Capstone Project

GitHub Username: broccolibird

Cheep

Description

Cheep plays bird sounds from your neighborhood and around the world. Not sure what bird songs are waking you up in the morning? Cheep let's you listen to bird sounds by location so you can learn which birds live right outside your home. Take Cheep with you on vacation and learn about the bird songs in the area everywhere you go!
All data and sounds are provided by xeno-canto.

Intended User

Cheep is intended for nature-lovers of all ages.

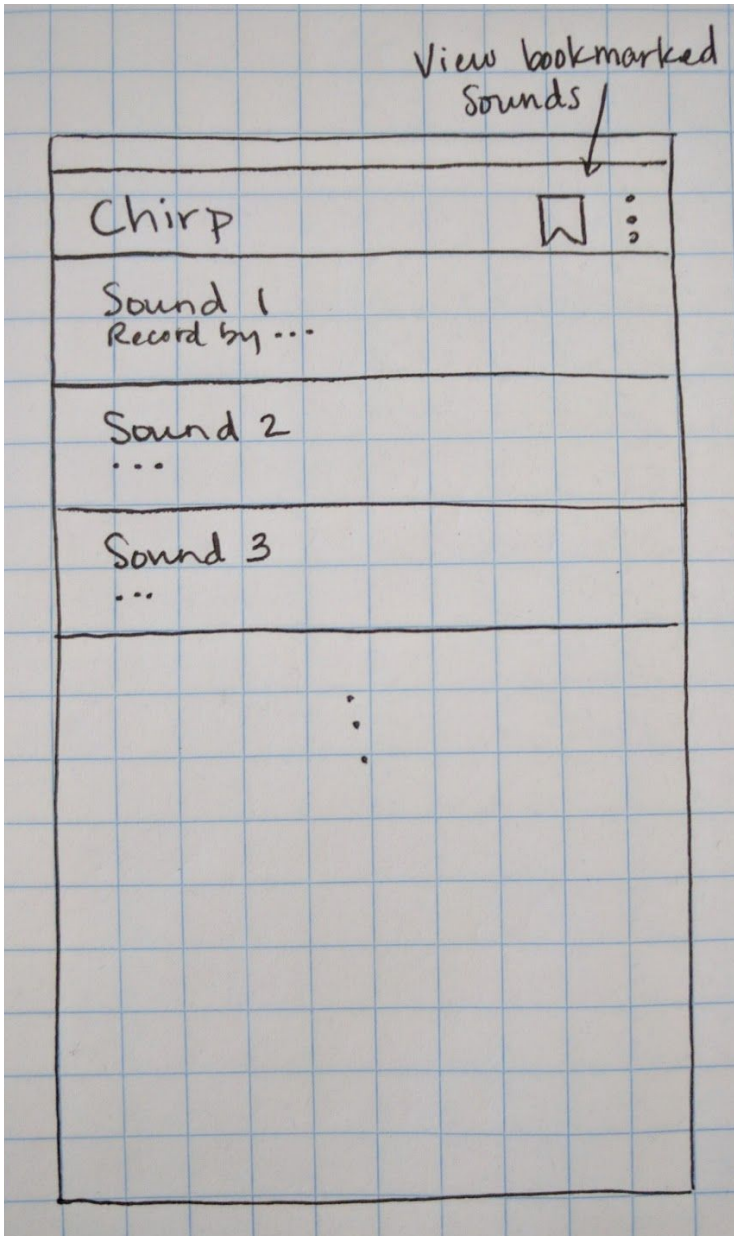
Features

- Displays a list of bird sounds recorded in an area
- Bird sound audio playback
- Cancel playback from notification
- Bookmark bird sounds to listen to later
- Display bookmarked bird sounds in widget

User Interface Mocks

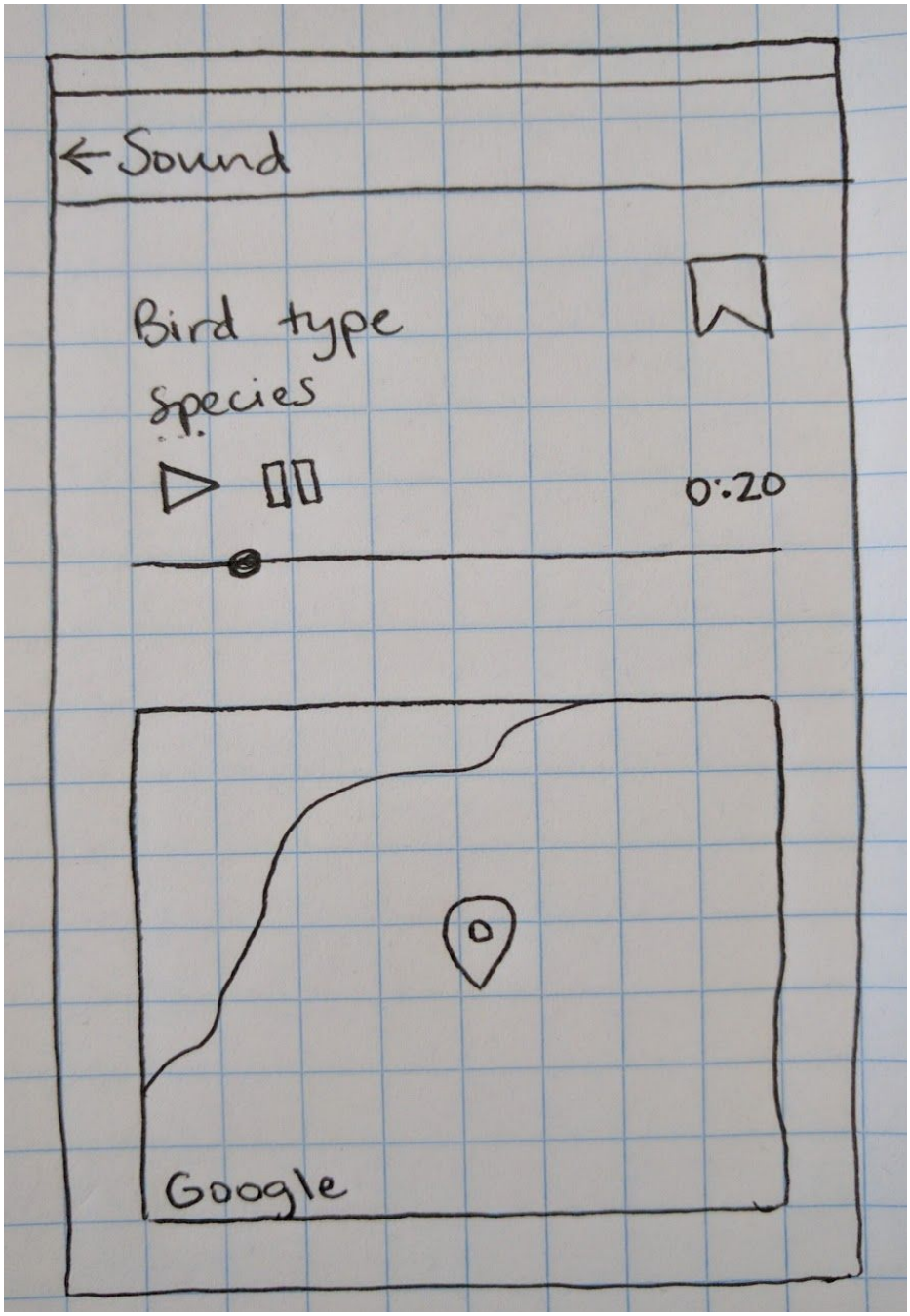
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Home



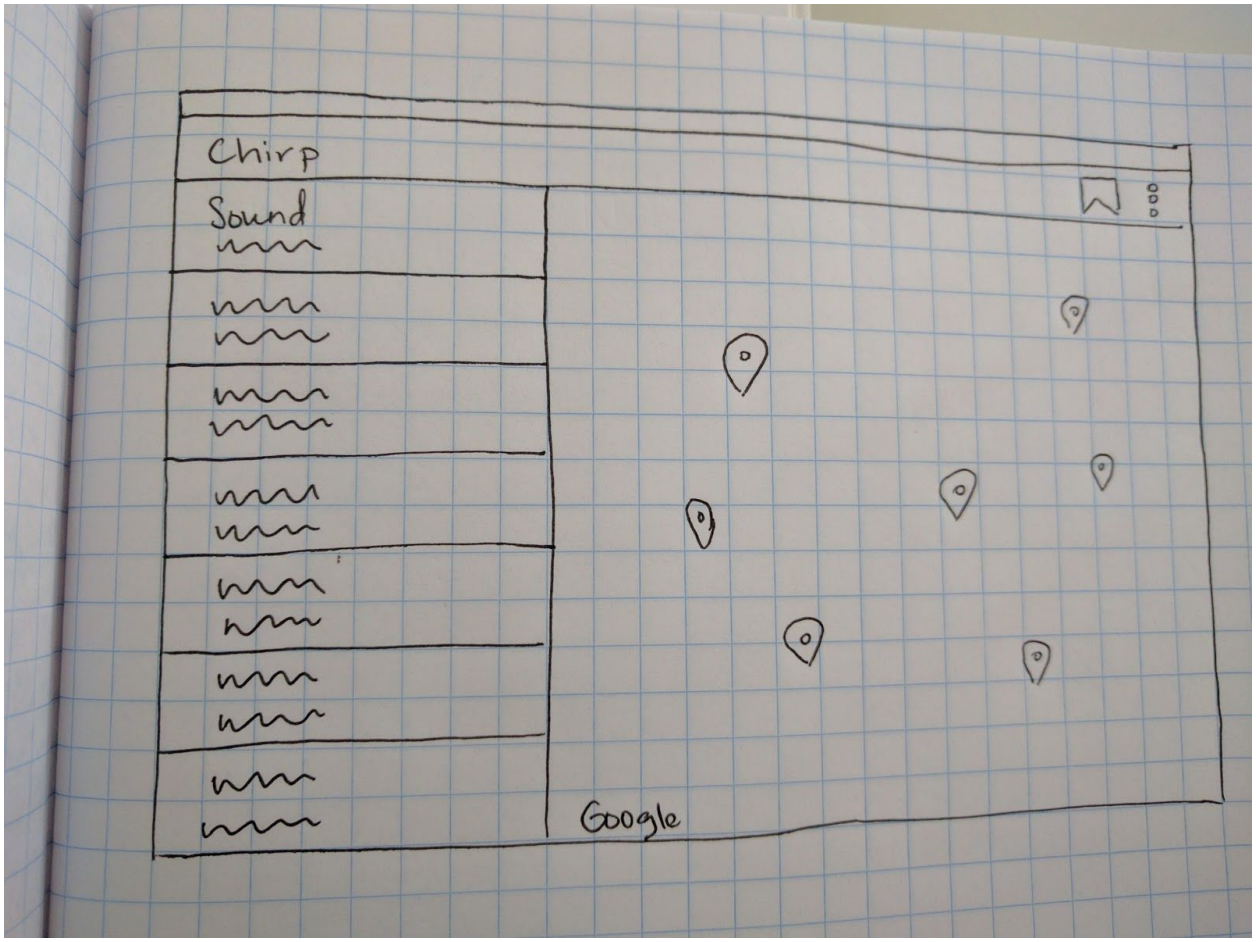
The home screen shows a list of recordings that happened near the user's current location. Clicking on a sound will take the user to the detail for that sound. The toolbar contains a bookmark to filter for bookmarked sounds.

Detail



The detail screen is where the user can play the sound that they selected. It also shows details about the sound including the type of bird that made the sound, who created the recording, and where the recording was created.

Home - Tablet



In the tablet interface, users can see the locations for all of the recordings in the list.

Key Considerations

How will your app handle data persistence?

Most data will be pulled from the [xeno-canto api] on an as-needed basis. However, data for bookmarked sounds will be stored in a SQLite database with a Content Provider to handle adding and removing data from the database.

Describe any corner cases in the UX.

Although the bird sounds are short, the user may want to cancel playback after leaving the sound detail screen. Users may do this by cancelling the sound in the notification panel.

Describe any libraries you'll be using and share your reasoning for including them.

- Retrofit - http client
    - OkHttp - used for http calls underneath of Retrofit
- Gson - json parsing
- Appcompat/ support libraries - for Material UI components

Describe how you will implement Google Play Services.

- Google Play Services: Maps - to display the locations where the sounds were recorded
- Google Play Services: Location - uses the user's location to find sounds close to them

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

- Add networking libraries
    - Retrofit
    - Gson
- Make a simple api call and log the response to show data flowing into the app.

Task 2: Implement home list to display sounds

- Add a RecyclerView to the HomeFragment
- Create a row layout to display bird sounds
- Add an Adapter to show data from api in the list

Task 3: Add Location check

- Follow this guide to get user's last known location: https://developer.android.com/training/location/retrieve-current.html
- Update the api request to use the user's last known location

Task 4: Add detail screen

- Create a second activity to display the sound detail information
- Using an intent, open the second activity when a user selects a sound from the list in the home activity
- Playback the sound when a user hits "play" on the detail screen

Task 5: Add storage

- Create a SQLite database with columns to hold bird sound data
- Create a Content Provider to give access to the database
- Add bookmark action to detail screen
- Save bookmarked sounds to database
- Remove bookmarked sounds when sound is unselected
- Add bookmark filter to homescreen

Task 6: Add Maps

- Follow the Google guide to set up the Maps API: https://developers.google.com/maps/documentation/android-api/start
- Add a Map View to the detail screen
- Add a tablet layout for the home screen and display a map there

Task 7: Add notification for audio

- Move audio playback to a bindable service
- When playing audio on the detail screen, bind to the new service
- Display a notification allowing audio playback control while the service is active

Task 8: Add widget to display bookmarked sounds

- Create a list widget to display bookmarked sounds
- When a user clicks a sound on the bookmark, start the audio service

Add as many tasks as you need to complete your app.

Submission Instructions

1. After you've completed all the sections, download this document as a PDF [File → Download as PDF]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone_Stage1.pdf**"