**BIG DATA AND MOBILE DEVICES**

**ECGR 6090-M01**

**Fall 2015**

**Kruti Satish ­ 800820958** [**ksatish1@uncc.edu**](mailto:ksatish1@uncc.edu)

**Shyaam Kumar Krishnamurthy ­ 9803378910 [skrish14@uncc.edu](mailto:skrish14@uncc.edu)**

**Objective**

In this project we demonstrate a smart grid system where in multiple servers coordinatewith an android application which enables consumers to schedule their appliances at different time of the day, as per their convenience. The project also aims at reducing the overall peak demand at the service provider as well an overall reduction of cost for the consumer.

**Detailed process**

The project is comprised of three modules. The android application, the Home server and the Utility server. The android application takes all the details of an appliance from the consumer and relays it to the Home server and the Utility server.

1. The android app
2. Home Server
3. Utility server

The communication between the mobile app, home server and the utility server is as follows.

* The android app has buttons to view, edit and delete several appliances. In addition to these there are options to view schedule after optimizations have been made on the Home as well as the Utility server.
* The user enters the details of the appliances that are to be scheduled. The details include the appliance name, wattage it uses, start time, stop time and the constraints of the appliance. This information is sent to the home server where the appliances are scheduled according to greedy algorithm.
* The RPS is subtracted from the schedule and another schedule is prepared to be sent to the utility. The schedule sent to the utility is the one that cannot be handled by the RPS.
* Once the utility receiver the schedule a Hadoop map reduce is started. The aim of the map reduce is to add the schedules from many users. The output from the Hadoop is the total power consumption for each hour for all users.
* This is compared to a randomly generated threshold. If the total power consumed is more than the threshold then the users have to pay more.
* So we have an algorithm in which, for each of the hours the power consumed by the users is more than the threshold the price is calculated by multiplying the price with the excess wattage.
* The mobile app displays the excess wattage and price that the consumer has to pay and asks if the consumer is willing to pay the extra price and maintain the schedule. Or the user can rescheduled.
* If the user decides to reschedule a new schedule is prepared according to the algorithm below. Else the user can agree to pay the extra cost and run the appliance according to the schedule already prepared.
* If the user chooses to reschedule, then a new schedule is prepared. Based on the difference between the power consumed by an appliance and the permissible power consumption threshold for the hour, the algorithm finds the lowest price from the list of day ahead prices per hour and places the power consumed by that appliance there such that the overall peak demand is decreased and there is a reduction in overall price for that customer. The method to replace the excess power to another time with the objective to decrease the peak demand is called valley filling.
* The data thus processed is then provided to the app which prompts the user to either accept or reject the schedule again.
* When the consumer commits to this pricing information is sent to the utility which does the necessary scheduling in reality.