

Instructions for running the code:

Login(to see a populated FriendsList in the profile page):

- Email: daniellevi@daniel.com
- Password: 12341234
- “Ittran” account will be labeled as private to demonstrate the initial functionality. Need to implement friend check (if this person is a friend, then you can see their account no matter if its private or public. Also need to add an “Add Friend” button)

Contributions:

- Daniel Levi (30%):
 - Login and SignUpPage UI.
 - userName implementation on SignUpPage
 - Tab Bar Controller and Tab Bar implementation
 - (Bar at the bottom of the home, profile, map and settings screens)
 - Segues to each respective view controller based on what is tapped on the tab bar.
 - PostPage UI
 - Displays a selected post and its associated information (likes/comments count, the owner of the post, date posted, and location)
 - ProfilePage UI and implementation
 - userName and bioField labels update based on what is stored in Firebase (based on who is currently signed in)
 - Segmented Control that displays a gridOfPosts, mapView, or list of friends, based on what is selected on the segmentedControl
 - Posts/Friends are clickable.
 - Friends list is connected to Firestore, and when clicked maps to OtherProfilePage(where their profile is displayed)
 - When tapping on a post, you are taken to PostPage, which displays the post that was clicked on.
 - OtherProfilePage UI and implementation
 - Has all the same implementation as the ProfilePage
 -
- Isaac Thomas (30%):
 - Home Page UI and Navigation
 - Created base design for the 4 main pages (Home, Profile, Discover, and Settings)

- Worked with Daniel on Navigation through the Tab Bar Controller
 - FeedViewController, FeedPost, PostTableViewCell
 - Hard-coded functionality right now, will connect to DB to get posts from friends, sort by timestamp, and present on table view
 - Home Screen Feed
 - Created a Table View with a custom prototype cell tied to PostTableViewCell
 - Created UI design on storyboard
 - Add Post View Controller and Functionality
 - Hard Coded right now, will connect to DB to add post to list of posts under the user
- Austin Nguyen (20%):
 - Discover Page
 - Added prototypes posts data with names and coordinates
 - UISearchBar with search functionality to search posts by name
 - UIMapKit with pinned locations and zoom functionality
 - Mode View Controller
 - View controller superclass that manages dark and light mode for sub classes
 - Theme manager class
 - Handles user's preferred settings and provides synchronous alerts across all view controllers
- Ian Tran (20%):
 - LoginPage SignupPage Implementation
 - Email and password fields connect to Firebase to create and authenticate accounts
 - Implemented logic for handling empty fields
 - Settings Page UI
 - Created base design for settings page
 - Created the buttons and skeleton code for settings page framework
 - Logout button that stops firebase session
 - Implemented Privacy toggle to reflect current value in Firebase
 - FireBase Setup
 - Worked with Daniel to configure firebase for the application
 - Created Firebase database and project
 - Privacy implementation
 - Added functionality to privacy toggle inside settings UI to connect to Firebase
 - Implemented hidden profiles in OtherProfilePage if privacy is enabled

Deviations:

- We haven't fully established how we will handle storing user data, posting data, and likes/comments data. We have Firestore set up, but we are still finalizing the details on how we will actually utilize it. We specifically ran into problems with storing images, as we need an extension of Firestore to store images for things like posts/userProfilePictures. We have also contemplated using cloud storage from another service and storing links to the photos. This is pushed to Beta Release
- Currently, we have:
 - Users []
 - `userId : String`
 - `bio String`
 - `friends [String]`
 - `userId String`
 - `privacy : boolean`
- We ideally want:
 - Users []
 - `userId : String`
 - `profilePicture : String (link)`
 - `Bio : String`
 - `Posts []`
 - `postId : String`
 - `userId : String`
 - `postImage: String (link)`
 - `timestamp : Int`
 - `likeCount : Int`
 - `Comments []`
 - `commentId : String`
 - `userId : String`
 - `Text : String`
 - `timestamp : Int`
 - `Coordinates []`
 - `Latitude : Double`
 - `Longitude : Double`
 - `Friends []`
 - `userId : String`
 - `privacy : boolean`

- Friend/Private profile functionality is not fully present. If logged in with the “daniellevi@daniel.com” account, basic, initial functionality is implemented. We still need to implement a friend check as communicated above.