**Software Implementation and Testing Document**

**For**

**Group <Anabolix>**

Version 1.0

**Authors**:

Sam Childers

Kevin Nguyen

Tiffany Medina

Carter Reiff

Russel Lewis

# Programming Languages (5 points)

*List the programming languages use in your project, where you use them (what components of your project) and your reason for choosing them (whatever that may be).*

A majority of our work so far has been done in Android Studio. Most of the code governing interactions between the different areas of the app is written in java, but a substantial amount of material describing the appearance and location of different design elements is in XML. We also are using gradle scripts to control which libraries are implemented when the app is built.

# Platforms, APIs, Databases, and other technologies used (5 points)

*List all the platforms, APIs, Databases, and any other technologies you use in your project and where you use them (in what components of your project).*

Android Studio – official IDE for android app development, dozens of tools for writing and testing code. One item worth mentioning is the Android Nexus 5x emulator we use to test features as we implement them.

Google Maps API- Will be used to track potential and complete routes for running and biking.

Glide- Will be used to load and set images.

Material Components Library- Helps with easier implementation of bottom navigation bar. Also includes a lot of useful and/or aesthetically pleasing icons and elements which we may use later.

Firebase- Google development platform which allows for easy implementation of information-based functionality. In our case, we’re using it for authentication.

Facebook Auth. API – allows users to login/authenticate using their Facebook account.

Google Auth. API - allows app users to login/authenticate using their Google account.

# Execution-based Functional Testing (10 points)

*Describe how/if you performed functional testing for your project (i.e., tested for the* ***functional requirements*** *listed in your RD).*

# Execution-based Non-Functional Testing (10 points)

*Describe how/if you performed non-functional testing for your project (i.e., tested for the* ***non-functional requirements*** *listed in your RD).*

# Non-Execution-based Testing (10 points)

*Describe how/if you performed non-execution-based testing (such as code reviews/inspections/walkthroughs).*