

CMPE 281 – Team 18

Project Design Document

Date: November 03rd, 2015

Cloud Base Weather Sensing Team Members:

Anuja Dinesh Vaidya(009982674)

Harshada Ravindra Mone(010717200) Harleen Kaur Chawla(010744773)

Sirichandana Gaddampally(010316865)

Table of Contents

1. INTRODUCTION .....................................................................................................................3
2. INFRASTRUCTURE DESIGN..................................................................................................3
   1. SYSTEM ARCHITECTURE ....................................................................................4
   2. COMPONENTS .........................................................................................................4
3. AWS DESIGN VIEW ...............................................................................................................5
   1. CONTROL NODE ............................................................................................5
   2. NETWORK NODE AND HUB MANAGEMENT ......................................6
   3. MOBILE SENSOR NODE (WITH MOBILE AGENT)…........................6
   4. SERVICE NODE (SUPPORTING SERVICE FUNCTIONS, DATA) ......6
   5. MOBILE USER CLIENT NODES .................................................................6
4. SOLUTION DESIGN................................................................................................................7
   1. MOBILITY NETWORK SERVICE..............................................................7
   2. MOBILITY CONTROL SERVICE..............................................................8
   3. MOBILITY DATA SERVICE .......................................................................10
5. COMPONENT AND SERVICE DESIGN.................................................................................11
   1. COMPONENT FUNCTIONS WITH INPUTS/OUTPUTS, ALGORITHM AND

DATA FORMATS.........................................................................................................11

* 1. COMPONENT API DESIGN FOR FUNCTIONAL COMPONENTS

(AMAZON

AWS)...............................................................................................................................11

Database .......................................................................................................................................11

Storage and Content Delivery Network.............................................................12

Analytics......................................................................................................................................12

Compute & Networking ..............................................................................................12

Deployment & Management............................................................................................13 App Services ..........................................................................................................................13

* 1. MOBILE SERVICE APIS (RESTFUL).....................................................................14
  2. GUI DESIGN.......................................................................................................................14
  3. TECHNOLOGY SELECTION AND USAGE ..............................................................17

1. BIBLIOGRAPHY.....................................................................................................................17

1 Introduction

Sensors on (or which is attached to) mobile phones will enable attractive sensing applications in different domains such as environmental monitoring, social networking, healthcare, etc. wireless sensor network (WSN) applications have been used in several important areas, such as healthcare, military, critical infrastructure monitoring, environment monitoring, and manufacturing. These sensors can enable attractive sensing applications in various domains such as environmental monitoring, social networking, healthcare, transportation, safety, etc.

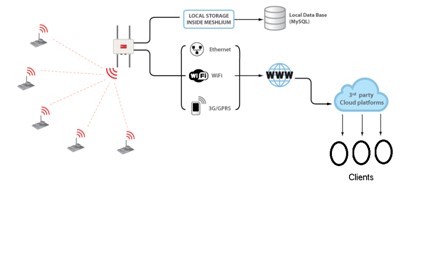
The concept of Mobile Sensor Cloud Computing (MSCC) is to extend Sensor cloud-computing ecosystem to the world of future sensor enabled mobile applications and Internet clouds. Also it introduces new technologies, hardware, software, communication protocols, etc., which together forms ecosystem of mobile cloud. Smart world environment that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network.

# 2. Infrastructure Design

## 2.1. System Architecture

The cloud infrastructure is shown in the figure below. The various components and the connections can be seen clearly. The process includes:

* The data is collected by sensors, from the environment.
* The data from the sensors is given to the Controller hub.
* This data then gets stored in a local storage (database) which keeps a track of the all the sensors present in the environment.
* The hub via internet, does the task of transferring data to the respective cloud storage.
* The cloud can be accessed by ‘n’ number of clients, who send requests, for data fetching.



2.2. **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: WiFi 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 :

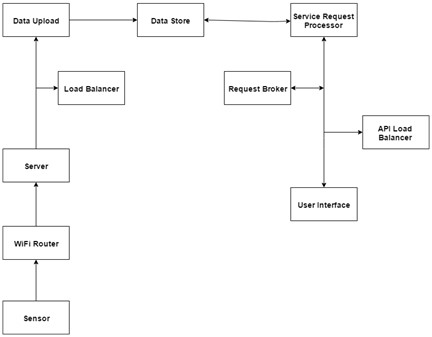
The request broker enables effective communication between the client and the server.

* 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.

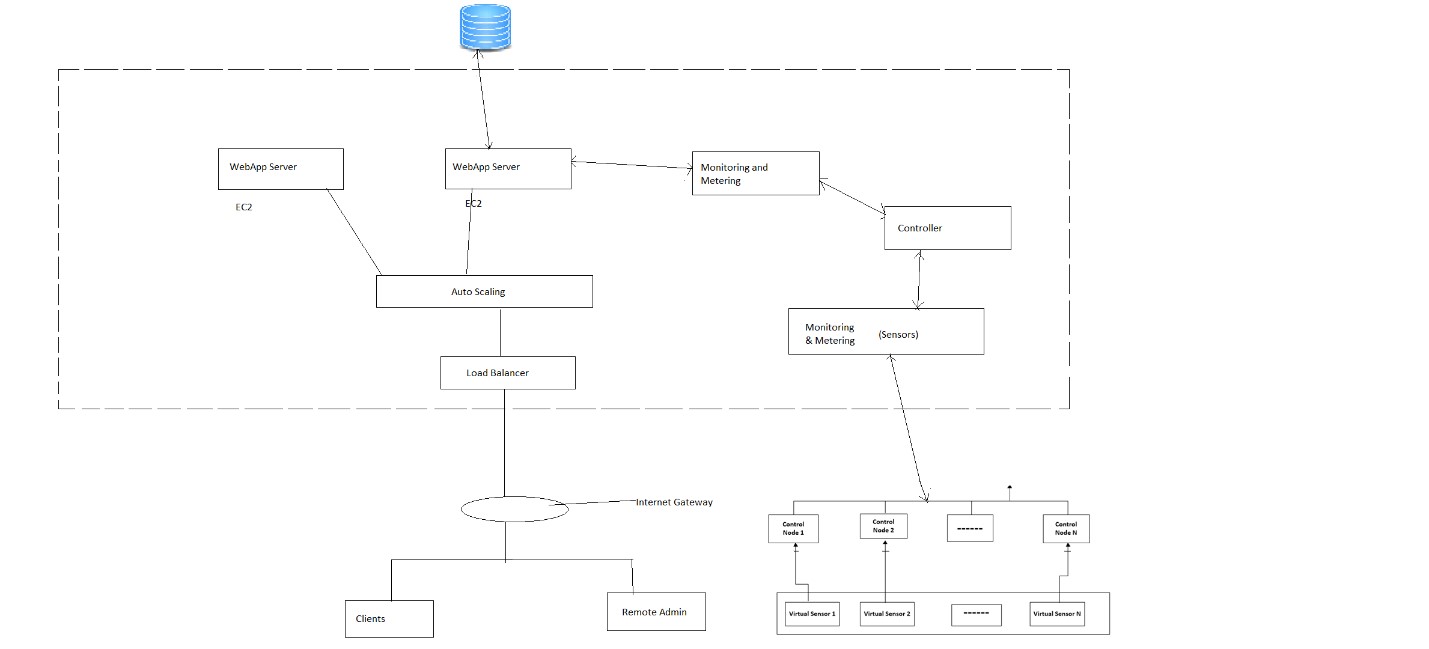
* User Interface :

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



1. **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.

* 1. 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.

* 1. 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.

* 1. 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.

* 1. 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.

# 4. Solution Design

## Mobility Network Service

**Network as a service** (**NaaS**) describes services for network transport connectivity. It takes in consideration all the factors that affect the efficiency of the network. It can be separately considered as a cloud provider like Infrastructure as a service or platform as a service. It looks after the optimal use of network resources based on the network needs. The network may include virtual or physical sensors. The owners of the network maybe the network managers or 3rd party vendors. Network virtualization is better than have a large number of physical sensors for optimized use of sensors.

NaaS can include different models and protocols like flexible and extended Virtual Private Network (VPN), Bandwidth on demand, custom routing, multicast protocols, security firewall, intrustions detection and prevention, Wide Area Network (WAN), content monitoring and filtering, and antivirus. There is no standard specification as to what is included in NaaS. It’s totally up to the user as to how many features are to be provided.

The end user just has to be concerned about having an access to a NaaS portal and has nothing to do with the physical hardware required for the network. The end users, admin and the vendors will have interactions on the dashboard in accordance with their access rights. The network maintains all the components involved in the network.Some of the concerns related to NaaS could be the scale of availability, vendor related issues, Service level Agreements (SLAs).

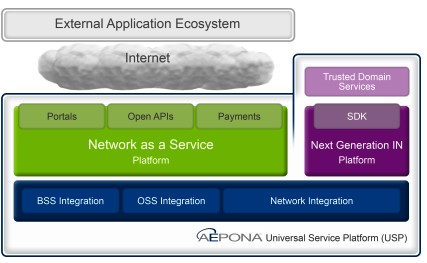


Fig 1: AEPONA Network as a Service

## Mobility Control Service

Control-as-a-service provides user a control server and control nodes which collect and manage sensors and their data locally and send it to a centralized control server. The control nodes act like the middleware between sensor nodes and control network. The user is provided with a selfserving portal in order to access the control nodes and control server. The control server has the ability to enable and disable the sensor nodes and the data flow from them.

The control server has to look after the load balancing of the network so that the network doesn’t crash. Load is the number of requests the server is receiving. The control server also has to schedule what request is served based on priority.

Control-as-a-service is favorable for a couple of reasons. First, its simple to work with as the control flow is pretty clear. Second since the control server control all other components, so security issue is better than other scenarios.

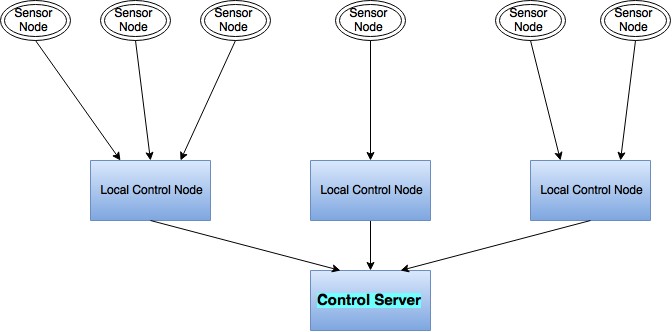


Fig 2. Control as a service

## Mobility Data Service

Data-as-a-Service is similar to Saas as in DaaS management makes handling data easier and more efficient. It’s basically on-demand data service. It provides required data when requested independent of the physical locations of the requester and provider. It will perform various functions like reducing data redundancy, compression and expansion of data and its efficient retrieval. The user can use the console to list down the functionalities that needs to be associated with the service.

The various functionalities include

1. Maintaining the sensor data to monitor its status
2. To keep track of all sensor nodes
3. Enabling and disabling the data access based on need

# 5. Component and Service Design

5.1 Mobile Service Component Analysis and Design

· Functional Components

1. Amazon Dynamo DB

Dynamo DB is a noSQL database from Amazon which offers fast and consistent performance. It is a fully managed database service. This keeps you away from the tedious jobs such as setups, hardware and software provisioning, distribution, configuration. It empowers with stronger users access management features to control access to resources and services. Dynamo DB supports not only key value data structures but also documents. Hence it achieves flexibility. Dynamo DB is highly scalable.

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.

1. 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.

1. Billing component

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

1. Sensor control component

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

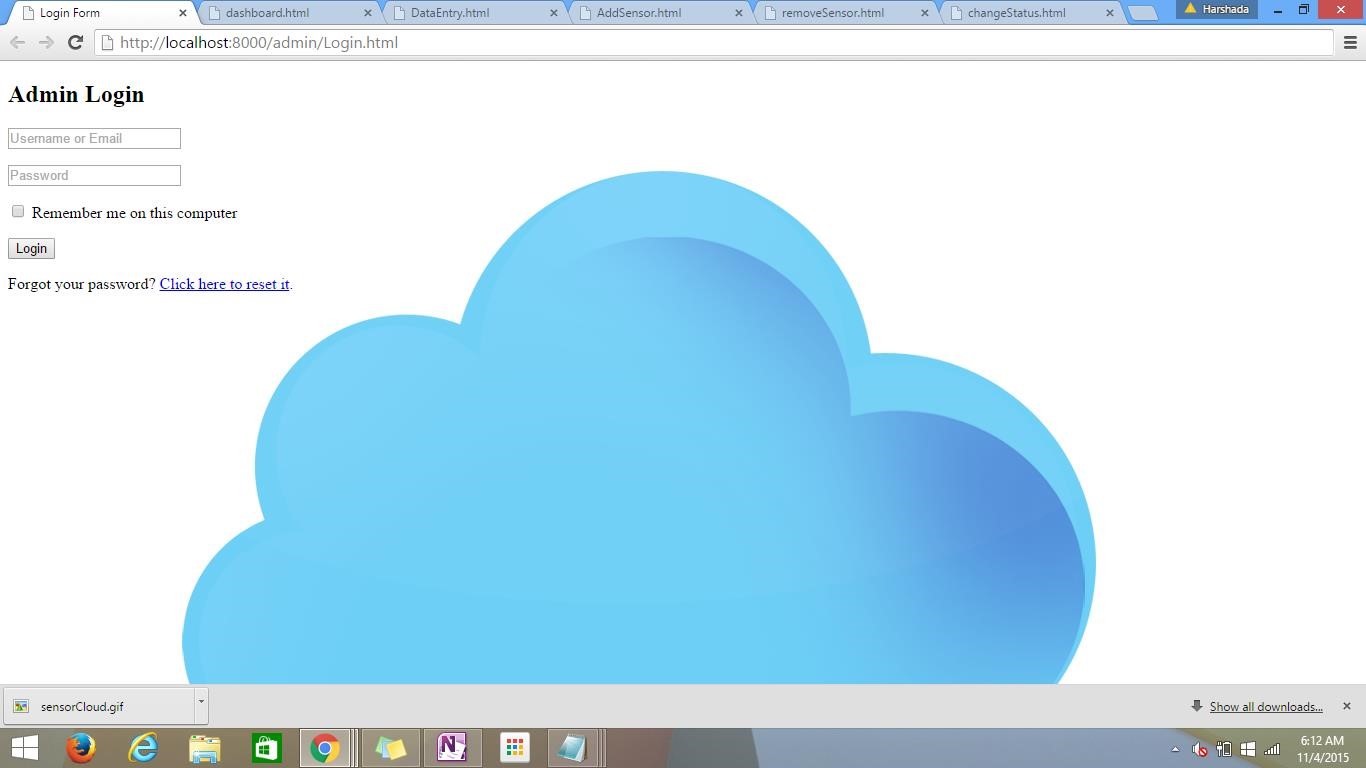
5.3 GUI design and RESTful web services.

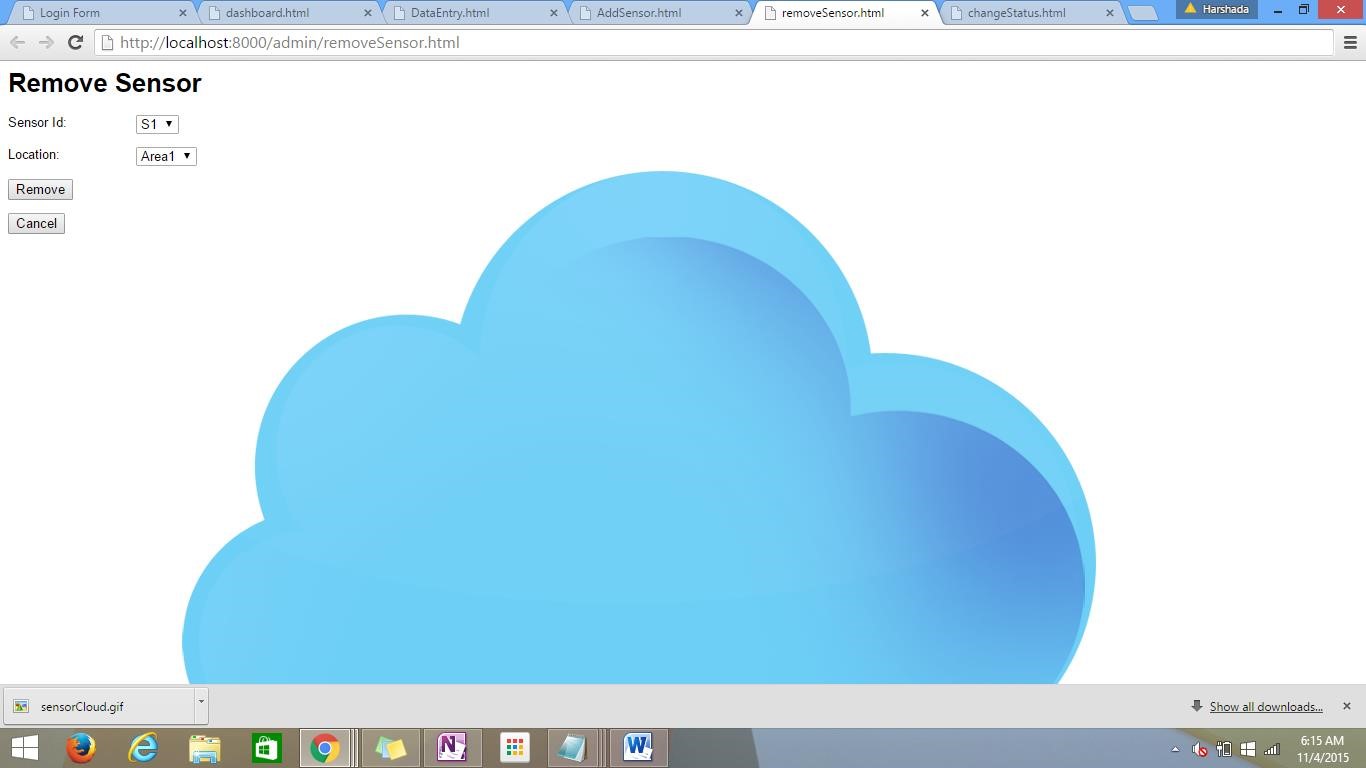
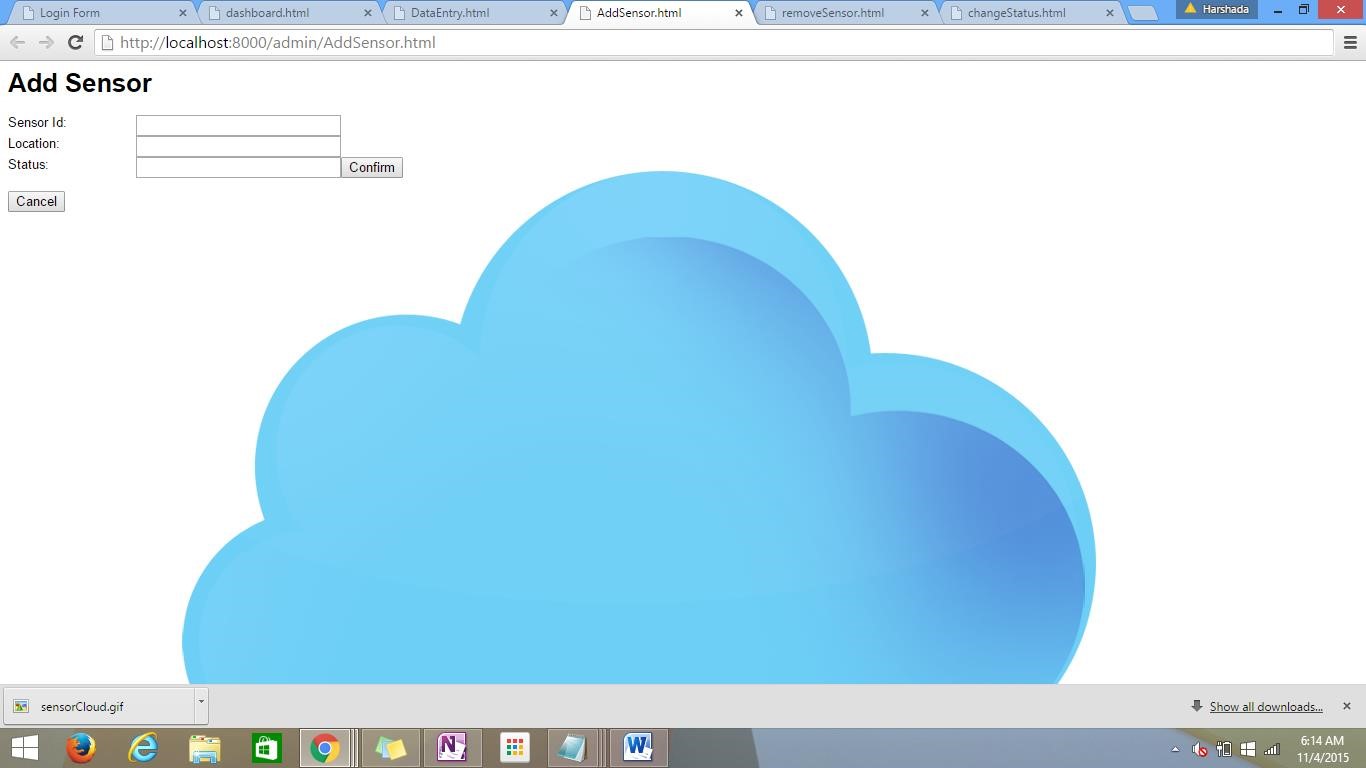
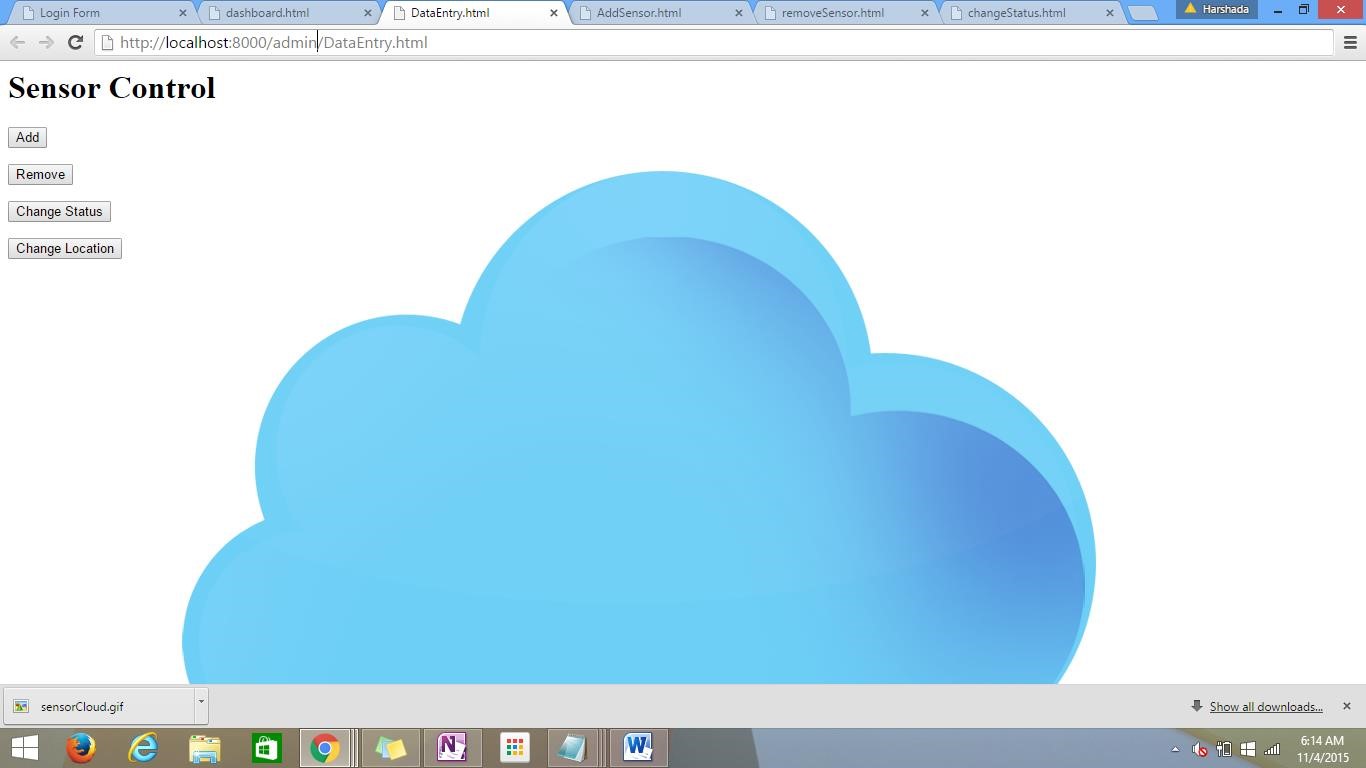
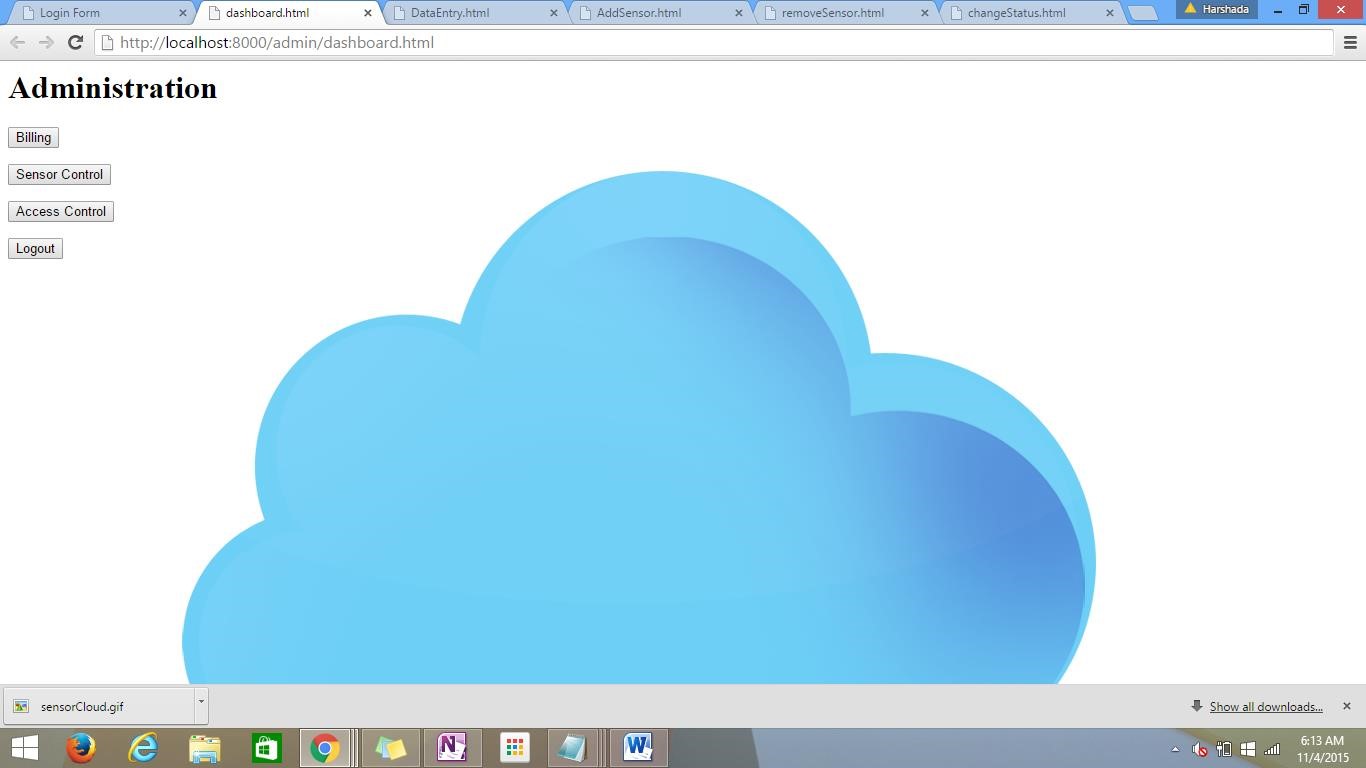
1. Control Node for Sensor

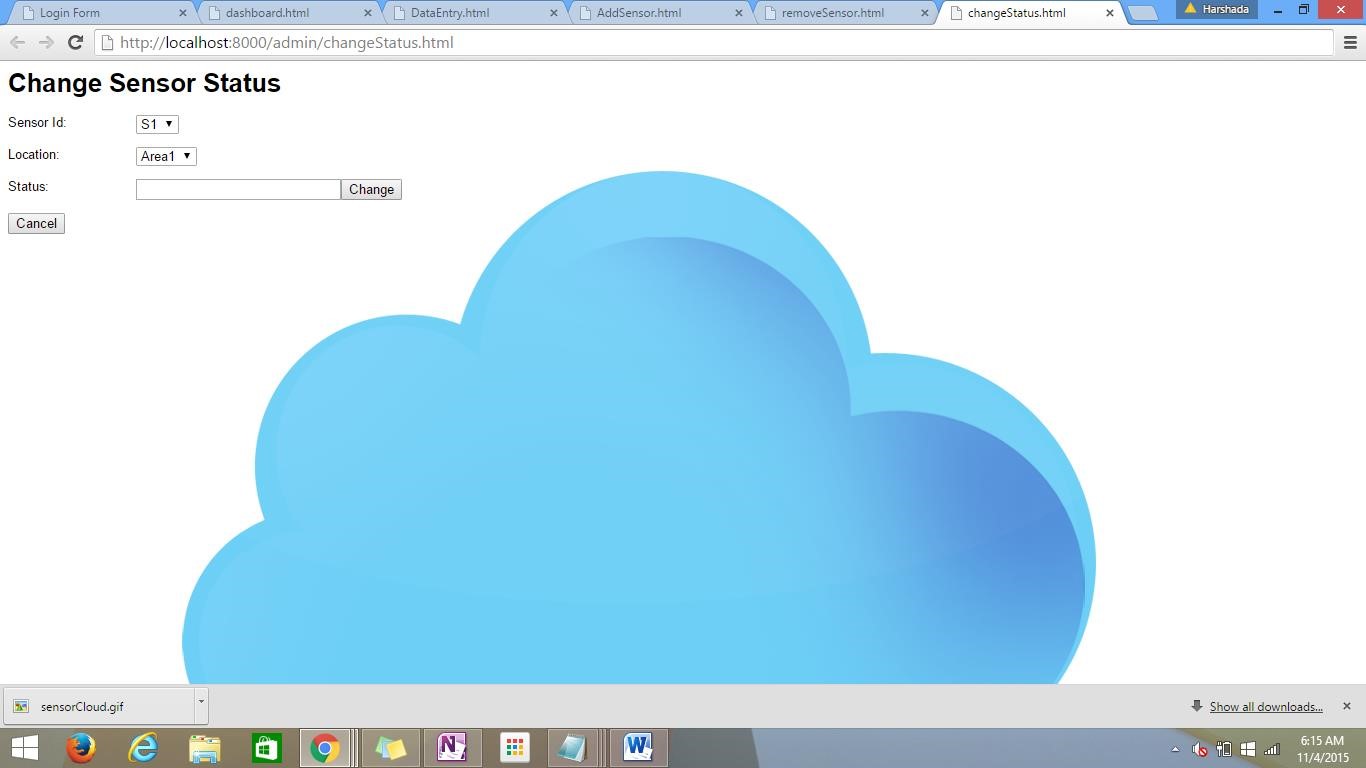
The sensors are bootstrapped, registered through the GUI of the hub for controlling the sensors. It receives the sensor status information and the weather information recorded by the sensors and passes it on the local database and server.

1. 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.







1. Customer

Customer GUI allows customers to access the weather information based on their location. It also allows customer to view bills and receipts and manage payments.

5.4 Technology Selection and Usage:

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

Bibliography:

* + 1. <http://www.ijcsit.com/docs/Volume%206/vol6issue01/ijcsit2015060135.pdf>
    2. <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>
    5. [http://www.service-architecture.com/articles/cloudcomputing/network\_as\_a\_service\_naas.html](http://www.service-architecture.com/articles/cloud-computing/network_as_a_service_naas.html)
    6. http://www.definethecloud.net/