**PROJECT REPORT**

**ONLINE ASSESSMENT RESTful WEB APPLICATION USING JWT BASED AUTHENTICATION**

Submitted in partial fulfilment of the requirements for

the award of the degree of

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**OF**

**SASTRA UNIVERSITY**

**Submitted by**

**PENUKONDA SAI NAVYA SREE 117003142**



**Under the Guidance of**

**Prof. KAMAKSHI .S**

**ASSOCIATE PROFESSOR**

**COMPUTER SCIENCE DEPARTMENT**

**SASTRA UNIVERSITY, THANJAVUR**

**SCHOOL OF COMPUTING**

**SHANMUGHA**

**ARTS, SCIENCE, TECHNOLOGY & RESEARCH ACADEMY**

**(A University Established under section 3 of the UGC Act, 1956)**

**TIRUMALAISAMUDRAMTHANJAVUR – 613 401**

**April 2017**

**SCHOOL OF COMPUTING**

**SHANMUGHA**

**ARTS, SCIENCE, TECHNOLOGY & RESEARCH ACADEMY**

**(A University Established under section 3 of the UGC Act, 1956)**

**TIRUMALAISAMUDRAM, THANJAVUR – 613401**



**BONAFIDE CERTIFICATE**

Certified that this project work entitled “**ONLINE ASSESSMENT RESTful WEB APPLICATION USING JWT BASED AUTHENTICATION**” submitted to the Shanmugha Arts, Science, Technology & Research Academy (SASTRA University), Tirumalaisamudram-613401 by **PENUKONDA SAI NAVYA SREE** with reg.no:**117003142** in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** is the original and independent work carried out under my guidance, during the period December 2016 - April 2017**.**

|  |  |
| --- | --- |
| **INTERNAL GUIDE** | **ASSOCIATE DEAN** |
| **Prof. KAMAKSHI.S** | **Dr. A. UMAMAKESWARI** |
| **SCHOOL OF COMPUTING** | **SCHOOL OF COMPUTING** |

Submitted for University Examination held on\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**EXAMINER - I** **EXAMINER - II**

**SCHOOL OF COMPUTING**

**SHANMUGHA**

**ARTS, SCIENCE, TECHNOLOGY & RESEARCH ACADEMY**

**(A University Established under section 3 of the UGC Act, 1956)**

**TIRUMALAISAMUDRAM, THANJAVUR – 613401**



**DECLARATION**

We submit this project work entitled “**ONLINE ASSESSMENT RESTful WEB APPLICATION USING JWT BASED AUTHENTICATION**” to the Shanmugha Arts, Science, Technology & Research Academy (SASTRA) University, Tirumalaisamudram–613 401, in partial fulfilment of the requirements for the award of the degree of **BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING** and declare that it is our original and independent work carried out under the guidance of **ANUJA RANI,SASIDHARAN**, Learning & Development Team, iNautix Technologies, Chennai.

|  |  |  |
| --- | --- | --- |
| **Place :** | **Name:** P. Sai Navya Sree | Signature: |
| **Date:** | **Reg.no:**117003142 |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**ACKNOWLEDGEMENT**

First and Foremost, we take pride in thanking the almighty to gave us strength for the successful completion of this project.

We have immense pleasure in expressing my heartfelt thanks to our Vice-Chancellor **prof.R.Sethuraman** for the benevolent advice and guidance during my tenure in the college.

We would like to express our gratitude to **Dr.Balachandran, Registrar,**SASTRA UNIVERSITY, for his benevolent guidance through-out my college life.

We wish to express our thanks to **Dr.S.Vaidhyasubramanian, Dean of Planning and Development** for his encouragement and providing us with all the needed amenities.

We would like to express our gratitude to **Dr.P.Swaminathan,** Dean **,** School of Computing,SASTRA UNIVERSITY, for his benevolent guidance through-out my college life.

We would like to express our gratitude to **Dr.A.Umamakeshwari,** Associate Dean **,** Computer Science and Engineering,School of Computing, SASTRA UNIVERSITY, for her benevolent guidance through-out my college life.

We wish to express our thanks to **Prof. Kamakshi . S,** Associate ProfessorComputer Science and Engineering,School of Computing, SASTRA UNIVERSITYfor her encouragement and providing us with all the needed amenities.

we deeply thank our family and friends for supporting me in all the tasks that I have carried for the successful completion of this project.

**TABLE OF FIGURES USED IN THIS PROJECT:**

|  |  |  |  |
| --- | --- | --- | --- |
| S.NO: | FIG.NO | NAME OF THE FIGURE: | PAGE NUMBER: |
| 1 | FIG 4.1 | Block Diagram for LandMine Detection Robot controlled by Android |  |
| 2 | FIG 4.2 | GPIO Pin Diagram for Raspberry pi controlled by Android |  |
| 3 | FIG 4.3 | Block Diagram of L293D |  |
| 4 | FIG 5.1 | Usecase Diagram for Robot controlled by Android application |  |
| 5 | FIG 5.2(1) | Activity Diagram for LandMine Detection Robot |  |
| 6 | FIG 5.2(2) | Activity Diagram for server driven model |  |
| 7 | FIG 5.3 | Sequence Diagram for the sensor detection |  |
| 8 | FIG 5.4 | Class Diagram for the developed autonomous robot |  |
| 9 | FIG 5.5 | Collaboration Diagram for detecting landmine |  |
| 10 | FIG 7.1 | Output Screen of Android Applicatin controlling Robot |  |
| 11 | FIG 7.2 | Output Screen of Google Maps android application which retrieves path from source to destination |  |

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Title** | **Page No.** | |
|  | |
| **ᵢ** | **Abstract** |  |
|  |  |
| **ᵢᵢ** | **Introduction** |  | |
|  |  |  | |
| **ᵢᵢᵢ** | **Problem Statement** |  | |
|  |  |  | |
| **ᵢᵥ** | **Literature Survey** |  | |
|  |  |  | |
| **ᵥ** | **Software, Hardware Requirement Specification** |  | |
|  |  |  | |
| **ᵥᵢ** | **Conceptual Model / Proposed Architecture** |  | |
|  |  |  | |
| **ᵥᵢᵢ** | **Interaction Scenario** |  | |
|  |  |  | |
| **ᵥᵢᵢᵢ** | **Methodology and Approach** |  | |
|  |  |  | |
| **ᵢᵪ** | **Output/Results** |  | |
|  |  |  | |
| ᵪ | **Conclusion** |  | |
|  |  |  | |
| ᵪᵢ | **References** |  | |
|  |  |  | |

**1)ABSTRACT**

This project is aimed to develop a web based assessment application using Rest API web Services. Various Assessment applications are currently being used in the process of recruiting the employees for an organization and they are paying a huge amount for that to the Third party vendor. Assessment Application is a web service based one, which is having further more features than the third party vendor applications, and also in which it supports customization based on the organization’s present needs which will be beneficial to the organization in long run.

Online Assessment Application will be dedicated to a particular organization before actually being used. The website has an administrator who controls the functioning of the system. The interface is designed in such a way that it enables candidates and organization to have efficient and user-friendly interaction by using latest technologies like AngularJS which makes UI high responsive. [pending jwt]

RESTful API allows manipulation of objects through URL which acts as the identifier of every resource which makes easy access of application. A WAR file is created and deployed in Tomcat Server.

The challenging part after developing the application is to move into other platform which helps the end user to use other similar applications signed in smoothly.

**2)INTRODUCTION**

**OVERVIEW:**

Assessment Application is an application used for the purpose of conducting assessments which helps in understanding the knowledge gained by the users using it. It will be much more useful in the recruitment process of any organization and the specialty of using it is customization based on the present needs of the organization and it leads to the productive development of the organization. The organization’s objectives can be achieved and much more efficiency will lead to higher production and much beneficial to the organization.

**OBJECTIVES:**

To develop an Assessment Application which provides

* Organisation based customization.
* Low cost.
* Upgradability with upcoming database technologies.
* Lightweight processing.
* Helps in easy recruitment and organization development process.
* Concurrency.
* Loose Coupling by implementing Dependency Injection using Spring MVC Architecture.
* Secure transformation of information between parties through JWT based authentication.
* Create strong and secure database that allow for any connection in a secret way, to prevent any outside or inside attacks.
* High availability or disaster recovery.
* Storage and compression.
* Ease of integration with external libraries.
* Pricing.
* Integration with reporting and archive tools.
* Backup option.

**3)PROBLEM STATEMENT:**

Most particularly it is very difficult to detect land mines without any risk and casualities. Even though Indian government found some machines and technologies to detect landmines but they are not completely autonomous. In our project we are designing GPS GUIDED ROBOT DRIVEN BY ANDROID APPLICATION to solve this problem.

In order to eliminate the error in GPS system, we will use the mobile in built positioning feature for the calculation of the location of the robot.If we add the dedicated hardware for the GPS then it needs to process the positioning and it should send the coordinates back to the processor.But in our proposed system we will be using the mobile application, the main advantage of this mobile application is that it will not send the positioning data to the processor back, it will process the positioning data and it wil send the refined and the processing data to the raspberry pi. That means, the mobile will process the data retrieved from the check points of the map written in android application and the mobile processor itself will process the data.The only data that is sending to the raspberry pi is about the next check point that is going to be parsed by the mobile application

In the LAN connection, both the raspberry pi and the mobile are connected to the same network,thus the data sending will not be delayed due to the dedicated connection. it will process the positioning data and it will send the refined and the processing data to the raspberry pi.

**4)LITERATURE SURVEY:**

**EXISTING SYSTEM:**

The existing third party applications are designed in a fixed way so that the organizations needs to pay alone for their customized designing and any specialization that need to be wished by the organization should be done by the vendor providing it , any changes in the application must be carried out only with the help and support of the vendor providing the application.

**DRAWBACKS OF EXISTING SYSTEM**

* **PRICE:**

The costs of the third party vendor applications are cost wise higher and even though the organization gets the service, these organizations cannot be able to work independently on its own pace. Any modifications in those applications can be done only with the vendor support.

* **INABILITY TO EXPAND:**

There are several applications which uses SQL database and it has the inability to withstand when the number of records exceeds certain limits, the ability to retrieve gets slow and as it is web service based application the user may have a slower response as the database gets loaded.

* **MIGRATION TO OTHER DATABASE:**

There are so many third party applications which doesn’t supports No SQL databases, it cannot be further scaled up to a higher data contented database and it will not be useful for the organization in a long term use.

**PROPOSED SYSTEM:**

The proposed system is based on completely supportable to the organization in all cases. The proposed system supports No SQL database as it uses java based spring REST web services. The customization of organization based assessment can be done with the proposed system. The proposed system is lightweight and it doesn’t get slow over loading of data.

**FEATURES OF PROPOSED SYSTEM:**

* **Lightweight:**

Spring is lightweight in terms of size and transparency. The basic version of spring framework is around 2MB.

* **Inversion of control:**

Loose coupling is achieved in spring using the technique Inversion of

Control. It minimizes the amount of code in your application.

* **Dependency Injection:**

You do not create your objects but describe how they should be created. Do not directly connect your components and services but try to describe which services are needed by which components in a configuration file.

* **Container:**

Spring contains and manages the life cycle and configuration of

application objects.

**5)SOFTWARE/HARWARE REQUIREMENT SPECIFICATION**

**HARDWARE REQUIREMENT SPECIFICATION:**

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirement list is often accompanied by a Hardware Compatibility List (HCL), especially in case of operating systems. An HCL lists tested, compatible and sometimes incompatible hardware devices for a particular operating system or application.

* **PROCESSOR :** Intel Core 2 duo and Advance
* **SPEED :** 2.0 GHz
* **MEMORY :** 2 GB RAM and above
* **HARD DISK DRIVE :** 100 GB and above

**HARDWARE INTERFACES:**

**Server side hardware:**

* Hardware compatible and recommended by all the software used
* Communication hardware which is used to perform client requests.

**Client side hardware:**

* Hardware used by respective client’s side Operating system and web browser.
* Communication hardware to retrieve data from server.

**SOFTWARE INTERFACES:**

**Server side software:**

* Apache Tomcat Server v7.0
* Server side scripting tools : JSP
* Database tools : Derby 10.1
* Compatible Operating System :Windows

**Client side software:**

* Web browser supporting JavaScript

**COMMUNICATION PROTOCOL:**

**Server side :**

HTTP incoming request and HTTPS incoming request for secure gateway.

**Client side :**

HTTP outgoing request and HTTPS outgoing request for secure gateway.

**SOFTWARE REQUIREMENTS SPECIFICATION:**

**1)Apache Server Installation:**

Software part of the project mainly involves in connectivity between web application and Tomcat Server. Apache Tomcat is an open-source Web server and servlet container. It requires a Java Standard Edition Runtime Environment (JRE) version 6 or later.

**STEPS TO INSTALL:**

1. Download and install JRE from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>.
2. Download and install Apache Tomcat, a binary distribution of tomcat from <http://tomcat.apache.org/>
3. Unpack the binary distribution so that it resides in its own directory (conventionally named "apache-tomcat-[version]").
4. Configure Environment Variables
   * 1. Set CATALINA\_HOME and CATALINA\_BASE(optional).
     2. The CATALINA\_HOME environment variable should be set to the location of the root directory of the "binary" distribution of Tomcat.
     3. The CATALINA\_BASE environment variable specifies location of the root directory of the "active configuration" of Tomcat. It is optional. It defaults to be equal to CATALINA\_HOME.
5. Set JRE\_HOME or JAVA\_HOME.

**5.1.1** The JRE\_HOME variable is used to specify location of a JRE. The JAVA\_HOME variable is used to specify location of a JDK.

**5.1.2** Using JAVA\_HOME provides access to certain additional startup options that are not allowed when JRE\_HOME is used. If both JRE\_HOME and JAVA\_HOME are specified, JRE\_HOME is used. The best place to include these variables is a "setenv" script.

1. Other Variables like CATALINA\_OPTS are optional to set with. It allows specification of additional options for the java command to start tomcat.
2. Start Tomcat
   * 1. On Windows

%CATALINA\_HOME%\bin\startup.bat

or

%CATALINA\_HOME%\bin\catalina.bat start

or

$CATALINA\_HOME/bin/catalina.sh start

**7.1.2**  After startup, the default web applications included with Tomcat will be available by visiting:

<http://localhost:8080/>

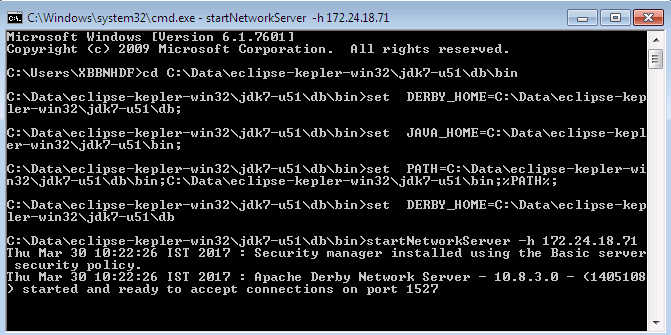
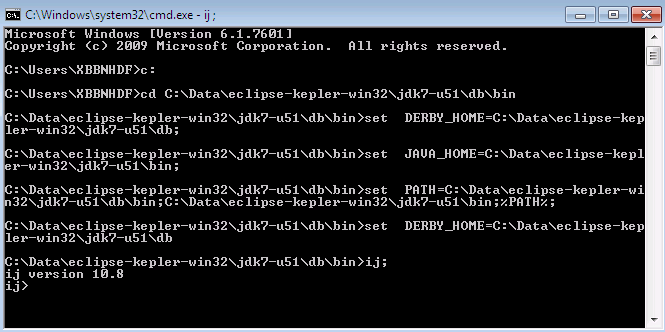
1. Shut Down Tomcat

**2)DERBY DATABASE INSTALLATION:**

Apache Derby is a relational database management system (RDBMS) developed by Apache Software Foundation which is embedded in Java programs and used for online transaction processing. It has a 2.6 MB disk-space footprint.

The Derby network server increases the reach of database engine

by providing basic client server functionality. The server allows clients to connect over TCP/IP using the standard DRDA protocol. Derby comes along with Java 7 and has been named as "JavaDB" but it is exactly

the same bit-for-bit as Derby is. For developers with Java 6, they can still download Derby as before, but for developers using JRE 7 or later, Derby is included in the Java API.

system("gpio -g write 10 0");

see when we run the application and Java is the backend which processes all actions based on the inputs from xml.

In the application which is developed , there are four buttons start,stop,right,left, in xml which sends a unique request to Apache server running on PI. Onclicking each button a Java code module is executed, the following Java code is executed when we press back button and it establishes connection between Android and Apache server. Similarly, for each button a Java function is executed.

**Java Code :**

private void insertToDatabase1(String name, String add){

class SendPostReqAsyncTask extends AsyncTask<String, Void, String>

{

@Override

protected String doInBackground(String... params) {

String paramUsername = params[0];

String paramAddress = params[1];

try {

HttpClient httpClient = new DefaultHttpClient();

HttpPost httpPost = new HttpPost(

"http://192.168.81.103/index1.php?username=back");

HttpResponse response = httpClient.execute(httpPost);

HttpEntity entity = response.getEntity();

} catch (ClientProtocolException e) {

} catch (IOException e) {

}

return "success";

}

@Override

protected void onPostExecute(String result) {

super.onPostExecute(result);

Toast.makeText(getApplicationContext(), result, Toast.LENGTH\_LONG).show();

TextView textViewResult = (TextView) findViewById(R.id.textViewResult);

textViewResult.setText("Back");

}

}

SendPostReqAsyncTask sendPostReqAsyncTask = new SendPostReqAsyncTask();

sendPostReqAsyncTask.execute(name, add);

}

In the above Java code module, insertToDatabase1 method is called on clicking back button.In this, there is SendPostReqAsyncTask class which extends AsyncTask class which is used to communicate with server. It uses HttpPost to post the request to server and receives HttpResponse and if it is successfully posted to a toast is prompted indicating that request is succeddfully posted.

**3)Path retrieval between source and destination using Google Polyline concept**

An android mobile application for path retrieval between source and destination developed based on Google Polyline concept. Android Studio IDE was utilized to develop this android mobile application. Android Studio is also an IDE like eclipse which is used to develop android applications easily.

Polylines were added in Google’s local maps which helps to retrieve path locally. To access Google maps and polylines in android mobile application following dependency is to be added in build.gradle file

**compile 'com.google.android.gms:playservices:8.4.0'**

After adding thisdependency, Google maps can be accesed. Google play services must be installed in SDK manager to access Google’s services. User permissions are to be added in manifest file to access Internet and GPS location services in end-user phone. Following permissions are to be added in manifest file

**<uses-permission android:name="com.karthik.mybot.permission.MAPS\_RECEIVE" />**

**<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />**

**<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />**

**<uses-permission android:name="com.google.android.providers.gsf.permission.READ\_GSERVICES" />**

**<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" />**

**<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />**

After adding above permissions to manifest file, user can access various services like GPS on phone. In front-end layout a GoogleMaps fragment is used on which we point source and destination. The path between the source and destination is acquired through Google Polyline from Java code for path retrieval.

**Java Code**

public List<List<HashMap<String,String>>> **parse**(JSONObject jObject){

List<List<HashMap<String, String>>> routes = new ArrayList<List<HashMap<String,String>>>() ;

JSONArray jRoutes = null;

JSONArray jLegs = null;

JSONArray jSteps = null;

try {

jRoutes = jObject.getJSONArray("routes");

for(int i=0;i<jRoutes.length();i++){

jLegs = ( (JSONObject)jRoutes.get(i)).getJSONArray("legs");

List path = new ArrayList<HashMap<String, String>>();

for(int j=0;j<jLegs.length();j++){

jSteps = ( (JSONObject)jLegs.get(j)).getJSONArray("steps");

for(int k=0;k<jSteps.length();k++){

String polyline = "";

polyline = (String)((JSONObject)((JSONObject)jSteps.get(k)).get("polyline")).get("points");

List<LatLng> list = decodePoly(polyline);

for(int l=0;l<list.size();l++){

HashMap<String, String> hm = new HashMap<String, String>();

hm.put("lat", Double.toString(((LatLng)list.get(l)).latitude) );

hm.put("lng", Double.toString(((LatLng)list.get(l)).longitude) );

path.add(hm);

}

}

routes.add(path);

}

}

} catch (JSONException e) {

e.printStackTrace();

}catch (Exception e){

}

return routes;

}

private List<LatLng> **decodePoly**(String encoded) {

List<LatLng> poly = new ArrayList<LatLng>();

int index = 0, len = encoded.length();

int lat = 0, lng = 0;

while (index < len) {

int b, shift = 0, result = 0;

do {

b = encoded.charAt(index++) - 63;

result |= (b & 0x1f) << shift;

shift += 5;

} while (b >= 0x20);

int dlat = ((result & 1) != 0 ? ~(result >> 1) : (result >> 1));

lat += dlat;

shift = 0;

result = 0;

do {

b = encoded.charAt(index++) - 63;

result |= (b & 0x1f) << shift;

shift += 5;

} while (b >= 0x20);

int dlng = ((result & 1) != 0 ? ~(result >> 1) : (result >> 1));

lng += dlng;

LatLng p = new LatLng((((double) lat / 1E5)),

(((double) lng / 1E5)));

poly.add(p);

}

return poly;

}

In the above Java Module there is a method Parse which is called from main activity and it returns a route object which is list of list of hashmaps. This Parse method indeed utilizes decode method to retrieve list of latitudes and longitudes .Decode method returns a poly object which is list of latitudes and longitudes. Using these two methods, path between source and destination is retrieved.

**6) Conceptual Model / Proposed Architecture**

**BLOCK DIAGRAM:**

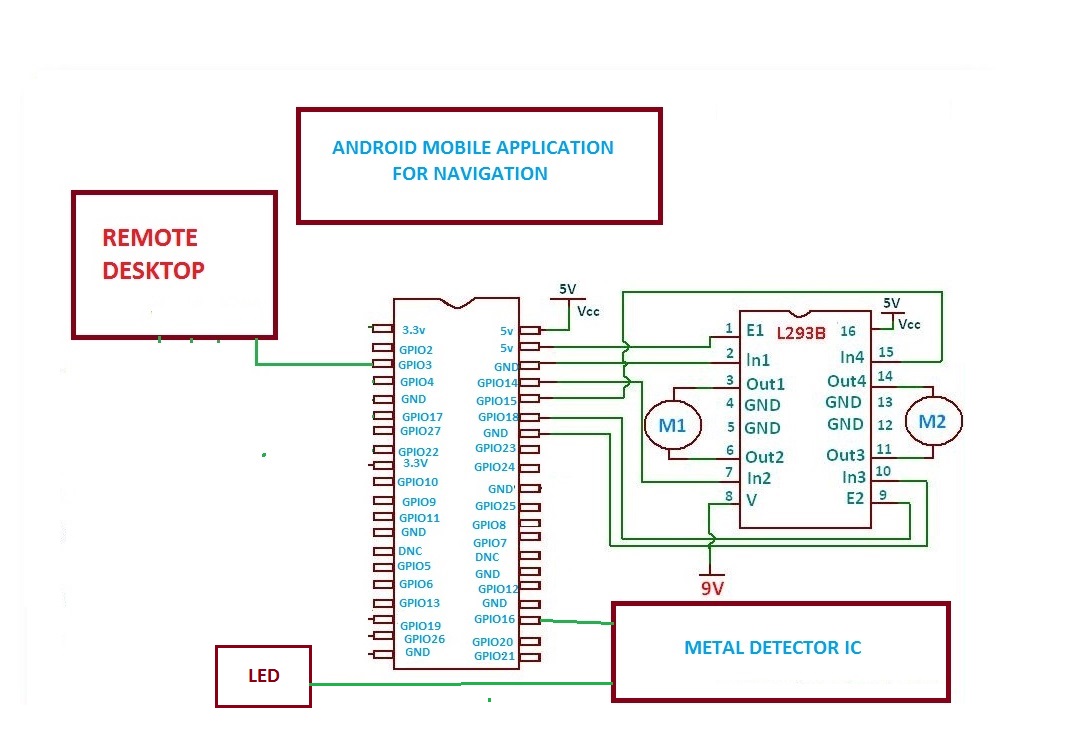
****

Fig 4.1: Block Diagram for LandMine Detection Robot controlled by Android

This figure represents the GPIO pins of the Raspberry pi, and the flow of the entire process that is going to taken part in our project.The pins will be given input from the mobile application then the GPIO pins wil respond accordingly and the motor drivers will be given the high signal, they will in turn move the motors and there by tyres.

**GPIO PINS FOR RASPBERRY PIE:**

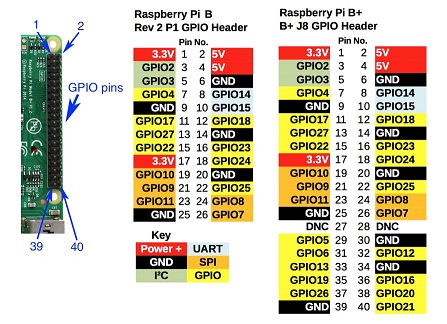


Fig 4.2:GPIO Pin Diagram for LandMine Detection Robot controlled by Android

This diagram shows the GPIO pins and the different pins that are embedded in the raspberry pi, mainly there are common and special keys, common keys are GPIO pins,ground and VCC of 5v and 3.3v. Special pins contains the power+,UART,GND,SPI,PC and also some kinds of GPIO.

**FUCTIONAL BLOCK DIAGRAM OF L293D:**

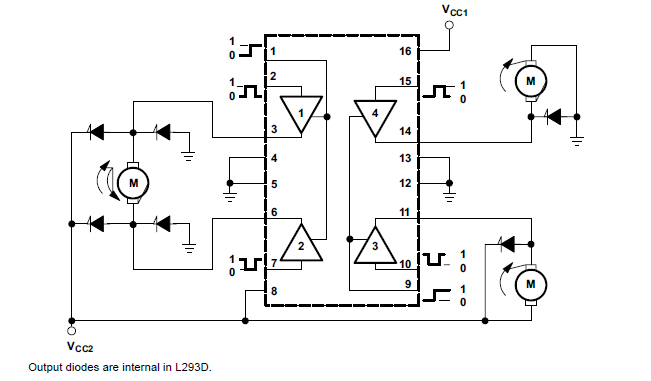


Fig 4.3: Block Diagram of L293D

The L293x have TTL-compatible readings and higher potential difference outputs for inductive load driving.

This is the functional diagram of the L293D block diagram that is used in this project.This figure shows us the different Gates that are used in the scratch of the integrated circuit of the L293D

**7) Interaction Scenario:**

**USECASE DIAGRAM:**

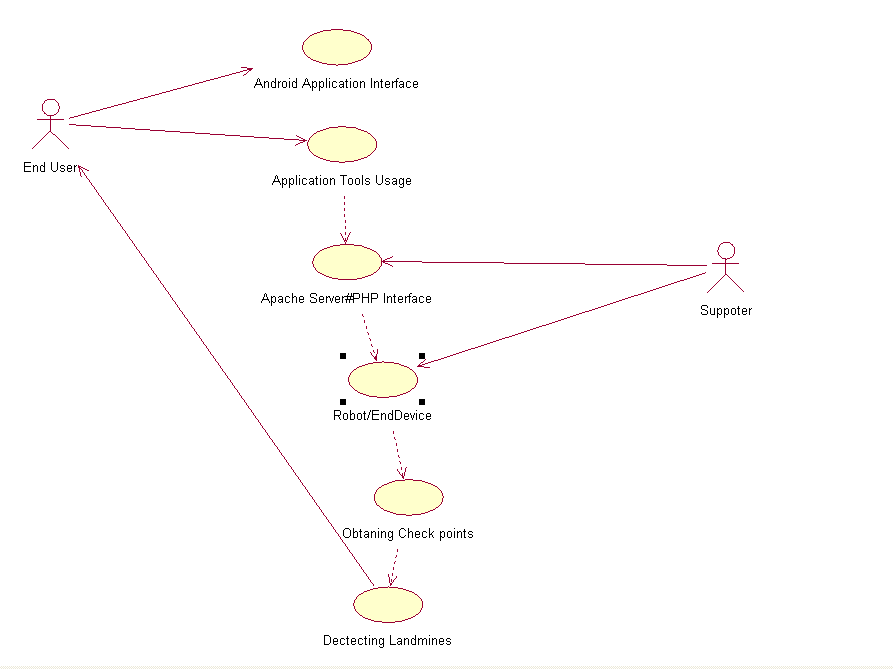


Fig 5.1: Usecase Diagram for LandMine Detection Robot controlled by Android

In this Usecase diagram, end-user is the actor who is interacting with the robot. This end-user interacts with the robot through android mobile application which acts as an interface to apache server php program.End-user gives input to the android application and this input is processed through php program in Apache Server. In Apache Server, as shown in Fig 5.1 several process flows occur and response is delivered to end-user.

After processing this data in PHP server,these inputs will be passed to raspberry pi processor, this in turn connects to L293D which drives the motor and moves along the path tracing all the check points from source to destination to detect landmines.

Usecase Diagram (Fig:5.1)Process flow:

1.End User to Application Interface

2.End User uses Application Tools

3.PHP server processes this inputs from Application Tool.

4.Processed Data to Raspberry pi.

5.Robot moves by Obtaining check points continuously.

6.Finally robot will detect landmines.

**ACTIVITY DIAGRAM:**

**ACTIVITY DIAGRAM FOR APPLICATION TO APACHE SERVER**

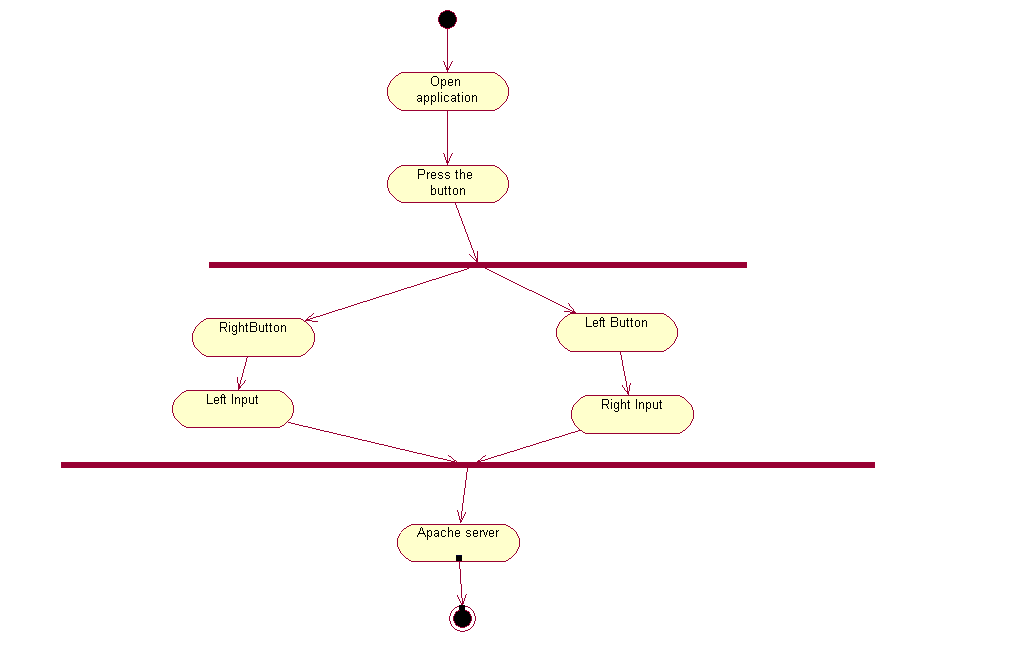


Fig 5.2(i): Activity Diagram for LandMine Detection Robot controlled by Android

In this Activity diagram, basic flow of process through which connection between android mobile application and server is explained. Android mobile application acts as an interface to apache server php program.End-user gives input to the android application and this input is processed through php program in Apache Server. In Apache Server, as shown in Fig 5.2(i) several process flows occur and response is delivered to end-user.

After processing this data in PHP server,these inputs will be passed to raspberry pi processor, this in turn connects to L293D which drives the motor and moves along the path tracing all the check points from source to destination to detect landmines.

Activity Diagram(Fig 5.2(i)) Process flow:

1.End-User to Application.

2.Pressing Particular Button.

3.If right button is pressed,robot moves in right direction.

4.If left button is pressed, robot moves in left direction.

5.If up button is pressed, robot moves in forward direction.

6.If down button is pressed, robot moves in downward direction.

**ACTIVITY DIAGRAM FOR APACHE SERVER TO ROBOT ACTUATORS**

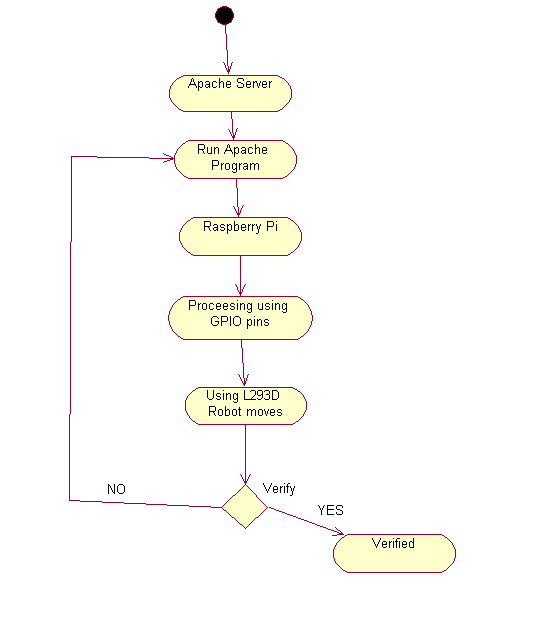
****

Fig 5.2(ii): Activity Diagram for LandMine Detection Robot controlled by Android

In this Activity diagram, basic flow of process through which connection between Apache Server and Robot is explained. The PHP program which runs on Apache Server receives request and process request .After processing the request, it sends a signal to motors through GPIO pins. Depending on the signal received from PHP program, bot is moved.

After processing this data in PHP server,these inputs will be passed to raspberry pi processor, this in turn connects to L293D which drives the motor and moves along the path tracing all the check points from source to destination to detect landmines

Activity Diagram(5.2) process flow:

1.End user to Apache Server.

2.Running program in Apache Server.

3.PHP Server data to Raspberry pi.

4.Raspberry pi includes GPIO pins.

5.Motor drives using L293D

**SEQUENCE DIAGRAM:**

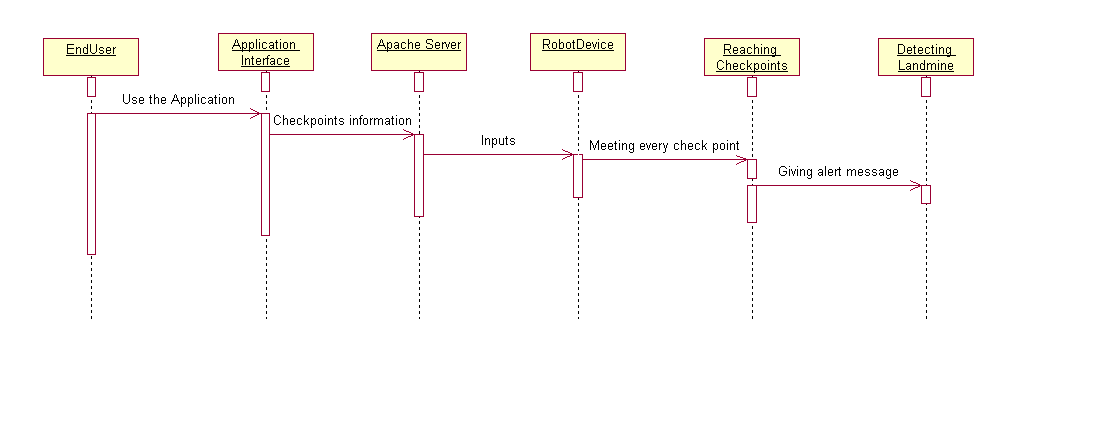


Fig 5.3:Sequence Diagram for LandMine Detection Robot controlled by Android

This sequence diagram explains process flow and interaction of processes in a sequence. It explains how input is taken at each stage and processed to next stage. It explains sequence of events through which entire process flow occurs. In the given diagram, process flow starts with end user interacting API and feeding input and this input is taken to server and processed in server. After the processing of input in server ,the signal is given to motors through GPIO pins .Then land mine detection event occurs. All these events occur in a sequence.

After processing this data in PHP server,these inputs will be passed to raspberry pi processor, this in turn connects to L293D which drives the motor and moves along the path tracing all the check points from source to destination to detect landmines

Sequence Diagram process flow:

1.End User to Application Interface

2.End User uses Application Tools

3.PHP server processes this inputs from Application Tool.

4.Processed Data to Raspberry pi.

5.Robot moves by Obtaining check points continuously.

6.Finally robot will detect landmines.

**CLASS DIAGRAM:**

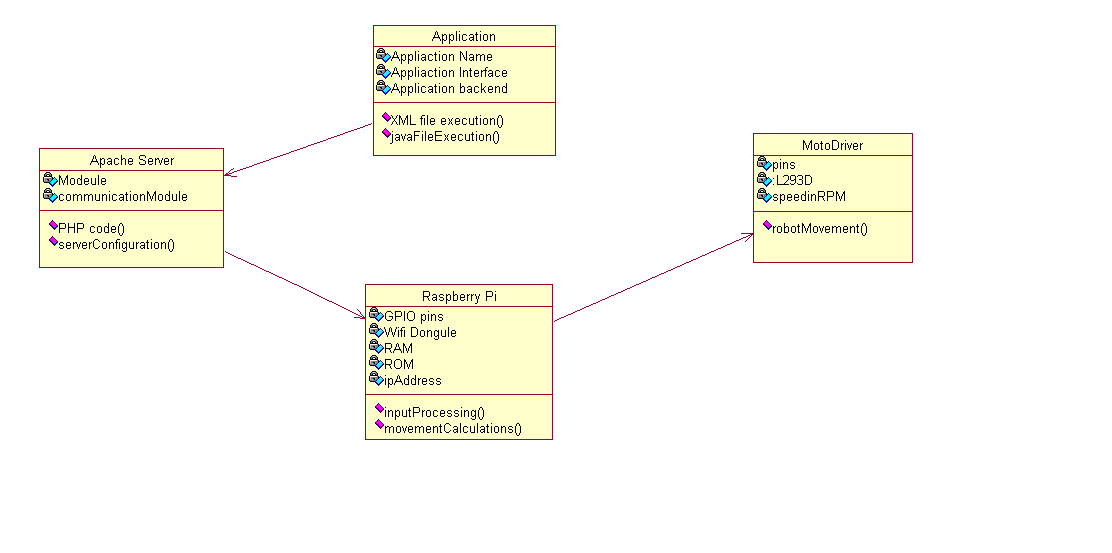


Fig 5.4: Class Diagram for LandMine Detection Robot controlled by Android

In the above class diagram, four classes Application, Apache Server, Raspberry Pi and Motor Driver are used. These four classes form main contents of project and help in achieving goals. Each class has several attributes which determines properties of class respectively. These classes interact with each other for successful functioning of bot.

Class Diagram Process Flow:

1.Application to Apache Server.

2.Class Diagrams contains Attributes and Functions.

EXAMPLE:

\*Application contains attributes and functions as follows:

\*Attributes:

\*Application Name

\*Application Interface

\*Application Backend

\*Functions:

\*XML File Execution

\*Java File Execution

3. In this way all the class Diagrams are linked :

Application to Apache.

Apache Server to Raspberry pi.

Raspberry pi to Motor Driver.

**COLLABORATION DIAGRAM**

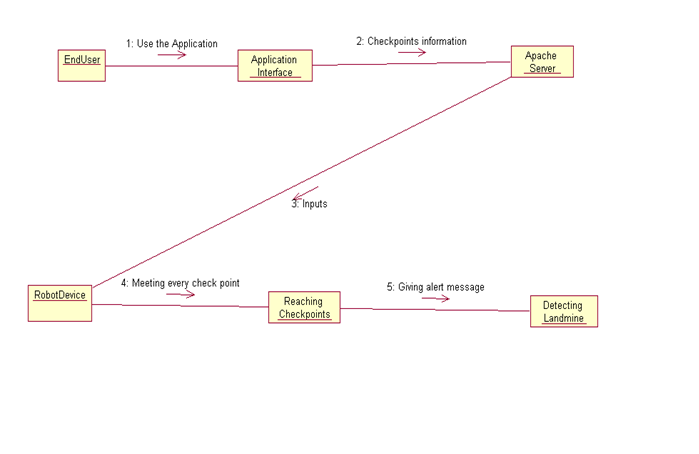


Fig 5.5: Collaboration Diagram for LandMine Detection Robot controlled by Android

This collaboration diagram describes how various objects are organized in the project. It also depicts communication between objects which send and receive messages. It describes dynamic behaviour of the system. In the above diagram, various objects like end-user, Apache sever interact with each other transferring system messages which help in successful functioning of the system.

In collaboration Diagram of this project explains about the way of process flow with Arrow marks having inputs to next object.

Process Flow:

1.End User will use the application interface.

2.Application Interface will give check point information.

3.Apache Server will give inputs to Robot Device.

4.Robot Device will meet all the check points in the path.

5.Finally Landmines will be detected by giving Alert message.

**8) Methodology and Approach:**

**Module No. 1: Assembling Robot and Integrating with Remote Desktop**

Assembled a robot which has raspberry pi Micro-Controller, operated through the remote desktop connection with a client server connection through the static IP of the raspberry pi. Raspberry pi consists external memory interface in which we load Raspbian operating system useful for accessing the files that are loaded into the Micro-controller. Input is given from the connected desktop to controller board, which sends the necessary signal to the actuators (motors) depending on the input received. Establishing remote desktop connection is done by setting up a local area network connection, both the raspberry pi and desktop are connected to the LAN. The raspbian operating system is loaded into the desktop for accessing, executing the functions of the robot.

**Module No. 2: Integrating Mobile with the Robot for getting the GPS location and Controlling the Robot Movement along the required checkpoints**

In this phase of project, we interface the robot with android application deployed in a Mobile. The android application use the Google’s polyline line concept for navigation.The application makes use of Google maps and determines a path between two places. Along the path obtained from it, more number of check points are created and the robot is made to trace through those check points for detecting the landmine.

**Module No. 3: Integrating Metal Detector with the Robot and Sending SMS to the End-User upon detection of Landmines.**

In this phase of project, we attach a landmine detection module (metal detector IC) to the robot. In the location where landmine gets detected an interrupt is generated and sent to raspberry pi controller which in-turn sends a signal to android application. Android application receives this signal from raspberry pi and sends corresponding location to end-user as a SMS.

In this, a client-server model is used between Robot and Android Mobile application. Apache Server which acts as a server and Android mobile application acts as a client which sends requests to the server which are processed on the server and based on the processing of request , a signal is generated and sent to motors of the robot through GPIO pins. Motors help in the rotation of the wheel based on the signal received.

**Apache Server :**

Apache Server is a leight-weight server which can respond to client requests very quickly. In this project , client is android mobile application which connects to this server and communicates information.

To develop Android mobile application, first knowledge on android development environment like Android Studio and Eclipse is necessary.

**ANDROID STUDIO :**

Android Studio is the official IDE for Android app Development, here we will do application for our project which controls the bot by giving inputs as check points along the source to destination path. Each project in Android Studio contains one or more modules with source code files and resource files. Different types of modules include:

1. Android app modules

2. Test modules

3. Library modules

4. App Engine modules

By using above tools, Successful functioning of robot is achieved and user can control the robot effectively.

**9) Output/Results:**

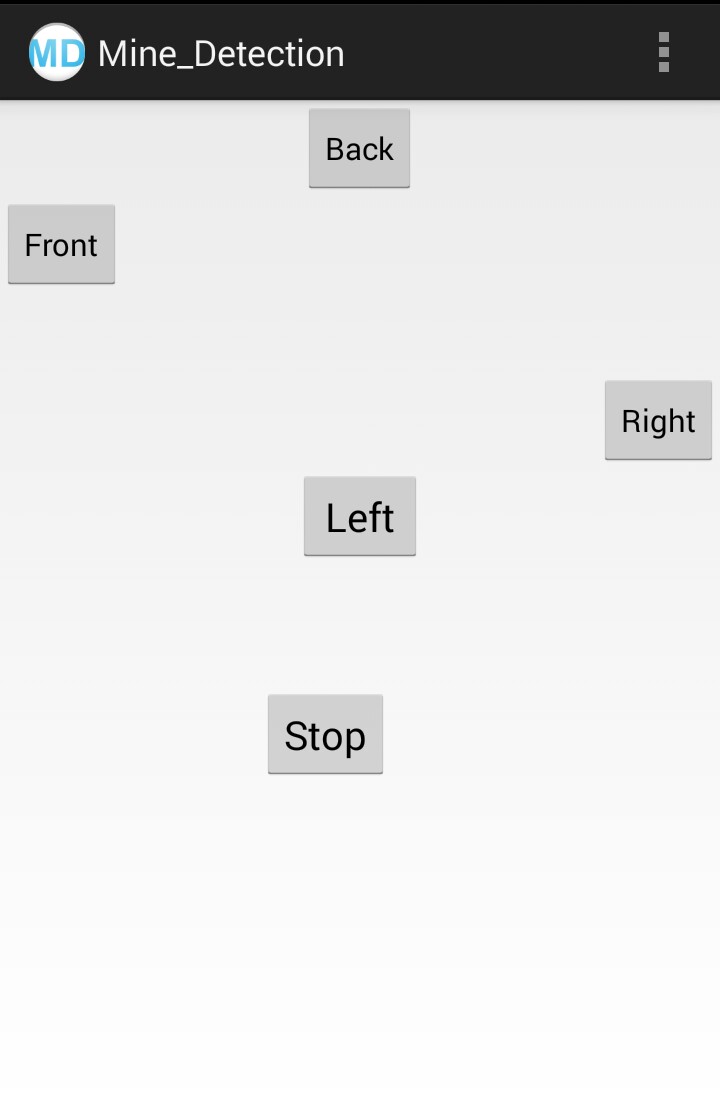


Fig 7.1 : Output Screen of Android Applicatin controlling Robot

In the above Fig 7.1 : output screen of android mobile application controlling robot is shown. It contains five buttons Front,Back,Right,Left and Stop respectively. It sends request to apache server based on the button clicked. Request is processed in Apache Server based on the command received.

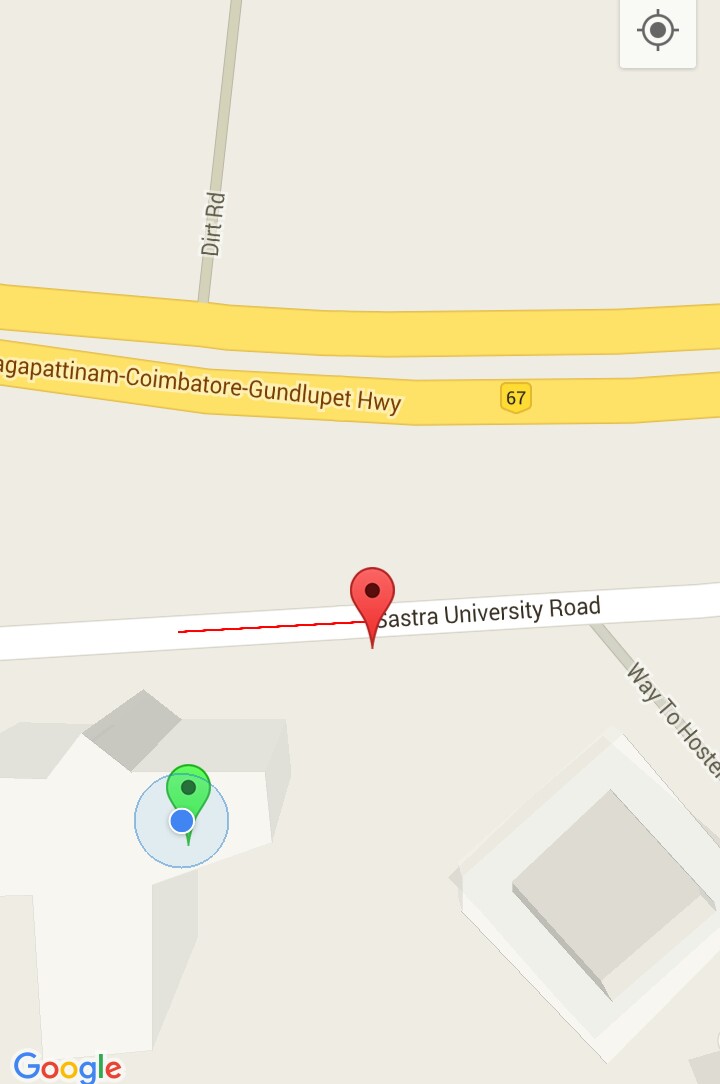


Fig 7.2 : Output Screen of Google Maps android application which retrieves path from source to destination

In the above output screen, path between source is destination is shown through red colour polyline. Green colour is source and red colour is destination.

**10) Conclusion:**

Thus, the main aim of building an android driven bot is to provide a completely autonomous robot for the detection of the land mines without any casualities is achieved.This robot can be used in the military applications where the risk for the human life will be more.These completely autonomous robots will provide a secure authentication since the LAN connected by the devices are completely in the private network.The developed project will be used in vast areas, not only in this land mind detection. This autonomous robot can be used for deep diving and also for the other applications.Our proposed system is the low cost product with in the 6000-7000 rupees of the components,where as in the existing system they are using the costly products which are not needed for the basic functions.

**11) References:**

**REFERENCE:**

[Yuanliang Zhang](http://link.springer.com/search?facet-creator=), [Dong Pyo Hong](http://link.springer.com/search?facet-creator=), “Navigation of mobile robot using Low-cost GPS” in [*International Journal of Precision Enginee…*](http://link.springer.com/journal/12541)*(2015)*,.