Term Project Milestone 2

Zhongxuan Song

Getting Permission of Sensors in IOS devices:

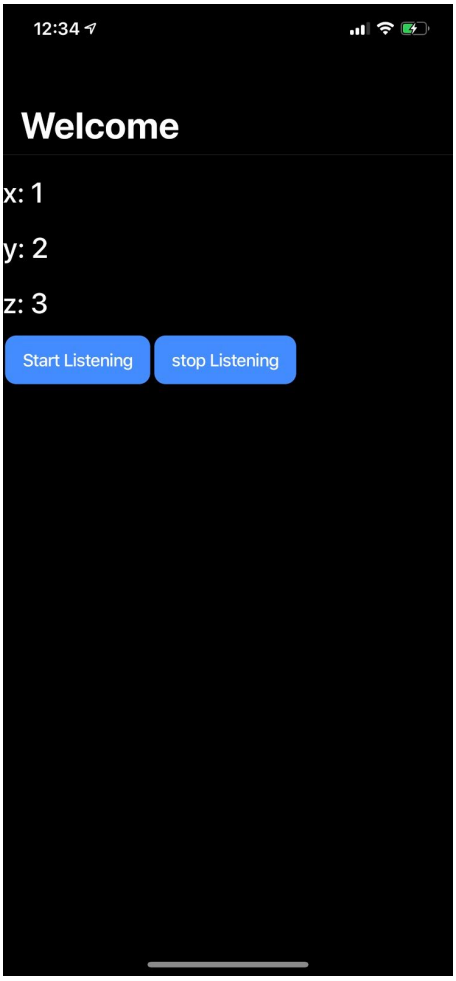
One of the main tasks of this term project is to figure out how to get permission to access the built-in accelerator in an IOS device. After doing research, I implement the project by adopting **ionic-native**, which is a framework that can provide native access to the IOS sensors without worrying about the restrictions of the permission.

Scenario:

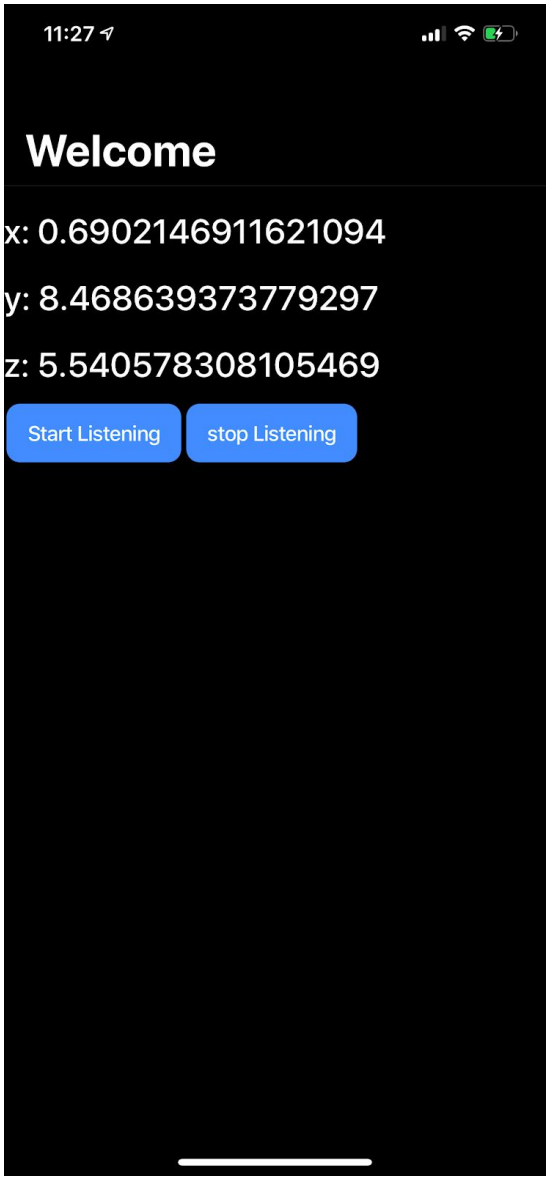
When the user clicks the start monitoring button on the homepage of the app, the app will start to detect the accelerator. If the phone drops on the floor, the alert window will pop up and ask whether this is triggered by a mistake or the user needs help. If the user selects cancel, then the alert will get dismissed. If the user clicks the 'help' button, the app will send out the help message to the SIS server. At the same time, the alert will count down 10 seconds to let the user response. If the user did not do any action in 10 seconds, then it will automatically send out the help message to the SIS server.

Screenshots:

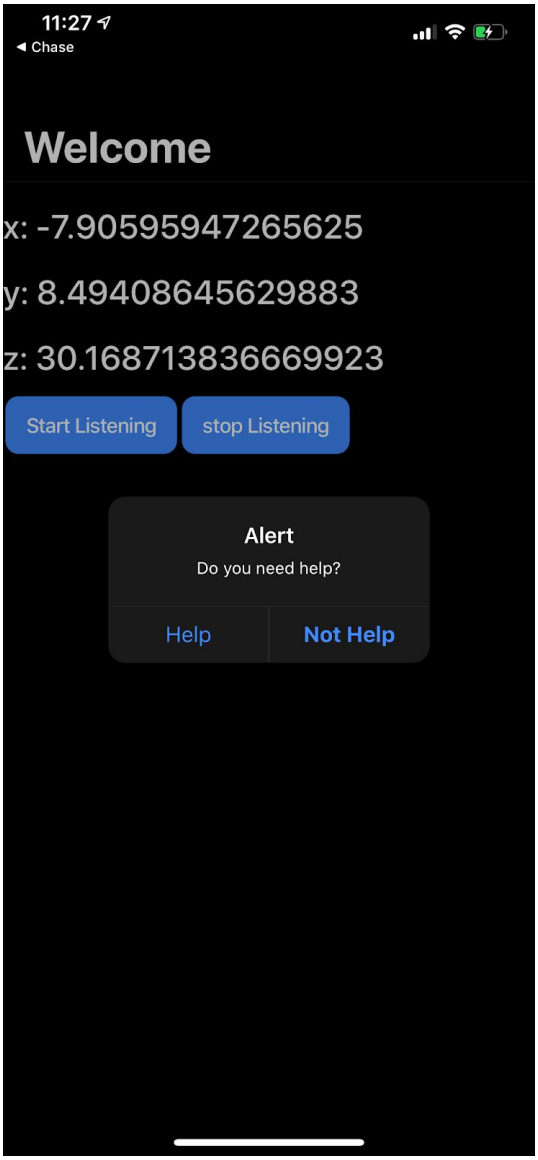
1. Default home page of the app



2. When the user clicks the “Start Listening”, the app will immediately start to read the data from the accelerator.



3. When the user falls, the alert will pop on the screen.



4. When the user clicks the “Help” button, it will send out the message to the SIS server.

Algorithm of detecting falling

The design of detecting falling off is by comparing the data of the accelerator with the threshold values. After the real-world experiment, I found out that the static value of the Z-axis is about 9.8m/s^2 when the phone is flat on the floor. So the acceleration of falling should be 9.8 + x. The current threshold is set 12.5. Although the alert could be triggered by accident occasionally, it will always successfully detect falling.

Tasks TO DO:

1. Implement the cont down in the app.
2. Do more experience on the drop threshold to make the app work more realistic.

Timeline:

1. March 24: turn in Milestone 1.
2. March 25 - April 6: making any necessary modification on Milestone 1 based on the feedback, and starting to implement the application.

3. April 7: turn in Milestone 2.
4. April 8 - 13: Make the application work as expected and connected with the SIS server.
5. April 15 - 20: fix any bug and issues and polish the design.
6. April 21: Present the application.