Android Technical Manual

Introduction

The Android App for our website Intranet Redefined provides some basic feature for the Android mobile Users.

Development

The App was developed in Microsoft Windows Operating System (Windows 7/8), using Eclipse Kepler 4.3 and Android Software Development Kit and Java JRE. These are the basic and minimum requirement for further development and debugging of this project.
The languages used is Java for making the Android App and MySql and Php for writing the queries and connecting with the database.

Workflow

The Android app will be installed on the mobile with the name Intranet. It is is built keeping in mind the Android SDK 19. However it also works well with the lower versions till Android SDK 10.

The Applications open into a Login Page which asks for a user_id and password. The User can atmost make 3 attempts to login into the app else the permissions will be declined. All the further pages require this data to access other information of the user from the database.

The App extracts the data from the database “testdatabase”. It calls the following php script in order to extract information and check if the user and password exist in the database.

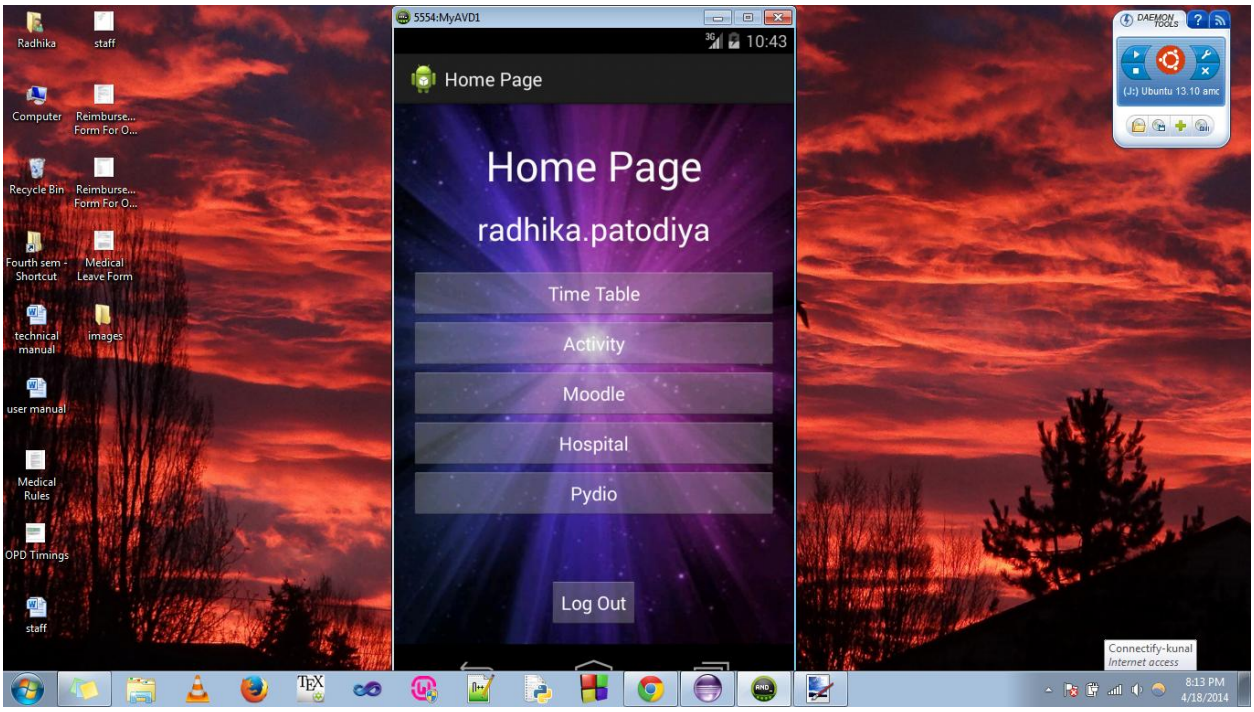
```
try{
                        HttpClient httpclient = new DefaultHttpClient();
                        HttpPost httppost = new
HttpPost("http://10.0.2.2/database.php"); //YOUR PHP SCRIPT ADDRESS
                        HttpResponse response = httpclient.execute(httppost);
                        HttpEntity entity = response.getEntity();
                        isr = entity.getContent();
                        Log.e("Pass1","connection success");
                }
```

```
                catch(Exception e){
                        Log.e("log_tag", "Error in http connection "+e.toString());
                        resultView.setText("Couldnt connect to database");
                }
                //convert response to string
            try{
                        BufferedReader reader = new BufferedReader(new
InputStreamReader(isr,"iso-8859-1"),8);
                        StringBuilder sb = new StringBuilder();
                        String line = null;
                        while ((line = reader.readLine()) != null) {
                                sb.append(line + "\n");
                        }
                        isr.close();

                        result=sb.toString();
                        Log.e("pass2","connection success");
                }
                catch(Exception e){
                        Log.e("log_tag", "Error  converting result "+e.toString());
                                                }
```

The string result will have the required strings to compare with the user_id and password entered by the user.
The php code running in background is as follows:

```
<?php

$con=mysqli_connect("localhost","root","root","testdatabase");

if (mysqli_connect_errno()){

  echo "Failed to connect to MySQL: " . mysqli_connect_error();

}

$sql="SELECT * FROM student";

$result = mysqli_query($con,$sql);

while($row=mysqli_fetch_assoc($result))

{      $output[]=$row;

}

print(json_encode($output))

mysqli_close($con);

?>
```

This will redirect the user to the user Home Page.



The home page have the buttons to redirect the user to either of the following options: Timetable, Activity, Moodle, Hospital, and Pydio.

The data is sent from one activity to other using the following code snipets:

```
Intent j2=new Intent(MainActivity1.this,
com.example.my_first_database.MainActivity3.class);
                        j2.putExtra("user_name", var_name);
                        j2.putExtra("user_dept", var_dept);
                       startActivity(j2);
                       Log.e("pass6","connection success");
                       finish();
```

and

```
String var_name = this.getIntent().getExtras().getString("user_name");
             String var_dept = this.getIntent().getExtras().getString("user_dept");
```

Timetable:

It provides the user with the day-wise timetable of their department. The php running in background is as follows:

```
<?php

$con=mysqli_connect("localhost","root","root","testdatabase");

// Check connection

if (mysqli_connect_errno())

{  echo "Failed to connect to MySQL: " . mysqli_connect_error();

}

$dept = $_GET['deptname'];

$day = $_GET['day'];

//$user="Radhika";

$sql="SELECT * FROM timetable where (department='".$dept."' && day = '".$day."')";

$result = mysqli_query($con,$sql);

while($row=mysqli_fetch_assoc($result))

{      $output[]=$row;

}

print(json_encode($output));

mysqli_close($con);

?>
```

The app will send the username and the department of the user to extract the relevant information by the following code:

```
HttpClient httpclient = new DefaultHttpClient();
                        HttpPost httppost = new
HttpPost(http://10.0.2.2/database1.php?deptname=”+department+”&day=”+weekday+””);
//YOUR PHP SCRIPT ADDRESS
```

The user can also access information of other days by using a spinner. The code for spinner is as follows:

```
Spinner day;

	private final String[] days = { "Monday", "Tuesday", "Wednesday",
			"Thursday", "Friday" };
	private int d_index = 0;
	private ArrayAdapter<String> dayAdapter;

day = (Spinner) findViewById(R.id.spinner1);

		dayAdapter = new ArrayAdapter<String>(this,
				R.layout.support_simple_spinner_dropdown_item, days);

		day.setAdapter(dayAdapter);

		day.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
			@Override
			public void onItemSelected(AdapterView<?> parent, View view,
					int position, long id) {
				String var_name = getIntent().getExtras()
						.getString("user_name");
				String var_dept = getIntent().getExtras()
						.getString("user_dept");

				d_index = position;
				String d = days[d_index];
				send_data(var_name, var_dept, d);

			}
```

The spinner will send the selected data to send_data functions which will extract the relevant information about timetable from the database.

Activity:

This will provide the user with the activities going on in the campus according to the current date.

The functioning of this module is similar to the functioning of the Timetable module. The salient feature is that it extracts the current date using the following function:

```
String date = new SimpleDateFormat("yyyy-MM-dd").format(new Date());
```

Hospital

This will redirect the user to the hospital user_homepage of our website. This is achieved by the following code snippet:

```
try{
                                    Intent browserIntent = new
Intent(Intent.ACTION_VIEW, Uri.parse("http://10.0.2.2/hospital/user/user_home.php"));
                                    startActivity(browserIntent);
                                    Log.e("pass3","connection success");
                                    }
                                    catch (Exception e){
                                          Toast.makeText(getApplicationContext(),
"Problem Loading...",

Toast.LENGTH_SHORT).show();
                                    }
                  }
```

Moodle:

This feature redirects the user to the Moodle App if its installed on his Android Phone. Else it will send the user to PlayStore to download the same. This was achieved by the following code:

```
try{
                                    Intent intent =
getPackageManager().getLaunchIntentForPackage("com.moodle.moodlemobile");

                                                startActivity(intent);
                                                finish();
                                    }
                                    catch (Exception e){
                                          Intent intent = new
Intent(Intent.ACTION_VIEW, Uri.parse("market://details?id=com.moodle.moodlemobile"));
                                          startActivity(intent);
                                          finish();
                                    }
                  }
```

However, when testing on Android Emulator, this gives error as none of the above two apps are installed on emulator.

Pydio

This feature redirects the user to the Pydio App if its installed on his Android Phone. Else it will send the user to PlayStore to download the same. This was achieved by the following code:

```
try{
                                    Intent intent =
getPackageManager().getLaunchIntentForPackage("com.pydio.android.Client");

                                                startActivity(intent);
                                                finish();
                                    }
                                    catch (Exception e){
                                          Intent intent = new
Intent(Intent.ACTION_VIEW,
Uri.parse("market://details?id=com.pydio.android.Client"));
                                          startActivity(intent);
                                          finish();
                                    }
```

However, when testing on Android Emulator, this gives error as none of the above two apps are installed on emulator.

Scope Of Improvement

The project very limited features due to time and resources constraint. Many features can be implemented in the project.

- The app is hosting its own database. It can be connected to the database of website to make it more efficient and admin-friendly.
- The app lacks features like live chatting, SAC information, discussion forums, Electives etc.

References

- http://www.tutorialspoint.com/android/
- http://developer.android.com/
- http://stackoverflow.com/