Climate Code Ideas

Tools:

* General
  + Firebase
  + Figma
* Android Studio
  + Google Maps API
* Visual Studio Code
  + Google Maps
* Presentation software
  + Slides

Brainstorm:

* Carbon emission tracker
  + Could use Google Maps API
  + Calculator?
* Forest fire reporting app
  + Two personas:
    - Volunteer → can report fires in area
      * Need to submit photo & brief description
    - User
      * Red spot on map → Google maps API
  + Google Firebase
  + Government can access the database 24/7
  + Two sides:
    - Client app
    - Web
      * Has more info; analysis on data
      * For gov & developers
* //Potentially → natural disaster tracker
  + We pull data from satellite imagery?
* //App that allows users to share personal tips on how to be more environmentally-friendly
* //Educational game on climate change
* Food expiry app
  + (idea may not be as original)
* Environmeter
  + Water bottle
* Environmentally-friendly version of Uber Eats

## 

## **EnvironMeter**

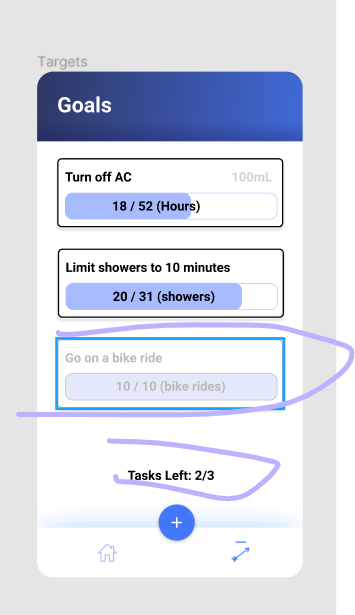
Description:

* An Android app that incentivizes environmentally-friendly actions through a visual rewards system
* The user completes environmentally-friendly tasks, and then they’re rewarded with “water” in a bottle

Features:

* Level upgrade if user has done a specific amount of thing which helped the climate
* Achievement page with goals for the user to do
  + Also includes number of mL each task is worth
* Physics animation for water (https://github.com/tangqi92/WaveLoadingView)
* Extra\*\*\*
  + Review system in which developers/admin look over every achievement to ensure it has been properly completed

To-do List:

* Finish designing the user interface in Android Studio (XML files)
* Clean up the bottom nav bar
* Level System
  + Write the logic in Java
  + For each task the user completes, the number of mL is added to the home screen, more water is added to home screen & the percentage filled increases too
    - 
  + Level number increase when the water is filled up
* The user clicks the add button (floating action button) → they can add submission
  + Write the logic in Java
  + Submission
    - User should be able to choose task from drop-down menu, write description, upload image
* Physics animation to reflect the water amount
* PITCH
  + Tomorrow evening / Sun morning

(Optionally)

* //Database for submission
* If possible, add animations to make it smoother
* AI (review system), database, animations, fully-implementing the levels system
* User can add their own custom tasks
* User can see history of submissions

Colours:

Targets for user to achieve:

1. Turn off AC
2. Walk to School
3. Open the windows
4. Use public transportation when possible
5. Recycle

## Project Description

When we first knew this hackathon was related to climate, we instantly knew we wanted to build something that can help people change their life habits. Life habits hold an important place in climate change, and it can directly affect our carbon footprint. According to a news article from theStars.com, ***“Experts say that it is, but we’ve a very long way to go before we get close to it. The average consumer around the world is well over the emissions target of 2 tonnes a head – meaning big savings will have to be found.”***  So, we decided to make a mobile app to help people sustain an environmentally friendly lifestyle, hence decreasing our carbon footprint. Meet Envirometer: an app that incentivizes green-living through a visual rewards system, which can help users to lower their carbon footprint.

To describe our approach, our first step of this hackathon was to create a brainstorming document with a list of different ideas. From there, we decided on our current Android application idea, and we started to research activities that help the climate and the environment.

We used Figma to create a blueprint of our UI interface which gave us a clearer path on how to develop the application programmatically. After creating most of the blueprints, we then began working on the actual app. We used Android Studio to develop the software and wrote it in Java; the layouts and views were created with XML. We also implemented a moving water animation plugin created by tangqi92 (<https://github.com/tangqi92/WaveLoadingView>) to make a water bottle-like object in our app and to visually represent how much water the user has earned.

The final version of Envirometer features two primary pages: a home screen and a tasks screen. When the user first opens the app, they are met with the home screen that displays how much water they have accumulated for their level — this information is displayed in three ways: with the wave animation in the background, the progress bar in the header, as well as the numeric amount in a glowing box. They can either switch to the tasks page which contains a RecyclerView of all the tasks the user has to complete for their level (each cardView also contains information about how many hours the user has already spent completing each task); or, they can record their progress towards an existing goal by pressing the floating action button. This will allow the user to select which task they worked on, select how many hours they spent doing the task, and upload a photo of the task as evidence/to hold them accountable. Once the user submits their progress, the Tasks page will update itself accordingly.

If the user has completed the required number of hours for a task, they will earn the volume of water the task is worth; the progress bar, wave animation, and numeric representation will update accordingly. Once all the tasks have been completed and the user has earned all the water, the user will level up and have more environmentally-tasks to accomplish. The water the user earns will eventually be donated to a non-profit partner.

## Citations

* Physics animation for water (<https://github.com/tangqi92/WaveLoadingView>)
* <https://en.wikipedia.org/wiki/Stanford_marshmallow_experiment#External_links>