Introduction

The navigation app is specifically used for helping people retrieve text from paper using the phone's camera and cloud services to extract and parse the address instead of manually enter the address. The app also has a REST API backend to help store and retrive data by the authorized user. OAuth is supported by Firebase using email and password.

Firebase Authentication

Firebase provided the authentication service and manages user accounts. Only the email/password is implemented in the app. The backend server verify with Firebase that the token is valid and process the request.

Documentation is at `https://firebase.google.com/docs/auth/android/start`

Google Cloud Vision API

Specifically the Text Detection is used to extract text from photo taken by the camera.

```
POST https://vision.googleapis.com/v1/images:annotate?key=YOUR_API_KEY
```

Documentation is at `https://cloud.google.com/vision/docs/detecting-text`

SmartyStreets API

The standardized address is not a regular language or context-free, I used the API from SmartyStreets for parsing the text return by the Text Detection.

```
POST https://us-extract.api.smartystreets.com\?auth-id=123\&auth-token=abc
```

Documentation can be found here: `https://smartystreets.com/docs/cloud/us-extract-api`

Nav-app API

The backend server uses the following data model for address note. The URL and valid verbs are GET, PATCH, DELETE for `/notes/note-id` and GET, POST, DELETE for `/notes`.

```
https://backend-dot-firebase-nav-app.appspot.com
```

Note

```
"friendsly_id":"Fred Smith",
"message":"verified",
"address":"123 Fake St, Nowhere, Earth",
"latitude":"44.5",
"longitude":"123.2"
"created":"2017-03-12 (20:40:28.764) MST",
```

1. Get all the address notes

   ```
   GET /notes
   { }
   ```

   This will return all the notes belong to the user.

[Nav-app API] continued on next page...

2. Add an adddress note

```
POST /notes
{
"name":"value",
"name2":"value2",
...
}
```

This will add a new address note to the user collection.

3. Delete all the adddress notes

```
DELETE /notes
```

4. Get an address

```
GET /notes/<note-id>
{ }
```

This will return the specified address data

5. Delete an address

```
DELETE /notes/<note-id>
```

6. Change an address

```
PATCH /notes/<note-id>
{
"name":"value",
"name2":"value2",
...
}
```

continued on next page...