Understanding the Code.

This is a very high-level explanation of how the modules of the project work and what they do.

At first, the whole entity is dived into two subsystems:

1. The application, SignalApp
2. The build system, Gradle Scripts.

The build system folder contains the build.gradle file, the gradle wrapper properties file, the local properties file, and the ProGuard Rules files. The build system is used in order to automatically build the project. With the use of scripts, it knows how to deal with each and every type of file. For example, the java files are transformed into dex files. Afterwards, all the files are compressed into APK, which allows for software distribution. The ProGuard Rules are simply rules to reduce code in order to speed up the application.

The application itself, however, is divided into multiple subsystems of its own.

1. The manifests folder contains only the AndroidManifest.xml file, which contains valuable information about the app that is needed for the android system to run the app.
2. The java folder is the heart of the program and contains three folders. The first folder contains the core functionality of the application. In this project, all functionalities have a folder of their own to have a distinct organization. The second and third folders are used for testing. The first testing folder is used for android testing. As of now, it only tests attaching to the database, along with saving an attachment. It also provides a test case for secure texting. The second testing folder conveys general tests about a few different functionalities of the application, though they mostly deal with the cryptography aspect of the application.
3. The third folder, the assets folder, contains images that users can use, such as stickers and emojis, along with the user database. It also contains a font that can be used in the application.
4. The fourth folder is the jniLibs folder, which is the standard library folder for any application. These libraries contain ARMEABI and libnative libraries, which are compiled binaries matching android devices CPU architecture for the application to properly work.
5. The fifth and final folder is the res folder, which contains all the front-end UI xml files. This means that this folder is basically what the client sees when he uses the application. This also includes all possibilities the users can choose as UI, such as changing fonts, or themes of the application.