MSc CS - Part II SEM III

E-Journal

|  |  |
| --- | --- |
| **Roll No.** |  |
| **Name** |  |
| **Subject** |  |

1



CERTIFICATE

This is here to certify that Mr/Ms. , Seat Number M.Sc. II Computer Science, has satisfactorily completed the required number of experiments prescribed by the syllabus during the academic year 2022 – 2023.

Date:

Place: Mumbai

|  |  |
| --- | --- |
| Teacher In-Charge  External Examiner | Head of Department |

2

**Index**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr.**  **No.** | **Practical Name** | **Page No** | **Date** | **SIGN** |
| **1** | Design and develop location based messaging app | 3-9 |  |  |
| **2** | Design and develop chat messaging app which is a location-based. | 10-15 |  |  |
| **3** | Design and develop app for Device  Group Messaging | 16-21 |  |  |
| **4** | Demonstrate use of OpenGTS (Open Source GPS Tracking  System) | 22-26 |  |  |
| **5** | Develop application demonstrating Human Computer Interaction | 27-28 |  |  |
| **6** | Write a Java Card applet | 29-31 |  |  |

**Practical No: 1**

**Aim**: **Design and develop location-based messaging app.**

**Source Code: activity\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*

<**RelativeLayout xmlns:android=**[**"http://schemas.android.com/apk/res/android"**](http://schemas.android.com/apk/res/android) **xmlns:tools**[**="http://schemas.android.com/tools"**](http://schemas.android.com/tools) **android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:paddingBottom="@dimen/activity\_vertical\_margin" android:paddingLeft="@dimen/activity\_horizontal\_margin" android:paddingRight="@dimen/activity\_horizontal\_margin" android:paddingTop="@dimen/activity\_vertical\_margin" tools:context="com.example.dell.lbmsg.MainActivity"**>

<**TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:textAppearance="?android:attr/textAppearanceLarge " android:text="Large Text" android:id="@+id/textView" android:layout\_alignParentLeft="true" android:layout\_alignParentStart="true"** />

<**TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:textAppearance="?android:attr/textAppearanceLarge " android:text="Large Text" android:id="@+id/textView3" android:layout\_below="@+id/textView" android:layout\_alignParentLeft="true" android:layout\_alignParentStart="true" android:layout\_marginTop="96dp"** />

<**Button android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Get Longitude and Latitude" android:id="@+id/button" android:layout\_centerVertical="true" android:layout\_alignParentLeft="true" android:layout\_alignParentStart="true"** />

<**TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"**

**android:textAppearance="?android:attr/textAppearanceLarge " android:text="Large Text" android:id="@+id/textView2"**

**android:layout\_alignParentBottom="true" android:layout\_alignRight="@+id/button" android:layout\_alignEnd="@+id/button"** />

</**RelativeLayout**>

**MainActivity.java**

**package** com.example.dell.lbmsg; **import** android.Manifest; **import** android.content.Context; **import** android.content.pm.PackageManager; **import** android.location.Location; **import**

android.location.LocationListener;

**import**

android.location.LocationManager;

**import** android.net.Uri;

**import** android.os.Bundle;

**import** android.support.v7.app.AppCompatActivity; **import**

android.support.v4.app.ActivityCompat; **import** android.util.Log; **import** android.view.View; **import** android.widget.Button; **import** android.widget.TextView;

**import** com.google.android.gms.appindexing.Action; **import** com.google.android.gms.appindexing.AppIndex; **import** com.google.android.gms.common.api.GoogleApiClient;

**public class MainActivity extends AppCompatActivity implements LocationListener {** TextView **t1**, **t2**, **t3**; Button **b1**; **protected** LocationManager **locationManager**; **protected** LocationListener **locationListener**; **double lat**, **longg**;

**private** GoogleApiClient **client**; @Override

**protected void** onCreate(Bundle savedInstanceState)

{ **super**.onCreate(savedInstanceState); setContentView(R.layout.***activity\_main***);

**t1** = (TextView) findViewById(R.id.***textView***); **t2**

= (TextView) findViewById(R.id.***textView3***); **t3**

= (TextView) findViewById(R.id.***textView2***); **b1**

= (Button) findViewById(R.id.***button***);

**b1**.setOnClickListener(**new** View.OnClickListener() { @Override

**public void** onClick(View view) { **t1**.setText(**"Latitude = "** + **lat**); **t2**.setText(**"Longitude = "** + **longg**);

**if** (**lat**<38 &&**lat**>36 &&**longg**<122 &&**longg**>118) {

**t3**.setText(**"In-Side The Area"**);

} **else** { **t3**.setText(**"Out-Side The Area"**);

}

}

});

**locationManager**= (LocationManager) getSystemService(Context.***LOCATION\_SERVICE***);

**if** (ActivityCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_FINE\_LOCATION***) != PackageManager.***PERMISSION\_GRANTED***&&ActivityCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_COARSE\_LOCATION***) != PackageManager.***PERMISSION\_GRANTED***) {

**return**;

}

**locationManager**.requestLocationUpdates(LocationManager.***GPS\_PROVIDER***, 0, 0,

**this**);

**client** = **new** GoogleApiClient.Builder(**this**).addApi(AppIndex.***API***).build();

}

@Override

**public void** onLocationChanged(Location location) {

*// txtLat = (TextView) findViewById(R.id.textview1);* **lat**= location.getLatitude(); **longg**= location.getLongitude(); Log.*d*(**""** + **lat**, **""** + **lat**);

Log.*d*(**""** + **longg**, **""** + **longg**);

**if** (**lat**== 38 &&**longg**== 118) { **t3**.setText(**"You Are at Perfect Place !!!!"**);

} **else** { **t3**.setText(**"You are not at Perfect Place**

**!!!!"**);

}

@Override

**public void** onStatusChanged(String provider, **int** status, Bundle extras) { Log.*d*(**"Latitude"**,

**"status"**);

}

@Override

**public void** onProviderDisabled(String provider) { Log.*d*(**"Latitude"**, **"disable"**);

}

@Override

**public void** onProviderEnabled(String provider) { Log.*d*(**"Latitude"**, **"enable"**);

}

@Override

**public void** onStart() {

**super**.onStart();

//*ATTENTION: This was auto-generated to implement the App Indexing API.*

//*See https://g.co/AppIndexing/AndroidStudio for more information.*

**client**.connect();

Action viewAction = Action.*newAction*( Action.***TYPE\_VIEW***, *//*

***TODO: choose an action type.***

**"Main Page"**, *//* ***TODO: Define a title for the content shown.***

//***TODO: If you have web page content that matches this app activity's content,***

//*make sure this auto-generated web page URL is correct.* //*Otherwise, set the URL to null.*

Uri.*parse*[(**"http://host/path"**](http://host/path)),

*//* ***TODO: Make sure this auto-generated app URL is correct.***

Uri.*parse*(**"android-app://com.example.dell.lbmsg/http/host/path"**)

);

AppIndex.***AppIndexApi***.start(**client**, viewAction);

}

@Override **public void** onStop()

{

**super**.onStop();

//*ATTENTION: This was auto-generated to implement the App Indexing API.*

//*See https://g.co/AppIndexing/AndroidStudio for more information.*

Action viewAction = Action.*newAction*( Action.***TYPE\_VIEW***, *//* ***TODO: choose an action type.***

**"Main Page"**, *//* ***TODO: Define a title for the content shown.***

//***TODO: If you have web page content that matches this app activity's content,***

//*make sure this auto-generated web page URL is correct.* //*Otherwise, set the URL to null.* Uri.*parse*[(**"http://host/path"**](http://host/path)),

*//* ***TODO: Make sure this auto-generated app URL is correct.***

Uri.*parse*(**"android-app://com.example.dell.lbmsg/http/host/path"**)

);

AppIndex.***AppIndexApi***.end(**client**, viewAction);

**client**.disconnect();}

}

**AndroidManifest.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*

<**manifest xmlns:android**[**="http://schemas.android.com/apk/res/android"**](http://schemas.android.com/apk/res/android) **package="com.example.dell.lbmsg"**>

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

<**uses-permission android:name="android.permission.INTERNET"** />

<**application android:allowBackup="true" android:icon="@mipmap/ic\_launcher" android:label="@string/app\_name" android:supportsRtl="true" android:theme="@style/AppTheme"**>

<**activity android:name=".MainActivity"**>

<**intent-filter**>

<**action android:name="android.intent.action.MAIN"** />

<**category android:name="android.intent.category.LAUNCHER"** /> </**intent- filter**>

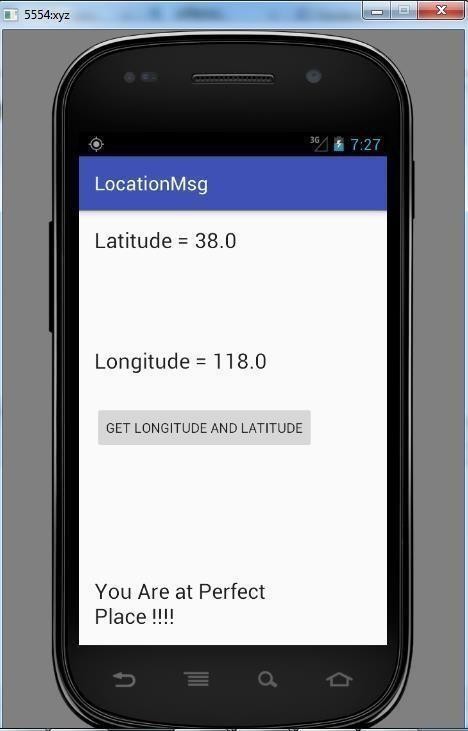
</**activity**>

<**meta-data**

**android:name="com.google.android.gms.version" android:value="@integer/google\_play\_services\_version"** />

</**application**>

</**manifest**>

**Output:**

# Practical No: 2

**Aim: Design and develop chat messaging app which is a location based.**

**Source Code :**

**activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>*

<RelativeLayout xmlns:android=["http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) xmlns:tools=["http://schemas.android.com/tools"](http://schemas.android.com/tools) android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:paddingBottom="@dimen/activity\_vertical\_margin" android:paddingLeft="@dimen/activity\_horizontal\_margin" android:paddingRight="@dimen/activity\_horizontal\_margin" android:paddingTop="@dimen/activity\_vertical\_margin" tools:context="com.example.dell.location\_based.MainActivity">

<TextView android:layout\_width="wrap\_content "

android:layout\_height="wrap\_conten t"

android:textAppearance="?android:attr/textAppearanceLar g e" android:text="Large Text" android:id="@+id/textView" android:layout\_alignParentLeft="true" android:layout\_alignParentStart="true" />

<TextView android:layout\_width="wrap\_content "

android:layout\_height="wrap\_conten t"

android:textAppearance="?android:attr/textAppearanceLar g e" android:text="Large Text" android:id="@+id/textView3" android:layout\_below="@+id/textView" android:layout\_alignParentLeft="true" android:layout\_alignParentStart="true" android:layout\_marginTop="96dp" />

<Button

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Get Longitude and

Latitude" android:id="@+id/button" android:layout\_centerVertical="true" android:layout\_alignParentLeft="true" android:layout\_alignParentStart="true" />

<TextView android:layout\_width="wrap\_content "

android:layout\_height="wrap\_conten t"

android:textAppearance="?android:attr/textAppearanceLarg e" android:text="Large Text" android:id="@+id/textView2" android:layout\_alignParentBottom="true" android:layout\_alignRight="@+id/button" android:layout\_alignEnd="@+id/button" />

</RelativeLayout>

**MainActivity.java**

**package** com.example.dell.location\_based;

**import** android.Manifest; **import** android.content.Context;

**import** android.content.pm.PackageManager; **import** android.location.Location; **import** android.location.LocationListener; **import** android.location.LocationManager; **import** android.support.v4.app.ActivityCompat; **import** android.util.Log; **import** android.view.View; **import** android.widget.Button; **import** android.widget.TextView;

**import** android.support.v7.app.AppCompatActivity; **import**

android.os.Bundle;

**public class MainActivity extends AppCompatActivity implements LocationListener** { TextView **t1**, **t2**, **t3**; Button **b1**; **protected** LocationManager **locationManager**; **protected** LocationListener **locationListener**; **double lat**, **longg**;

@Override

**protected void** onCreate(Bundle savedInstanceState)

{ **super**.onCreate(savedInstanceState);

setContentView(R.layout.***activity\_main***); *// setContentView(R.layout.activity\_main);*

**t1** = (TextView) findViewById(R.id.***textView***); **t2**

= (TextView) findViewById(R.id.***textView3***); **t3**

= (TextView) findViewById(R.id.***textView2***); **b1**

= (Button) findViewById(R.id.***button***);

**b1**.setOnClickListener(**new** View.OnClickListener() { @Override

**public void** onClick(View view) { **t1**.setText(**"Latitude = "** + **lat**); **t2**.setText(**"Longitude = "**+ **longg**);

**if**(**lat**<38 &&**lat**>36 &&**longg**<122 &&**longg**>118)

{

**t3**.setText(**"In-Side The Area"**);

}

**else** {

**t3**.setText(**"Out-Side The Area"**);

} }

});

**locationManager**= (LocationManager) getSystemService(Context.***LOCATION\_SERVICE***); **if**

(ActivityCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_FINE\_LOCATION***) != PackageManager.***PERMISSION\_GRANTED***&&ActivityCompat.*checkSelfPermission*(**this**, Manifest.permission.***ACCESS\_COARSE\_LOCATION***) != PackageManager.***PERMISSION\_GRANTED***) {

**return**;

}

**locationManager**.requestLocationUpdates(LocationManager.***GPS\_PROVIDER***, 0, 0,

**this**);

}

@Override

**public void** onLocationChanged(Location location) { **lat**=location.getLatitude(); **longg**= location.getLongitude();

Log.*d*(**""**+**lat**,**""**+**lat**);

Log.*d*(**""**+**longg**,**""**+**longg**);

**if**(**lat**<38 &&**lat**>36 &&**longg**<122 &&**longg**>118) {

**t3**.setText(**"In-Side The Area"**);

}

**else** { **t3**.setText(**"Out-Side The Area"**);

}

}

@Override

**public void** onProviderDisabled(String provider) { Log.*d*(**"Latitude"**,**"disable"**);

}

@Override

**public void** onProviderEnabled(String provider) { Log.*d*(**"Latitude"**,**"enable"**);

}

@Override

**public void** onStatusChanged(String provider, **int** status, Bundle extras) { Log.*d*(**"Latitude"**,**"status"**);

}

}

**AndroidManifest.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*

<**manifest xmlns:android**[**="http://schemas.android.com/apk/res/android"**](http://schemas.android.com/apk/res/android) **package="com.example.dell.location\_based"**>

<**application android:allowBackup="true" android:icon="@mipmap/ic\_launcher" android:label="@string/app\_name" android:supportsRtl="true" android:theme="@style/AppTheme"**>

<**activity android:name=".MainActivity"**>

<**intent-filter**>

<**action android:name="android.intent.action.MAIN"** />

<**category android:name="android.intent.category.LAUNCHER"** /> </**intent- filter**>

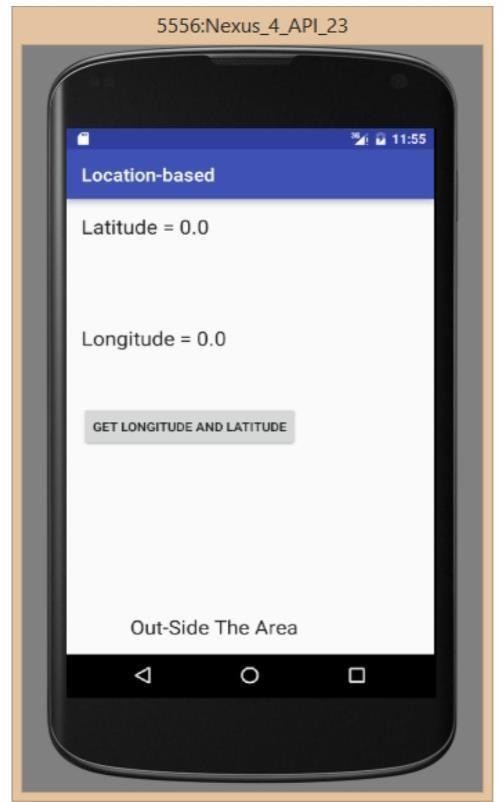
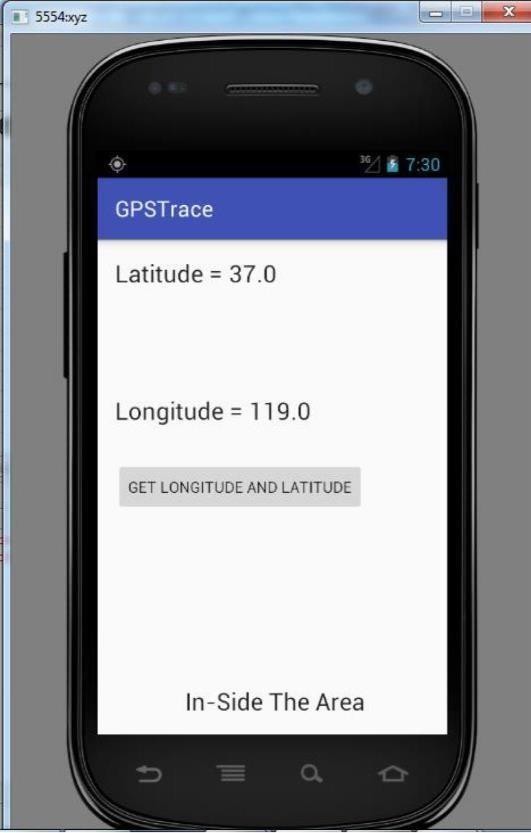
</**activity**>

</**application** >

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

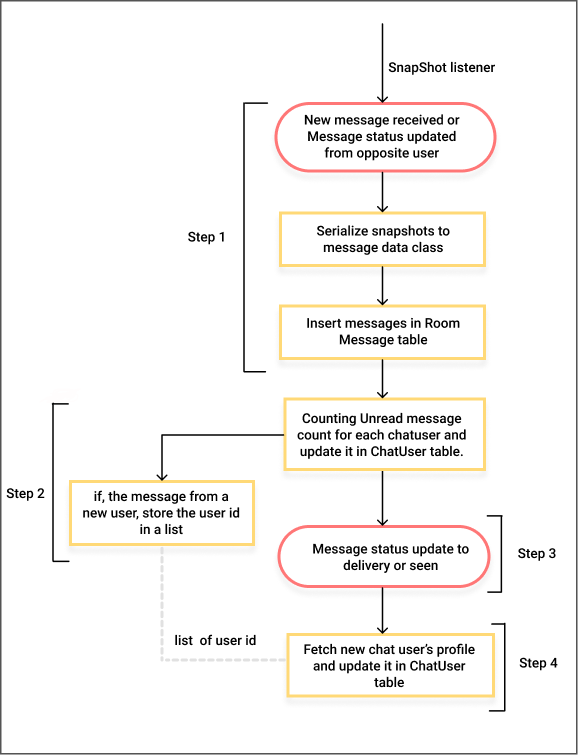
<**uses-permission android:name="android.permission.INTERNET"** />

</**manifest**>

**Output :**

# Practical No: 3

**Aim: Design and develop app for Device Group Messaging**



class FSingleChat : Fragment(), MessageSenderCallback{ @Inject

lateinit var dbRepository: DbRepository

fun sendMessage(){

val message= createMessage("my first message") msgDao.insertMessage(message) val db = FirebaseFirestore.getInstance() val messageCollection= db.collection("Messages")

val messageSender= MessageSender(msgCollection= messageCollection, dbRepo: dbRepository, chatUser: currentChatUser, listener: this) messageSender.send(fromUser= myUserId,

toUser= currentChatUser.id, message= message)

}

fun createMessage(msg: String): Message{ return Message(id= generateId(), createdAt= dateToUtc(), from= myUserId, to= currentChatUser.id,

chatUsers= listOf(myUserId,currentChatUser.id), textMessage= TextMessage(msg))

}

fun generateId(length: Int= 20): String{ val alphaNumeric = ('a'..'z') + ('A'..'Z') + ('0'..'9')

return alphaNumeric.shuffled().take(length).joinToString("") //ex: bwUIoWNCSQvPZh8xaFuz

}

fun dateToUTC(date: Date=Date()): String {

// will be converted back to local time zone before showing in recycler view

val formatterUTC: DateFormat = SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss.SSS'Z'", Locale.getDefault())

formatterUTC.timeZone = TimeZone.getTimeZone("UTC") return formatterUTC.format(date)

}

override fun onSuccess(message: Message) { LogMessage.v("Message sender Sucesss ${message.id}") messageDao.insertMessage(message)

//push notification code goes here

}

override fun onFailed(message: Message) { LogMessage.v("Message sender Failed ${it.message}") messageDao.insertMessage(message)

}

}

interface MessageSenderCallback{ fun onSuccess(message: Message) fun onFailed(message: Message)

}

class MessageSender(private val msgCollection: CollectionReference, private val dbRepo: DbRepository, private val chatUser: ChatUser, private val listener: MessageSenderCallback) { fun send(fromUser: String, toUser: String, message: Message) { val docId = chatUser.documentId if (!docId.isNullOrEmpty()){

Timber.v("Case 1 ${chatUser.documentId}") send(docId, message)

} else {

//so we don't create multiple nodes for same chat msgCollection.document("${fromUser}\_${toUser}").get(

) .addOnSuccessListener { documentSnapshot -> if (documentSnapshot.exists()) {

//this node exists send your message Timber.v("Case 2")

send("${fromUser}\_${toUser}", message)

} else {

//senderId\_receiverId node doesn't exist check receiverId\_senderId msgCollection.document("${toUser}\_${fromUser}").get()

.addOnSuccessListener { documentSnapshot2 -> if (documentSnapshot2.exists()) { Timber.v("Case 3") send("${toUser}\_${fromUser}", message)

} else {

//no previous chat history(senderId\_receiverId & receiverId\_senderId both don't exist)

//so we create document senderId\_receiverId then messages array then add messageMap to messages

//this node exists send your message

//add ids of chat members Timber.v("Case 4")

msgCollection.document("${fromUser}\_${toUser}")

.set(mapOf("chat\_members" to FieldValue.arrayUnion(fromUser, toUser)), SetOptions.merge()

).addOnSuccessListener {

LogMessage.v("chat member update successfully") send("${fromUser}\_${toUser}", message)

}.addOnFailureListener {

LogMessage.v("chat member update failed ${it.message}")

}

} }

} }

}

}

private fun send(doc: String, message: Message){ try

{

chatUser.documentId=doc dbRepo.insertUser(chatUser)

message.status=1 //changing message status to sent message.chatUsers= arrayListOf(message.from,message.to) val messageCopy=message.copy(message).apply {

chatUserId= null //chatUserId field is being used only for relation query,changing to null will ignore this field

createdAt= null // will get this message in snapshot listener with server time replaced

}

msgCollection.document(doc).collection("messages").document(message.id).set( messageCopy,

SetOptions.merge()

).addOnSuccessListener { listener.onSuccess(message)

}.addOnFailureListener { message.status=4 listener.onFailed(message)

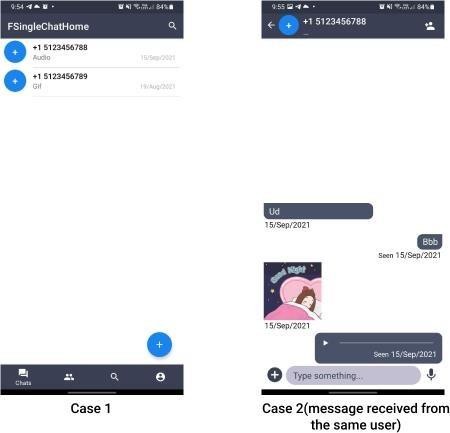
}

} catch (e: Exception) { e.printStackTrace()

} }

}

**Output:**



**Practical No: 4**

**Aim: Demonstrate use of OpenGTS (Open Source GPS Tracking System)**

**Required Software:-**

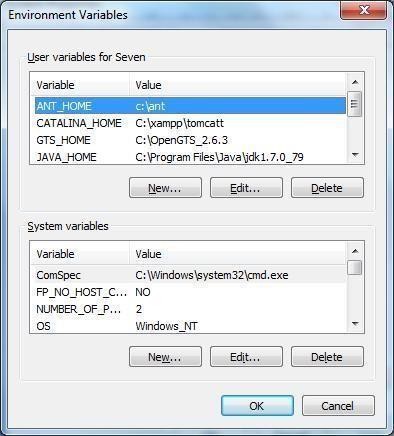
1. **JDK 1.6(Any Version)**
2. **XAMPP Server**
3. **Mysql-java connector(copy .jar file and paste it into)**

----- C:\Program Files\Java\jdk1.8.0\_111\jre\lib\ext -----

C:\Program Files\Java\jre1.8.0\_111\lib\ext

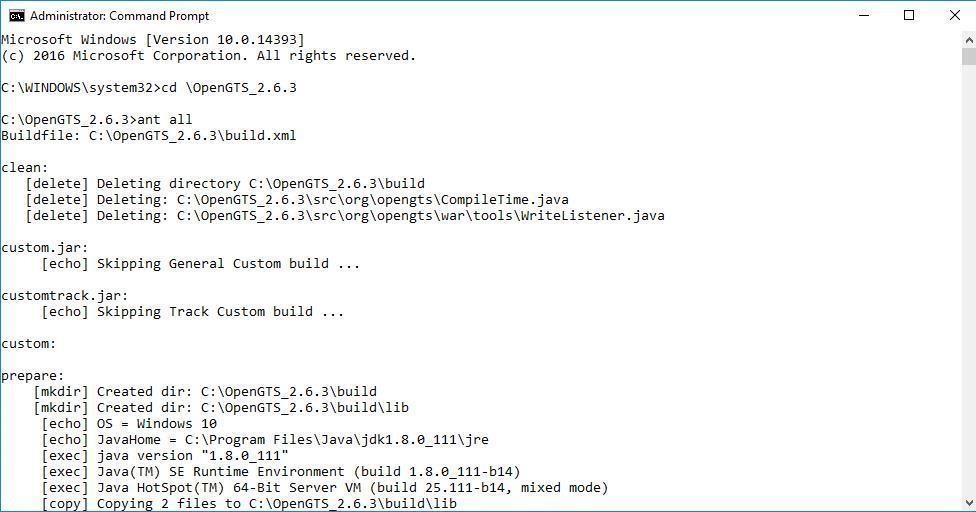
1. **OpenGTS application.**
2. **Apache Ant** [**http://www.opengts.or**](http://www.opengts.or/)

**g**

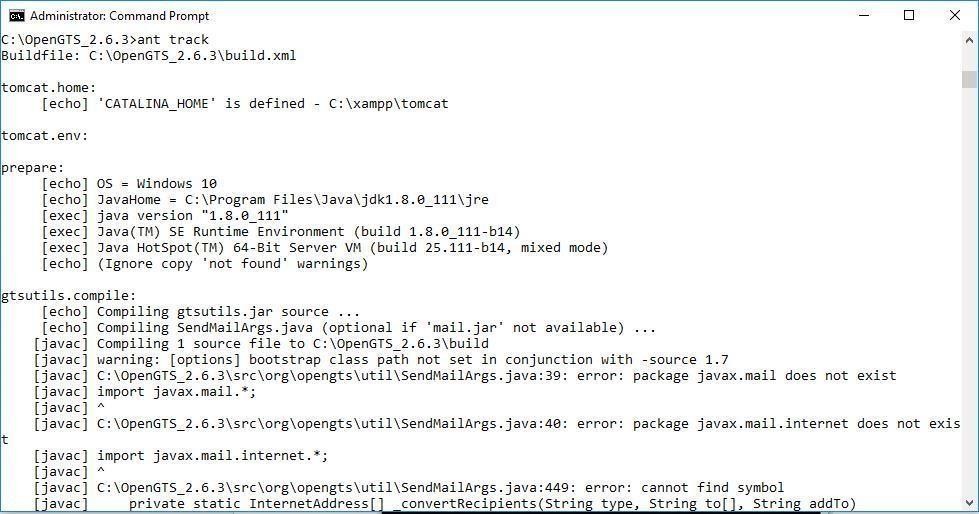


**ADMIN CMD--**

1. **Open command Prompt and go to D:\OpenGTS\_2.6.2**
2. **Type command ant all**



1. **Type command ant track**

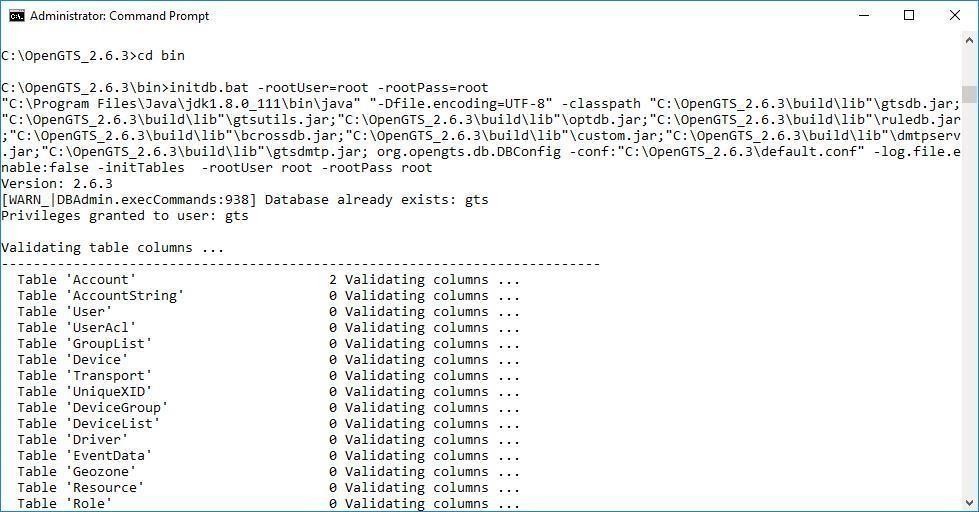


1. **Type command ant track.deploy**



1. **(type cd bin)Type command initdb.bat –rootUser=root –rootPass=root** --

--- start Tomcat/MySQL.

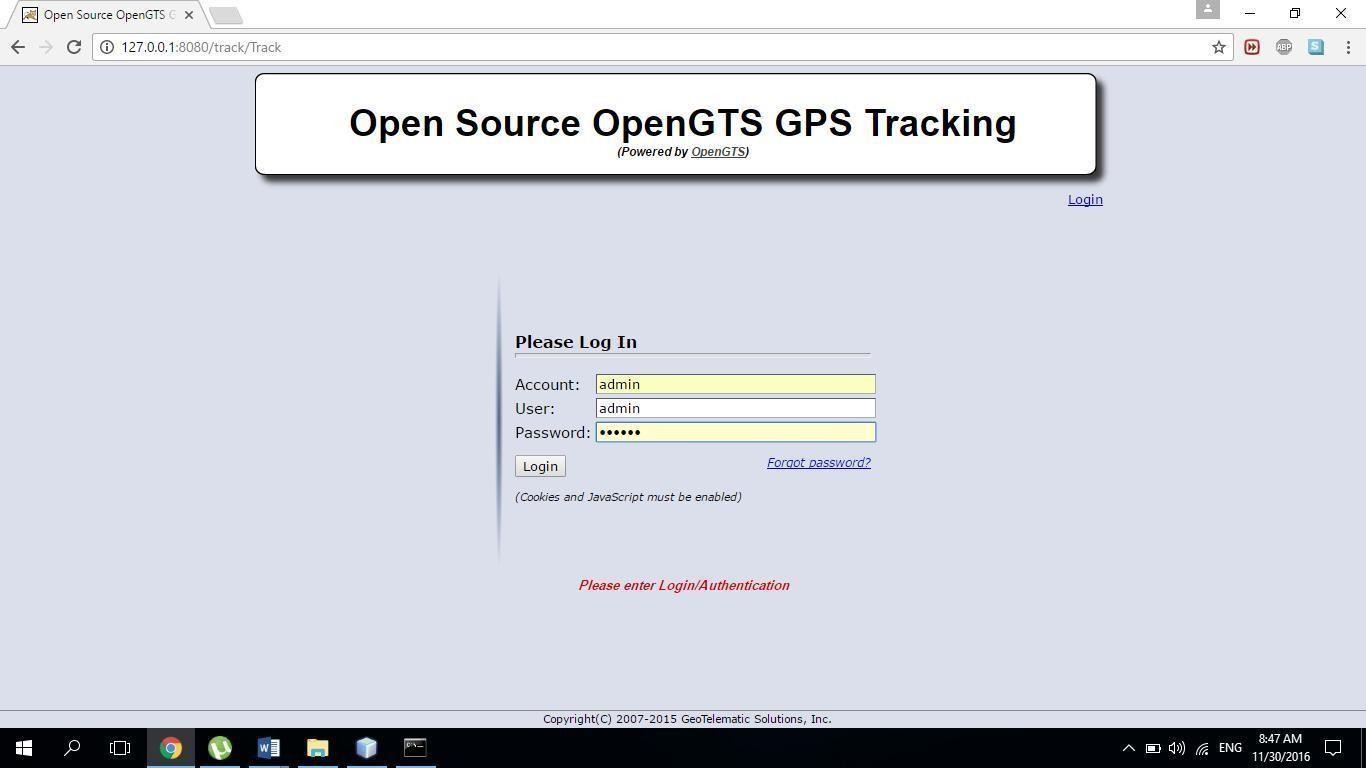


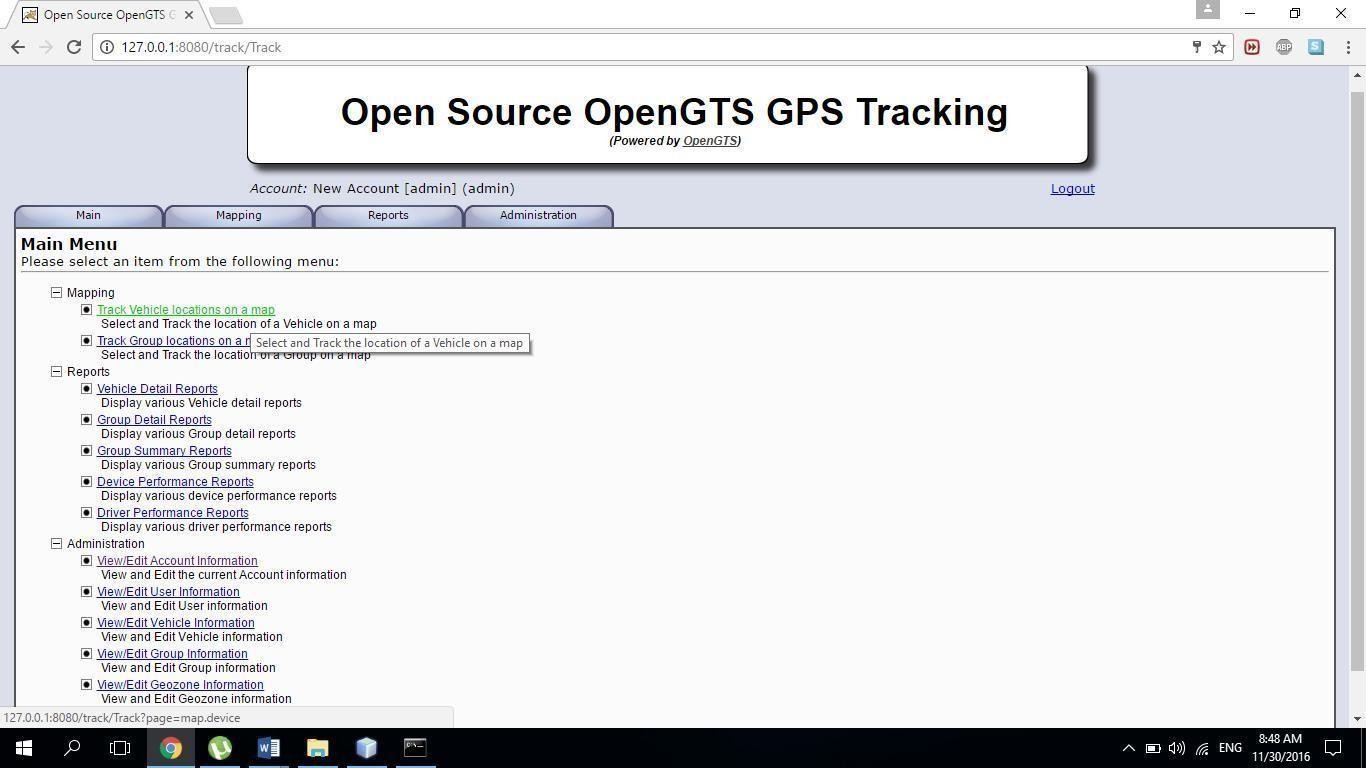
1. **Type Command admin.bat Account –account:admin –pass:123456 –**

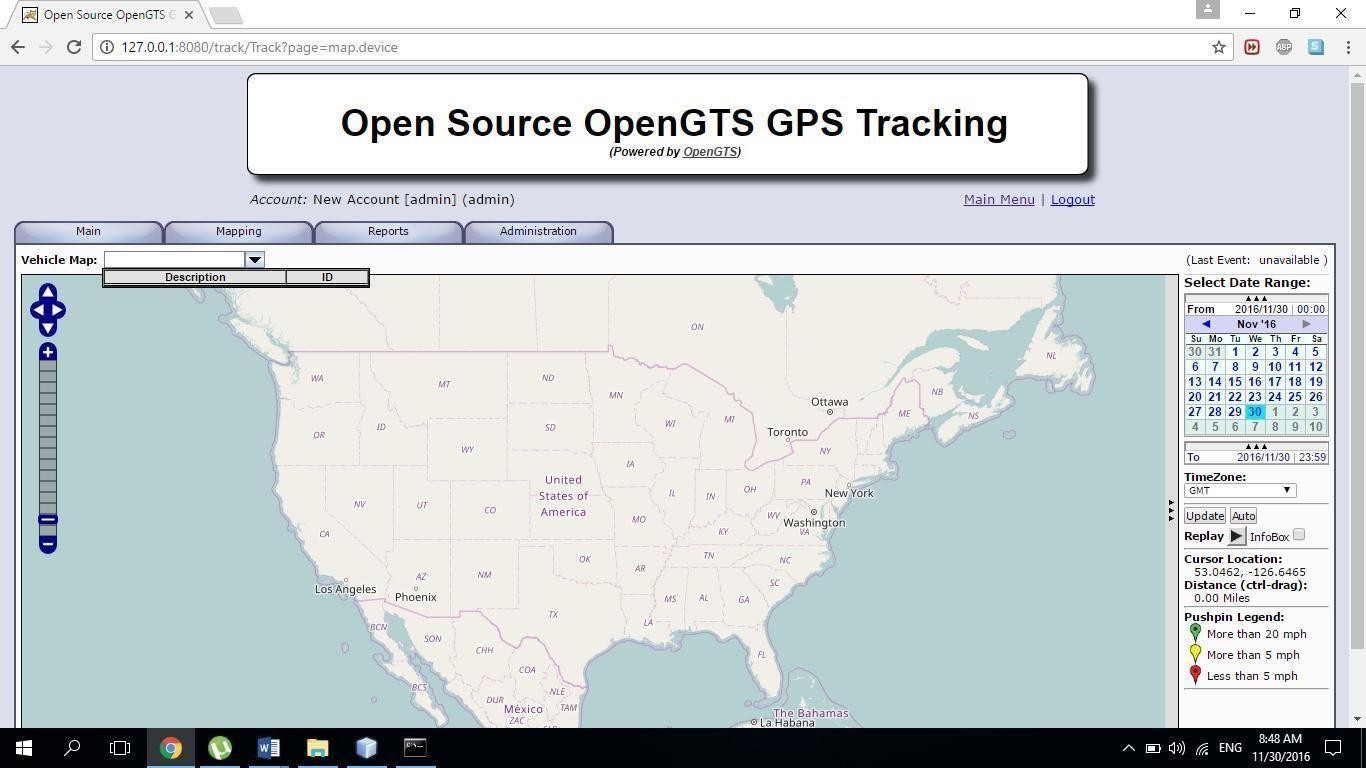
**create**



1. **Type url 127.0.0.1:8080/track/Track and login with admin and 123456**







# Practical No: 5

**Aim : Develop application demonstrating Human Computer Interaction.**

**Source Code:**

import java.awt.Color; import java.awt.Frame; import

java.awt.Window; import java.awt.event.WindowEvent; import java.awt.event.WindowListener;

**public class CloseableSimpleWarning extends Frame implements WindowListener{**

static private final int frame\_height = 150; static private final int frame\_width = 250;

public CloseableSimpleWarning() { setBackground(Color.gray); setForeground(Color.black); setTitle("HCI"); setSize(frame\_width, frame\_height); addWindowListener(this);

}

@Override

public void windowClosing (WindowEvent e) { System.out.println("Window is closing."); dispose();

}

public void windowClosed (WindowEvent e) { System.out.println("Window closed.");

}

@Override

public void windowIconified (WindowEvent e) { System.out.println("Window iconified.");

}

@Override

public void windowDeiconified (WindowEvent e) { System.out.println("Window deiconified.");

}

@Override public void windowOpened (WindowEvent e) { System.out.println("Window opened.");

}

@Override

public void windowActivated (WindowEvent e) { System.out.println("Window activated.");

}

@Override

public void windowDeactivated (WindowEvent e) {

System.out.println("Window deactivated.");

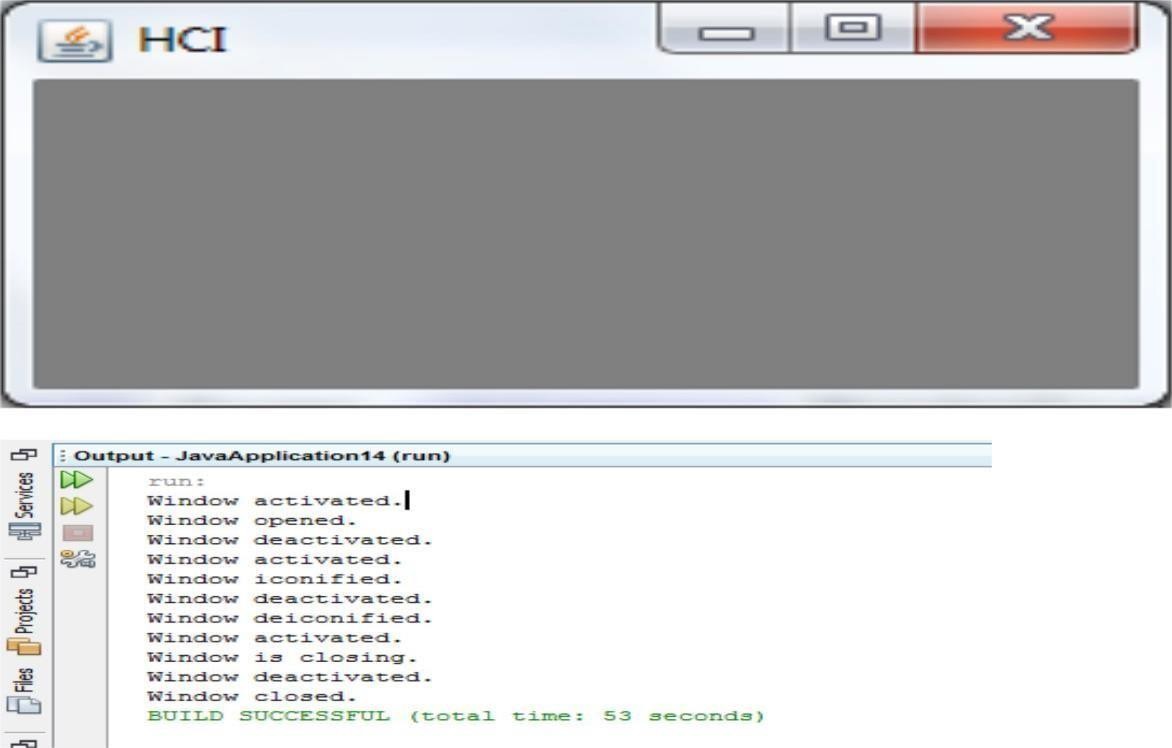
}

public static void main(String [] args) { CloseableSimpleWarning f = new CloseableSimpleWarning(); f.setVisible(true);

}

}

OUTPUT:



# Practical No: 6

**Implementation :**

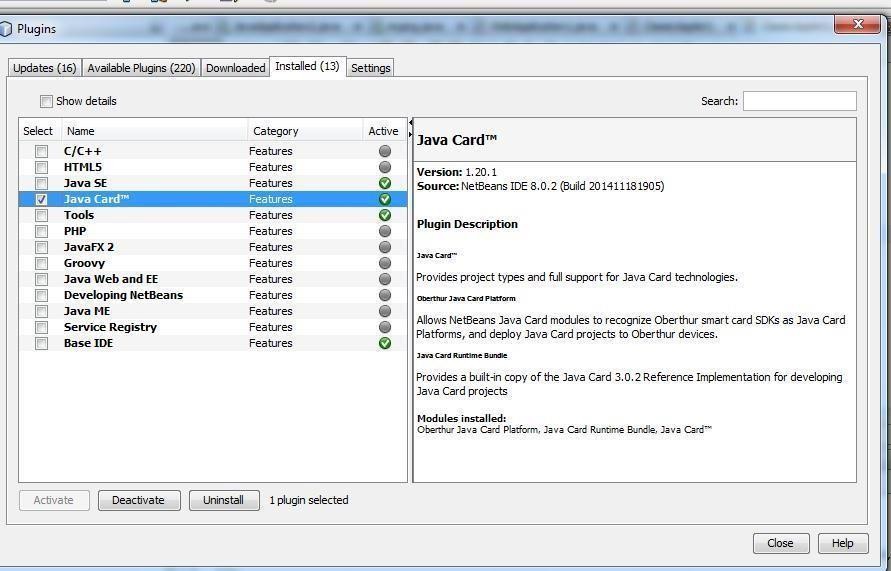
# Aim: Write a Java Card applet.

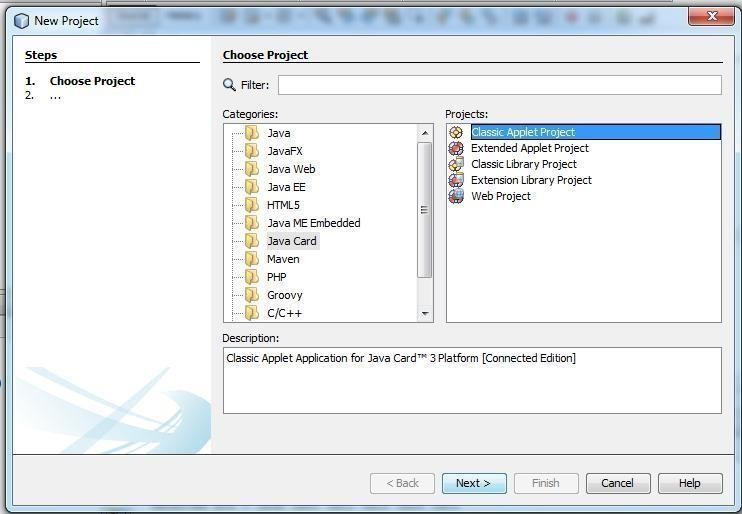
**Step 1 :**-> Download Java Card Sdk from [**http://www.oracle.com/technetwork/java/embedded/javacard/downloads/javacard- sdk-**](http://www.oracle.com/technetwork/java/embedded/javacard/downloads/javacard-sdk-2043229.html)[**2043229.html**](http://www.oracle.com/technetwork/java/embedded/javacard/downloads/javacard-sdk-2043229.html)

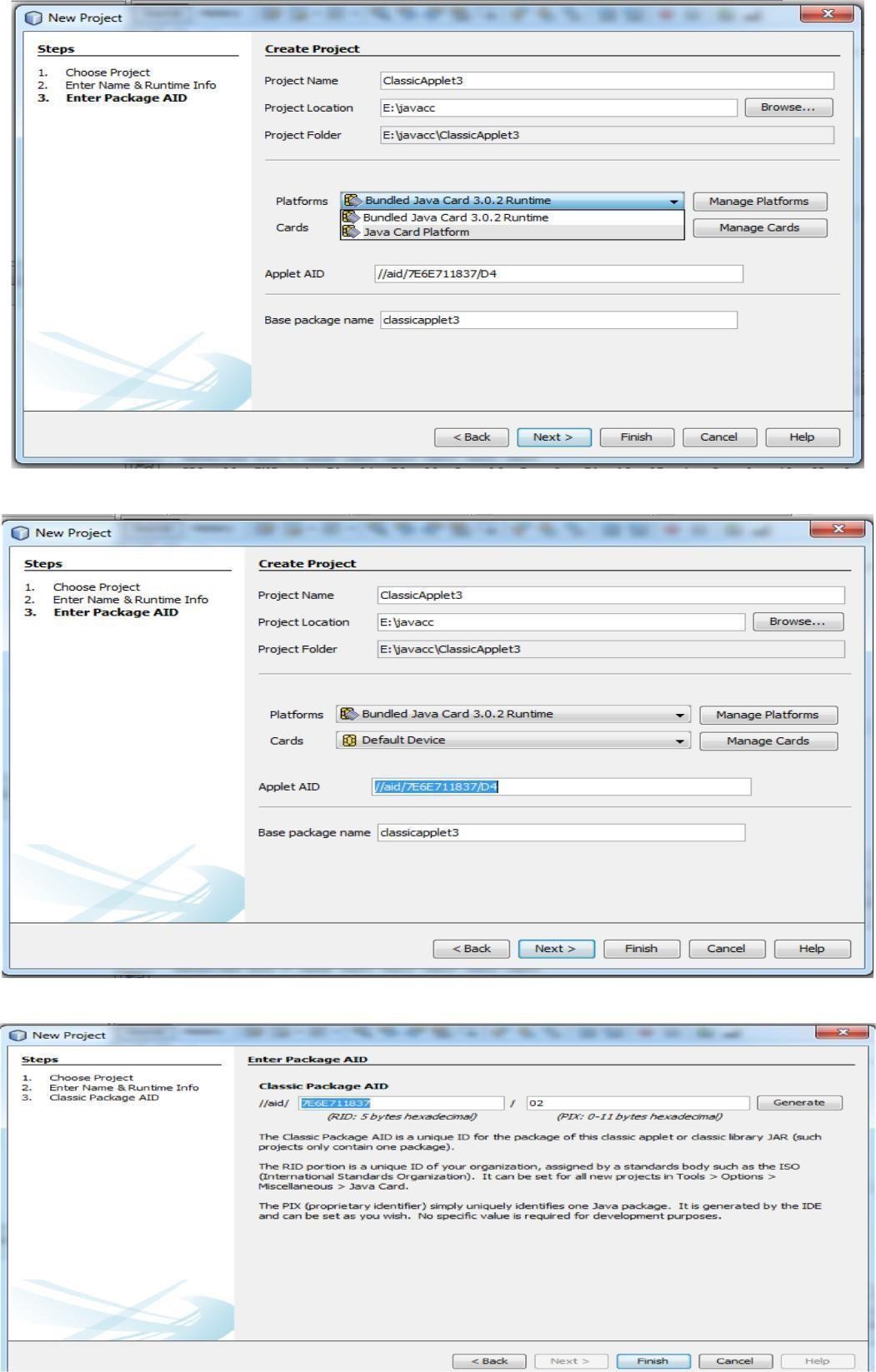
if (plugin is there in NetBeans) Do nothing;

else

Install in Netbeans as plugin :-

**Tools -> Plugin**

**Step 2:->Create a Java Card Application**



**Source Code :**

package classicapplet1; import javacard.framework.\*; public class ClassicApplet2 extends Applet { private byte[] received;

private static final short MAX\_LENGTH = 256; private static final byte[] helloFidesmo={(byte)'H',(byte)'e',(byte)'l',(byte)'l',(byte)'o',(byte)'

',(byte)'F',(byte)'i',(byte)'d',(byte)'e',(byte)'s',(byte)'m',(byte)'o',(byte)'!'};

public static void install(byte[] bArray, short bOffset, byte bLength) { new ClassicApplet2();

}

/\*\*

\* Only this class's install method should create the applet object. \*/

protected ClassicApplet2() { received

= new byte[MAX\_LENGTH]; register();

}

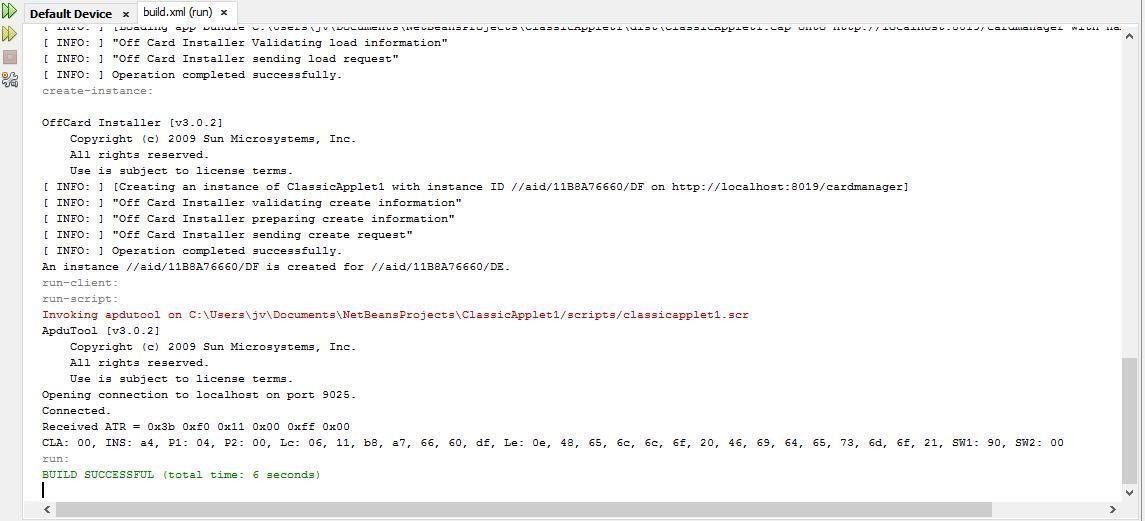
public void process(APDU apdu) {

//Insert your code here byte buffer[] = apdu.getBuffer();

short length = (short) helloFidesmo.length; Util.arrayCopyNonAtomic(helloFidesmo, (short)0, buffer, (short)0, (short)length); apdu.setOutgoingAndSend((short)0, length);

}

}

OUTPUT: