1. **Introduction**

The world is facing increased disaster risks due to various factors such as climate change, environmental imbalances, increasing population density, ad-hoc urbanization, deforestation and desertification. If any disaster occurs the emergency services has to respond as quick as possible. The mission of emergency services is to protect valuable lives, ensure public safety and health by addressing emergencies. This project mainly focuses on medical service centers or hospitals as it plays a key role in providing emergency medical services to the victims when any disaster occurs. The location selection decision for such centers is an important strategic issue. The location analyses of medical service centers are often focused on their accessibility. Accessibility is defined as “the relative nearness or proximity of one place to another”. In a broader term, accessibility means ease of reaching opportunities within a reasonable time, cost and comfort.

* 1. **Purpose**

The purpose of this project is to locate the geographical locations in Rajahmundry where new health care service centers has to be established to minimize the risk of losing valuable lives, property when any kind of disaster occurs.

* 1. **Existing system**

Rajahmundry presently has areas within it that are facilitated with medical service centers within 500 meters and can have access to the emergency services as quick as possible. And there are also areas that are unsafe and cannot have access to emergency services when any kind of disaster occurs.

* 1. **Proposed System**

This project aims to approach to suitably situating new emergency service center locations using GIS. This project is implemented within a GIS environment to locate the additional emergency services so as to reduce the response time to disasters in Rajahmundry. Other considerations include categorizing and ranking the city in terms of safe, manageable, and unsafe zones. It also gives longitude and latitudes of the geo locations. It provides open source alternative for good governance.

* 1. **Scope**
* Identify the places in Rajahmundry that are safe which can have access to health care services.
* Rank the places that don’t have access to health care services.
* Provide suitable geographical locations where new health care services can be established.
* Provide longitudes and latitudes of the geographical locations.
  1. **Limitations**
* This project is done on the data of Rajahmundry city only.
* It only focuses on Health care service centers.

1. **Software and Hardware requirements**
2. **Minimum Hardware requirements**
   * CPU: Pentium 42.4 GHZ or AMP 2400xpt
   * System memory(RAM): 512MB
   * Hard disk: 2GB free space N/W speed: 256 Kbps
3. **Software requirements**
   * Operating System
     + Windows

Windows xp and its later versions.

[OR]

* + - Lynux

Ubuntu 12.04 and its later versions.

* + Technologies
    - Openlayers [Open source API]
    - QGIS [ Open source desktop GIS]
    - PostgreSQL/PostGIS [Open source ORDBMS]
    - XAMPP 1.8.0

OpenLayers is an open source JavaScript library for displaying map data in web browsers. It provides an API for building rich web-based geographic applications similar to Google maps and Bing maps.

QGIS is a cross-platform and open source desktop geographic information systems (GIS) application that provides data viewing, editing, and analysis capabilities. QGIS allows users to create maps with many layers using different map projections. A map projection is a systematic transformation of the latitudes and longitudes of locations on the surface of the sphere or an ellipsoid (3D) into locations on a plane (2D).

PostgreSQL often simply “Postgres” is an object- relational database management system (ORDBMS) with an emphasis on extensibility and standards-compliance. As a database server, its primary function is to store data, securely and supporting best practices, and retrieve it later, as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). It can handle workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users. PostGIS is an open source software program that adds support for geographic objects to the PostgreSQL. PostGIS follows the simple features for SQL specification from the Open Geospatial Consortium (OGC). The OGC is an international voluntary consensus standards organization, originated in 1994.

XAMPP is a free open source cross-platform web server solution stack package, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP (PHP: Hypertext Processor) and Perl programming languages. XAMPP actually supports MySQL but necessary changes are made in the configuration file to support the PostgreSQL/PostGIS ORDBMS needed for this project.

1. **Literature Survey**

**3.1 Introduction to GIS**

“A **Geographic Information System (GIS)** is a system used for capturing, storing, manipulating, querying, analyzing and displaying geographically-referenced data related to positions on the earth’s surface”.

GIS enables us to better plan and manage the information all around us – taking information like maps and making it digital to allow for change and analysis. Typically, GIS is used to display data of one kind or another. This data might be represented as several different layers, in which each layer holds information about a particular kind of feature. Each feature is linked to a position on the map and a record in an attribute table.

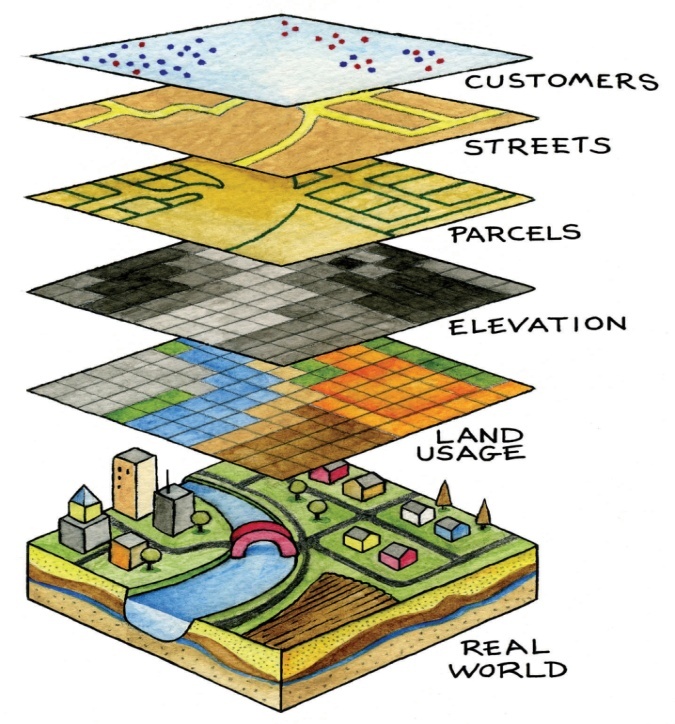


Figure 3.1.1: Representation of data in GIS

With GIS, we can explore the spatial element of our data to display soil types, tree stands, find the best location for a new business and make decisions for many types of complicated problems.

GIS simplifies decision making by providing quick, accurate information that can be used to help with economic development, capacity building, planning and maintenance. There is almost unlimited potential in GIS technology.

* 1. **Introduction to Web-GIS**

**Web-GIS** started with an idea of serving dynamic and interactive maps on-line. Web-GIS allows us to view, edit and manipulate spatial data over the web.

Distributing geospatial information on the Internet is an enforcing factor for information providers. Internet allows users to access geospatial information, and provides a media for processing geo-related information with no location restrictions. Web-based GIS is evolved from different web maps and client server architecture to distributed ones. Internet reshapes all functions of information systems including: gathering, storing, retrieving, analyzing, and visualizing data.

Development of the web and expansion of the internet provide two key capabilities that can greatly help geoscientists. First, the web allows visual interaction with data. By setting up a web server, clients can produce maps. Since the maps and charts are published on the internet, other clients can view these updates, helping to speed up the evaluation process. Second, because of the near ubiquitous nature of the internet, the geospatial data can be widely accessible. Clients can work on it from almost any location.

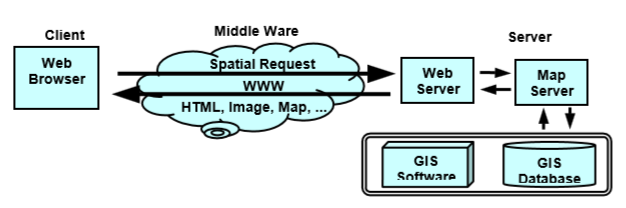


Figure 3.2.1: How a typical Web GIS model works

In performing the GIS analysis tasks, Web GIS is similar to the client/server typical three-tier architecture. The geo-processing is breaking down into server-side and client-side tasks. A client typically is web browser. The server-side consists of a Web Server, Web GIS software and Database.

The following is the image of a Google map that is retrieved using web GIS technology.

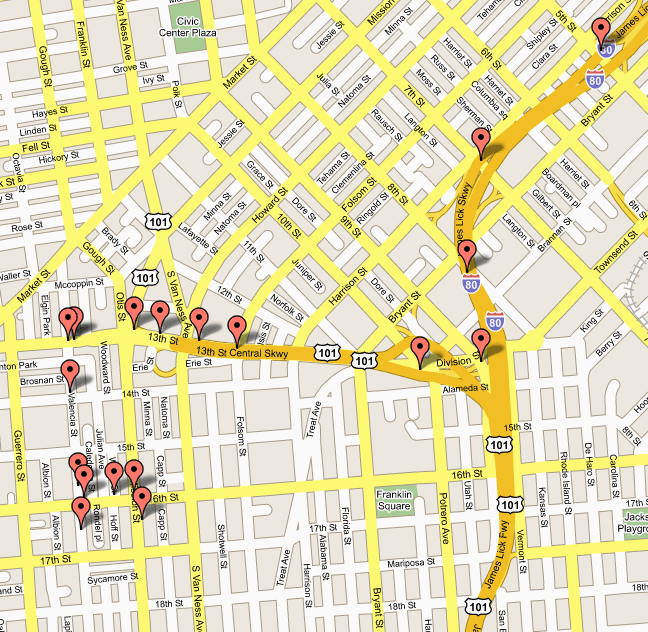


Figure 3.2.2: A map which is retrieved using web-GIS technology

1. **Software Requirement Analysis**
   1. **Define** **the** **problem**

This project mainly focuses on selecting the suitable geographical area for establishing new health care service centers. As this project is done on Rajahmundry data, its data is to be analyzed by finding the areas which are:

* **Safe** that can have access to health care services within 500 meters.
* **Manageable** that can have access to health care services within 1.5 kilometers.
* **Unsafe** that cannot have access to health care services when any disaster occurs.

Now, the task is to find out the geographical locations where new health care service centers can be established in the areas which are categorized as unsafe by also considering the area’s accessibility i.e., its nearness to the road.

* 1. **Modules**
     1. **Analyzing the data**

The data is analyzed using GIS tools. The GIS tool used for analyzing the data is QGIS. The following are the screen shots of the data that is viewed using desktop GIS tool i.e., QGIS.

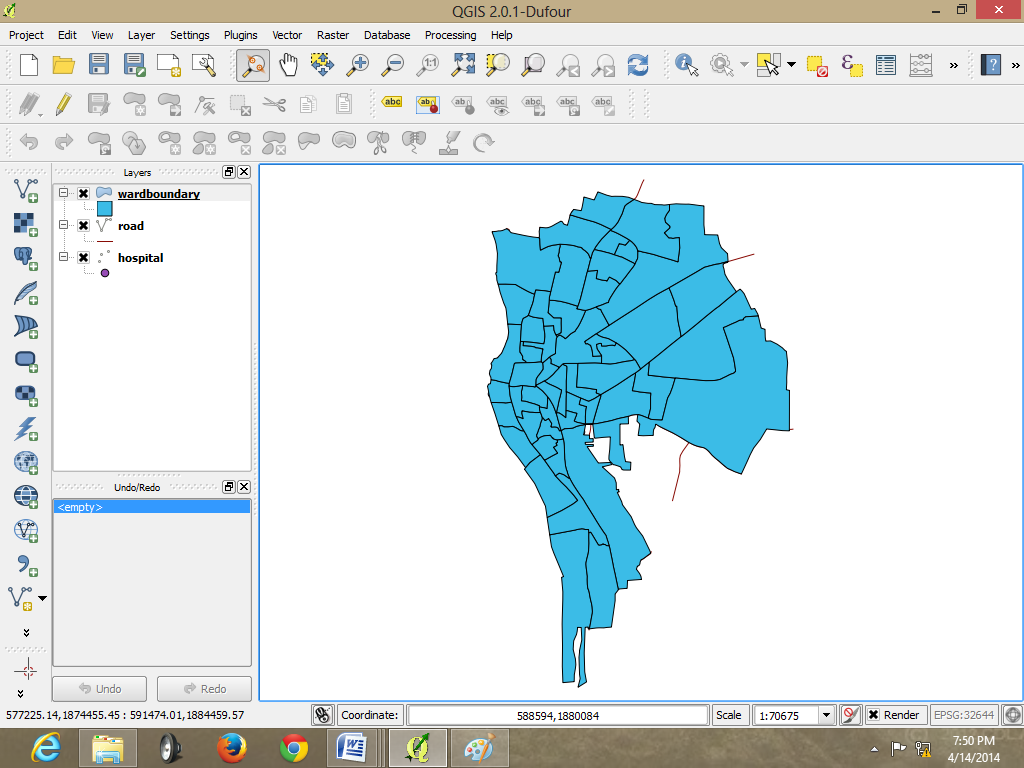


Figure 4.2.1.1: Rajahmundry ward information viewed in QGIS

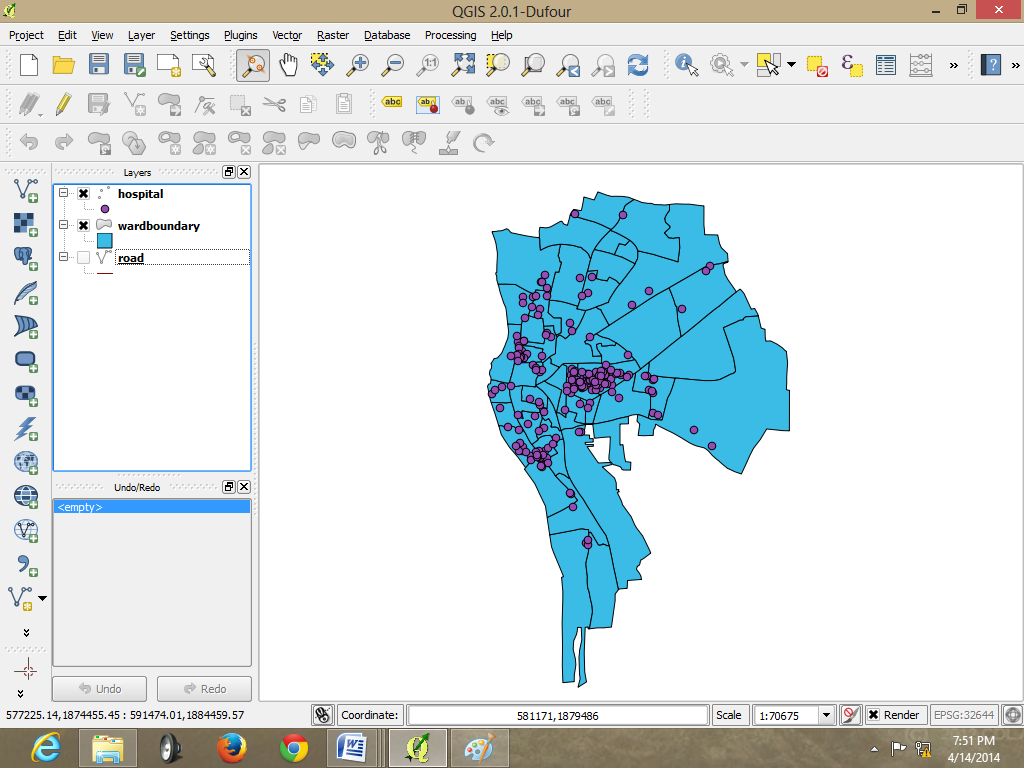


Figure 4.2.1.2: Currently situated hospitals in Rajahmundry viewed in QGIS

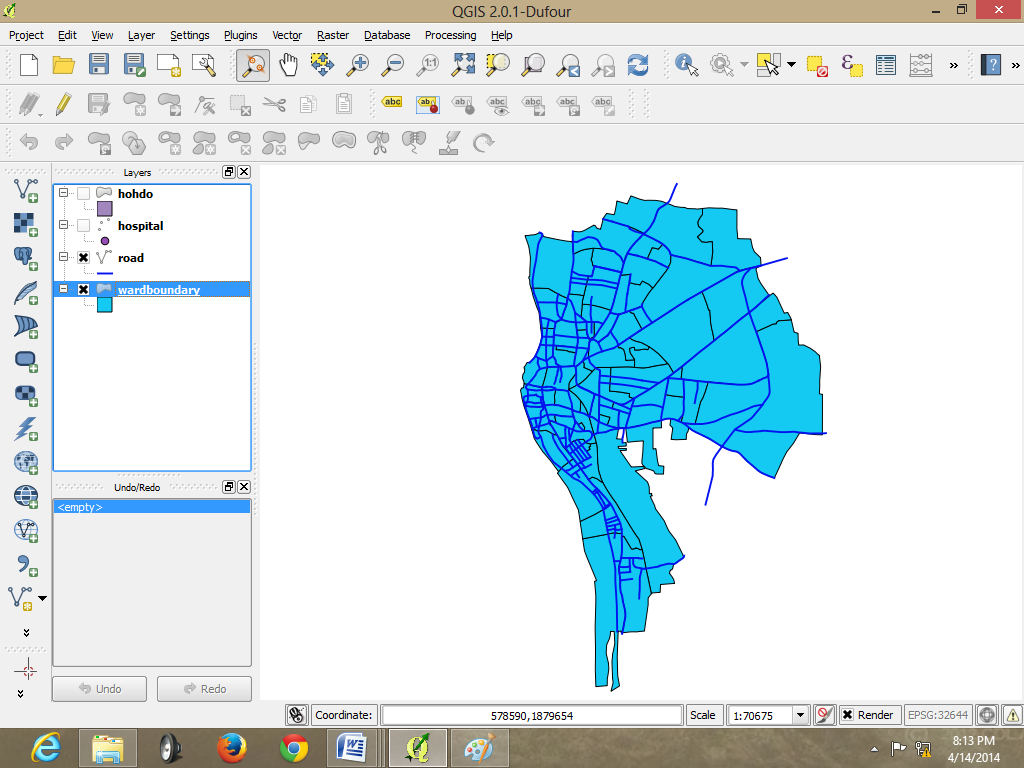
****

Figure 4.2.1.3: Road network of Rajahmundry viewed in QGIS

* + 1. **Categorizing and ranking the area of Rajahmundry**

Area of Rajahmundry is to be categorized in to three zones. They are Safe zone, Manageable zone, Unsafe zone. This can be achieved using database querying.

* + 1. **Finding geographic locations for new hospitals**

Once we get areas which are in safe, manageable and unsafe zones, the next step is to find the geographic locations for establishing new hospitals in unsafe zone. These geographic locations can be achieved using database querying again. It is also checked for its accessibility by considering the road network. All the query results have to be viewed in the web-GIS which can be achieved using OpenLayers.

* + 1. **Longitude and latitude information**

This project also gives information regarding the longitudes and latitudes of the geographical locations. By clicking on the map it gives information regarding the longitude and latitude of the point where we clicked. This can be achieved using OpenLayers.

1. **Software Design**

**5.1 UML Diagrams**

**5.1.1 Use case diagram for Establishing new Health care Service Centers**

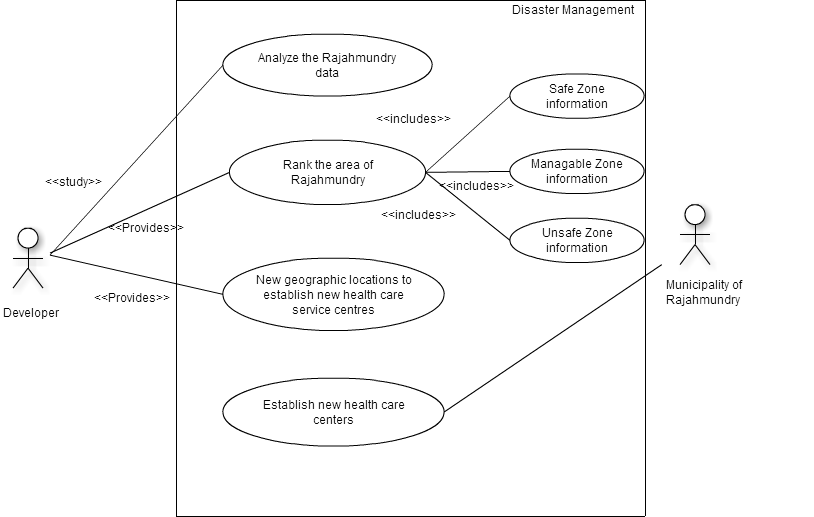


Figure 5.1.1 Use case diagram for Establishing new Health care Service Centers

The above diagram is a use case diagram for establishing new emergency service centers. In this diagram the actors are ‘Developer’ and ‘Municipality of Rajahmundry’. Developer first analyzes the data of Rajahmundry. And then rank the area of Rajahmundry in terms of safety. Then, provide new geographical locations to establish new hospitals. Municipality of Rajahmundry uses this work for the development of Rajahmundry.

**5.1.2 Class Diagram for establishing new health care service centers**

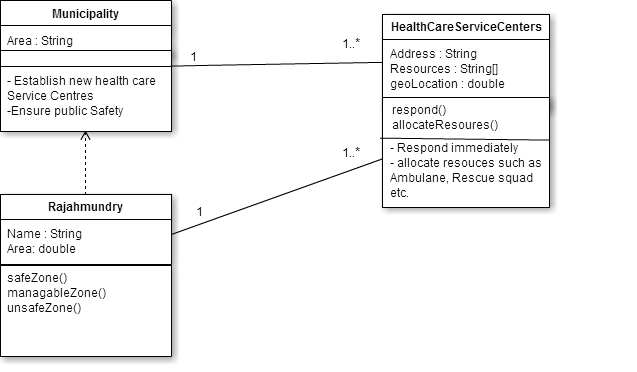


Figure 5.1.2 Class Diagram for establishing new health care service centers

Above diagram describes the class diagram for ‘Establishing new Health Care Service Centers’. The classes are ‘Rajahmundry’, ‘Municipality’, ‘HealthCareServiceCenter’, the relationship between classes as follows:

* One Municipality may establish one to many Health care service centers.
* Rajahmundry may contain one to many HealthCareServiceCenters.
* Rajahmundry is dependent on Municipality.

**5.1.3 Sequence Diagram for Disaster management**

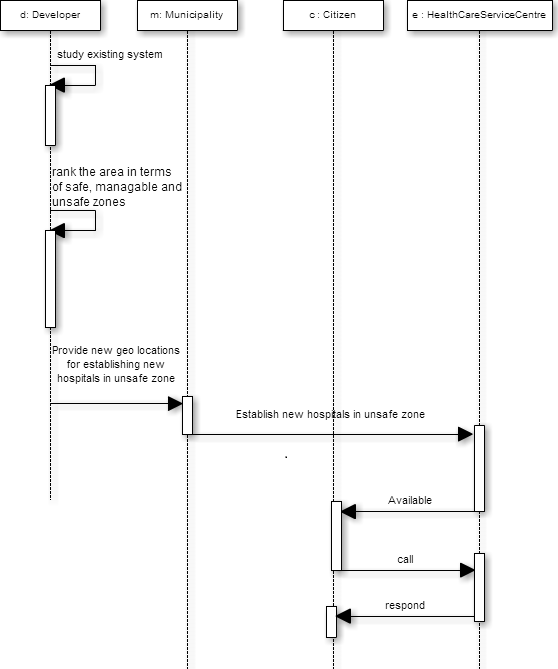


Figure 5.1.3 Sequence Diagram for Disaster management

The above describes the sequence diagram for establishing new health care service centers. In this diagram the objects are ‘d:Developer’, ‘m:Municipality’, ‘c:Civilian’, ‘e:EmergencyServiceCenter’. First d studies the existing system and provides the new geographical locations to the m. m establish new emergency service centers. Now e is available for c. When c makes a call e responds.

**5.1.4 Collaboration Diagram for Disaster Management**

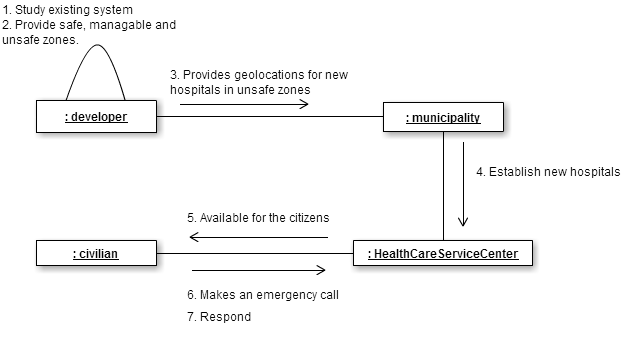


Figure 5.1.4 Collaboration Diagram for Disaster Management

The above diagram describes the collaboration diagram for disaster management. In this diagram the developer collaborates with the municipality by providing appropriate geographic locations for establishing new hospitals. municipality establishes new hospitals in the locations suggested by developer. Now, these hospitals are made available for the citizens. When civilian makes call emergency services responds.

**5.2 Database Design**

The ORDBMS (Object Relational Database Management System) used in this project is PostgreSQL. To support the spatial data PostgreSQL is extended with PostGIS.

This project uses only one spatial database named “Rajahmundry”. This database has the required tables namely, “hospital”, “road”, “wardboundary”.

**Table** **‘hospital’**

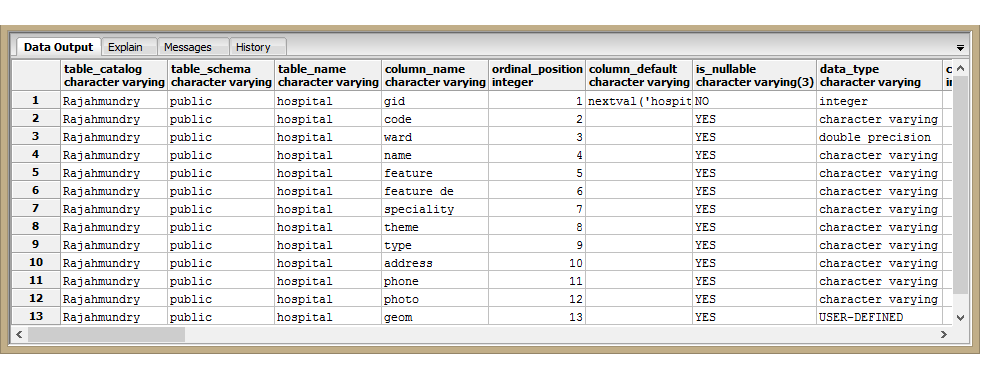


Figure 5.2.1 ‘hospital’ table viewed in pgAdmin III

**Table ‘road’**

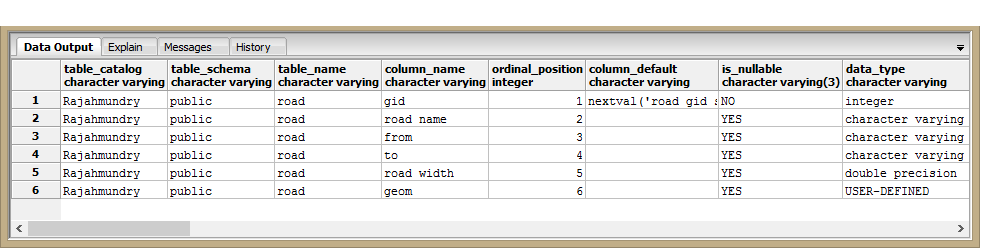
****

Figure 5.2.2 ‘road’ table viewed in pgAdmin III

**Table ‘wardboundary’**

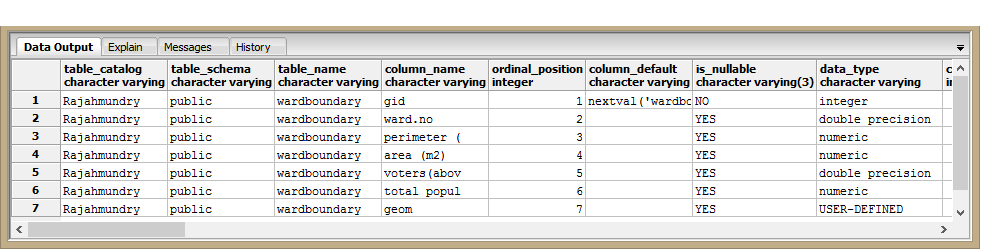
****

Figure 5.2.3 ‘wardboundary’ table viewed in pgAdmin III

**5.2.1 E-R Diagram for Disaster Management**

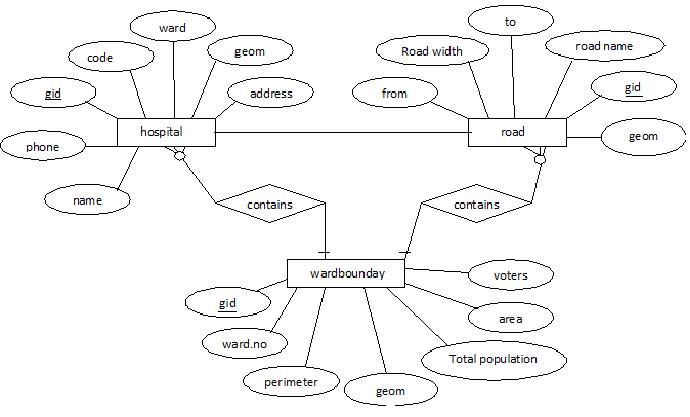
****

Figure 5.2.1.1 E-R diagram for Disaster Management

This E-R diagram had three entities hospital, road and wardboundary. All the entities have primary keys.

* hospital is associated with road.
* One wardboundary contains many hospital
* One wardboundary contains many road

1. **CODING**

<! doctype html>

<html>

<head><title>Disaster Management</title>

<h1 align=center style="color:lightyellow;"> Web GIS for Disaster Management</h1>

<style> body {background-color:#A8AEB2;} </style>

<script type='text/javascript' src='OpenLayers.js'></script>

<?php

$host = "host=localhost";

$port = "port=5432";

$dbname = "dbname=Rajahmundry";

$credentials = "user=postgres password=1234";

$db = pg\_connect( "$host $port $dbname $credentials" );

if(!$db){

echo "Error : Unable to open database\n";

}

$ward='select st\_astext(st\_transform(st\_union(st\_setsrid(geom,32644)),3857)) from wardboundary';

$safe='with wardboundaryUnion as (select st\_union(geom) as geom from wardboundary),

hospital\_buffer1 as (select st\_union(st\_buffer(geom,500.0))as geom from hospital),

safezone as (select st\_intersection(w.geom,h.geom) as geom from wardboundaryUnion w, hospital\_buffer1 h)

select st\_astext(st\_transform(st\_union(st\_setsrid(geom,32644)),3857)) from safezone;';

$manageable='with wardboundaryUnion as (select st\_union(geom) as geom from wardboundary),

hospital\_buffer1 as (select st\_union(st\_buffer(geom,500.0))as geom from hospital),

hospital\_buffer2 as (select st\_union(st\_buffer(geom,1500.0))as geom from hospital),

buff\_diff as (

select st\_difference(h2.geom,h1.geom) as geom from hospital\_buffer1 h1, hospital\_buffer2 h2),

managable as (

select st\_intersection(w.geom,b.geom) as geom from wardboundaryUnion w, buff\_diff b)

select st\_astext(st\_transform(st\_union(st\_setsrid(geom,32644)),3857)) from managable;';

$unsafe='with wardboundaryUnion as (select st\_union(geom) as geom from wardboundary),

hospital\_buffer2 as (select st\_union(st\_buffer(geom,1500.0))as geom from hospital),

unsafe as (

select st\_difference(w.geom,h2.geom) as geom from wardboundaryUnion w, hospital\_buffer2 h2)

select st\_astext(st\_transform(st\_union(st\_setsrid(geom,32644)),3857)) from unsafe;';

$geolocation='with wardboundaryUnion as (select st\_union(geom) as geom from wardboundary),

hospital\_buffer2 as (select st\_union(st\_buffer(geom,1500.0))as geom from hospital),

unsafe as (

select st\_difference(w.geom,h2.geom) as geom from wardboundaryUnion w, hospital\_buffer2 h2),

road\_buffer as (select st\_buffer(st\_union(geom),500.0) as geom from road),

geo as (select st\_intersection(r.geom,s.geom)as geom from unsafe s, road\_buffer r)

select st\_astext(st\_transform(st\_union(st\_setsrid(geom,32644)),3857)) from geo;';

$road='select st\_astext(st\_union(st\_transform(st\_setsrid(geom,32644),3857))) from road';

$hosp='select st\_astext(st\_union(st\_transform(st\_setsrid(geom,32644),3857))) from hospital';

$wret = pg\_query($db, $ward);

$rret=pg\_query($db, $road);

$hret=pg\_query($db, $hosp);

$sret=pg\_query($db, $safe);

$mret=pg\_query($db, $manageable);

$uret=pg\_query($db, $unsafe);

$gret=pg\_query($db, $geolocation);

if(!$wret||!$rret||!$hret||!$sret||!$mret||!$uret||!$gret){

echo pg\_last\_error($db);

exit;

}

$wout=pg\_fetch\_row($wret);

$rout=pg\_fetch\_row($rret);

$hout=pg\_fetch\_row($hret);

$sout=pg\_fetch\_row($sret);

$mout=pg\_fetch\_row($mret);

$uout=pg\_fetch\_row($uret);

$gout=pg\_fetch\_row($gret);

?>

<script src="OpenStreetMap.js"/>

</script>

<script type="text/javascript">

var map;

function init() {

var fromProjection = new OpenLayers.Projection("EPSG:4326"); // Transform from WGS 1984

var toProjection = new OpenLayers.Projection("EPSG:3857"); // to Spherical Mercator Projection

map = new OpenLayers.Map("demoMap",{projection:"EPSG:3857"});

osmLayer = new OpenLayers.Layer.OSM();

map.events.register("click", map, function(e) {

var position = map.getLonLatFromPixel(e.xy);

lonlat1=new OpenLayers.LonLat(position.lon,position.lat).transform(toProjection,fromProjection);

var popup = new OpenLayers.Popup.FramedCloud("popup",

new OpenLayers.LonLat(position.lon,position.lat),null,

"<div style='color:purple'>You have clicked at <br> LONGITUDE:" +lonlat1.lon+" <br> LATITUDE: "+lonlat1.lat+"</div>",

null, true);

map.addPopup(popup);

});

var WKT\_ward= "<?php echo $wout[0] ; ?>";

var WKT\_hosp= "<?php echo $hout[0] ; ?>";

var WKT\_road= "<?php echo $rout[0] ; ?>";

var WKT\_safe= "<?php echo $sout[0] ; ?>";

var WKT\_manage= "<?php echo $mout[0] ; ?>";

var WKT\_unsafe= "<?php echo $uout[0] ; ?>";

var WKT\_geo= "<?php echo $gout[0] ; ?>";

var wkt\_format = new OpenLayers.Format.WKT();

raj = new OpenLayers.Layer.Vector('Rajahmundry',{});

hosp = new OpenLayers.Layer.Vector('Hospitals',{});

road = new OpenLayers.Layer.Vector('Road network',{});

safe = new OpenLayers.Layer.Vector('Safe zone',{});

manage = new OpenLayers.Layer.Vector('Manageable zone',{});

unsafe = new OpenLayers.Layer.Vector('Unsafe zone',{});

geo = new OpenLayers.Layer.Vector('Proposed geolocations',{});

var rajFeature = wkt\_format.read(WKT\_ward);

var hospFeature = wkt\_format.read(WKT\_hosp);

var roadFeature = wkt\_format.read(WKT\_road);

var safeFeature = wkt\_format.read(WKT\_safe);

var manageFeature = wkt\_format.read(WKT\_manage);

var unsafeFeature = wkt\_format.read(WKT\_unsafe);

var geoFeature = wkt\_format.read(WKT\_geo);

raj.addFeatures(rajFeature);

hosp.addFeatures(hospFeature);

road.addFeatures(roadFeature);

safe.addFeatures(safeFeature);

manage.addFeatures(manageFeature);

unsafe.addFeatures(unsafeFeature);

geo.addFeatures(geoFeature);

map.addControl(new OpenLayers.Control.LayerSwitcher());

map.addLayers([osmLayer, raj, hosp, road, safe, manage, unsafe, geo]);

map.zoomToExtent(raj.getDataExtent());

}

</script>

<body onload="init();">

<center><div id="demoMap" style='width: 750px; height: 500px;' /></center>

<p align=justify style="color:lightyellow;"><font size="4">This project helps in finding the geographical locations where there is a need of health care service centers. This web page displays the map which provides information about the location of hospitals and road network in Rajahmundry. It shows the areas which are safe, manageable, unsafe in accessing the health care service centers. It also displays the research results i.e., the geographical location where new health care service centers needs to be established.</font></p></body>

</body>

</html>

1. **Testing**

**Test case1:**

Viewing the Rajahmundry region on the OpenStreetMap.

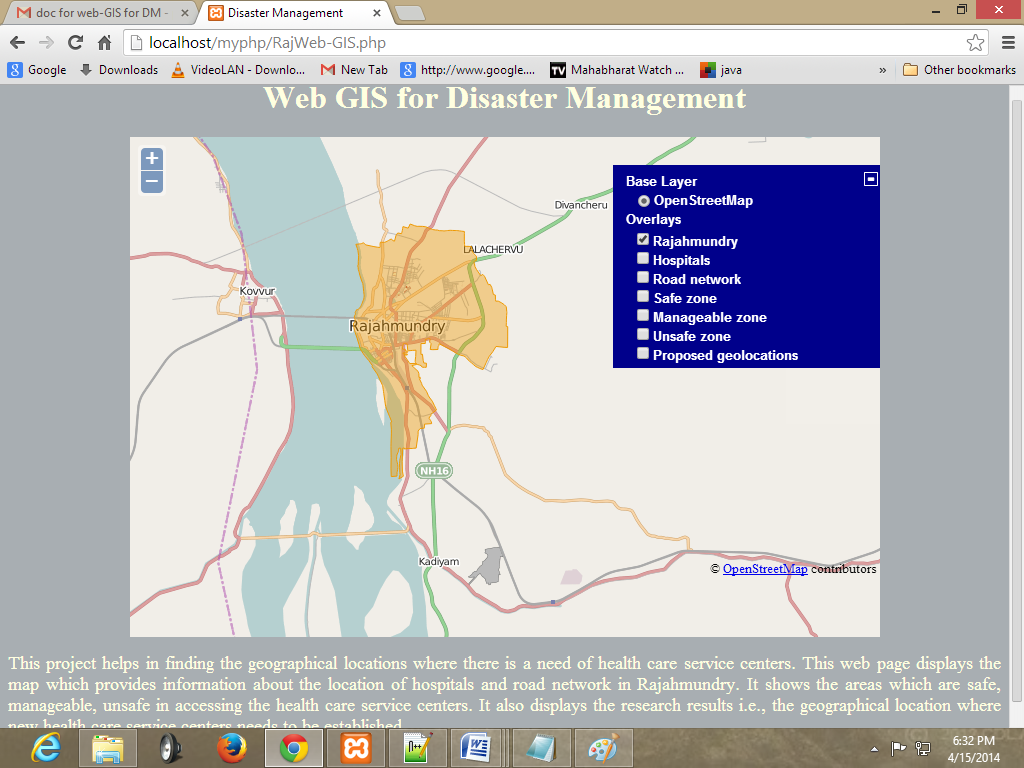


Figure 7.1 Rajahmundry region on the map

**Test case 2**

Testing the Longitude and Latitude module.

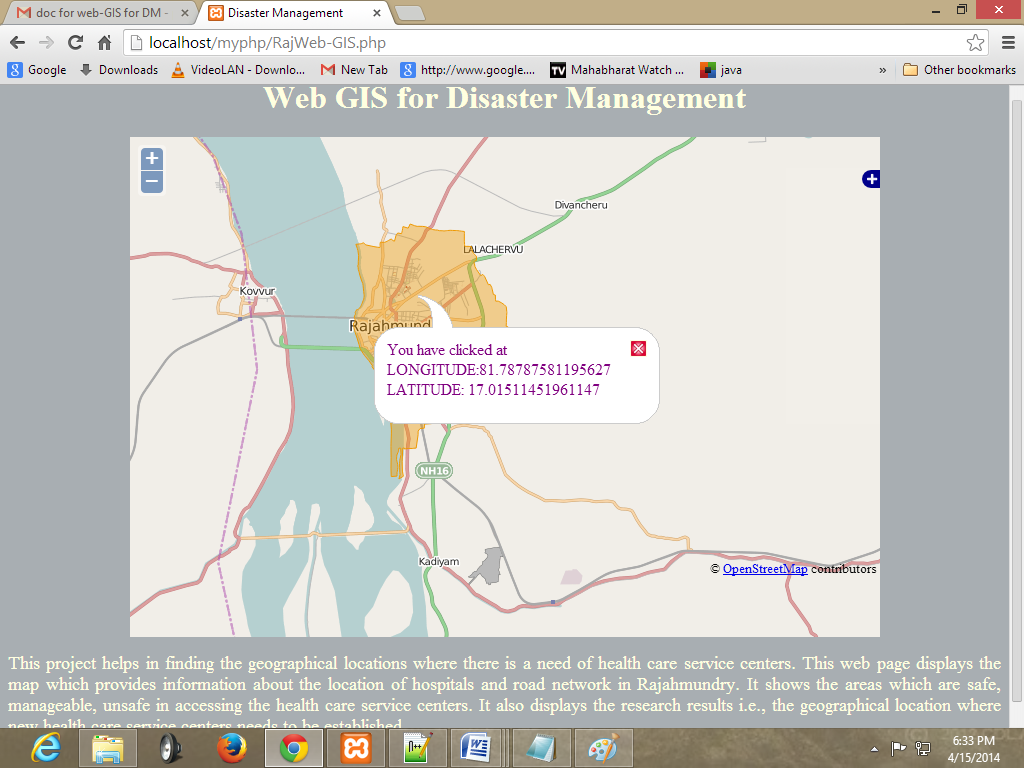


Figure 7.2 Longitude and Latitude check up.

1. **Output Screens**

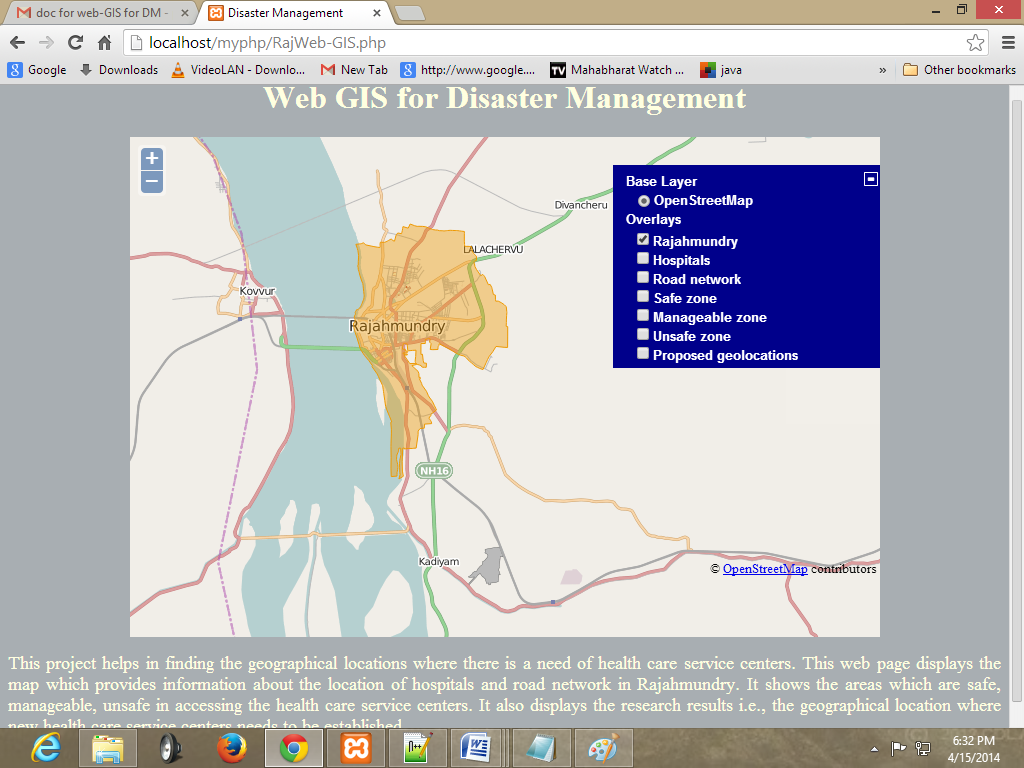
****

Figure 8.1 Showing Rajahmundry in the map

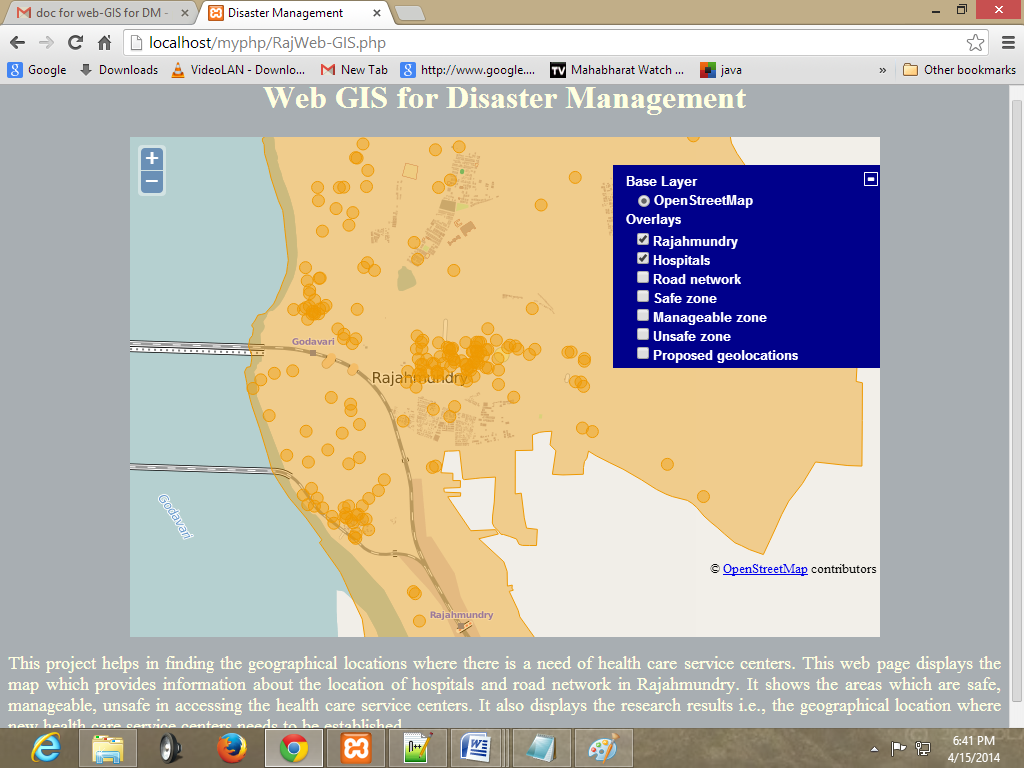
****

Figure 8.2 Showing hospital locations in Rajahmundry

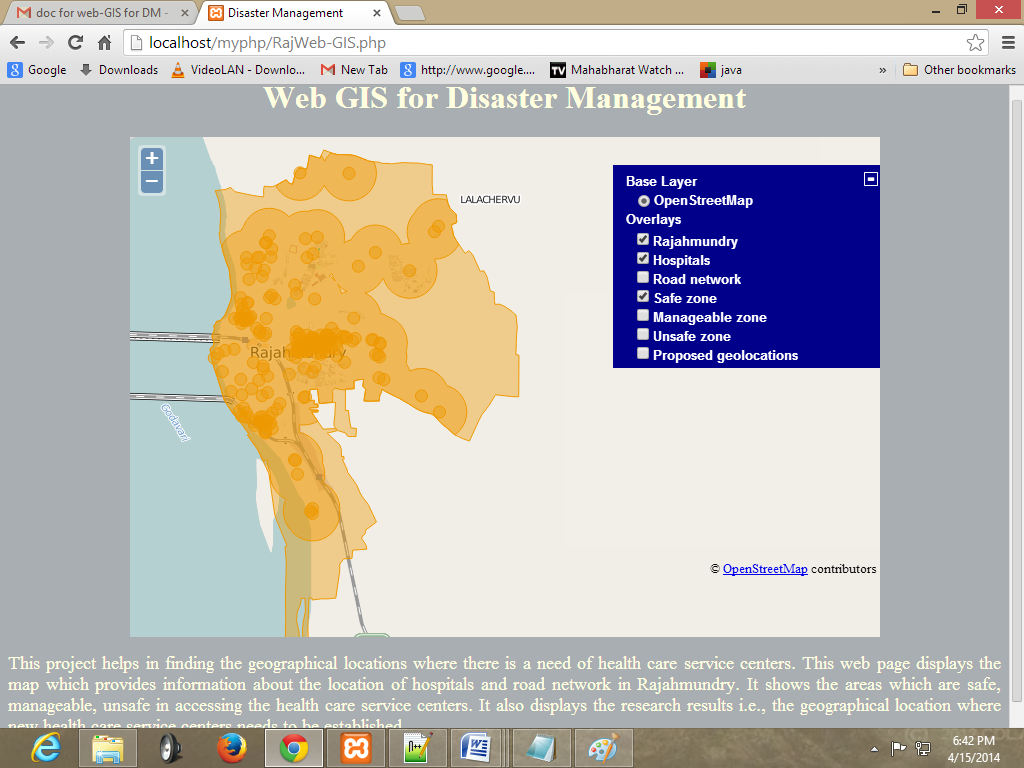
****

Figure 8.3 Showing the safe zone in Rajahmundry

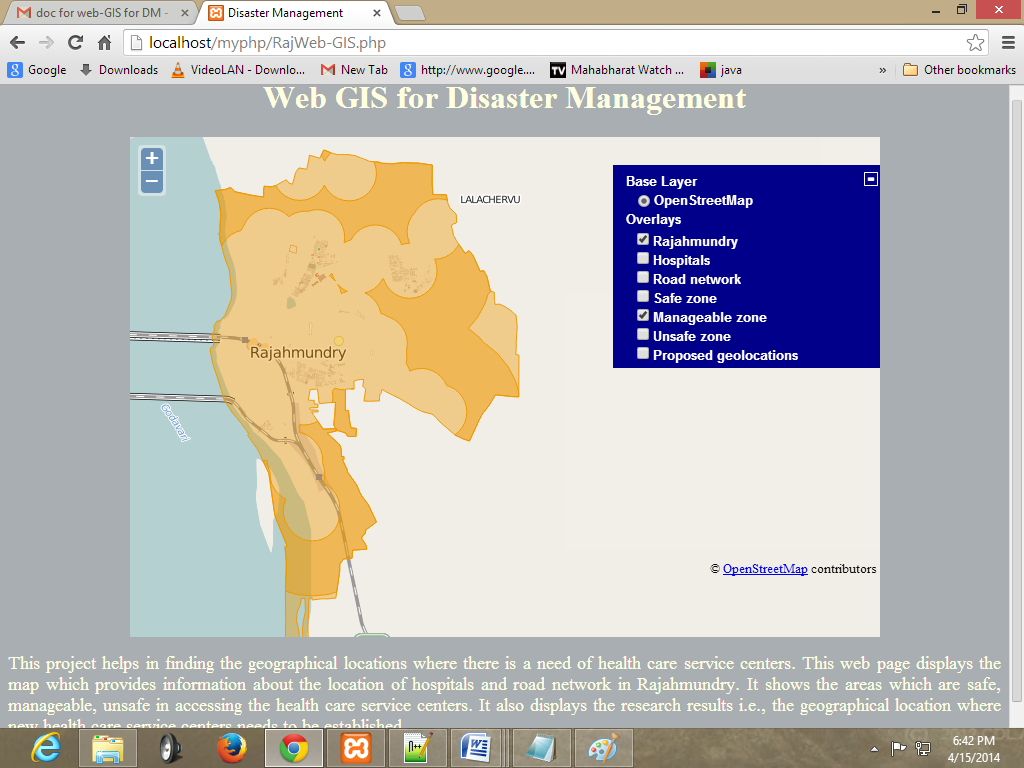
****

Figure 8.4 showing the Manageable zone

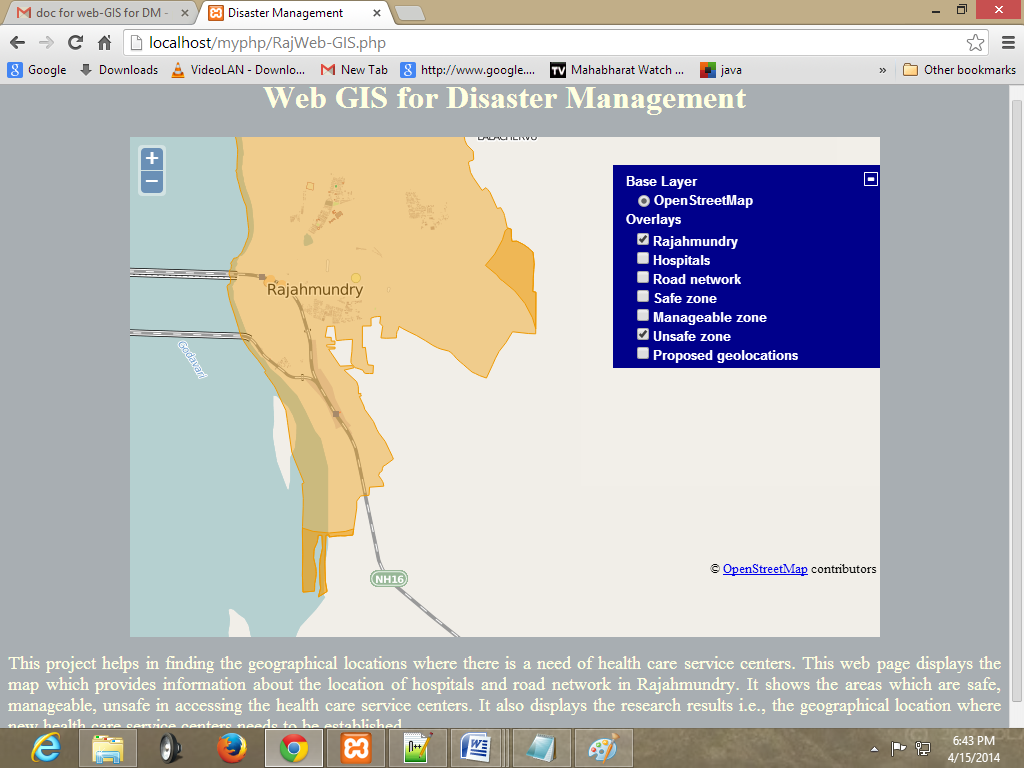
****

Figure 8.5 showing the unsafe zone in Rajahmundry

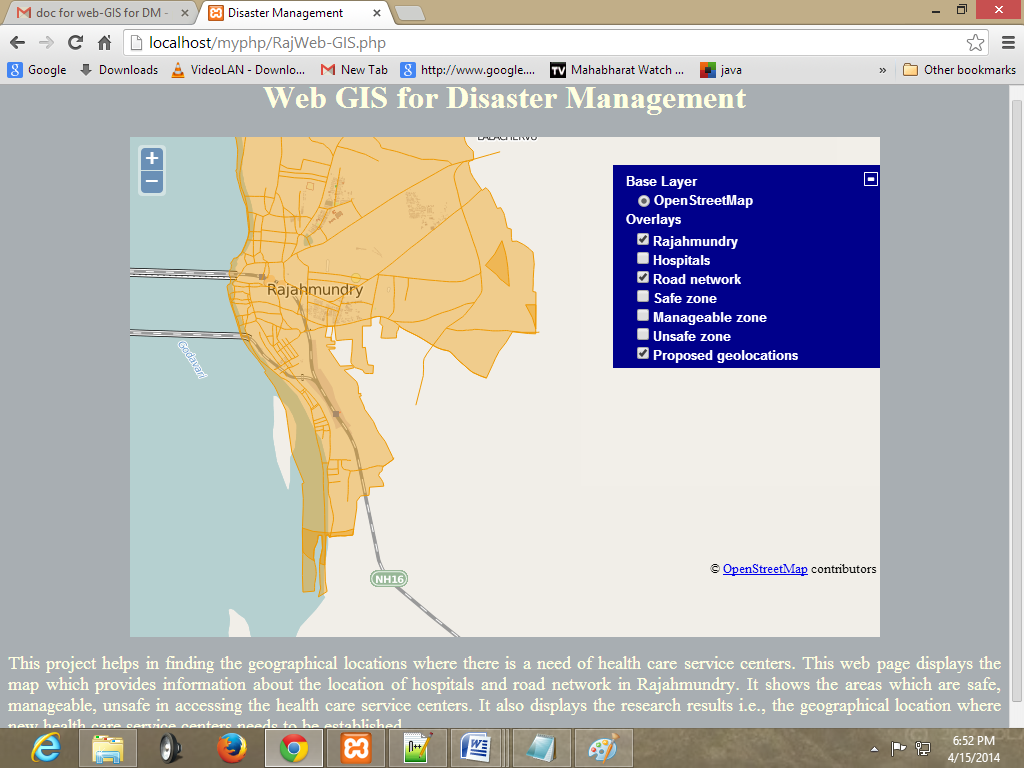
****

Figure 8.6 Showing the proposed geo locations nearer to the road.

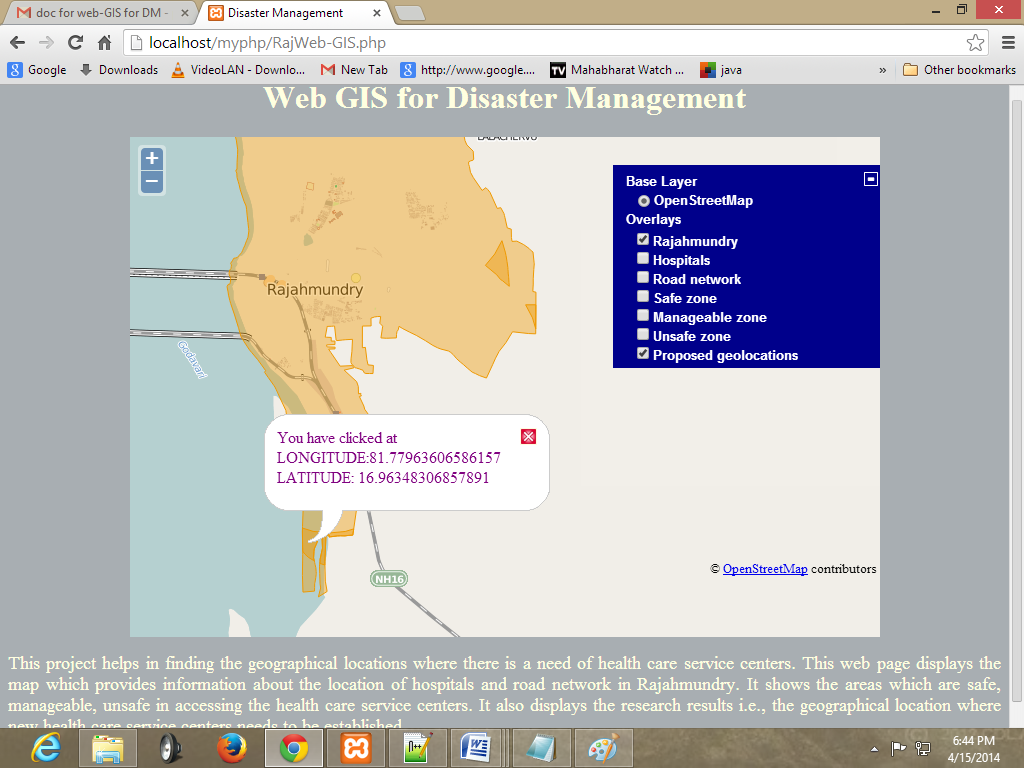
****

Figure 8.7 Showing the longitude and latitude of the proposed geo locations

1. **Conclusions**

**Web-GIS for Disaster Management** has been implemented successfully. Functionality of the modules is tested. Validity of the results is also been tested.

* To implement this project the technologies learnt are OpenLayers, QGIS, PostgreSQL/PostGIS.
* Scripting languages used are PHP and JavaScript.
* The web server used is XAMPP.

1. **Further Enhancements**

This approach to the solution is not only applicable for Rajahmundry data for finding geo locations for establishing new emergency service centers but also for any such kind of analysis where the problem is same.

This project shall be enhanced in future by adding modules such as ‘Finding shortest path from emergency services to incident occurring place’, considering fire stations, providing a user friendly interface for emergency service center personnel to use it with ease.

1. **References/Bibliography**

1. [www.openlayers.org](about:blank)
2. [www.tutorialspoint.com](about:blank)
3. workshops.boundlessgeo.com
4. [www.w3schools.com](about:blank)
5. live.osgeo.org
6. [www.openstreetmap.org](about:blank)
7. osm-wms.de