**CS590BD Big Data Analytics and Apps**

**Second Increment Report – Group2**

**Motion Detector**

**Kommineni, Siva Krishna**

**Ponnam, Balakrishna**

**Boyanapalli, Swathi**

**Pathuri, Savya Sri**

**Summary:**

For the second iteration the main focus is made on the selection of the open source applications and looking for establishing a connection between the game movements and the sensor tag movement directions. We used certain tools for analyzing the connection establishment and finding out the right algorithm for data classification and analysis. Our back end work majorly consists of analyzing the data and cluster the list of motions based on certain actions and movement in the space. Some amount of work has also been spent on the developing of UI part and investigations were made for developing an open source application according to our specifications and direction movement for the detection of various motions.

**Framework Specifications:**

In this increment we made progress in the selection of the game and analyzing the direction movements present in it so that we can match it with the sensor tag direction movement. First we should collect the data, push it into HBase and host a restful web service using the Glassfish server. We are trying for the connection establishment between the accelerometer data in the application file and the sensor tag accelerometer connections. Currently we are running the application on the emulator for searching the exact movements. The mobile client is entirely developed in Android and it is a native application for android users. The version supports from Froyo to Jellybean’s.

**Application Specification:**

* **Software Specification**
  + Tools: Mahout, R, RHadoop, Android Development Kit, Eclipse Kepler
  + Operating System: Android
  + Development Operating System: Windows 8
  + Programming Language: Java 7.0
  + Databases: HBase, Hadoop

**Activity Recognition Scenario and Data Collection**

* **Devices/Sensors:**

## Sensors: TI Sensor Tag - CC2541DK-SENSOR

**Devices: Android devices**

* Bluetooth 4.0 compatible android device
* Android 4.3 or above OS
* GPS

**TI chronos Watch – EZ430-Chronos**

**Implementation:**

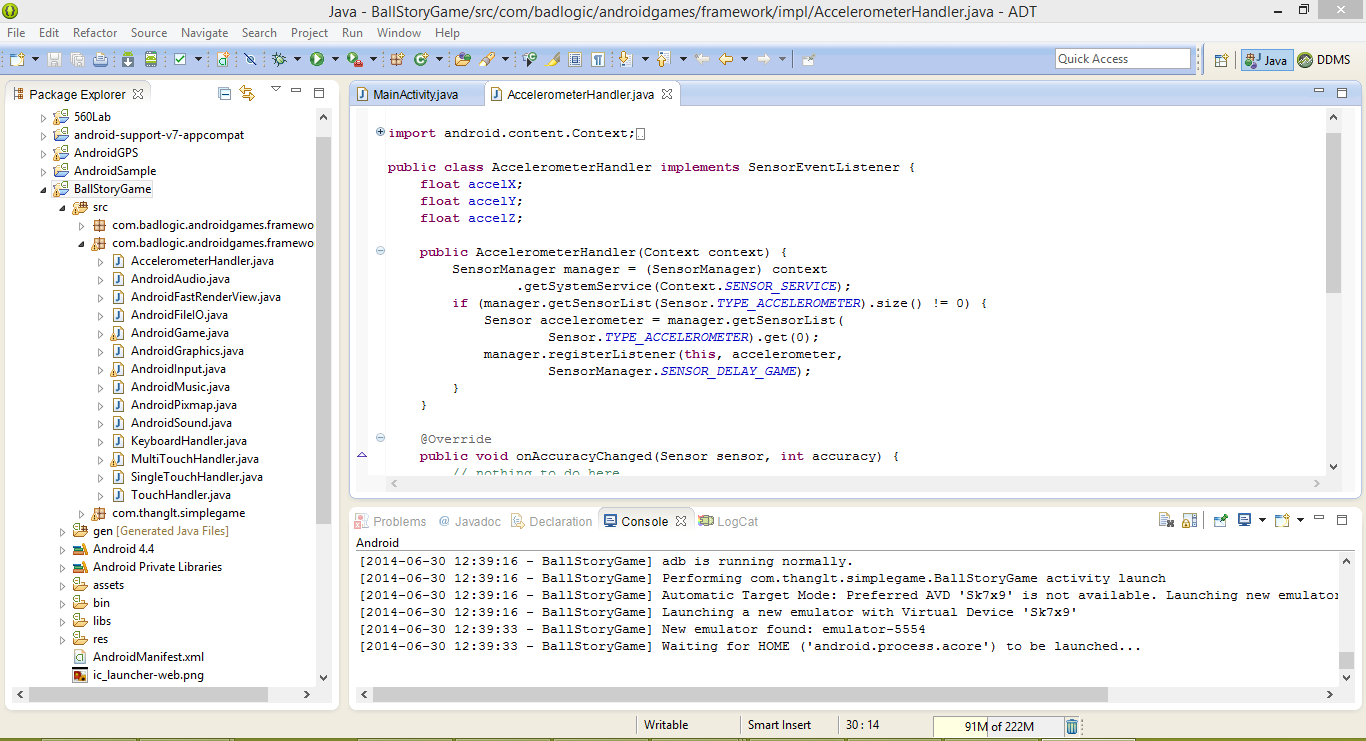
The implementation part typically consisted of three parts. They are:

* Establishing the connection between the sensor tag movement and the open source game actions.
* Collection of the Data and pushing it on to HBase and creating a restful web service by deploying it on the glassfish server.
* Analyzing the collected data based on the actions and sensor movement and classifies the actions based on the classification and clustering algorithms.

First we implement the application using an open source android game called a simple Ball game.

We installed it on our local Eclipse ADT and the package structure is formed as shown below in the frame work.

There is an Accelerometer handler class in the application so that we can control and collect the acceleration data from the application by connection establishment with the sensor tag.

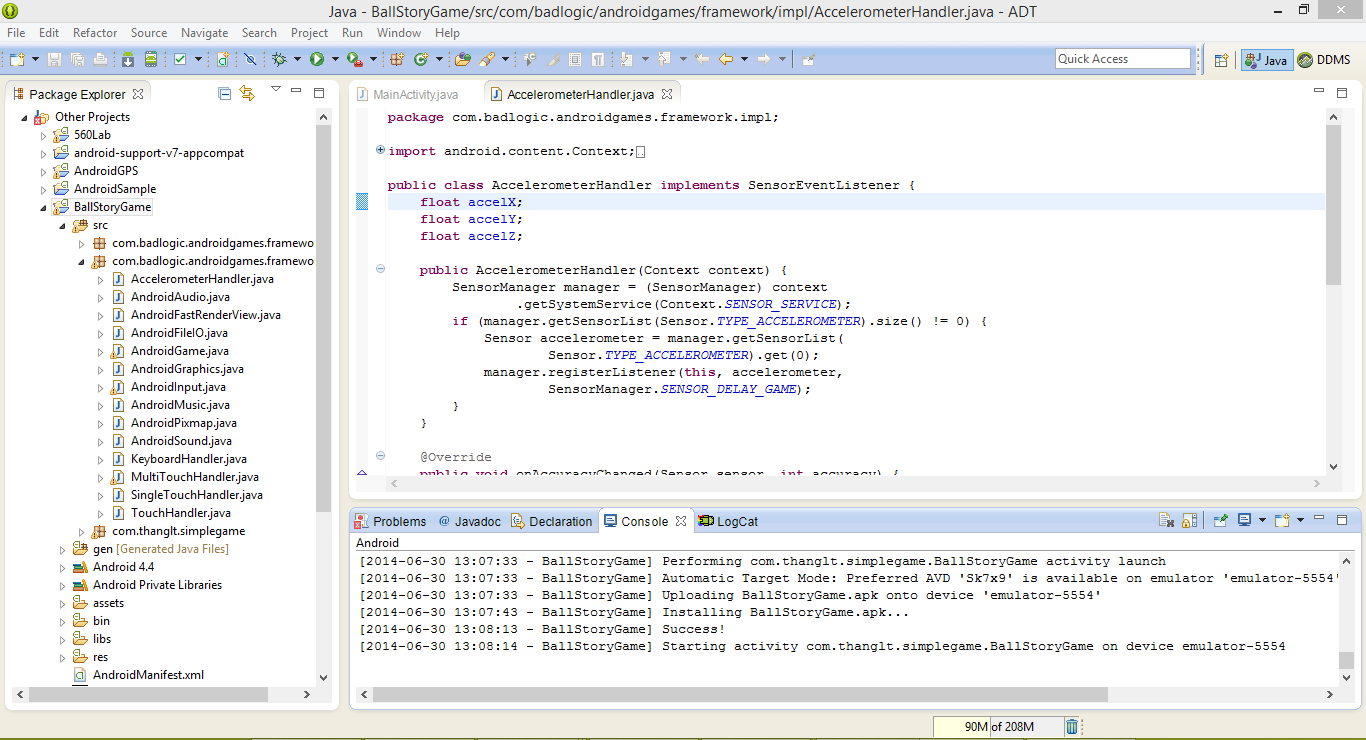
****

It also has the sensor tag related services and applications present in it. We are investigating in the part of the service connections and establishment with the sensor tag.

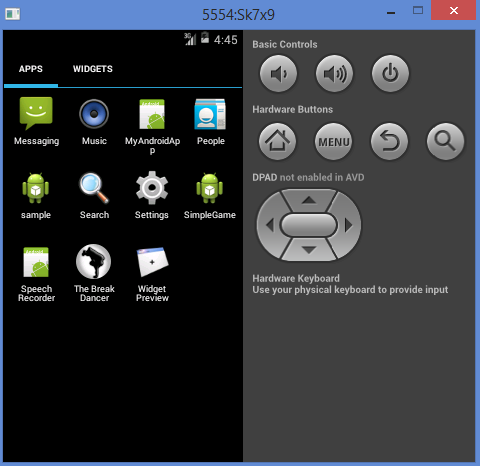
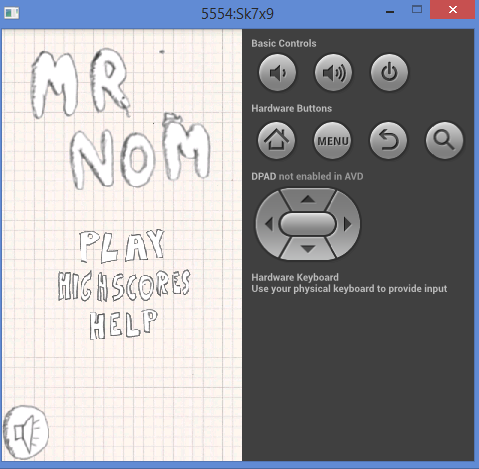
To test the application we are running it on the emulator for performing the basic operations.

We run it as an Android application so that it will be launched on the emulator.

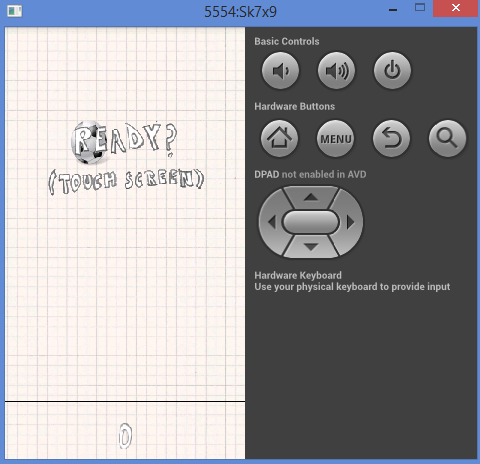
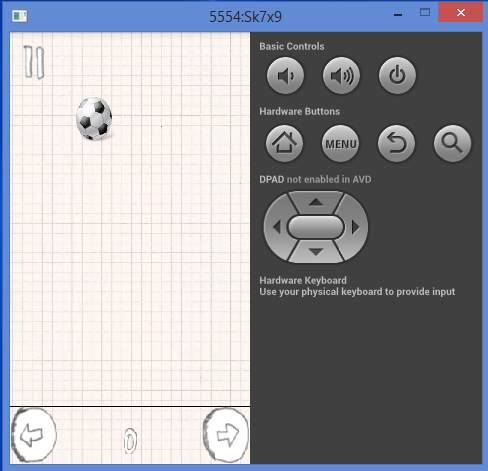
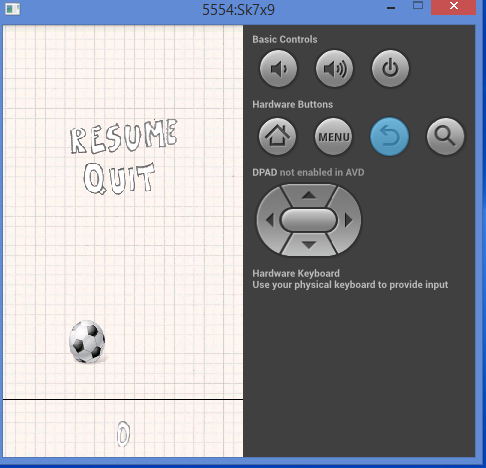
The screen below shows that the emulator is being launched successfully in the console.

****

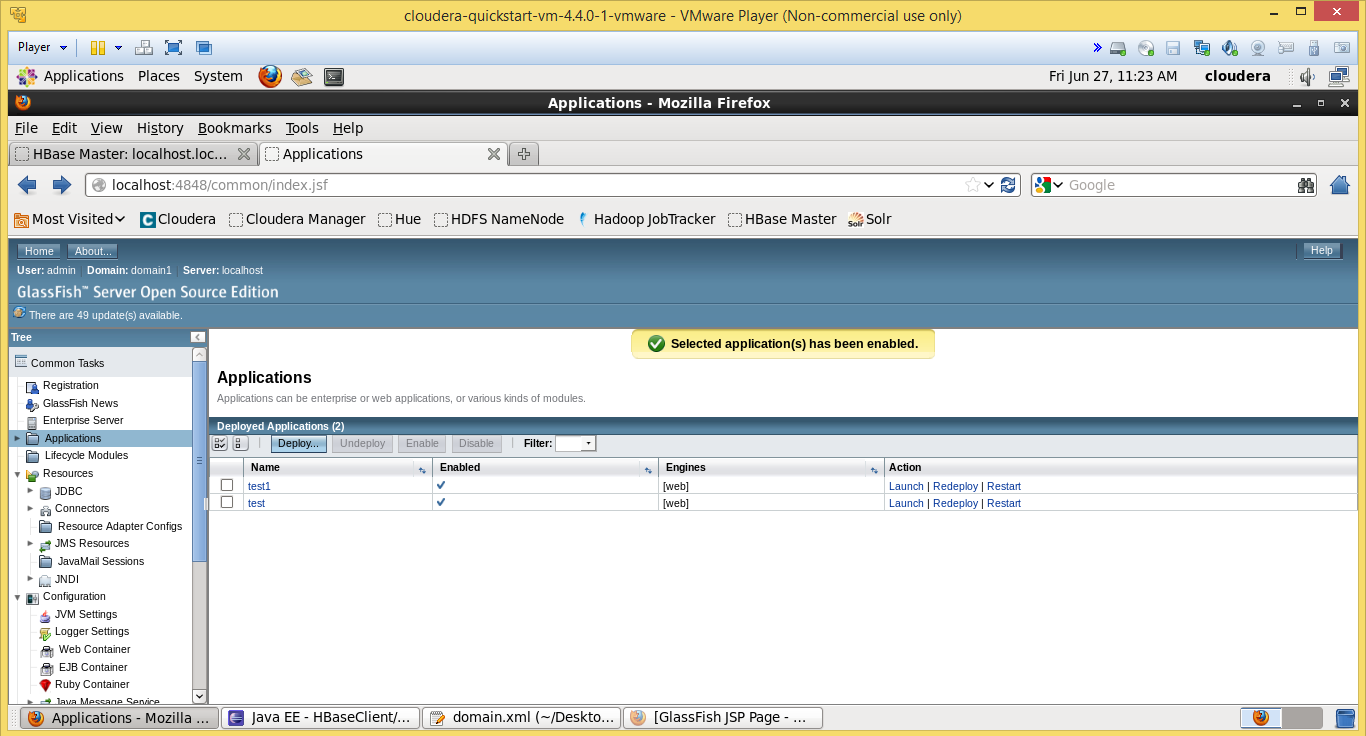
The below figures shows the sequence of basic operations in the open source game. First it shows the app named Sample Game in the UI. After launching it by clicking on the icon we get the page shown as below.

** **

Later when we click on the play it asks to get ready to play the game. After touching the screen it starts moving the ball in all the directions based on the arrow key controls shown in the screen below. The ball moves in all the directions, While this movements we should collect the accelerometer readings for the movement of the ball.

**  **

* We collect the data from these applications and then cluster, analyze it and compare them by using the machine learning algorithms to form grouping of the data based on the actions, direction of movements.
* Here in this application the ball moves in all the directions. We are drilling through for establishing the ball movements connection based on the sensor movement and later collect the data of all the available motions.
* We later on deploy it as a service and we can use the data at any time once the data is hosted as a service.
* We are using glassfish server to install as a restful web service and use the data collected in our applications. We can run the deployed applications as shown by launching them

****

We are deploying some sample data and hosting a web service for it to run as a web server.

**Project Management:**

All the task planning according to the iterations are maintained in scrumdo and task allocation is shared equally among the team members.

<https://www.scrumdo.com/organization/umkc94/dashboard>

* **Third Increment:**

The tasks that will be included for third increment are:

1. Completion of the entire UI changes according to the required actions
2. Establishing a connection between the application and sensor
3. Collection of the data from the application by playing it.
4. Analyze the collected data and use machine learning techniques.

The above mentioned tasks will be uploaded the scrumdo tool with specified timelines.