



**CMPE 281-01: Cloud Technologies**

**Project Report  
Real Time Weather Sensing**

**Submitted To**  
Dr. Jerry Gao

**Submitted By**  
**Group 18**

Team Member	Student ID
Harshada Ravindra Mone	010717200
Sirichandana Gaddampally	010316865
Harleen Kaur Chawla	010744773
Anuja Dinesh Vaidya	009982674

## **Table of Contents**

- 1. Introduction
- 2. Cloud Technology: AWS
  - 2.1 System Infrastructure
  - 2.2 AWS Design View
  - 2.3 Components
  - 2.4 Why AWS?
- 3. Component Overview
  - 3.1 Functional Components
    - 3.1.1 Weather Monitoring component
    - 3.1.2 Billing component
    - 3.1.3 Sensor control component
  - 3.2 RESTful Web Service Interface
- 4. Technology Selection and Usage
- 5. Conclusions
- References

## 1. INTRODUCTION

Sensors which are subjected to mobile phones can enable appealing sensing applications in different fields such as monitoring of environment, social networking, healthcare, etc. Many essential areas, such as healthcare, military, critical infrastructure monitoring, environment monitoring, and manufacturing have and are still using Wireless sensor network (WSN). These sensors can allow pleasing sensing applications in transportation, safety, etc. The basic structure of Mobile Sensor Cloud Computing (MSCC) is, extending Sensor cloud-computing system to the world of future sensor enabled mobile applications and Internet clouds. It also introduces new technologies, hardware, software, communication protocols, etc., which, when combined, form an ecosystem of mobile cloud. Smart world environment that is plentiful and concealed, related with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network.

## 2. CLOUD TECHNOLOGY: AWS

*“Online services for other web sites or client-side applications is provided by Amazon Web Services. Most of these services aren’t exposed directly to end users, but instead offer functionality that other developers which can be used in their applications. Amazon Web Services’ offerings are accessed over HTTP, using the REST architectural style and SOAP protocol. Billing of all services are based on usage, but how the usage is measured for billing varies from service to service.”*

”



**Amazon Web Services (AWS)**, is a set of remote computing services, also called web services, that make up a cloud-computing platform offered by Amazon.com.

**Amazon Elastic Compute Cloud (EC2)** makes a central part of Amazon.com's cloud-computing platform, Amazon Web Services (AWS), which allows users to rent virtual on which to run their own computer applications. EC2 promotes scalable deployment of applications by providing a web service through which a user can boot an Amazon Machine Image to configure a virtual machine, which Amazon calls an "instance", which consists of any software desired. A user will be able to create, launch, and terminate server-instances as needed, paying by the hour

for active servers - hence the term "elastic". EC2 gives users a control over the geographical location of instances that allows latency optimization and high levels of redundancy.

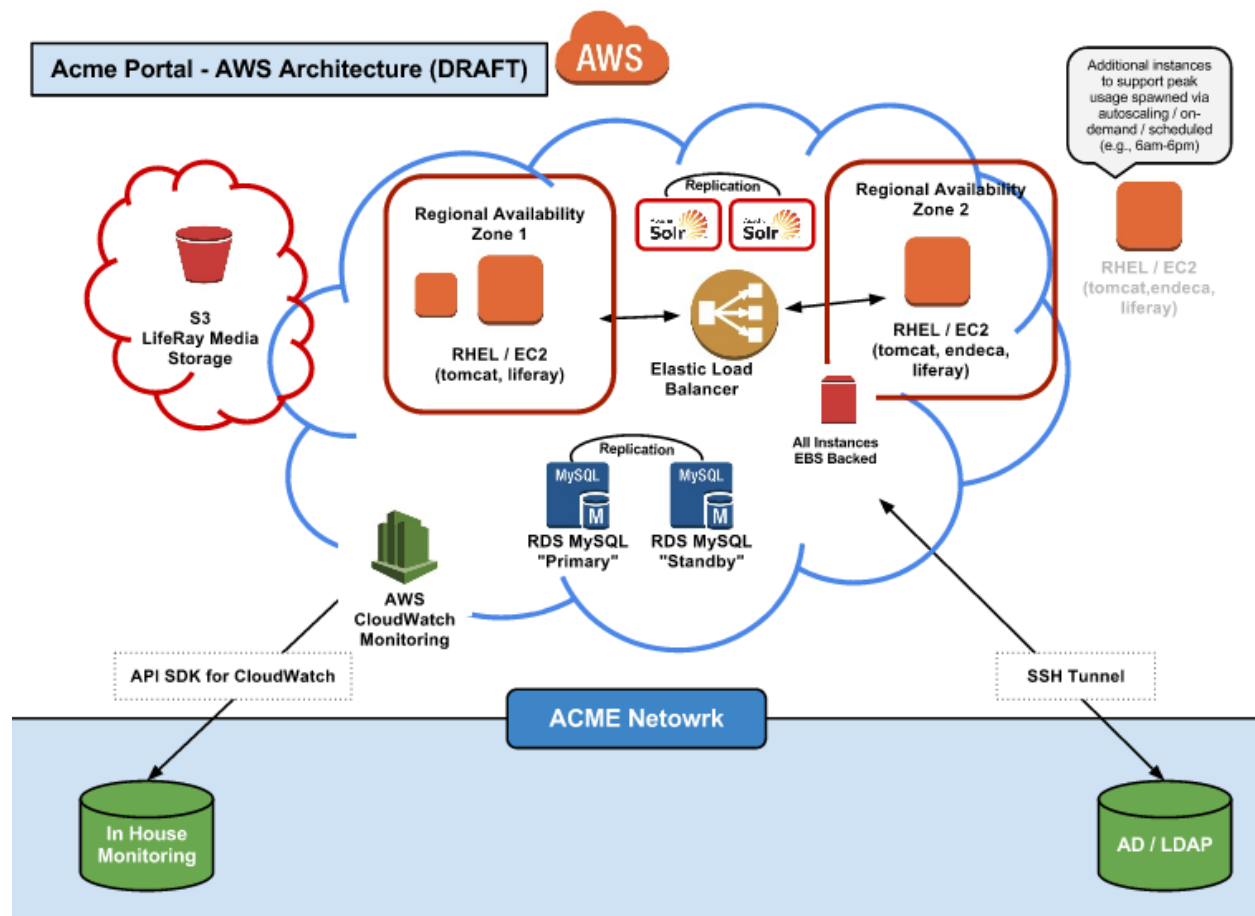
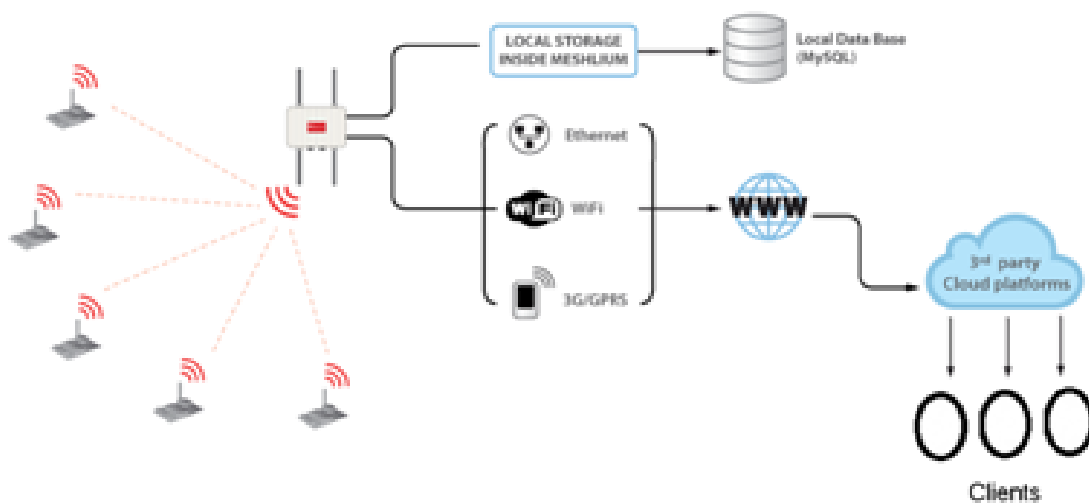


Figure 1. Amazon Web Services Cloud Infrastructure

## 2.1 System Infrastructure

There are multiple components (or projects) in an AWS platform. Every component in AWS infrastructure is managed by a technical committee and the components which should be included in the AWS core is decided by the AWS Foundation. These components work together to provide the required services to deliver the cloud.

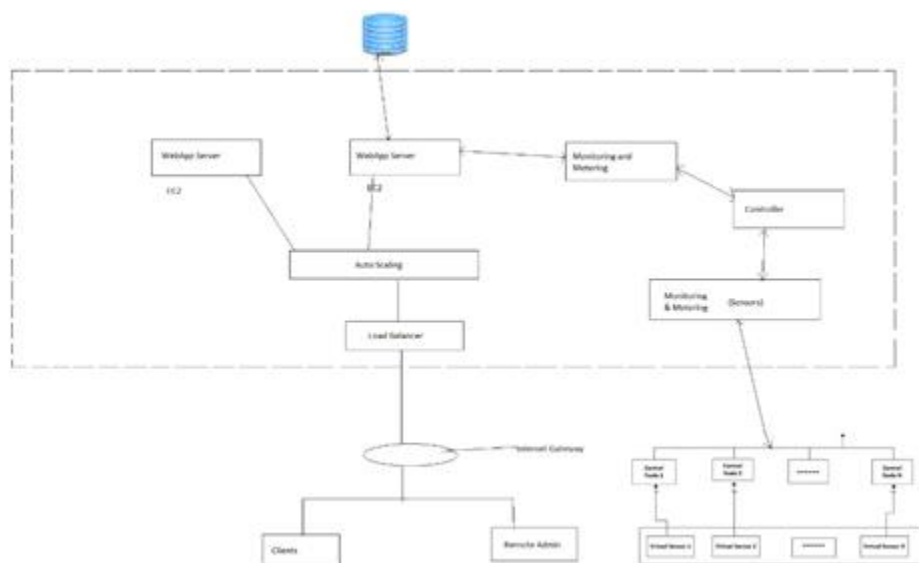


**Figure 2. AWS Infrastructure**

## 2.2 AWS Design View:

The figure below shows the infrastructure of AWS along with its components or projects as they are known in AWS. These include Controller, Various Servers, Load Balancing and Auto Scaling. Additional projects are the Metering and Monitoring Service (Ceilometer) that handles the billing, benchmarking, scalability and statistics.

1. Control Node: Controllers act as unified security platform for which will authorize users, agents, sensor owners, etc. Data regarding credentials and also plan used by user is stored in controller is stored in Databases. Depending on the usage of the user and data shared on the cloud, bill will be calculated accordingly using cost-model and billing metric.
2. Network node and Hub management: Varied physical sensors are continuously fetching data from varied location. Network node will help to connect them all in one single network. Collecting data from sensors and routing them to varied data sources to Amazon DB. Each sensor will route data to its dedicated data source.
3. Mobile sensor node (with mobile agent): Mobile Sensor fixed at various places will send the data and this data will be stored in the local Databases. User can ask for this data based on what kind of data he wants.
4. Service node (supporting service functions, data): Portal for users to choose the services they require. Users can ask for various type of data depending what kind of parameters he wants to be displayed. This will change based on the users and also there can be multiple users.
5. Mobile user client nodes: Web browser from where user will be able to login to interact with cloud, can make requests, can get sensor data, can view their usage and bill.



**Figure 3. AWS Design View**

## 2.3. Components

- Sensors:

The data collection will be done by sensors from the location that is placed by the user and will forward the data to the cloud using any medium (ex: Wi-Fi network) that is connected to the internet.

- Wi-Fi Router/Ethernet/3G:

A Wi-Fi router or any other device connects the sensors to the internet and helps them give data to the cloud.

- Server:

This server receives all the requests to upload the data of the sensors to the cloud.

- Data Store:

This stores the data from the sensors along with other data details.

- Service Request Processor:

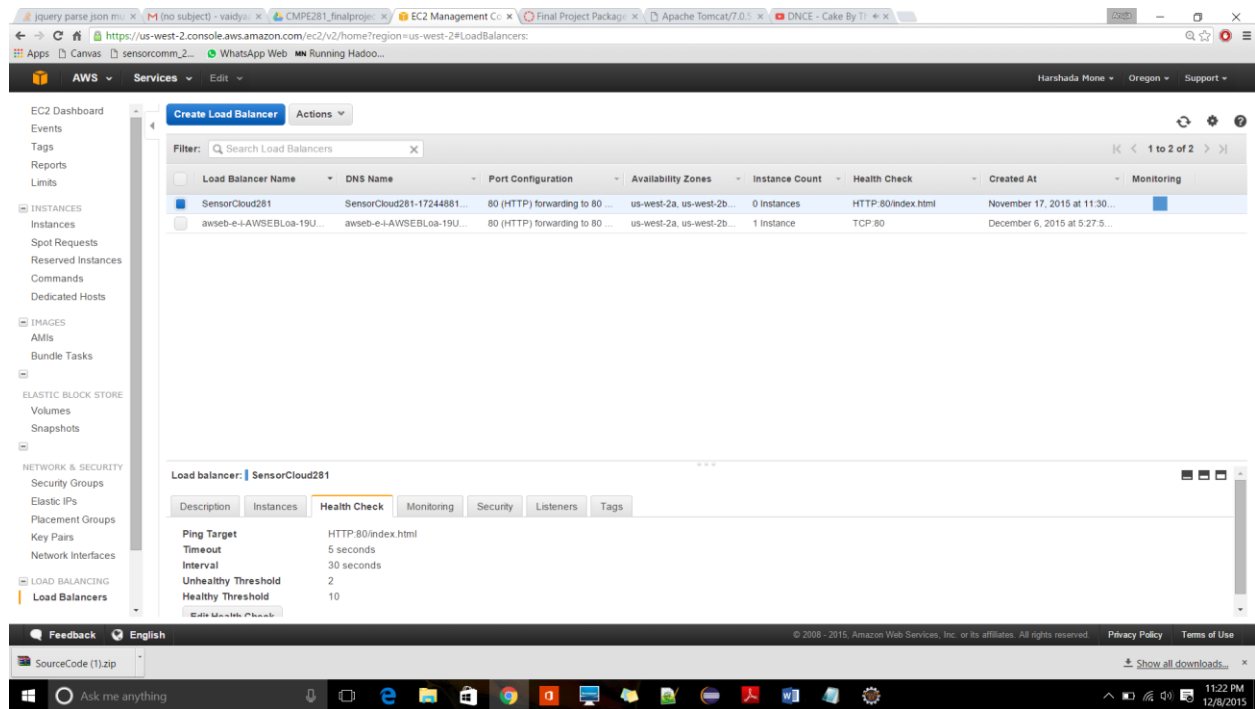
This processor does the task of handling requests from client or request broker.

- Request broker:

The request broker enables effective communication between the client and the server.

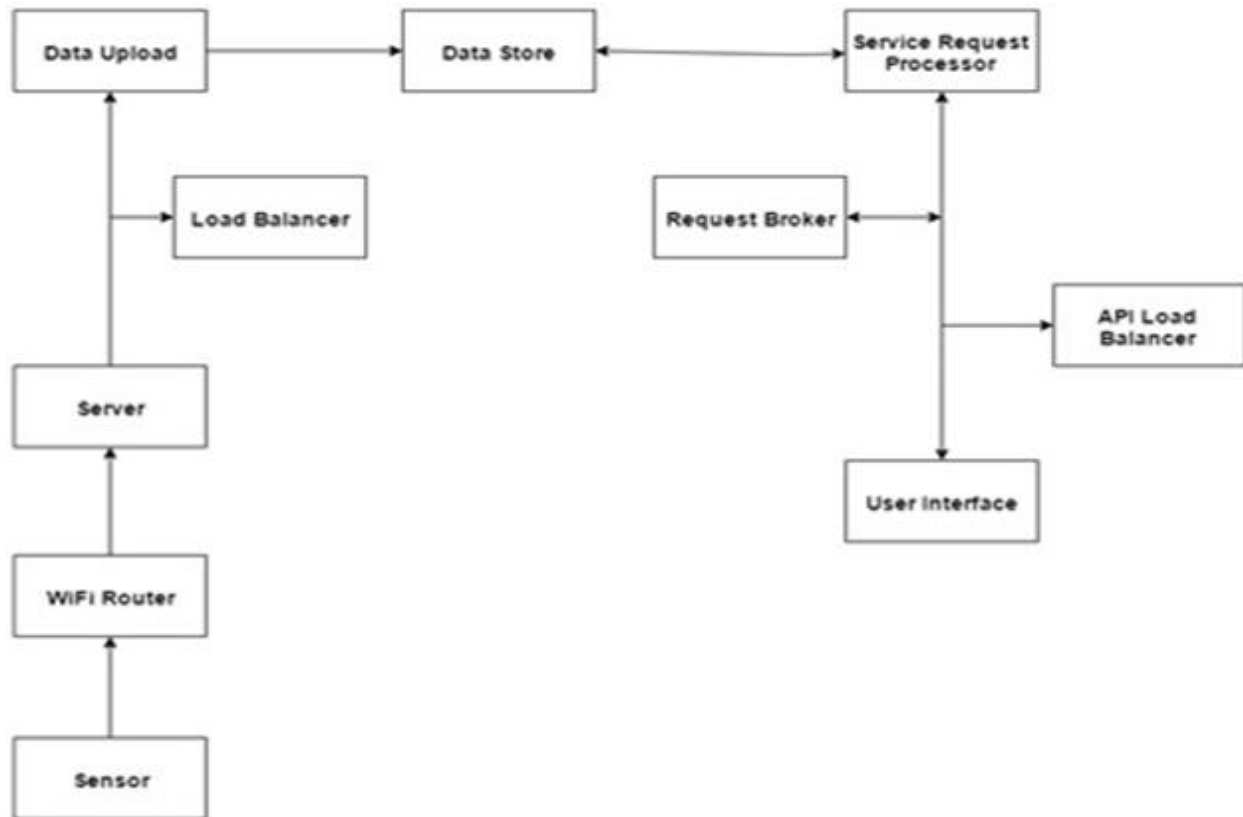
- API Load Balancer:

It manages large number of incoming API requests. In that case, it will forward requests to another API request handler which is less loaded.



- User Interface:  
End users or clients are able to access the cloud and its services through this platform.





**Figure 4. Components**

## 2.4. Why AWS?

This provides multiple benefits to users and vendors:

1. Simple Storage Service – This covers storing of application programs and data and online backup. The service is of low cost, high speed and scalable.
2. CloudDrive – This lets users utilize web connected devices to access and upload photos, music, documents and videos. They can also use their devices to stream music.
3. RedShift – The service is designed for analytic workloads which will connect with business intelligence tools and standard SQL based clients. It is a data warehouse service that handles petabyte-scale data.
4. CloudSearch – It is used for the integration of a customized search capabilities and is a scalable search service.
5. Dynamo Database – is a NoSQL database which is fully managed which is known for its scalability and low latencies.
6. Elastic Compute Cloud – can be used as an unlimited number of virtual machines and will let business subscribers run app programs.

7. **Cost-Effectiveness** There is no long-term commitments or upfront expenses as AWS provides low and 'pay as you go' pricing. Amazon passes the benefits of cost saving on to customers in the form of lower prices as it manages and builds global infrastructure at scale
8. **Flexibility and Openness** AWS is a platform agnostic to operating systems and languages. You select the programming model or development platform that can be the most suitable for your business.

### **3. Component Overview:**

#### **3.1 Functional Components:**

##### **1. Mongo DB**

Mongo DB is used at the sensor node controller to store the sensor information locally. Mongo DB is a noSQL database which stores both structured/unstructured and polymorphic data. It is highly scalable and processes large amount of data quickly. It is highly available and has features like load balancing, automatic failover, and security.

##### **2. Amazon EC2**

Amazon EC2 provides scalable computing without having to invest in expensive hardware. Hence it handles the spikes and up and downs in the traffic by scaling up and down automatically. It provides instances which are the virtual computing environments. These instances can be launched, terminated or created as per the need of the users. The payment model for amazon Ec2 is pay as you go.

#### **Services**

##### **1. Weather Monitoring component**

The weather monitoring component is responsible for receiving weather information from sensors to sending out weather updates to the clients through the web.

##### **2. Billing component**

Based on consumption of the web services the customer bills are produced by the billing component.

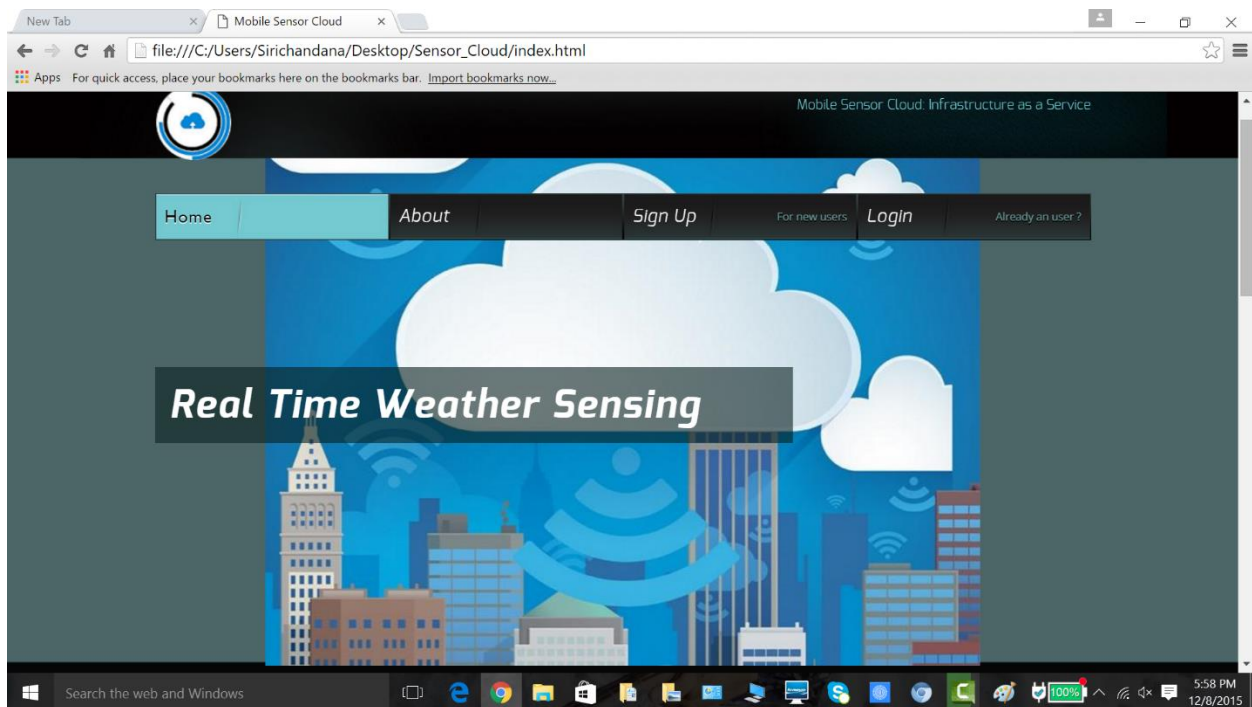
### 3. Sensor control component

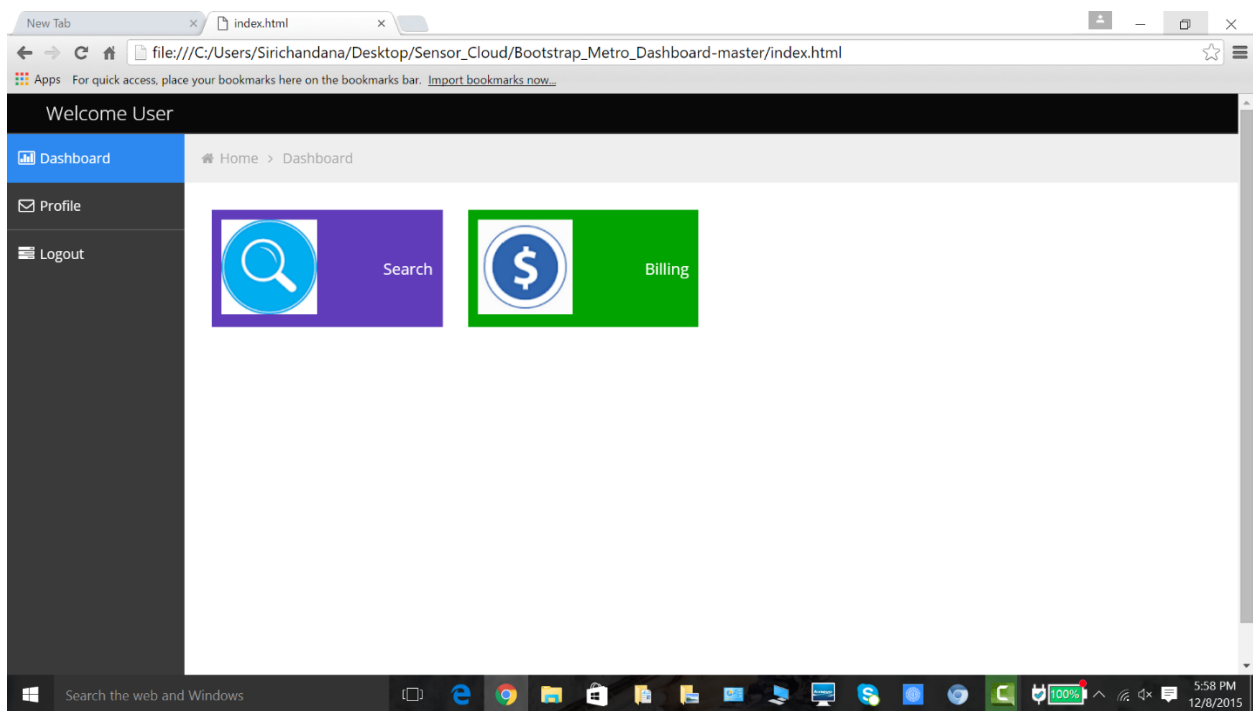
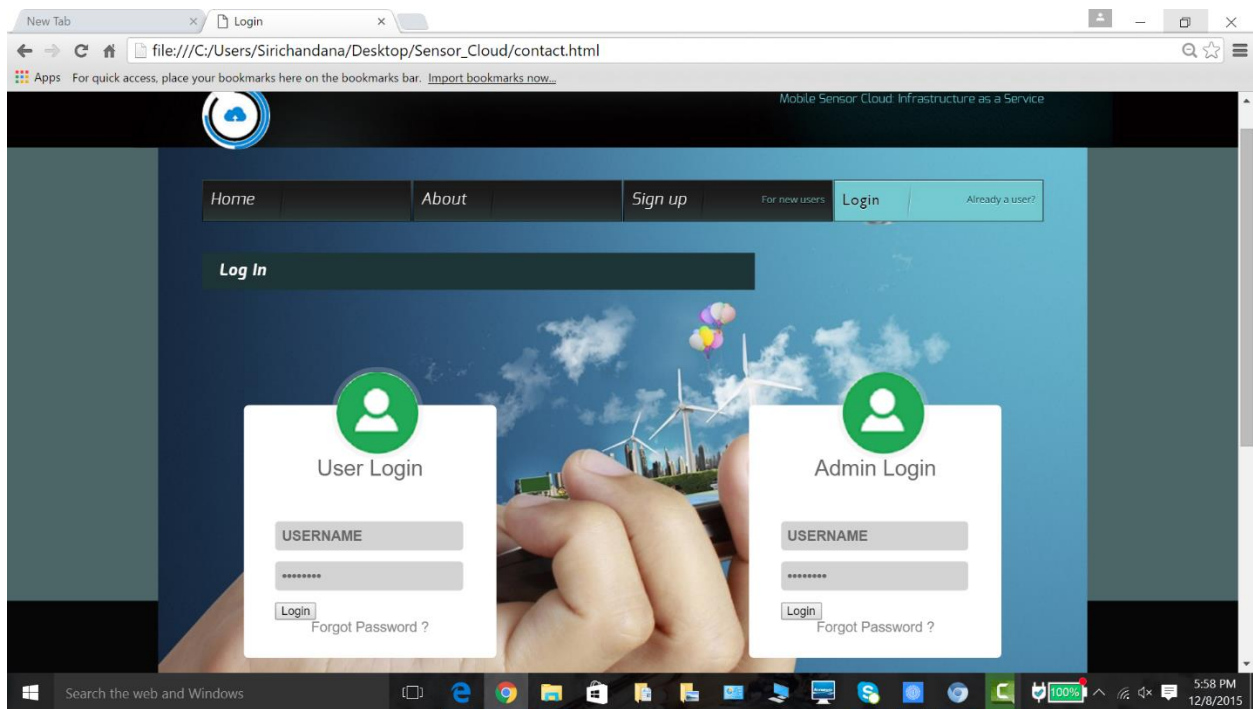
The sensor control component is responsible for controlling the activity of the sensors. The sensors can be relocated, added, removed, and switched on and off using the sensor control component.

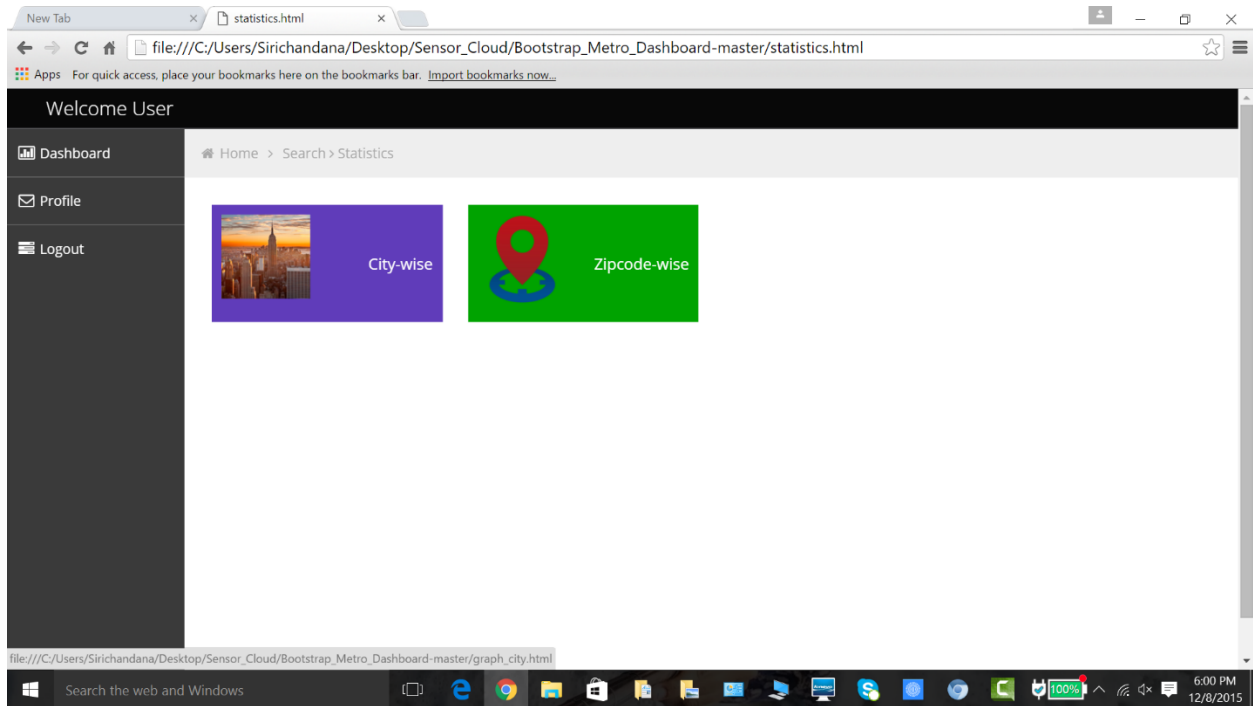
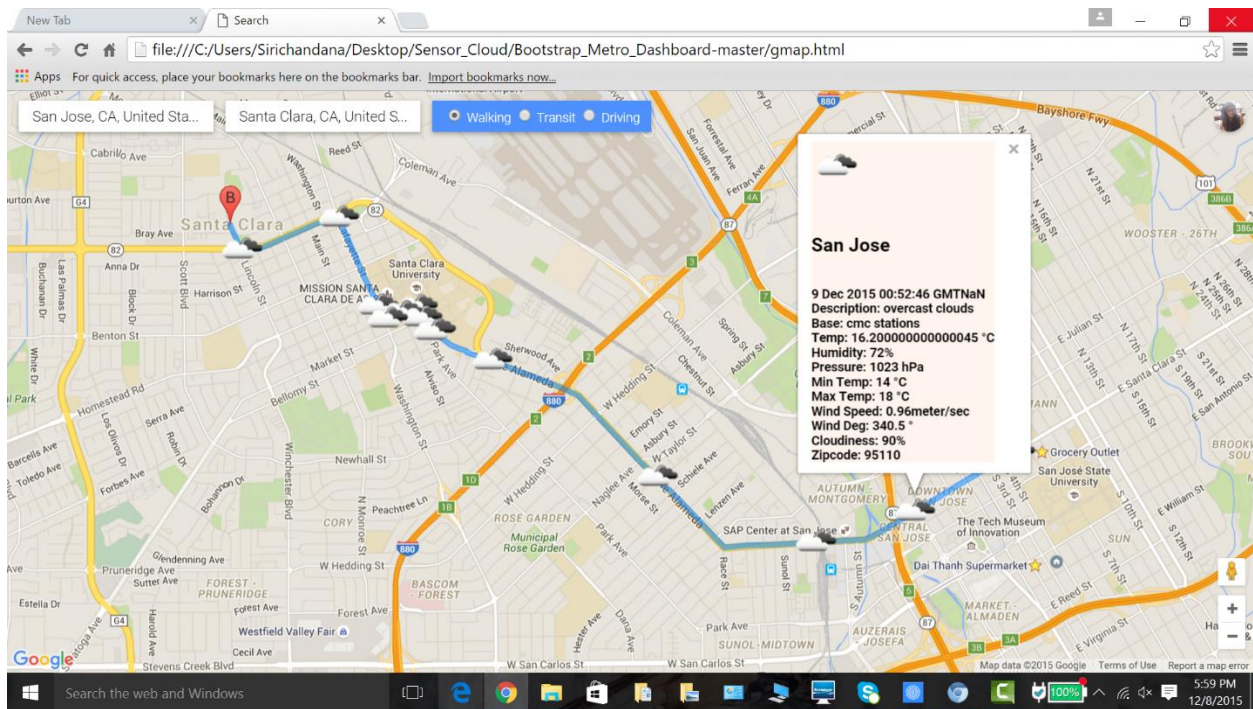
## 3.2 RESTful Web Services Interface

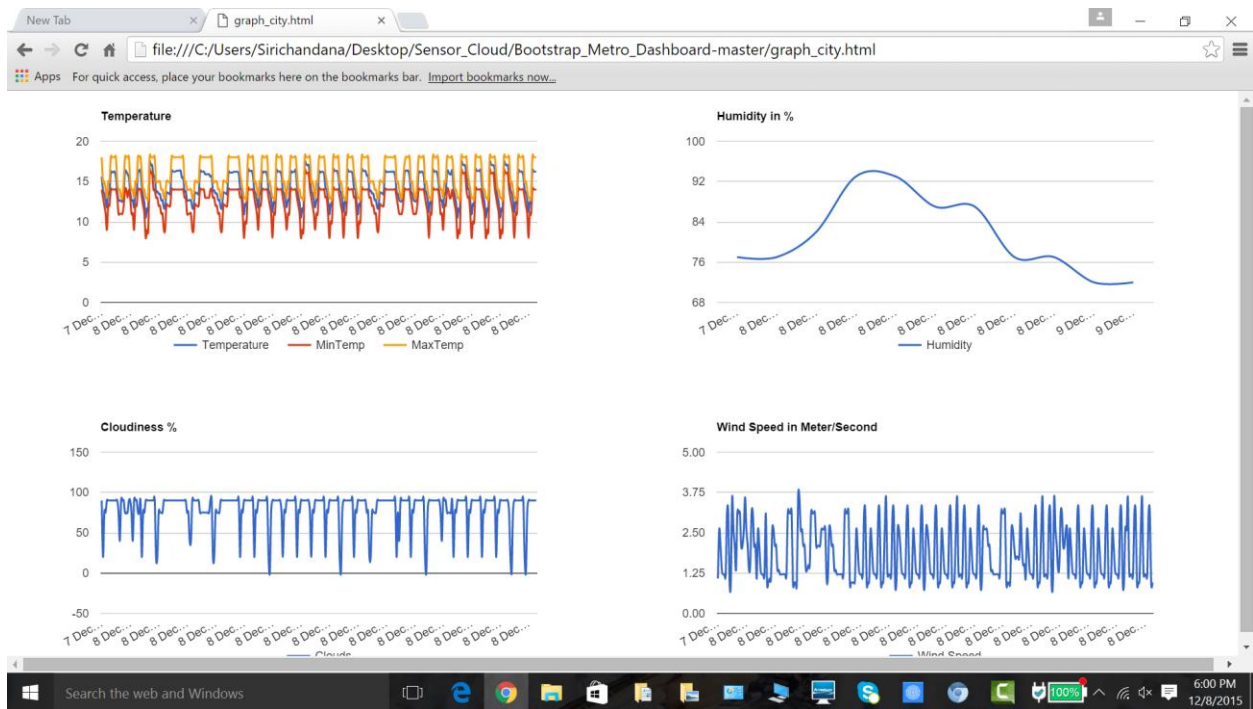
### 1. Control Node for Sensor

The sensors are bootstrapped, registered through the GUI of the hub to control the sensors. It receives the sensor status information and the weather information recorded by the sensors and passes it on the local database and server.









New Tab x User Profile x


file:///C:/Users/Sirichandana/Desktop/Sensor\_Cloud/Bootstrap\_Metro\_Dashboard-master/messages.html

Welcome User

Dashboard Profile Logout

Home > Profile

ABC's Profile



User Details	
Name	ABC XYZ
FirstName	ABC
Last Name	XYZ
Username	ABC
Registered Email	abc@gmail.com

Windows Taskbar: Search the web and Windows. 6:01 PM 12/8/2015

January

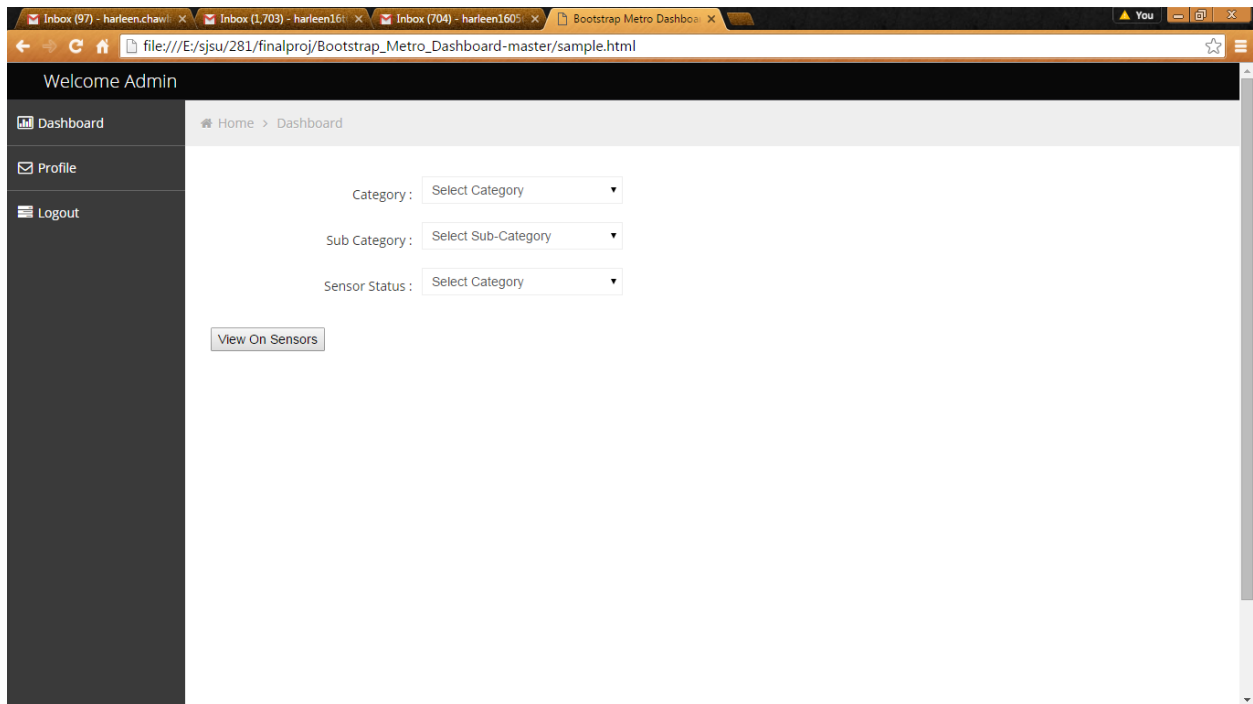
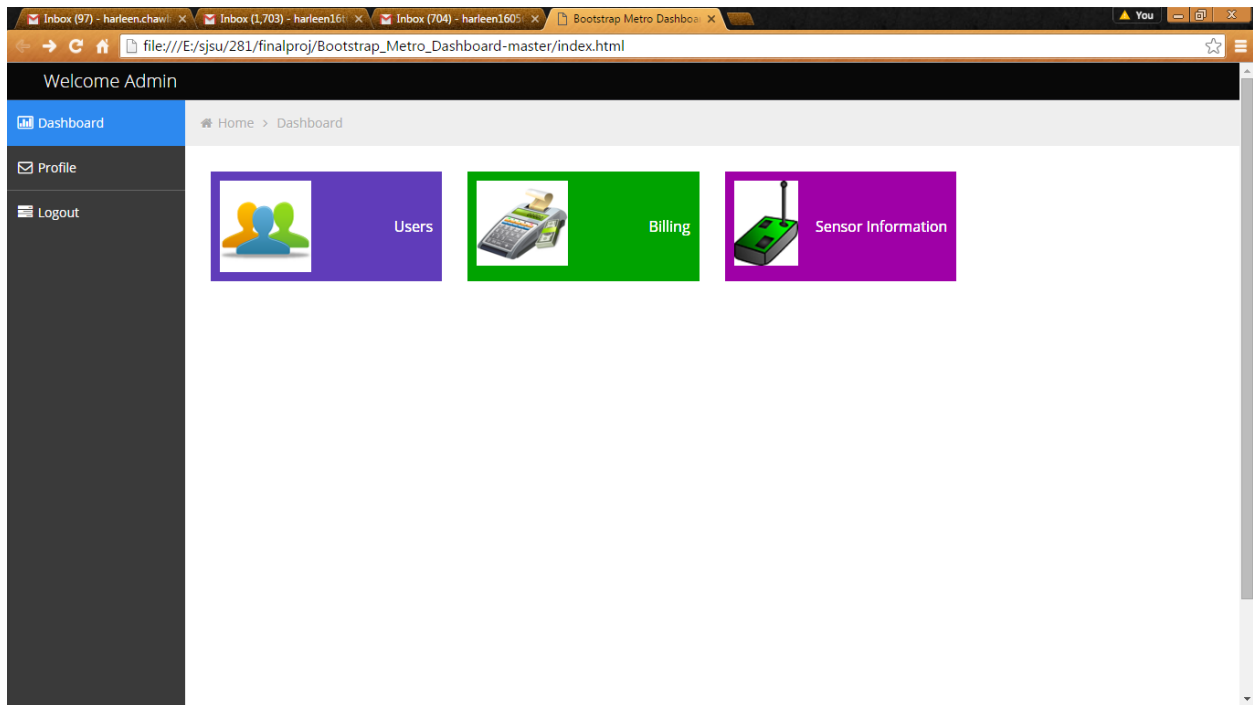
User	Net Usage(in units)	Price unit(in dollars)	Monthly Rental
User : ABC			
Usage Plan: Plan ABC	20	1	20
Usage Allowed: 100 call requests			

**Current Bill**

Billing	
Name	ABC XYZ
Current Month	December

## 2. Administrator

The GUI for the administrator has a login form for authentication of the remote admins. The functions of the administrator like controlling of the sensors, managing the database, billing of the customers, and controlling the access to the data.





#### 4. Technologies Selection and Usage:



Figure 5. Technologies Used

<b>Components</b>	<b>Tools</b>	
<b>Database Server</b>	<b>Mongo DB</b>	<b>Mongo DB- Sensor Information at client side Dynamo DB- User,</b>
		<b>Sensor, Admin, Billing Information</b>
<b>Web Server</b>	<b>Apache Tomcat</b>	<b>Application Server</b>
	<b>HTML,CSS, JavaScript</b>	<b>GUI</b>
<b>Request generator</b>	<b>JSON</b>	<b>RESTful API</b>
<b>Language</b>	<b>JAVA</b>	
<b>Emulator</b>	<b>Eclipse IDE</b>	
<b>Deployment</b>	<b>AWS</b>	<b>IaaS</b>

## **5. Conclusions**

The objective of this project was to implement Real Time Weather Sensing for a given city which will display the given parameters from source to destination and will be helpful to any particular driver in understanding an optimal route to take because of the parameters.

After analyzing three different load balancing algorithms which are Ant algorithm, Particle Swarm algorithm and Round robin algorithm, our system implemented Round Robin algorithm due to the fact that it is very effective for static environment as compared to the other algorithms.

## References

1. <http://www.ijcsit.com/docs/Volume%206/vol6issue01/ijcsit2015060135.pdf>
2. [https://aws.amazon.com/products/?nc2=h\\_ql\\_ny\\_livestream\\_blu](https://aws.amazon.com/products/?nc2=h_ql_ny_livestream_blu)
3. <https://aws.amazon.com/ec2/>
4. [https://en.wikipedia.org/wiki/Network\\_as\\_a\\_service](https://en.wikipedia.org/wiki/Network_as_a_service)
5. [http://www.service-architecture.com/articles/cloudcomputing/network\\_as\\_a\\_service\\_naas.html](http://www.service-architecture.com/articles/cloudcomputing/network_as_a_service_naas.html)
6. <http://www.definethecloud.net/>