

Oracle FLEXCUBE Direct Banking

Mobile Blackberry Native Client Developer
Guide

Release 12.0.3.0.0

Part No. E52543-01

April 2014

Mobile Blackberry Native Client Developer Guide
April 2014

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax:+91 22 6718 3001

www.oracle.com/financialservices/

Copyright © 2008, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface.....	1
Intended Audience.....	1
Documentation Accessibility	1
Access to OFSS Support	1
Introduction	1
Structure	2
Related Information Sources	2
Blackberry Native Client	3
System Requirements.....	3
Technical Architecture	4
FcdbApp	4
Transaction Screen.....	4
Menu Screen	5
Menu Classification.....	5
Handlers	5
HTTP Worker.....	6
Resource Bundle	6
Navigation.....	6
Error / Success / Warning messages	6
Installation.....	6
BLACKBERRY NATIVE Client Configurations.....	10
Application Flow	11
ATM/Branch Locator.....	12
Development Environment	13
Steps for Project Set-up in Eclipse.....	13
Creating JAD, ALX and COD	23
Automated Build for Mobile Client	24
Related Information Sources	25
XML Structure	25

Intended Audience

Any interested party working on the delivery of Oracle FLEXCUBE Direct Banking may read this document. The following profile of users would find this document useful:

- Application Architects
- End to End Designers
- Business Service Detailed Designers and Developers
- Implementation Partners

Specifically, however, this document is targeted at Implementation Partners, Customization Development Teams or Vendors providing customization, configuration and implementation services around the Oracle FLEXCUBE Direct Banking product.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to OFSS Support

<https://support.us.oracle.com>

Introduction

Mobile banking is a secure channel to access the banking accounts & financial transactions similar to the Internet banking channel.

FLEXCUBE Direct Banking provides various client applications that can be distributed by the Bank to their end customers. These client applications are to be installed on the mobile device of end customer and enable the customers to access banking facilities over the mobile device.

The mobile client communicates with FLEXCUBE Direct Banking application server in order to execute the Inquiry or payment transaction. The user has to enter his/her login credentials in order to access the functionalities.

The request is in the form of a HTTP request to the FLEXCUBE Direct Banking application server. And the response from the server is in the form of XML over HTTP. This XML is parsed on the client and rendered into appropriate screen display.

The client applications use network to communicate with FLEXCUBE Direct Banking application server. Hence network data charges may be applicable to the end user based on his/her network provider. In order to reduce the payload on the mobile, application is designed to transfer minimum data over the network. The response XML contains single character node and attribute names. These names are taken from the real HTML tags, e.g. H for hidden fields, T for table etc. This XML is in a pre-defined format which is produced by the mLEAP framework of FLEXCUBE Direct Banking.

The details of mLEAP Framework are available in
Oracle_FLEXCUBE_Direct_Banking_mLEAP_Framework_DeveloperGuide.docx.

For details on the supported XML tags please refer to document
Oracle_FLEXCUBE_Direct_Banking_Mobile_App_API.docx.

FLEXCUBE Direct Banking mobile clients are designed to understand the XML generated by mLEAP and convert it to appropriate screen definition to be rendered on mobile device screen.

Structure

This document provides a detail note on the technical architecture, development and packaging of the BlackBerry® Native Client.

Related Information Sources

Blackberry Native Client

BlackBerry® is a line of mobile e-mail and smart-phone devices developed and designed by Research In Motion (RIM) since 1999. BlackBerry phones function as a personal digital assistant and portable media player. They are primarily known for their ability to send and receive (push) Internet e-mail wherever mobile network service coverage is present, or through Wi-Fi connectivity.

RIM has published a proprietary BlackBerry® application programming interface (API) for application developers. Using this API enables the application developer to develop software that provides standard BlackBerry® navigation and appearance. It also allows the application to capture some of the BlackBerry® specific events during its run time, track ball or track pad related events for example.

System Requirements

- Blackberry phone running OS 5.0 or higher
- Network connectivity to FLEXCUBE Direct Banking server

Technical Architecture

Oracle FLEXCUBE Direct Banking Blackberry client is developed using BlackBerry® JDE 5.0.0. The application comes in the form of a COD file, which is in a proprietary BlackBerry® format.

The .COD file contains following components:

- Classes containing business logic such as handlers, HTTP worker, XML parser etc.
- Resources like images, properties that are used within the application.

The JAD file and ALX file is also released along with the application. The JAD file provides details about the application such as Name, Vendor, Version etc. It is useful while downloading the application Over the Air (OTA). The ALX file is used by BlackBerry® Desktop software when the application is to be installed on device from a desktop/laptop.

Below sub-sections describe various components used in the application

FcdbApp

The class `com.iflex.fcat.mobile.blackberry.app.FcdbApp` is the main class for this application. This is the starting point of the application.

Transaction Screen

This is the class responsible for rendering of each transaction initiated by the user. This class is available in package `com.iflex.fcat.mobile.blackberry.app`.

Various methods provided are as below:

- **PaintingScreenField:** Here all the screen components are added to the screen.
- **makeMenu:** This method is used for addition of menuitems(button).
- **NewMenuItem:** This class contains a method called 'run()' which is invoked when any menu item is clicked and necessary action is taken.
- **LoadingMethod:** this method is used to show loading screen.

- **pushNextScreen**: this method is used for checking if any error is on the screen otherwise next screen is called.

Menu Screen

This is the class responsible for rendering of Menu items on the screen. This class is available in package `com.iflex.fcat.mobile.blackberry.app`.

It also contains methods similar to `TransactionScreen` for rendering of components and navigation.

Menu Classification

This is the class responsible for rendering of submenu items on the screen. This class is available in package `com.iflex.fcat.mobile.blackberry.app`.

It also contains methods similar to `TransactionScreen` for rendering of components and navigation.

Handlers

Handlers are the components responsible for handling different kind of events.

Various handlers provided are as below:

- **KeyPadEventHandler**: This Handler is responsible of handling of trackwheel and Keypad .During Loading time , we disable the trackball and other keypad buttons with the help of this handler .This Handler is available in package `com.iflex.fcat.mobile.blackberry.app` .
- **customhandler**: This whole package contains custom handlers to create custom dialogue popup custom menu , submenu icons and default flow of the transactions.
- **MapHandler**: This Handler is used to display Blackberry map for branch and ATM locations.

HTTP Worker

The class `com.iflex.fcat.mobile.blackberry.httpworker.HttpConnectionWorker` is responsible for communication with FLEXCUBE Direct Banking server. This class creates a HTTP connection with the server using the URL provided in 'APP-SERVER-URL' property.

Resource Bundle

Resource Bundle in Blackberry is similar to properties file. It has key-value relationship which helps in defining our own properties from the outside of the code.

When you create a resource header file called 'ResourceFcdb.rrh' and corresponding Resource Content File called 'ResourceFcdb.rrc' file(s), the BlackBerry compiler will automatically generate an interface called `ResourceFcdbResource` in the same package (or to be specific, in the package corresponding to the directory your .rrh file is in).

This is done at compile time, so you don't actually get a corresponding .java file for that interface at any point. But the JDE and the JDE Plugin for Eclipse will also recognize when you save changes to your .rrh file and make those changes available to the editing environment so you don't get warnings while you're developing.

Navigation

The navigation is possible through various buttons provided. After logging in a main menu is displayed that shows list of available transactions.

Error / Success / Warning messages

Various messages like error, warning or success messages are displayed on the top of the transaction screen.

Installation

More help on the installation can be found in the BlackBerry documentation available at http://docs.blackberry.com/en/developers/deliverables/11938/Packaging_and_distributing_BB_Java_applications_512511_11.jsp

The installation is typically done in two ways:-

- Installing Third Party Software OTA (OVER THE AIR).
- Installing Third Party Software Via BlackBerry Desktop Manager.

3.3.10.1 INSTALLING THIRD PARTY SOFTWARE OTA (OVER THE AIR)

Performing the OTA download and installation of a third party application is usually the easiest option. Most application developers will provide direct links to the .jad or .jar files required for internal installation on your BlackBerry.

Downloading OTA is super easy. Simply navigate to the download page for the application you wish to install and find the direct (OTA) download link. From within the BlackBerry browser menu, select "Get Link" and then click OK for each step of the installation. Here are the steps you must perform in order to download a third party application OTA to your BlackBerry.

The BlackBerry smartphone has the ability to download and install applications over the wireless network. For a web server to allow a BlackBerry smartphone to download applications, it must be configured with the appropriate Multipurpose Internet Mail Extension (MIME) types. The following is a list of required MIME types that must be configured on a web server to allow the downloading and installation of BlackBerry smartphone applications.

- cod application/vnd.rim.cod
- jad text/vnd.sun.j2me.app-descriptor
- jar application/java-archive

However, if the COD file is too large (exceeds 64K code and/or 64K data), the JAD and COD may not be able to download over the air. The large COD size results in generation of additional COD files like myapp-1, myapp-2, etc. These are "zipped" up into the aggregate COD file.

For OTA deployment, you typically need to extract these and copy all of the resulting files to your deployment server.

STEPS:

1. Navigate with the BlackBerry Browser to the desired application download page.
2. Often you will be presented with a link to the download, or the option to enter your cellular phone number and have the link sent to your device via SMS.
3. Follow the supplied link (either direct or SMS) from within your BlackBerry Browser and follow all the instructions that appear on your screen.

4. The new application will be unpacked and installed by your BlackBerry Browser and your new software will be now be available in your applications list.

NOTE: If you are using your BlackBerry in a BES configuration which is overseen by your IT department, you may not have the ability to install your own third party applications to your device. If this is the case, you will have to contact your IT professional and request that they add the software and deploy it to your device. This may be against the policy that your employer has dictated so it is best to get approval before any such request.

3.3.10.2 INSTALLING THIRD PARTY SOFTWARE VIA BLACKBERRY DESKTOP MANAGER

Most of the time when you find a third party application that you wish to install on your BlackBerry they will be available in multiple formats. The OTA download is usually the easiest and most convenient way to install software; however, for those who face technical issues or for those without an adequate data plan, there is a more desirable solution.

Installing third party software on your BlackBerry via the BlackBerry Desktop Manager is a simple process. In only a few steps you will have all the applications you like installed on your device and ready to go.

The first step you need to take is to ensure that you have the latest version of the BlackBerry desktop Manager installed on your computer.

Once you have downloaded and installed your BlackBerry Desktop Manager, you are ready to try out your first third party software installation. In order to complete the installation, simply follow the steps outlined below:

STEPS:

1. Navigate to a download link for any piece of third party software you like and download the required installation files to your desktop.

Example: BBSmart Email Viewer (use the free trial as a test and choose the full download option)

2. Once you have downloaded the desired application to your desktop, you may have to unzip/extract the contents to a new folder. Inside your newly created folder you will find a file that ends in .alx

3. Connect your BlackBerry via USB to your computer and launch the BlackBerry Desktop Manager.

4. In the menu you will see an option for "Application Loader". Click this and you will see a screen appear that displays all the applications currently loaded on your BlackBerry.

- 5.** Above the list of currently installed BlackBerry applications you will see a button that says "Add". Click on this button to bring up the add application dialog box.
- 6.** Navigate to the folder that you have saved on your desktop in the previous step and locate the .alx file contained within. Choose OK and this application will appear in the list with all of your other installed software.
- 7.** If you have downloaded multiple pieces of software to install, you can go ahead and click the add button again and navigate to the next .alx file and click OK. Repeat this process for each additional third party application you wish to install.
- 8.** Once all desired applications have been introduced into the application loader, verify that they each have a check mark next to them and click the button labeled "Next" which appears below the list of applications.
- 9.** You will see the BlackBerry Desktop Manager run through a process and your BlackBerry may restart.
- 10.** Upon reboot of your BlackBerry you will see all of your newly installed applications on the applications screen of your device.

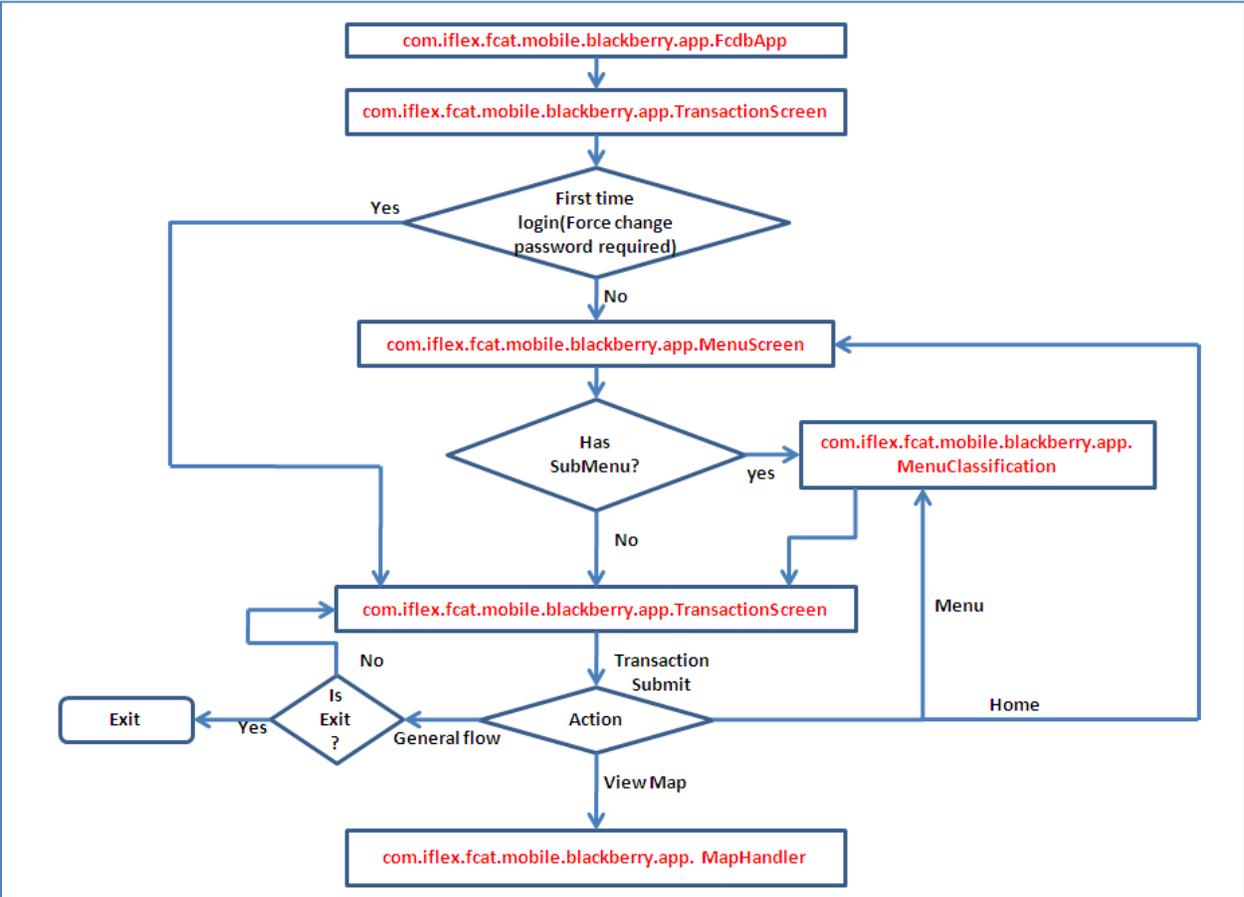
BLACKBERRY NATIVE Client Configurations

Various configuration files for the Blackberry Native client are as follows:

- **JAD files:** An application descriptor that stores information about the application itself and the location of .COD files
- **.JAR files:** a JAR file (or Java Archive) is used for aggregating many files into one. It is generally used to distribute Java classes and associated metadata.
- **.COD files:** A COD file is a proprietary file format developed by RIM that contains compiled and packaged application code.
- **.ALX files:** Similar to the .JAD file, in that it holds information about where the installation files for the application are located.

Application Flow

The FCDB Blackberry Application flow is mentioned in the below diagram.



ATM/Branch Locator

A special mention of ATM/Branch Locator is worth as the flow defers a bit for this transaction. The application needs to connect to RIM's internal blackberry apps i.e. Blackberry Maps. Overall flow for this transaction is as below:

- User clicks on the ATM/Branch Locator transaction available in the menu.
- User searches the desired location – a list of branches is displayed on the screen.
- User clicks on the 'View Map' button
- A request is sent to Blackberry Map apps with the coordinates of the branch.
- Blackberry Map returns image of the map, which is displayed on the device screen
- User can zoom in / zoom out the map. For every zoom in / zoom out operation, a request would be sent to Blackberry Map with appropriate zoom level. A new image of the map is received which will be displayed on the device screen.

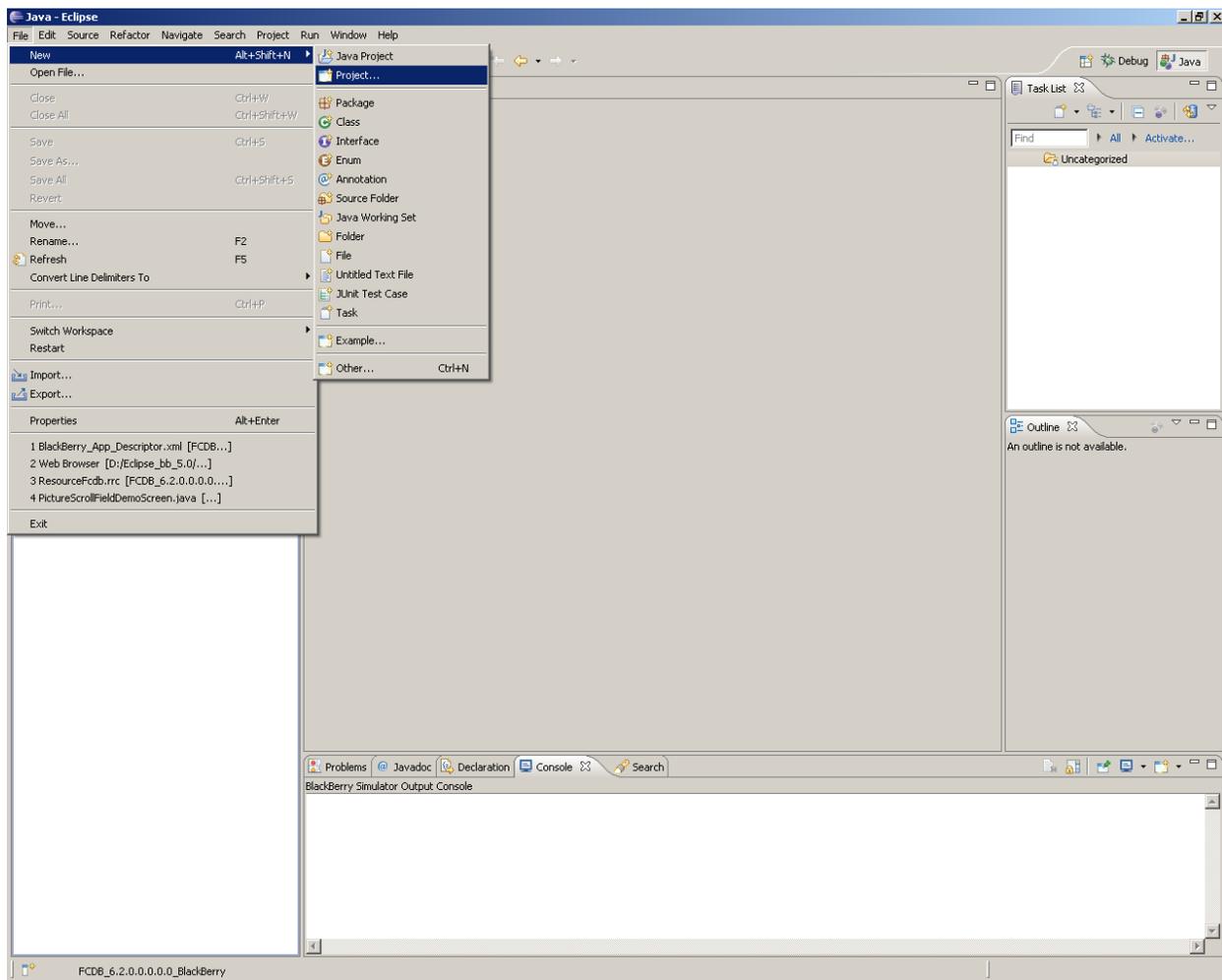
Development Environment

Introduction

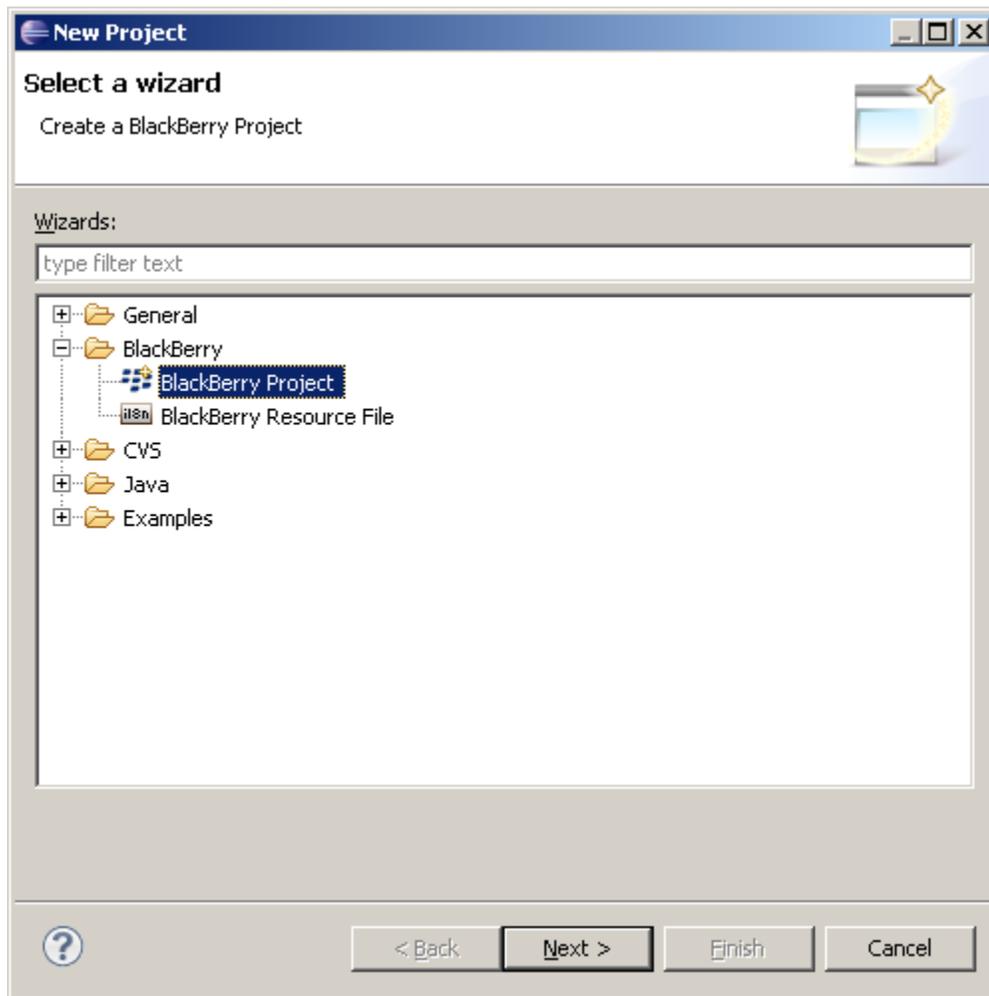
The eclipse that is distributed along with BlackBerry® JDE 5.0.0 is used to develop and configure the project for Blackberry native application.

Steps for Project Set-up in Eclipse

1. File -> New ->Project

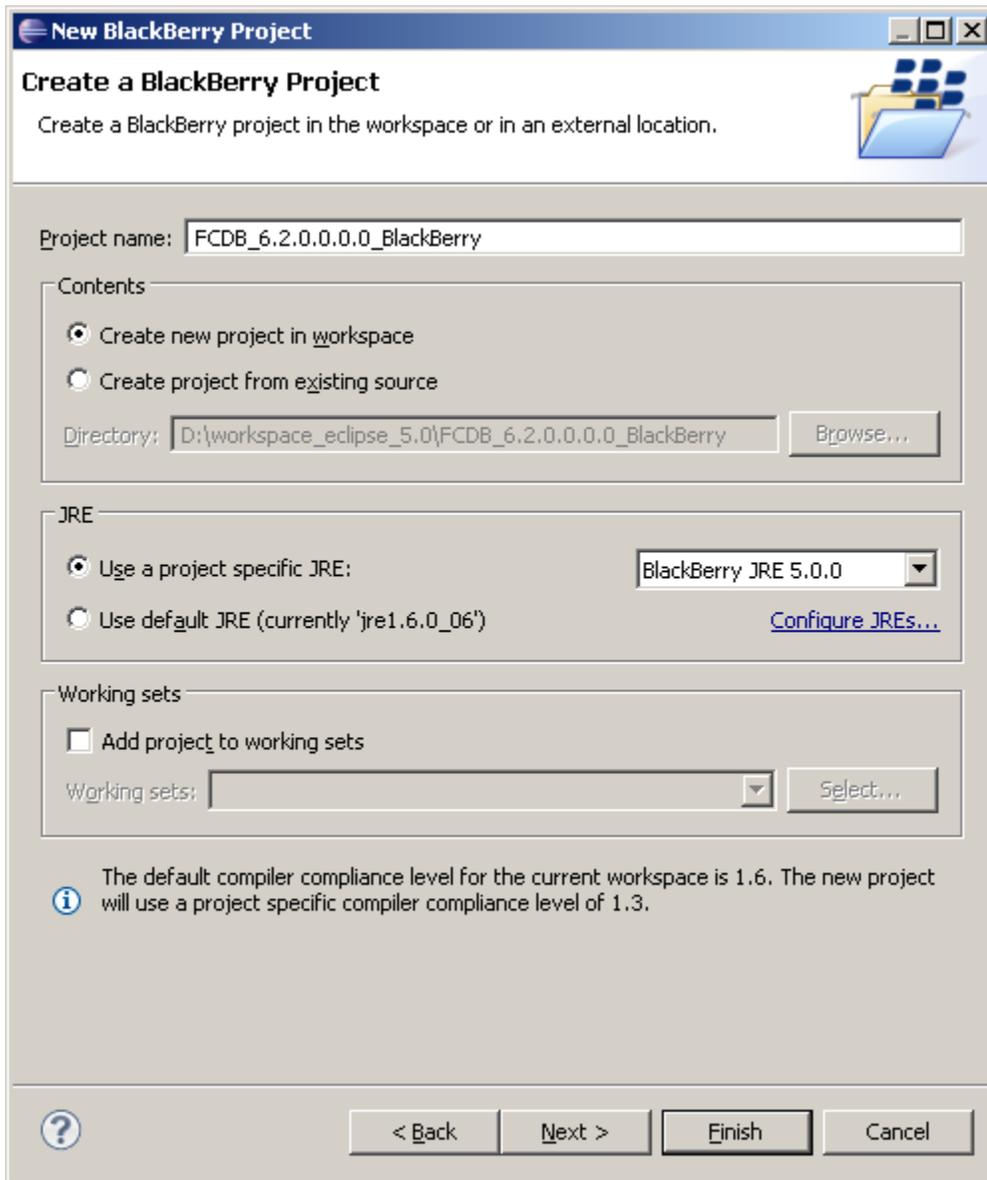


2. Select project as 'BlackBerry Project'. Click next.

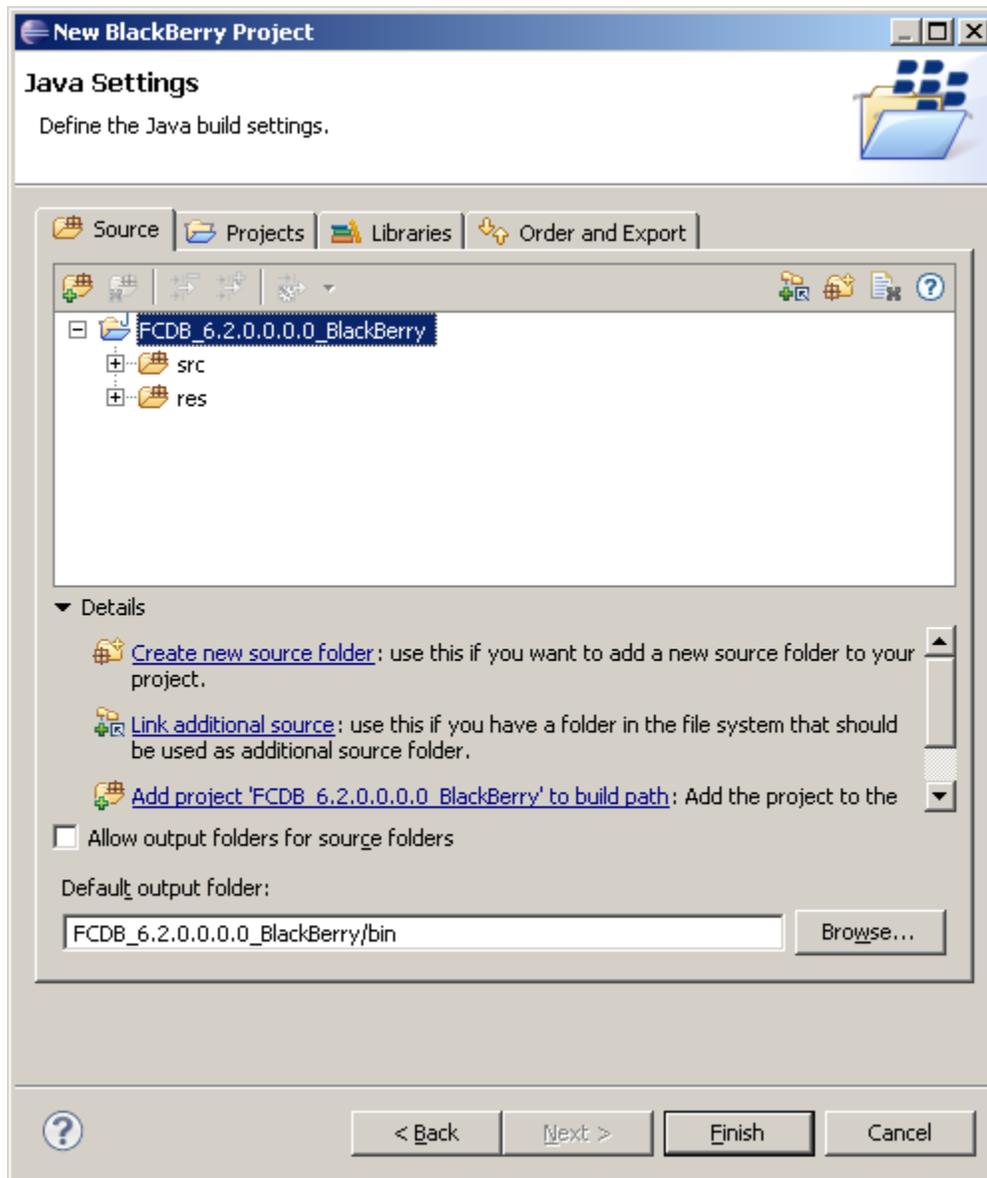


3.Mention Project name.

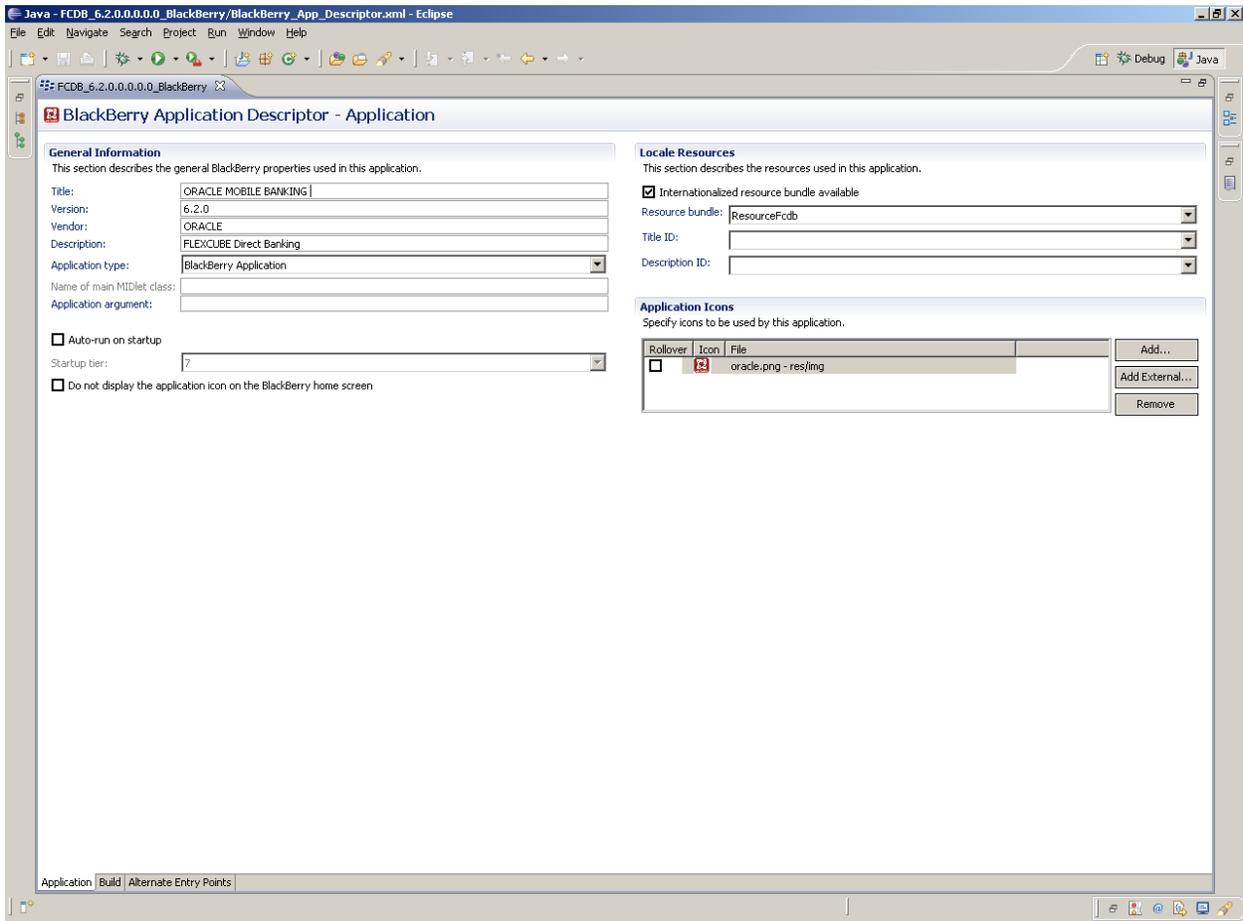
Choose project jre as “Blackberry JRE 5.0.0(version)”



4. Click Finish.



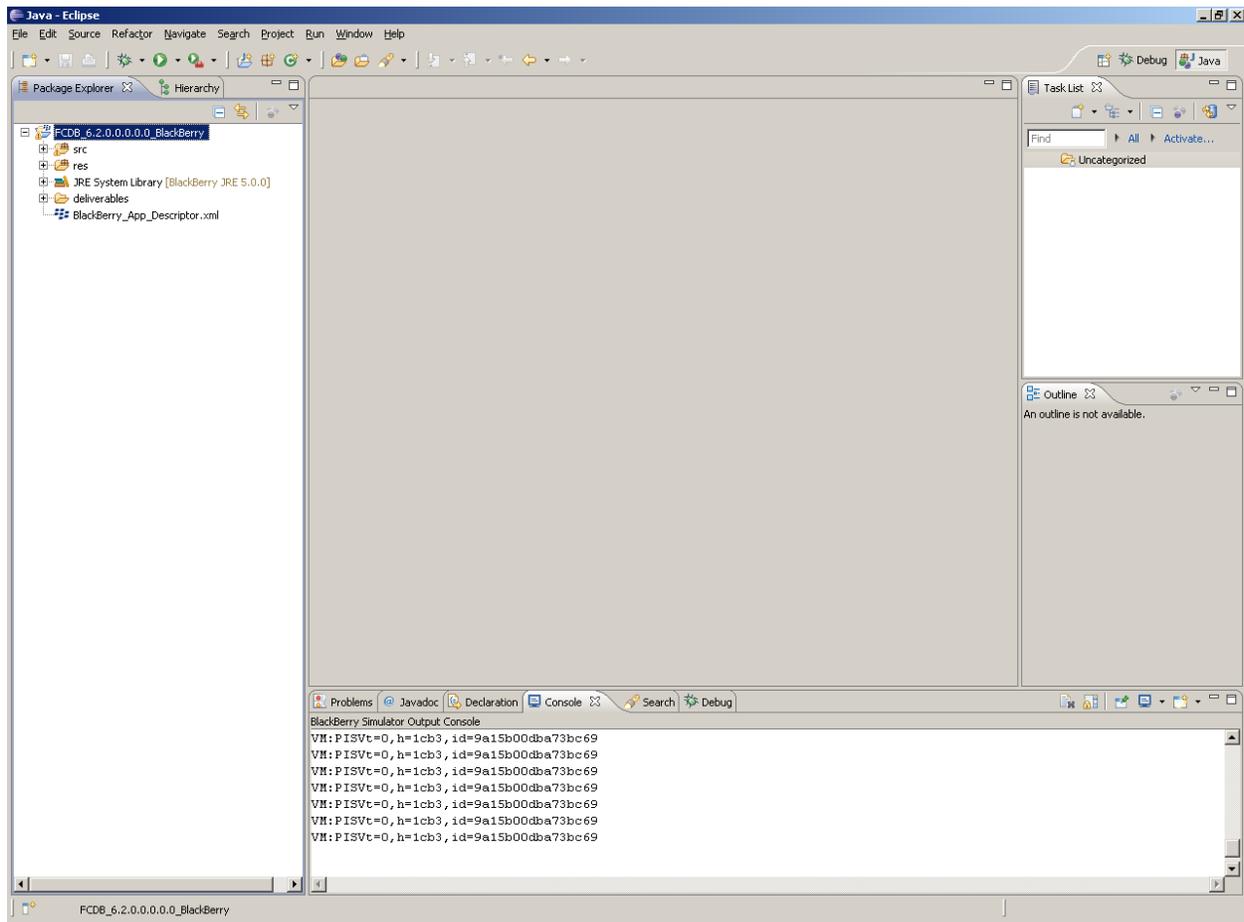
5. Give Details of project in the file called “BlackBerry_App_Descriptor.xml”.



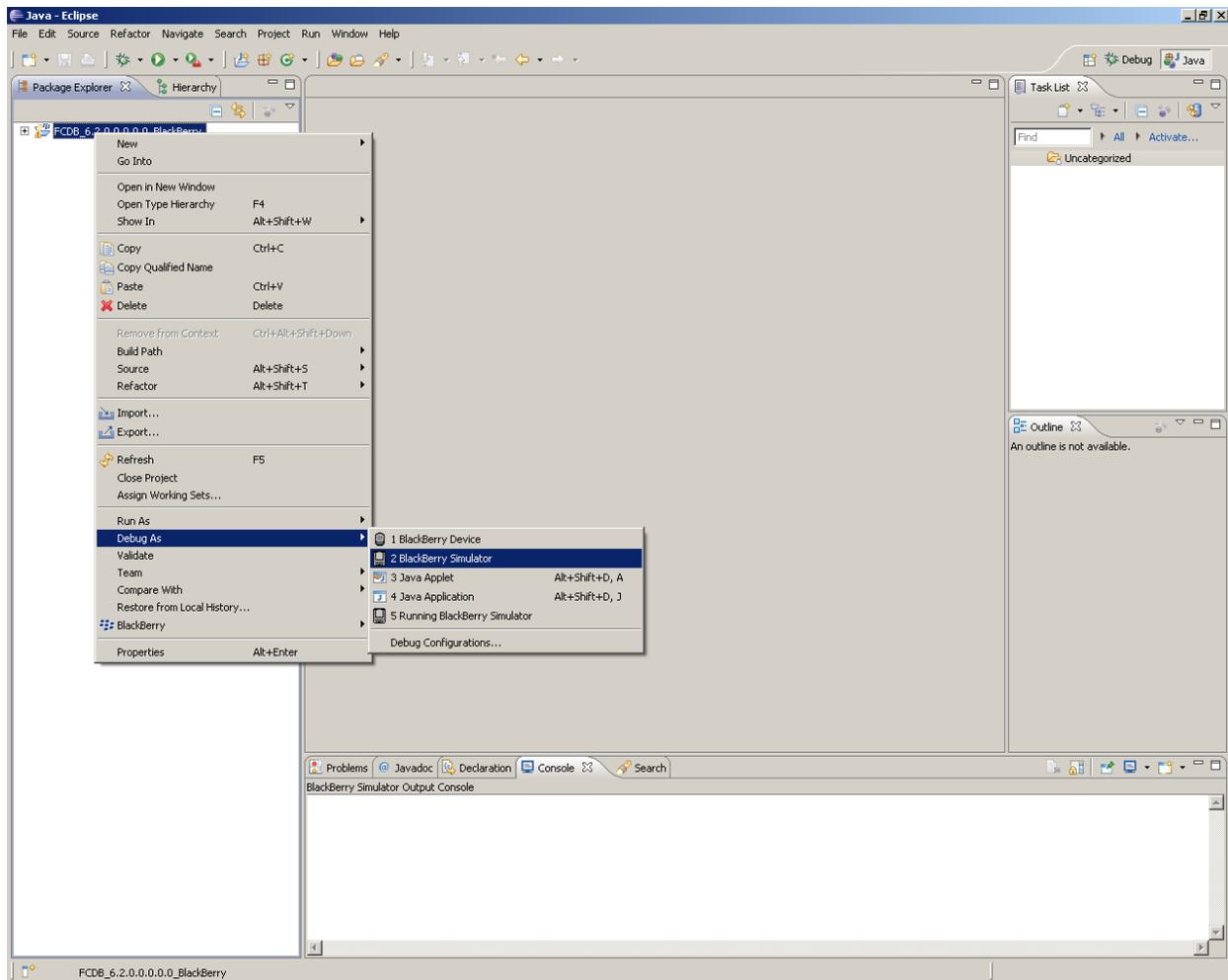
6. After successful creation of Project, 2 folders namely 'src' , 'res' will be created inside the base folder mentioned in step 3.

Copy the src folder to src.

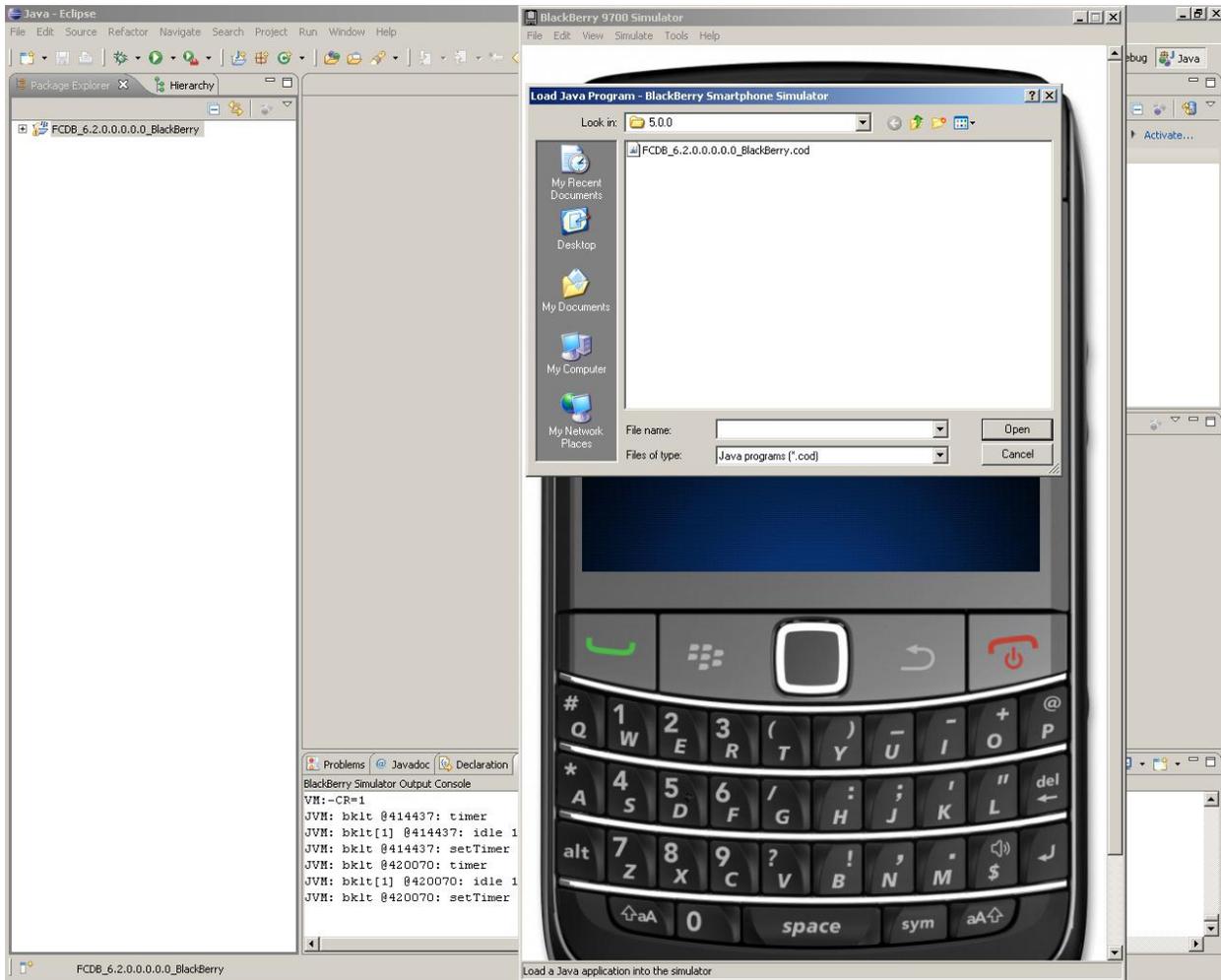
Copy images and resource file to res and refresh the project.

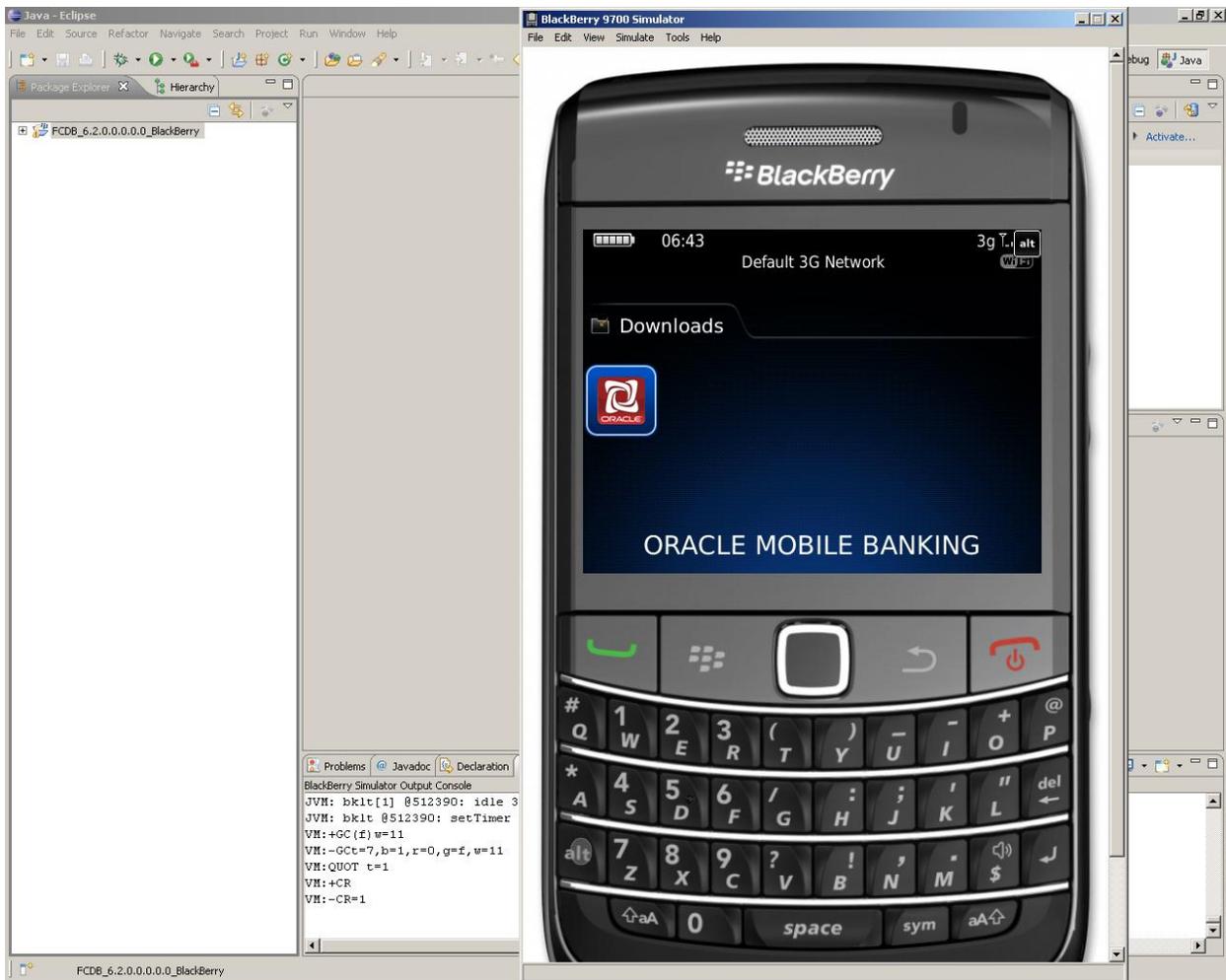


7. To Launch the Application::Right Click to Project -> Debug As->Blackberry Simulator.

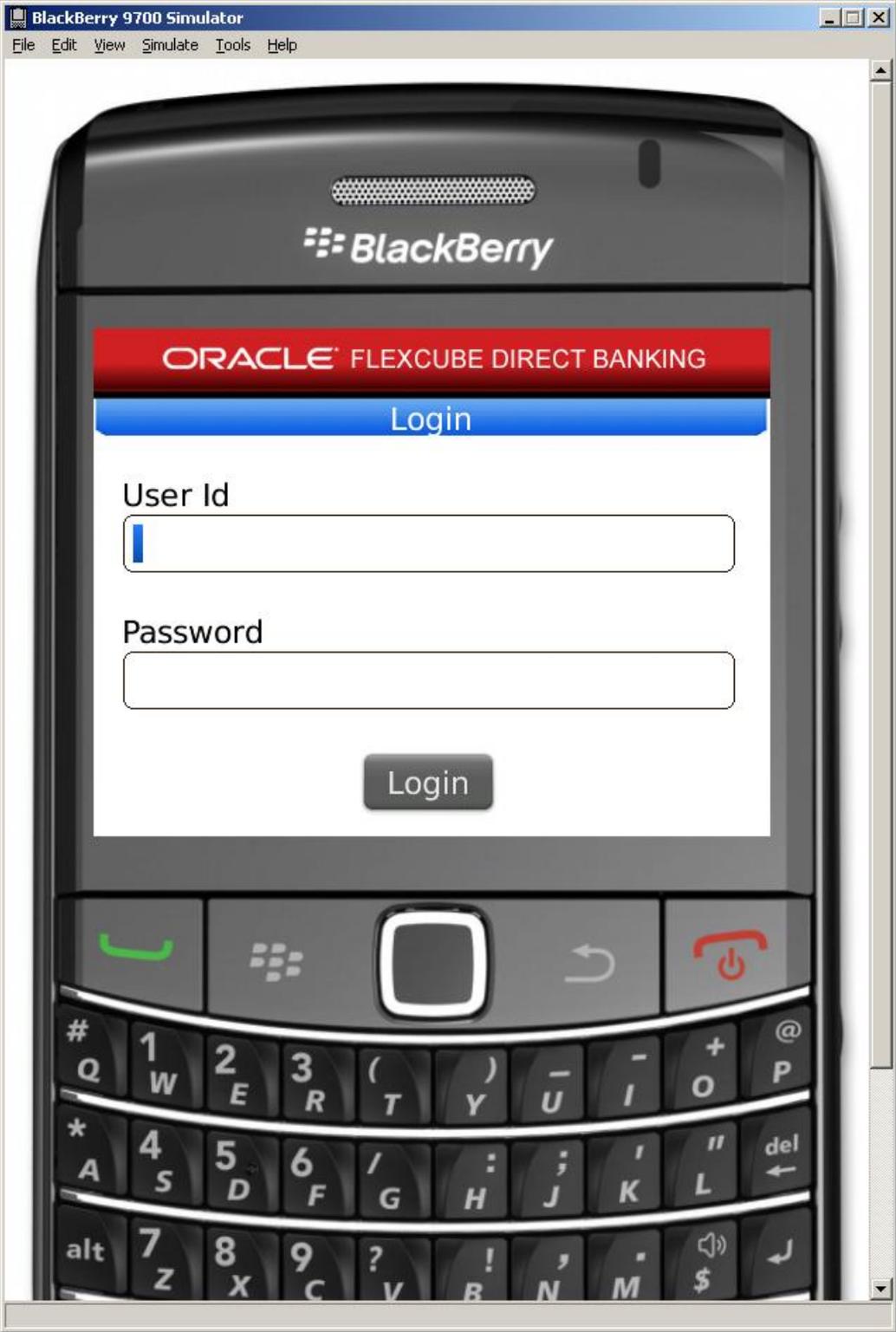


8. After debug, the application icon should appear in the downloads folder. If not, then Load the application by going to File of simulator --> Load Program.





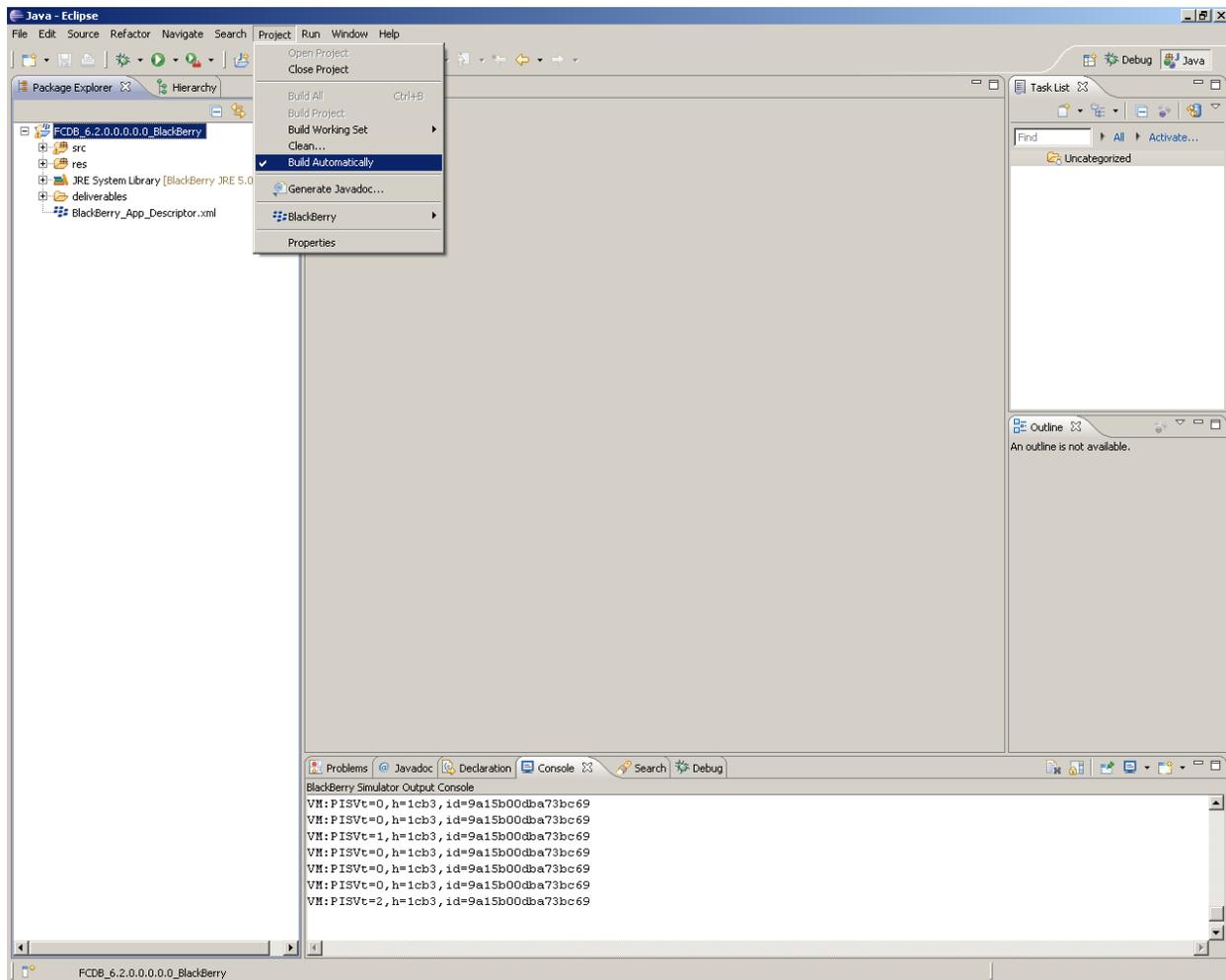
11. To 'Launch' application , Click the Trackball on the simulator to get the login screen.



Creating JAD, ALX and COD

Using Build option user can create JAR/JAD

Right click on project select Option Build.



JAR/JAD/COD will be created inside folder deliverables

Deliverables->Standard->5.0.0

Automated Build for Mobile Client

The build process of J2ME based mobile clients is automated using 'Ant' script. This process retrieves required code/resources from VSS and builds the deployable JAR/JAD file. Reference to the mobile build document

Oracle_FLEXCUBE_Direct_Banking_Mobile_Automated_Build.docx

Related Information Sources

XML Structure

The XML generated by mLEAP framework acts as input to mobile client. Detailed description of supported tags in this XML is provided in document

Oracle_FLEXCUBE_Direct_Banking_Mobile_App_API.docx.