



# Circle

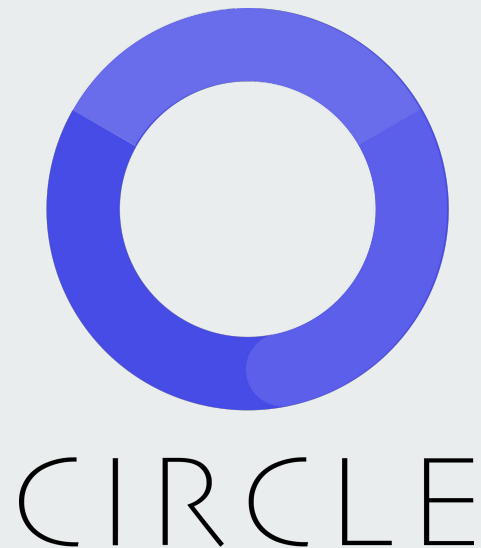
Team HDMI | CS8803-MAS Sprint 4

Xiuxiu Yuan, xyuan40@gatech.edu

Xuejin Tan, tanxjs@gatech.edu

Bin Xie, bxie41@gatech.edu

Anjian Peng, penganjian@gatech.edu



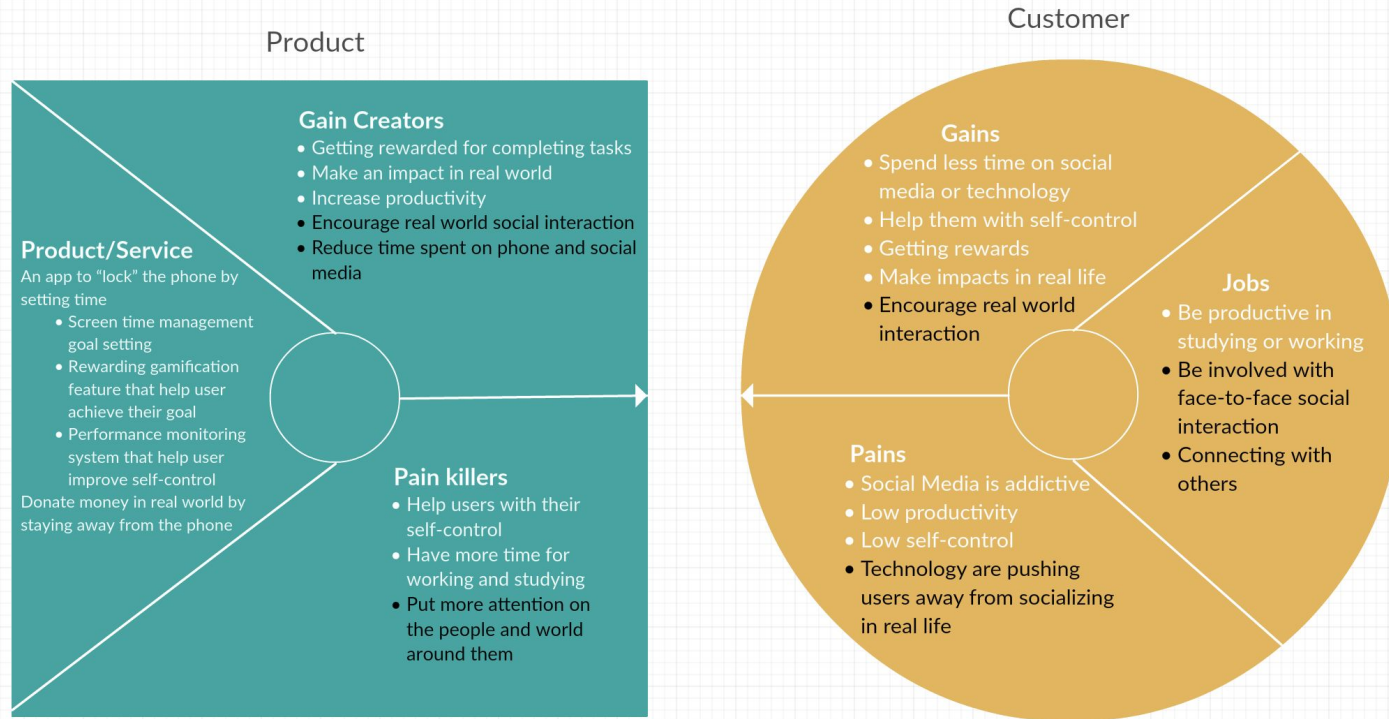


# Problem Description

“Great technology should improve life, not distract from it”. - Google

People spend too much time on their smartphones, which is not good for their digital wellbeing. Our design aims to **assisting people better manage tech usage, so they can focus on what really matters for them, and develop healthy tech usage habit.**

## VALUE PROPOSITION DESIGN EXPLANATION





## Peer Feedback from Sprint 4

- What counts as being focused? How do you prevent users from abusing the functionality in order to get rewards?
- What does star has to do with traveling?
- Future direction? A parental-control app?

We have **refined app features and demo flow** to better explain the confusions.



# Design Updates since Sprint 4

Blue: Presentation Red: User Testing/Interview

## Wireframe changes:

- Separate "Postcard" as its main section from menu
- Statistics screen: delete charity and move it to individual section, could access from the menu bar
- Remove charity goal from "my journey" to side menu

## UI changes:

- Journey Screen: tab names updates
- Add one 'revisit' button in visited list in "Journey Screen"
- change the # to progress bar to make it charity goal more visible
- Change "circular slider" range/intervals so every drage is 5 min, and the minimal time for charity goal is 15 min
- Statistics screen: create more data to visualize the feature better
- Add # of cities that could be unlocked on the side menu
- My journey: add star needed for unlocking city, and change label of button from "travel" to "unlock"

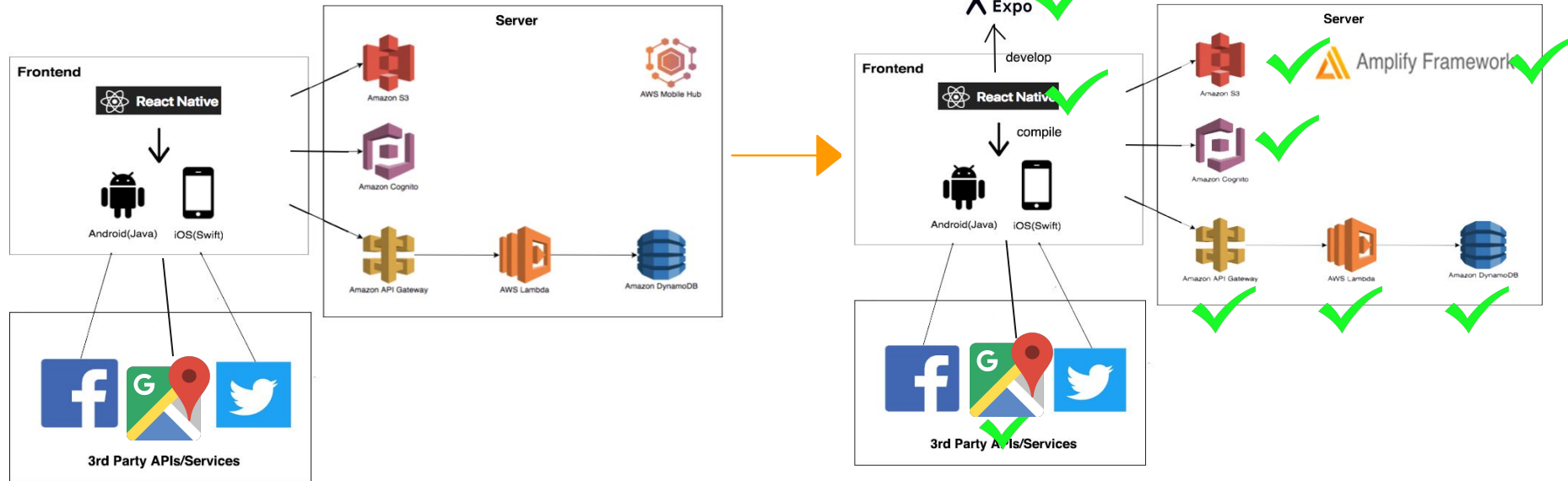
## New features:

- Add "Friends" functionality so people can see their time focused ranking compared to others
- Add Facebook login option (import user profile, avatar)
- Access "my journey" by clicking the main "timer image" on home screen

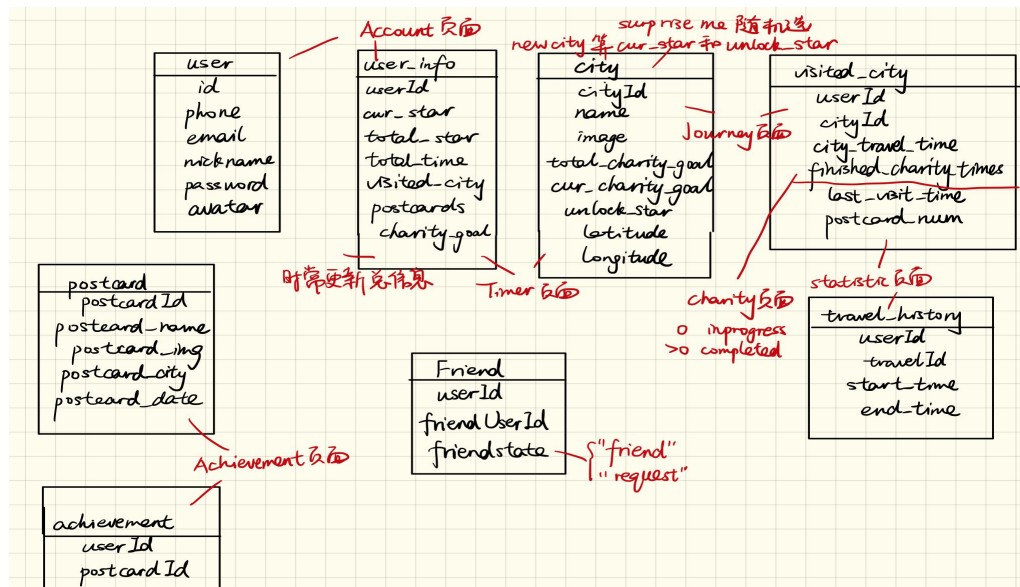
## Feature changes:

- The timer should still work even if the screen is down.
- When timer completed successfully, add stars accordingly.
- Update city icons and postcards with new assets
- There should be notifications when users quit the App.
  - Notification text: Go back to CIRCLE now or your timer will end.

# Architecture Update



## Architecture Update - Database design





## Architecture Update - API design

REST Resources

- ▼ /
- ▶ /achievement
- ▶ /city
- ▶ /friendrequest
- ▶ /friendship
- ▶ /info
- ▶ /postcard
- ▶ /user
- ▶ /visitedCity





**Demo** Show features with a **first time user** & a frequent user.



**Tyler**

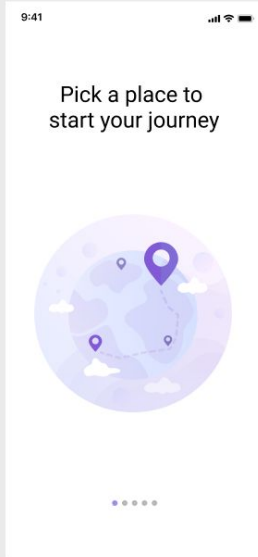
First Time User

**Why CIRCLE?**

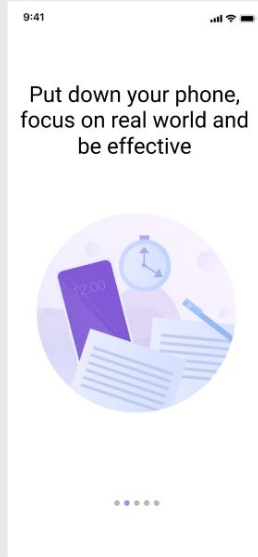
She wants to spend more quality time with her family.

# First-Time User: On-Boarding

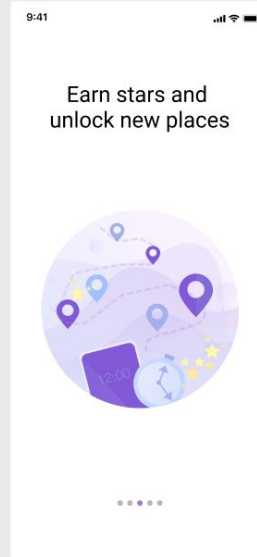
On boarding - 01.3



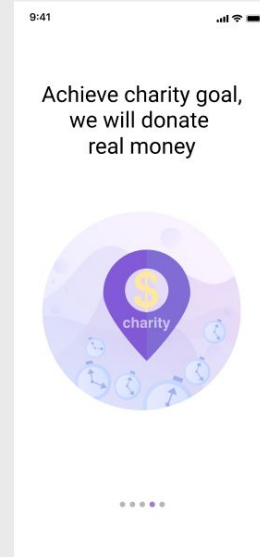
On boarding - 02.2



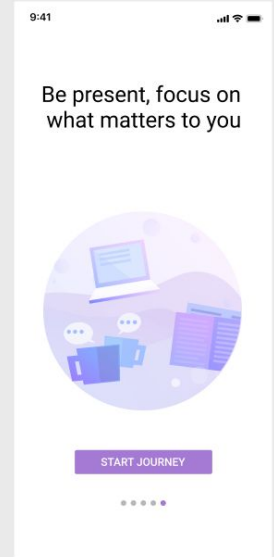
On boarding - 03.2



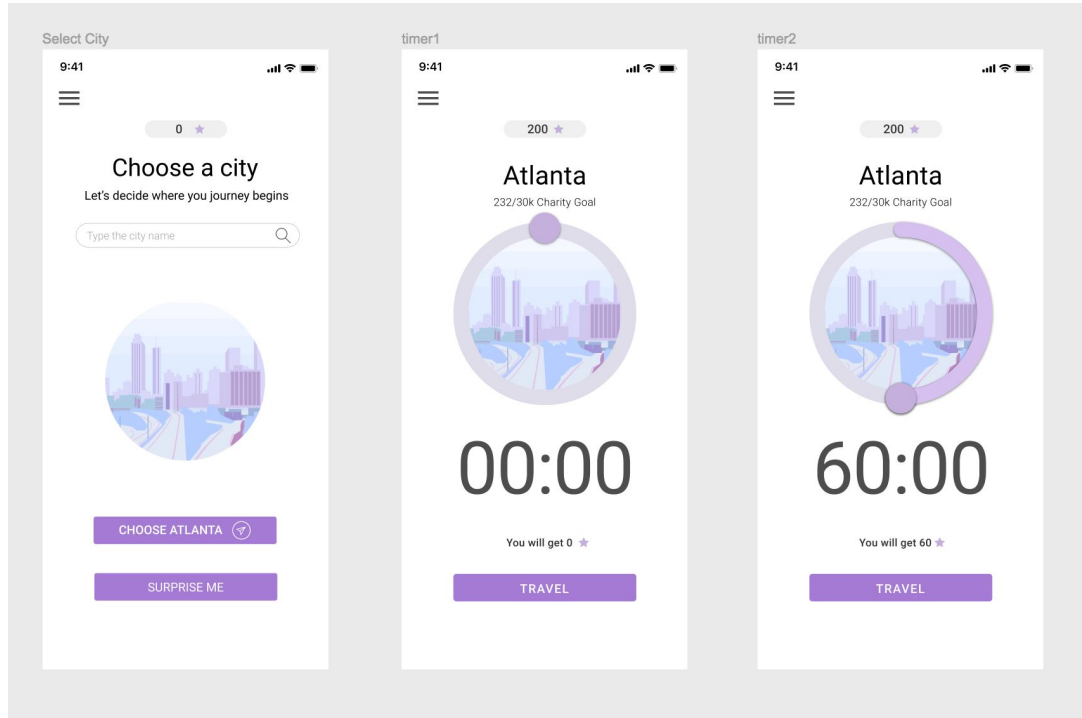
On boarding - 04.2



On boarding - 05.2

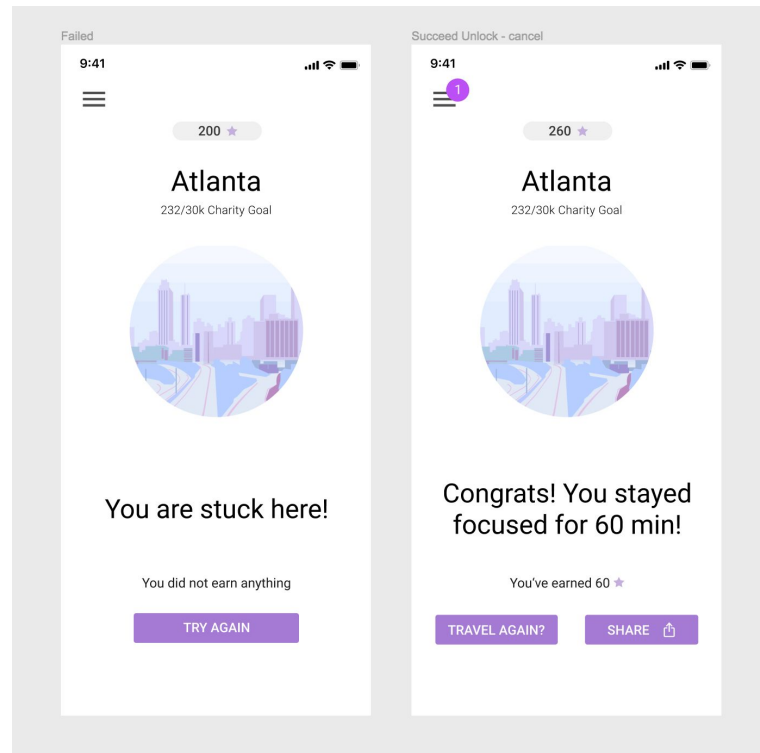


# First-Time User: Start a timer





# First-Time User: Timer Fail/ Success



**Demo** Show features with a **first time user** & a frequent user.



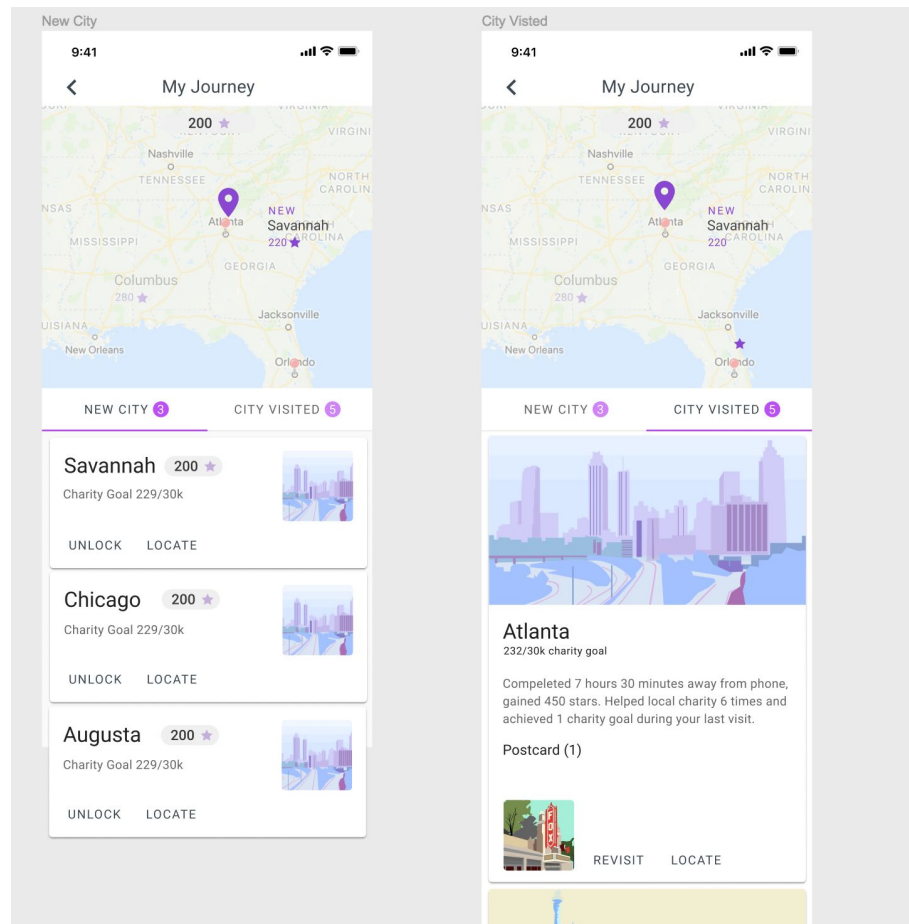
**Lin**

Frequent User

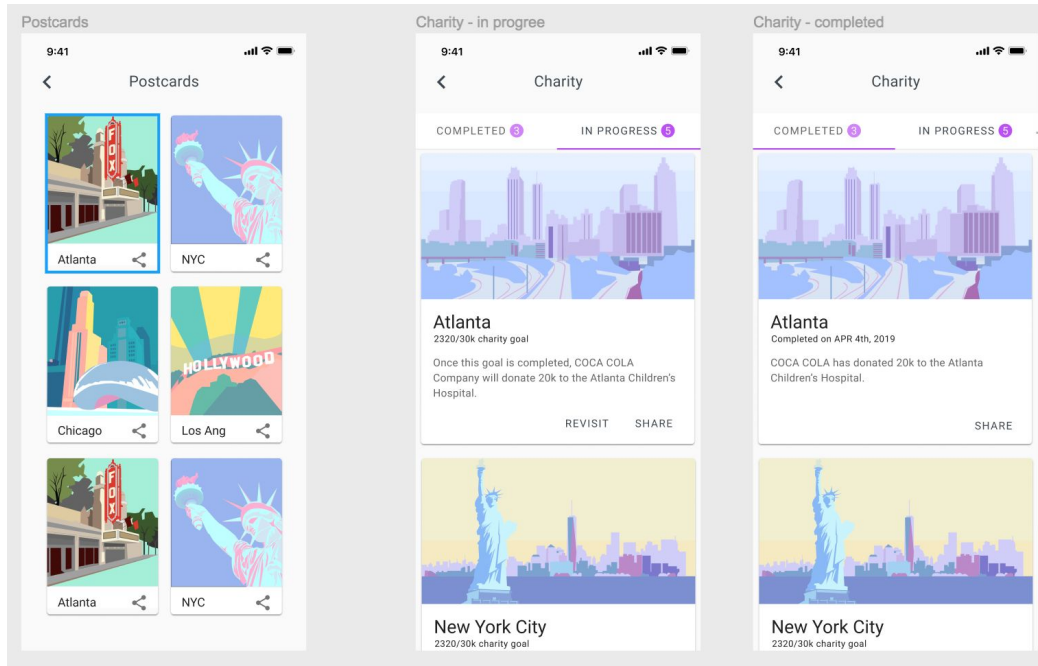
**Why CIRCLE?**

She wants to focus more on her work.

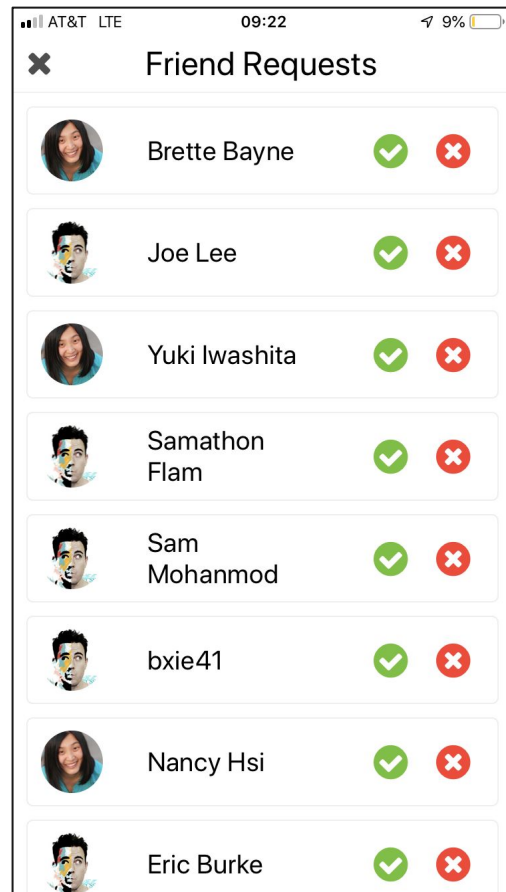
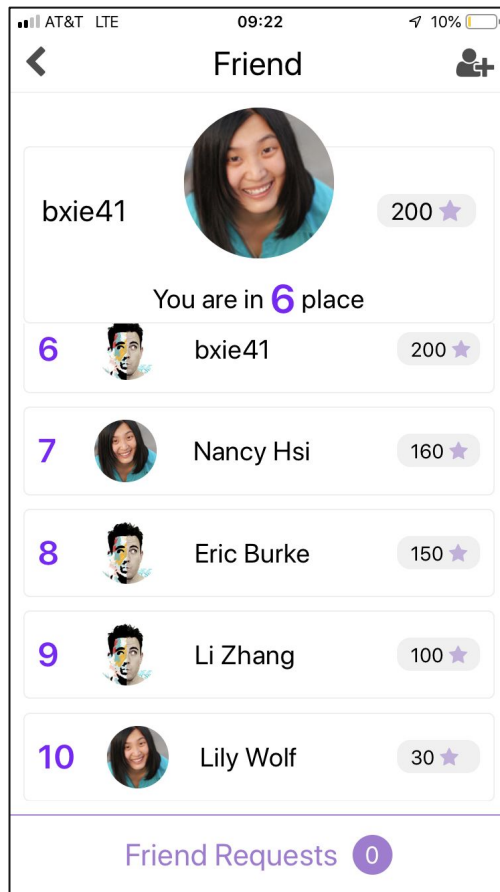
# Return User: Unlock a new city / View Summary



# Return User: Check Postcards and Charity Goals accomplishments



# Return User: Check Ranking & Adding Friends







# Future Directions

- Implement more cities
- Test with larger audience
- Find sponsors
- Implement new modes (parental control?)
- Real Charity Goal



# Code Review - front end

- Views Folder: for screen layout and interaction
- Components Folder: for components layout
- APIs: APIs to call back-end services
- Constants: common constants used in the whole project
- aws-export.js: for the configuration

```
src
├── APIs.js
├── Constants.js
├── aws-exports.js
├── components
│   ├── CharityGoalList.js
│   ├── CircularSlider.js
│   ├── FriendHeader.js
│   ├── JourneyMap.js
│   ├── MyHeader.js
│   ├── NewCityList.js
│   ├── SideMenu.js
│   ├── Swiper.js
│   ├── TimeChart.js
│   ├── TimerHeader.js
│   ├── VisitedCityList.js
│   └── VisitedMap.js
├── global.styles.js
├── img_path.js
├── navigations
│   └── stackNav.js
└── views
    ├── AccountScreen.js
    ├── AchievementScreen.js
    ├── CharityScreen.js
    ├── CityScreen.js
    ├── FriendScreen.js
    ├── InitScreen.js
    ├── JourneyScreen.js
    ├── OnboardScreen.js
    ├── StatisticScreen.js
    └── TimerScreen.js
```



## Code Review - back end

- **api:** for the AWS API Gateway configuration
- **auth:** for the AWS Cognito configuration
- **function:** back-end business logics
- **storage:** for the AWS dynamodb configuration

```
└─ backend
  ├── api
  ├── auth
  ├── awscloudformation
  └─ function
    ├── achievement
    ├── city
    ├── friendrequest
    ├── friendship
    ├── info
    ├── postcard
    ├── user
    ├── visitedCity
    └── storage
```



# Code Review - sample

Github: <https://github.com/Lazylce/circleApp>

OnboardScreen: screen shown when providing onboarding information

```
import React, { Component } from 'react';
import { StyleSheet, Text, View, Image, Dimensions } from 'react-native';
import Swiper from '../components/Swiper';
import { board1, board2, board3, board4, board5 } from '../img_path';

const WIDTH = Dimensions.get('window').width;

export default class OnboardScreen extends Component {
  constructor(props) {
    super(props);
  }

  render() {
    return (
      <View style={styles.slide, { backgroundColor: '#fff' }}>
        <Swiper {...this.props}>
          <View style={styles.slide, { backgroundColor: '#fff' }}>
            <Text style={styles.text}>Pick a place to start your journey</Text>
            <Image source={board1} style={styles.img} />
          </View>
          <View style={styles.slide, { backgroundColor: '#fff' }}>
            <Text style={styles.text}>Put down your phone,</Text>
            <Text style={styles.text}>focus on real world and be effective</Text>
            <Image source={board2} style={styles.img} />
          </View>
          <View style={styles.slide, { backgroundColor: '#fff' }}>
            <Text style={styles.text}>Earn stars and unlock new places</Text>
            <Image source={board3} style={styles.img} />
          </View>
          <View style={styles.slide, { backgroundColor: '#fff' }}>
            <Text style={styles.text}>Achieve charity goal, we will donate really money</Text>
            <Image source={board4} style={styles.img} />
          </View>
          <View style={styles.slide, { backgroundColor: '#fff' }}>
            <Text style={styles.text}>Be present, focus on what matters to you</Text>
            <Image source={board5} style={styles.img} />
          </View>
        </Swiper>
      </View>
    );
  }
}
```