SENG330 Assignment 3 Design Document

Our team repo is: https://github.com/SENG330/assn3-partner-wanted

Use **gradlew run** to run application. Default admin username: “admin”, regular user: “reguser”; password is “password” for both.

We are targeting the following acceptance tests:-

- Scenario A: 1, 2, 3, 4, 5
- Scenario B: 1, 2, 3, 4, 5, 6
- Scenario D: 1, 2, 3, 4, 5
- Scenario E: 1, 2
- Scenario F: 1, 2, 3

In our assignment 3 design document, we will go through our MVC pattern and application structure.

All the Java classes from Assignment 2 (Hub, Devices, Clients, etc.), and some additional ones(User) act as the **Models** for our application.

FXML files generated with Scene Builder act as our **View**. Each controller corresponds(approximately) to each view.

Controller classes draw the GUI, and detect and handle user requests.

Data is shared between the controllers by passing instances of the Client and Hub between them (facilitated by the **HubInstance** and **ClientInstance** classes). For offline storage, our application serializes Device objects and the Client object to JSON and stores them in json files. Logs are stored in a txt file.

- The main application file is **Main.java**. This file loads the data from the json files and passes control off to the login screen.
- The login screen accepts credentials and depending on if the credentials are of an admin or not, it passes control to the **AdminController** or the **UserController**.
- The **AdminController/View** has three major widgets. The **LogTable**, which displays previous activity in reverse chronological order, the **DeviceTable**, which displays all registered devices, and the **UserTable**, which displays all registered users. The admin can add or remove users and devices, as well as clear the log.

- Adding and removing of users and devices happens in their own respective controllers, but the clearing of the log is a method in the **AdminController**.
- The admin can use devices by selecting one of them from the **DeviceTable** and clicking on the **Launch Selected Device** button. This passes control to a device controller for the specific kind of device that was selected. The **DeviceInstance** class facilitates sharing device objects between the **Hub/AdminController** OR **UserController/DeviceController.**
- The **UserController/View** has only the DeviceTable. It is identical to the one from the **AdminController/View**, except the user can only use the devices and not manage them.
- Whenever the Main Window of the application is closed, it saves all data before exiting.