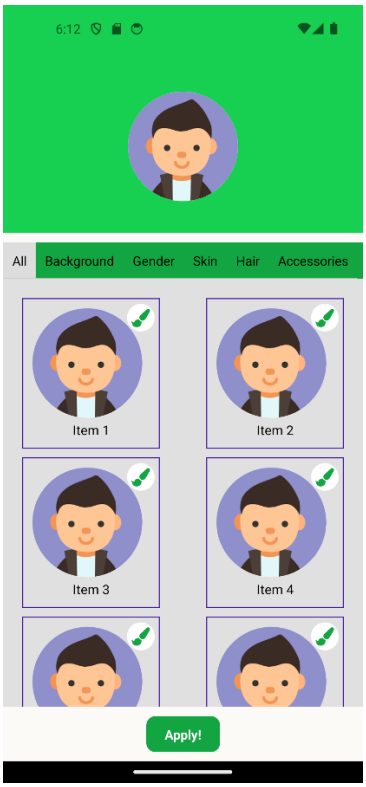
- **Shop:** Access the shop to purchase items using your Eco Coins.
- **Customize:** Personalize your avatar's appearance (see separate User Manual for details).

**General Section:**

- **Leaderboard:** Compare your recycling progress with friends and other users on the leaderboard.
- **Overall Statistics:** View your overall recycling statistics, including total items recycled, points earned, and badges achieved.
- **Log Out:** Securely log out of your account.

## 7- Customize Page:

**Access the Page:** By clicking the customize button in your profile screen or clicking the "paint brush" button in bottom navigation bar, you will be directed to this page.




In this page, you can adjust your avatar's gender, skin, hair, and accessories settings. And by pressing apply! Button, you can apply these changes to your avatar.

Gender: You can choose your avatar’s gender from the available options.

Skin: You can choose your avatar’s skin tone from the available options.

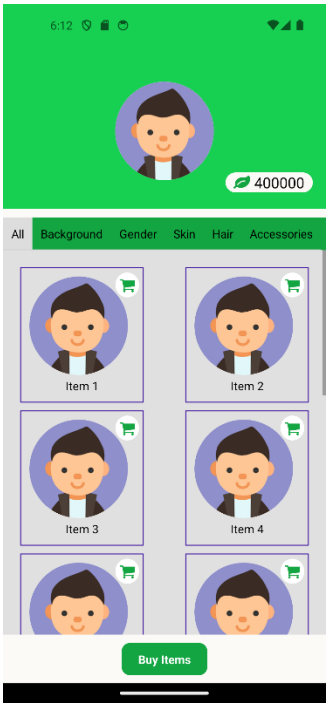
Hair: Choose from a variety of hair styles to create your desired look.

Accessories: Add a touch of personality with accessories.

**Finalize Your Look:** Once you're satisfied with your avatar's appearance, click the "Apply!" button to save the changes.

## 8- Shop Page:

**Access the Page:** By clicking the shop button in your profile screen or clicking the “shopping cart” button in bottom navigation bar, you will be directed to this page.

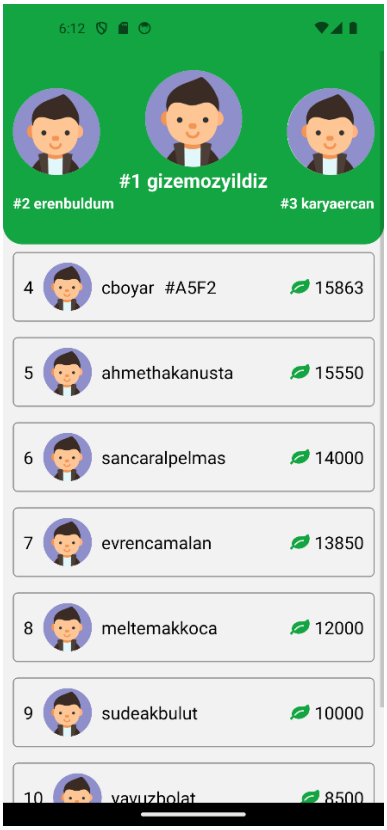



In this page, you can buy items for your avatar. And by pressing buy items! Button, you can buy these changes to your avatar.

## 9- Leaderboard Page:

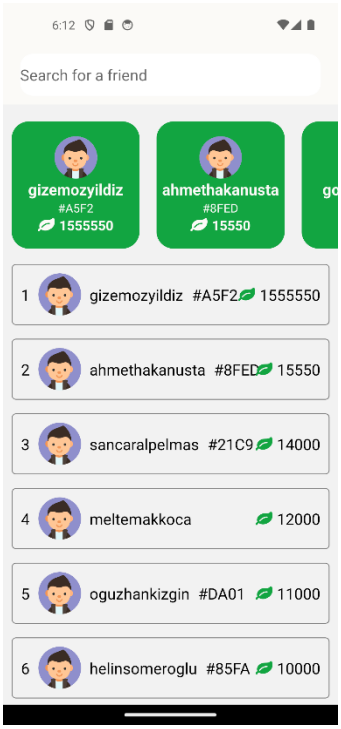
**Access the Page:**

At the profile page, there is a leaderboard button. Also clicking the ranking button in bottom navigation bar you can reach global ranking. When you click on that part, you will see the following screen:




Welcome to your Recycling Leaderboard! This page is your one-stop shop for tracking your recycling progress, comparing with friends, and earning cool badges for your eco-efforts.

You can see global ranking part here, bottom of the screen you can click “friends rankings” and you will be directed to Friends Leaderboard.



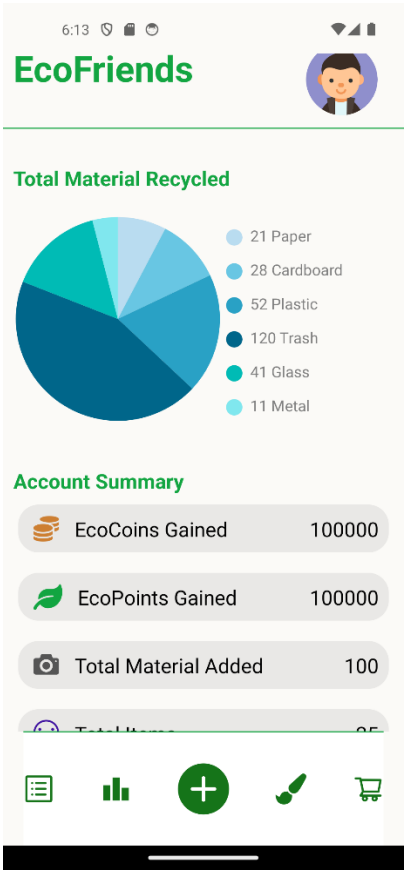

In Friends Leaderboard you can slide your friends points and you can view ranking among your friends below. You can also search for your friend by using search bar. You can return the Global ranking by clicking the button bottom of the page.

Don’t forget to send your friends an request from both of the pages. You can see the button at the bottom of the page too.

## 10- Overall-Statistics Page:

### Accessing the Page:

Tap on the "Statistics" icon or navigation bar of the EcoFriends app.




This screen displays the total amount of each material you have recycled, categorized as follows:

- Paper
- Cardboard
- Plastic
- Trash
- Glass
- Metal

This screen summarizes your overall progress and rewards:

- EcoCoins Gained: The total number of EcoCoins you have earned.
- EcoPoints Gained: The total number of EcoPoints you have earned.
- Total Material Added: The total number of items you have added for recycling.
- Total Items Gained: Total customization items bought.

At the middle bottom of the page, you will see an + symbol it is an **Add Materials Button**

This button, typically represented by a "+" symbol, allows you to add new recycling activities.

**Interpreting the Information:**

- Review the quantities listed under each material type to see your breakdown of recycled items.
- Check your EcoCoins and EcoPoints balances to track your rewards and potential for avatar customization or shop purchases.
- The Total Material Added count reflects your overall recycling engagement.

Congratulations on taking the first step towards making a positive impact on the environment with EcoFriends! We hope this user manual has equipped you with everything you need to navigate the app and unlock its full potential. Remember, every recycled item counts, and EcoFriends is designed to make your journey towards a greener lifestyle fun, rewarding, and engaging.

# 12. References

Visit our website: https://ecofriends.github.io

https://clerk.com/docs

https://docs.expo.dev

https://firebase.google.com/docs?hl=tr

https://reactnative.dev

https://huggingface.co/datasets/garythung/trashnet