



Midway Milestone

Jourdann Fraser
Vy Mai
Tiffany Manuel

Problem/Solution Overview

Journaling is time-consuming, and photos only capture visual details. Rechords uses the connection between music and memories to **recreate the environment of the memory itself** and give users **a better way to remember.**

Mission Statement

By **connecting music to memories**, we hope to **add a new dimension** to memory keeping.

Overall Design Changes ***HE Results***

Corresponding Revisions

Prototype Status

Tools

Implemented Features

Unimplemented Features

Wizard of Oz

Hard-Coded Data

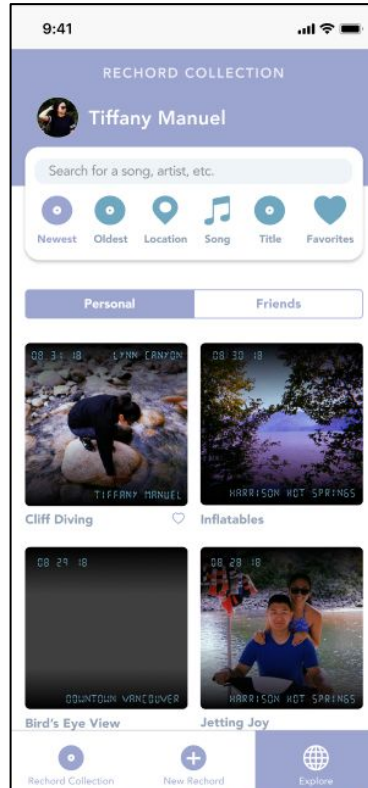
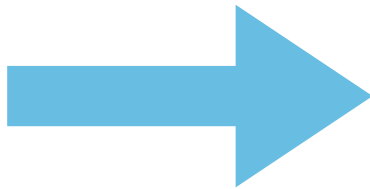
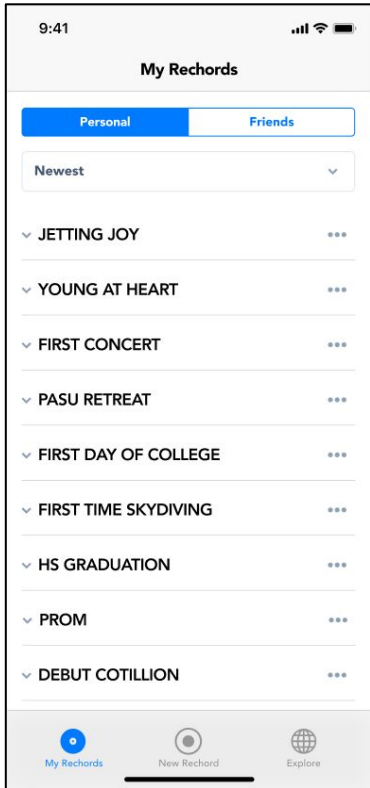
Demonstration

Summary

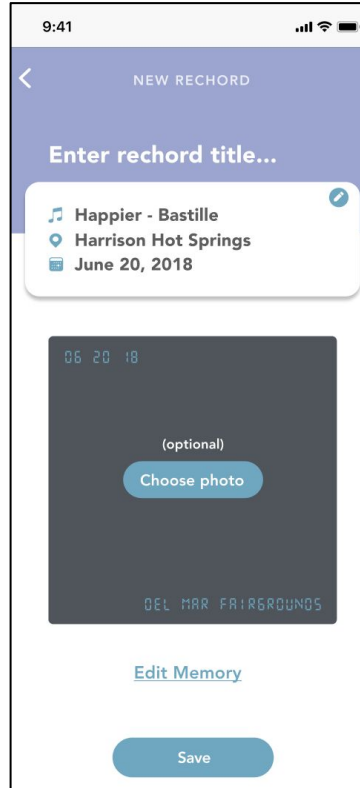
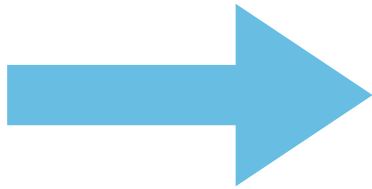
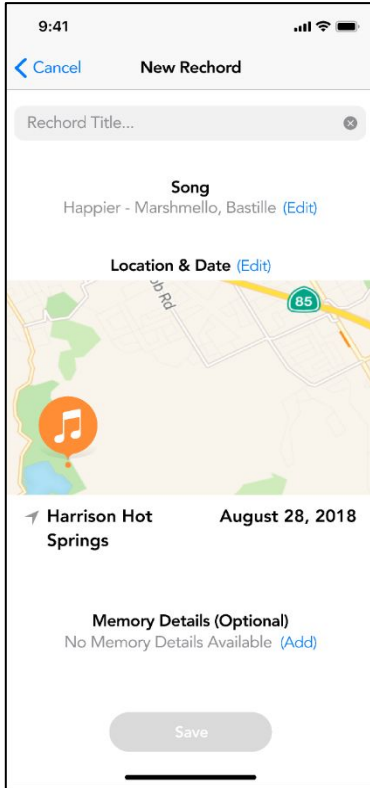


Overall Design Changes

Playlist Design to Album Design

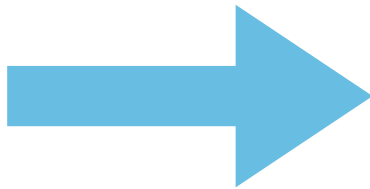
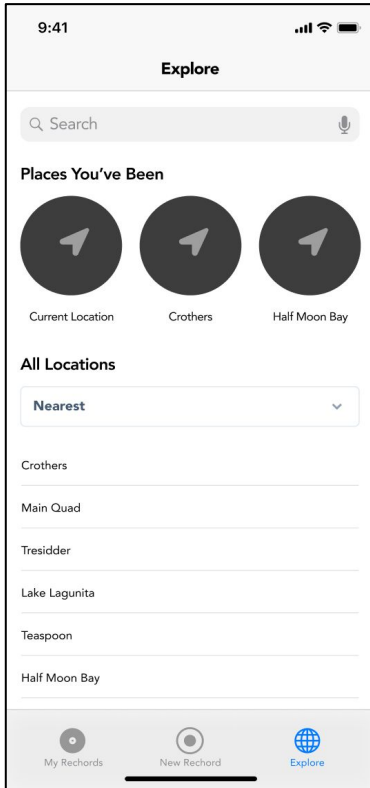


“Create New Rechord” Screen



- Required info grouped together at the top
- Optional info grouped together at the bottom
- Album Design Applied

“Explore” Screen



- Applied redesign
 - Made consistent with the feel of the rest of the app
- List view easier to look at
 - Less cluttered

Heuristic Evaluation Results

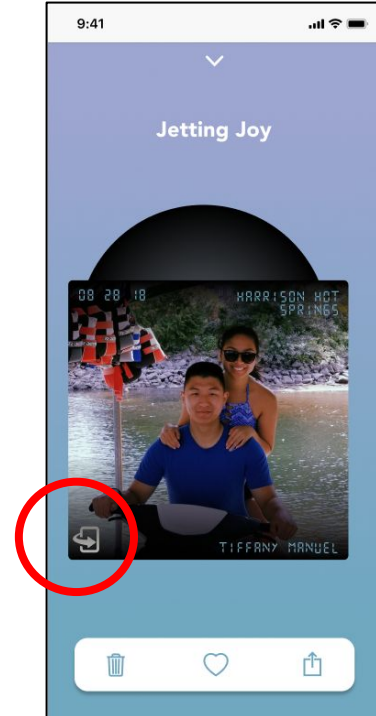
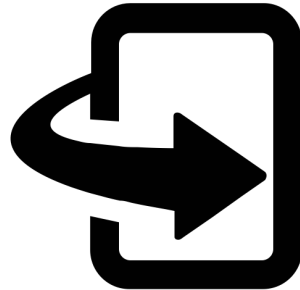
Unclear that rechorde card is flippable/clickable

Problem

- Unable to find “Rechorde Description”

Solution

- Include “Flip” icon



No “Add Friend” option

Problem

- Unable to find friends when sharing

Solution

- No database of friends
- Search through all users for now

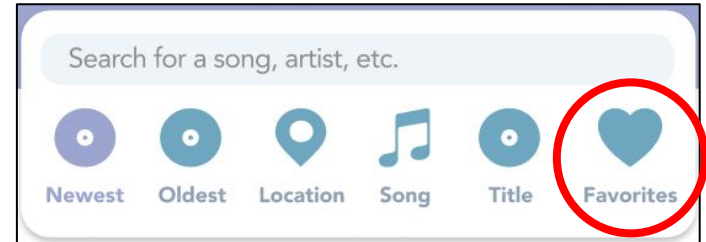
“Favoriting” is unclear

Problem

- Unclear what results of favoriting are
- Are your favorites private or public?

Solution

- “Favorites” added to “Sort By” options



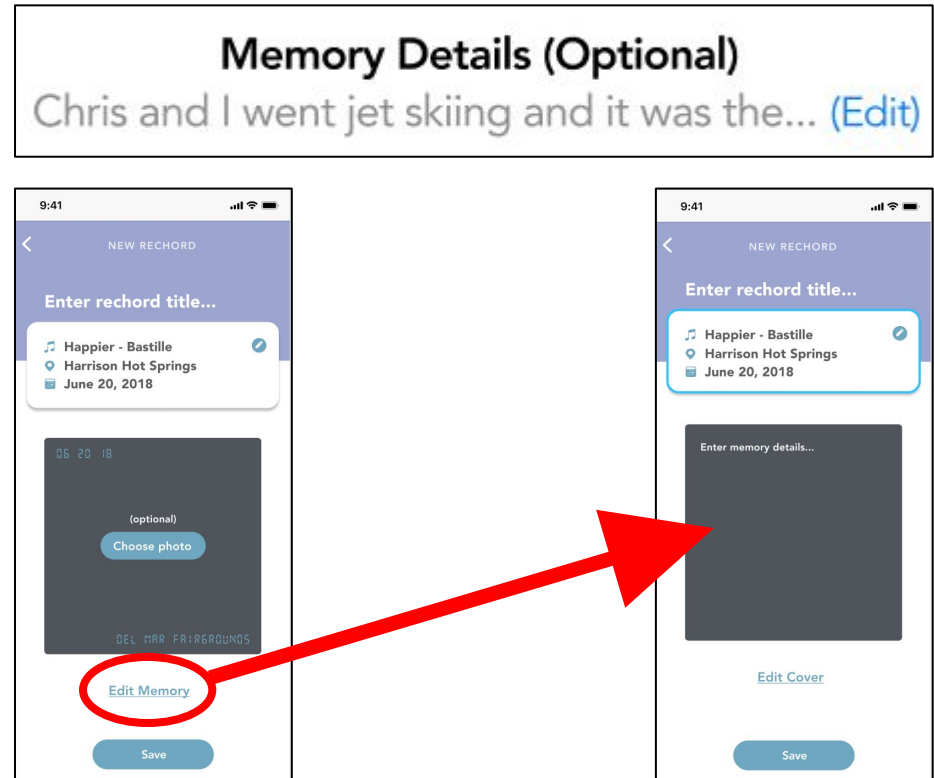
Proofreading “Memory Details” is difficult

Problem

- Proofreading requires an extra click
- Lack of error prevention

Solution

- Text field remains on the same screen



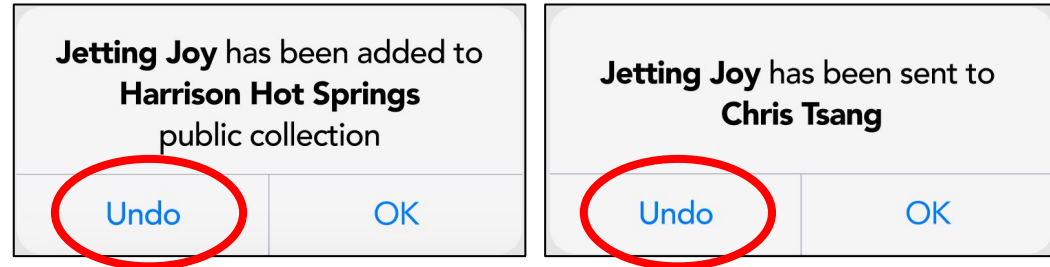
No “Undo” option for adding to location/sharing

Problem

- Unable to remove record from Location Playlist in case user accidentally adds
- Unable to unshare record in case user accidentally shares

Solution

- Include “Undo” option



Cannot edit song and location

Problem

- After saving rechorde, only able to edit “Memory Details”
- Accidentally save incorrect song/location

Solution

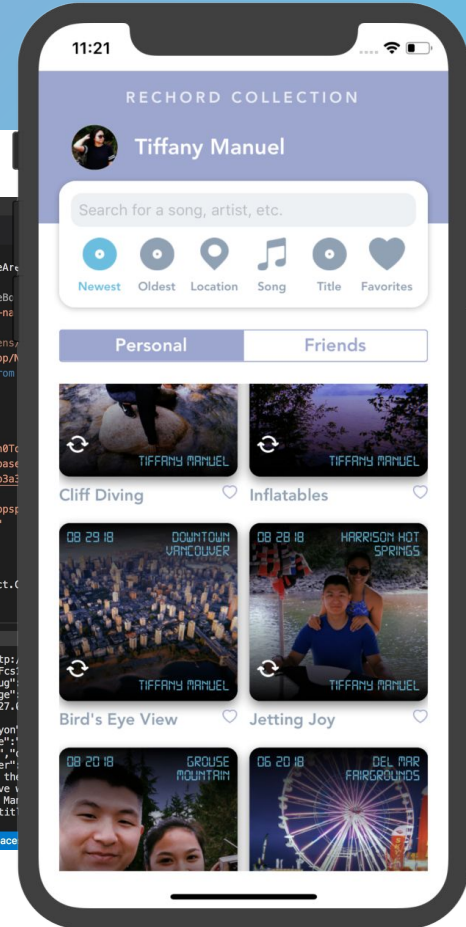
- Allow edits to song and location

Prototype Status

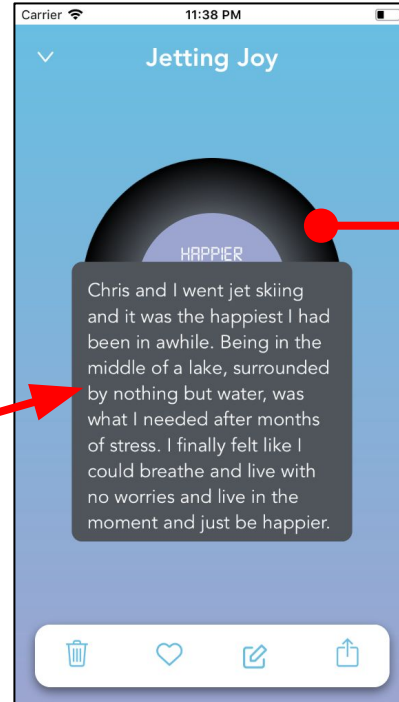
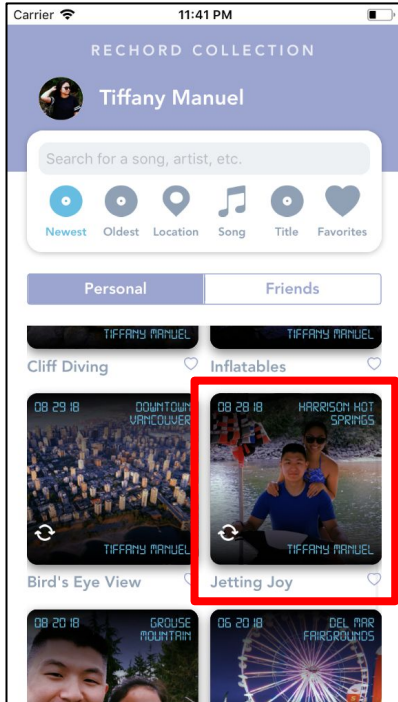
Tools Used

- React Native and Expo
- IDEs
 - Visual Studio Code
 - Sublime Text
- iOS Simulator / iPhone
- GitHub

```
App.js — cs147hifi
1 import React from 'react';
2 import { StyleSheet, Text, View, SafeAreaView,
3 import { Font } from 'expo';
4 import { createStackNavigator, createBottomTabNavigator,
5 import { setCustomText } from 'react-native-reanimated';
6
7 import * as screens from './App/Screens';
8 import { CollectionStack } from './App/Navigation';
9 import { Metrics, Images, Colors } from 'react-native-paper';
10
11 import firebase from 'firebase';
12
13 var config = {
14   apiKey: "AIzaSy0_vD_Nv5vJ46_Tsvvn9Tc",
15   authDomain: "records-7b3a3.firebaseio.com",
16   databaseURL: "https://records-7b3a3.firebaseio.com",
17   projectId: "records-7b3a3",
18   storageBucket: "records-7b3a3.appspot.com",
19   messagingSenderId: "396699023083"
20 };
21 firebase.initializeApp(config);
22
23 export default class App extends React.Component {
24
25   state = {
```

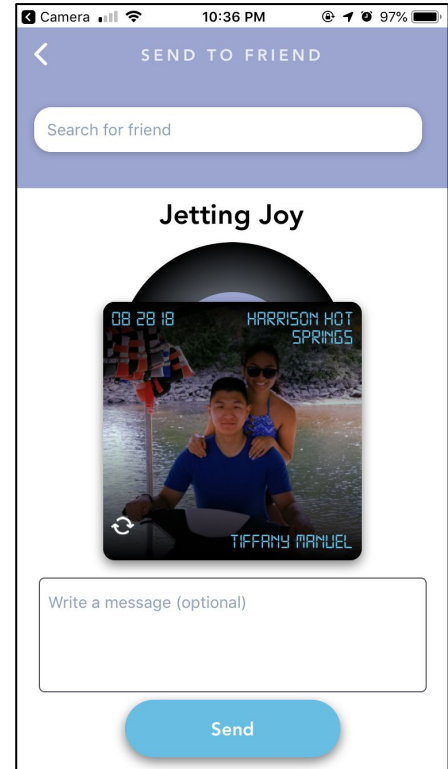
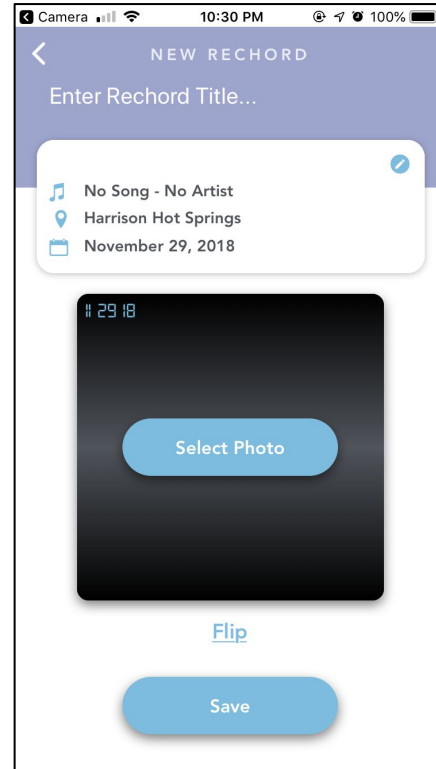
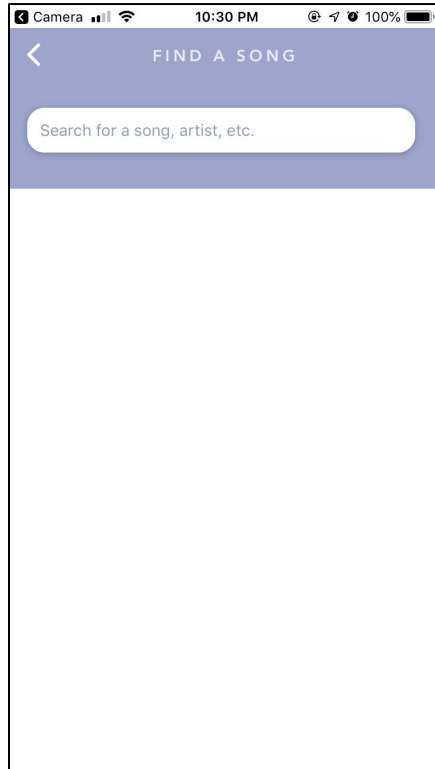
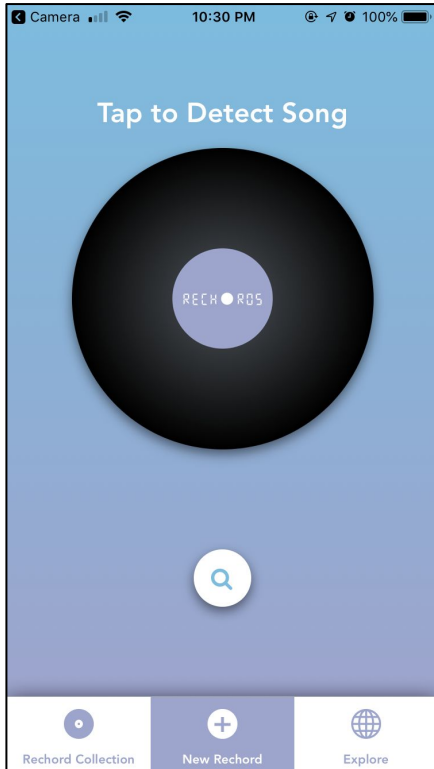


Implemented Features

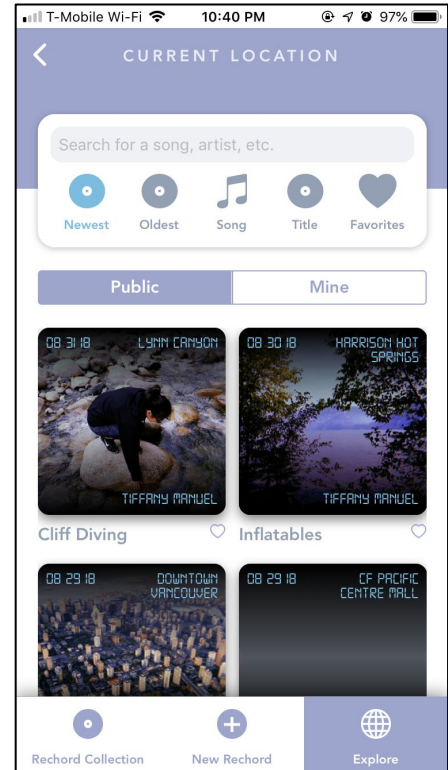
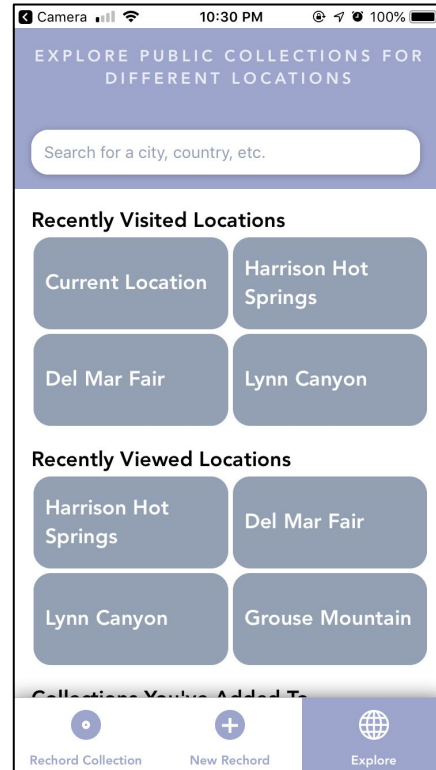
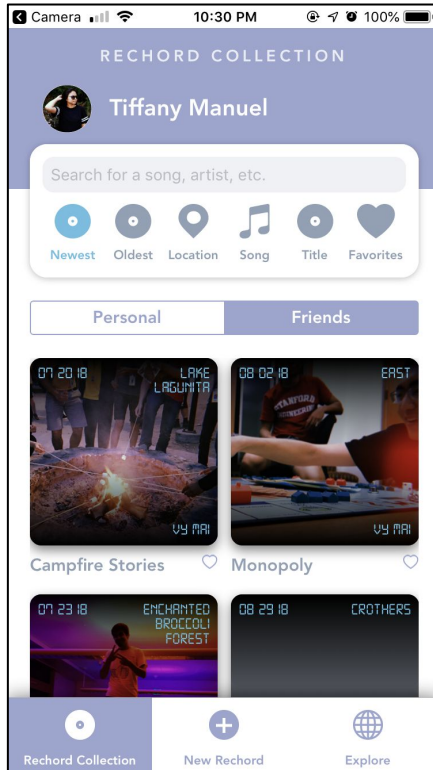
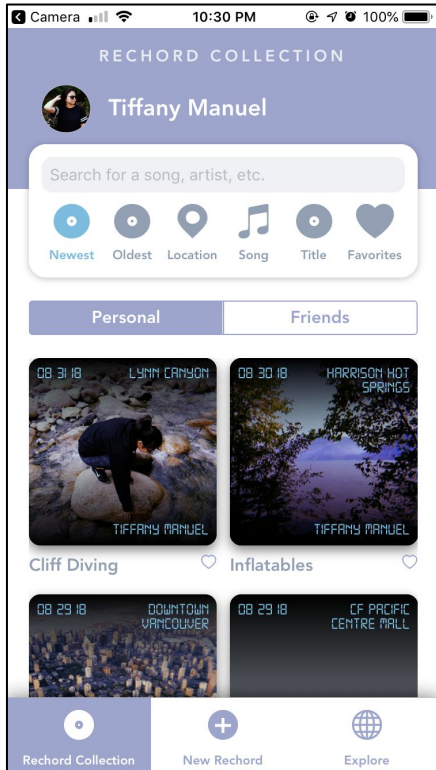


Rechord Collection / Rechord Viewer (Simple Task)

Implemented Features



Implemented Features



Unimplemented Features

ToDo

- “Create New Rechord” functionality
- Public Location Collections
 - Data
 - Adding personal rechords
- “Send to friend” functionality
- Recognize song playing

Plan

- Re-use code from similarly styled components
- Figure out database
- Use existing Spotify API

Unimplemented Features

ToDo

- “Create New Rechord” functionality
- Public Location Collections
 - Data
 - Adding personal rechords
- “Send to friend” functionality
- Recognize song playing

Plan

- Re-use code from similarly styled components
- Figure out database
- Use existing Spotify API

Hardcoded Data

- Details of records for “Friends” and “Public Location Collection” (song, location, date, owner, etc.)
- List of friends
- Sort by functionality
- Recognized song (if API plan doesn't work)

Questions / Issues

- Not sure how to use Spotify API in React Native
 - If we can't get the feature to work, should we just default to searching/manually inputting the song?

Demonstration

Summary

Summary

- Playlist Design changed to Album Design
 - Terminology changed from “playlists” to “collections”
- User given more control and freedom
 - “Undo” options
 - Editing song and location
- Most of the screens have been created
 - Few minor fixes will be made in final project
- Working on database this weekend to complete functionality
 - Primarily for creating a new record