Ira A. Fulton Schools of Engineering
Arizona State University
Engineering | School of Computing, Informatics, and Decision Systems Engineering

CSE 535: Mobile Computing

Accelerometer Data Graph
Assignment 2

Purpose:

Assignment 2 will be an extension of Assignment 1. In this assignment, you will be developing a database for a patient. There are 3 parts to complete for Assignment 2. The components from Assignment 1 should be present in the submission for this assignment as well.

Technology Requirements:

- Android Studio
- Either MatLab or Python

Project Description:

Part A

In Part A of this assignment, we will extend Assignment 1 with the capability of creating **a database** for a patient (the database name should be the same name used for your application). On entering the patient name, ID, age, and sex, your app should instantiate a database with the table name **"Name_ID_Age_Sex"**.

The table will have four columns labeled:

1. Time Stamp,
2. X Values,
3. Y Values, and
4. Z Values

The app will then initiate a service that will connect to an accelerometer and collect data at a sampling frequency of 1 Hz (meaning it will receive one data per one second). The data will be stored in the table with the timestamp of a sample and the raw three axis values of the accelerometer.

The database folder name will be **"Android/Data/CSE535_ASSIGNMENT2"** in the SDCARD, which should be created by your application. The SDCARD path can be different depending on your selected environment, such as the emulator, the smartphone API, etc.

The Assignment 1 requirements should be present in your submission to illustrate the accelerometer data graph. Additionally, your app needs to draw three graphs (the X-, Y-, and Z-axis of the accelerometer data) separately on the screen, which means that the X-axis in the graph view is the time, and there are three Y-axis values (each axis data of the accelerometer sensor value – not 3D graph or sum of all the accelerometer axis values). Therefore, your app will display three different graphs based on the smartphone movement.

Part B

In Part B, we will extend Part A with the capability of uploading the database to a web server. Please create your own server either using a free AWS account or host a local server in your personal computer.

Add another button to your UI named **"Upload Database"**. When you hit that button, your database should be uploaded to the web server.

You **do not** need to implement the server-side programming. The server side is already implemented and the server is running. If you look at the provided file named **"UploadToServer.php"**, you can review how the server side works.

Part C

Add another button in the UI named **"Download Database"**. When you hit that button, the app should download the data (if any) from the web server and plot the last 10 seconds of the data in the graph. (**Note:** Do not copy it from saved database file.) The download path should be **"Android/Data/CSE535_ASSIGNMENT2_DOWN"** in the SDCARD.

Demo Video

There are two parts required for the demo video submission: (1) the application demonstration and (2) the source code description. Submit only **one (1)** video link – both parts should be included in one video.

1. Application demonstration (maximum 2 minutes in length)
    a. Begin recording from when you start the application. Show all of the required functions including the exception handling, such as clicking the RUN button multiple times.
2. Source description (maximum 5 minutes in length)

a. Show the overall UI.

b. Describe each UI component functionality by showing your java source code (not XML). Explain how it works based on the source code. Then describe the reason why you chose the functionality (i.e. it takes less time and space complexity compared to another method, or it is a unique way to avoid some specific cases, etc.).

In the video demo, you should show all of the requirements from Parts A-C, including whether your app creates the database folder or not, whether your app creates the download folder and succeeds in downloading the file or not, etc.

Submission:

- Individual submission
- Please upload (1) **the zipped source code folder** from your Android Studio workspace and (2) **the readme.txt file**.
    - In the **readme.txt file**, you should include the demo video link (YouTube) and your name.
    - In the source code of your app, acknowledge any reused code and mark what is your own code. **These comments are required.**
- Before submitting your assignment, please double-check whether the **"app-debug.apk**" file is within the project folders as follows: **app > build > output > apk**. If the file is not within the project folders, please build your project at least one time before zipping. After building your project, you will be able to see the apk file. Do not change anything for the **"app-debug.apk"** file (e.g. do not change the name of the file).
- All of the source code files should be attached to the specific comments acknowledging any reused code and your own code.
- **The application and Android Studio project name should be your first initial and last name.**
    - E.g. Alice Jones would be "**AJones**" for the application and project name.
- To submit the demo video link, you should create a YouTube account and upload the required video for the assignment to that account.

Grading Rubric:

Testing Environment for Grading

If possible, please use one of the environment settings listed below to avoid receiving the wrong grade. Occasionally, a source code will not work correctly when the development environment is different from the listed environments. If this occurs, we are not able to give you the correct grade because it is not easy to find the reason why your application failed.

- Pixel 8
- API 29

This assignment is worth a total of 100 points. The “Grade Deductions” column represents the number of points that will be deducted based on the condition in the “Condition” column.

Condition	**Grade Deductions**
● The app runs without being unresponsive ● The app can fill patient data ● The app successfully uploads the DB file ● The app correctly created the DB file	**0**
The app is unresponsive at any stage	**50**
The three graphs do not show	**50**
The app cannot fill patient data	**30**
Unable to hit the RUN button multiple times	**30**
Unable to hit the STOP button multiple times	**30**
The graph has the correct layout but no data	**40**
The graph contains data but it does not update	**25**
The graph is missing axis labels	**20**
The app fails to upload or download	**50**

The wrong DB was created	**50**
The wrong folder was created or it was not created at all	**30**

FAQs:

Many students who previously took CSE 535 asked this question regarding this assignment.

Question 1: Is there any particular format in which we need to upload the database?

Yes, Dr. Banerjee will show the source code and how to create the DB file. Based on the example, you can create the DB file using sqlite, and the DB file should be uploaded to receive full credit.