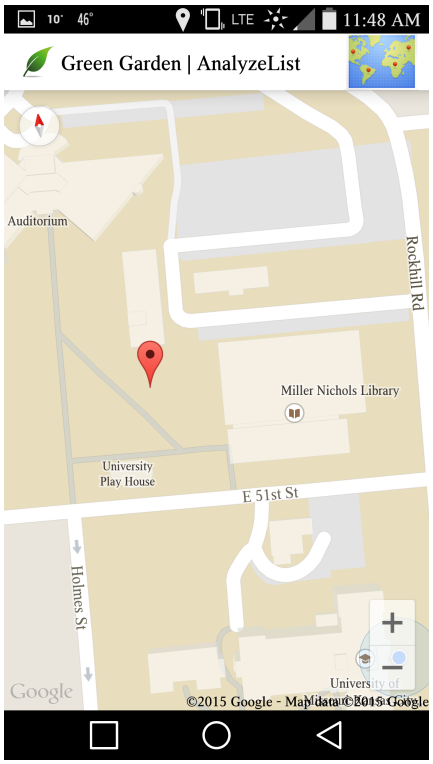
CS551 First Increment Report
By
Rishabh Bhojak (SG2)
Bhargava Gellaboina (SG2)

Import Existing Services/API

Google Maps Android API v2: Coordinator opens the coordinator view in android application which calls the MapView.getMap() function to load the Google maps for the Coordinator. At the same time, android application also pulls all the images using getAllImageData() function using Asynchronous Http Service it displays the images on google maps and shows it to the Coordinator.

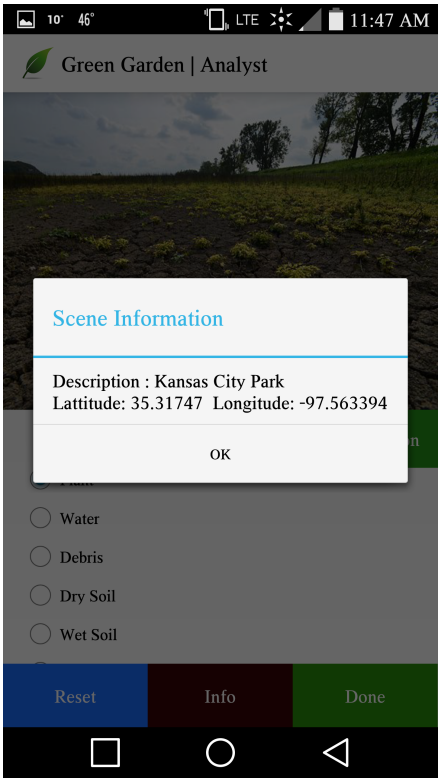
Example code to load the map in android application:

```
public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
        super.onCreateView(inflater, container, savedInstanceState);
        //this.setContentView(R.layout.coordinator_layout);
View rootView = inflater.inflate(R.layout.coordinator_layout,
container, false);
mapView = (MapView) rootView.findViewById(R.id.map);
        mapView.onCreate(savedInstanceState);
        System.out.println("mapview:");
// Gets to GoogleMap from the MapView and does initialization stuff
map = mapView.getMap();
map.getUiSettings().setMyLocationButtonEnabled(false);
map.setMyLocationEnabled(true);
map.addMarker(new MarkerOptions().position(new LatLng(39.035147,
-94.5774246)).title("test"));

MapsInitializer.initialize(this.getActivity());
System.out.println("MapsInitializer:  initialized");

// Updates the location and zoom of the MapView
CameraUpdate cameraUpdate = CameraUpdateFactory.newLatLng-
Zoom(new LatLng(39.035147, -94.5774246), 10);
  map.animateCamera(cameraUpdate);
```



Google Maps Geolocation API: When the observer takes pictures using the application, we are saving Geolocation of the user(where the picture was taken)using geoLocService.getLatitude() and geoLocService.getLongitude() functions in to MongoDB and using the Latitude and Longitude values we are querying for the location using Geolocation API.

```
public void addGeoLocation() {
geoLocService = new GeoLocationService(ObserverActivity.this);
// check if GPS enabled
if (geoLocService.canGetLocation()) {
double latitude = geoLocService.getLatitude();
double longitude = geoLocService.getLongitude();
latTv.setText("Lat: " + latitude);
lonTv.setText("Lon: " + longitude);
double latlon[] = { latitude, longitude };
scene.setLocation(latlon);
} else {
geoLocService.showSettingsAlert();
}
}
```



Detail Design of Services

User case stories(ScrumDo)

#4 Add events as a spinner item to the Android Application

Add events as a spinner item to the Android Application and get the events from MongoDB and add the items to spinner

Todo Tasks | 0 Comments **rbx44, bgz82** 3

#3 Create a login page for the administrator

We will create a login page for administrator and validate the credentials and move to appropriate page

Doing Tasks | 1 Comment loginAdmin **rbx44, bgz82** 5

#2 As a Developer, we want to provide an interface to Administrator to add the events to the MongoDB database

Using Web Page Interface , provide an option to adminsitrator to add the events.

Doing Tasks | 2 Comments addEvents **rbx44, bgz82** 3

#1 As a Developer , we want to use jassor slider and bootstrap to create a webpage in the server

Main purpose of this layout creation is to use the server memory efficiently by loading only six images at a time. User can move to the next six images using the slider option.

Done Tasks | 1 Comment jassor_slider image_loader **rbx44, bgz82** 8

#8 As a developer I want to implement Restful services to get the geo location of the images and show it on google maps.

The main purpose of this is to implement REST service to connect to the server.

Done Tasks | 1 Comment REST **rbx44, bgz82** 5

#7 As a developer I want to set up my server and upload data to Mongo.

Install Apacha Tomcat, update Java, setup Mongo, upload data to Mongo, connect to Application

Done Tasks | 2 Comments Mongo Tomcat Java DataUpload **rbx44, bgz82** 8

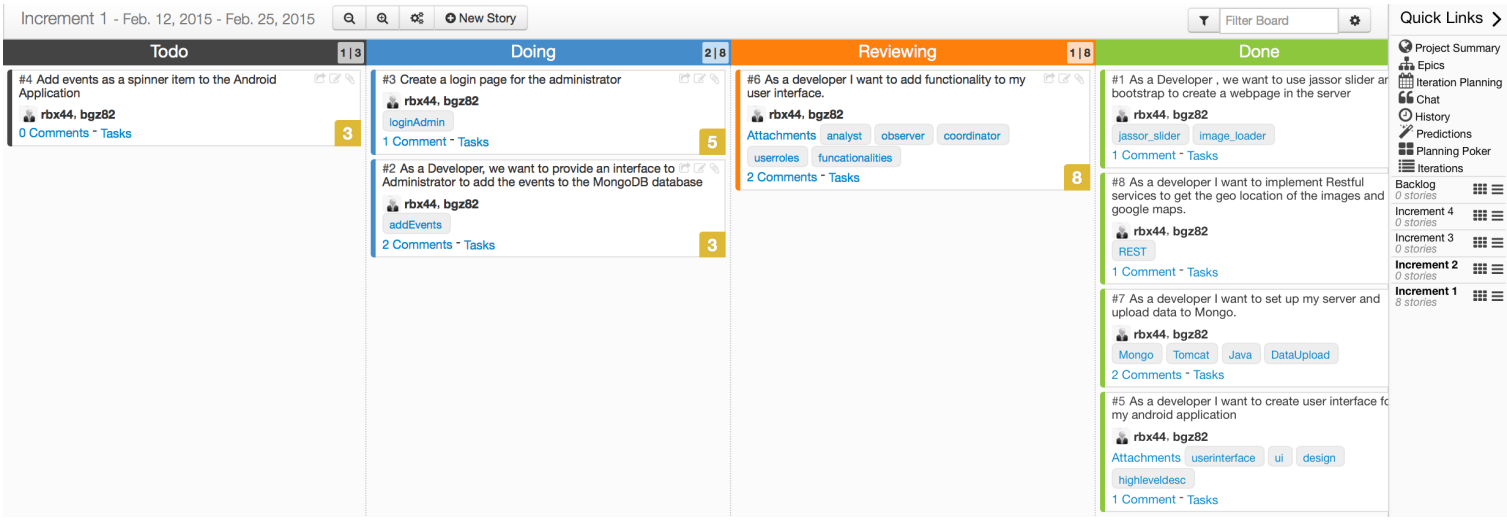
#6 As a developer I want to add functionality to my user interface.

I want to add roles and functionalities for the different types of users in this application, including Observer, Analyst and Coordinator

Reviewing Tasks | 2 Comments | Attachments analyst observer coordinator userroles funcationalities **rbx44, bgz82** 8

#5 As a developer I want to create user interface for my android application

Done Tasks | 1 Comment | Attachments userinterface ui design highleveldesc **rbx44, bgz82** 2



Service Description:

Data Layer:
MongoDB, for backend storage.
We created the database ‘umkc’, as part of this project we created collection ‘green’.
To make the connection with the application and the server.
It makes the Http request to get the ID of the images.
Once the connection has been made and the ID is retrieved it makes another Http request with the MongoDB to get the actual image.
To get the text data associated with image it makes the Http get request to get the attributes(type, degree of damage associated with the image).
Using the Rest service we get the geo location of the images and we are visualizing it in google maps.

Schema – JSON Format fields explained below with an example:

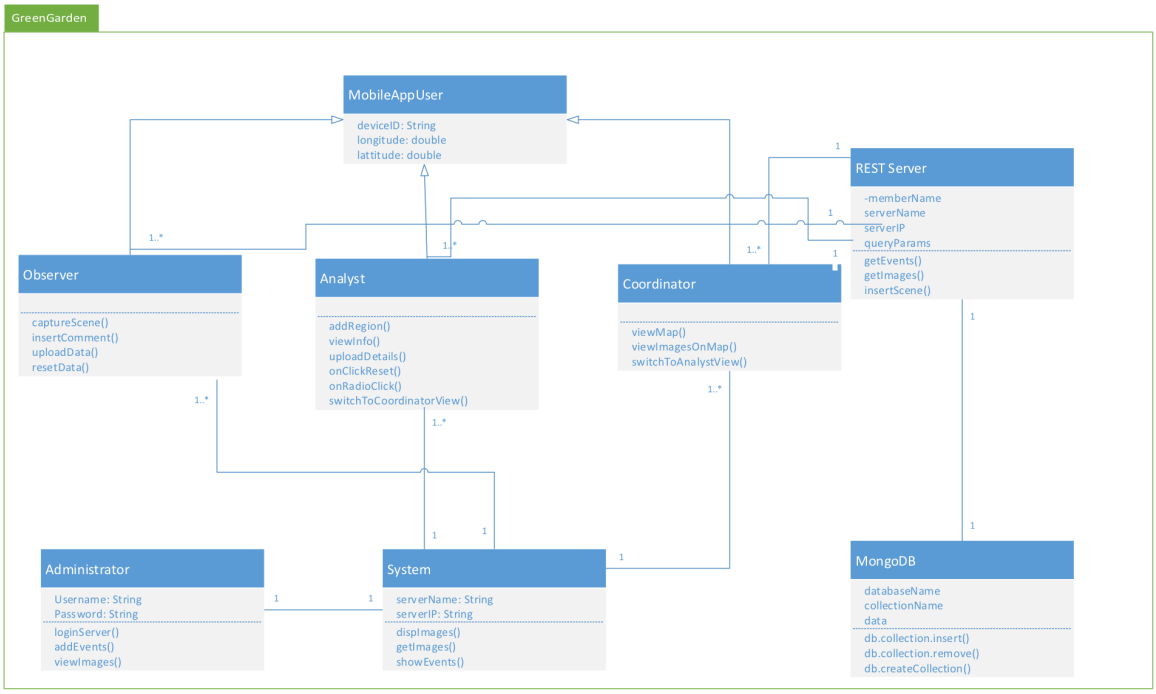
Field	Type	Description
_id and $oid	JSON Object	The id that uniquely refers the object
sceneId	String	default filename is used here

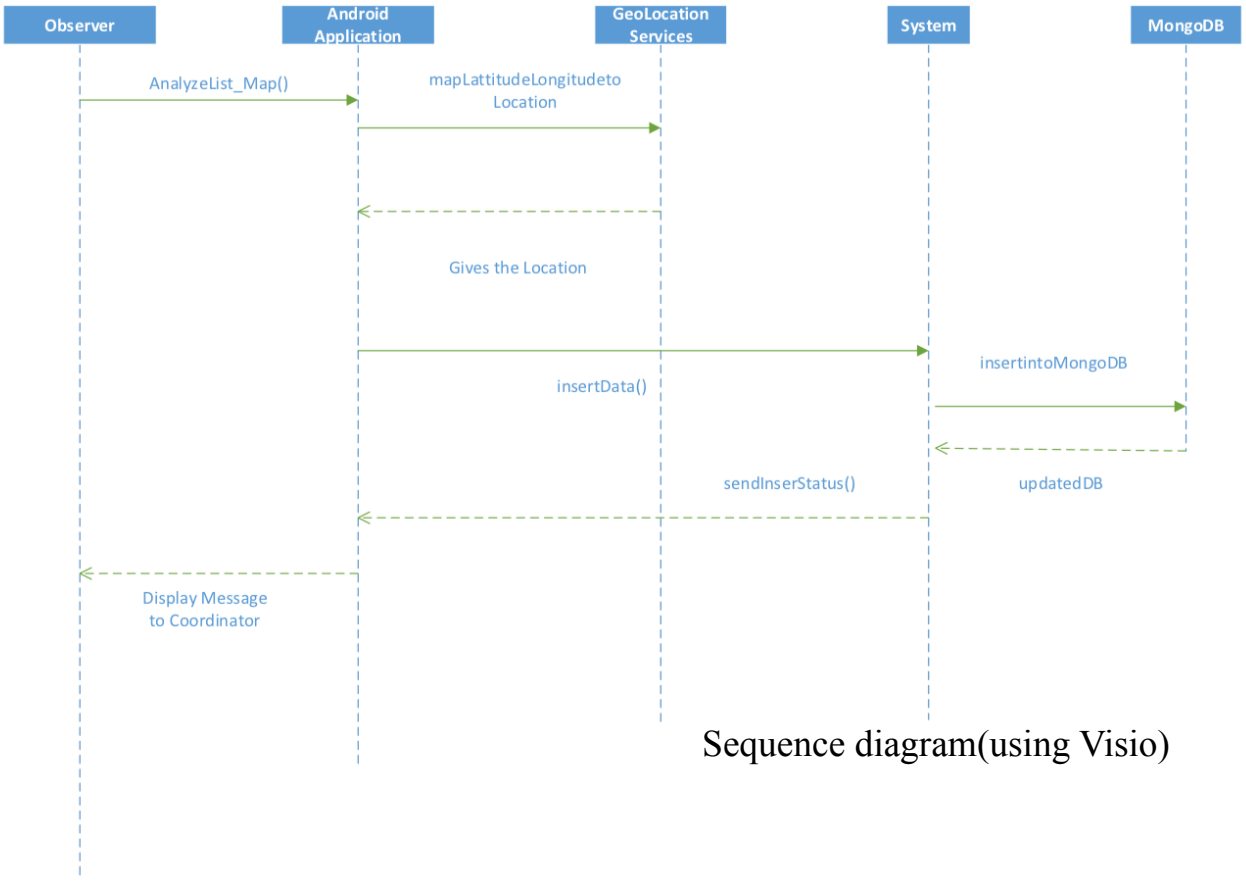
imageData	Json Object	Image information like type and data is stored in JSON format
type(child of imageData)	String	image type - “JPEG, PNG”
data(child of imageData)	String	Image encoded as base64 string
description	String	Scene description
location	Array of double values	Latitude and longitude values stored in JSON array
results	JSON Array	Stores the region information and from the device from which it is marked
deviceId(child of imageData)	String	Network id used to identify the device uniquely
region(child of imageData)	JSON Array	Selected region information is stored
category	Integer	Used to find type of Object
damageLevel	Integer	Damage level of the selected region
boundary	JSON Object	Selected region in the scene

Example


```
{
  "_id": {
     "$oid": "53d26a76e4b0ad6bce969780"
  },
  "sceneId": "http://esridev.caps.ua.edu/MooreTornado/Images/Day3/Christine/IMG_4311.JPG",
  "imageData": {
     "type": "JPG",
     "data": "/9j/4AAQSkZJRgABAgAAAQABAAD/2wBDACgcHiMeGSgjISMtKygwPGRBPD-
c3PHtYXUlkkYCZlo+AjIqgtObDoKrarYqMyP/L2u71////m8H////6/+b9//j/……."
  },
  "description": "One completely damaged building",
  "location": [35.31747, -97.563394],
  "results": [{
     "deviceId": "e8:99:c4:8f:ef:5d",
     "region": "[{\"categroy\":3,\"damageLevel\":1,\"boundry\":{\"bottom\":477.96045,\"right\":
399.5773,\"left\":5.310051,\"top\":145.84058}},{\"categroy\":1,\"damageLevel\":0,\"boundry\":
{\"bottom\":481.42346,\"right\":390.42566,\"left\":-23.818665,\"top\":154.68658}}]"
  }]}
```

Class diagram(using Visio)





Sequence diagram(using Visio)

Design of Mobile Client Interface

Analyst: Responsible for analyzing a provided scene by identifying the type of objects, degree of damage, and marking boundaries.

Coordinator: Responsible for crowdsourcing management and decision- making based on the available scenes and crowdsourcing results.

Observer: Responsible for capturing and collecting the images of disaster scenes.

(a) (b)

(c) (d) (e)

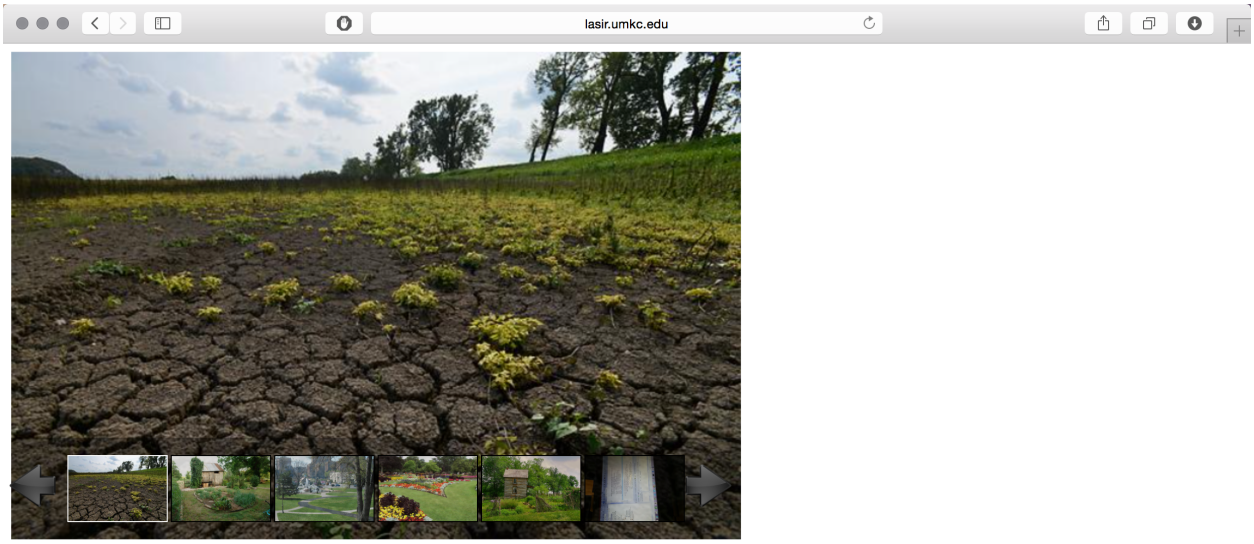
- Main Activity: This is the screen which will be displayed when the application is loaded. The options can be (a) Analyst, (b) Coordinator and (c) Observer. On Selecting Analyst will takes to Analyst List page.

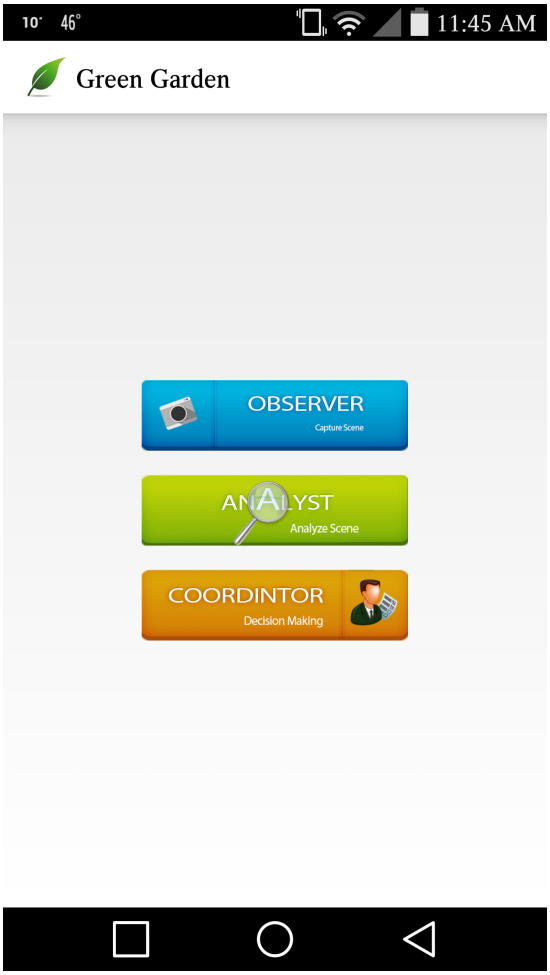
- Analyze Item List: The user will be provided with the list of scenes. Users can switch to Co- coordinator page, load more scenes from the cloud and select a scene to analyze.

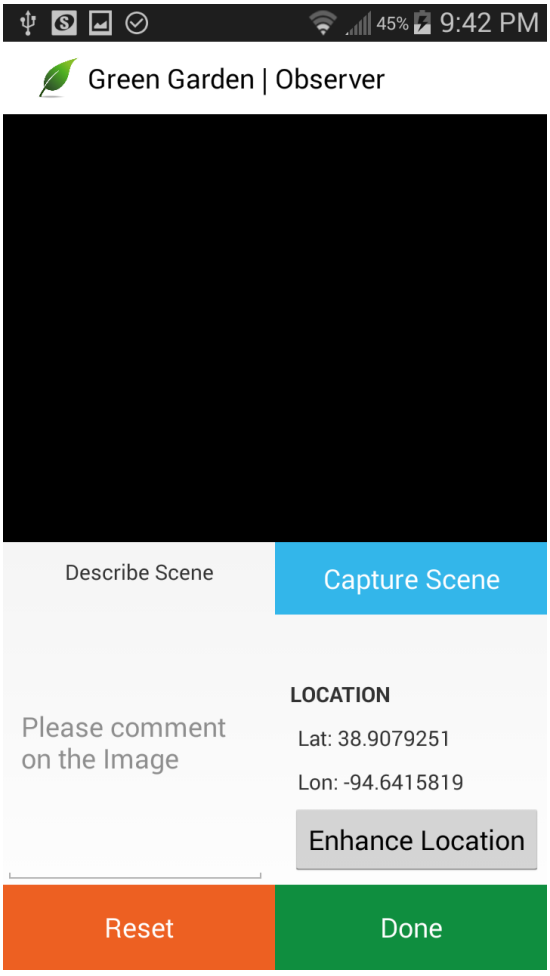
- Analyze A Scene: A scene can be analyzed with the detection techniques. We are still thinking what all methods will be and set the damage degree for each categorized object.

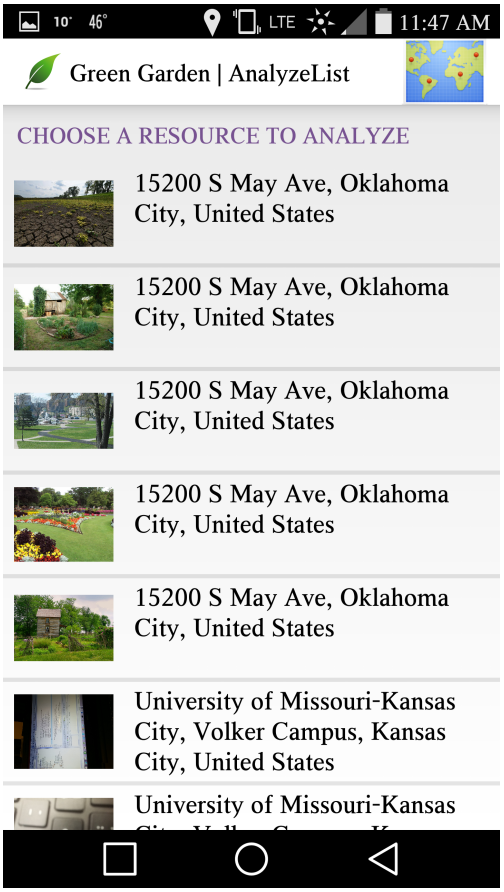
- Co-coordinator View: All the scenes are visualized on a Map with markers representing each scene at a particular location. Selecting a marker provides more details of the crowdsourced information for the selected scene. Once more details are listed, if the user wish to analyze the scene identifying different category objects, it can be done by selecting the Analyze button.

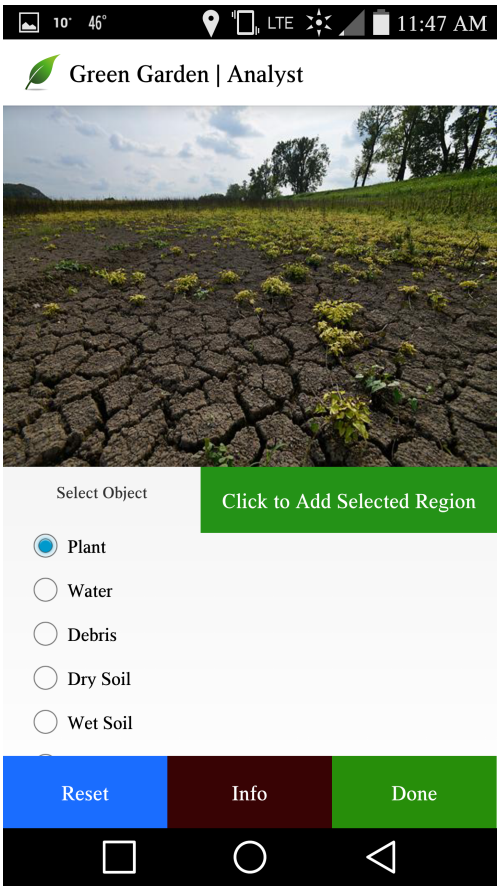
- Observer View: Provides the interface to capture the images. User can take picture. These pictures will be uploaded to the cloud automatically. We are thinking to use MongoDB at the backend.

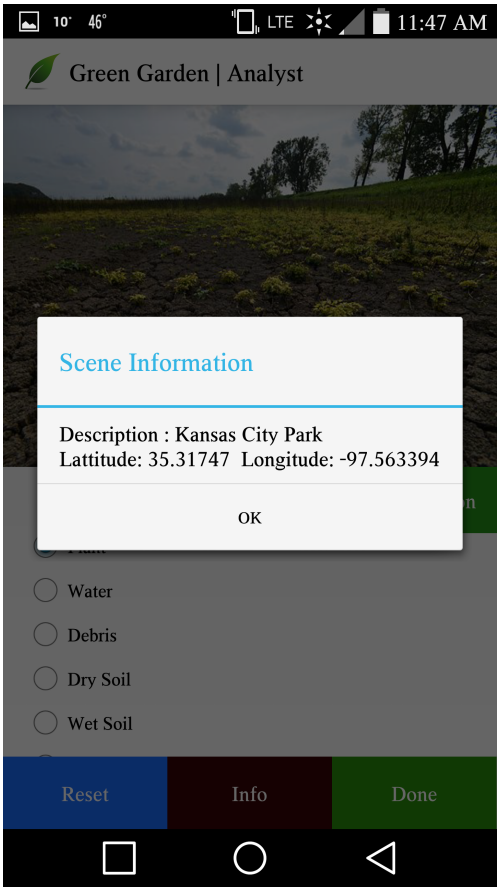

lasir.umkc.edu

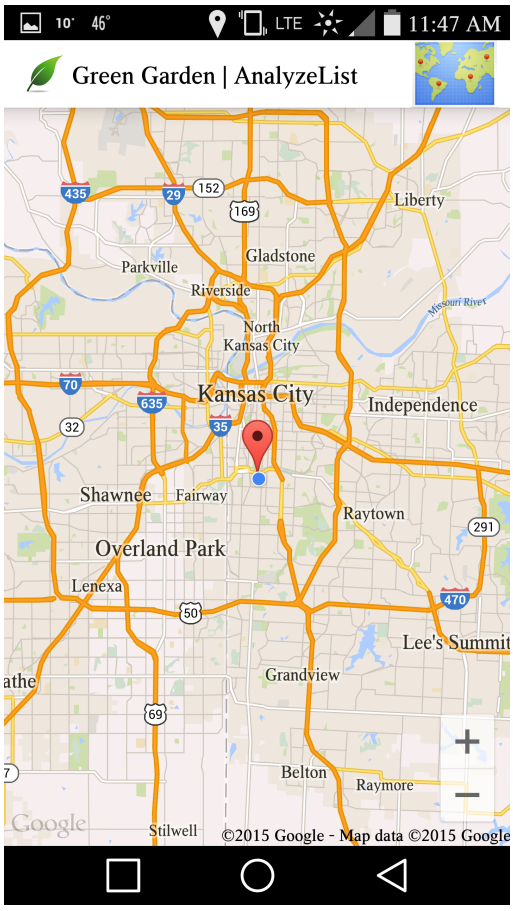

11:45 AM
Green Garden
OBSERVER
Capture Scene
ANALYST
Analyze Scene
COORDINTOR
Decision Making

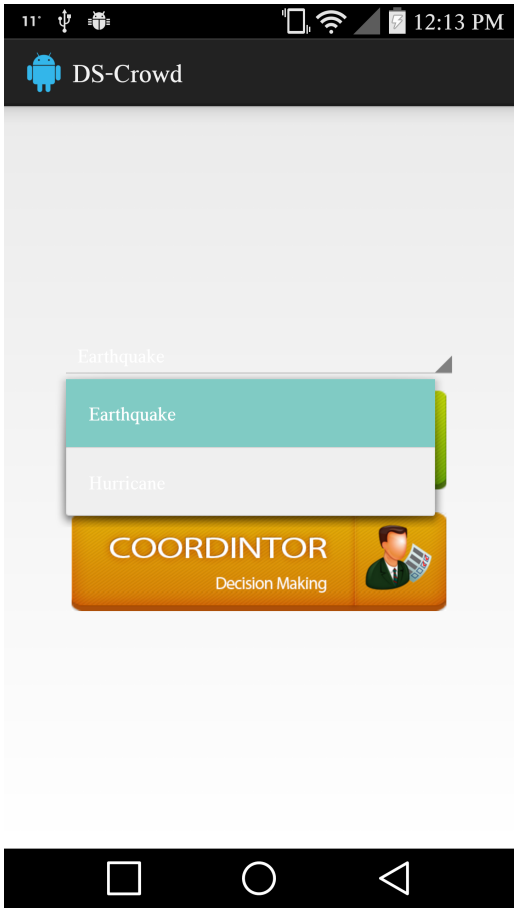

9:42 PM
Green Garden | Observer
Describe Scene
Capture Scene
Please comment on the Image
LOCATION
Lat: 38.9079251
Lon: -94.6415819
Enhance Location
Reset
Done


11:47 AM
Green Garden | AnalyzeList
CHOOSE A RESOURCE TO ANALYZE
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
University of Missouri-Kansas City, Volker Campus, Kansas City, United States
University of Missouri-Kansas


11:47 AM
Green Garden | Analyst
Select Object
Click to Add Selected Region
Plant
Water
Debris
Dry Soil
Wet Soil
Reset
Info
Done


11:47 AM
Green Garden | Analyst
Scene Information
Description : Kansas City Park
Lattitude: 35.31747 Longitude: -97.563394
OK
Water
Debris
Dry Soil
Wet Soil
Reset
Info
Done





Implementation

Rest service is implemented to connect to MongoDB.
Http Request/Response methods are used to pull and push the data.
Below is the example for connecting to MongoDB, and pulling the events data for the Spinner.

```
@Path("/cisa")
public class MyResource {

        @GET
        @Produces("text/plain")
        public String getIt() {
                return "Hi there!";
        }
    @GET
    @Path("events")
    @Produces("text/plain")
    public String getEventsInfo() throws JSONException
     {
            MongoClient mongo;
            JSONObject spinnerArray;
            //JSONArray sArray;
            String result[]= new String[100];
            String concat="";
            int i=0,j=0;
            try {
                    mongo = new MongoClient("lasir.umkc.edu", 27017);
                    DB db = mongo.getDB("umkc");
                    DBCollection table = db.getCollection("events");
                    DBCursor cursor = table.find();
                    while (cursor.hasNext()) {
                            spinnerArray=new JSONObject((cursor.next().toString()));
                            result[i]=spinnerArray.get("name").toString();
                            i++;
                    }
            } catch (UnknownHostException e) {
                    e.printStackTrace();
            }
           for(j=0;j<i;j++)
           {
               concat=concat + result[j] + ";";
           }
           return concat;
     }
```

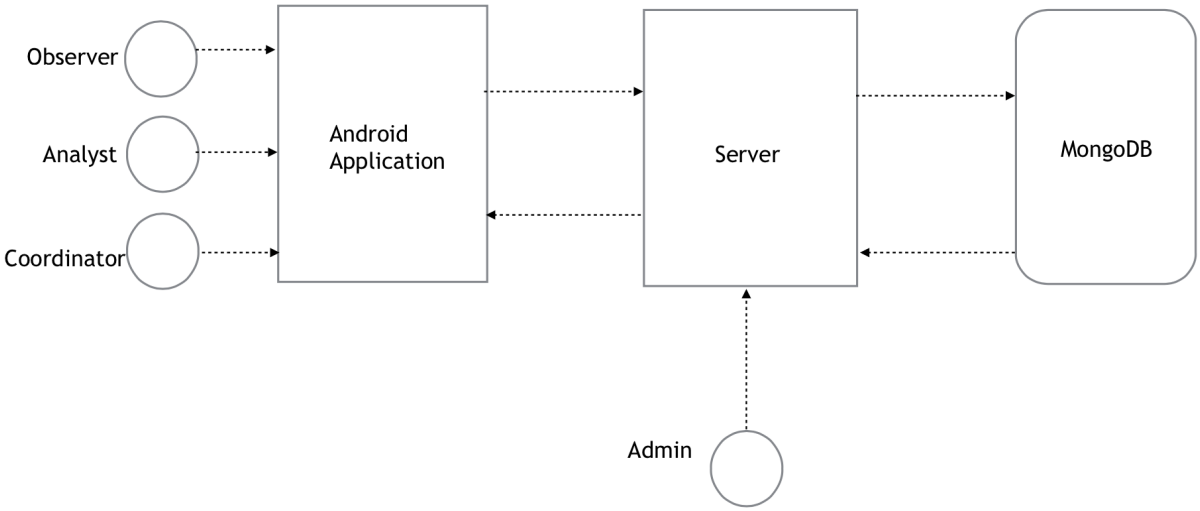
To make the application efficient, we thought of loading only 6 images at a time. To implement the same, we used REST. Here we are making the connection with the Server using REST to pull 6 images at one time. Below is the implementation.

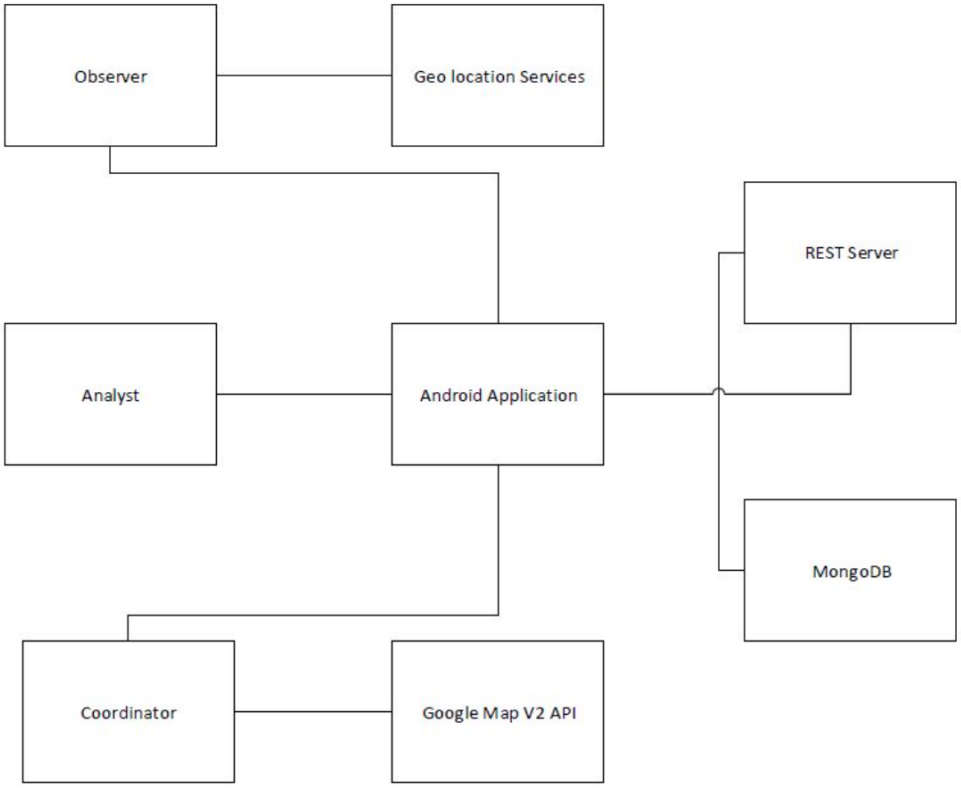
```
  int start=0;
  String[] ids = new String[6];
  try
  {
   URL dest = new URL("http://lasir.umkc.edu:8080/greengarden/webresources/ggarden/sceneids?start="+start+"&limit=6");
   URLConnection yc = dest.openConnection();
   BufferedReader in = new BufferedReader(new InputStreamReader(yc.getInputStream()));
   String inputLine="";
   String getData="";
   int i=0;
   while ((inputLine = in.readLine()) != null)
     {
       getData+=inputLine;
       i++;
     }
   in.close();
   i=0;
   StringTokenizer st=new StringTokenizer(getData,", ");
   while(st.hasMoreTokens())
    {
      ids[i]="http://lasir.umkc.edu:8080/greengarden/webresources/ggarden/image/"+st.nextToken();
      i++;
    }
   }
  catch(Exception e)
   {
     out.println(e);
    }
start=start+6;
%>
```

We are using AJAX to implement the next & previous button to load the next six images or the previous six images. Connection is made through REST service and we also implemented the design using Jssor_slider.

```
    <script>
        jssor_slider1_starter('slider1_container');
    </script>
</div>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
    <script type="text/javascript">
    function getNextImages()
     {
            var ids;
            var check;
            var starts = document.getElementById("start").value;
            console.log(starts);
            $.ajax({
                    url: "http://lasir.umkc.edu:8080/greengarden/webresources/ggarden/sceneids?start="+ starts + "&limit=6",
                    type: 'get',
                    async: false
            })
            .done( function (data, status) {
                    ids = data.split(', ');
                    var i=0;
                    if(data == "")
                     {
                       alert("End of Images");
                       check="0";
                       return;
                     }
                    while(i<ids.length)
                     {
                       ids[i] = "http://lasir.umkc.edu:8080/greengarden/webresources/ggarden/image/" + ids[i];
                       i++;
                     }
                    console.log(ids.length);
             })
             .fail( function (data, status) {
             });
           if(check == "0")
            {
              return;
            }
            document.getElementById("start").value = parseInt(starts) + 6;
            document.getElementById("image1").src = ids[0];
            document.getElementById("image2").src = ids[1];
```

Implementation of User Interface.





Detailed architecture with services

Task #1:
The goal of this project is to build an android application which has the following the system features.
Analyst: Responsible for analyzing a provided scene by identifying the type of objects, degree of damage, and marking boundaries. Coordinator: Responsible for crowdsourcing management and decision- making based on the available scenes and crowdsourcing results. Observer: Responsible for capturing and collecting the images of disaster scenes.

Implemented.

Task #2:
Add roles and functionalities for the different types of users in this application, including Observer, Analyst and Coordinator.

Implemented. However, Analyst needs fixes. When the observer takes the picture and the analyze wants to analyze the same picture, the application crashes.

Task #3:
Install Apacha Tomcat, update Java, setup Mongo, upload data to Mongo, connect to Application

Implemented. Everything works fine.

Task #4:
To implement REST service to connect to the server and MongoDB.

Implemented. Everything works fine.

Task#5:

To use the server memory efficiently by loading only six images at a time. User can move to the next six images using the slider option.

Implemented. Made the memory usage very efficient, thumbnails are not updating though, reviewing.

Task#6:
Using Web Page Interface , provide an option to administrator to add the events.

Implemented. Everything is working.

Task#7
Login page for administrator and validate the credentials and move to appropriate page.

In **doing** stage. We haven't implemented that yet.

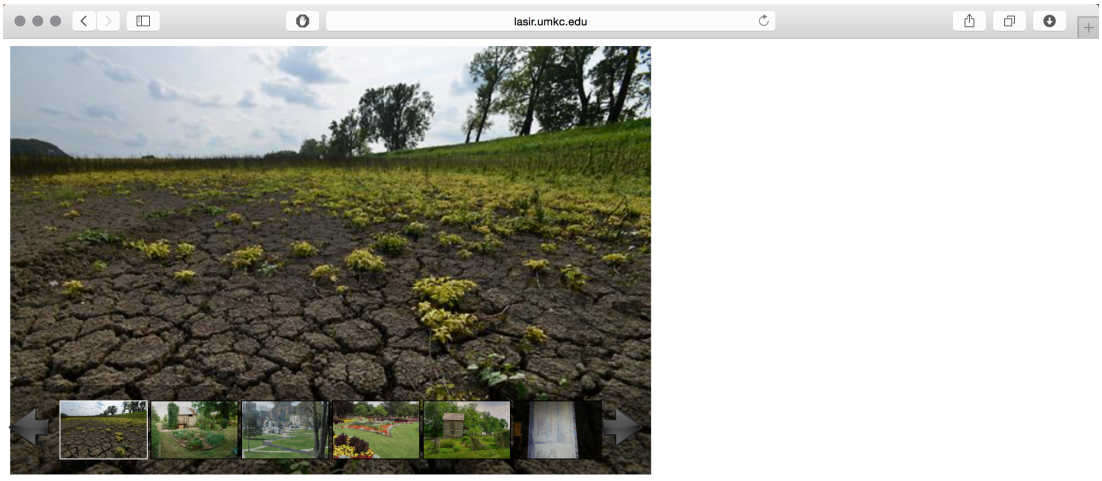
Task #8
Add events as a spinner item to the Android Application and get the events from MongoDB and add the items to spinner
Implemented. However, in the application, when the analyst selects the image it should only upload images associated to that event. Reviewing

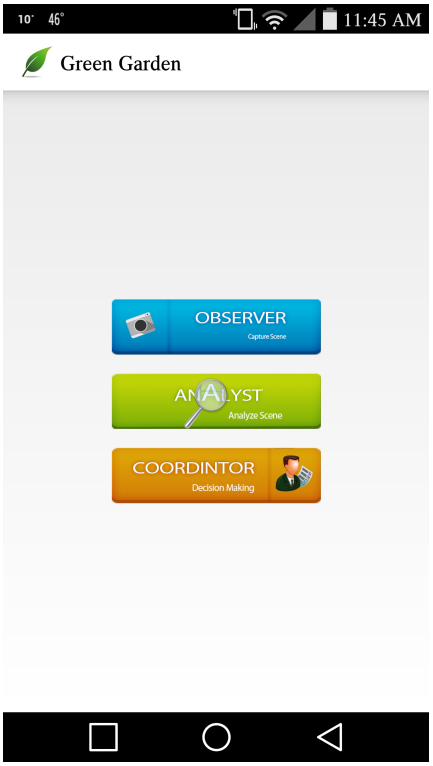
We equally worked on the application, other details can be found in scrumdo account.

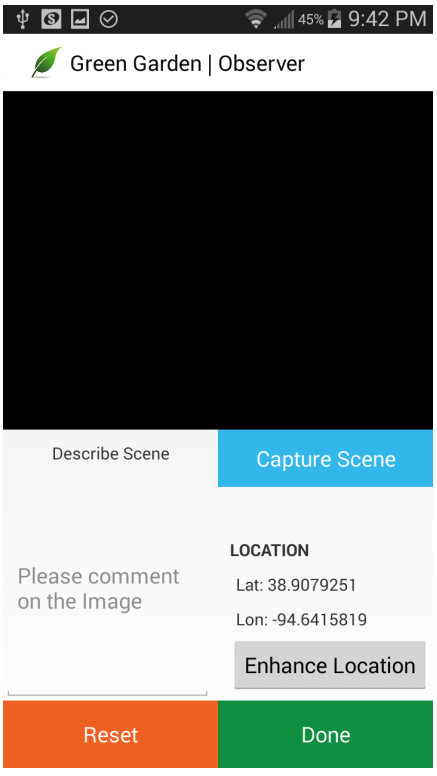
Deployment

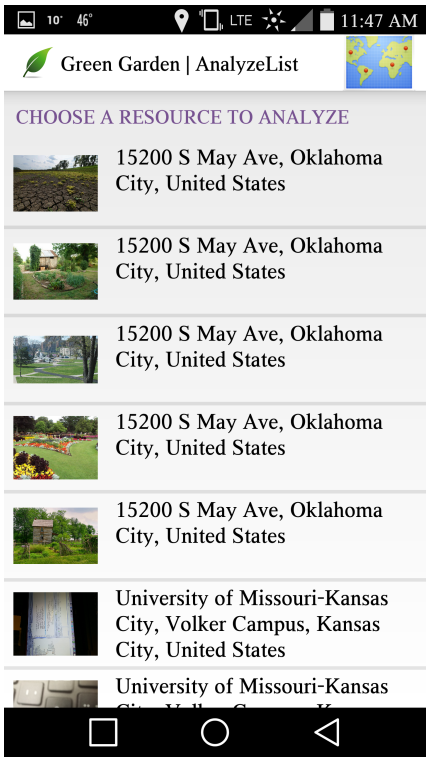
https://www.scrumdo.com/projects/project/dscrowd/iteration/121689/board
http://lasir.umkc.edu:8080/serviceengine
http://lasir.umkc.edu:8080/greengarden
https://github.com/rbx44/ASE

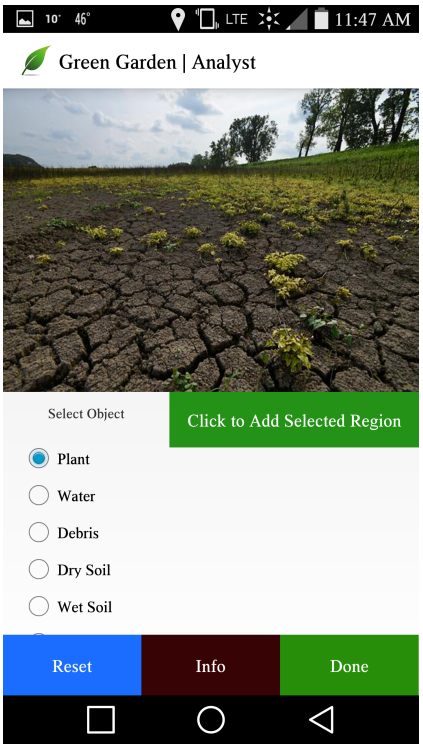
Screenshots:

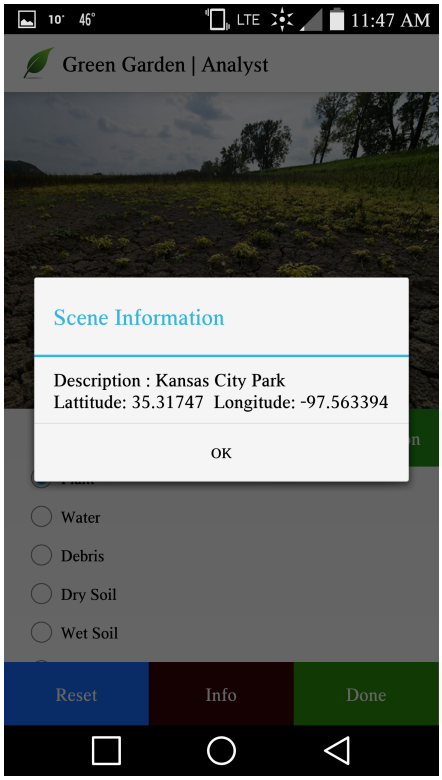


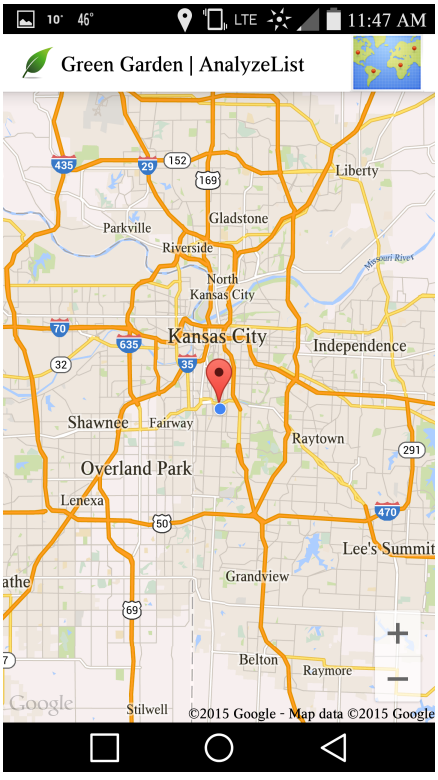


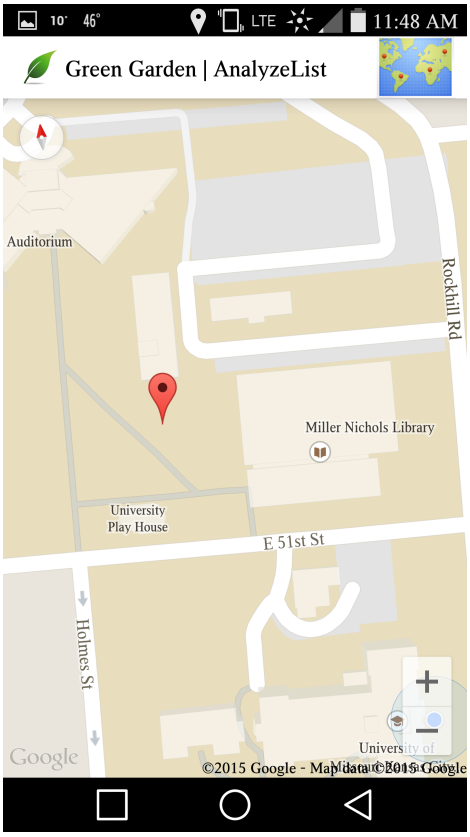



11:47 AM
Green Garden | AnalyzeList
CHOOSE A RESOURCE TO ANALYZE
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
15200 S May Ave, Oklahoma City, United States
University of Missouri-Kansas City, Volker Campus, Kansas City, United States
University of Missouri-Kansas

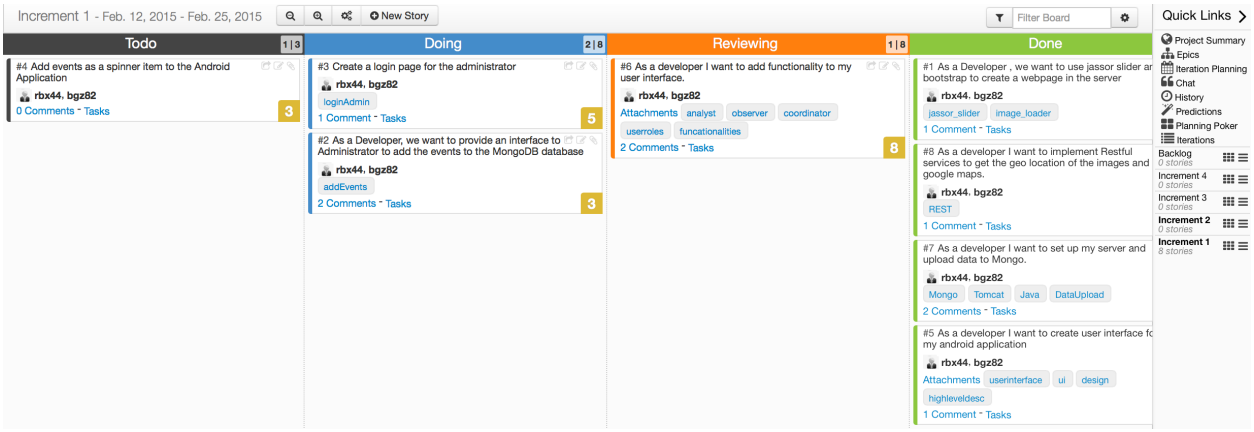

11:47 AM
Green Garden | Analyst
Select Object
Click to Add Selected Region
Plant
Water
Debris
Dry Soil
Wet Soil
Reset
Info
Done


11:47 AM
Green Garden | Analyst
Scene Information
Description : Kansas City Park
Lattitude: 35.31747 Longitude: -97.563394
OK
Water
Debris
Dry Soil
Wet Soil
Reset
Info
Done


11:47 AM
Green Garden | AnalyzeList
Liberty
Gladstone
Parkville
Riverside
North Kansas City
Kansas City
Independence
Shawnee
Fairway
Raytown
Overland Park
Lenexa
Lee's Summit
Grandview
Belton
Raymore
Stilwell
©2015 Google - Map data ©2015 Google


11:48 AM
Green Garden | AnalyzeList
Auditorium
Rockhill Rd
Miller Nichols Library
University
Play House
E 51st St
Holmes St
Google
©2015 Google - Map data ©2015 Google


12:13 PM
DS-Crowd
Earthquake
Earthquake
Hurricane
COORDINTOR
Decision Making


Increment 1 - Feb. 12, 2015 - Feb. 25, 2015
New Story
Filter Board
Quick Links
Todo
Doing
Reviewing
Done
#4 Add events as a spinner item to the Android Application
rbx44, bgz82
0 Comments - Tasks
#3 Create a login page for the administrator
rbx44, bgz82
loginAdmin
1 Comment - Tasks
#2 As a Developer, we want to provide an interface to Administrator to add the events to the MongoDB database
rbx44, bgz82
addEvents
2 Comments - Tasks
#6 As a developer I want to add functionality to my user interface.
rbx44, bgz82
Attachments analyst observer coordinator
userroles funcationalities
2 Comments - Tasks
#1 As a Developer , we want to use jassor slider ar bootstrap to create a webpage in the server
rbx44, bgz82
jassor_slider image_loader
1 Comment - Tasks
#8 As a developer I want to implement Restful services to get the geo location of the images and google maps.
rbx44, bgz82
REST
1 Comment - Tasks
#7 As a developer I want to set up my server and upload data to Mongo.
rbx44, bgz82
Mongo Tomcat Java DataUpload
2 Comments - Tasks
#5 As a developer I want to create user interface fo my android application
rbx44, bgz82
Attachments userinterface ui design
highleveldesc
1 Comment - Tasks
Project Summary
Epics
Iteration Planning
Chat
History
Predictions
Planning Poker
Iterations
Backlog
Increment 4
Increment 3
Increment 2
Increment 1

#4 Add events as a spinner item to the Android Application

Add events as a spinner item to the Android Application and get the events from MongoDB and add the items to spinner

Todo Tasks | 0 Comments **rbx44, bgz82** 3

#3 Create a login page for the administrator

We will create a login page for administrator and validate the credentials and move to appropriate page

Doing Tasks | 1 Comment loginAdmin **rbx44, bgz82** 5

#2 As a Developer, we want to provide an interface to Administrator to add the events to the MongoDB database

Using Web Page Interface , provide an option to adminsitrator to add the events.

Doing Tasks | 2 Comments addEvents **rbx44, bgz82** 3

#1 As a Developer , we want to use jassor slider and bootstrap to create a webpage in the server

Main purpose of this layout creation is to use the server memory efficiently by loading only six images at a time. User can move to the next six images using the slider option.

Done Tasks | 1 Comment jassor_slider image_loader **rbx44, bgz82** 8

#8 As a developer I want to implement Restful services to get the geo location of the images and show it on google maps.

The main purpose of this is to implement REST service to connect to the server.

Done Tasks | 1 Comment REST **rbx44, bgz82** 5

#7 As a developer I want to set up my server and upload data to Mongo.

Install Apacha Tomcat, update Java, setup Mongo, upload data to Mongo, connect to Application

Done Tasks | 2 Comments Mongo Tomcat Java DataUpload **rbx44, bgz82** 8

#6 As a developer I want to add functionality to my user interface.

I want to add roles and functionalities for the different types of users in this application, including Observer, Analyst and Coordinator

Reviewing Tasks | 2 Comments | Attachments analyst observer coordinator userroles funcationalities **rbx44, bgz82** 8

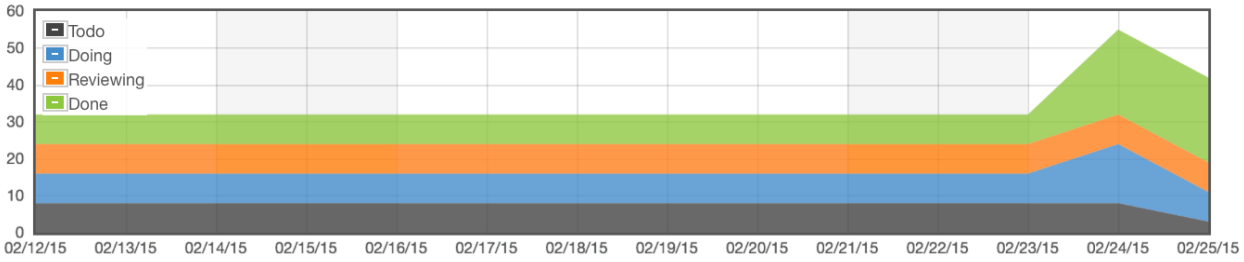
#5 As a developer I want to create user interface for my android application

Done Tasks | 1 Comment | Attachments userinterface ui design highleveldesc **rbx44, bgz82** 2

Increment 1 Feb 12, 2015 - Feb 25, 2015

Scrum Board

Stories	Total Points	Points In Progress	Points Completed
8	42	16	23



Burnup | Burndown | Stacked | Time