

Final Project: CTB: Android – Java Servlet – Jdbc-Odbc

This project has 2 major Modules:

Module 1: Android Project

Module 2: Java Servlet for Business Processing and Db Connectivity.

Specifications for Module 1: Android Project:

Objective: The Android app should allow the Job-Seeker or candidate to Login and search for Jobs. It is (minimum) expected that the App should have 3 screens (activities), namely:

- 1> **Splash Screen:** Showing an Image for 5 secs. After 5 secs the control would go to Login screen. (**// refer Code snippet -1**).
- 2> **Login Screen:** The user credentials (username and pwd) should be verified by contacting DB server through Java servlets. (**// refer Code snippet -2**).

If the User is authenticated, control should be transferred to the 3rd Screen (SearchJob_By_JobTitle)

- 3> **SearchJob_By_JobTitle Screen :** Here the user (or candidate) would enter the Job title and search should happen by contacting DB server through Java servlets. **Only 1 Job posting relevant to the title Keywords should be shown in the TextView.**

Final Project: CTB: Android – Java Servlet – Jdbc-Odbc

// refer Code snippet -1

```
// Create Thread that will sleep for 5 seconds
Thread background = new Thread() {
public void run() {

try {
// Thread will sleep for 5 seconds
sleep(5*1000);
// After 5 seconds redirect to another intent
Intent i=new Intent(getApplicationContext(),Login.class);

startActivity(i);

//Remove activity
finish();

} catch (Exception e) {

}
}
};
```

Final Project: CTB: Android – Java Servlet – Jdbc-Odbc

// refer Code snippet -2

```
new Thread(new Runnable() {
public void run() {
try{
URL url = new
URL("http://10.0.2.2:8084/JobPortal/ValidateServlet");
URLConnection connection = url.openConnection();

String inputString = inputValue.getText().toString();
String mypass = pass.getText().toString();

connection.setDoOutput(true);

OutputStreamWriter out = new
OutputStreamWriter(connection.getOutputStream());

out.write(inputString+", "+mypass);

out.close();

BufferedReader in = new BufferedReader(new
InputStreamReader(connection.getInputStream()));

String returnString="";

numberOfRecords=0;

// This loop runs Just Once
while ((returnString = in.readLine()) != null)
{
numberOfRecords = Integer.parseInt(returnString);
}
```

Final Project: CTB: Android – Java Servlet – Jdbc-Odbc

```
in.close();
```

```
runOnUiThread(new Runnable() {  
    public void run() {  
        inputValue.setText("");  
        pass.setText("");  
  
        Display(numberOfRecords);  
    }  
});  
  
} catch (Exception e)  
{  
    Log.d("Exception", e.toString());  
}  
}  
}).start();
```

We can also run the N/w connectivity operation of the UI thread , by using the method :

runOnUiThread(Runnable Object)

It can also be done by using AsyncTask (as learned at Level 1 Android Programming)

Final Project: CTB: Android – Java Servlet – Jdbc-Odbc

Specifications for Module 2: Java Servlet for Business Processing and Db Connectivity.

On coding Module 1 , it is Clear that Validating the User and Searching according to Job titles is done by the servlets at the web server side.

2 servlets needs to be coded with following responsibility:

- 1> **ValidateServlet** : This Servlet searches in the Data base for valid username and password. If Authenticated return **#Records as String. This return value is fetched in android (as in code snippet – 2 above)**.
 - 2> **SearchJob_By_JobTitleServlet** : This Servlet searches in the Data base according to JobTitle. If many Jobs are posted with the same Job title , then the Job last posted (i.e. the record at the end of the JobsTable , matching the JobTitle) should only be shown.
-

Guidelines for doing the project:

- 1> All variable names in Servlet and Android Modules should be properly named relevant to their purpose.
- 2> All important code parts should have comments where ever needed.
- 3> A **readme.txt** file should be prepared along with the project. Explaining steps to configure (if any) and running the project.

Deadline: 4 days (6 hrs per day)