To Create an companion app

1. Install Android Studio.
2. In the command prompt
    - C:\Users\Pudean>cd fitbit **(as fitbit was my initial folder for this project)**
    - C:\Users\Pudean\fitbit>mkdir fitbitandroid **(where I will download the app files)**
    - C:\Users\Pudean\fitbit>cd fitbitandroid
    - C:\Users\Pudean\fitbit\fitbitandroid> git clone https://github.com/gondwanasoft/android-fitbit-fetcher.git
    - C:\Users\Pudean\fitbit\fitbitandroid>cd android-fitbit-fetcher **(where is all the app belongings)**
3. Now open the Android studio and load the folder with 'open'
4. Open the Terminal
    - PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> ./gradlew --refresh-dependencies **(This ensures that all dependencies are downloaded and configured correctly.)**
    - PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> ./gradlew assembleDebug **(To build the debug APK, run)**
    - PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> ./gradlew assembleRelease **(If you want to build a release APK)**
    - PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> ls app/build/outputs/apk/debug/ **(You can confirm the location of the built APK by listing the output directory directly in the terminal)**
    - PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> ls app/build/outputs/apk/release/ **(For the release apk)**
5. Now turn on the developer mode of your phone and enable usb debugging
6. Return to the terminal
    - PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> C:\Users\Pudean\AppData\Local\Android\Sdk\platform-tools\adb devices **(After running this command you can see you device is connected)**
7. PS C:\Users\Pudean\fitbit\fitbitandroid\android-fitbit-fetcher> C:\Users\Pudean\AppData\Local\Android\Sdk\platform-tools\adb install -r app/build/outputs/apk/release/app-release-unsigned.apk **(In the device you can see a request to install the apk and the apk is installed after that)**

To Create the Fitbit App

1. In the command prompt
    - C:\Users\Pudean>cd fitbit **(as fitbit was my initial folder for this project)**
    - C:\Users\Pudean\fitbit>mkdir fitbitaccel **(where I will download the app files)**
    - C:\Users\Pudean\fitbit>cd fitbitaccel
    - C:\Users\Pudean\fitbit\fitbitaccel> git clone https://github.com/gondwanasoft/fitbit-accel-fetcher.git
    - C:\Users\Pudean\fitbit\fitbitaccel>cd fitbit-accel-fetcher **(where is all the app belongings)**
    - C:\Users\Pudean\fitbit\fitbitaccel\fitbit-accel-fetcher>npx fitbit **(To enter the fitbit simulator)**
    - **$fitbit>**build
    - **$fitbit>**install

Important Links For setup: 1. https://gist.github.com/princessleia1/2a0b2d9f5f8438a4ee0050d16d86ab0a
2. https://dev.fitbit.com/getting-started/

Analog to Digital Data

1. Accelerometer Sensing and Analog Signal Generation

- The accelerometer in the Fitbit device measures physical acceleration forces acting on it along its axes (X, Y, Z).
- These forces are sensed as continuous analog signals corresponding to the intensity of the acceleration.

---

2. Analog-to-Digital Conversion (ADC)

- Inside the Fitbit device, the accelerometer contains an **Analog-to-Digital Converter (ADC)** that:
    1. Samples the analog acceleration signal at a specific sampling rate (e.g., 50 Hz or 100 Hz, depending on the device and settings).
    2. Converts the analog voltage levels into discrete digital values using a specific bit depth (e.g., 12-bit, 16-bit). The resolution of the ADC determines the range and granularity of these digital values.
    3. Outputs digital data representing the acceleration along the X, Y, and Z axes at each sampled time point.

---

3. Binary Data Packaging

- The digital accelerometer data is stored in binary format on the Fitbit device. Binary storage is efficient and compact, conserving both memory and transmission bandwidth.
- Data is typically stored in batches, where each data point includes:
    - A **timestamp**: The time at which the reading was recorded.
    - Digital values for the X, Y, and Z axes.
    - Optionally, metadata such as temperature or battery status, depending on the accelerometer features.

---

4. Binary-to-Text Conversion

- When the binary data is transferred to the Fitbit companion app, it is converted to plain text in CSV format for easier processing. For instance:
    - Binary format: A compact and efficient format like [0x00A3, 0xFF02, 0x10B7].

Converted CSV format: Human-readable text such as:

```
Timestamp, X, Y, Z
428608923, -1.224, 6.588, 6.014
```

- This conversion simplifies further use in text editors or spreadsheets for analysis.