# Final Project: Phase I

## Due: Nov 4, 2019

Implement an app named “***49er Sense***” on your Android mobile device, that closely models the CPI inTouch app: <https://play.google.com/store/apps/details?id=com.alarm.alarmmobile.android.cpi&hl=en>

In addition to the sensors and actuators shown in the CPI app, 49er Sense should be built for homes with the following additional features:

1. The home is also a renewable power generator.
2. Electric power consuming appliances should be part of the home and its IoT.
3. The App should be accessible to a Utility company which can control the appliances to optimize their schedules (subject to customer constraints) and reduce cost of operation to the consumer (assume that power cost varies with time).

There are 2 types of users for this App: Consumers (who could be simultaneously power producers), and the Utility company.

The app should interact with your backend LAMPP server (running on your laptop) – the backend server has the user DB implemented on MySQL. There are 2 such backend LAMPP servers: one for the Consumer (user home) and one for the Utility Company.

The IoT in the user home consists of the Laptop (backend server), 2 Raspberry Pi boards (corresponding to 2 floors in the home – each RPi board implements all sensor and actuator devices on a single floor, except the camera), a mobile phone/tablet with a Camera (as a security webcam), and a Router.

For this Phase I of the final project however, the IoT does not YET exist. So implement the RPi boards, the Router, and the Webcam as separate Java Classes with their own MySQL databases. These DBs will be stored on actual hardware once they are integrated in the IoT for later phases of the final project. In this phased implementation, the server php codes will execute the correct .java code based on user request.

The Utility company will be represented by a LAMPP server which will interact with the App on the user mobile device and with the Consumer backend server.



The app also displays:

1. The home total energy consumption and the breakdown by categories
2. The weather report for today and next 5 days
3. Video feed on request
4. Account settings
5. Notifications for user from the Consumer and Utility Company backend servers (using an Inbox)

Re (1): Each device is requested by the server for its energy consumption; the numbers are added by category; the results are reported to the app.

Re (2): Weather report is pulled by the server at the user’s home location from the cloud, and sent to the app.

Re (3): The video feed from the webcam is channeled through the router to the server to the app.

Re (4) & (5): Direct communication between server and app.

**Deliverables:**

1. App on your mobile device
2. Htdocs php codes on your servers (Consumer and Utility Company)
3. Java codes for sensors/actuators
4. All databases
5. Webcam implementation code