

JustShare

A real-time shared album



Proudly presented by Yuandong Zhang

Overview

We've all been there before, as sometimes we simply want to share photos among families and friends with our location but find no app capable of doing that.

However, with **JustShare**, a shared album based on Google map API and Google cloud database services, your dream will come true with one click of the shutter and we'll handle the rest for you!

This report, answers all your questions about how our app works, so what are you waiting for, let's dig into the technical details!



Component Rendering

Map: Upon creation of the page, the map starts rendering by calling `getMapAsync` asynchronously

Marker InfoBox: We designed a customized infobox, rendered together with the marker for info display

Marker: The markers are rendered in an iterative manner with the data pulled from cloud database

Shutter Button: We designed a shutter-like button for users to take pictures, rendered with the map together asynchronously

Photo Taking

By clicking the shutter button below, the camera will be initiated for users to take a picture

First-time users will be prompted with request for camera permission, only if the permission is granted will the camera be initiated

The user can review the image after it has been taken, if not satisfied, the user can retake ↶ or cancel ✕, otherwise, click ✓ and save the image

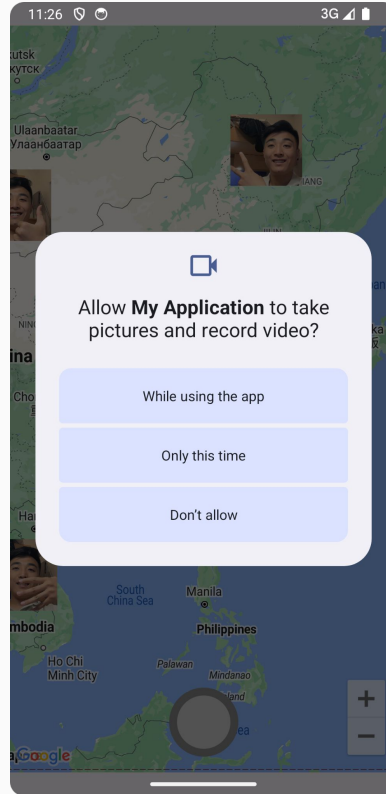
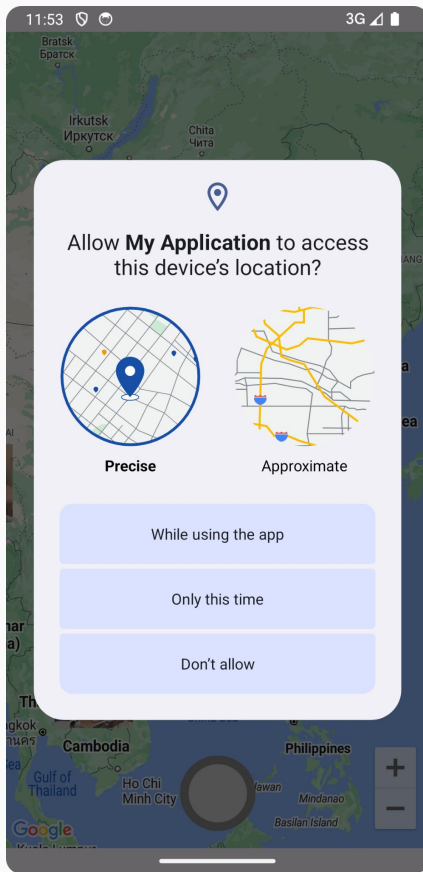


Photo Taking

After the photo uri is received with the taken photo, first-time user will be prompted with the request for their location to be accessed.

If the user granted the location access, the following will be uploaded to Google realtime database:

- the image, encoded in base64 string;
- the precise or approximate GPS coordinate of where the image was taken
- the timestamp of when the image was taken



sharedalbum ▾

Realtime Database

Data

Rules

Backups

Usage

Extensions

<https://sharedalbum-4da54-default-rtdb.firebaseio.com>

```
https://sharedalbum-4da54-default-rtdb.firebaseio.com/
└─ images
   └─ -NvGS9SC_8prHNH0b08a
      ├── imageData: "/9j/4AAQSkZJRgABAQAAQABAAQ/4gHY"
      ├── latitude: "46.17489897821003"
      ├── longitude: "126.48138329386711"
      └── timestamp: 1712911066430
   └─ -NvGT2SDfkb6stB13cI3
   └─ -NvGeU-CD-TyB000Dapr
   └─ -NvGeZ38bmJi_-GqkN3R
   └─ -NvGedJt9169Ch7UyEgX
   └─ -NvGegs4deooVQm6snR0
```

Database location: United States (us-central1)

Photo Displaying

Users will be prompted once the image is successfully uploaded and displayed on the map as a marker

A customized infobox will be displayed if the marker is clicked by the user, the infobox will display the GPS coordinate of the marker, retrieved from Google realtime database

Additionally, new photos, uploaded from different users, will be almost instantly rendered as well, as the map continuously listens the data change of the realtime database

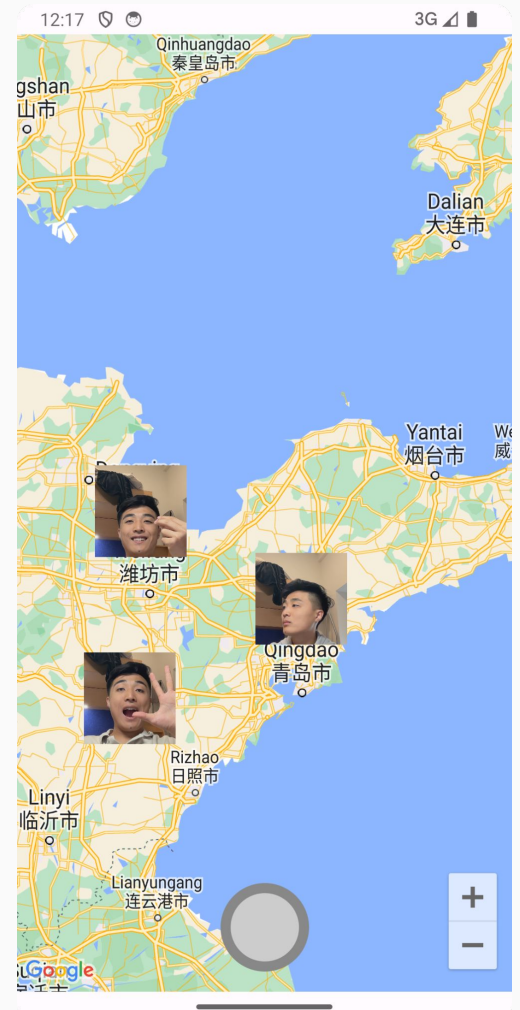


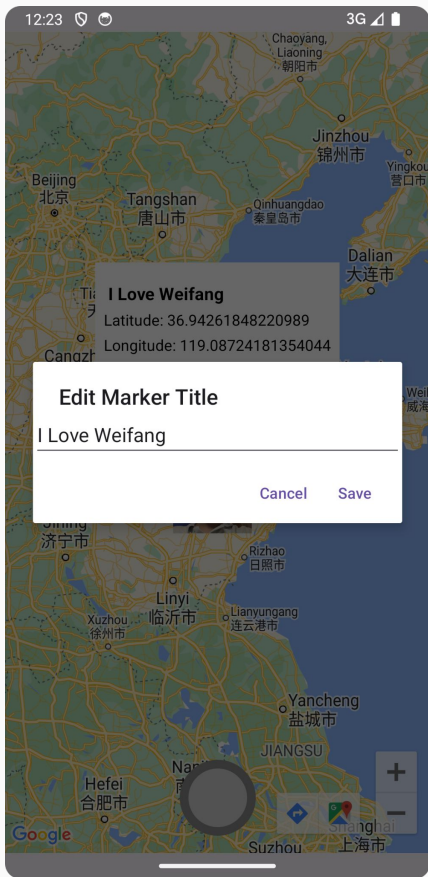
Photo Displaying

Images will automatically be clustered together with regard to different zoom levels of the map

Users can simply zoom in to de-cluster the photos and know more details, or zoom out to know the overall geographical distribution of the shared photos – all done with simple pinch in/out gestures

We've also added a zoom in/out button on the bottom right corner for accessibility of difference devices/android versions





InfoBox Customizing

We've already customized the infobox to display the latitude and longitude, however, if the user wants one step further, the title of the marker can be easily modified with one click of the infobox

Drag and Drop

If the user is not satisfied with the current location of the photo and wants to adjust it – we support it by allowing the user to drag the image, and drop it where they think fit

Of course, the updated GPS location will be uploaded to the realtime database and be displayed on the map



Future Enhancement

- Although I've used 'we' throughout this report, it is actually me soloing lol, I'll be actively looking for collaboration after CS306 and polish the idea and the project.
- The app now is mostly running on a single thread, which caused huge initialization overhead, we'll be looking into this issue and solve it with future simulations
- The map renderer is giving warnings regarding google certificate, we'll use google's new renderer in our next update

Thank You!

Code available at: https://github.com/Ydz0616/CS306_Project