Eloc Control Panel App

User Account Feature

1. Create firebase project

2. Add SHA1 for project

3. Add SHA1 for Google Play (when app is in Play Store)

4. Download and set JSON file

5. Set build.gradle files
   a) Add to top level file:
   ```
   buildscript {
       dependencies {
           classpath 'com.google.gms:google-services:4.3.15'
       }
   }
   ```
   b) Add to module level file:
   ```
   id 'com.google.gms.google-services'
   implementation platform('com.google.firebase:firebase-bom:32.0.0')
   implementation 'com.google.firebase:firebase-auth'
   implementation 'com.google.firebase:firebase-firestore'
   implementation 'com.google.firebase:firebase-storage'
   implementation 'com.google.android.gms:play-services-auth:20.5.0'
   ```

6. Project settings
   a) Project name: ELOC Control Panel (will be used in email templates)
   b) Project environment: Production
   c) Data Privacy: Disable sharing
7. Authentication
   a) Add email/password auth
   b) Add Google auth
8. Setup Google Sign-in
   a) Go to: https://console.cloud.google.com and make sure the right project is selected.
   b) Go to Oauth consent screen page and make sure everything is correct. → https://console.cloud.google.com/apis/credentials/consent
   c) Set app name, support email, app logo, home page url, privacy policy url, terms of service url (wildlifebug.com)
   d) Set scopes: userinfo.email, userinfo.profile
   e) On the Credentials page, create an Android type client ID. The package name for the sample app is com. You will also need to provide the SHA-1 hash from your signing certificate fingerprint. Remember to add the SHA-1 of the release app in the restrictions for the Android app client ID.
   f) Get resources from https://developers.google.com/identity/branding-guidelines

9. Set up firestore
    a) Set rules
    b) Set root collection - ‘accounts’
10. Set up storage
    a) Set rules
    b) Set root folder – ‘profile_pictures’