**COIT20259**

**Enterprise Computing Architecture**

Introduction:

This manual is designed to guide you to use the ‘E business site’ product that you have received either in form of a war file or a bundle of source code. This guide explains the installation (plus compilation if required), deployment and use of the product. This manual also covers testing by using the real data and real time scenario.

This product is a three tier e-business enterprise application that works as a website. It can be usable as the standalone way or can be used as distributed (from many places) way. The product is made use of JSP and MVC architecture. MVC means model view control, model is the one interacting with the database controller consist of business logic and act between model and view, whereas view is the presentation part. All these three part are independent part i.e. changes made in any one of them will not affect the other that’s why this architecture gives handy control over the project maintenance.

Objectives:

The objective is to develop an enterprise application using 3 tiers for the electronic item sellers. There are lot of additional plus objectives which should have been taken into account to build e-business application such as

* Provide a convenient distributed way to do inventory of electronic products
* It will increase the performance of user by decreasing the overhead time
* It will a fast, simple and best way to access from anywhere as per the need of end user

Design:

The conceptual designing can be divided into 2 parts: data model and process model. There the data model focuses on data that should be stored in the database. And the process model focuses on the process of processing the data. The database thing comes in the context of data model whereas the query things comes in the context of process data model.

To use the MVC (Model View Controller), we can use Java Server Faces (JSF) and we will use it. By using the MVC, we should care of motive of MVC that is separate the models and presentation things so developers able to set focus only on their core skills specific to part of the system. Where the designers will concentrate only on presentation part rather than model part, on other side the developers will able to concentrate on model part only rather than view part. In this scenario both can work independently. In MVC, controllers controls the user actions and connects actions to model side.

Our product to be done on the base of three tier enterprise application which separated into following three tiers and same has been demonstrated in the preceding diagram.

* Persistence Tier
* Business Tier
* Presentation Tier

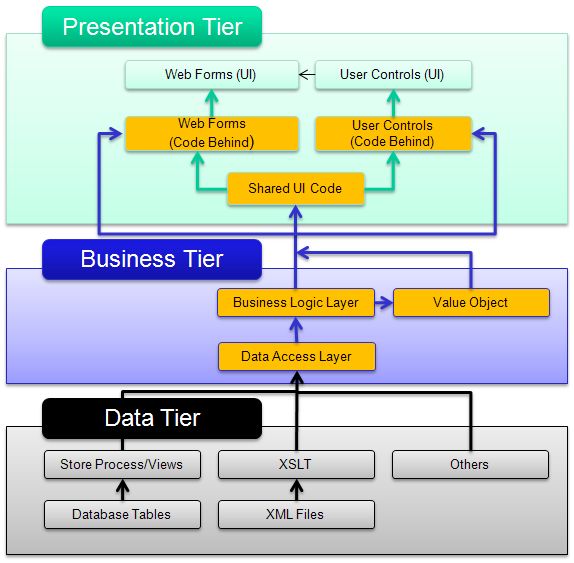


Figure : Architecture of Three Tier Enterprise System

From the business tier, controllers performs user actions. In presentation layer, the presentation can be managed by the authors manually or by the managed beans. There managed beans utilizes the business logics. The JSF provides the mechanism of managed beans and have lot of rendering components that can render the components as per the need of web designer. The developer never need to know the actions of designer.

Architecture of Application:

Architecture of required product is divided into the server and the client parts, as shown in diagram. The client side will contain the Java Server Faces (JSF) files and XHTML files. In the server side, several parts are there that interacts to each other to fulfil the requirement. There Data Access Object (DAO), provides an abstract view to data of database in form of objects and it uses the persistent mechanism to do. It maps the application calls from business layer to persistent layer. The DAO provides the data operations somehow similar to query operation on database but this technique provides a more efficient and abstracted way to access the database with lot of advantages including one most important advantage that is it does not expose the details of database.

When the user first interact with the application it sees a home page which is in xhtml and include java server faces code also from where it make request then comes the application layer where it plays a most important role in processing the request. We will be using glassfish derby database to store the data. We will make use of DAO i.e. data access object which interact with the database and application layer. EJB application is deployed on the server, so it is called server side component also. EJB consist of Session bean, Message bean, entity bean, we will make use of these entire component in our project.

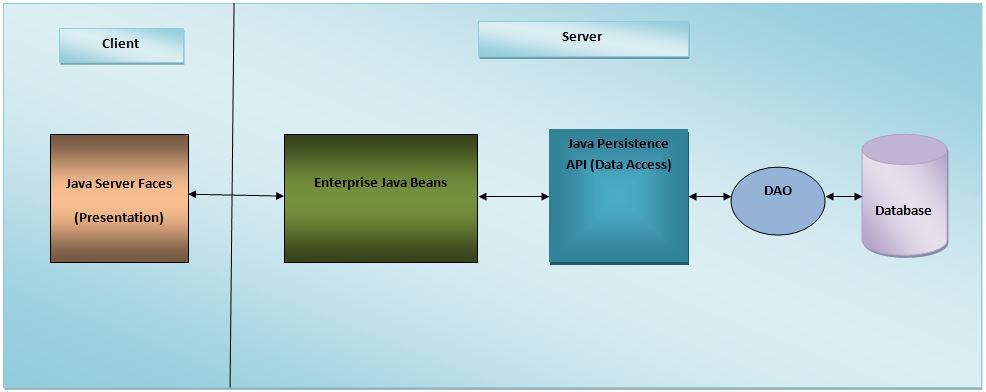


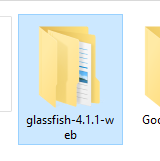
Figure : Architecture of the Application

# Installation (run/execute)

Follow the following steps to install the system and make it usable:

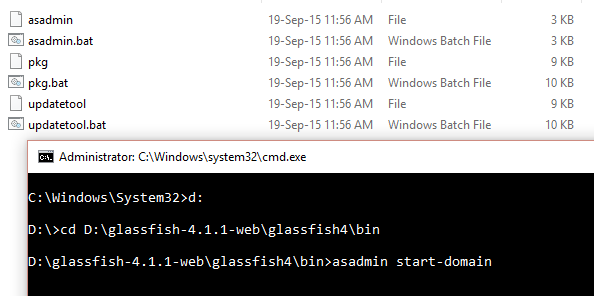
## Step 1

Get a setup file of glassfish server and install it. You can get it in zip file and extract to a folder.



## Step 2

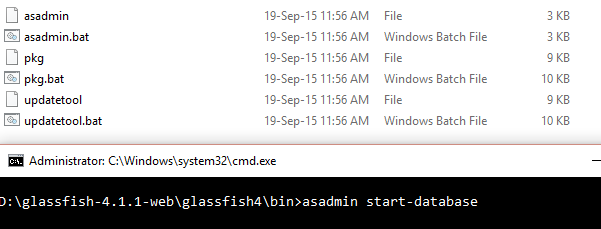
Go to bin directory of glassfish server and start the server. To do this, in terminal window do change directory operation and reach to bin and run asadmin by passing argument value ‘start-domain’ as shown in below snapshot:



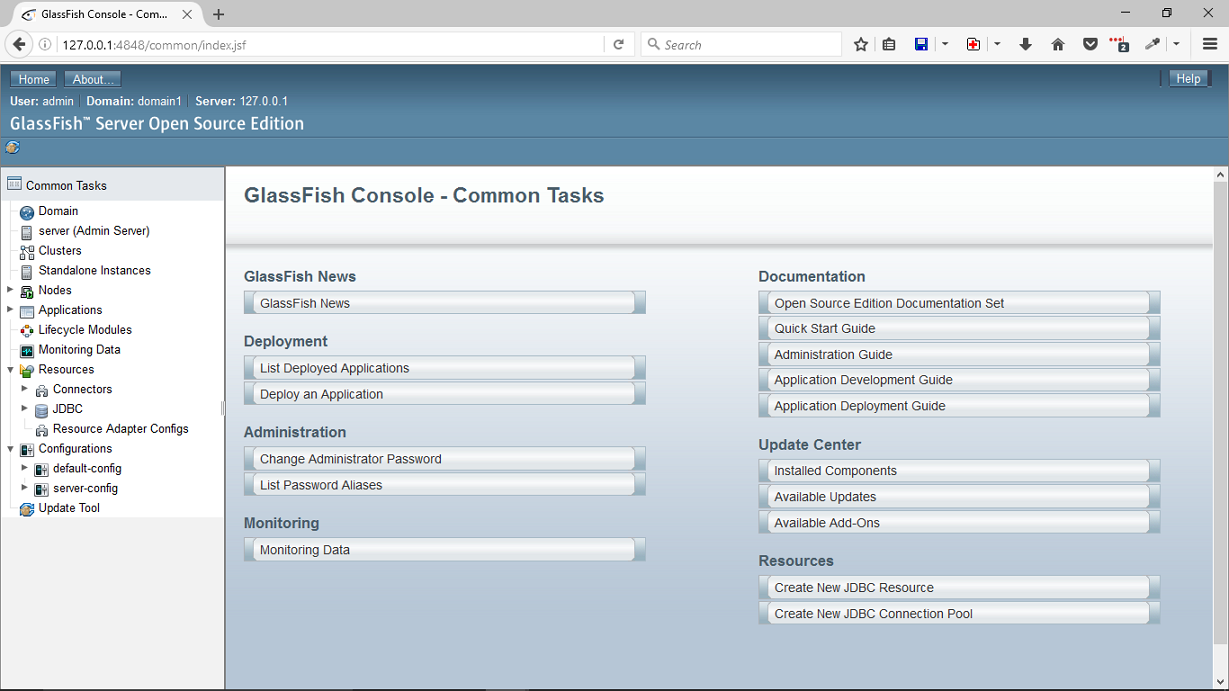
Fine. Server is up.

## STEP 3

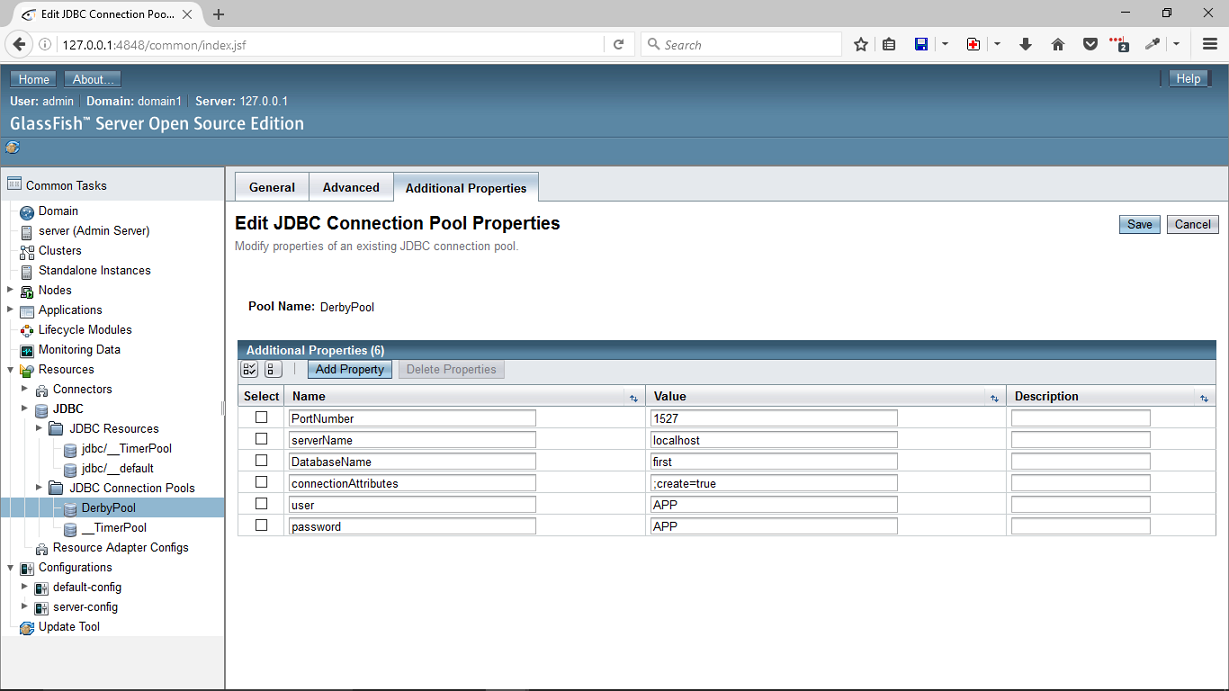
You will also need to have a database. This product uses derby database. You can use derby database and start the database server and do relevant setting on the server. You can use inbuilt derby database that comes with glassfish setup. To start this database, run asadmin by passing argument value ‘start-database’.



After having a running database, open the glassfish server’s administrator console in browser using your IP address and glassfish server’s default port number 4848 as shown below:

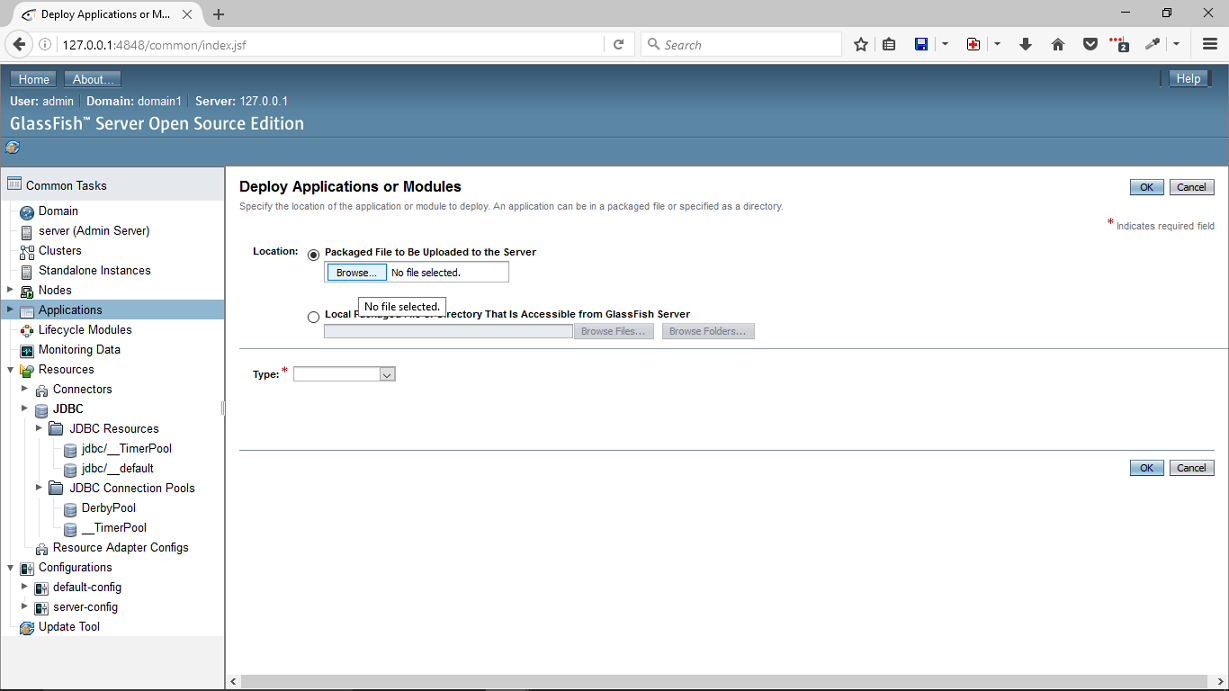


Do the database setting on server, it is optional. Product can work with default setting of glassfish inbuilt derby database. You can do modifications in database setting as shown in below snapshot.



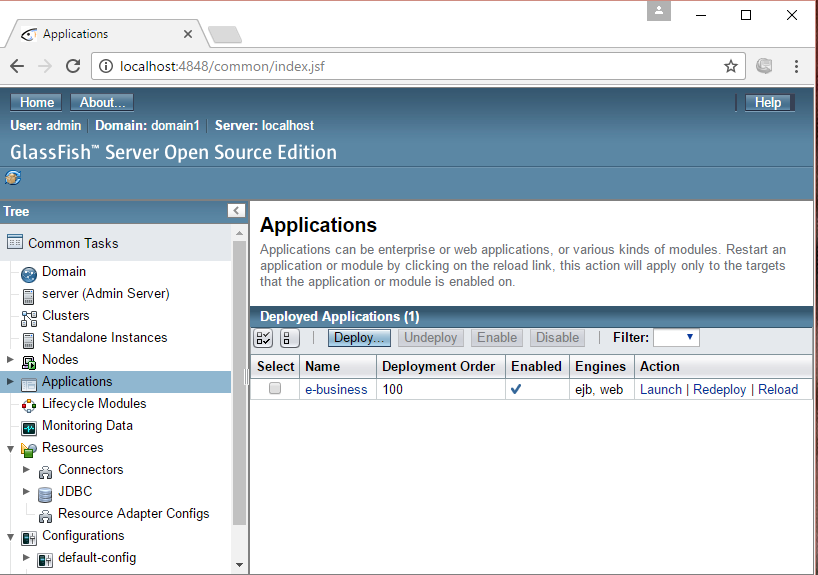
## Step 4

Deploy the war file as shown in following snapshot:



## Step 5

After the deployment completion, you will get option to launch. When you click on launch it will give link that opens this product in browser. Done.



Screenshots:

