**Proposed Level of Achievement: Gemini**

Github repository: https://github.com/orbital2019-team1952/EcoAction.git

# 1. Background

## 1.1 Motivation

Our environment has been in a severe situation for a long time. It is both damaging our current living conditions and our future generations. There are many urgent problems such as pollution, global warming, resource depletion, etc. As most of the issues are caused by human activities, it is important for people to internalize the mindset of environmental protection by living a sustainable lifestyle.

However, there are still a lot of people who lack the incentives to take real-life actions as they are not convinced about the impact which an individual can make, or they are not aware of the possible actions in the first place. Therefore, to increase users' incentives to practice green actions and raise awareness of environmental protection, we would like to develop an app to recommend possible actions and keep track of users' impact, so that users can be more environmentally conscious and more motivated to take real-life actions.

## 1.2 User Stories

1. As a person who concerns about the environment, I would like to know how I can contribute to reducing environmental damage.

2. As an environmental activist and educator promote the importance of conserving the natural environment, I would like to spread the message and encourage more people to take part.

3. As a person who is passionate about learning, I want to gain more knowledge about our natural environment.

## 1.3 Aim

We hope to make eco-friendly actions more accessible, to make individual impacts more visible, and to provide relevant knowledge regarding environmental protection to people.

# 2.Introduction

## 2.1 Features Developed and Scope of project

**Eco-Action** is an iOS mobile application with an interactive design to encourage a sustainable lifestyle among users, as well as to recommend and keep track of users' eco-friendly practices.

### 1) User registration / log in

This is the basic feature of our application. Users must register for an account in order to access the remaining features of the app. Users can use their email address with a password to register for an account. They are able to create a nickname at the same time. If users have already registered before, they can log in to their account directly by email and password. User accounts are essential to keep track of users' individual data, as well as collective data calculation by all users. We use firebase to record and analyze users' account and activity information. When a user registers for an account, it will reflect in the authentication in firebase, including the user's email address, time for registration and login, and their respective user ID.

### 2) UI design of all pages

We have developed UI design of all necessary pages, including start page, login page, register page, main menu, game page, article page, action page, add action page, and achievement page. We used custom flat buttons and text fields to improve usability. Auto layout is also set up to fit various screen sizes. The pictures of sea turtle and plastic waste are original drawings by HSIAO-HAN. We have also used various frameworks to improve the visual design of the app.

### 3) Track daily action

Front-end wise, Action page can be accessed from the main menu, which includes all the essential features of the app. This page keeps track of all the actions users have taken in a chronological manner. The time detailed to seconds when the user records the action is also reflected. The action timeline is presented in a table view. Users are able to record actions by clicking the "Add Action" button on the action page to access a list of possible actions. They can check the actions they have taken and click "Done" to add those actions in their own timeline. Upon selection, the action will be recorded in the user's database. The action will be transferred into points and

reflected in their account information. This data will then be used to calculate the total number of users who have taken the same action.

### 4) View achievements

Achievement page can be accessed from the main menu as well. Users can view their individual as well as the public's achievement in either the most recent week or month. The achievements are presented in bar charts. The charts record the number of users who have taken their respective actions. All data is collected and calculated from firebase.

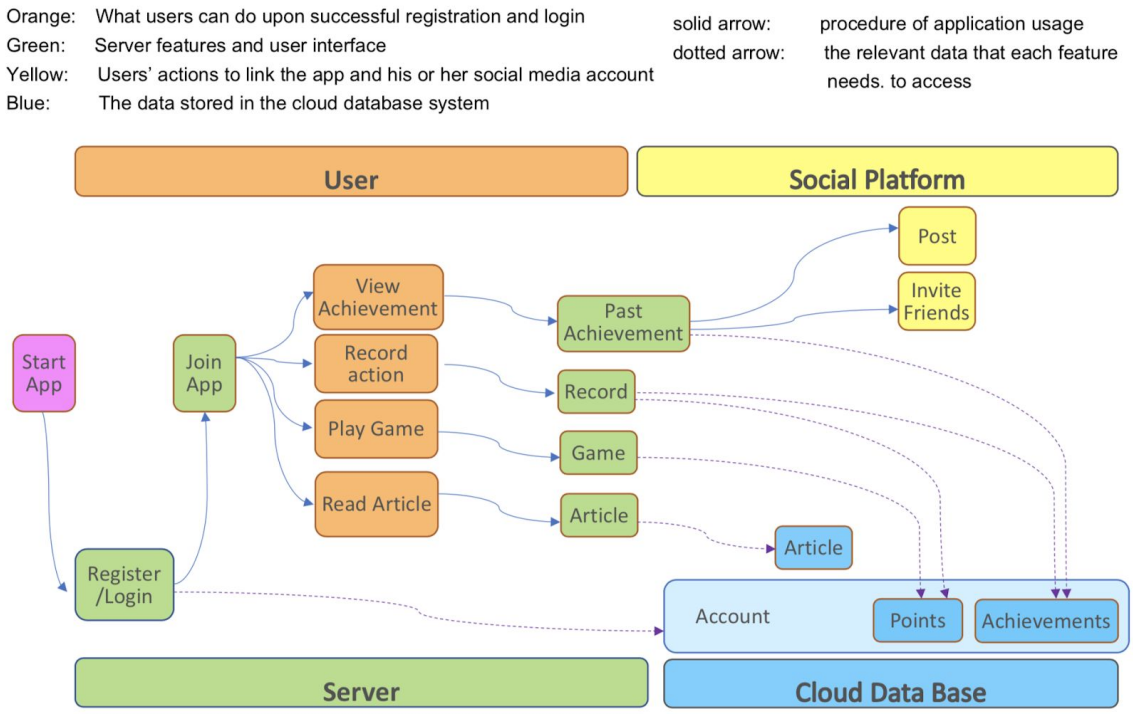
### 5) Gain knowledge

The application has the feature to introduce knowledge cited from credible sources regarding environmental protection and tips of sustainable practices. Like above, Article page can be accessed from the main menu page. There will be a list of articles for users to read. The individual article is presented in Sarafi view controller.

### 6) Game Feature

The game feature serves as an interactive design for users to engage. Upon successful registration, each user will be assigned an animal to take care of. Initially, the animal is trapped in severe environmental conditions such as pollution. By completing the green actions, users can gain reward points and use them to improve the environment that this animal is living in, such as reducing the amount of plastic waste around this animal.

## 2.2 Program Flow




# 3. Technology

### Front-end

- iOS app development: Client-side Swift with the UIKit framework
- UI/UX: Interactive design with proper functionality, usability and user adaptability

### Back-end

- Version control with Git
- The main logic of application: Server-side Swift
- Data visualization: managing and storing graphic/diagram representation of collective user data
- Database cloud management with Firebase: user account information, articles in the form of websites
- Game development: gamification design and point system

# 4. Problems and bugs squashed

We divide our project into several parts to search for online tutorials/solutions.

## 4.1 Firebase

- **Add Firebase to Xcode project**
  We use Cocoapod to add Firebase to Xcode project. However, the official document only shows the codes we need to add to our project, so we did not know which document need to modified and where it is. So we searched for some other websites that show the screenshot to help us. Also, the online resources seldom mention that pod files need to use Xcode workspace, so when we use Xcode project, it shows a lot of errors. So we spent some more time to search the difference between Xcode project and workspace.

  - Problems we encountered: Install Firebase via Cocoapod. Use Xcode workspace.

- **Log in / Register**
  Our app uses Firebase to store user information and data, so the app needs a function to add the user account to our Firebase project. This part we use the official Firebase document and other blogs as the tutorial, and also search for more online information to restrict the style of email and password(password need to be longer than 8 words, or an error message will appear)

  - Problems we encountered: Perform the segue way only when the email and password are correct.

- **User Data / User Property**
  Firebase Database use asynchronous method to read data, so it is a problem when need to use data in Firebase Database, cannot use functions to return data inside.

  - Problems we encountered: Cannot use different class to pass Firebase Database value.

- **Using CoreChart**
  Because of the asynchronous problem from Firebase Database, it is hard to reload chart using the CoreChart library because it need to pass value in its initiate function. Also, the simulator has some unknown problems so it cannot show the chart correctly even after rewriting the codes.

- Problems we encountered: Loading data to CoreChart and show it on the simulator.

## 4.2 Track daily action / View achievements

The page of daily action needs to show the chart and is able to add a new action to it. We decide to use tableview to show their achievements. However, most online tutorials use Chart.js, so we need to find the software that can use on Xcode. Also, Xcode does not provide multiple choice function, so we are still searching for a better way that does not need to create buttons to collect data.

- Problems encountered: Unfamiliar with Xcode, searching for more information on the implementation

## 4.3 Gain knowledge

We have tried various forms of article presentation. Initially, we plan to embed the articles in a web view or transform the articles to a customized content view. However, due to copyright constraints, we decided to use SFSafariViewController to present the articles. This method preserves the original content of the articles to the fullest without the need to design additional view controllers.

# 5. User Testing

We have done user testing on several iPhones to test its functionality and usability. We asked our family and friends with iPhones to download the app and test it out. Functionality wise, all users have successfully registered for an account. User information is correctly reflected in firebase authentication. We have also tried login which is also successful.

However, there were some problems with usability at first as the keyboard will block the text field when typing. Some labels were also exposed for editing. To fix those problems, we added a few functions to lift up the view when the user is typing on the text field, as well as to disable user interactions to labels.

Another problem we discovered is that the action page and achievement page cannot be accessed by new accounts. We then changed snapshot value to optionals, and used if-else statement for achievement page.