**Final Term Project CIS\_3270**

This project is about simulating an airline reservation system like expedia.com or cheaptickets.com.

**Functionality Requirements:**

1. Upon start of the application, the user should see a splash screen
2. The Main menu gives the option to the user to register or login
3. Login requires username and password
4. Registration requires information such as first name, last name, address, zip, state, username, password, email, SSN and a security question for password recovery.
5. User can login using username and password (Should register first if not a member)
6. You should enable user to retrieve his password knowing the username with the security question.
7. There should be two types of users. Admin and customer. The admin has more privileges regarding this application.
8. After login into application a customer should be able to:

7.1. Search flight database for flights based on different criteria such as from city/ to

city/ date and time of flight etc.

7.2. Be able to book a flight and add that to his account.

7.3. Be able to delete a flight from his account.

7.4. Customer should not be able to book same flight more than once.

7.5. If there is a conflict about date and time of a flight, application should warn the

customer about this and should not let user book the conflicting flight.

7.6. Application should keep track of number of passengers booked at each flight and

should not let a user book the flight if it is full.

1. An admin should be able to do all customer activities in addition to be able to add, update or delete a flight.
2. User should be able to logout and upon login; previously booked reservation should be there in his/her account.
3. You should always provide an option in any case to go back to Main Menu.
4. Overall the application should be functional in most parts considering the requirements mentioned above.

**Coding Requirements:**

1. The codes should be written in Java and at least in four different classes (files).
2. At least 3 concepts of Encapsulation, Inheritance, Polymorphism or Composition should be utilized in the codes.
3. Using Abstract classes and Interfaces in the codes. At least one of each
4. The codes should have proper comments.
5. Application should not crash and exception handling should be implemented in every section of the code.
6. The database programming part should be done using JDBC.
7. Oracle, MySQL, PostgreSQL or SQL server can be used as RDBMS.

**NOTE**: **Choosing any other RDBMS is automatic failure of the project.**

**Documentation Requirements:**

It should be at least 5 pages which explain:

1. Briefly explains requirements and specification of the project
2. Showing the class diagram using UML
3. Showing the Data Model (Tables and all the PKs and FKs)
4. Briefly explains about functionality and flow of the program

**Application GUI:**

GUI is a must for this project and application will not be considered for grading if it does not have a GUI. An elegant GUI is a plus but not a requirement for the project.

**Grading:**

%40 for Functionality Requirements

%40 for Coding Requirements

%20 for Documentation Requirements

An extra of %5 can be considered for an elegant GUI.

**Note**: ***You can assume anything about any other requirement that is not mentioned in this document.***