Building, Installing and Running:

Installing the Android Application from Android Studio:

1) Download and install Android Studio using these instructions: https://developer.android.com/studio/install

2) Select "Open an Existing Android Studio project" and point to the IdentityGuardApp folder.

3) Android Studio will begin loading the application. If you see a message which says that you are missing a version of the SDK or Build Tools, click the install link provided in the error. Ignore any requests to upgrade the projects Gradle version.

4) Turn on an Android device running Android 6.0 (Marshmallow) or later. Enable USB Debugging as described here: https://developer.android.com/studio/debug/dev-options

5) Plug in the device and press the run (or play) button to launch the app. You may be asked if you trust the computer to be used as a debugger, press the yes button

6) Make sure that the device is connected to the same network as the server

Note: The app may look distorted on smaller or lower resolution screens.

Installing the Rails Application:

1) Follow this tutorial to install the Ruby Version Monitor (RVM), stop at the section labeled "Rails", do not do "gem install rails --version 5.0.0": http://blog.teamtreehouse.com/installing-rails-5-linux

2) In the terminal now setup for login shell, navigate into the IdentityGuardApi folder

3) Type 'gem install bundler' to install the bundler

4) When the installation finishes, type 'bundle install --path ./local_gems', this will create a local gems folder and install the gems needed to use the server

5) When the installation finishes, type 'rake db:drop db:create db:migrate db:seed', this will create a new database and seed it with the client information built into the android app

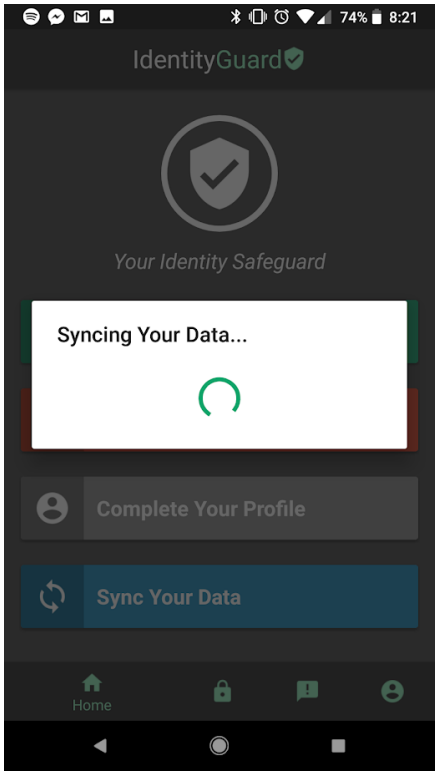
6) When the database setup is finished, type 'rails s', this will start the server in development mode

7) In another terminal window, type 'ifconfig' and record the IP Address of the machine. The app must be pointed to the server using this address

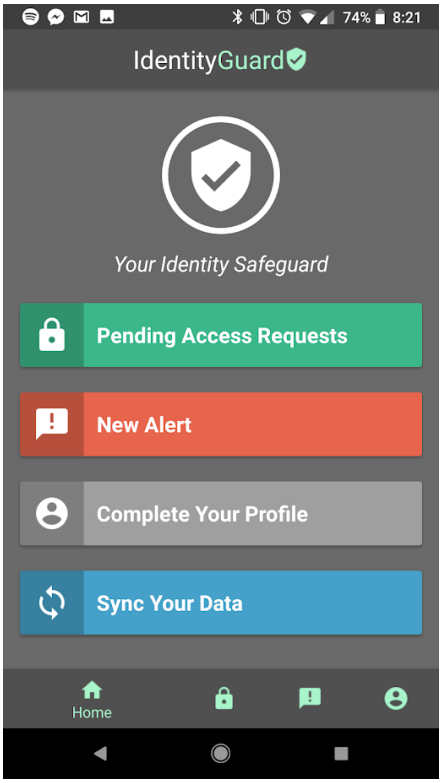
User Client Manual:

1) Begin by installing and deploying the IdentityGuardApp and IdentityGuardApi as described in the README.txt file
2) Open the IdentityGuardApp on the Android device

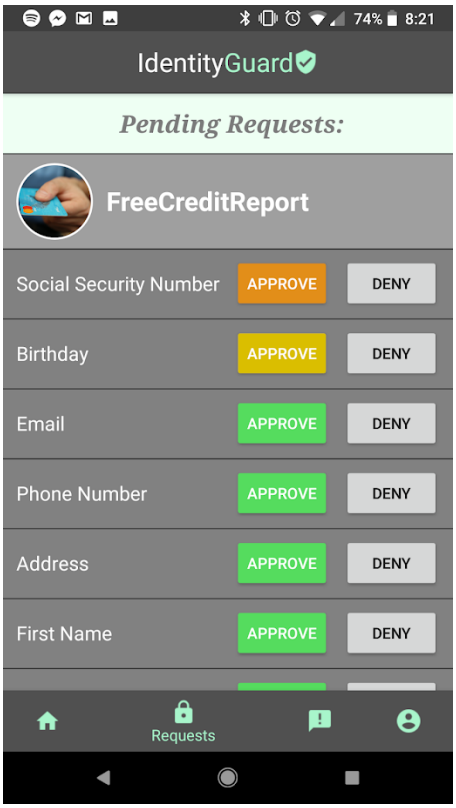
3) To use the app without connecting to the server, press **PROCEED IN MOCK MODE**. To use the app with the server, enter the IP address of the server and press **SET**.
4) The app will retrieve the data from the Real/Mock Server



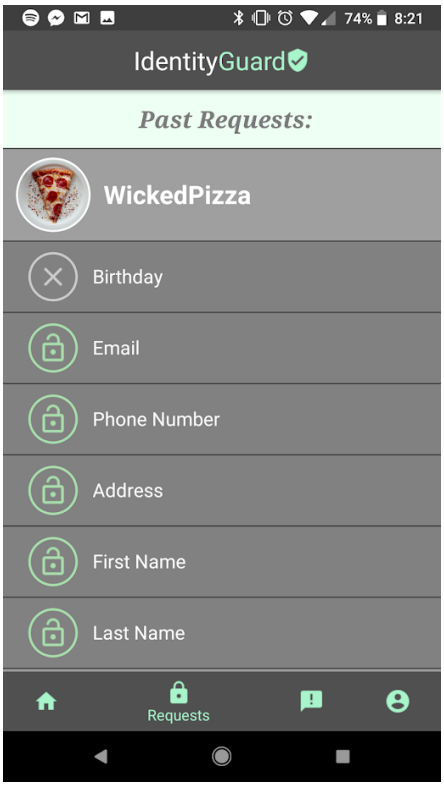
5) When the sync is complete, you will be brought to the Home screen



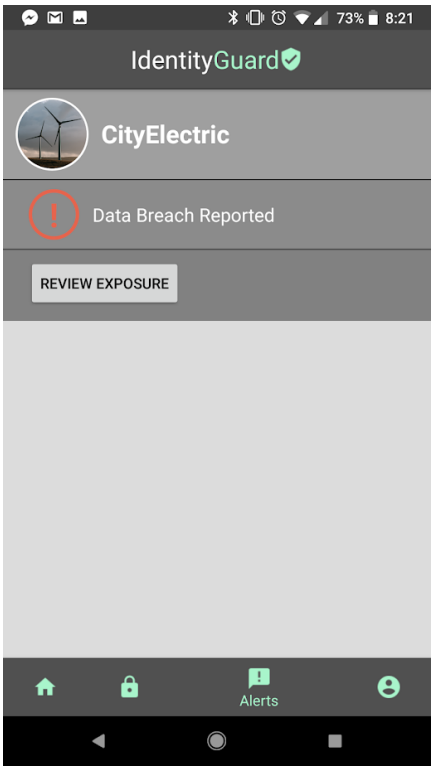
6) To be taken to the Requests screen and view clients' requests for data, press **Pending Access Requests**. To be taken to the Alerts screen and view alerts posted by clients, press **New Alert**. To be taken to the Profile screen to view stored datums or add more, press **Complete Your Profile**. To the profile again, press **Sync Your Data**. Begin by pressing **Pending Access Request**.



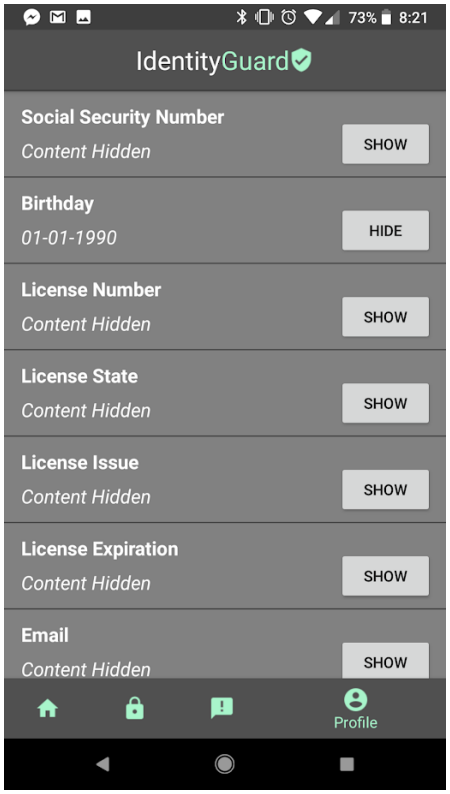
7) The Requests screen has two sections – A pending request section and a past request section. In each section, requests are grouped by the requesting client. Each row contains the name of the datum that the client is asking or asked for. In the pending request section, there are buttons to change the status of the request and either approve it or deny it. The approve buttons are colored to represent the sensitivity of data that is being requested. Pressing either will bring up a dialog to confirm the action. If the action is confirmed, a request is made to the server to update the status. Scroll down to see past requests.



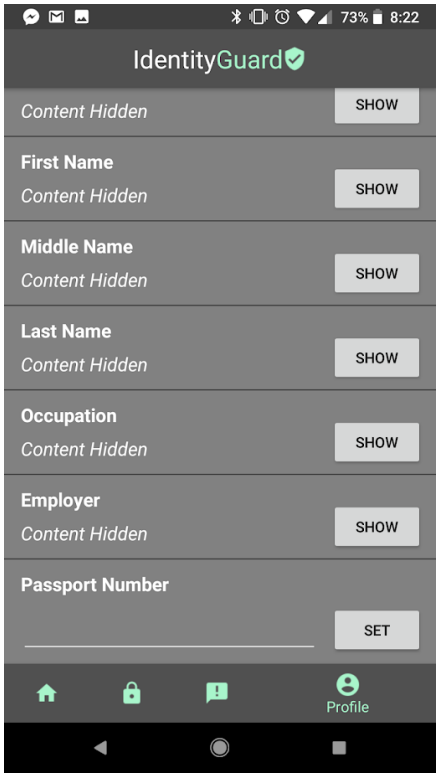
8) In the past requests section, there is an entry for each request and a status icon. The status icon color corresponds to the sensitivity of the data that was requested. If the icon is an unlocked lock, then the request was approved. If it is an “x”, then the request was denied. Next, press the Alerts button.



9) The Alerts screen contains reports from clients who have reported a data breach or accidental disclosure. The **REVIEW EXPOSURE** button navigates the user back to the Requests screen where they can look at what information they approved the client to have. Next, press the Profile Button.



10) The Profile screen contains the data that the user has provided to the Api. There are a predefined number of keys that are being accepted. If the user has a value for the key, the view defaults to displaying the datum name, hiding the datum value, and providing a **SHOW** button to toggle the visibility of the datum value. If the user does not have a value for a key, an entry view appears instead.



11) Entering a value and pressing **SET** will bring up a confirmation dialog. If the value is confirmed, it is transferred to the Api.