Software Design Task (JAVA)

April 2021

Task

Design, implement and test a software system that allows travel agencies to maintain their travel packages' itinerary and passengers.

Specification

1. Each travel package has a name, a passenger capacity, an itinerary (list of destinations) and a list of it's passenger.
2. Each destination has a name, and a list of the activities available at that destination.
3. Each activity has a name, a description, a cost and a capacity. Each activity is available at one destination only.
4. Each passenger can sign up for zero or more activity at each destination of the travel package.
5. Once an activity has reached it's capacity no more passengers can sign up for it.
6. Each passenger has a name and a passenger number.

7. A passenger can be a standard, gold or premium passenger.
    a. A standard passenger has a balance. And each time a standard passenger signs up for an activity the cost is deducted from their balance. They cannot sign up for an activity if they do not have sufficient balance.
    b. A gold passenger has a balance. Each time a gold passenger signs up for an activity, a 10% discount is applied on the cost of the activity and the discounted amount is deducted from their balance. They cannot sign up for an activity if they do not have sufficient balance.
    c. A premium passenger can sign up for activities for free.

In addition to the above specifications, the classes should provide the following functions:

1. Print itinerary of the travel package including:
    a. travel package name,
    b. destinations and details of the activities available at each destination, like name, cost, capacity and description.
2. Print the passenger list of the travel package including:
    a. package name,
    b. passenger capacity,
    c. number of passengers currently enrolled and
    d. name and number of each passenger
3. Print the details of an individual passenger including their
    a. name,
    b. passenger number,
    c. balance (if applicable),
    d. list of each activity they have signed up for, including the destination the at which the activity is taking place and the price the passenger paid for the activity.
4. Print the details of all the activities that still have spaces available, including how many spaces are available.

Sub-Tasks

The following are the sub-tasks that needs to be done. The code needs to be written using the Java programming language and needs to adhere to the guidelines detailed out in the next section.

1. Higher Level Diagram : A block diagram indicating interaction between different sub-blocks.
2. Lower Level Diagram : A UML class diagram of all the classes that needs to implemented. The classes must be designed with suitable attributes and methods to represent the requirement mentioned above.
3. Implementation : Java programming language should be used for implementation.
4. For every class, unit test cases should be written. Junit framework can be used for unit testing the classes.

> 💡 ***Note:*** *You may use https://app.diagrams.net/ for designing the Higher Level Diagram(HLD) and Lower Level Diagram(LLD).*

Guidelines

The code should adhere to the following guidelines.

1. Each class should follow the SOLID Design Principles.
2. Clean Code Guidelines should be kept in mind while writing the code (Dependency Injection can be skipped)
3. Unit Test guidelines should be followed.
4. Java code documentation needs to be done properly.

Deliverables

1. High-Level Diagram
2. Low-Level Diagram

3. Source Code

Submission

Upload the code to Github and share the repository link. It must be a mono repo project. Make sure it is a public repository.

Or you can compress the project (.zip) and mail it is an attachment to `shubham.sharma@nymble.in` and cc `aathira.vinod@eatwithnymble.com`