

**ECGR-6090 – M01**

**Mobile Devices and Big Data**

***Final Project Report:***

***Big Data and Mobile Devices in Smart Grids***

Instructor:

***Dr. Ari Mukherjee***

Submitted By:

***Aiman Tarabichi   
Rahul Kasula***

**Date Due: 12/8/2015**

**Submitted To:**

**TA Prasad Iyengar Date Submitted:   
12/8/2015**

**Table of Content**

Introduction …............................................................................................. 3

System Architecture …................................................................................ 3

Future Work …....................................................................................... 4

Results …..................................................................................................... 4

Conclusion …............................................................................................... 10

**Introduction**:

In a grid computing environment, dynamically and geographically distributed sites, makes task scheduling problems challenging to solve. It is hard for a site to obtain precise real-time information about other sites whose load and computing resources may change dynamically. Moreover, the large number of individual resources of grid platforms raises scalability issues.

In this project we consider the environment of multiple independent tasks arriving at distribution site. Reducing peak demand to achieve efficient energy production and delivery.

Home consumers have the ability to use their mobile device to add details regarding their appliance models and duration of use and the desired time of use. The system considers renewable power sources at the consumer home/office as well.

**System Architecture:**

**Mobile application**: the user logs in using his credentials then starts to add appliances. Details including appliance type, power consumption, and duration of use and time of use are needed. For this project we only have one user that will be sending scheduling requests. User logins is scalable so if it works with one user it will work with many more with some modification to the code. We also assume a continuous renewable power source at the consumer home/office

The utility server gives an optimized level schedule and sends it back to the consumer, if the consumer likes the schedule, he’ll be able to accept it. Furthermore, if they don’t agree on the schedule they have to send another request by editing the appliance time again.

No penalty is assumed by the utility if the user wants to change the proposed schedule or reject it which causes a levalized cost for all users.

**Home server**: The home server can use mobile computing power to relive the server from stress and improve response time and user privacy. Every house solves the algorithm and the house load scheduling result get broadcasted to the utility server.

**Utility server:** Have a global view of all home servers (consumers’ appliances) and the grid. It takes all scheduling requests from the home servers and runs the simulated annealing algorithm to get optimized appliance schedules in real time. (Ideally map-reduce on multi node hadoop should be used to compensate large number of requests in a short time)

All appliances have a total energy demand that must be satisfied over all its scheduled instances

We are assuming that some users have renewable sources connected to the consumer home/office. The system checks if the energy required by the consumer can be fulfilled by the power produced by the renewable power source then there won’t be any need to send a schedule request to the utility. By doing that we can save a lot of computation and resources, optimize the renewable source connected hence saving the consumer on their bill.

If the demand of the consumer is larger than the power produced then the request will go through the home server and updates the tables in MySQL. Optimized scheduling details gets updated every 5 min, and results get stored in the temp table on the home server. And the user uses a 1 hour interval period to schedule appliances.

**Future work:**

Implement map-reduce function with multi node hadoop environment.

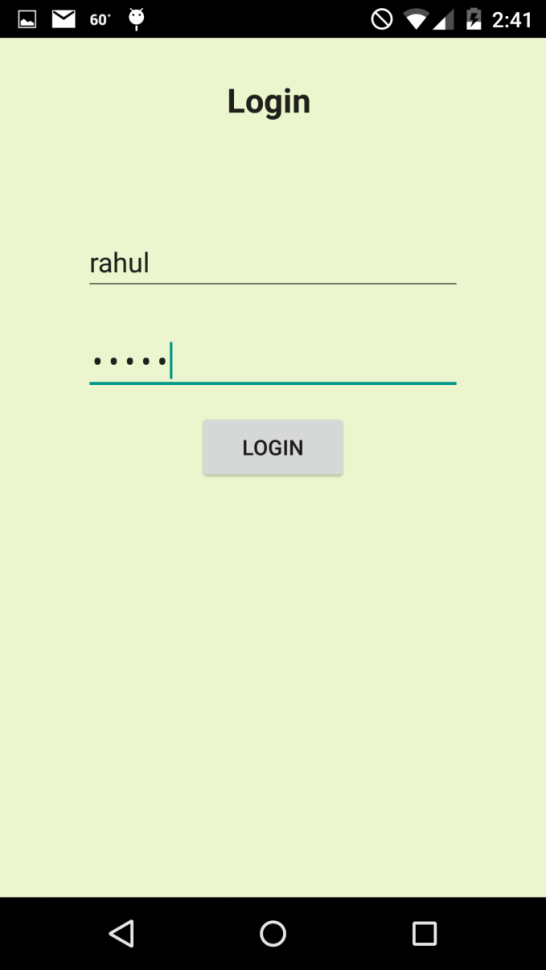
Accept or reject utility schedule

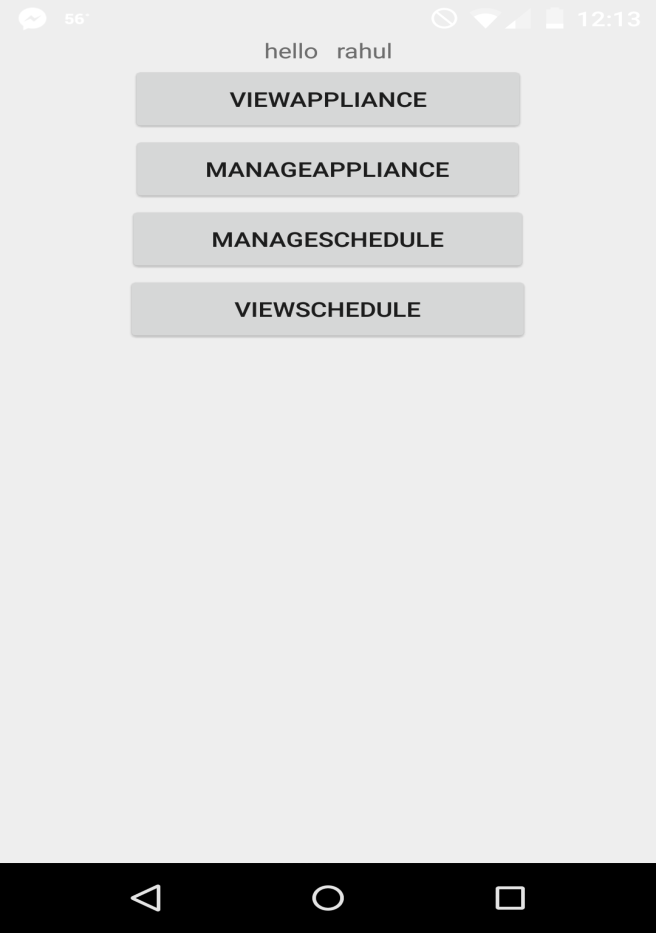
Sign up capability

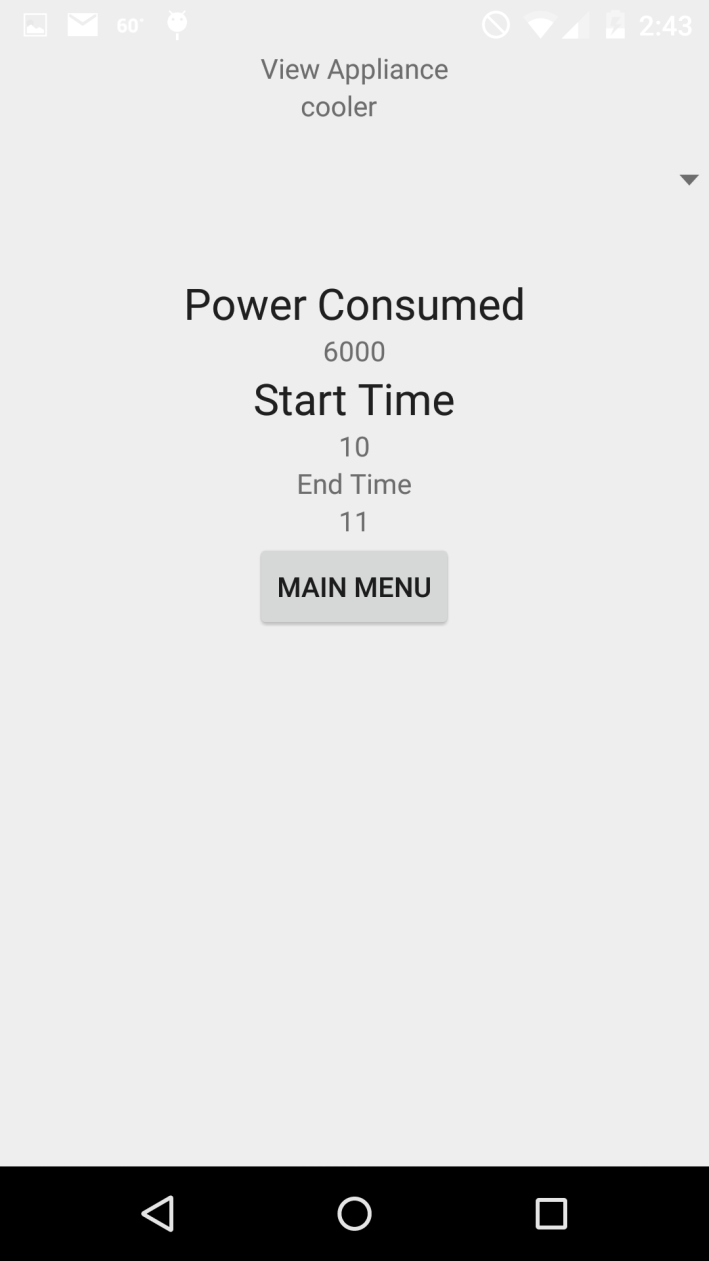
Consider dynamic renewable power source

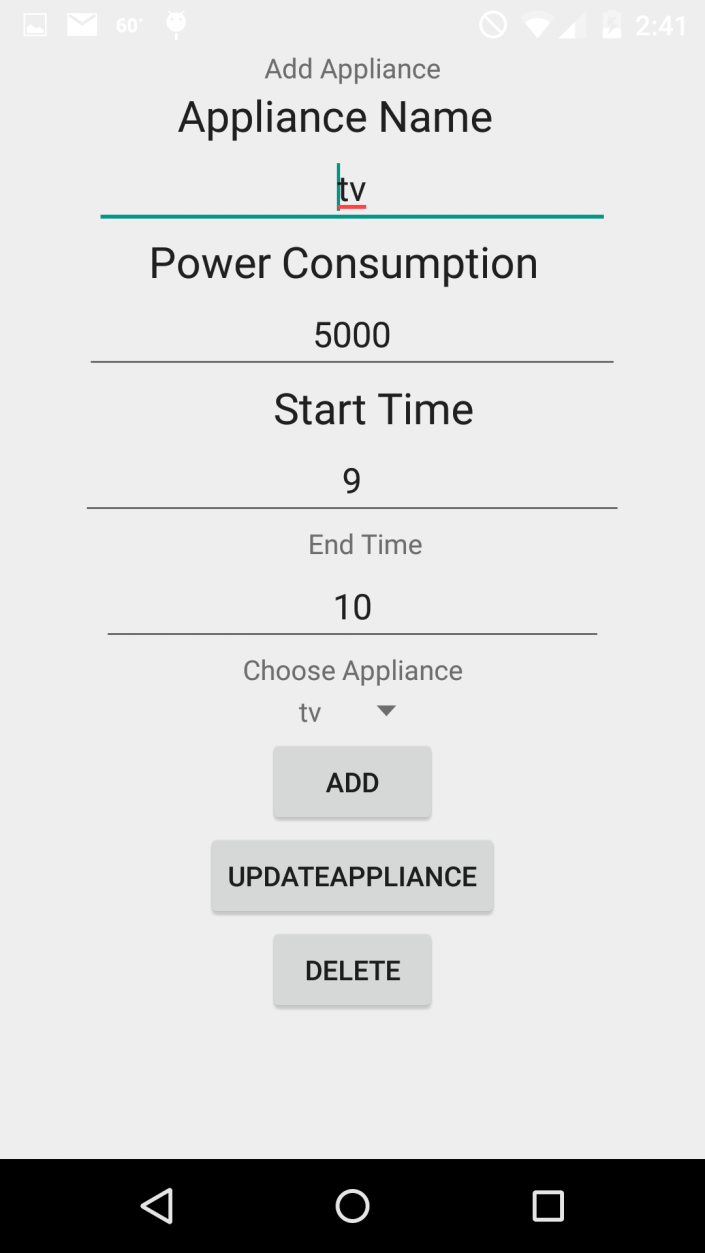
Incentive/ penalization of decisions

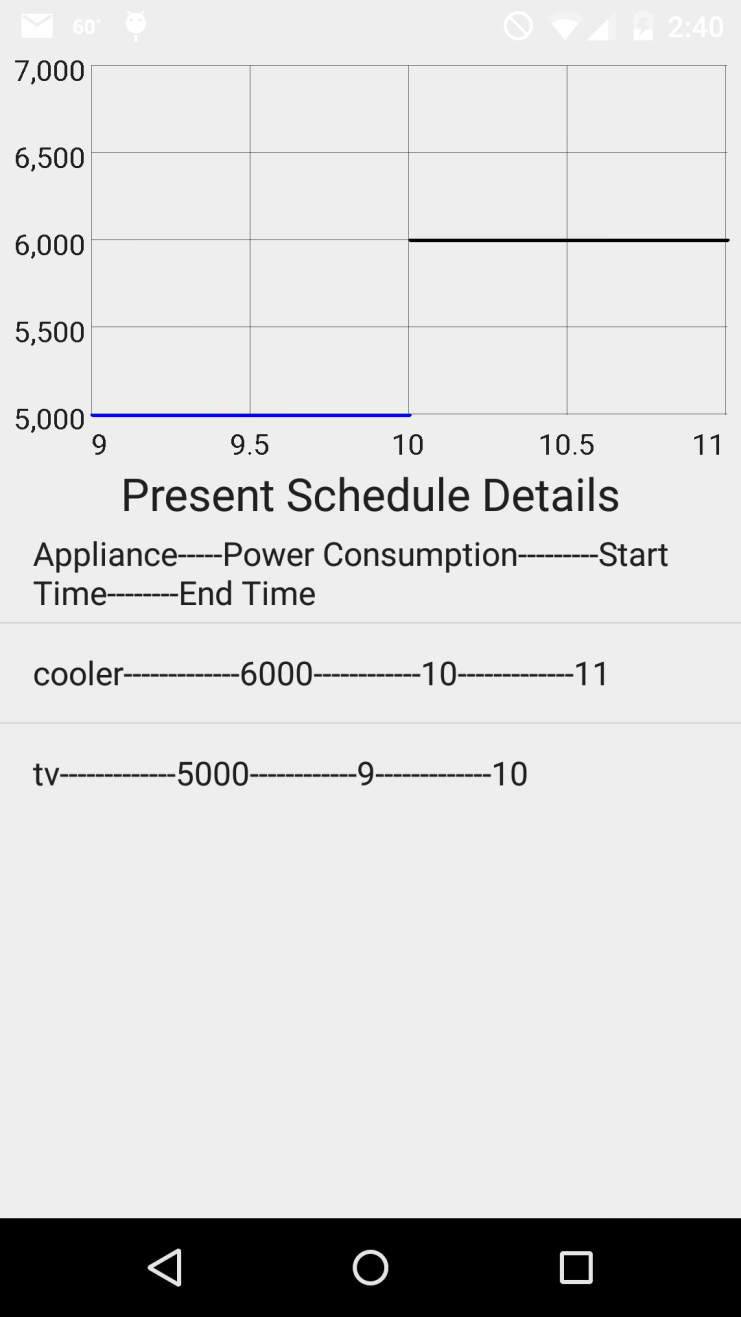
**Results**:

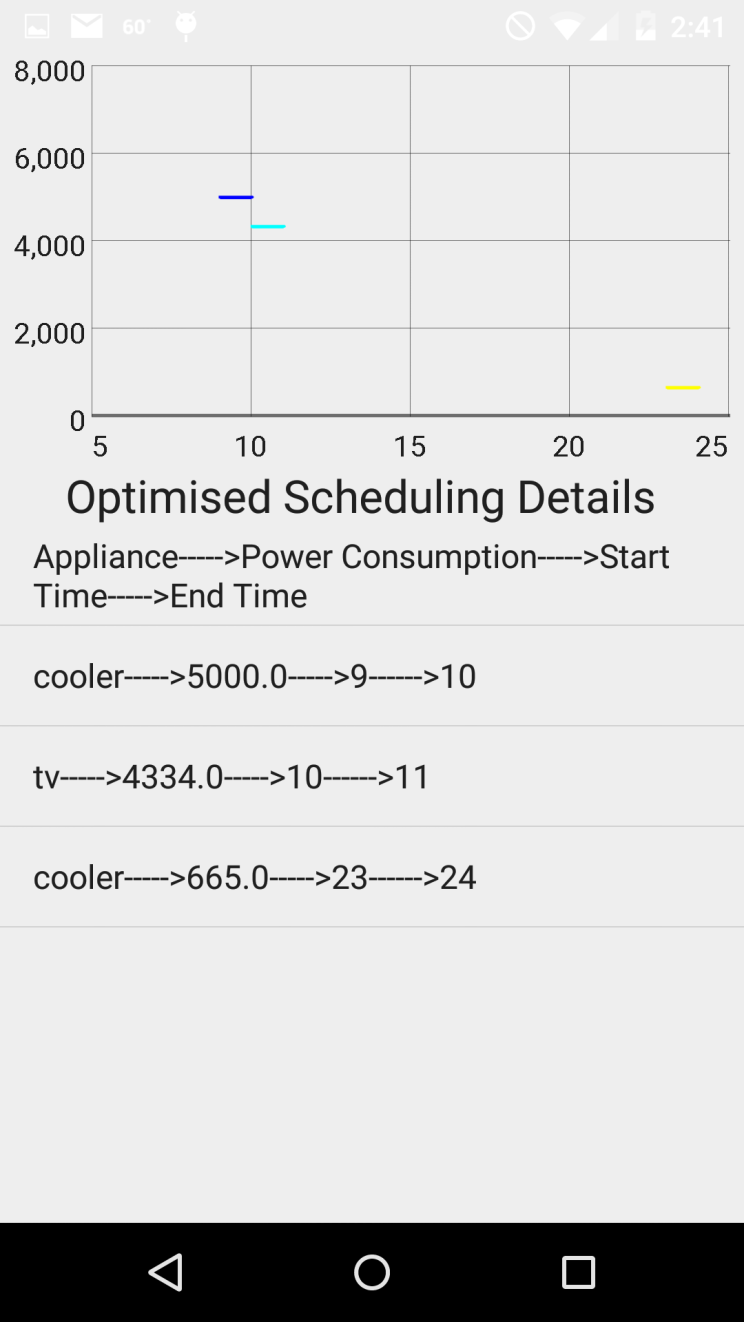
  
Figure 1: Login screen

  
Figure 2: App home

  
Figure 3: cooler appliance added

  
Figure 4: how to Add/edit/ delete appliance

  
Figure 5: home server schedule details

  
Figure 6: Optimized scheduling details sent from the utility server

**Conclusion**:

In this project we implemented a scheduling solution for a utility. A mobile application was used by the consumers to input detailed information regarding the appliances they are intending to use which updates a home server. The home server sends scheduling requests to the utility to give the consumer an optimized schedule taking to consideration a global view of all the consumers on the utility network.

This project was a great encompassing implementation to all the skills we learned in the class using a real world problem that utility companies face in our day and age.