# FIT5192 Enterprise and Internet Applications Development
# Semester 2B, 2018 (Suzhou)

## Practical Assignment One Specification

**A Web-based Java prototype of a Collectible Items Exchange**

The assignment is worth 25% of your total unit assessment

- This is an individual assignment. Discussion among students on related tutorial questions is strongly encouraged however working in groups for this assignment is NOT permitted.
- Due date: Tuesday 7 August 2018 @ 23:55.
- Late penalty: 10% of original mark per day, including weekends and public holidays.

## 1. Overview

You have been engaged to design and build a web-based Java prototype for a *Collectible Items Exchange (CIX)*

The main concept is an online system where people can swap or trade collectible items (for example; cars, coins, artworks, antique furniture, fluffy toys; or else units of any type of material or commodity). In order to evaluate the viability of this system and to develop the business model further, it is desirable to first build a mock-up of a live system.

**You are required to produce a prototype for the above system with Java enterprise technologies, such as JSF, RESTful Web Services, EJB, Persistence API with both web and application clients.**

**This document is intended to be open-ended** and specifies the general requirements for the assignment. **If you would like to modify or implement other business rules or features** that are not specified here, you may do so, as long as you **document your assumptions** and ensure that these are reasonable and that the basic technical requirements are still met. Please discuss any proposed changes or additions with your lecturer or tutor.

The following **Sections 1 to 5** describe the requirements of the complete system, however note that you do NOT have to meet all the requirements to pass the assignment. Also note that some of the requirements are left open to interpretation, or represent possible variants, leaving room for your individual creative input. Please refer to **Section 6 Assessment Guide** at the end of the document to find out the different requirements to reach the different assessment levels.

# 2. Business Entities

This section outlines the main business entities that are required by the proposed system. Depending on your design, you might need more business entities to facilitate the operations of your solution. For example, while you might catalogue each type of item (or commodity) within the exchange, you may want to separate the information about particular types of items, including any sub-categories of these within separate, derived entities. Alternatively, you may like to use a separate entity to store individual items of a given type, using an appropriate identification system.

## 2.1. Exchange Users

There are three regular types of CIX User, each with increasing status and privileges: Visitors (the default user type), Traders and Administrators. A fourth *Super-Administrator* type (SA) represents the central store, and there can only be one of these. (The ID number for this user should be set to 0 or 1). Each type of user has the following required information stored within the database:

- User ID number
- Email address
- Password, complying with appropriate validation rules
- Membership level (Visitor, Trader, Administrator, SA)

For Administrators and Traders, the following personal details must also be recorded:

- Last name
- First name
- Address
- Phone number, complying with a specified numbering system
- Available Credits (in any real or imaginary currency)

In addition to the information above, for each user there is a record of any tradeable items (or units of commodities) that they may have in their possession.

Visitors, only identified by their email, can log in to the system where they can search and view items. They cannot buy, sell or exchange these with the store. To be able to perform any of these actions, Visitors must first upgrade to Trader (or possibly Administrator) status.

Since it is hard to prevent multiple logins by the same individuals, it is permissible for Administrators to acquire and exchange collectible items or commodities themselves. (Moreover, Administrators may even be offered an incentive bonus in terms of trading discounts).

Business rules for all system users:

1. Each user is associated with a unique ID number in the system
2. A password, required by all users including visitors, must conform to the following rules:
    a) must be a minimum of 8 and a maximum of 20 characters long,
    b) contain at least one upper case letter,
    c) contain at least one lower case letter,
    d) contain at least one digit (0-9), and must
    e) contain at least one of a specified set of special characters

3. Email addresses, if recorded, must be in a valid email format (e.g. <Name>@<EmailProvider>.com).
4. Last name and first name, if recorded, must not contain any numeric values
5. Phone numbers, if recorded, must conform to a valid phone numbering system. For example, for a business based in Melbourne, Australia, the first digit of the phone number must be a 9 and the number must be 8 digits long. On the other hand, if the number is a mobile number, the first digit of the phone number must be 0 and the number must be 10 digits long.
6. Finally, users must be classified as Visitors, Traders, Administrators or Super-Administrator.

## 2.2. Collectible Items and Commodities

The CIX maintains a catalogue of items or commodities of particular types. These can be searched, viewed, exchanged, bought and sold by users according to their membership status. The actual information stored about each type of item or commodity will vary, however the required attributes in all cases are:

- ID number for each type of item or commodity
- One or more category labels which can be used for searching the catalogue
- A short title
- A thumbnail image
- The number of items or units available within the store
- The total number of items or units in circulation (note that this information is restricted)
- The price per item or unit.

Some suggestions for additional information that might be included, depending on the nature of the items or commodities within each prototype exchange, are listed below:

- A detailed description or review
- Color, size and fabrication material(s)
- Quality ratings or grades
- Author, artist or designer
- Date of manufacture, production or release
- Other relevant data

For the purpose of the assignment, this variable information should be selected to cover a range of data types.

*Suggested* Business rules for collectible items and commodities:

1. Collectible items or commodities must be classified into at least three different types or sub-types (these can be further sub-divided if desired)
2. The total number of collectible items or units of a commodity in circulation within the exchange and in possession of the exchange users, $T_i$ must at all times equal a fixed quantity, $Q_i$, unless this value, $Q_i$, is changed by the Super-Administrator as described in **Section 3.4**. [Note that this will alter the supply-demand dynamics.]
3. The price of each item of a particular type and subtype, or a unit of a given commodity, $P_i$, is based on a calculation of the product of a base price, $B_i$ and the number of items or units, $N_i$, currently available within the central store, such that $P_i = B_i / N_i$.
4. For simplicity, all items and commodities are assumed to be *non-perishable* and hold their value (i.e., $B_i$ is constant unless altered by SA as described in **Section 3.4**)

## 2.3. Exchanges (or orders)

An exchange is treated as a simultaneous purchase and sales transaction, or order. The following information about each exchange is recorded in the database:

- ID number of the exchange
- ID number of the responsible member
- The date and time when the exchange took place
- List of any types of items sold and the number of units of each
- List of any types of items bought and the number of units of each
- The net proceeds of the transaction (which may be a positive or a negative amount)

*Suggested* Business rules for exchanges:

- An exchange must be associated with one and only one member
- A single exchange may consist of multiple items for purchase and sale
- A valid exchange order must consist of at least one item for purchase or for sale
- There must be a sufficient number of items in the store for the exchange to take place
- At no time may the exchange result in total number of items of a given type (or units of a given commodity) $N_i$ exceed the total number in circulation $Q_i$
- If the net value of the exchange is negative (purchases greater than sales) then the user must have sufficient credit available for the transaction to proceed
- If the net value of the exchange is positive (sales greater than purchases) then the store (super-administrator) must have sufficient credit available for the transaction to proceed

# 3. System Functionalities

## 3.1. User (including Visitor)

The system permits any user to:

- Enter or update their name, address and phone number information
- Search for items or commodities in the exchange catalogue by name, price and by any of a defined set of categorical labels
- View the details of a catalogued item selected;
  - Search results are firstly displayed in tabular format with heading. Only title, thumbnail, identifier, units available and price (if applicable) are displayed
  - The system should also offer the user the option to view the full details from the search result.

## 3.2. Trader

In addition to the above, Traders may also

- View their own personal and account details
- View items or commodities in their possession
- Deposit funds to increase their account balance
- Make withdrawals from their current account balance
- Place exchange orders, according to business rules set out in **Section 2.3**

For the purpose of this prototype system, deposits and withdrawals are assumed to have been verified by an external bank.

## 3.3. Administrator

The system permits an administrator to do everything a Trader can do, plus:

- Search for users in the system by the following combination of criteria:
  - ID Number
  - Email
  - Last Name (if known)
  - First Name (if known)
  - Phone number (if known)

- Display search results in tabular format with headings. Only ID number, email, last name, first name and membership status will initially be shown.
- Display the current account details of the user, including the information about any items or commodities in their possession can also be displayed
- Upgrade the status of a user from Visitor to Trader, or from Trader to Administrator.

- Downgrade status of a user from Administrator to Trader.

### 3.4. Super-Administrator

The following special actions are only available to the Super-Administrator (in addition to the functions of regular Administrators):

- Add types of items or commodities to the catalogue
- Increase or decrease the total number of items or units of a commodity in the store, thereby changing the total numbers in circulation $\boldsymbol{Q_i}$ (and the supply-demand dynamics.)
- Adjust the base price of each item or unit, $\boldsymbol{B_i}$
- Set and adjust a percentage difference between sale and purchase price of items or commodities (or spread), ***S1***. Assume that this applies to all items, ***i***.
- Optionally, set and adjust a more favorable percentage spread for Administrators, ***S2***
- Downgrade the status of a user from Trader to Visitor (see **Note** below).
- Remove a user account from the list (see **Note** below).

**Note**. Downgrading to Visitor level or removal of a user account requires one of the following pre-requisites to have been met:

1. Forced acquisition of any items or units possessed by the user and compensation through deduction of an equivalent amount from the SA account
2. Reduction of the number of items or units in circulation, $\boldsymbol{Q_i}$

## 4. Design of the Website

The overall web site should have a consistent look and feel. Within each page of the web site there must be a navigation bar that links to other pages.

## 5. Technology

The prototype described above must be implemented using Java EE technologies (e.g. **EJB**, **JSF**, **RESTful Web Services** and **Persistence API**). A fully functional Java EE system must be developed for the assignment. You must follow the details below to set up the development environment for your solution.

**Development Environment:**

- JDK 1.7/1.8
- Java EE 7
- Netbeans 8.1 or above
- GlassFish 4.1+
- JavaDB that comes with the Java EE Development Kit

# 6. Assessment Guide

Appropriate and sufficient data validations and corresponding error messages are to be implemented. The system should also ensure that the user is not allowed to violate the integrity of the existing data in the system.

For every submission, all source code must conform to the Java coding standard and must be well commented.

When you design your solution, you should plan ahead based on the grade you aim at as the design on more advanced level may have an impact on the overall design of your system. ***You must complete all the features and technical requirements on one level before moving up to the next. No marks will be awarded for the higher-grade level's functionality if you have not completed ALL of the functionalities and technical requirements in the successive lower-grade levels.***

** The design of your implementation must use the Object-Oriented programming paradigm and must be flexible and easy to maintain.

## Pass Level

### Features

- Users can search for items or commodities by title, type and price (login not required).
- Search results are displayed in tabular format with headings. Only title, type, units available and price are shown.

### Technical Requirements

- The system must provide BOTH **web clients** and **application clients** (command-line based or GUI).
- The Web client must be implemented in **JSF**.
- The Application client can be implemented either based on command lines in console window, or using GUI with **Swing** library and it is **preferred** that the GUI be manually created by coding, rather than auto-generated by IDE tools.
- The persistence of the data is managed via **Persistence API**.
- The system must make use of either **application managed entity manager** or **container managed entity manager**.
- Retrieval of data must be done using **JPQL**.

### Additional Pass Requirements

You are required to submit a **Learning Summary Report** for the Unit as part of the Assignment.

Details of the Learning Summary Report are given in the Assignment Section of Moodle.

## Credit Level

### Features

- All features specified in the Pass Level.
- Users are required to login using a username and password to access the system. The password must conform to the validation rules as specified in **Section 2.1**
- After a successful login, users can view the full details of items or commodities in the store catalogue from the search results.
- After a successful login, Administrators can search users by ID, last name, first name, phone number or email.
- After a successful login Administrators can perform any permitted View/Add/Update/Delete (CRUD operation) on Users, according to business rules and limitations as set out in **Section 3.3**
- Traders can enter purchase or sale orders for items or units of a **single** type or commodity using the system.

### Additional Credit Requirements

As a requirement for Credit level and above, a **Design Report** must be submitted. See **Section 7** for details.

### Technical Requirements

- Web client must be implemented in **JSF**.
- ONLY **web client** is required for features required in credit level (**application client** is still required for features required in pass level).
- The persistence of the data is managed via **Persistence API**.
- Make use of either **application managed entity manager** or **container managed entity manager**.
- Use BOTH **Named** and **Dynamic JPQL** for data retrieval.
- The interaction between clients and database must be handled by **EJBs**.

## Distinction Level

### Features

- All features specified in the Credit Level.
- After a successful login, Administrators can search users by a *combination* of ID, last name, first name, phone number or email.
- Administrators can search Exchanges by a *combination* of Exchange ID, User information and Item information.
- After a successful login, the Super-Administrator can perform any required View/Add/Update/Delete (CRUD operation) on Items or Users, according to business rules and limitations set out in **Section 3.4**

- Traders can enter purchase or sale orders for items or units of **multiple** types or commodities using the system.

**Technical Requirements**

- Web client must be implemented in **JSF**.
- ONLY **web client** is required for features required in distinction level (**application client** is still required for features required in pass level).
- The persistence of the data is managed via **Persistence API**.
- Make use of BOTH **application managed entity manager** and **container managed entity manager**.
- Use BOTH **Named** and **Dynamic JPQL** for data retrieval.
- The interaction between clients and database must be handled by **EJBs**.
- Ability of mapping **inheritance** to database must be demonstrated.
- **Bean validations** must be used to validate data.

## High Distinction Level

**Features**

- All requirements of Distinction Level.
- When adding items to the system, thumbnail images and possibly other information such as reviews or other data can be obtained automatically via web services (e.g. Google Custom Search, Pixabay or many other openly available web services) instead of being entered manually.
- Securing the Application and encrypting user password using **JAAS API**
- Traders can enter **simultaneous** purchase AND sale orders for one or more items or units of **multiple** types or commodities using the system.

**Technical Requirements**

- Web client must be implemented in **JSF**.
- ONLY **web client** is required for features required in high distinction level (**application client** is still required for features required in pass level).
- The persistence of the data is managed via **Persistence API**.
- Make use of BOTH **application managed entity manager** and **container managed entity manager**.
- Make sure BOTH **Criteria API** and **JPQL** are used for data retrieval.
- The interaction between clients and database must be handled by **EJBs**.
- Ability of mapping **inheritance** to database must be demonstrated.
- **Bean validations** must be used to validate data.
- Consumption of web services must be conducted in **EJBs**.
- Application should be secured using **JAAS API**.

## 7. Design Report

The Design report should provide a detailed overview of the functionality of your application. This report is an opportunity to demonstrate your individual design decisions and approach which represents a significant part of the assessment.

Your design report must demonstrate originality, good communication skills and present a well thought out application design. It must clearly communicate the abstractions being created, and outline how the functionality is organised.

**Your design report must include the following elements:**

- Overview of your application's goals
- Functional/architecture diagram illustrating how the core aspects of the application fit together.
- Description of core application functionality and how it works.

To facilitate the above requirements, you may wish to include some or all of the following UML (or comparable design related) diagrams:

1. A **Use Case Diagram** with accompanying descriptions of the use cases
2. An **Entity Diagram** with attributes, primary and foreign keys and associations with directionality and cardinality specified
3. A **Class Hierarchy Diagram** detailing any composition or inheritance relationships or interface realizations. Important Java EE container annotations may be included in notes.

## 8. Assignment Deliverables

You must submit a zip file consisting of:

1. A completed individual **Assignment Cover** sheet;
2. **Source codes**;
3. **Script** for generating your database, tables and the testing dataset (for recreation of your database) with the following database naming convention: Database name: idStudentID. For instance, if your student id number is 12345678, then your database should be named **id12345678.** Please set your DB access account and password as both **fit5192.**
4. **Instructions** (the README file) for running the application (including the name, username and password of the database used);
5. Your **Learning Summary Report** in the **PDF** format;
6. The above and a **Design Report** in the **PDF** format if you are aiming for a High Distinction.

All the files above should be uploaded to Moodle as a zip file and use the following naming convention: FIT5192A1_StudentID.zip. For instance, if your student id number is 12345678, then the file you submit should be named **FIT5192A1_12345678.zip**.

You must submit your work by the submission deadline on the due date (a late penalty of 10% per day, including weekends and public holidays, of the possible marks will apply - up to a maximum of 100%). Individual students will be **interviewed** during their normal tutorial classes right after the assignment due date. You can expect to be asked to explain your original design, discuss design decisions and alternatives, and possibly to modify your code as required. A list of interview times will be posted on the FIT5192 Moodle forum and/or circulated in class.

## 9. Plagiarism

Before submitting your assignment, please make sure that you have not breached the University's plagiarism and cheating policy. It is the student's responsibility to familiarize themselves with the contents of these documents.

Plagiarism and cheating are regarded as very serious offences. In cases where cheating has been confirmed, students would be severely penalized, from losing all marks for an assignment, to facing disciplinary action at the Faculty level. While we would wish that all our students adhere to sound ethical conduct and honesty, please ensure that you are well acquainted with the following policy from the Plagiarism Procedures of Monash, available at

http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-plagiarism-collusion-procedures.html