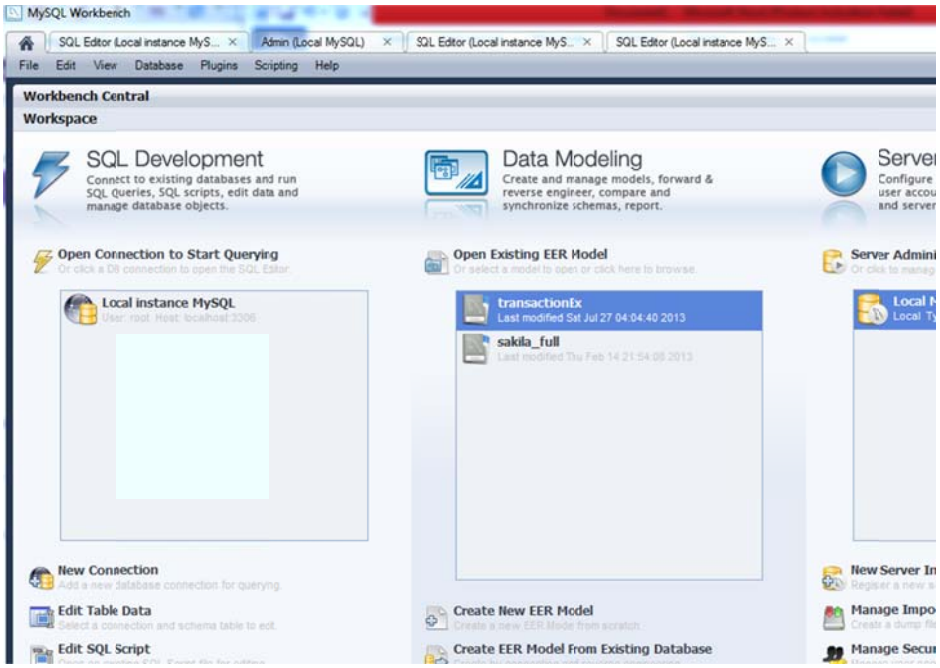
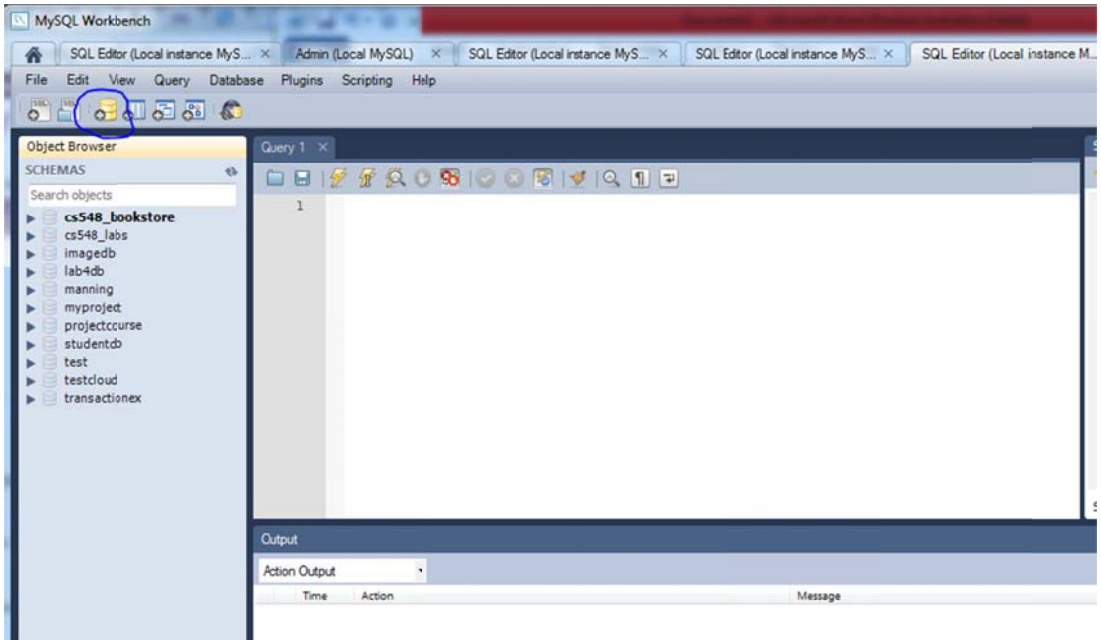
Phonegap and MySQL database Example

1. Setup and programming for server side with database.

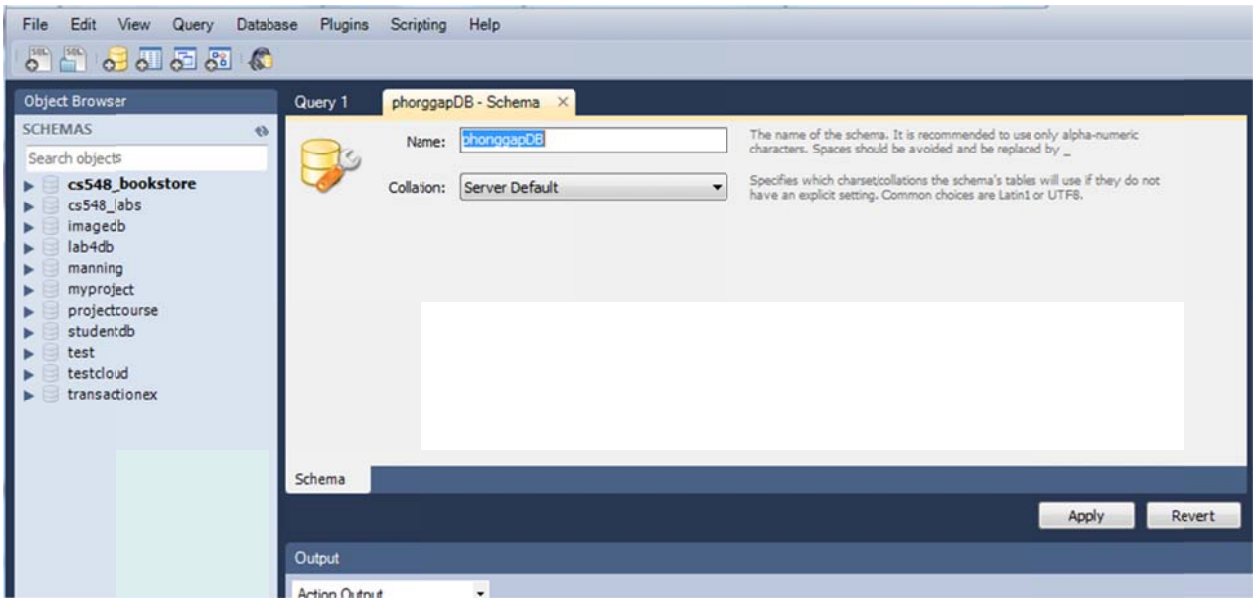
- Installed mySQL server and MySQL workbench.
- Open mySQL workbench, create a new database and a table as following:



Click on " Local Instance MYSQL"

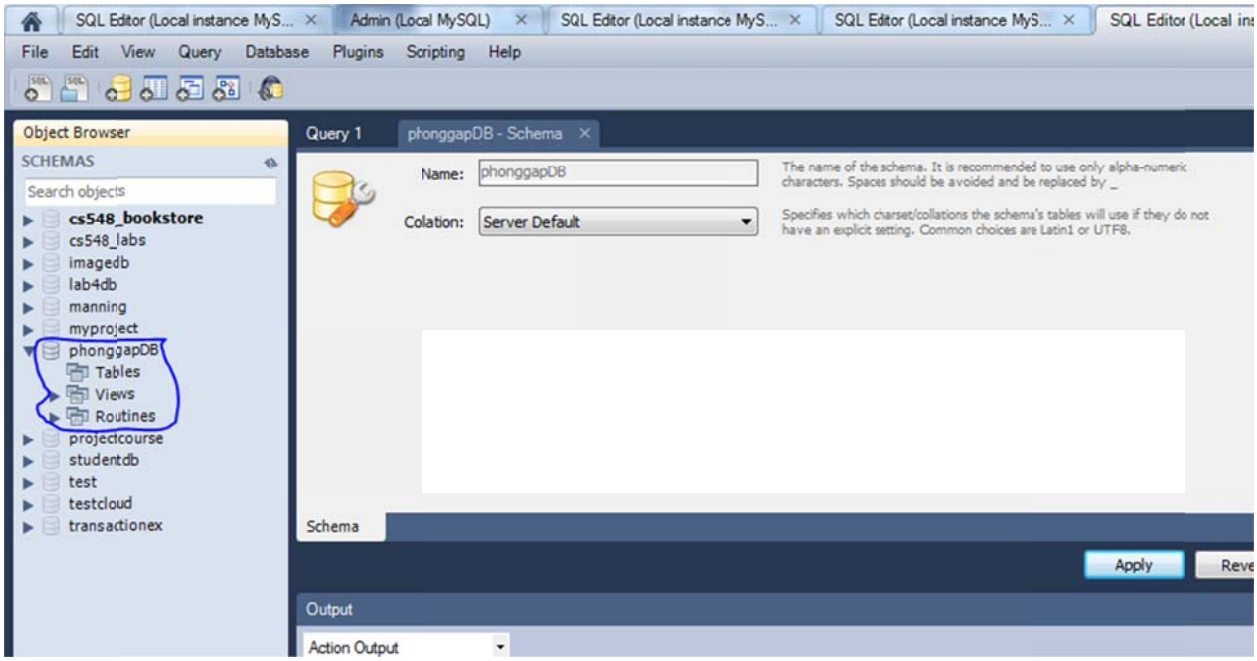


Click on create database icon to create a new database.

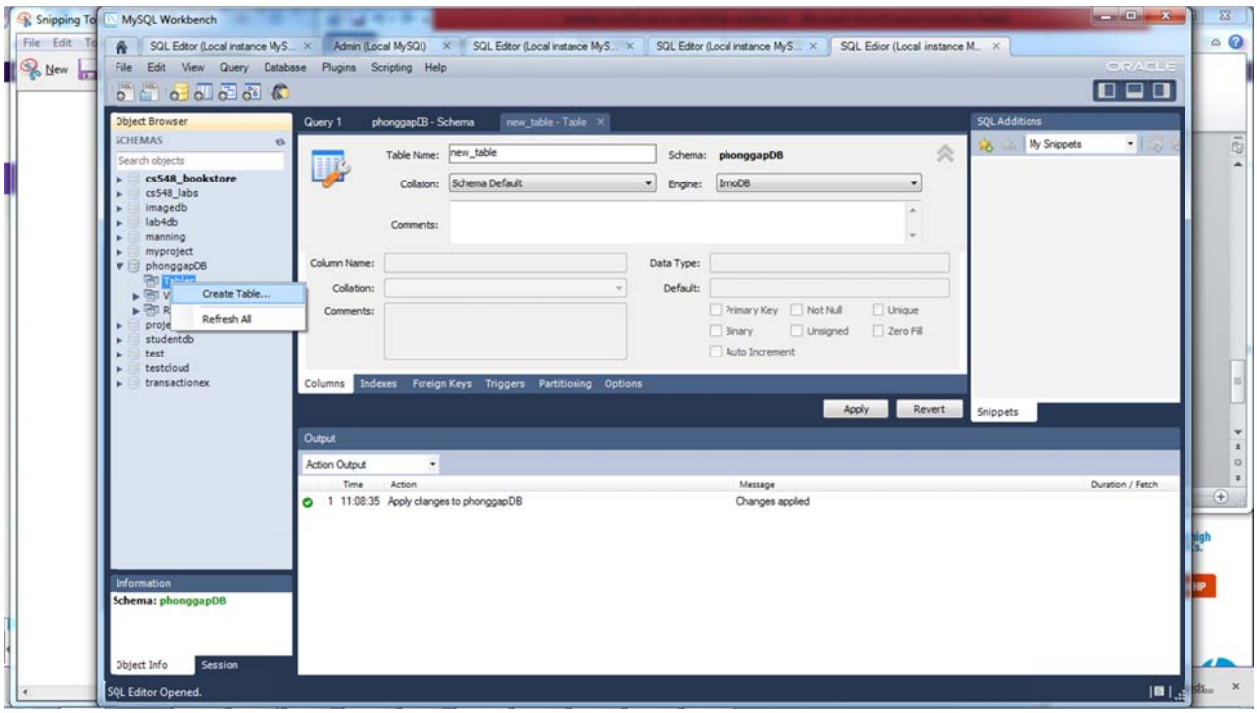


For example: type the name of database as “phonegapDB”

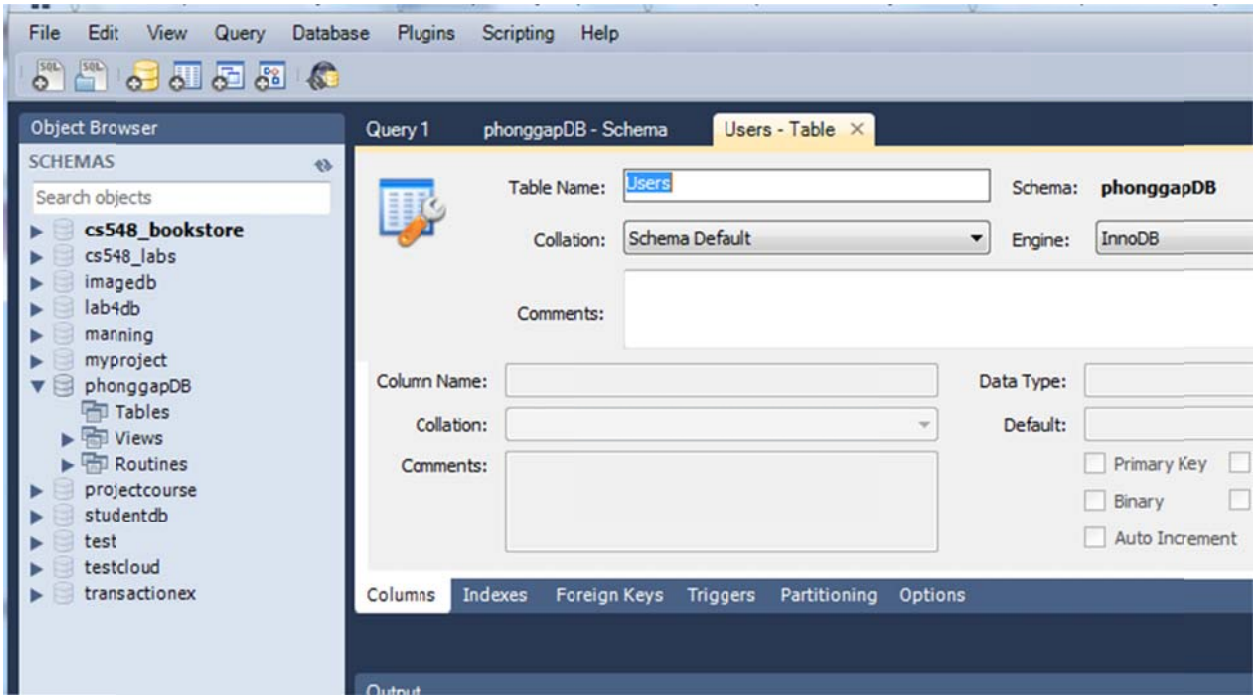
Click Apply.



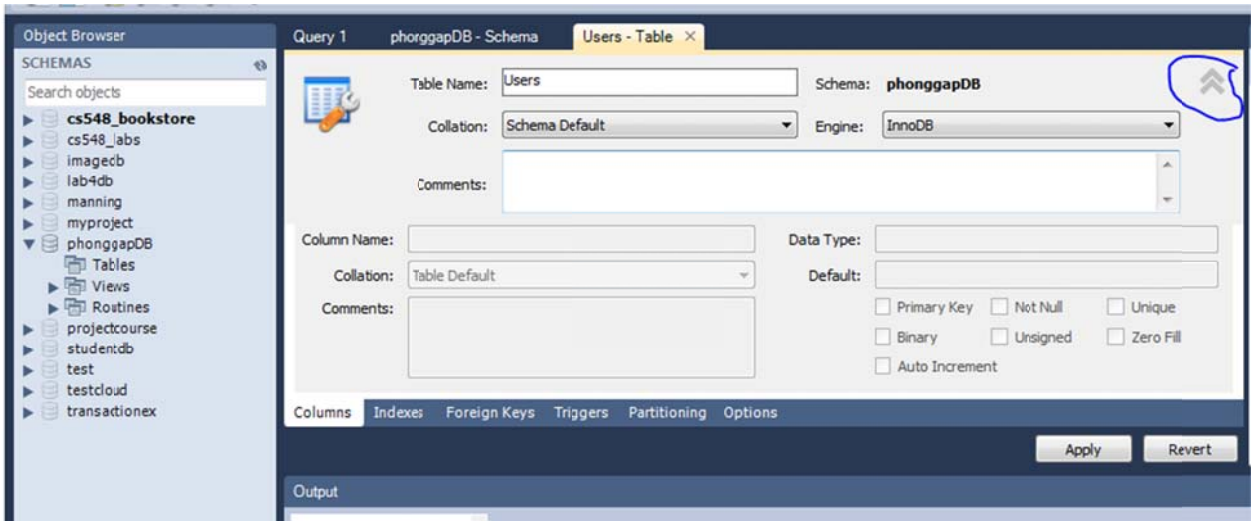
Right click on Tables of database phonggapDB to create a new table.



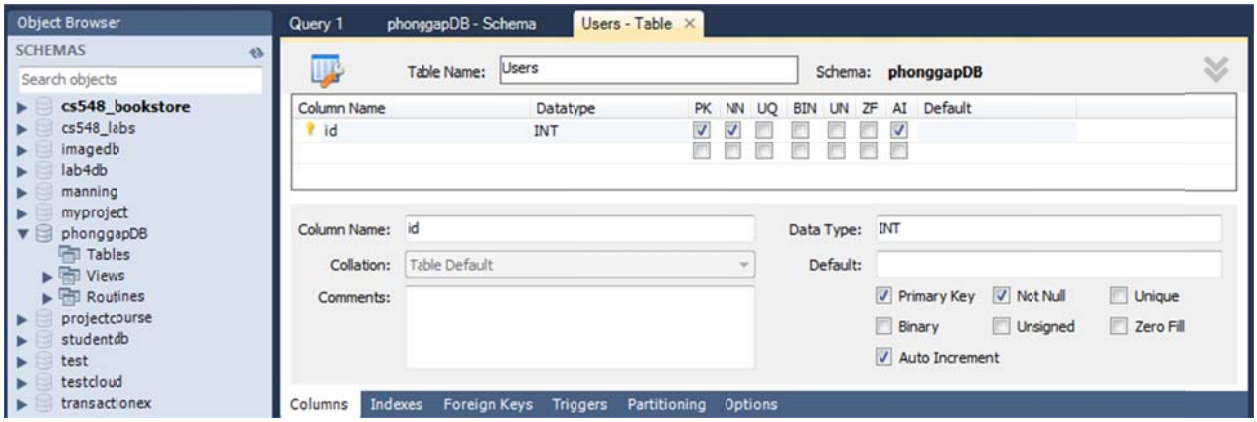
For example, enter table name as Users to create Users table.



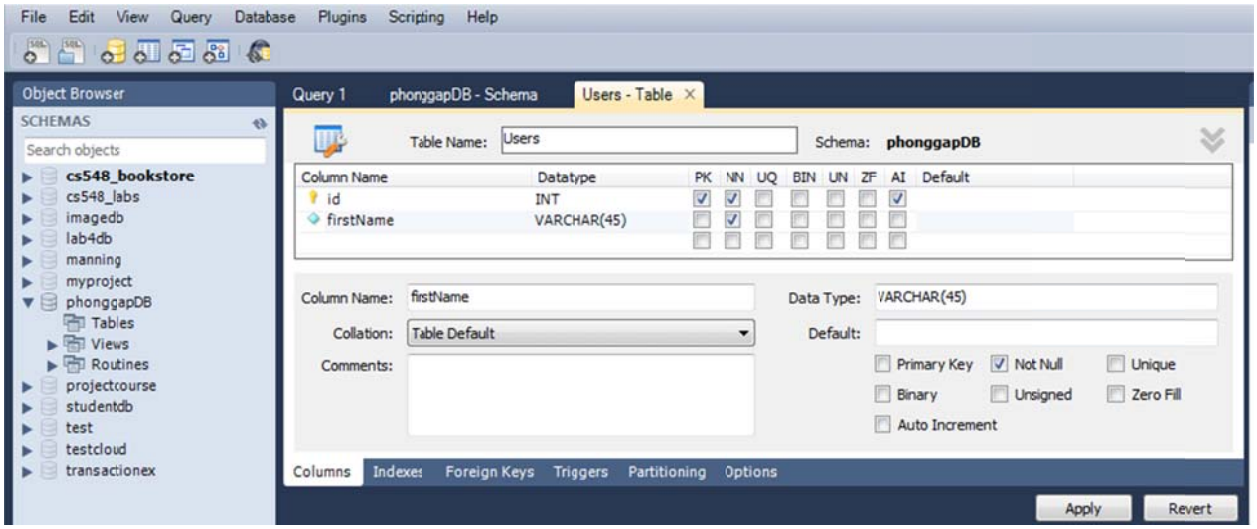
Click the icon “up” to go to the screen to add some columns.



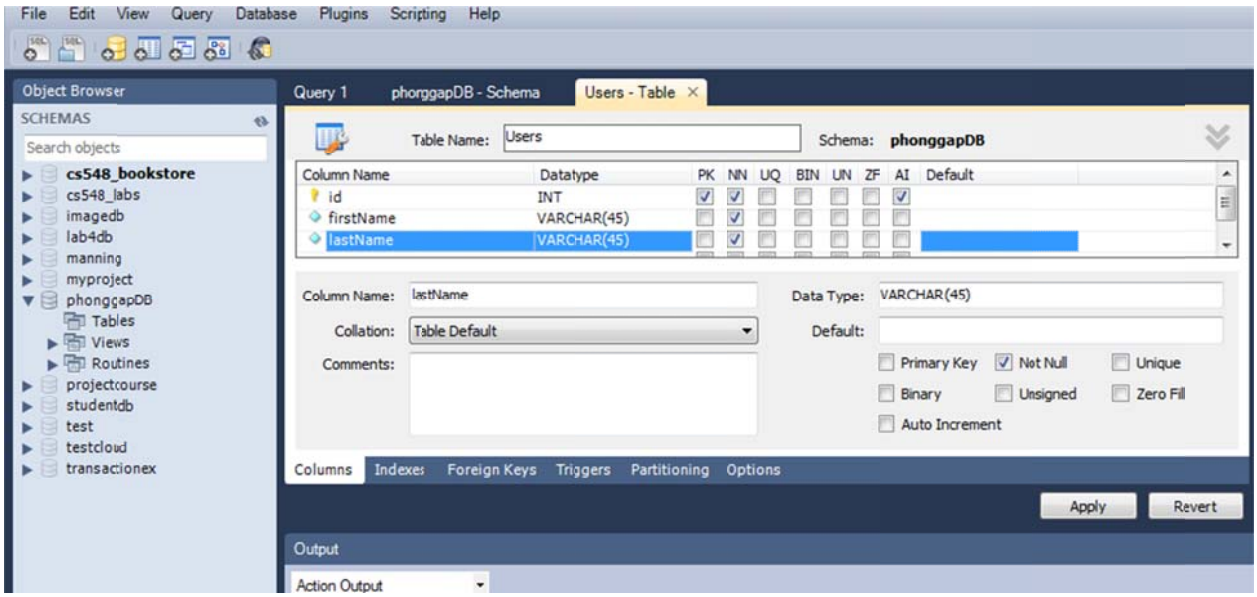
Add “id” column as following



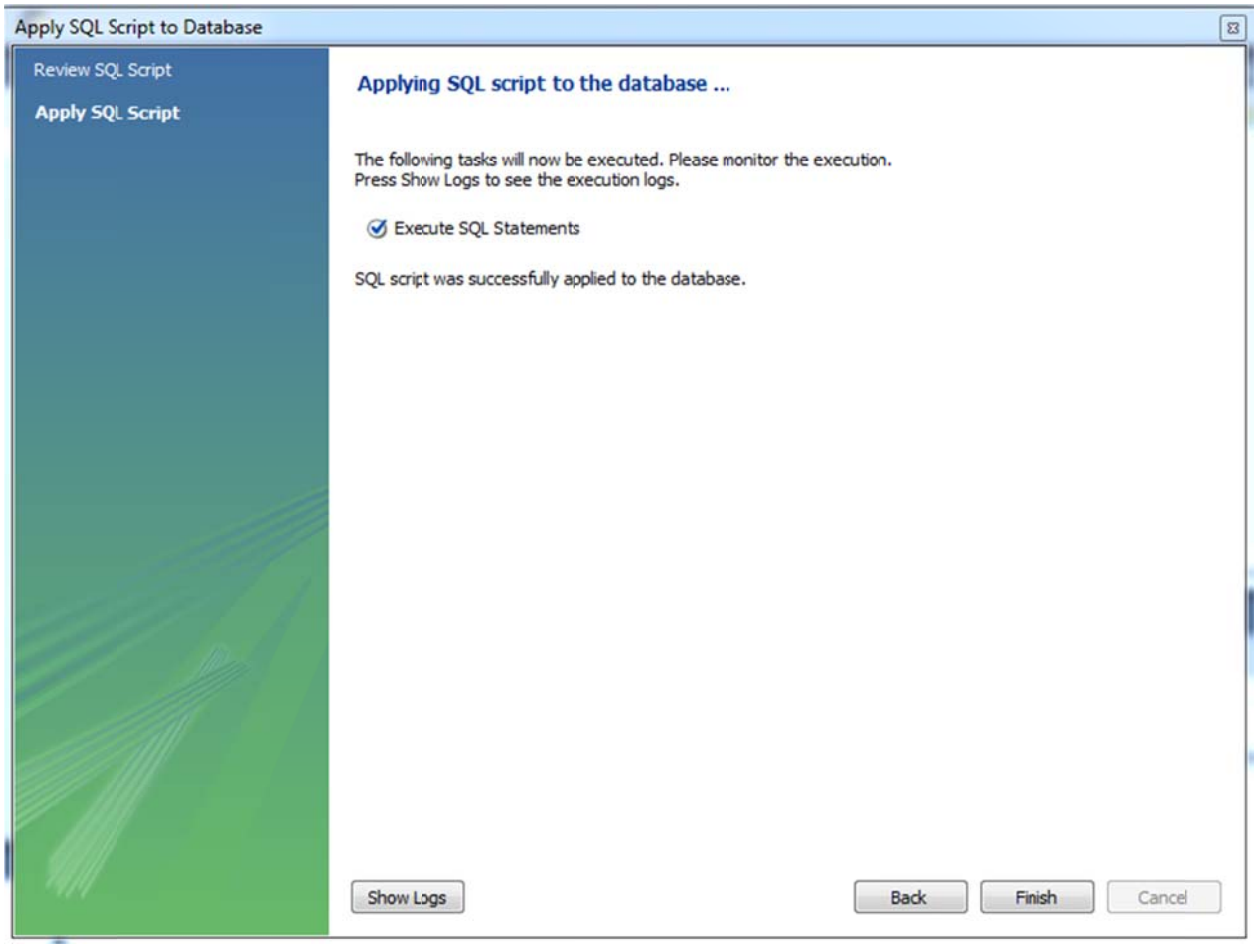
Add “firstName” column as following:



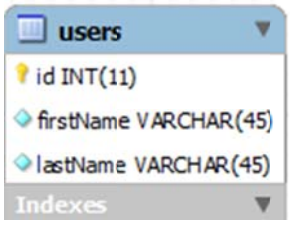
Add “lastName” column as following:



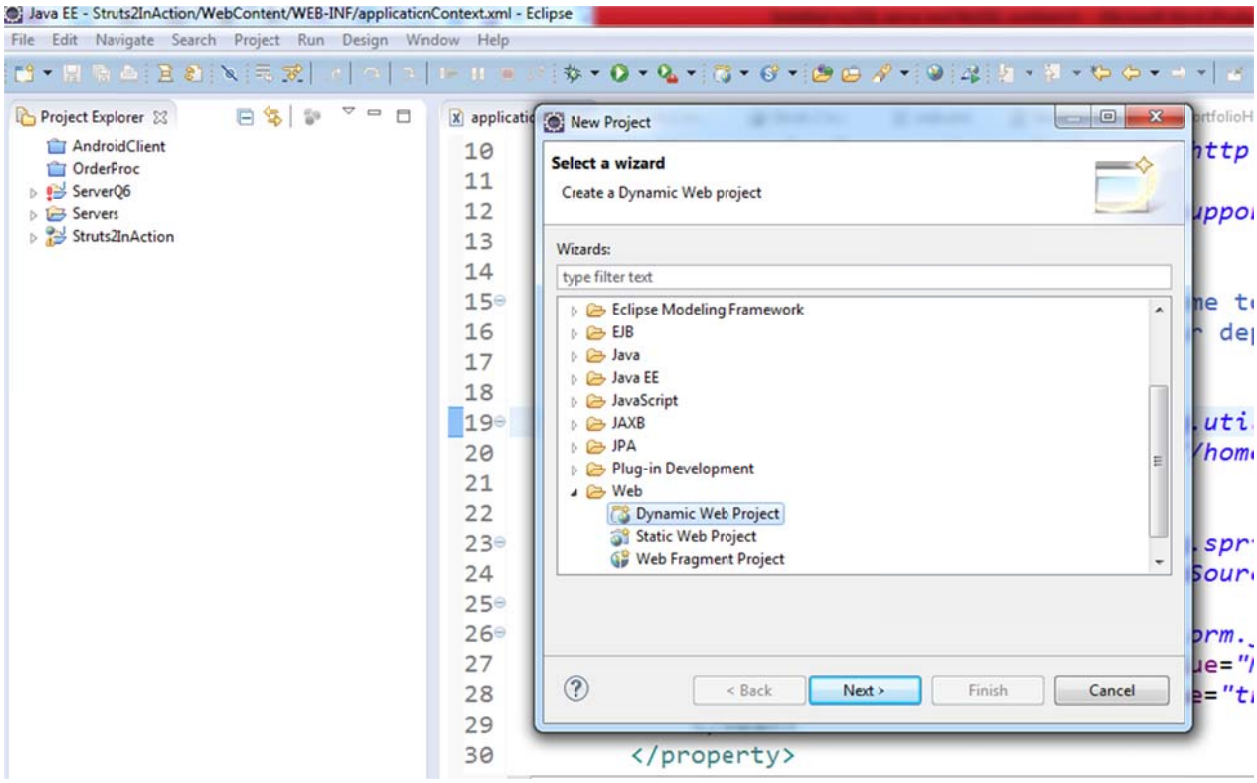
Click Apply

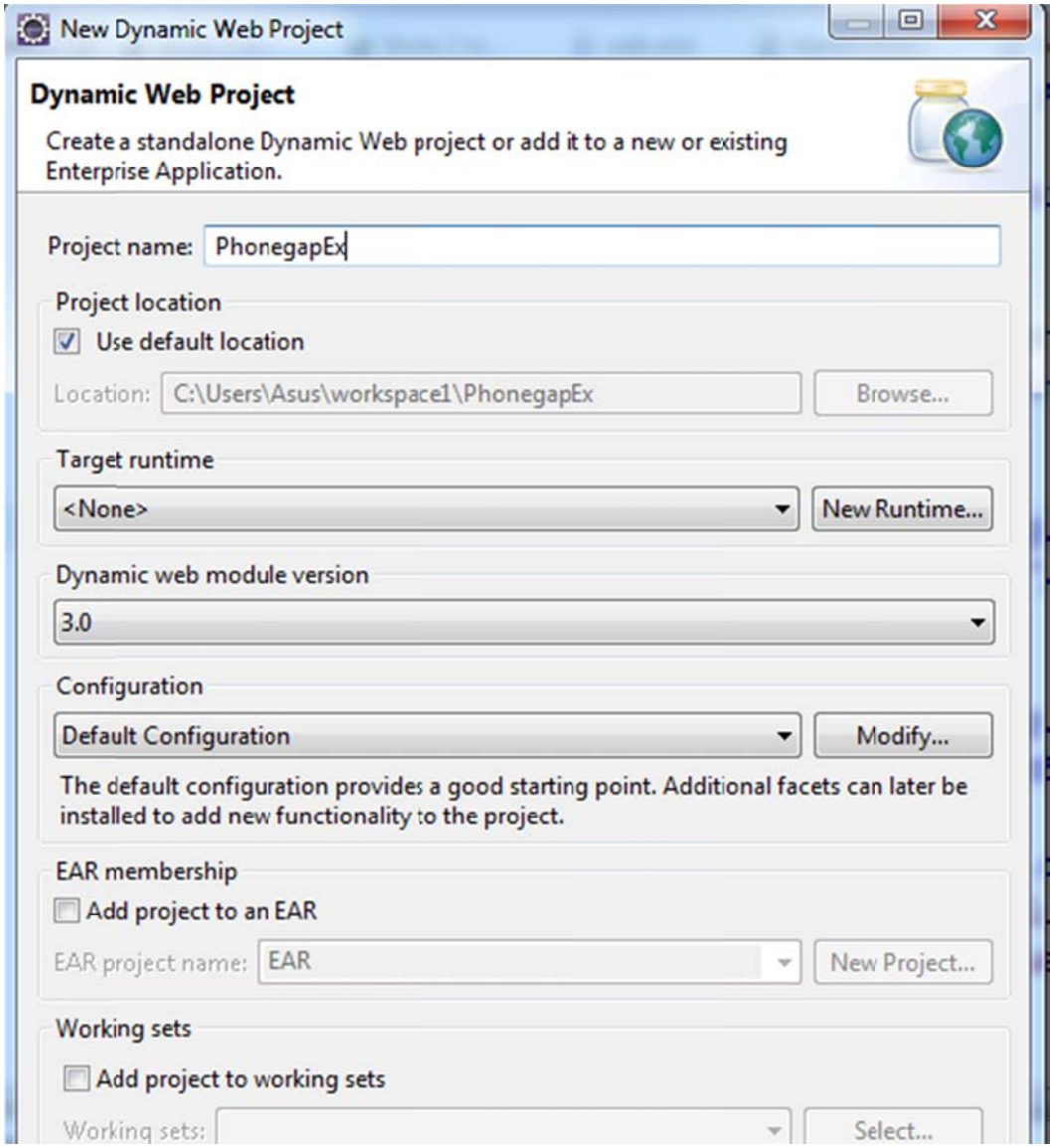


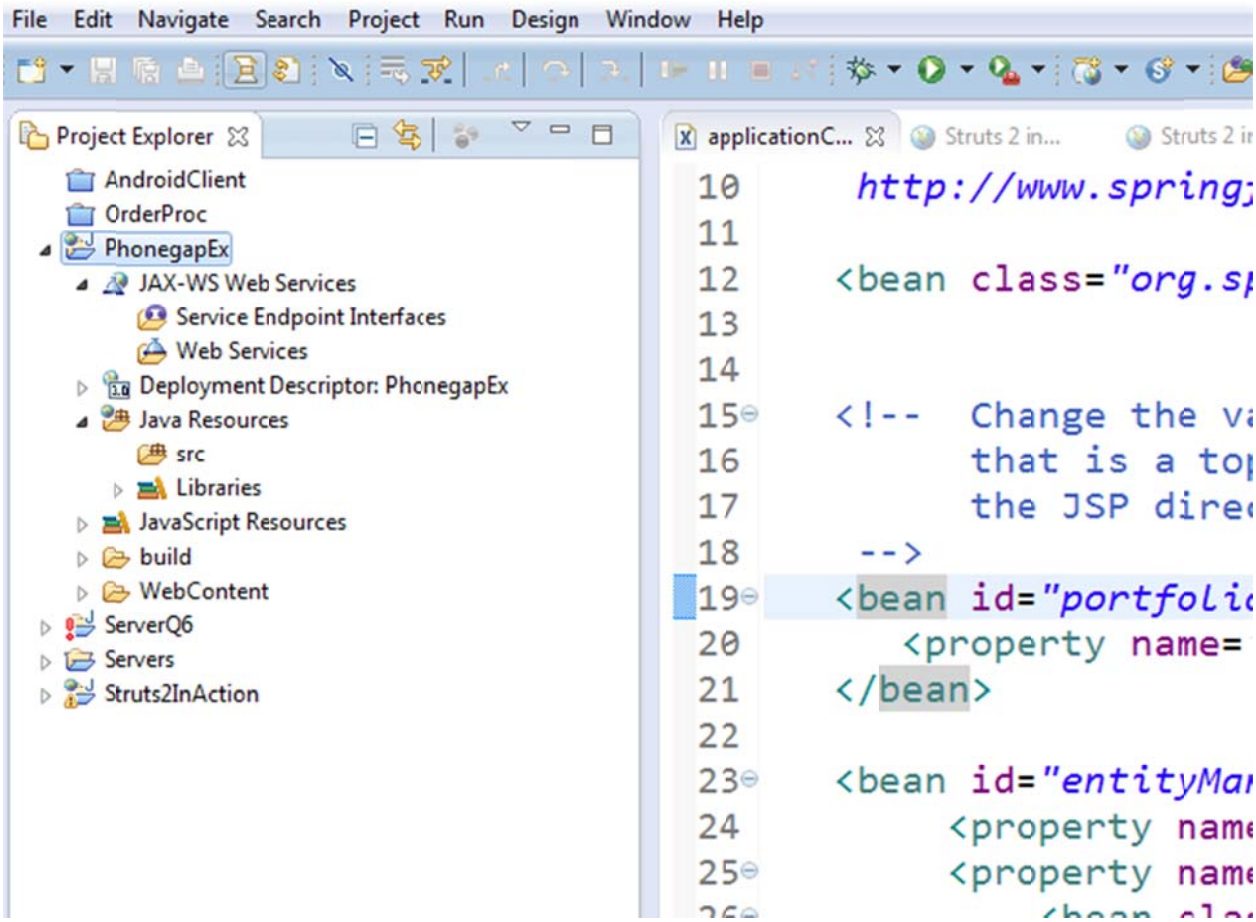
Now we got the following table.



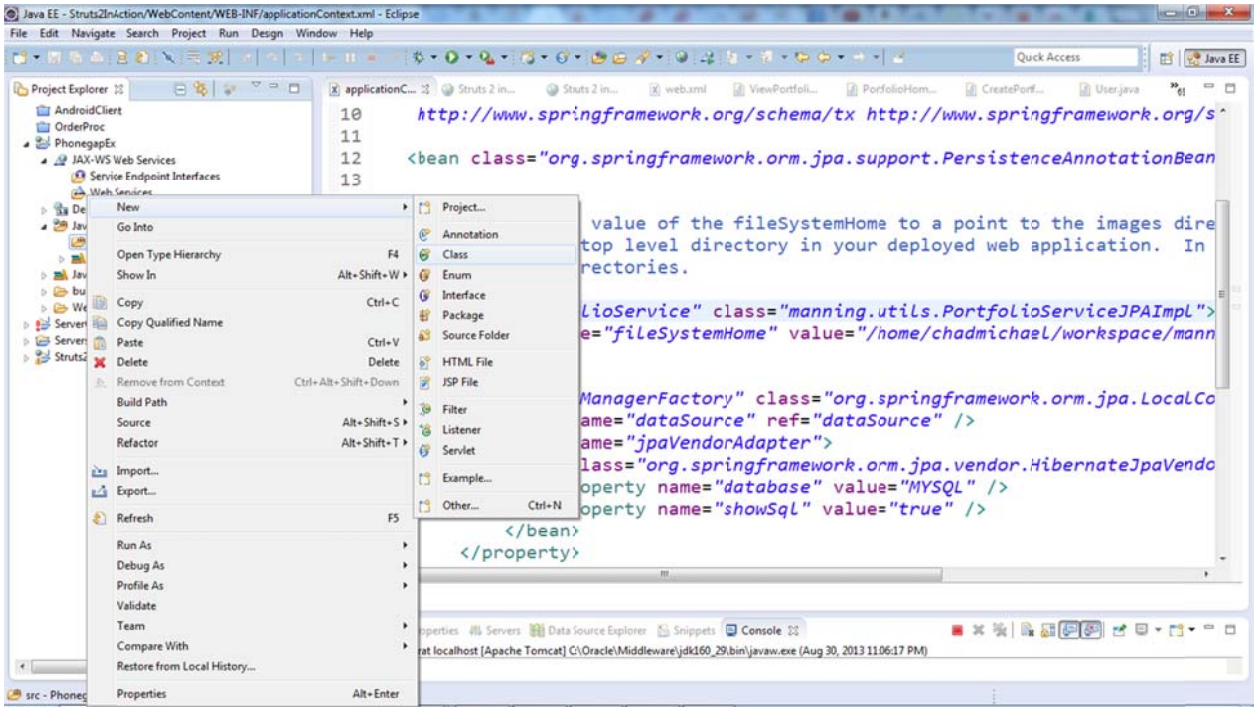
Write some java code to manipulate our database.


Java EE - Struts2InAction/WebContent/WEB-INF/applicationContext.xml - Eclipse
File Edit Navigate Search Project Run Design Window Help
Project Explorer
AndroidClient
OrderProc
ServerQ6
Servers
Struts2InAction
New Project
Select a wizard
Create a Dynamic Web project
Wizards:
type filter text
Eclipse Modeling Framework
EJB
Java
Java EE
JavaScript
JAXB
JPA
Plug-in Development
Web
Dynamic Web Project
Static Web Project
Web Fragment Project
< Back
Next >
Finish
Cancel
</property>

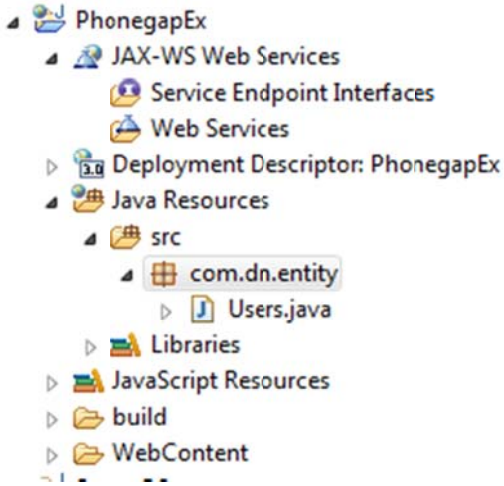

New Dynamic Web Project
Dynamic Web Project
Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.
Project name: PhonegapEx
Project location
Use default location
Location: C:\Users\Asus\workspace1\PhonegapEx
Browse...
Target runtime
<None>
New Runtime...
Dynamic web module version
3.0
Configuration
Default Configuration
Modify...
The default configuration provides a good starting point. Additional facets can later be installed to add new functionality to the project.
EAR membership
Add project to an EAR
EAR project name: EAR
New Project...
Working sets
Add project to working sets
Working sets:
Select...



Click on “src” create new class

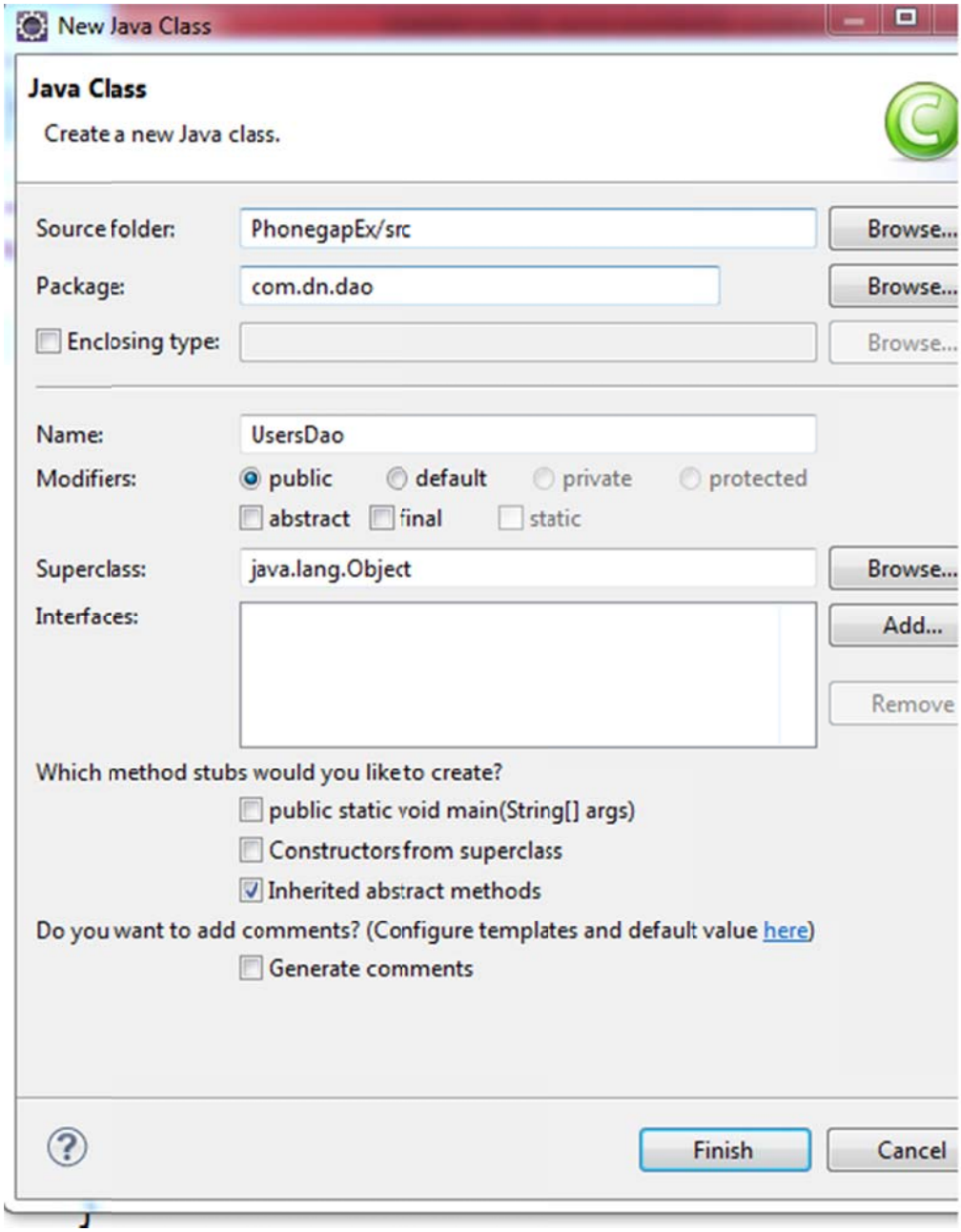


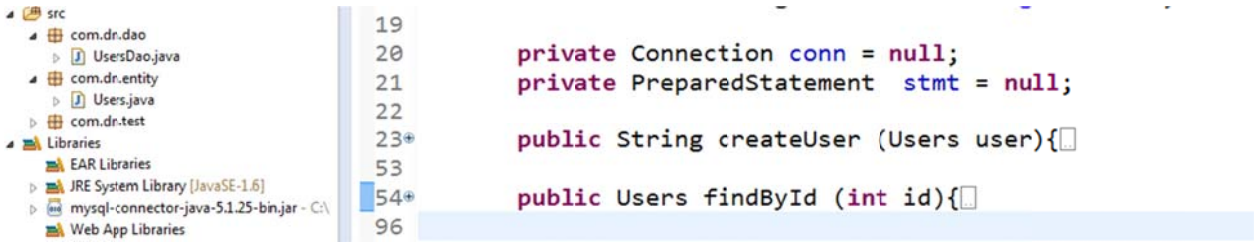
Enter Name class : Users and package” com.xxx.xxx”



```java
public class Users {
    private int id;
    private String firstName;
    private String lastName;
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getFirstName() {
        return firstName;
    }
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }
    public String getLastName() {
        return lastName;
    }
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
```

Create Dao (data access object) by using jdbc low level (this part can be replaced by jdbc template or Hibernate or so on).





The UsersDao has 2 methods.

createUser: user to create a user to database. Return “yes” if save to database successfully; “no” if fails.

findById: find a user by id.

Source code

```
 1 package com.dn.dao;
 2
 3 import java.sql.Connection;
 4 import java.sql.DriverManager;
 5 import java.sql.PreparedStatement;
 6 import java.sql.ResultSet;
 7 import java.sql.SQLException;
 8
 9 import com.dn.entity.Users;
10
11 public class UsersDao {
12     // JDBC driver name and database URL
13         static final String JDBC_DRIVER = "com.mysql.jdbc.Driver";
14         static final String DB_URL = "jdbc:mysql://localhost/phonggapDB";
15
16         //  Database credentials
17         static final String USER = "root";
18         static final String PASS = "Ilovehightech123";
19
20         private Connection conn = null;
21         private PreparedStatement  stmt = null;
22
```

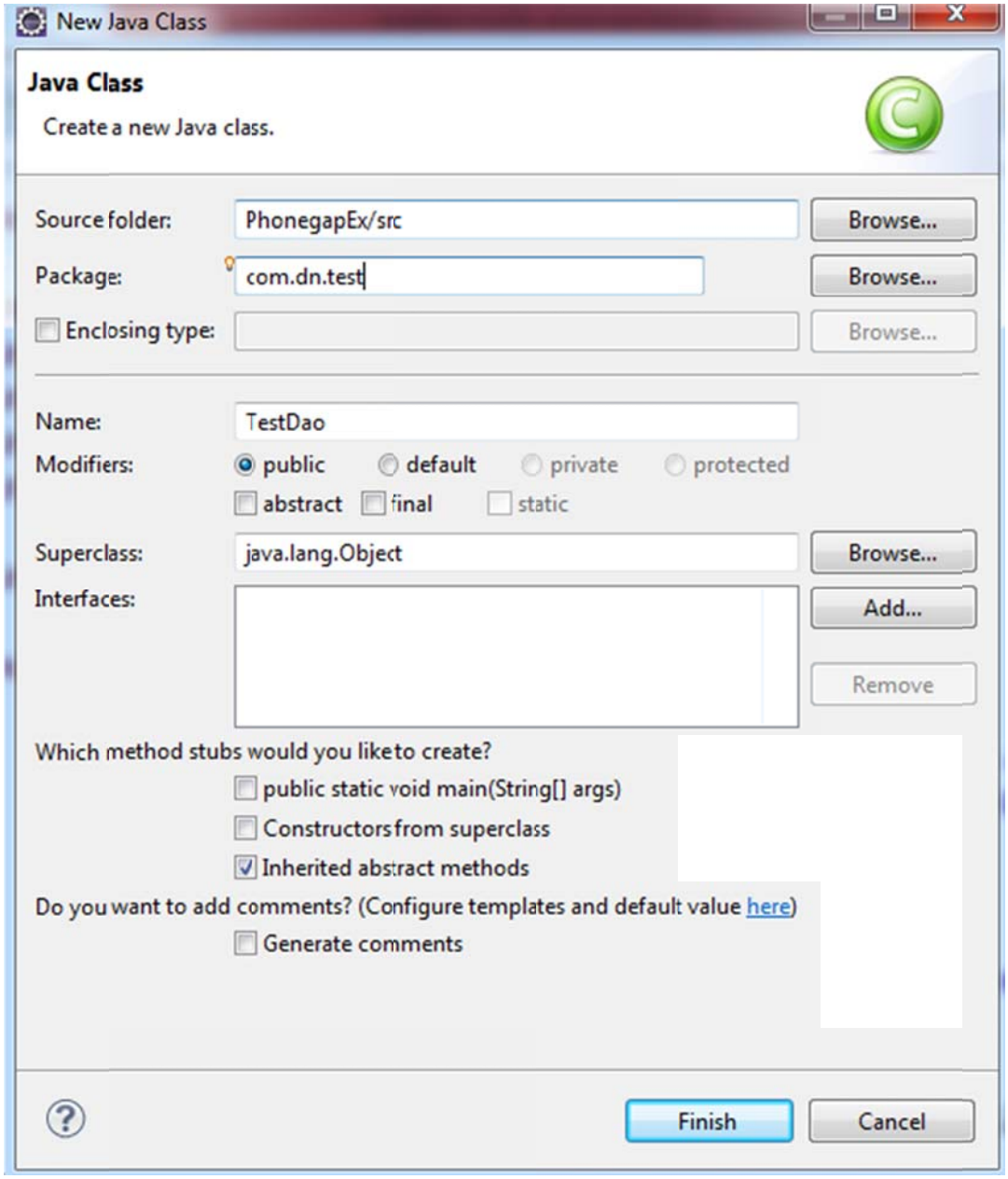
```
23       public String createUser (Users user){
24
25           try{
26               Class.forName("com.mysql.jdbc.Driver");
27
28               //Open a connection
29               conn = DriverManager.getConnection(DB_URL,USER,PASS);
30
31               //Execute a sql command
32               String sql;
33               sql = "INSERT INTO Users (firstName, lastName) VALUES (?,?);";
34               stmt = conn.prepareStatement(sql);
35               stmt.setString(1,user.getFirstName());
36               stmt.setString(2,user.getLastName());
37               stmt.executeUpdate();
38
39               //close
40               stmt.close();
41               conn.close();
42
43               return "yes";
```

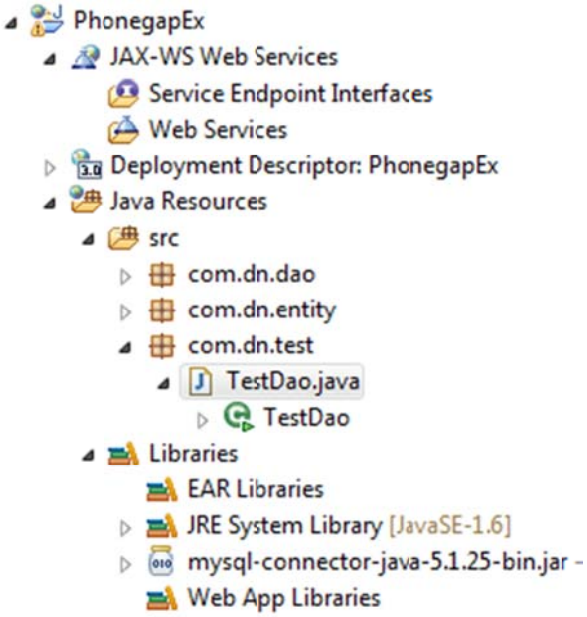
```
44
45              }
46              catch(SQLException se){
47                  return "no";
48              }
49              catch(Exception e){
50                  return "no";
51              }
52          }
53
54          public Users findById (int id){
55              Users user = new Users();
56              try{
57                  Class.forName("com.mysql.jdbc.Driver");
58
59                  //Oper a connection
60                  conn = DriverManager.getConnection(DB_URL,USER,PASS);
61
62                  //Execute a sql command
63                  String sql;
64                  sql = "SELECT * FROM Users WHERE id=?";
```

```
65                  stmt = conn.prepareStatement(sql);
66                  stmt.setInt(1,id);
67
68                  ResultSet rs = stmt.executeQuery();
69
70                  //Extract data from result set
71                  while(rs.next()){
72                      //Retrieve by column name
73                      user.setFirstName(rs.getString("firstName"));
74                      user.setLastName(rs.getString("lastName"));
75                      user.setId(id);
76                  }
77
78                  //close
79                  rs.close();
80                  stmt.close();
81                  conn.close();
82
83                  return user;
84
85              }
```

```
86              catch(SQLException se){
87                  return null;
88              }
89              catch(Exception e){
90                  return null;
91              }
92          }
93
94  }
```

Do a little test with UsersDao class by using simple main method not JUnit.

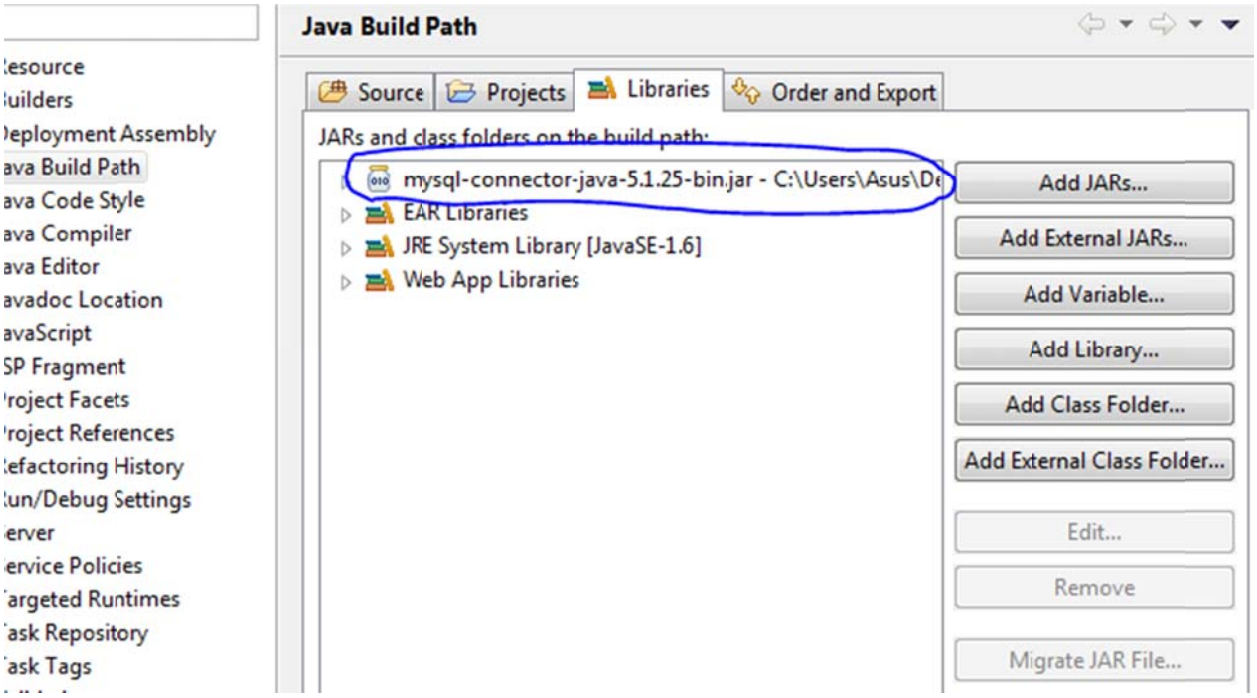

New Java Class
Java Class
Create a new Java class.
Source folder:
PhonegapEx/src
Browse...
Package:
com.dn.test
Browse...
Enclosing type:
Browse...
Name:
TestDao
Modifiers:
public
default
private
protected
abstract
final
static
Superclass:
java.lang.Object
Browse...
Interfaces:
Add...
Remove
Which method stubs would you like to create?
public static void main(String[] args)
Constructors from superclass
Inherited abstract methods
Do you want to add comments? (Configure templates and default value here)
Generate comments
Finish
Cancel



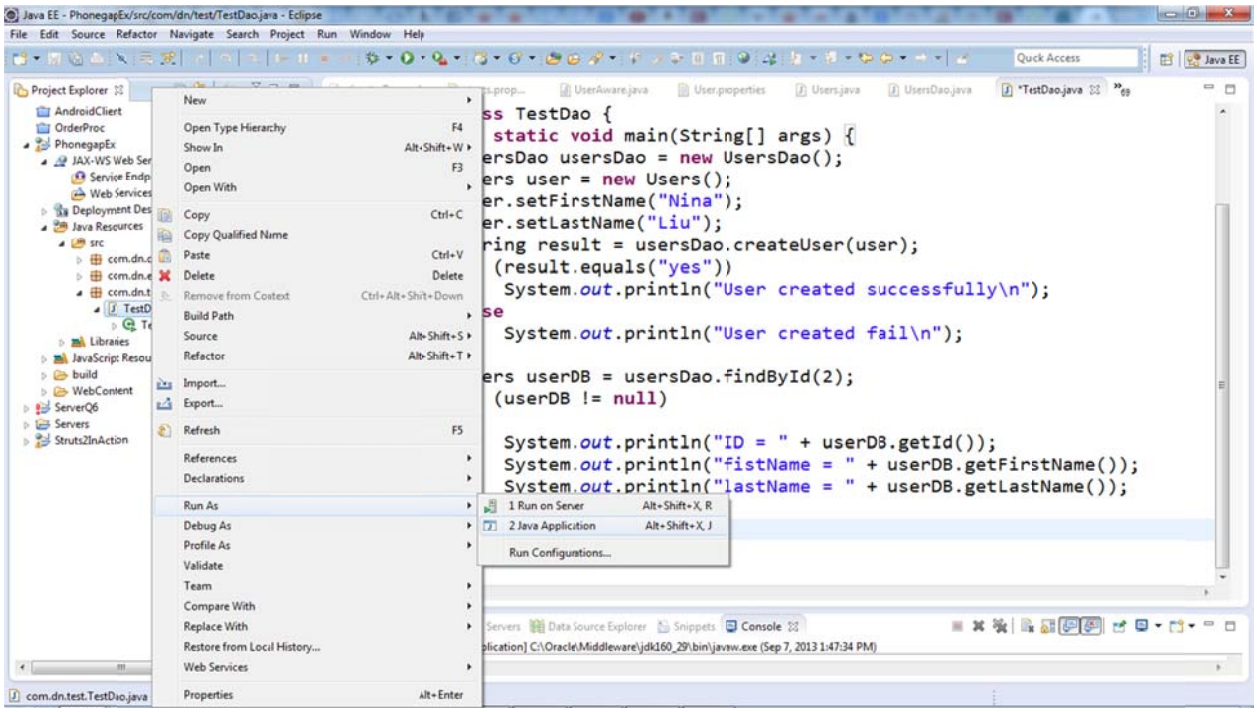
Source code

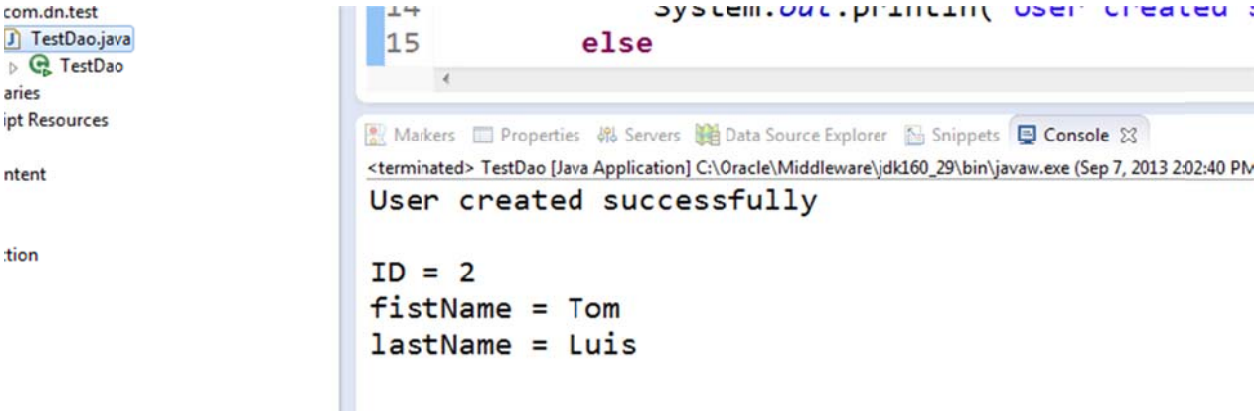
```
1  package com.dn.test;
2
3  import com.dn.dao.UsersDao;
4  import com.dn.entity.Users;
5
6  public class TestDao {
7      public static void main(String[] args) {
8          UsersDao usersDao = new UsersDao();
9          Users user = new Users();
10         user.setFirstName("Nina");
11         user.setLastName("Liu");
12         String result = usersDao.createUser(user);
13         if (result.equals("yes"))
14             System.out.println("User created successfully\n");
15         else
16             System.out.println("User created fail\n");
17
18         Users userDB = usersDao.findById(2);
19         if (userDB != null)
20         {
21             System.out.println("ID = " + userDB.getId());
22             System.out.println("fistName = " + userDB.getFirstName());
23             System.out.println("lastName = " + userDB.getLastName());
24         }
25     }
26 }
27
```

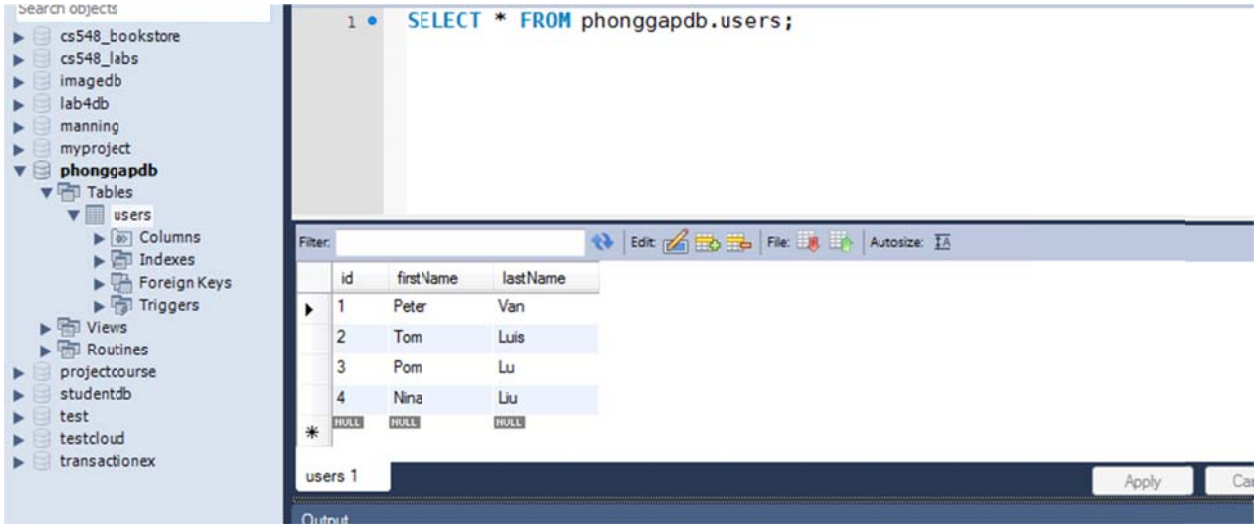
Remember: we need to add mysql-connector-java-xxx to the library



Click on TestDao and run



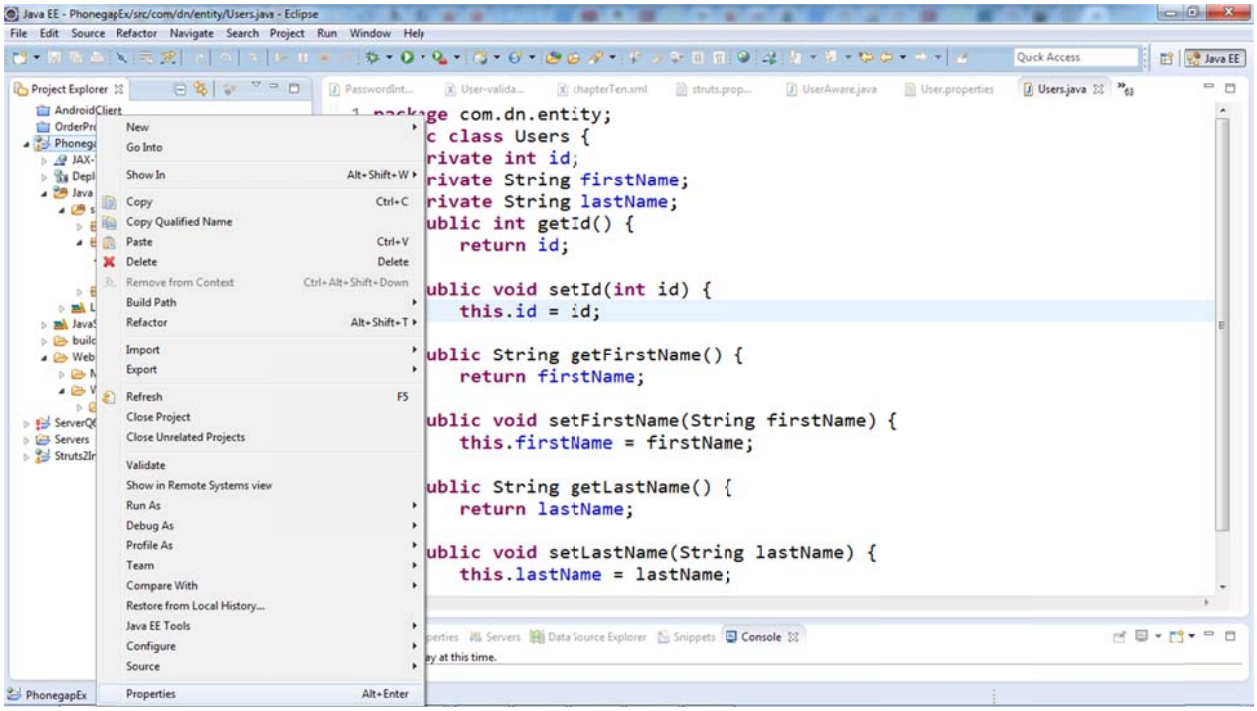


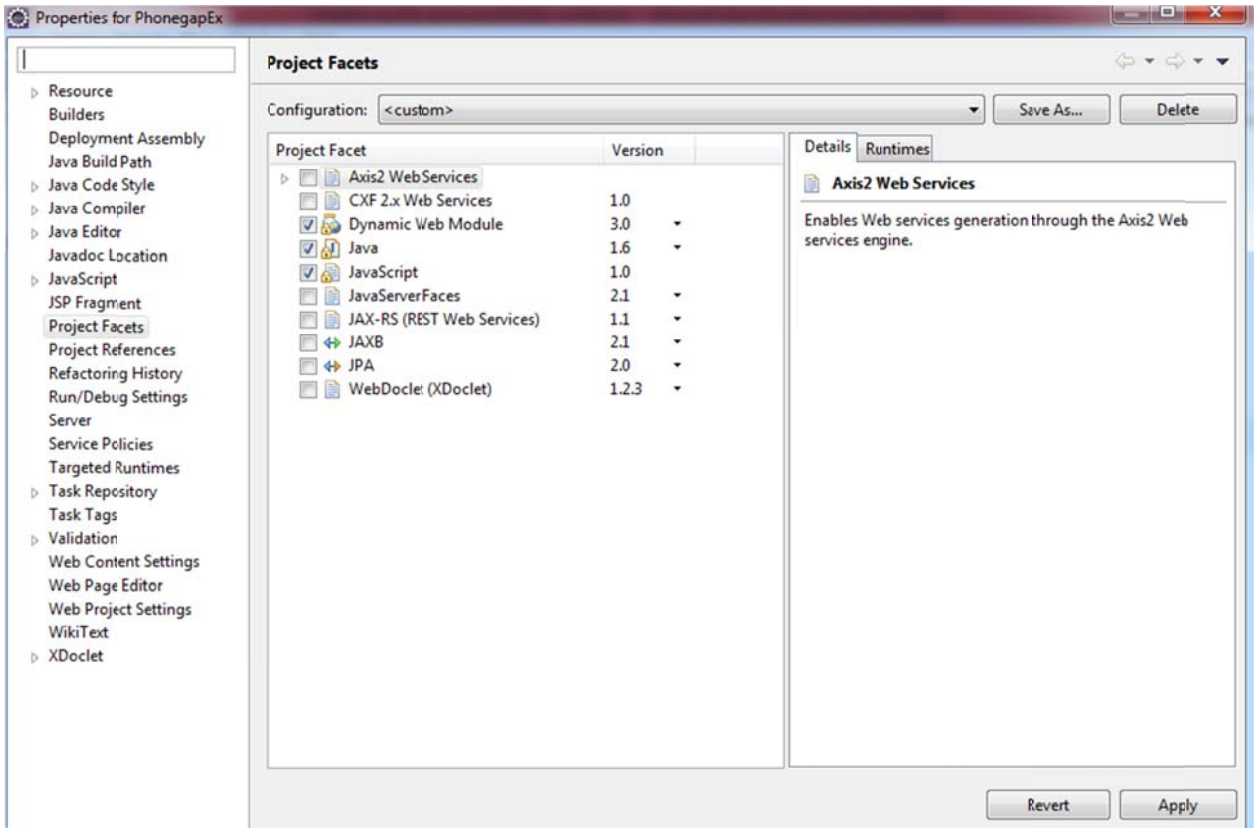


From mysql workbench, we see the user(Nina, Liu) is saved to database and findById(2) gave the right result.

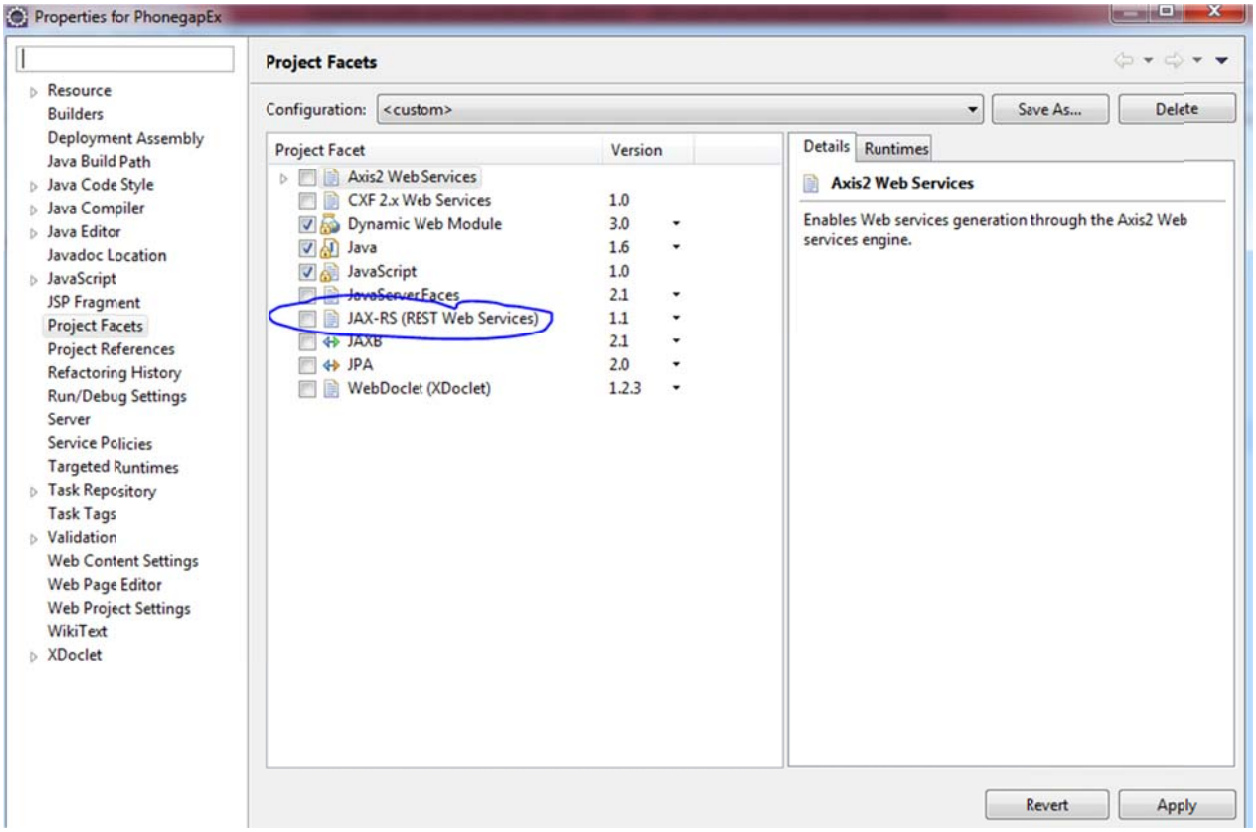
Next create a little Rest web service which wraps around UsersDao.

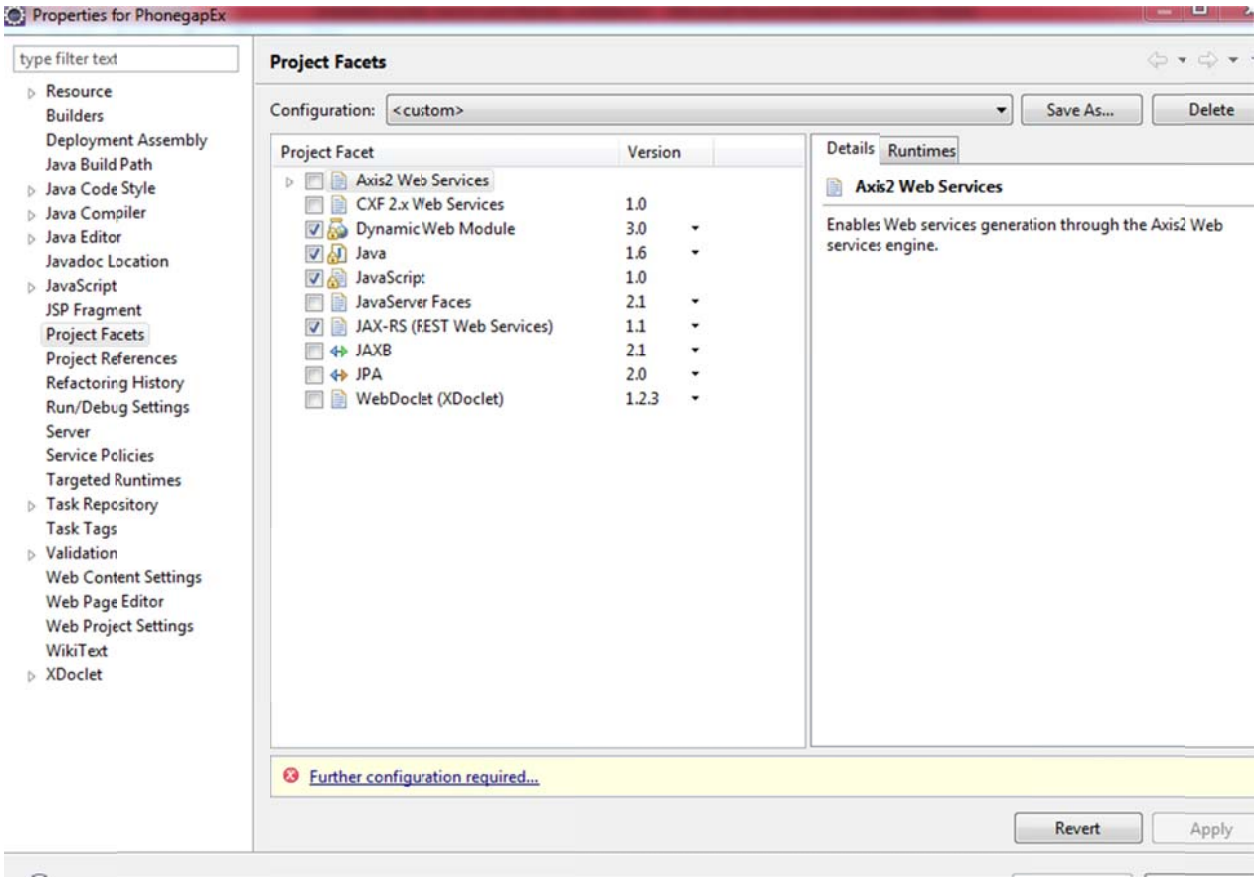
Right click on project and click Properties.


Java EE - PhonegapEx/src/com/dn/entity/Users.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
PhonegapEx
New
Go Into
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Refactor Alt+Shift+T
Import
Export
Refresh F5
Close Project
Close Unrelated Projects
Validate
Show in Remote Systems view
Run As
Debug As
Profile As
Team
Compare With
Restore from Local History...
Java EE Tools
Configure
Source
Properties Alt+Enter
Users.java
ge com.dn.entity;
c class Users {
rivate int id;
rivate String firstName;
rivate String lastName;
ublic int getId() {
return id;
ublic void setId(int id) {
this.id = id;
ublic String getFirstName() {
return firstName;
ublic void setFirstName(String firstName) {
this.firstName = firstName;
ublic String getLastName() {
return lastName;
ublic void setLastName(String lastName) {
this.lastName = lastName;
Console

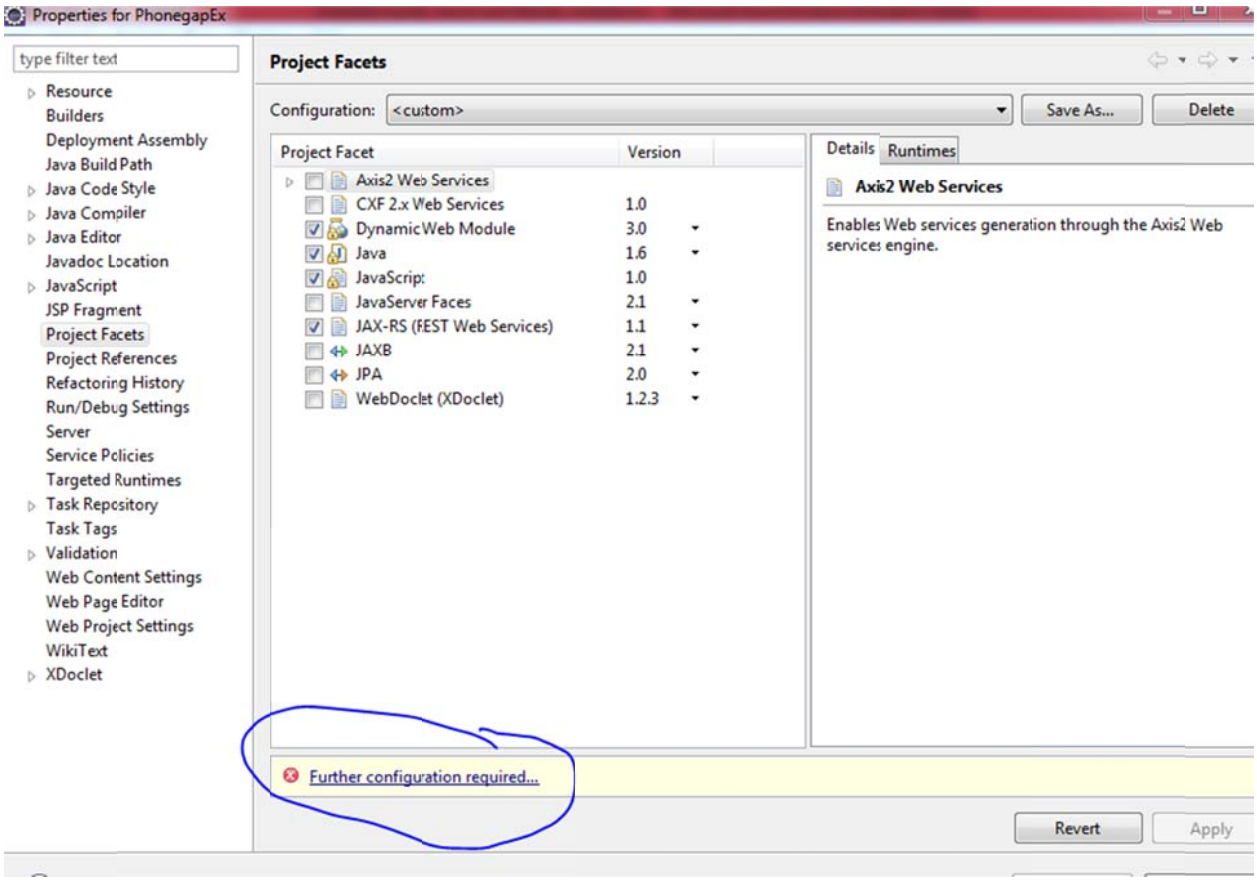

Properties for PhonegapEx
Resource
Builders
Deployment Assembly
Java Build Path
Java Code Style
Java Compiler
Java Editor
Javadoc Location
JavaScript
JSP Fragment
Project Facets
Project References
Refactoring History
Run/Debug Settings
Server
Service Policies
Targeted Runtimes
Task Repository
Task Tags
Validation
Web Content Settings
Web Page Editor
Web Project Settings
WikiText
XDoclet
Project Facets
Configuration: <custom>
Save As...
Delete
Project Facet
Version
Axis2 WebServices
CXF 2.x Web Services 1.0
Dynamic Web Module 3.0
Java 1.6
JavaScript 1.0
JavaServerFaces 2.1
JAX-RS (REST Web Services) 1.1
JAXB 2.1
JPA 2.0
WebDoclet (XDoclet) 1.2.3
Details
Runtimes
Axis2 Web Services
Enables Web services generation through the Axis2 Web services engine.
Revert
Apply

Click on JAX-RS

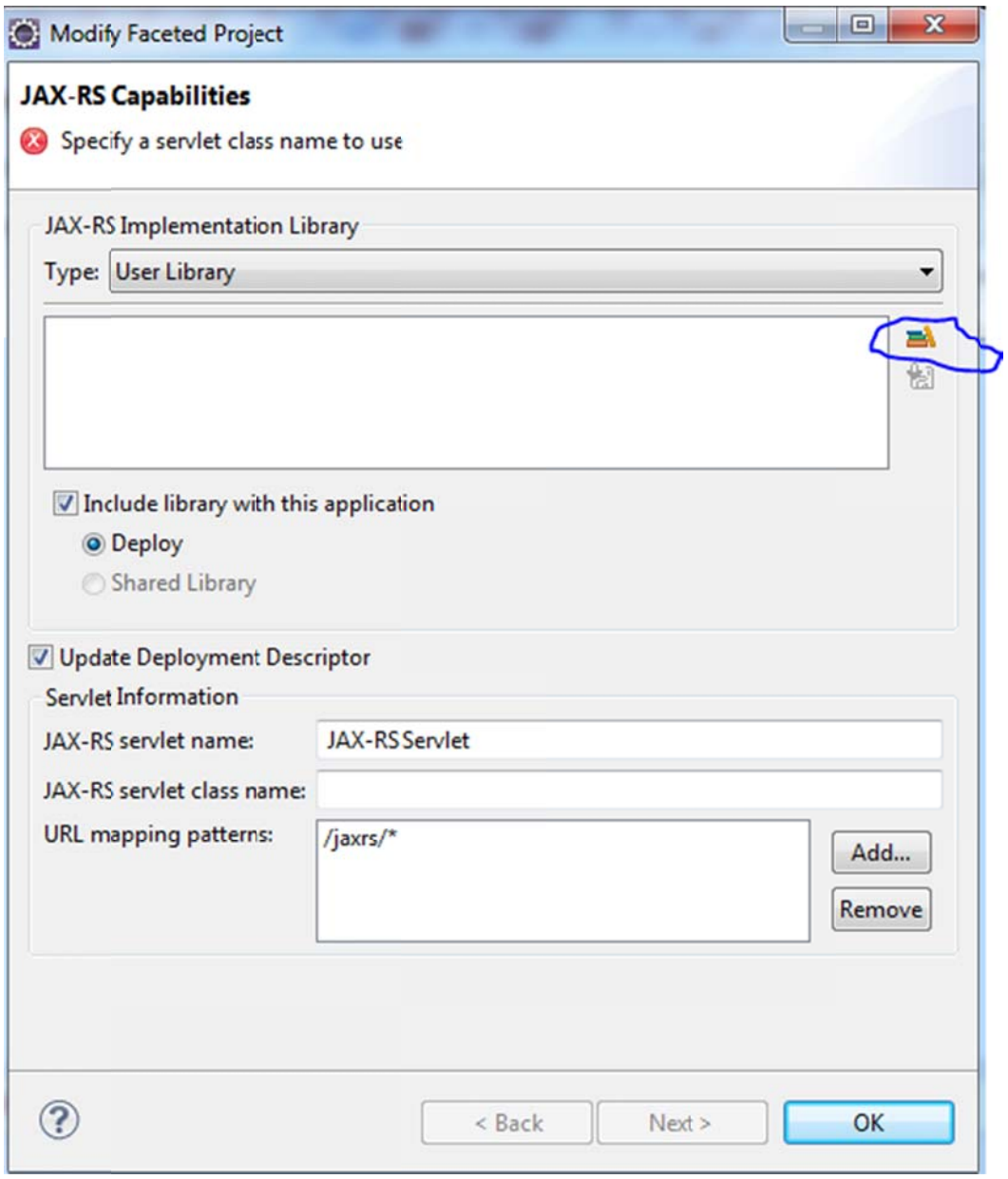


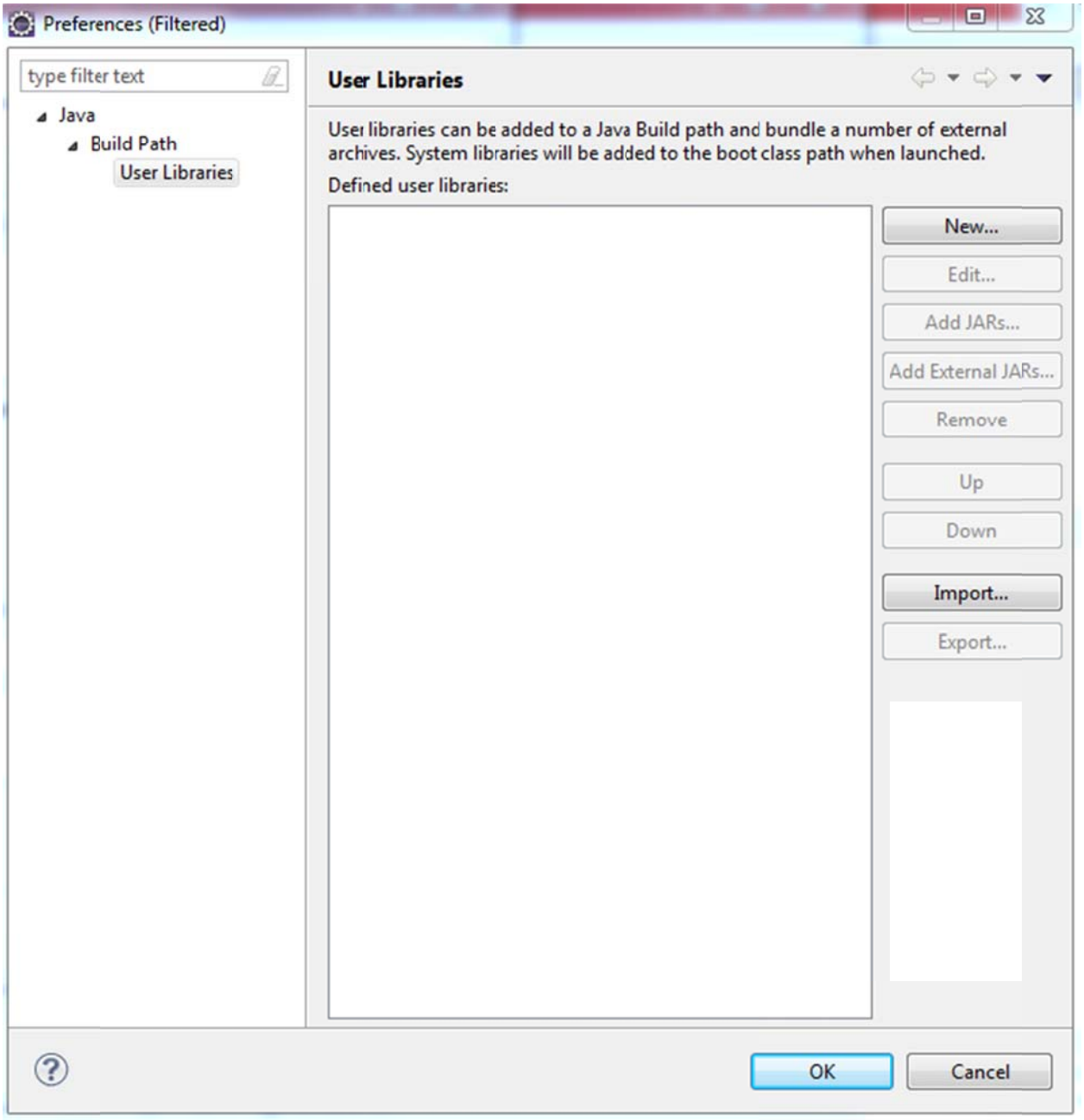


Click on further configuration



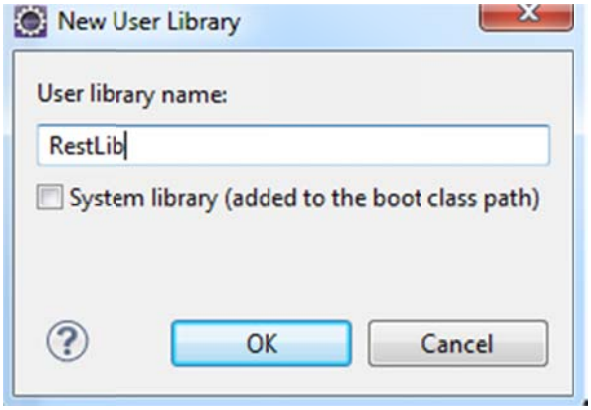
Click on Manage Library


Modify Faceted Project
JAX-RS Capabilities
Specify a servlet class name to use
JAX-RS Implementation Library
Type: User Library
Include library with this application
Deploy
Shared Library
Update Deployment Descriptor
Servlet Information
JAX-RS servlet name: JAX-RS Servlet
JAX-RS servlet class name:
URL mapping patterns: /jaxrs/*
Add...
Remove
< Back
Next >
OK

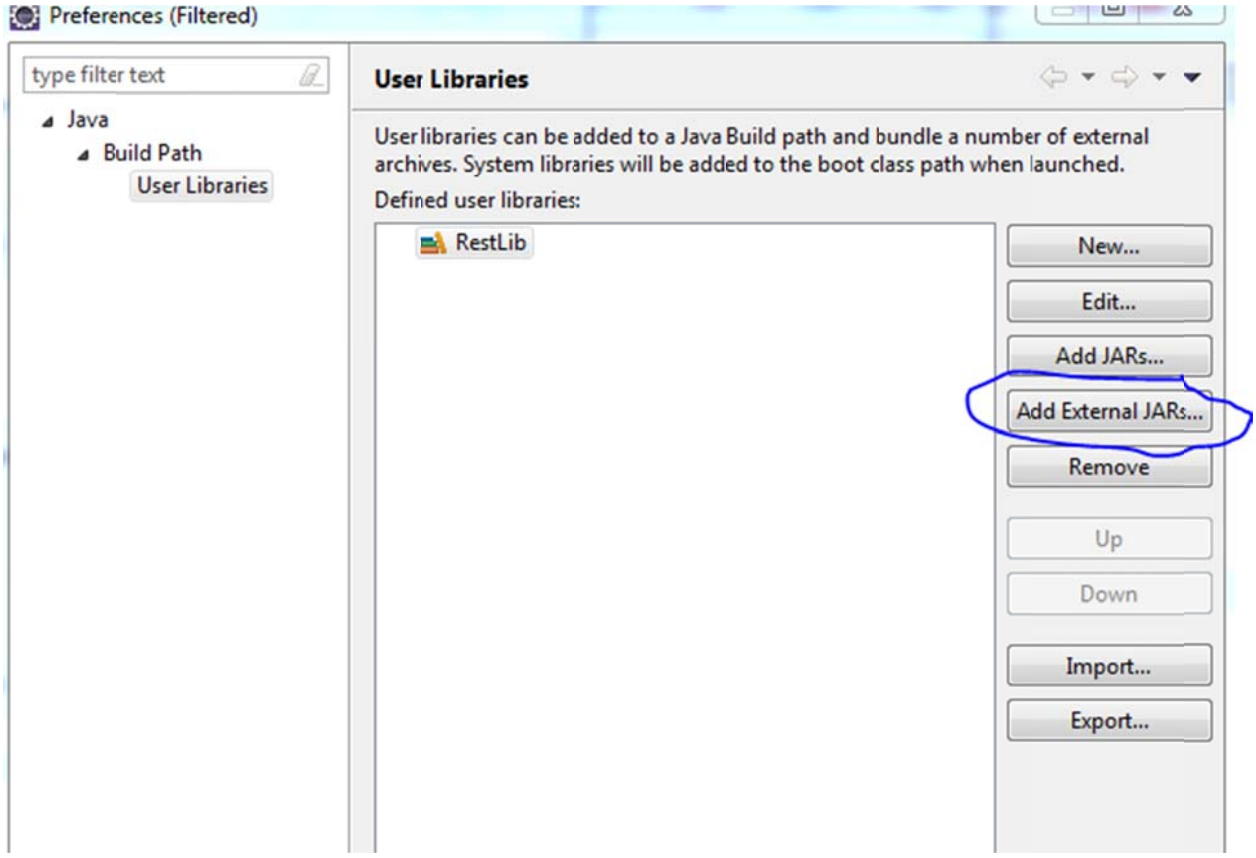


Click New

Give a name, For example RestLib

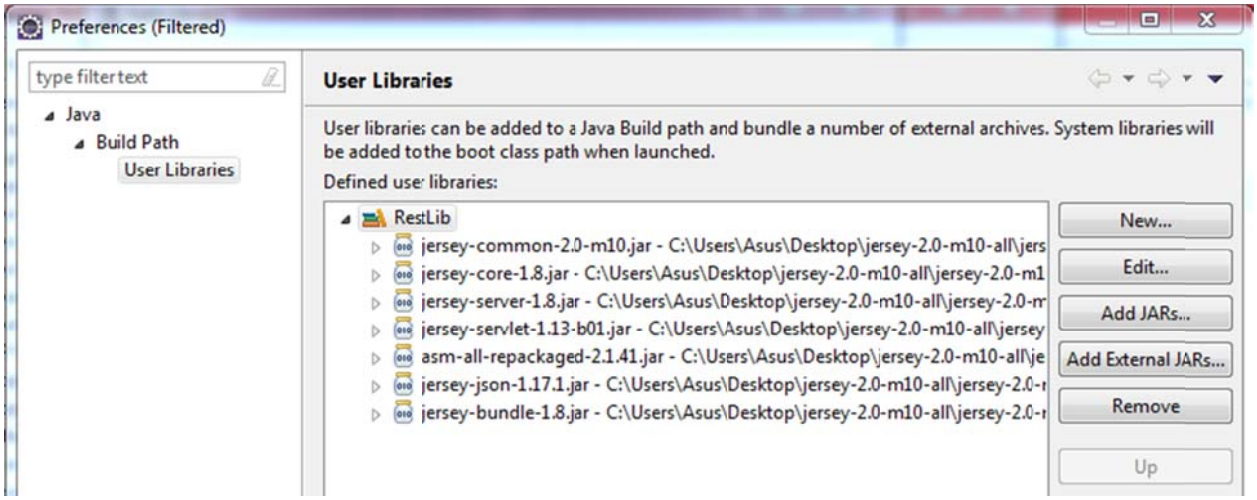


Click Ok.

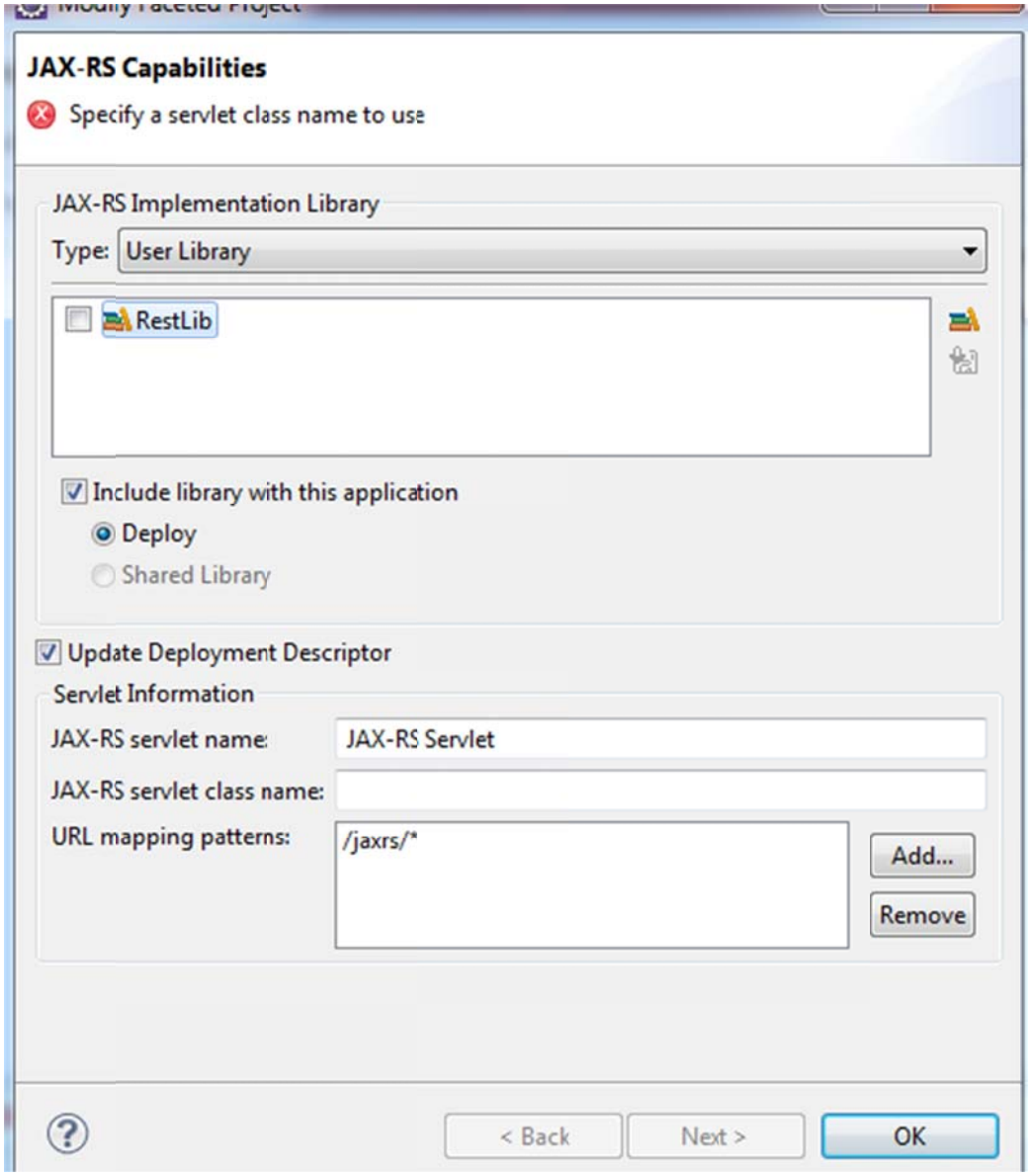


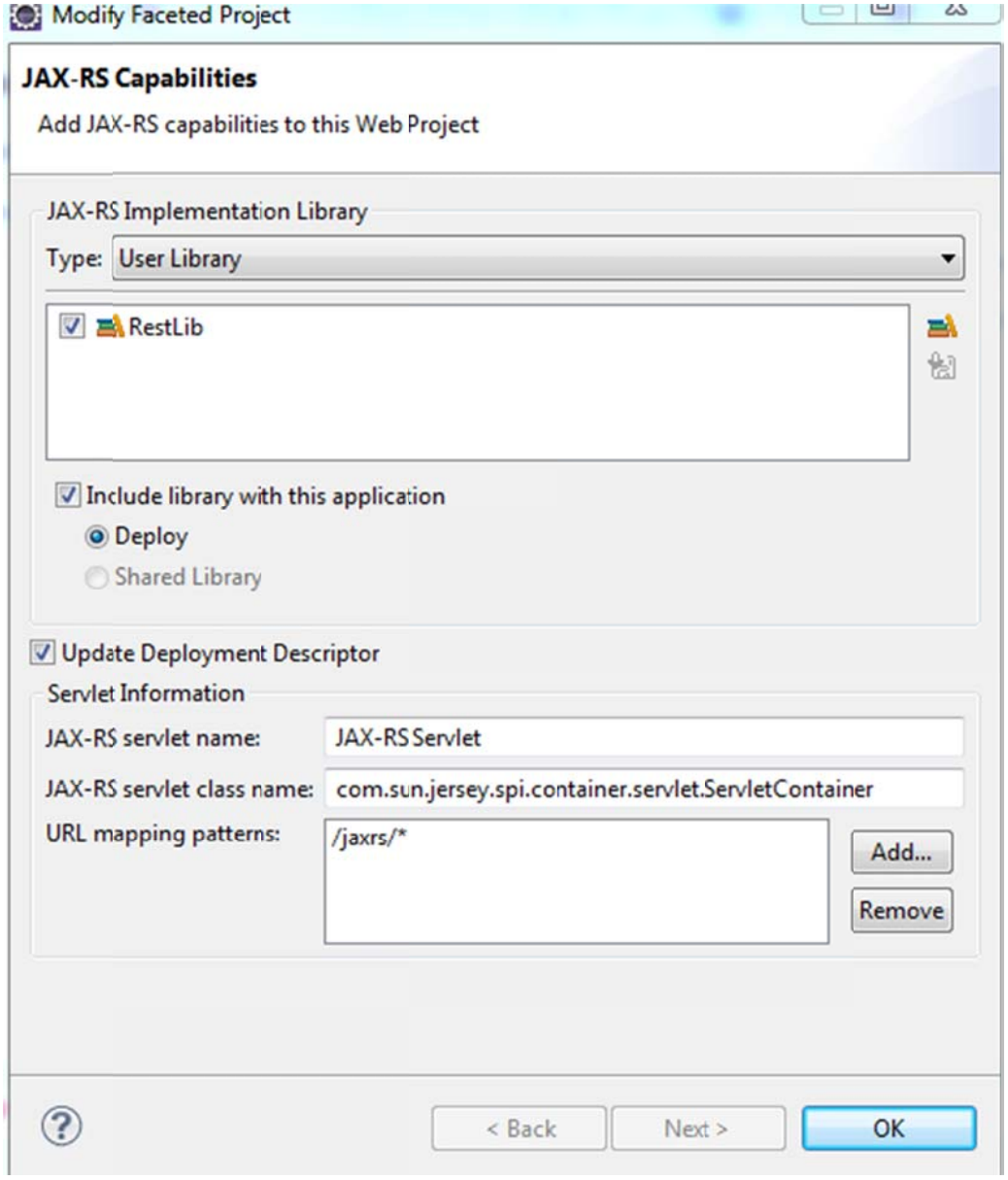
Click on Add external JARs

Add some necessary libraries

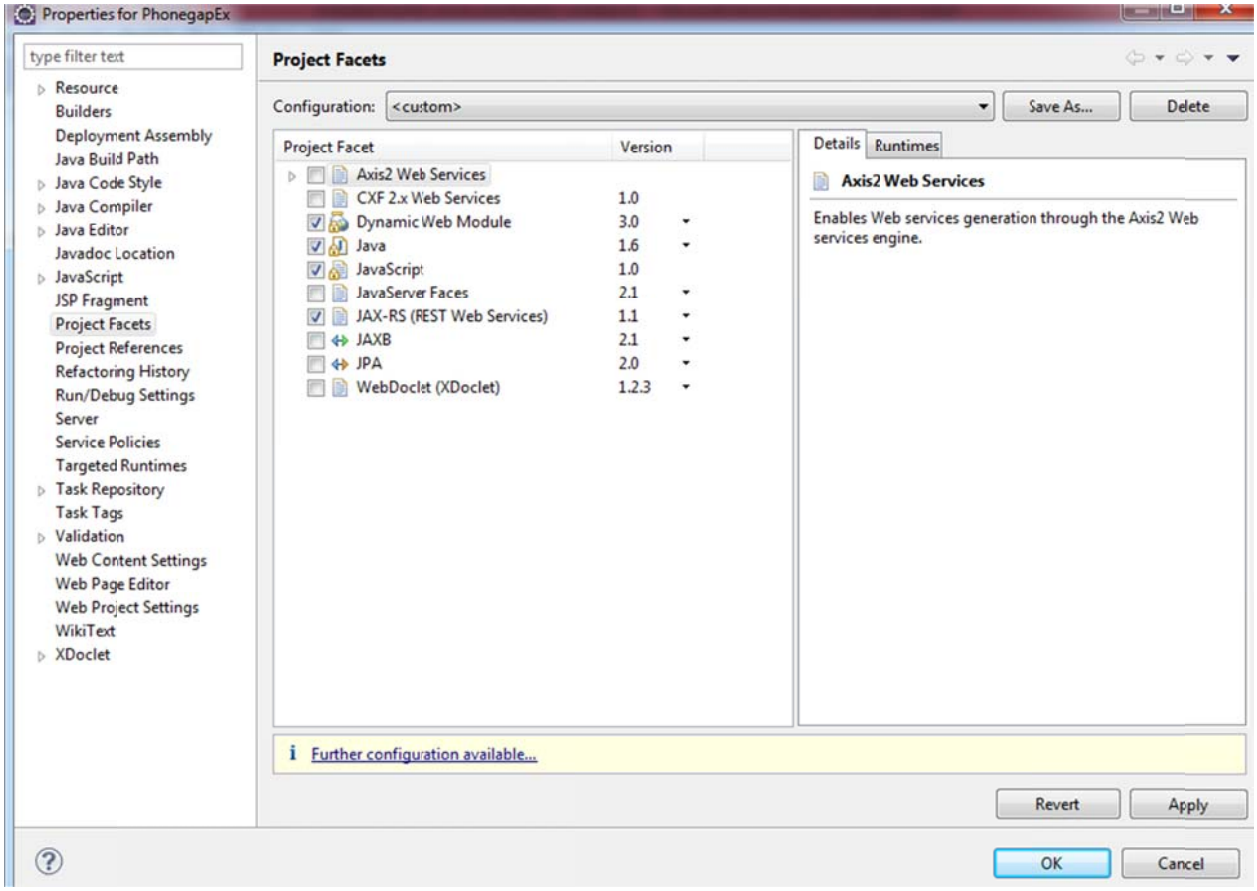

Preferences (Filtered)
type filter text
Java
Build Path
User Libraries
User Libraries
User libraries can be added to a Java Build path and bundle a number of external archives. System libraries will be added to the boot class path when launched.
Defined user libraries:
RestLib
jersey-common-2.0-m10.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\jers
jersey-core-1.8.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\jersey-2.0-m1
jersey-server-1.8.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\jersey-2.0-m
jersey-servlet-1.13-b01.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\jersey
asm-all-repackaged-2.1.41.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\je
jersey-json-1.17.1.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\jersey-2.0-
jersey-bundle-1.8.jar - C:\Users\Asus\Desktop\jersey-2.0-m10-all\jersey-2.0-
New...
Edit...
Add JARs...
Add External JARs...
Remove
Up

Click Ok

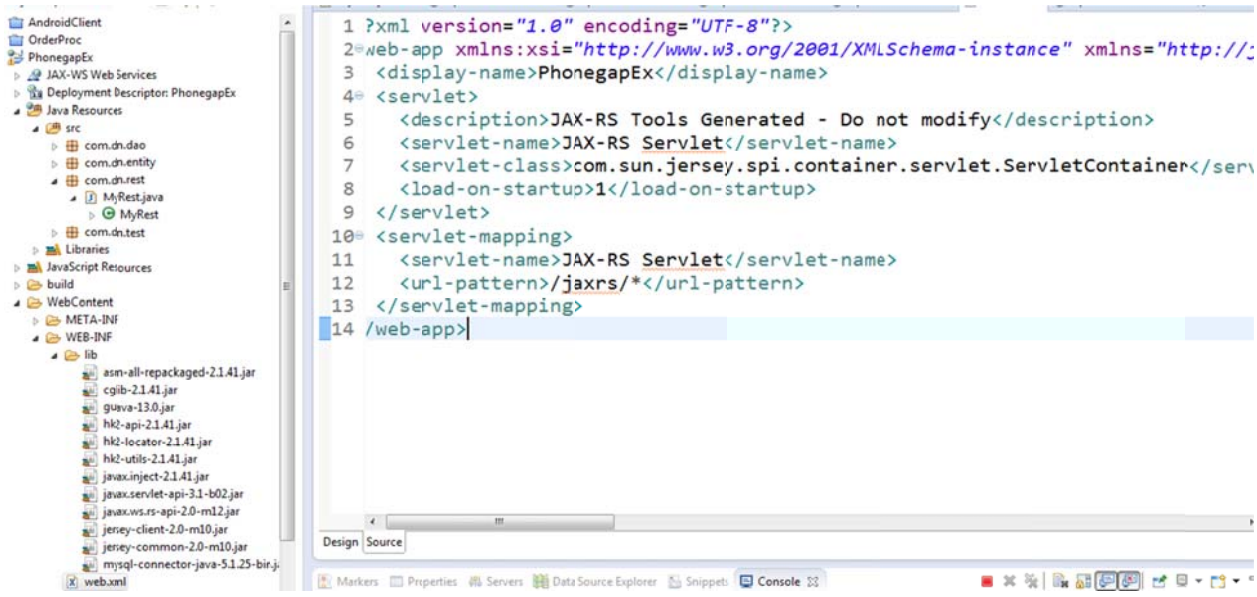

JAX-RS Capabilities
Specify a servlet class name to use
JAX-RS Implementation Library
Type: User Library
RestLib
Include library with this application
Deploy
Shared Library
Update Deployment Descriptor
Servlet Information
JAX-RS servlet name: JAX-RS Servlet
JAX-RS servlet class name:
URL mapping patterns: /jaxrs/*
Add...
Remove
< Back
Next >
OK

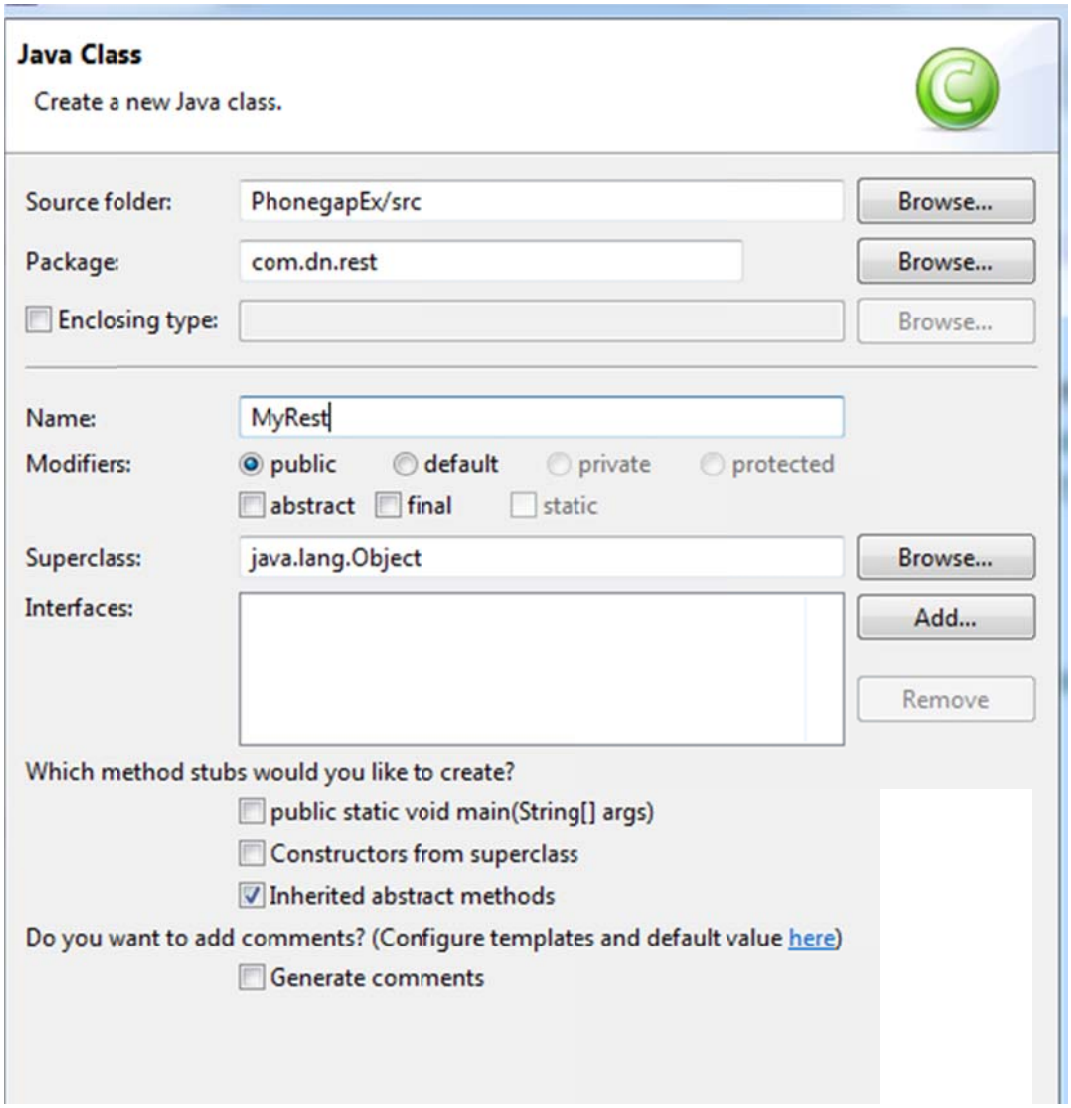

Modify Faceted Project
JAX-RS Capabilities
Add JAX-RS capabilities to this Web Project
JAX-RS Implementation Library
Type: User Library
RestLib
Include library with this application
Deploy
Shared Library
Update Deployment Descriptor
Servlet Information
JAX-RS servlet name: JAX-RS Servlet
JAX-RS servlet class name: com.sun.jersey.spi.container.servlet.ServletContainer
URL mapping patterns: /jaxrs/*
Add...
Remove
< Back
Next >
OK

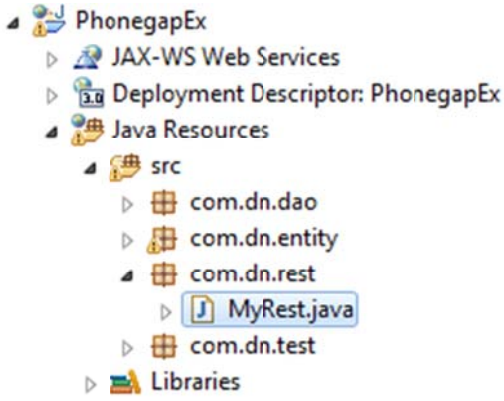
Click Ok



Click Ok. The web.xml is created as following



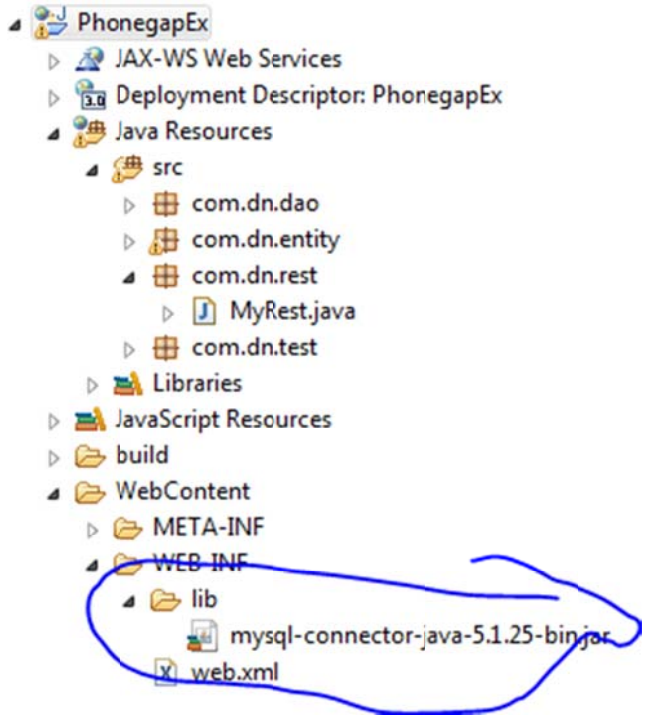




Put the following code to MyRest.java

```java
package com.dn.rest;

import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.PathParam;
import javax.ws.rs.Produces;

import com.dn.dao.UsersDao;
import com.dn.entity.Users;

@Path("/Dao")
public class MyRest {

	@GET
	@Path("/getUserById/{id}")
	@Produces({"application/xml", "application/json"})
	public Users getUser(@PathParam("id") int id) {
		UsersDao usersDao = new UsersDao();
		Users user = usersDao.findById(id);
		return user;
	}
}
```

Before run the project on Tomcat server, we need to copy mysql-connector-java-xxx to lib folder as following

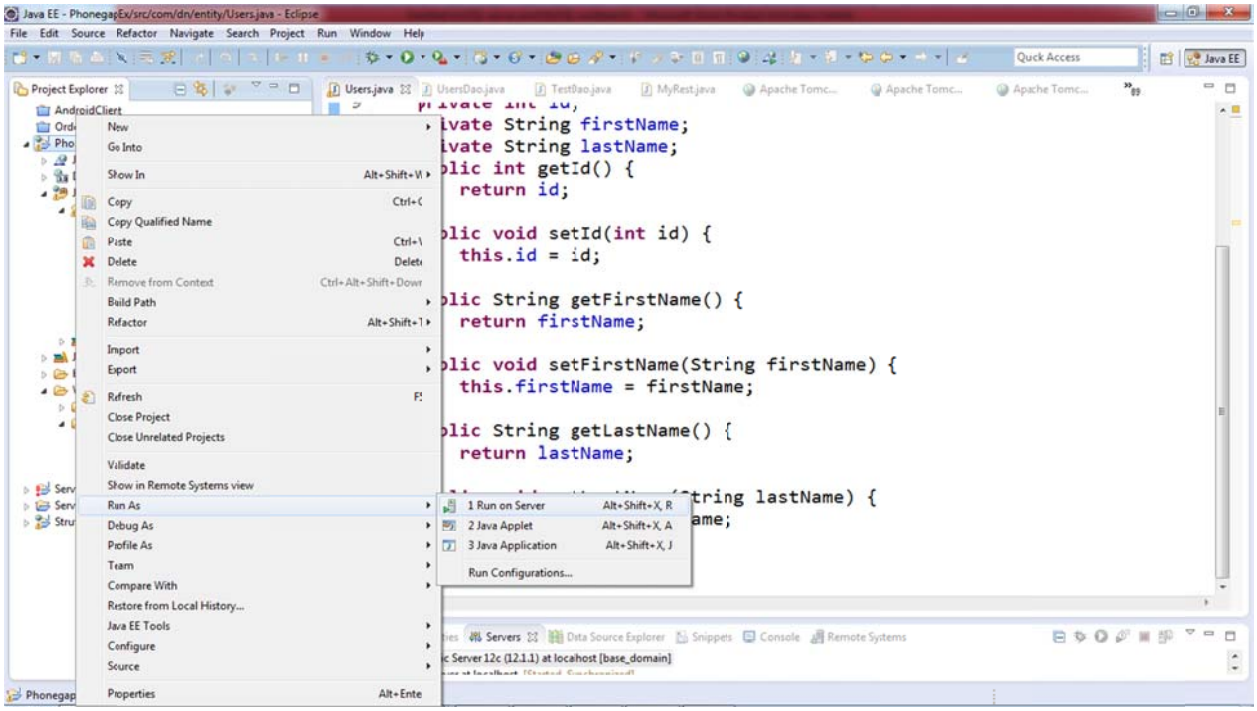


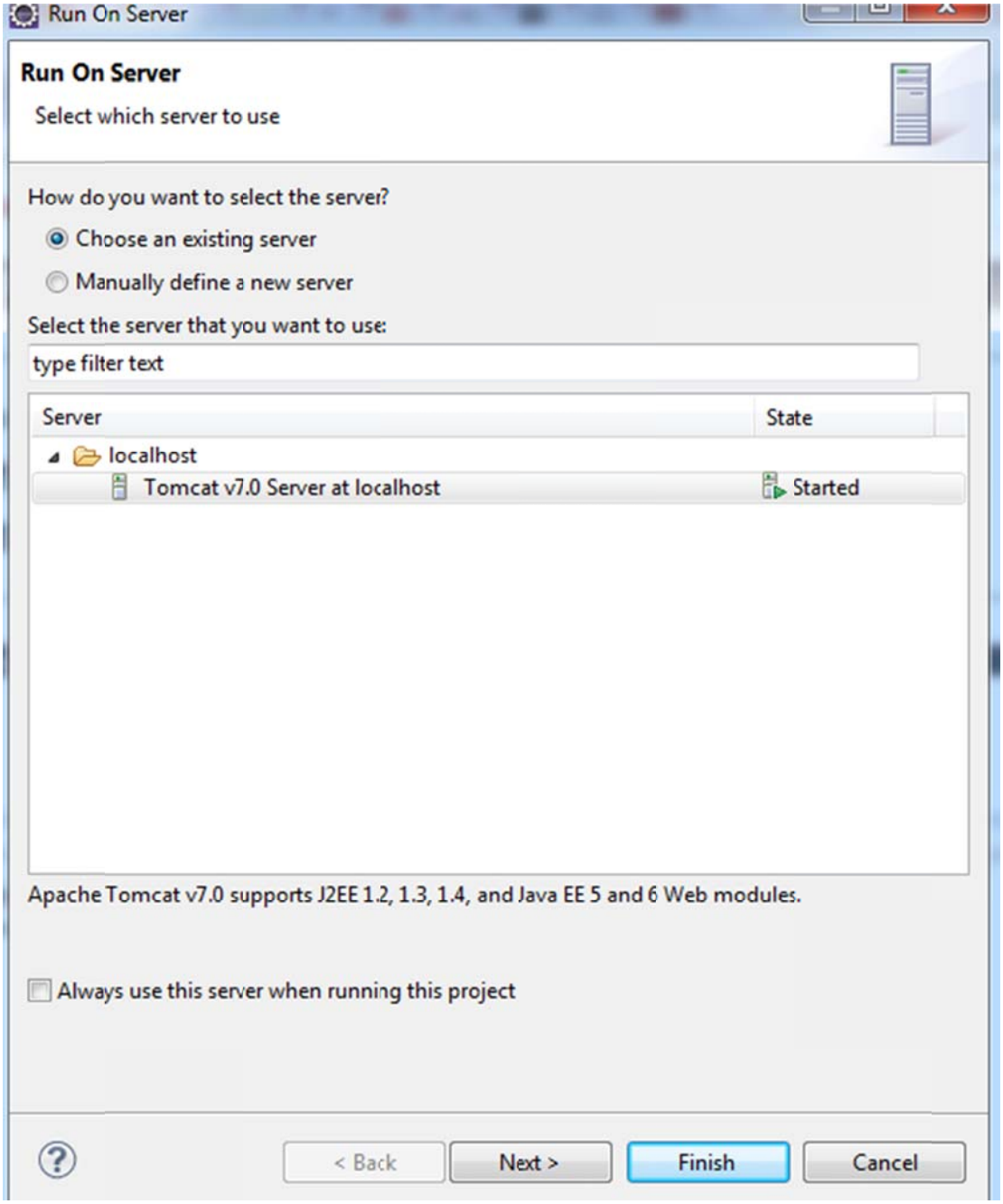
And

Change the code of the Users.java a little bit as following

```java
package com.dn.entity;

import java.io.Serializable;

@XmlRootElement
public class Users implements Serializable {
    private int id;
    private String firstName;
    private String lastName;

    public Users(){};

    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getFirstName() {
        return firstName;
    }
    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }
    public String getLastName() {
        return lastName;
    }
    public void setLastName(String lastName) {
        this.lastName = lastName;
    }
}
```

Now run the project on Tomcat server


Java EE - PhonegapEx/src/com/dn/entity/Users.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Java EE
Project Explorer
Users.java
UsersDao.java
Test0ao.java
MyRest.java
Apache Tomc...
Apache Tomc...
Apache Tomc...
AndroidClient
New
Go Into
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Piste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Refactor Alt+Shift+T
Import
Export
Refresh F5
Close Project
Close Unrelated Projects
Vilidate
Show in Remote Systems view
Run As
Debug As
Profile As
Team
Compare With
Restore from Local History...
Java EE Tools
Configure
Source
Properties Alt+Enter
1 Run on Server Alt+Shift+X, R
2 Java Applet Alt+Shift+X, A
3 Java Application Alt+Shift+X, J
Run Configurations...
ivate String firstName;
ivate String lastName;
lic int getId() {
return id;
lic void setId(int id) {
this.id = id;
lic String getFirstName() {
return firstName;
lic void setFirstName(String firstName) {
this.firstName = firstName;
lic String getLastName() {
return lastName;
Servers
Data Source Explorer
Snippets
Console
Remote Systems
Server 12c (12.1.1) at localhost [base_domain]

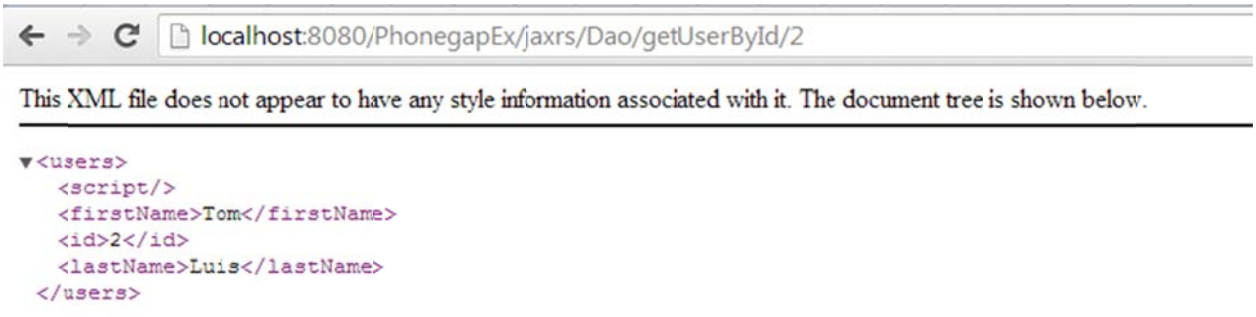


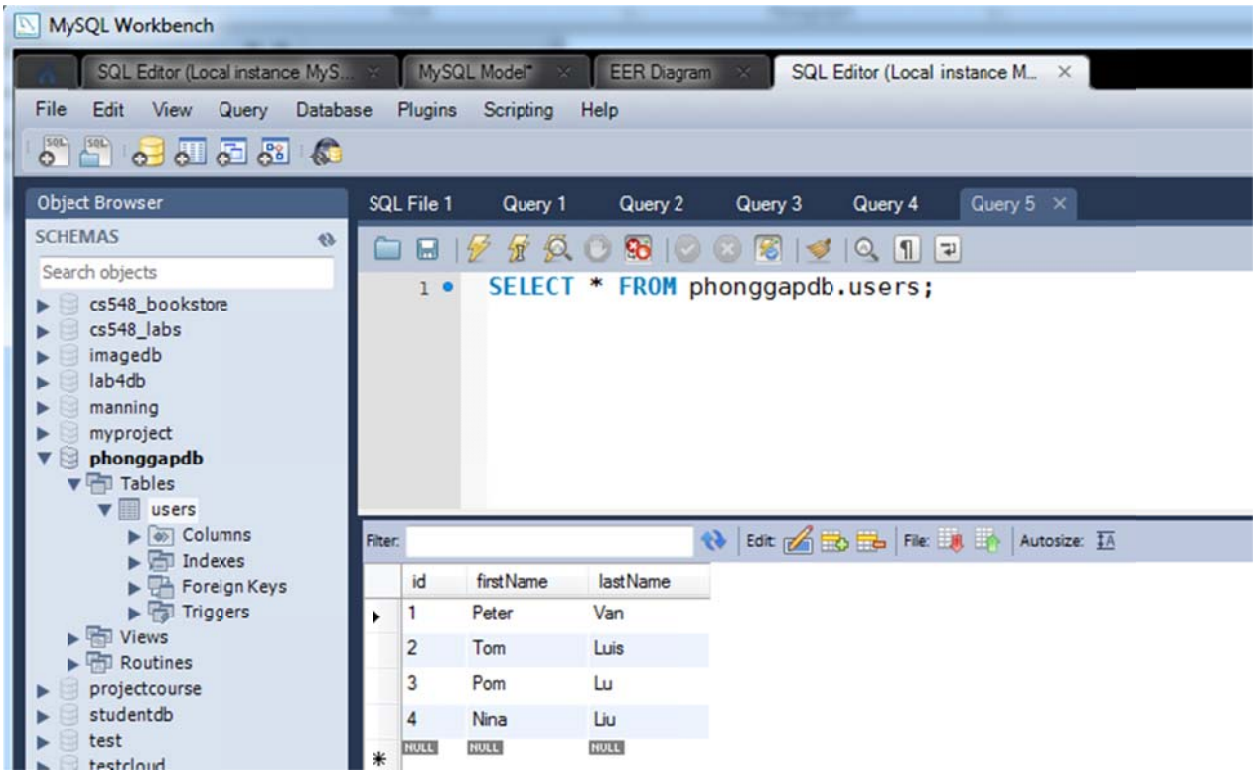
Use the browser to test the result.

Go to the link

http://localhost:8080/PhonegapEx/jaxrs/Dao/getUserById/2

➔ Get the user who has id =2





Try again with

http://localhost:8080/PhonegapEx/jaxrs/Dao/getUserById/4

Right now, the MyRest.java has only getUserById. Add more method to MyRest.java as the following.

```
 1 package com.dn.rest;
   PhonegapEx/src/com/dn/entity/Users.java
 2
 3 import javax.ws.rs.Consumes;
 4 import javax.ws.rs.GET;
 5 import javax.ws.rs.POST;
 6 import javax.ws.rs.Path;
 7 import javax.ws.rs.PathParam;
 8 import javax.ws.rs.Produces;
 9
10 import com.dn.dao.UsersDao;
11 import com.dn.entity.Users;
12
13 @Path("/Dao")
14 public class MyRest {
15
16       @GET
17       @Path("/getUserById/{id}")
18       @Produces({"application/xml", "application/json"})
19       public Users getUser(@PathParam("id") int id) {
20           UsersDao usersDao = new UsersDao();
21           Users user = usersDao.findById(id);
```

```
22          return user;
23      }
24
25      @POST
26      @Path("/CreateUser/{firstName}/{lastName}")
27      @Consumes()
28      public void createUser(@PathParam("firstName") String firstName,
29                             @PathParam("lastName") String lastName) {
30          UsersDao usersDao = new UsersDao();
31          Users user = new Users();
32          user.setFirstName(firstName);
33          user.setLastName(lastName);
34          usersDao.createUser(user);
35      }
36  }
37
```

And now we can save a new user to database by using uri with a POST method

http://localhost:8080/PhonegapEx/jaxrs/Dao/CreateUser/firstName/lastName

For example, to save a user with first name : Tina, last name: Le →

http:// localhost:8080/PhonegapEx/jaxrs/Dao/CreateUser/Tina/Le + POST method. More detail: next section.

1. Programming for client side.

The client side can use the Rest Web service in the above to get information of user from database. For example, access this link http://localhost:8080/PhonegapEx/jaxrs/Dao/getUserById/4, we will get

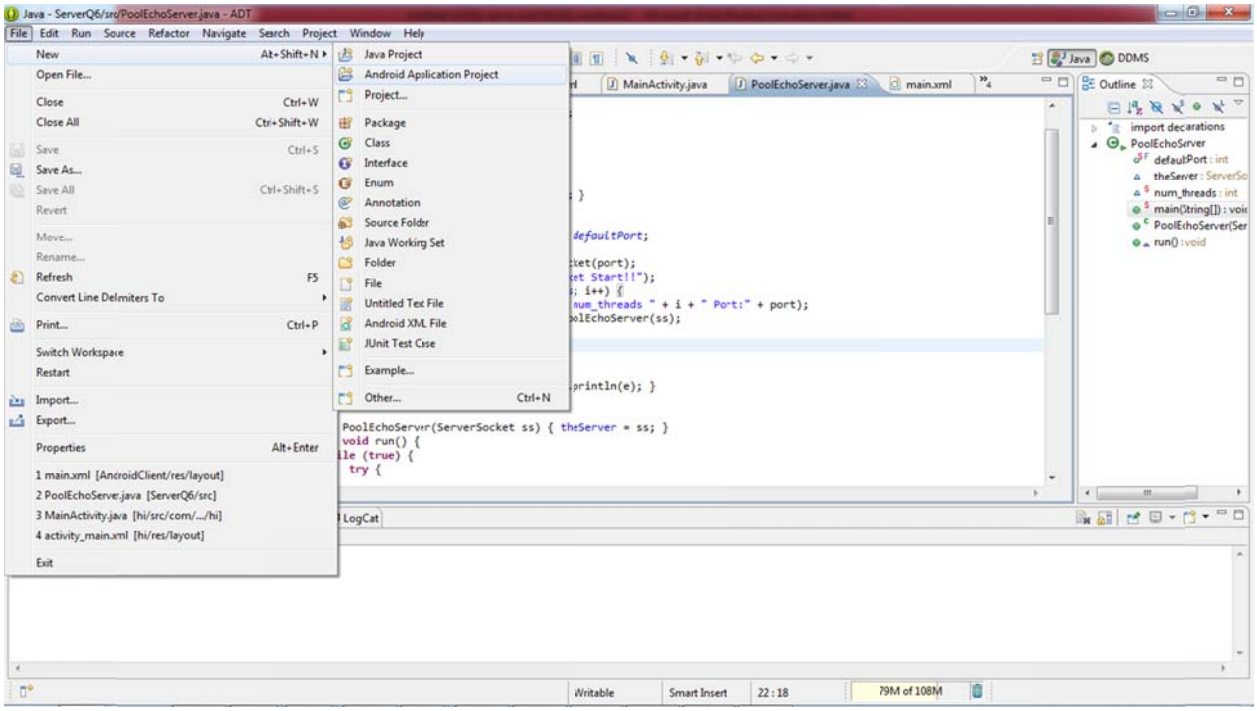
```
▼<users>
   <script/>
   <firstName>Nina</firstName>
   <id>4</id>
   <lastName>Liu</lastName>
 </users>
```

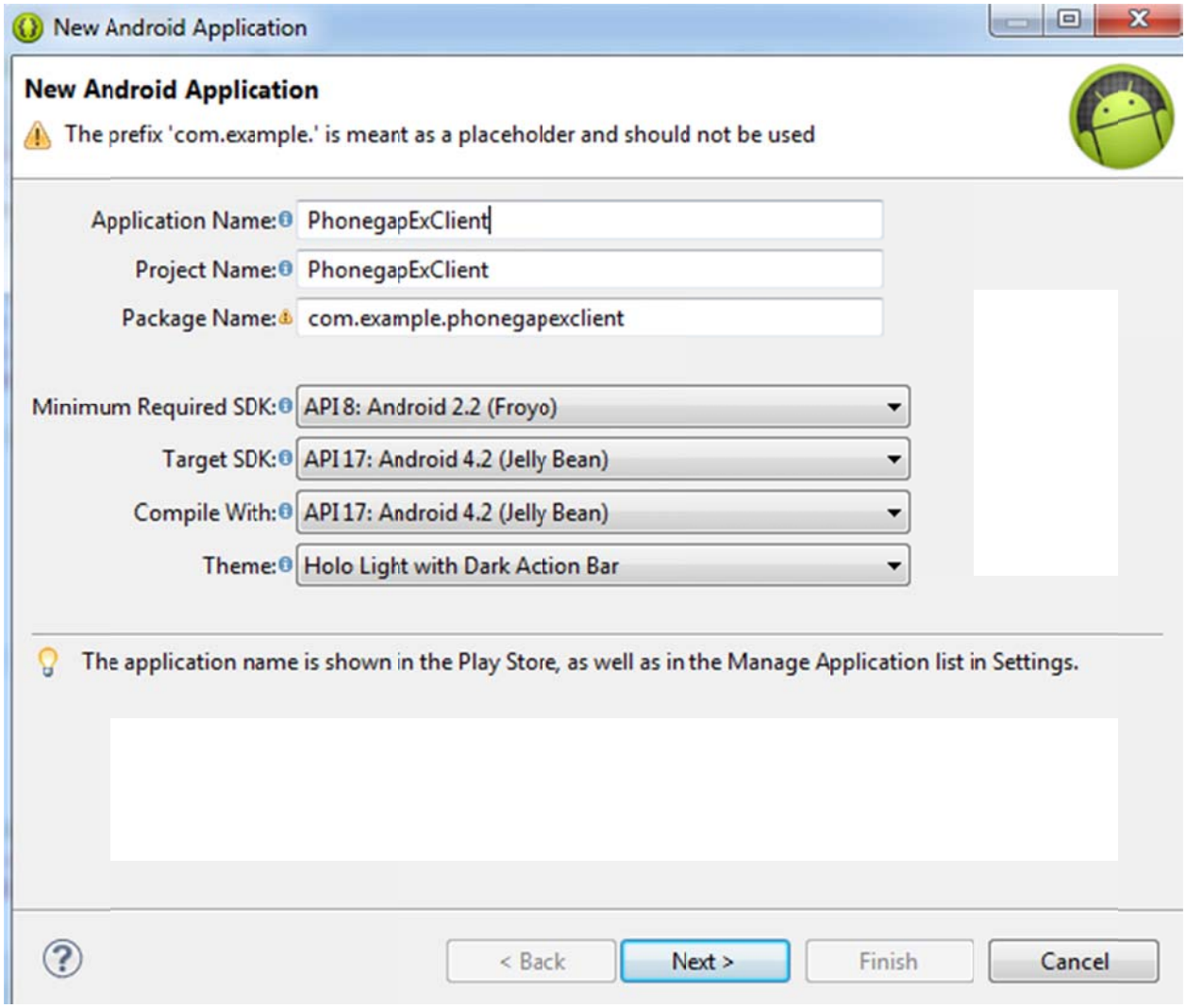
in XML or in Json.

The client can also create a new user and save to database. For example,

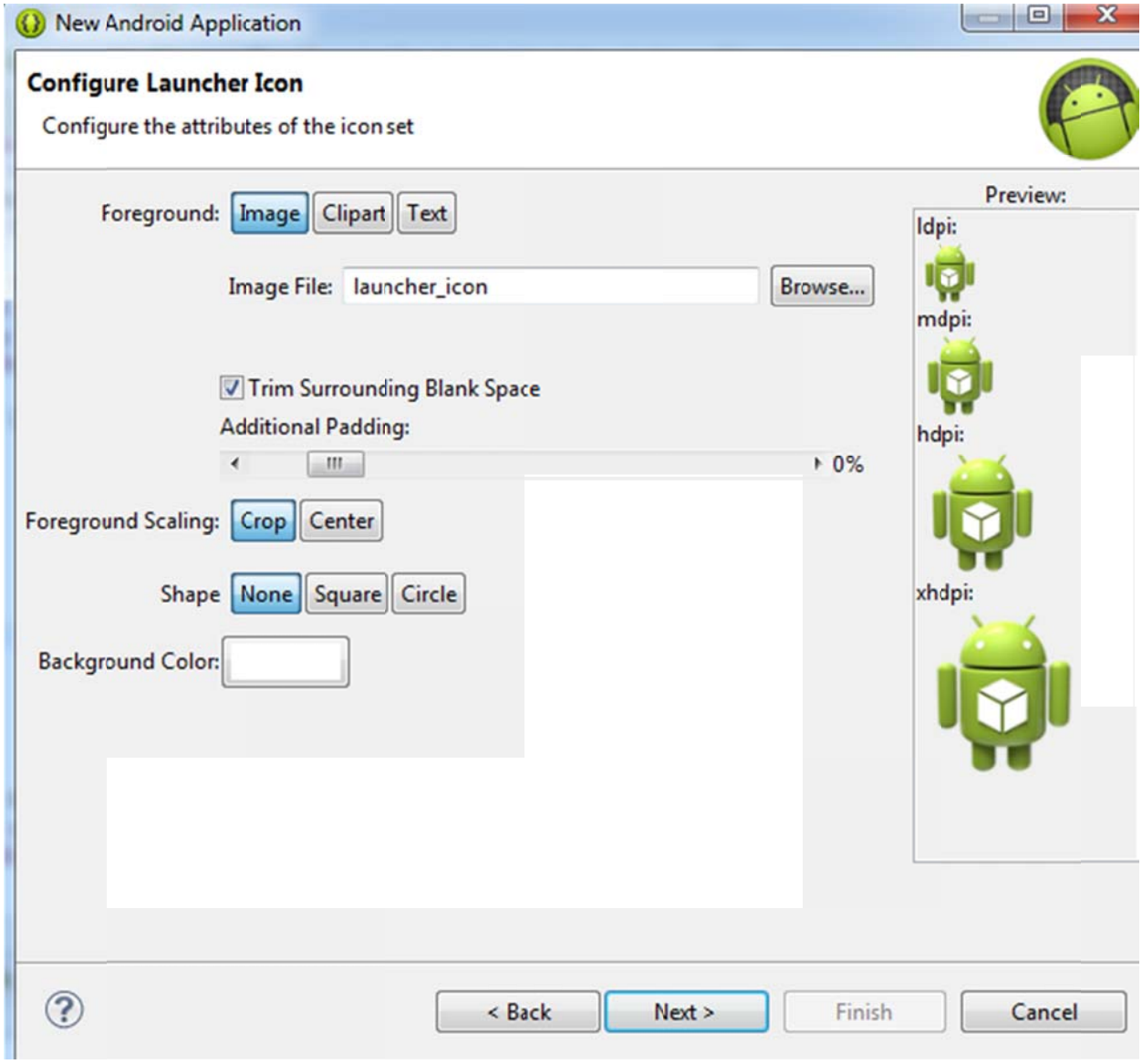
In this section, we will use html, jQuery, and Phonegap to build Android application. Even we don't know much about Android programming.

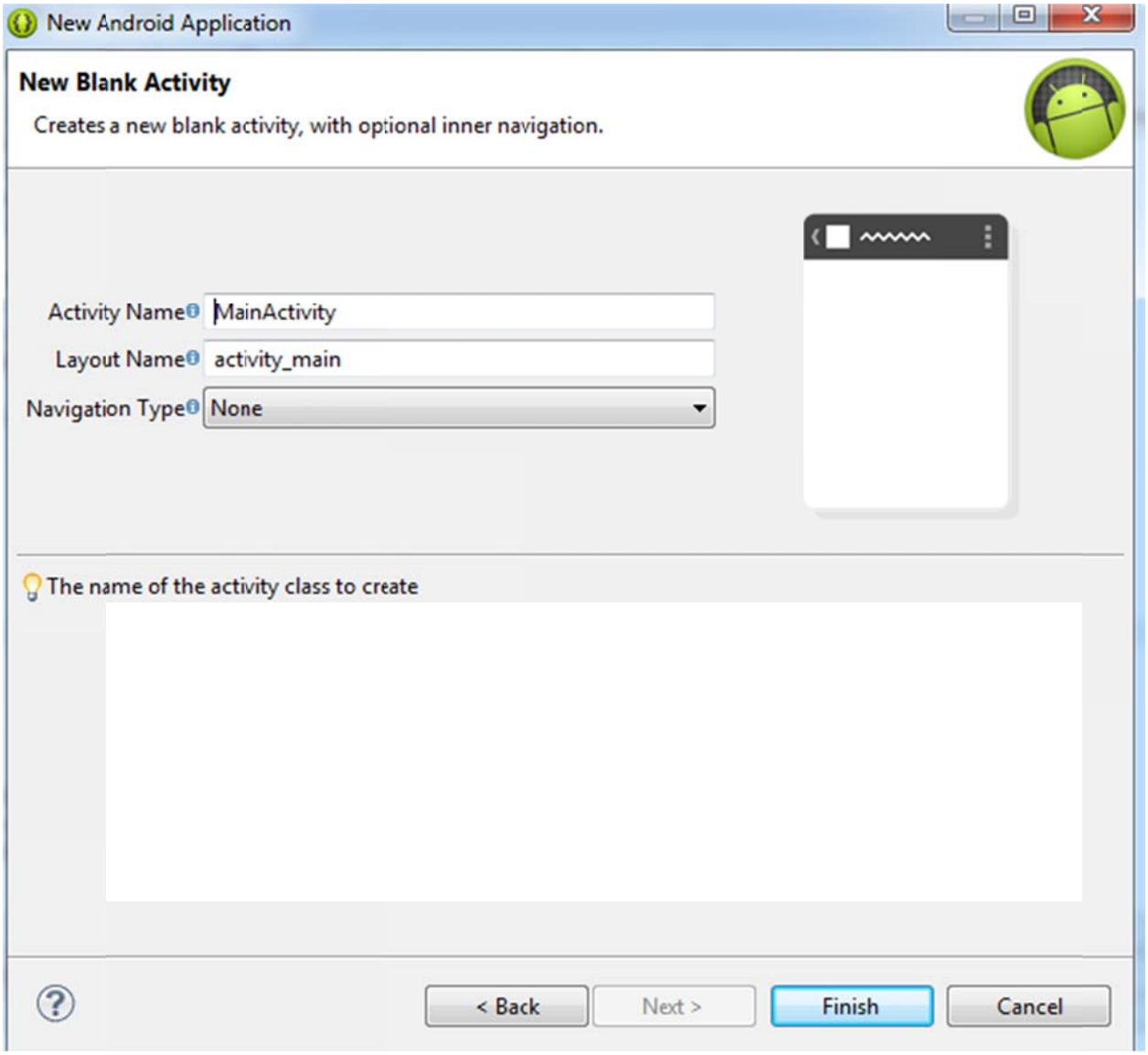
Open Eclipse ADT-bundle and create a new Android project.


Java - ServerQ6/src/PoolEchoServer.java - ADT
File Edit Run Source Refactor Navigate Search Project Window Help
New Alt+Shift+N
Open File...
Close Ctrl+W
Close All Ctrl+Shift+W
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename...
Refresh F5
Convert Line Delimiters To
Print... Ctrl+P
Switch Workspace
Restart
Import...
Export...
Properties Alt+Enter
1 main.xml [AndroidClient/res/layout]
2 PoolEchoServer.java [ServerQ6/src]
3 MainActivity.java [hi/src/com/.../hi]
4 activity_main.xml [hi/res/layout]
Exit
Java Project
Android Application Project
Project...
Package
Class
Interface
Enum
Annotation
Source Folder
Java Working Set
Folder
File
Untitled Text File
Android XML File
JUnit Test Case
Example...
Other... Ctrl+N
MainActivity.java
PoolEchoServer.java
main.xml
Java
DDMS
Outline
import decarations
PoolEchoServer
defaultPort : int
theServer : ServerSo
num_threads : int
main(String[]) : voi
PoolEchoServer(Ser
run() : void
LogCat
Writable
Smart Insert
22 : 18
79M of 108M

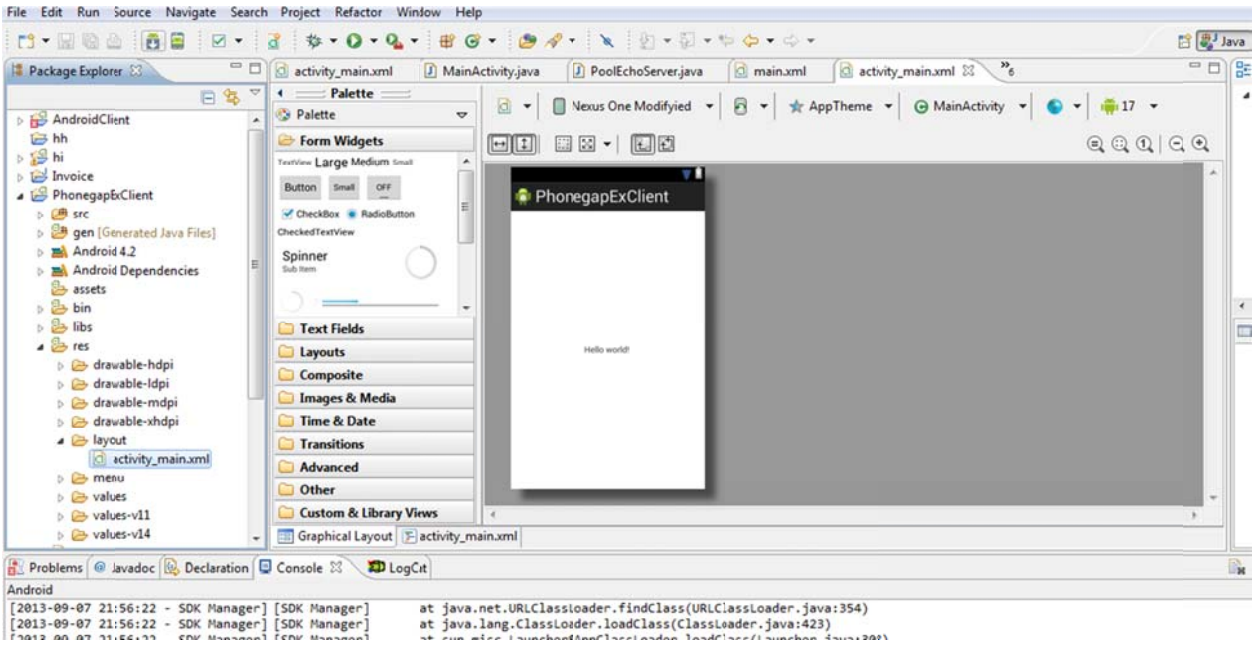


Click next, next..

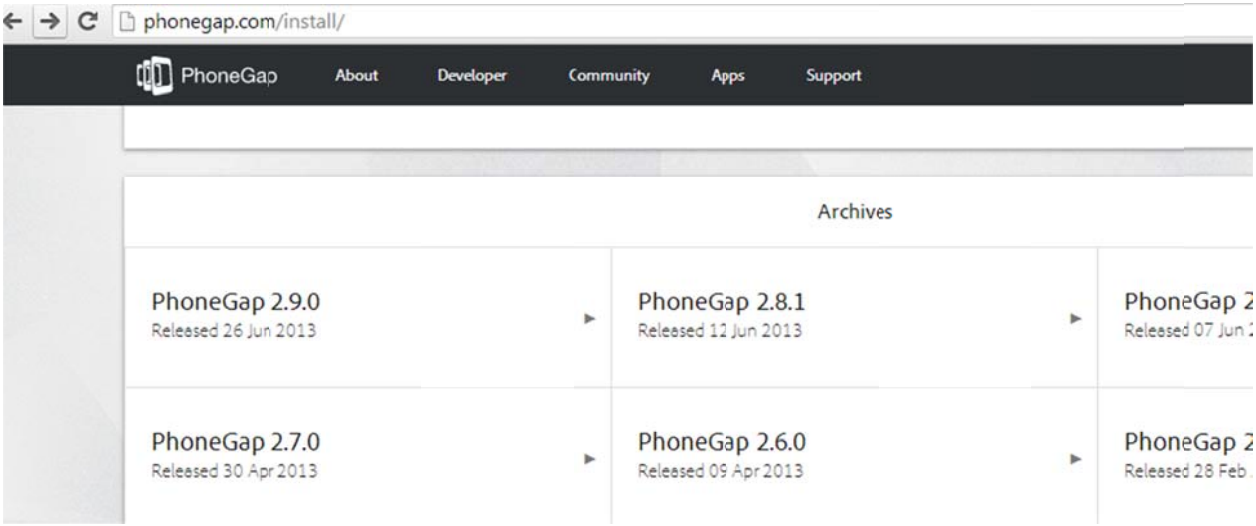

New Android Application
Configure Launcher Icon
Configure the attributes of the icon set
Foreground:
Image
Clipart
Text
Image File:
launcher_icon
Browse...
Trim Surrounding Blank Space
Additional Padding:
0%
Foreground Scaling:
Crop
Center
Shape
None
Square
Circle
Background Color:
Preview:
ldpi:
mdpi:
hdpi:
xhdpi:
< Back
Next >
Finish
Cancel


New Android Application
New Blank Activity
Creates a new blank activity, with optional inner navigation.
Activity Name
MainActivity
Layout Name
activity_main
Navigation Type
None
The name of the activity class to create
< Back
Next >
Finish
Cancel

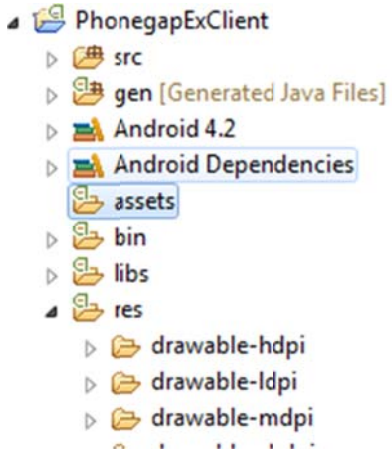
And finish

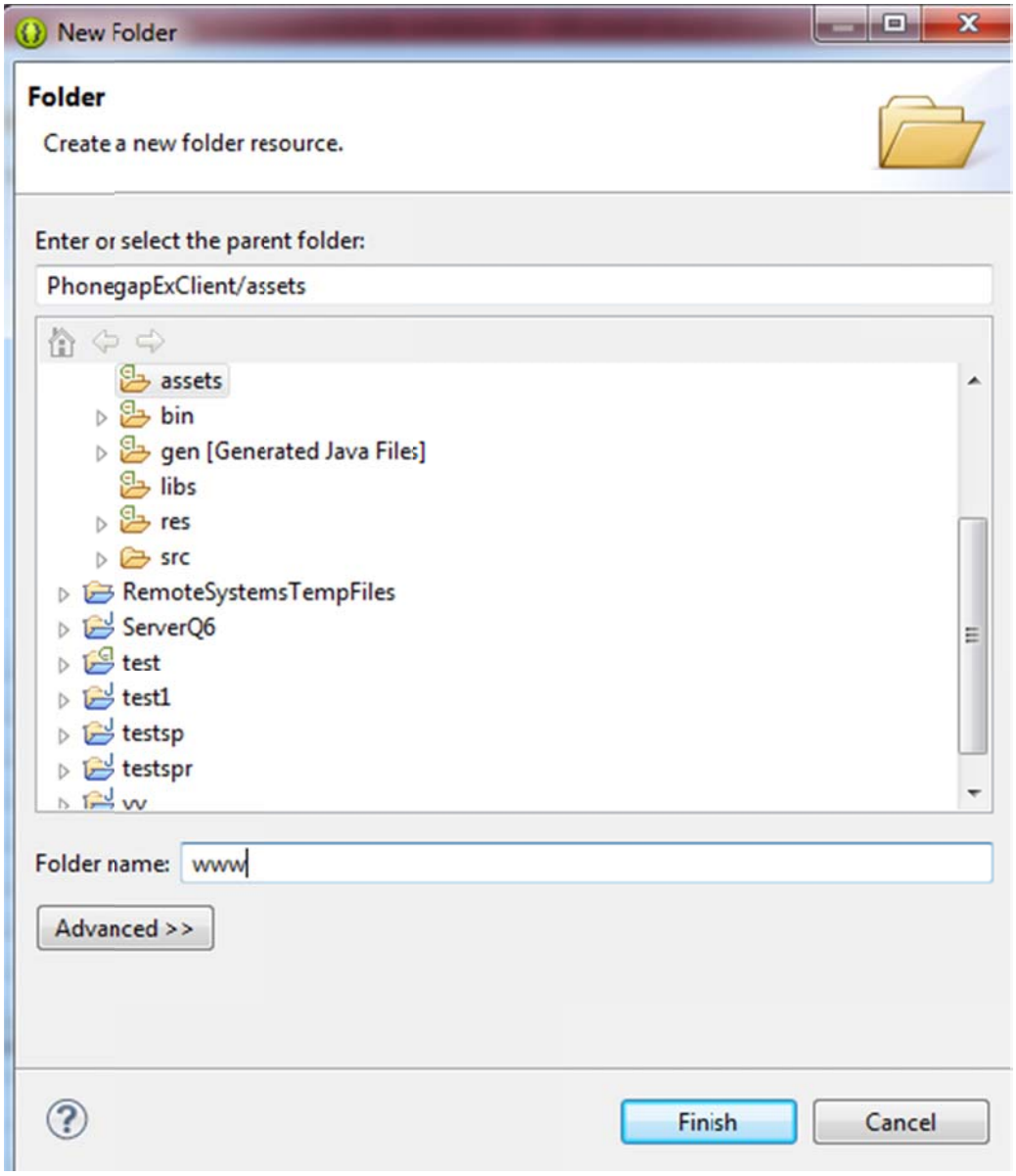


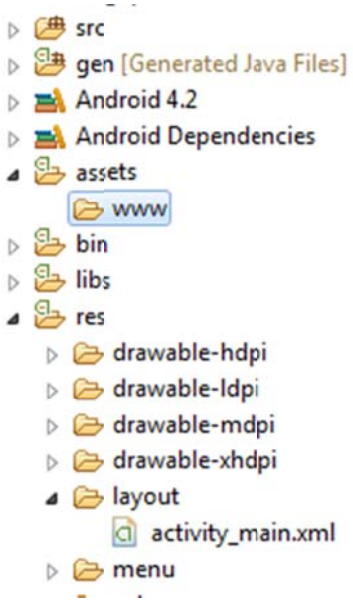
Go to phonegap website and download phonegap package.



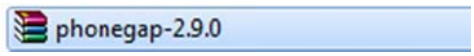
Create a folder www in the assets folder


PhonegapExClient
src
gen [Generated Java Files]
Android 4.2
Android Dependencies
assets
bin
libs
res
drawable-hdpi
drawable-ldpi
drawable-mdpi

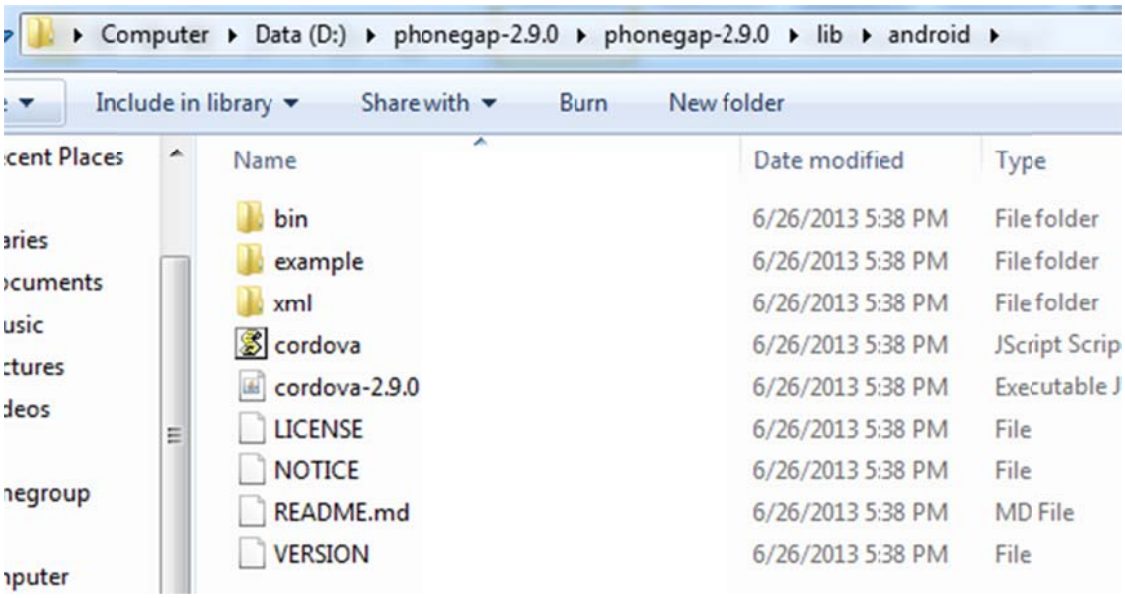

New Folder
Folder
Create a new folder resource.
Enter or select the parent folder:
PhonegapExClient/assets
assets
bin
gen [Generated Java Files]
libs
res
src
RemoteSystemsTempFiles
ServerQ6
test
test1
testsp
testspr
Folder name:
www
Advanced >>
Finish
Cancel



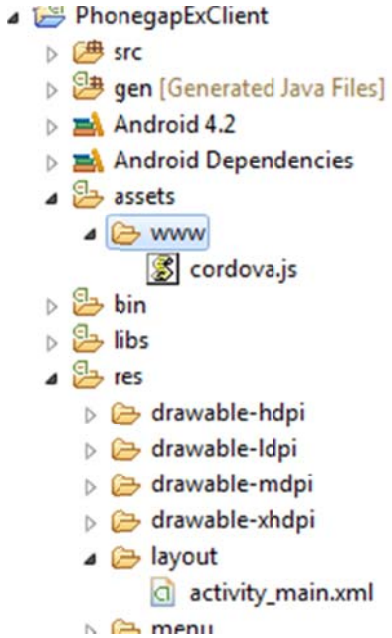
After downloading phonegap, unzip it



Go to lib folder, Android folder

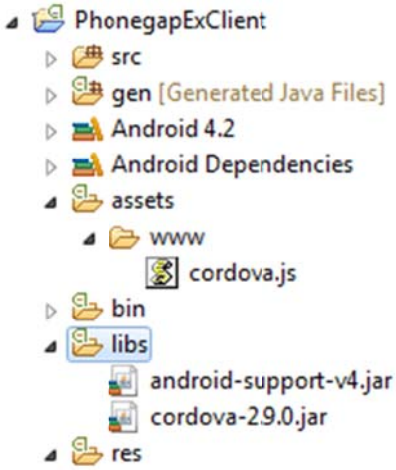


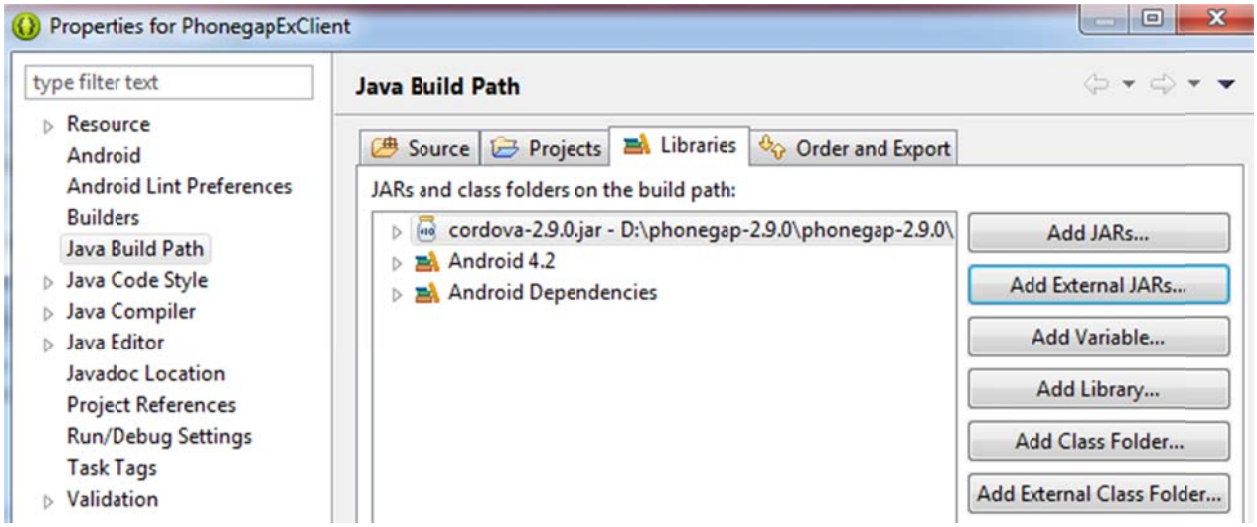
Copy cordova.js to www folder



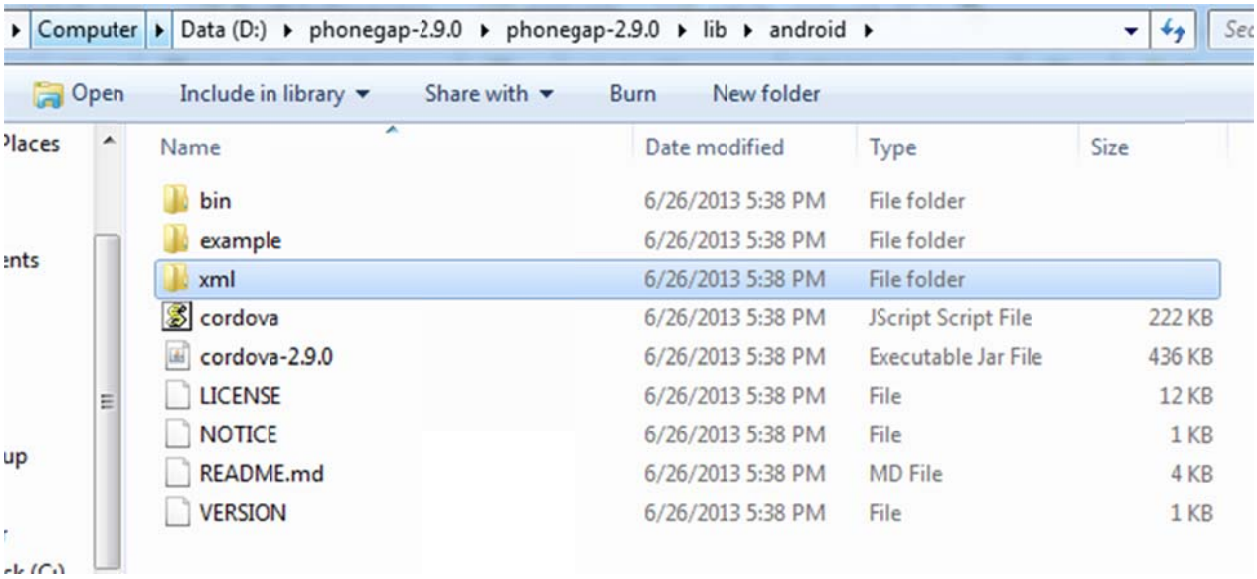
Also copy

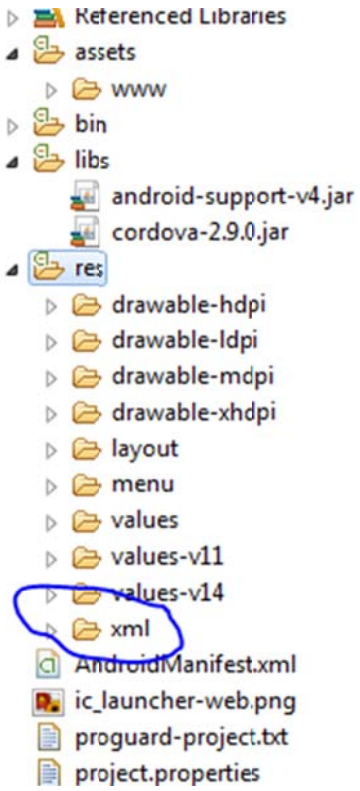
To libs folder and add it to classpath





Copy xml folder to res folder

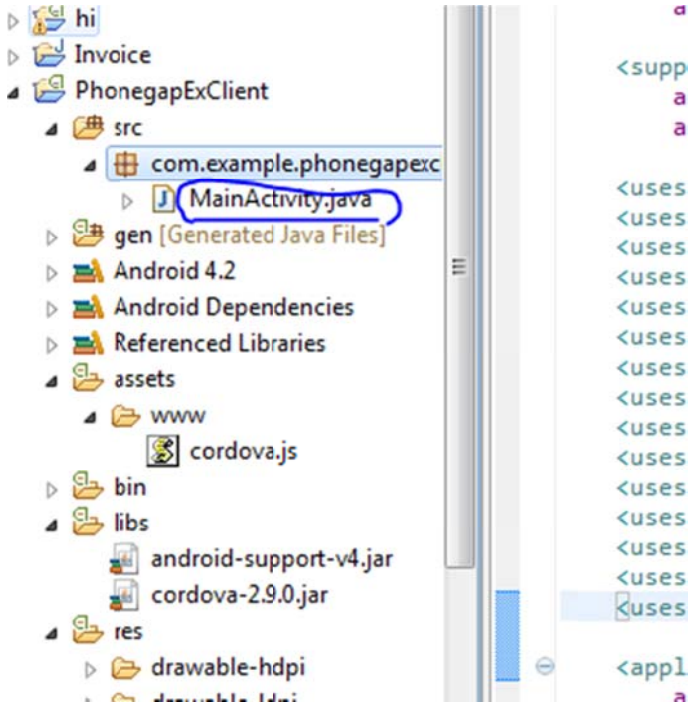


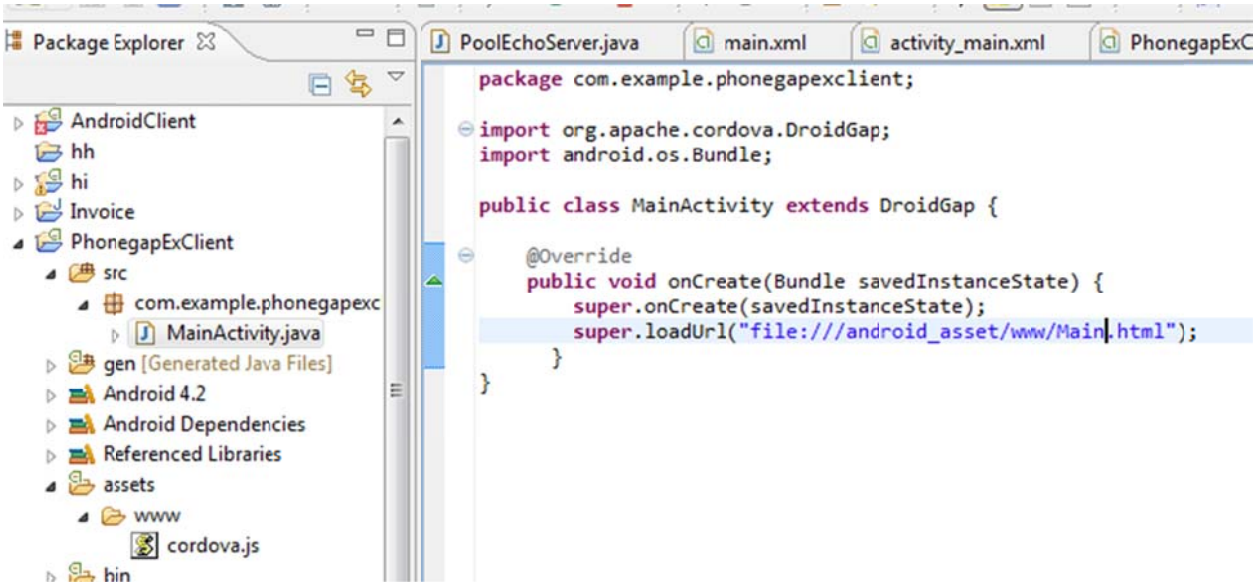


Open AndroidManifest.xml and add some code for screen and permission as following

```
<uses-sdk
    android:minSdkVersion="8"
    android:targetSdkVersion="17" />

<supports-screens android:largeScreens="true"
    android:normalScreens="true" android:smallScreens="true"
    android:resizeable="true" android:anyDensity="true" />

<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.RECEIVE_SMS" />
<uses-permission android:name="android.permission.RECORD_AUDIO" />
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.GET_ACCOUNTS" />

<application
    android:allowBackup="true"
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name"
    android:theme="@style/AppTheme" >
    <activity
        android:name="com.example.phonegapexclient.MainActivity"
```

Open MainActivity.java and change the code into

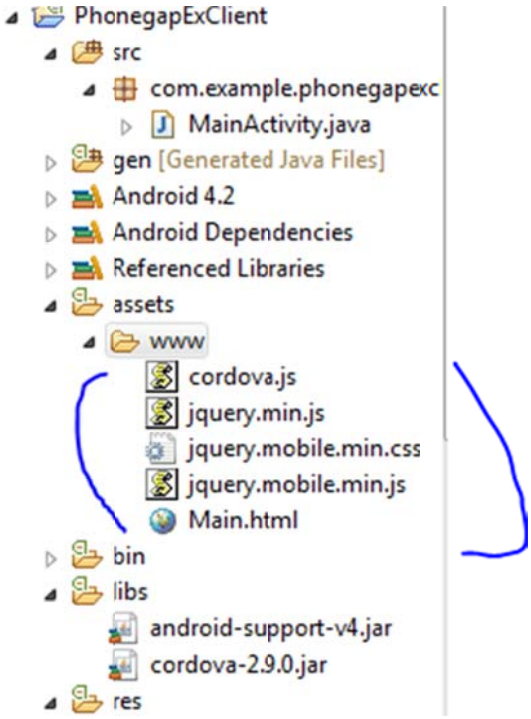




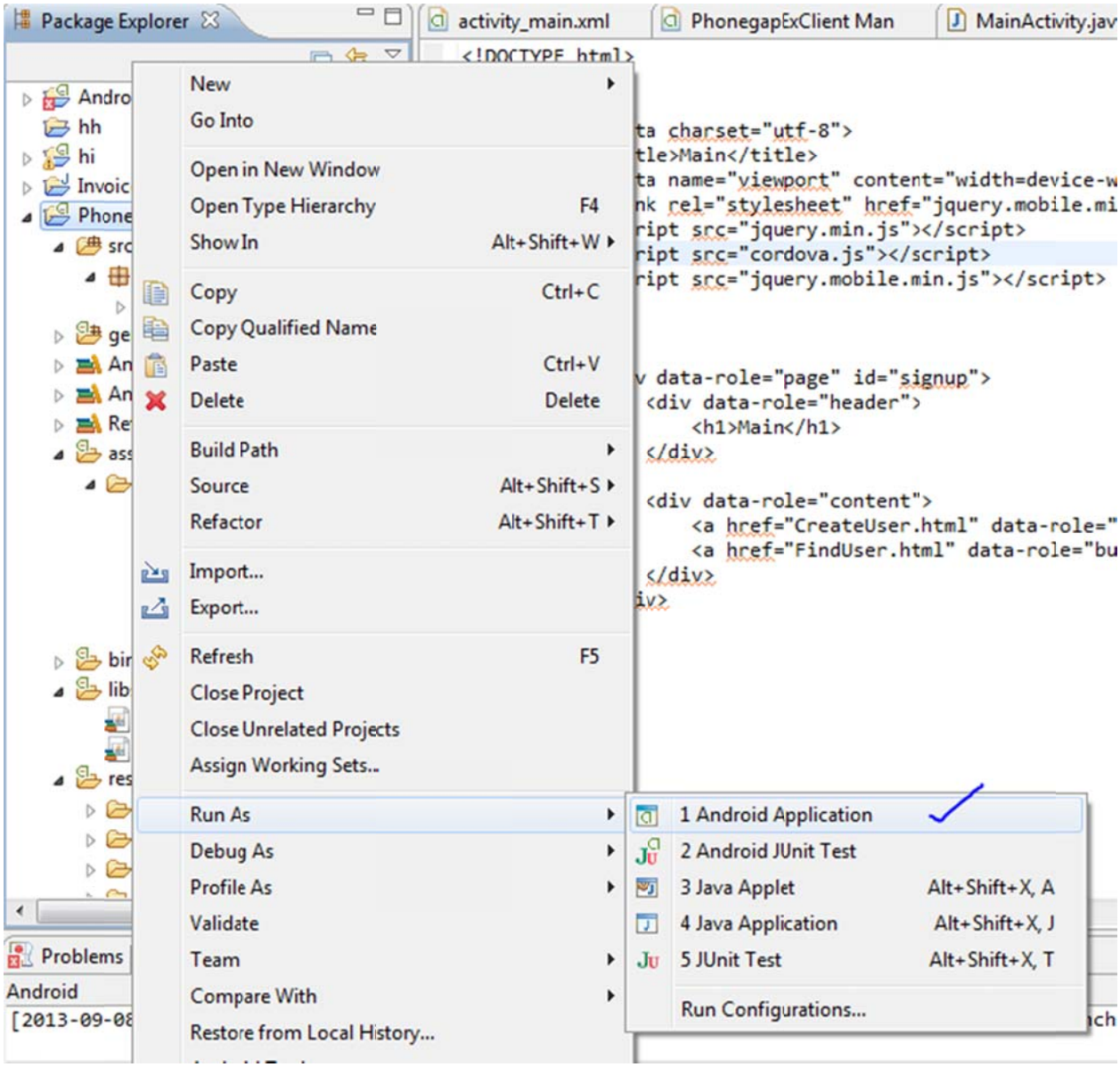
Add Main.html and some js libraries to www folder

Main.html

```
<!DOCTYPE html>
<html>
    <head>
            <meta charset="utf-8">
            <title>Main</title>
            <meta name="viewport" content="width=device-width, initial-scale=1">
            <link rel="stylesheet" href="jquery.mobile.min.css" />
            <script src="jquery.min.js"></script>
            <script src="cordova.js"></script>
            <script src="jquery.mobile.min.js"></script>
    </head>

        <body>
            <div data-role="page" id="signup">
                <div data-role="header">
                    <h1>Main</h1>
                </div>

                <div data-role="content">
                    <a href="CreateUser.html" data-role="button" id="CreateUser" rel="external">Create new User</a>
                    <a href="FindUser.html" data-role="button" id="FindUser" rel="external">Find User</a>
                </div>
            </div>
        </body>
</html>
```



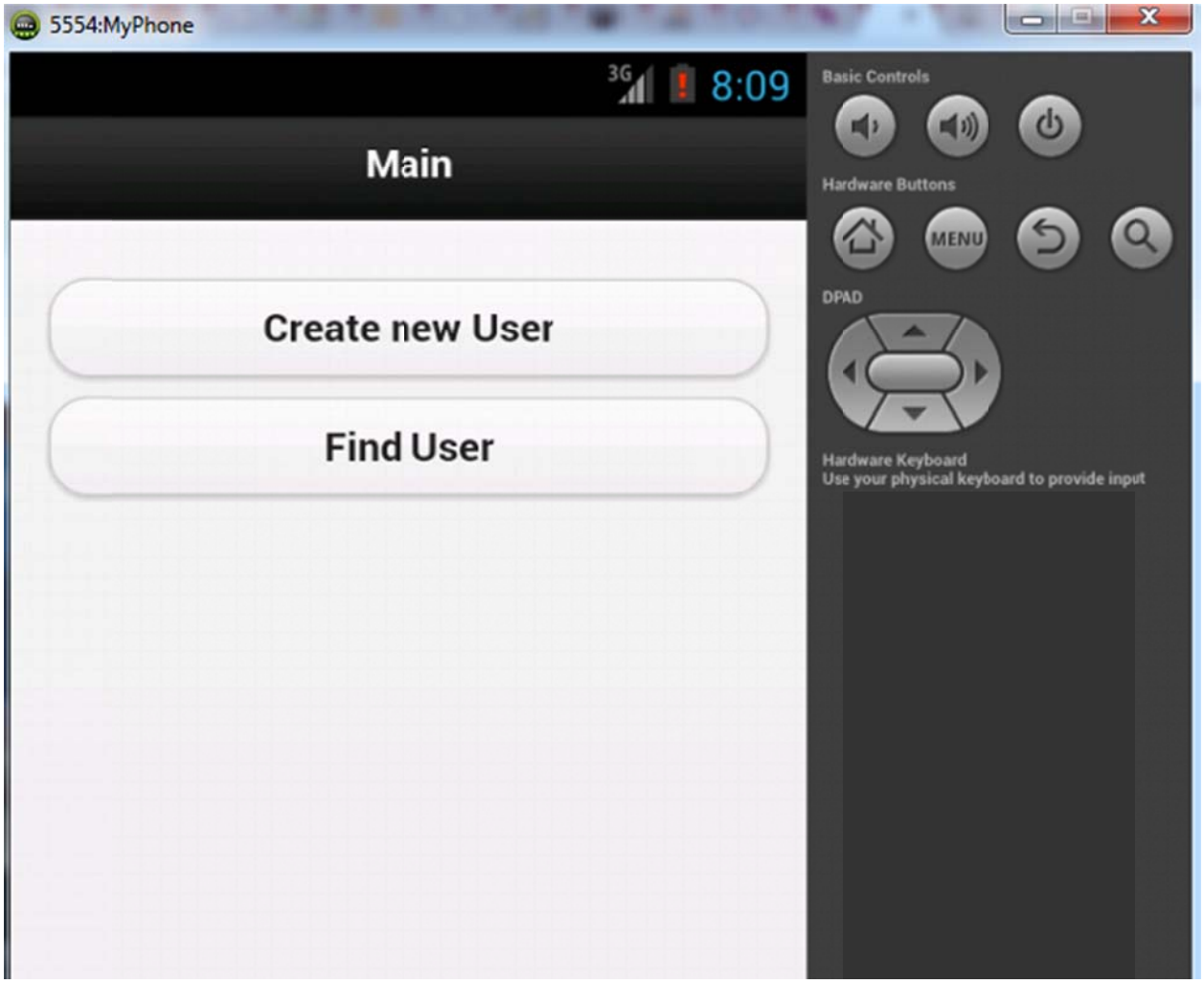
Click Run Android Application


Package Explorer
activity_main.xml
PhonegapExClient Man
MainActivity.jav
<!DOCTYPE html>
ta charset="utf-8">
tle>Main</title>
ta name="viewport" content="width=device-w
nk rel="stylesheet" href="jquery.mobile.mi
ript src="jquery.min.js"></script>
ript src="cordova.js"></script>
ript src="jquery.mobile.min.js"></script>
v data-role="page" id="signup">

<h1>Main</h1>

<a href="CreateUser.html" data-role="
<a href="FindUser.html" data-role="bu

iv>
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Run As
Debug As
Profile As
Validate
Team
Compare With
Restore from Local History...
1 Android Application
2 Android JUnit Test
3 Java Applet Alt+Shift+X, A
4 Java Application Alt+Shift+X, J
5 JUnit Test Alt+Shift+X, T
Run Configurations...
Problems
Android
[2013-09-08



Next we need add 2 files: CreateUser.html and FindUser.html to www folder

```
        <script src="jquery.mobile.min.js"></script>
    </head>

        <body>
            <div data-role="page" id="signup">
                <div data-role="header">
                    <h1>Main</h1>
                </div>

                <div data-role="content">
                    <a href="CreateUser.html" data-role="button"
                    <a href="FindUser.html" data-role="button" i
                </div>
            </div>
        </body>
</html>
```

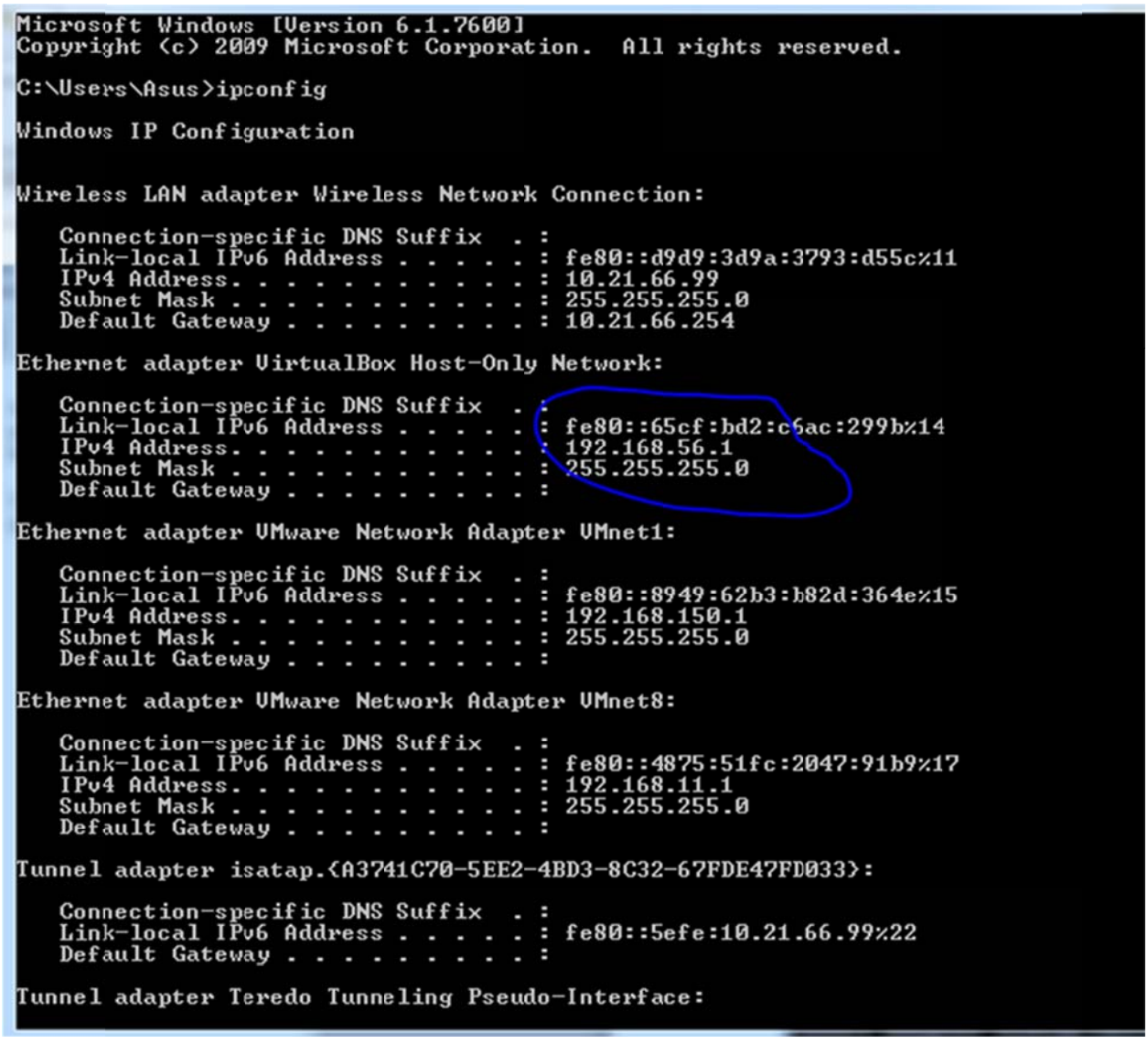
First, doing with FindUser.html

FindUser.html

```
!DOCTYPE html>
html>
        <head>
            <meta charset="utf-8">
            <title>Find User</title>
            <meta name="viewport" content="width=device-width, initial-scale=1">
            <link rel="stylesheet" href="jquery.mobile.min.css" />
            <script src="jquery.min.js"></script>
            <script src="cordova.js"></script>
            <script src="jquery.mobile.min.js"></script>
            <script type="text/javascript">

                var rootURLusers="http://192.168.56.1:8080/PhonegapEx/jaxrs/Dao/getUserById";

                function findById(id) {
                    $.ajax({
                        type: 'GET',
                        url: rootURLusers + '/' + id  ,
                        contentType: "application/json; charset=utf-8",
                        dataType: "json",
                        processData: false, async: false,
                        success: function (msg) {
                                $("#contentId").empty();
                                $("#contentId").append("id = "+msg.id + "<br>");
                                $("#contentId").append("First name = "+msg.firstName + "<br>");
                                $("#contentId").append("Last name = "+msg.lastName + "<br>");
                        }
                    });
                }

                document.addEventListener("deviceready", onDeviceReady, false);
                function onDeviceReady() {
                    $("#Lookup").bind("click", function(){
                        var id = $("#userid").val()+"";
                        findById(id);
                    });
                }

            </script>

        </head>

        <body>
            <div data-role="page" id="signup">
                <div data-role="header">
                    <h1>Find User</h1>
                </div>

                <div data-role="content">
```

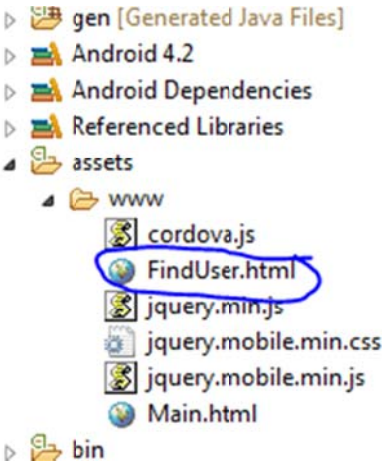
```
50                      <input type="text" name="userid" id="userid"/>
51                      <a href="#" data-role="button" id="Lookup">Look up</a>
52                      <a href="Main.html" data-role="button" id="Main" rel="external">Back</a>
53                      <div id="contentId"></div>
54                  </div>
55              </div>
56          </body>
57      /html>
58
```

For the link, we cannot use localhost, we need to replace with our host ip

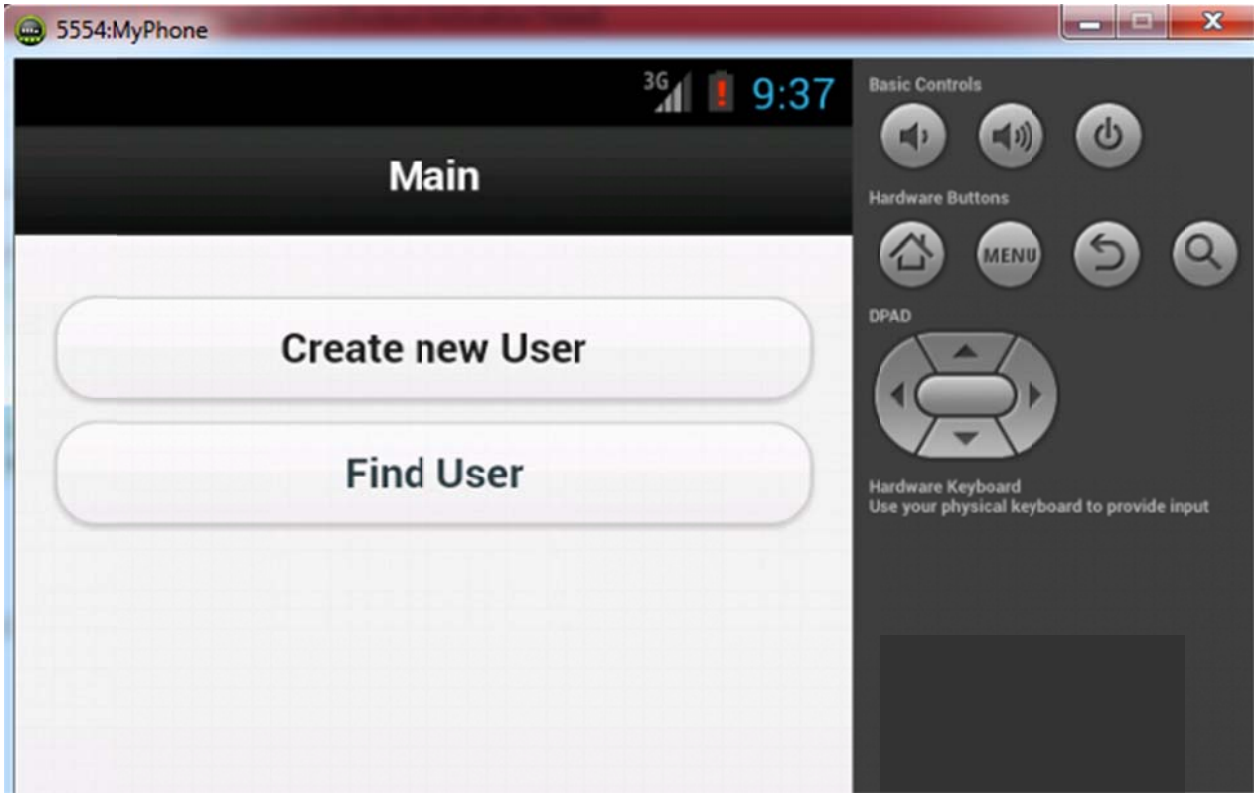


Use 192.168.56.1

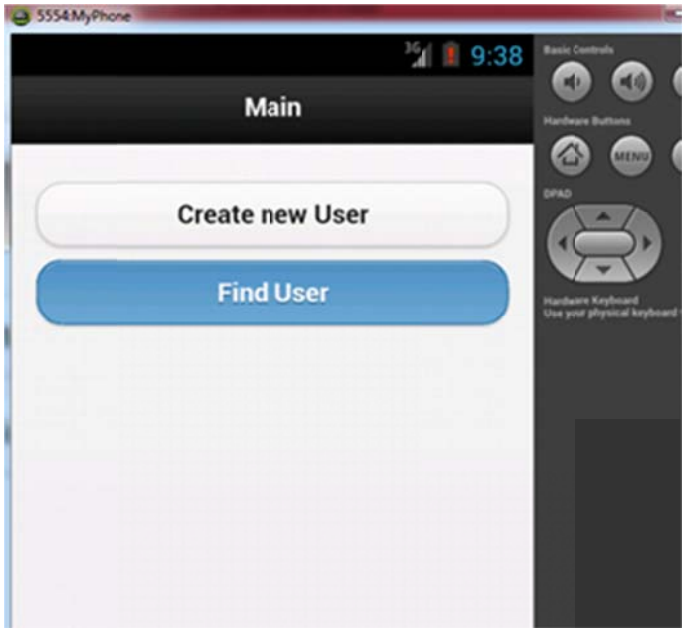
```
1  !DOCTYPE html>
2  html>
3      <head>
4          <meta charset="utf-8">
5          <title>Find User</title>
6          <meta name="viewport" content="width=device-width, initial-scale=1">
7          <link rel="stylesheet" href="jquery.mobile.min.css" />
8          <script src="jquery.min.js"></script>
9          <script src="cordova.js"></script>
10         <script src="jquery.mobile.min.js"></script>
11         <script type="text/javascript">
12
13             var rootURLusers="http://192.168.56.1:8080/PhonegapEx/jaxrs/Dao/getUserById";
14
15             function findById(id) {
16                 $.ajax({
17                     type: 'GET',
18                     url: rootURLusers + '/' + id  ,
19                     contentType: "application/json; charset=utf-8",
20                     dataType: "json",
21                     processData: false, async: false,
22                     success: function (msg) {
```

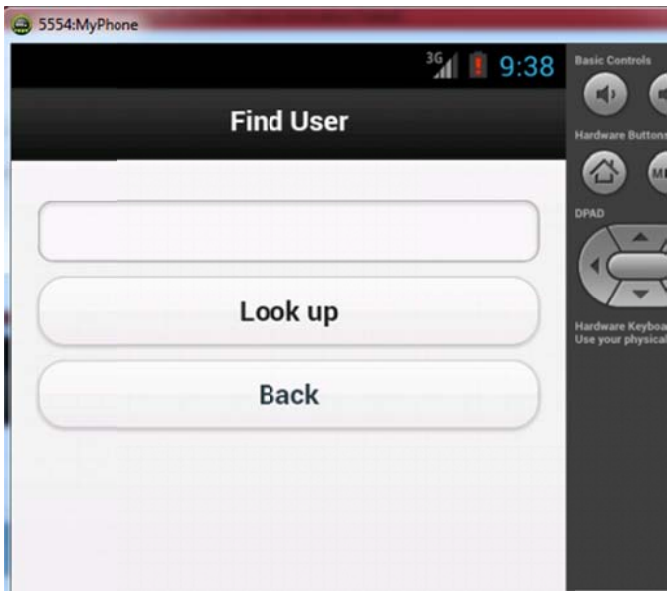


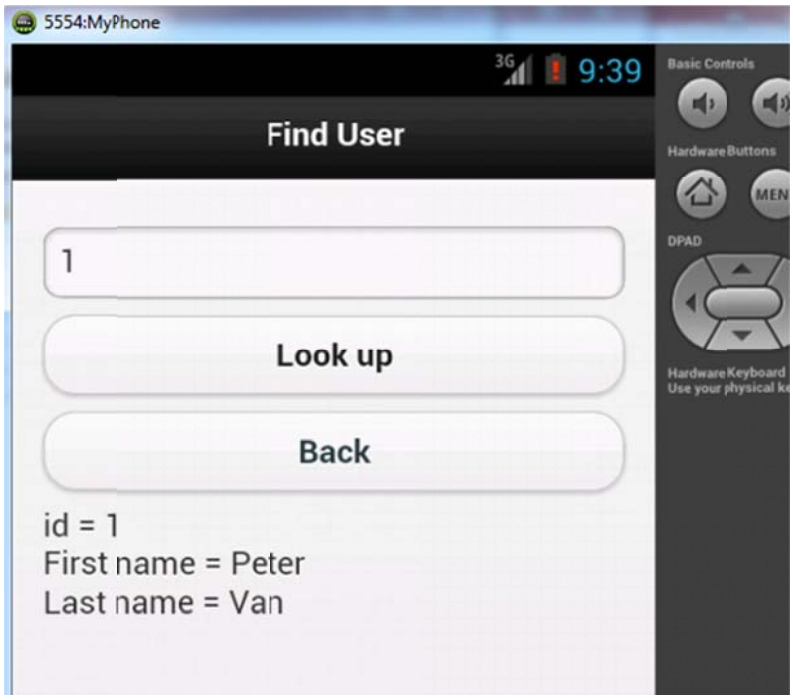
Now run the program

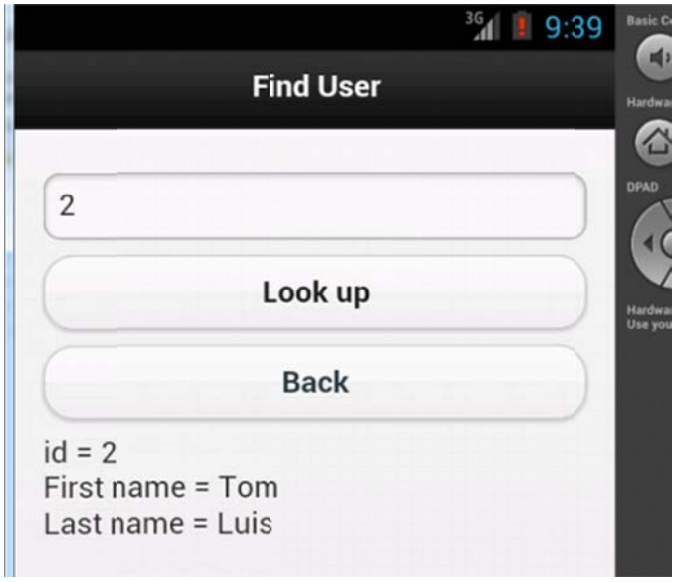


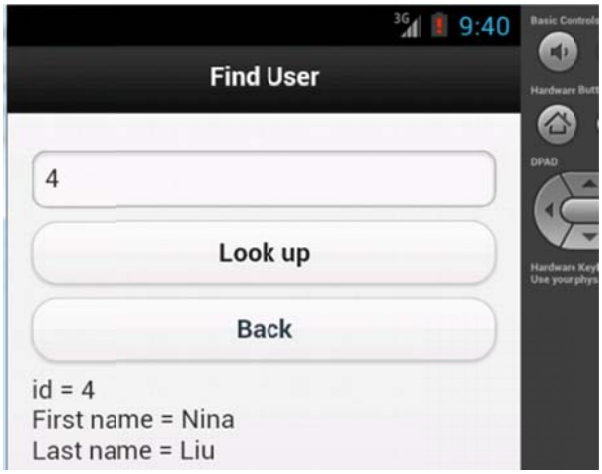
Click on Find User



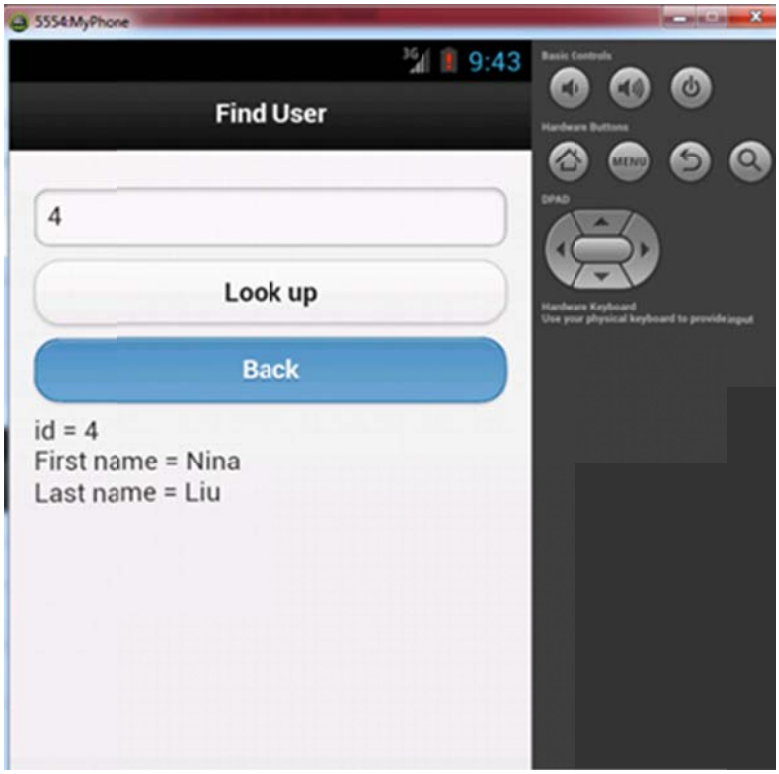

5554:MyPhone
9:38
Find User
Look up
Back
Basic Controls
Hardware Buttons
DPAD

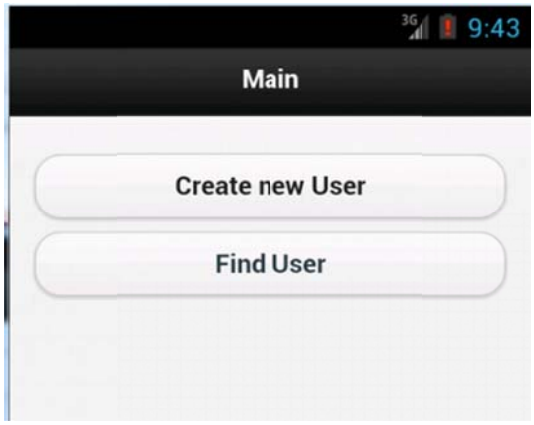

5554:MyPhone
9:39
Find User
1
Look up
Back
id = 1
First name = Peter
Last name = Van
Basic Controls
Hardware Buttons
DPAD


9:39
Find User
2
Look up
Back
id = 2
First name = Tom
Last name = Luis


9:40
Find User
4
Look up
Back
id = 4
First name = Nina
Last name = Liu

Click Back

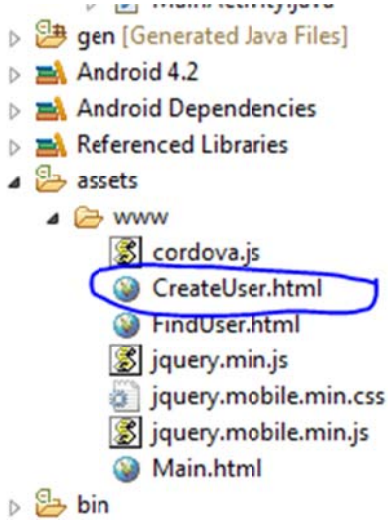




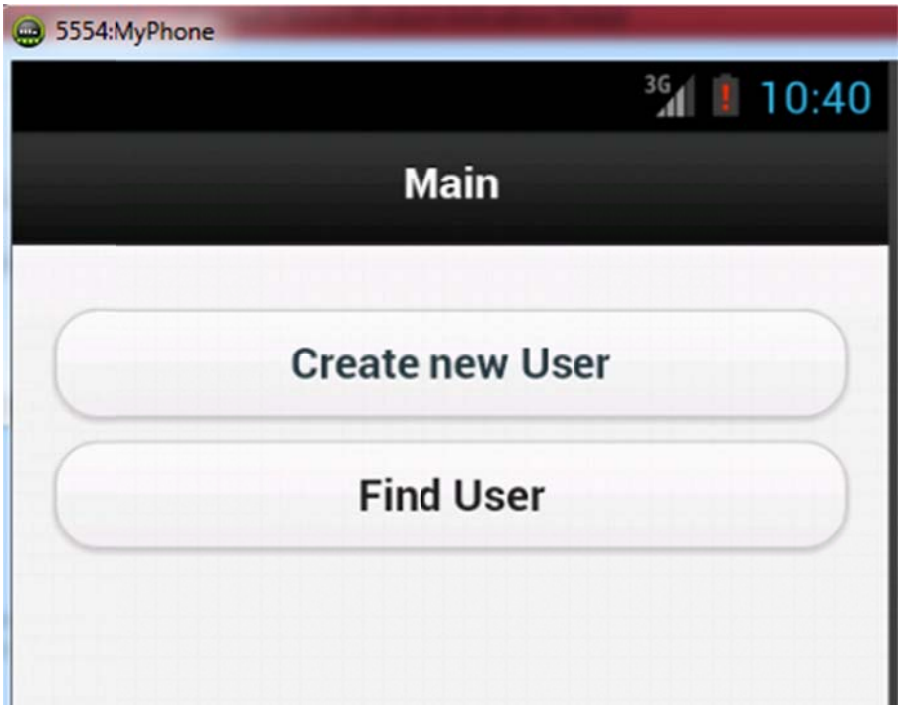
Second, doing with CreateUser.html

CreateUser.html

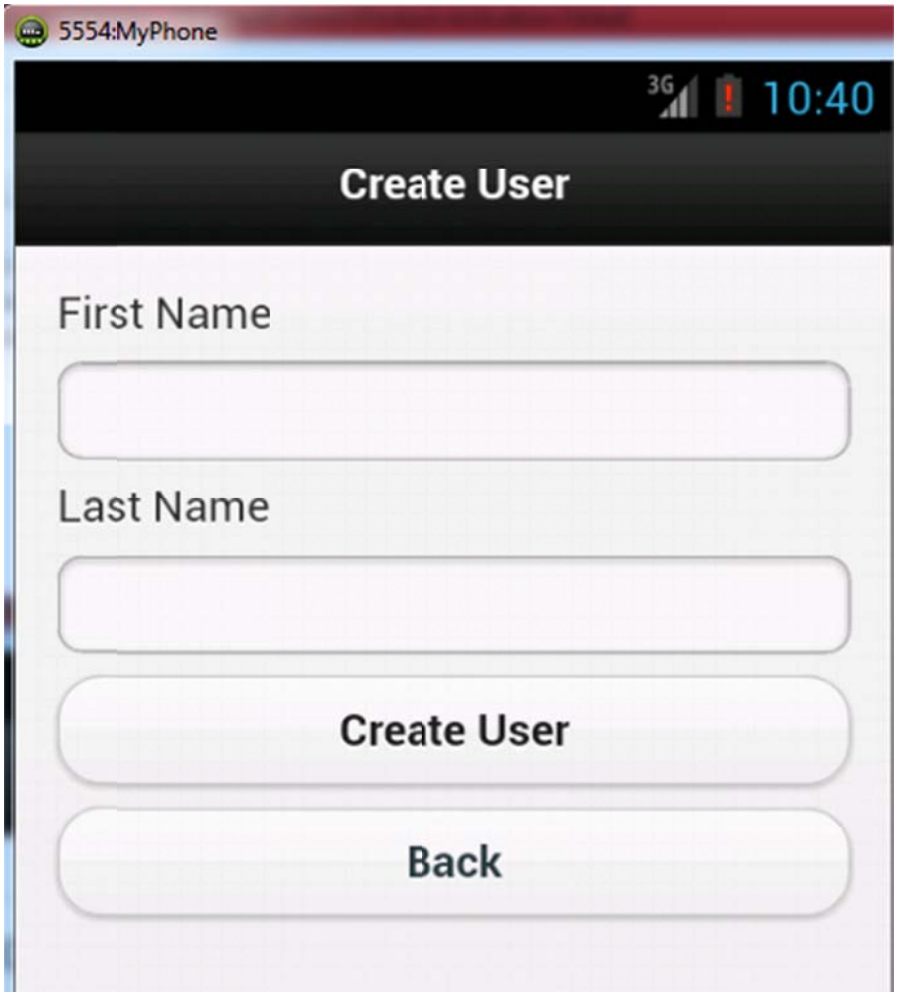
```
<!DOCTYPE html>
<html>
        <head>
            <meta charset="utf-8">
            <title>Create User</title>
            <meta name="viewport" content="width=device-width, initial-scale=1">
            <link rel="stylesheet" href="jquery.mobile.min.css" />
            <script src="jquery.min.js"></script>
            <script src="cordova.js"></script>
            <script src="jquery.mobile.min.js"></script>
            <script type="text/javascript">

                var rootURLusers="http://192.168.56.1:8080/PhonegapEx/jaxrs/Dao/CreateUser";

                function CreateUser(firstName, lastName) {
                    $.ajax({
                        type: 'POST',
                        url: rootURLusers + '/' + firstName + '/' +  lastName ,
                        success: function (msg) {
                                navigator.notification.alert("Create successfully",alertDismissed,
                        }
                    });
                }
                function alertDismissed() {}


                document.addEventListener("deviceready", onDeviceReady, false);
                function onDeviceReady() {
                    $("#Create").bind("click", function(){
                        var firstName = $("#first").val()+"";
                        var lastName = $("#last").val()+"";
                        CreateUser(firstName, lastName);
                    });
                }
            </script>
        </head>
        <body>
            <div data-role="page" id="signup">
                <div data-role="header">
                    <h1>Create User</h1>
                </div>
                <div data-role="content">
                    <label>First Name</label>
                    <input type="text" name="first" id="first"/>
                    <label>Last Name</label>
                    <input type="text" name="last" id="last"/>
                    <a href="#" data-role="button" id="Create">Create User</a>
                    <a href="Main.html" data-role="button" id="Main" rel="external">Back</a>
                </div>
            </div>
        </body>
</html>
```

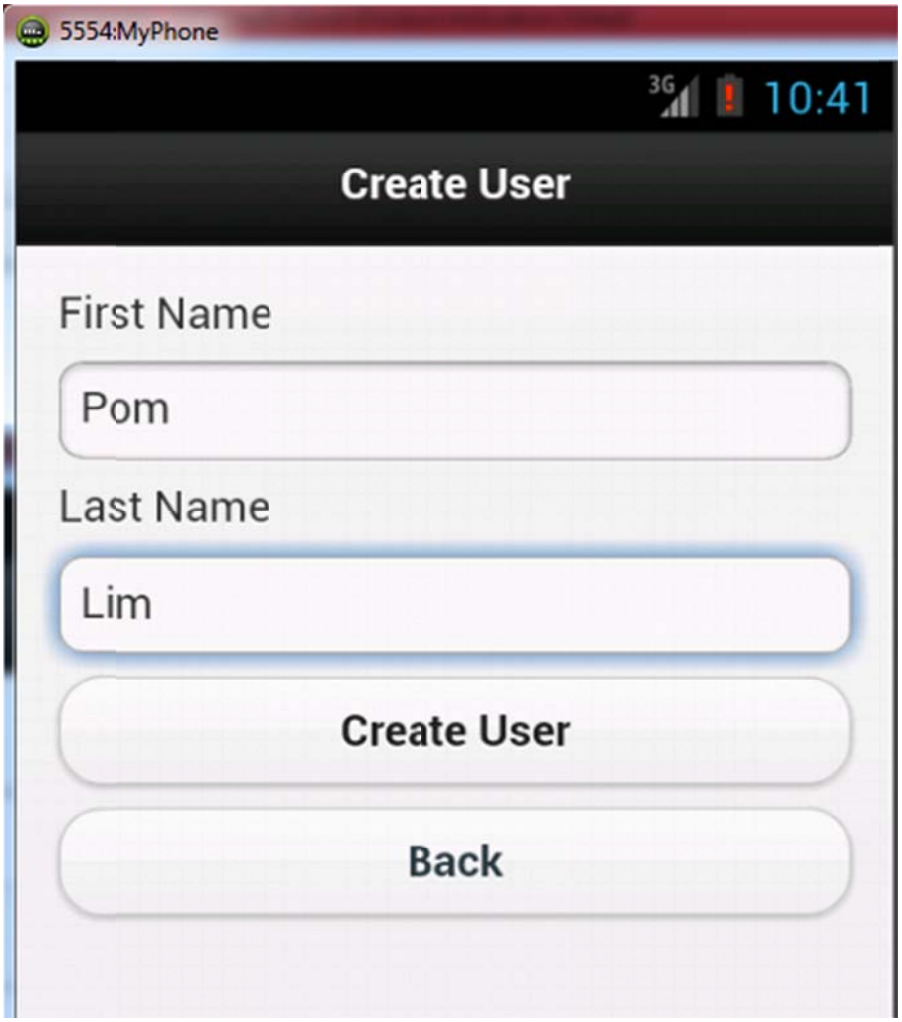


Run the program

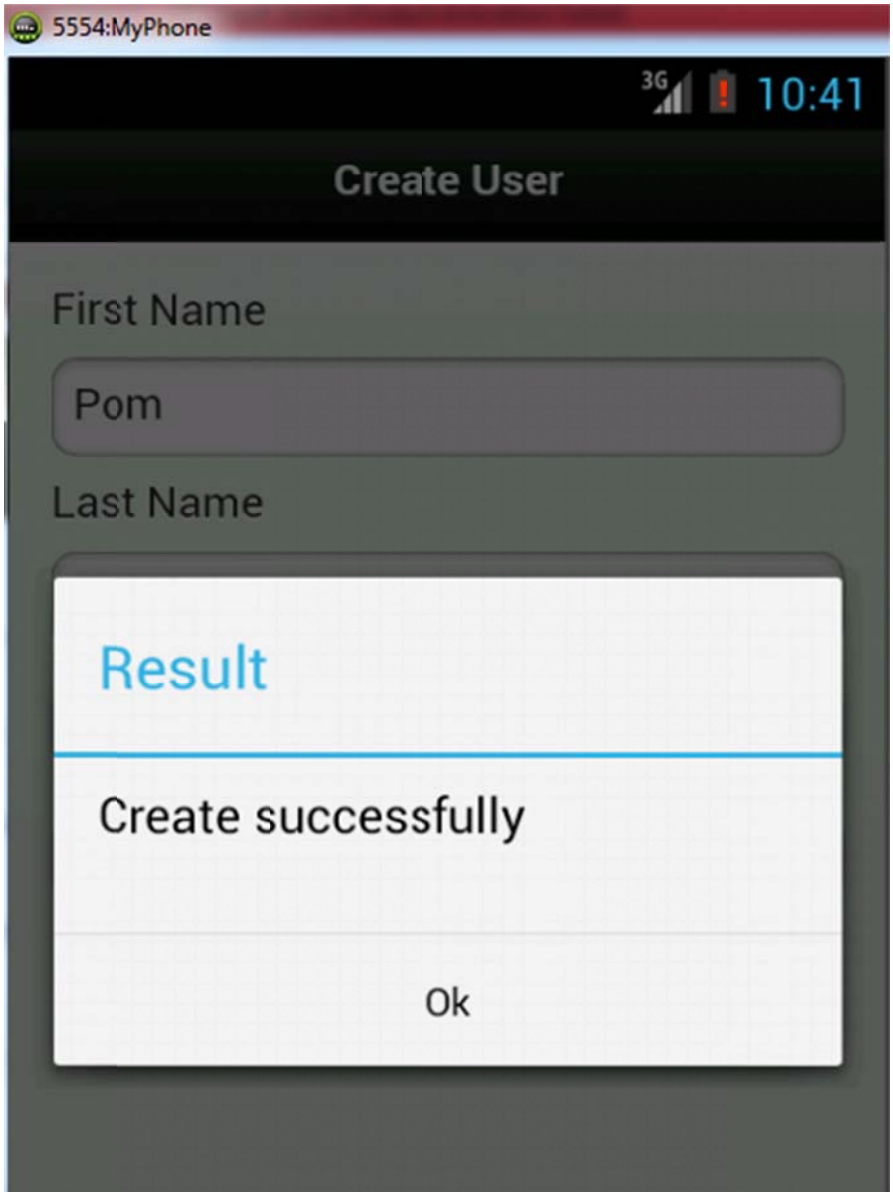


Click Create New User


5554:MyPhone
3G
10:40
Create User
First Name
Last Name
Create User
Back

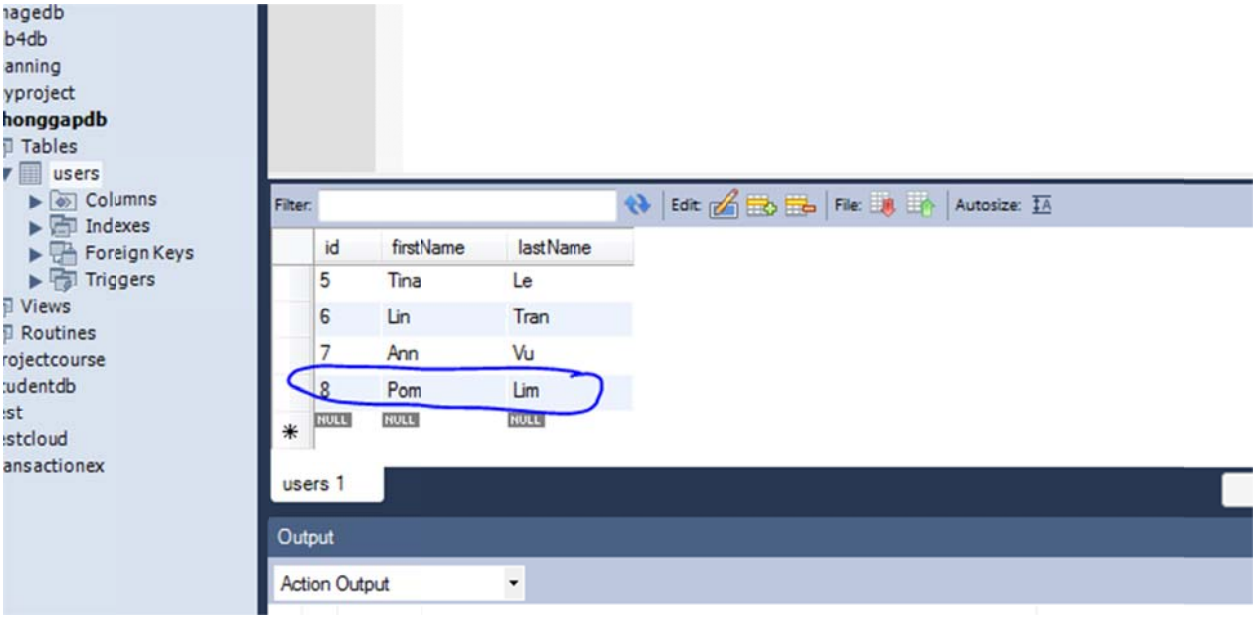


Click Create User

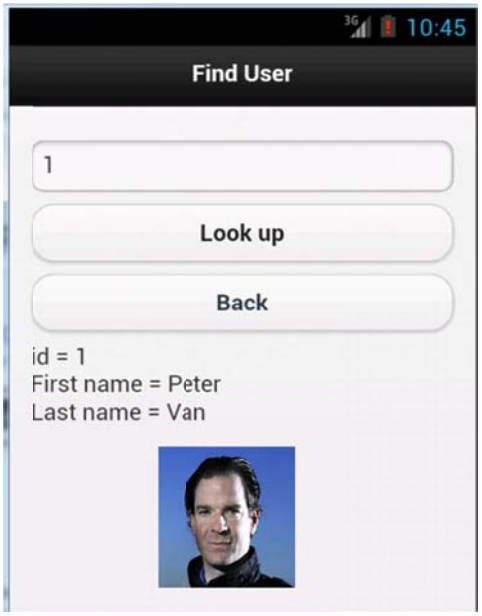


Click Ok

Check database



The above example can be added extended more such as add more column (image → allow upload image and retrieve image)



Or save information of users from server to contact database of the phone by using phonegap api : `navigator.contacts.create();…..`