



## FIT5042 Design Report (Major Tasks)

### Banking Enterprise Application

Student Name: Duo Pan

Student ID: 27554074

Tutor Name: ABM Russel

Tutorial Time: Mon 18:00-20:00

Lecturer: Chris Messom

**Contents**

**Page**

1. Overview
2. Functional diagram
3. Core program functionality
4. Usability Design Review
5. Checklist of site functionality.
6. User stories
7. Entity Relation Diagram
8. Data dictionary

## 1. Overview

This assignment is to build a web-based Java prototype of a Banking Enterprise Application. If I have time, I will also implement a desktop client application version. Some Java enterprise technologies should be used in the assignment: JSF, RESTful web services, EJB, Persistence API and so on.

The basic function is that users can login in the system by their username and password. System will then redirect the user to the specific page. If the user is a public customer, he/she can view and edit his/her own information, and can view, create, sort, multiple search their transaction records. If it is a bank worker, he/she can view, create, update, delete and search users.

There are two entities in the database: user and transaction. Their relationship is one to many. User id is a foreign key in table transaction, and on delete cascade. All the attributes are taken into consideration whether to set some check or not.

## 2. Functional diagram

There are 5 projects in total:

- Shared project: It contains bank system interfaces, such as add/delete users, two entity class files and a managed bean file;
- Ejb project: The main business logic is implement in this project. It implements the methods in the interface in shared project. And It will be used by the web project and desktop client project. Persistence API and database settings are in this project;
- War project: It uses JSF and a BootsFaces jar to build well appearance web pages.
- Client project: Use java Swing to create application GUI.
- Enterprise App: The entrance for the assignment. It includes ejb, war and client modules.

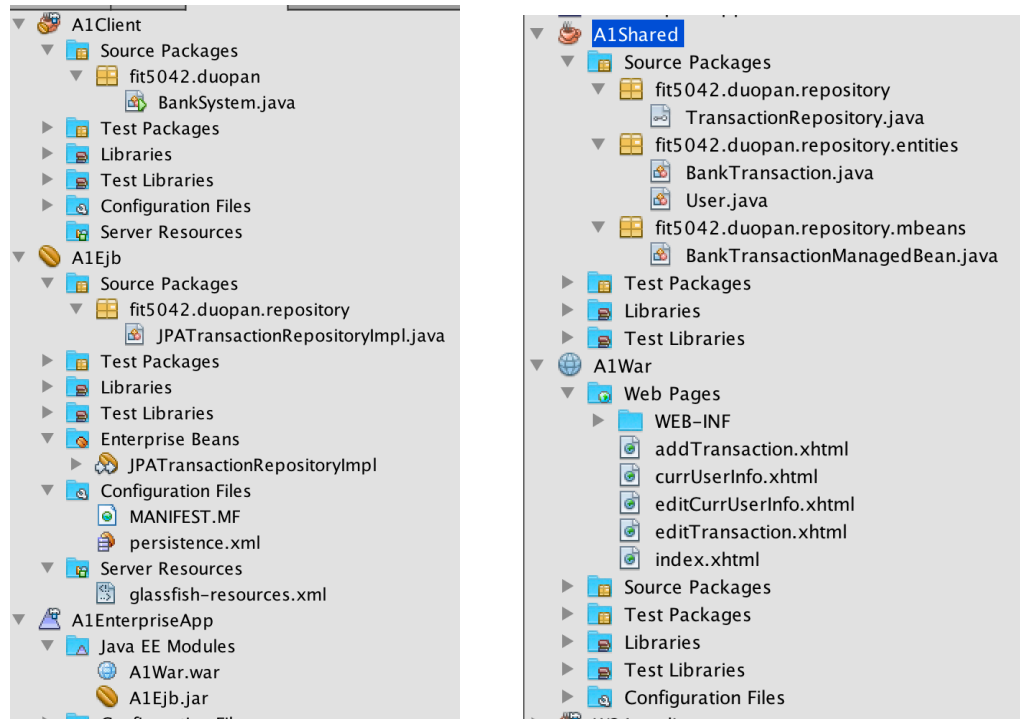


Figure 1. Projects structures

### 3. Core program functionality

- CRUD functions: The key point is pass the value between web pages and database via bean. However, these are basic function and not too hard;
- Validation: Use different annotation and regex patterns to control each input to be correctness.
- Multiple search: Use BootsFaces to realize this function, which is designed for JSF and based on Bootstrap. It has a data table control, and I can manage search function of each column;

### 4. Usability Design Review (identify usability features built into your design)

- Navigation: Each page has good navigation link to the other pages;
- Familiarity: This web site did not adopt any strange layout, so it is familiar with most people;
- Error Prevention: User inputs are check before store into the database;
- Feedback: When mouse moves over table rows, it will highlight.
- Visual Clarity: Data table makes the web site easy to read and use. Red color on delete button also remains user be careful when clicking.

## 5. Checklist of site functionality

<b>1. Credit Functionality</b>	
Search for Transaction by	✓
Transaction Name,	✓
Transaction No	✓
Transaction Type	✓
Results with tabular format with heading.	✓
Option to view the full details	✓
<b>6. Technical Requirements</b>	
<b>Credit</b>	✓
JSF web clients	✓
Persistence API	✓
Application managed entity manager or container managed entity manager.	✓
<b>Audit</b>	✓
No breaking of copyright	✓

**Optional requirements** (for Distinction version of Design report, once above requirements satisfied)

## 6. User stories (that are driving your design decisions)

## 7. Entity relation diagram

## 8. Data dictionary (of your application, including the main data structures and types used in your application.)