ASSIGNMENT : LAYERED ARCHITECTURE WITH COLLECTIONS

Design a console application for managing Credit Cards. This application should perform CRUD operations. The data for the application is stored in Collections. The following functionalities should be implemented.

1. Adding a new credit card
2. Displaying all the credit cards
3. Searching for a credit card
4. Sorting cards based on criteria
5. Delete card based in credit card number.

DISPLAY APPROPRIATE MENU

This application starts with displaying "Welcome to PowerBank" :

Menu Display

1. Add Card
2. Display Cards
3. Search Card
4. Sort Cards
5. Modify Card
6. Delete Card

FUNCTIONALITY OF THE APPLICATION

1. Add Card
    i. The card details to be accepted
        1. Card Number
        2. Card Holder Name
        3. Validity From
        4. Expires On
        5. Card Type

2. Display Cards
    i. The card details should be displayed as a neat report.

3. Search Card
    i. The card details should be searched based on
        1. Card Number
        2. Card Holder Name
        3. Card Type

4. Expiry Date

4. Sort Cards
    i. The cards should be sorted based on
        1. Card Number
        2. Card Holder Name
        3. Card Type

5. Modify Card
    i. The Card Holder Name can be modified.

6. Delete Card
    i. The cards should be deleted based on
        1. Card Number

7. Confirmation messages
    i. Display confirmation messages
        1. After adding a card and before saving.
        2. Before deleting a card

8. Validations
    i. Perform required validations
        1. Card Number should be 16 digits
        2. Card Number cannot be duplicate
        3. Validate From should be less than Expires On

9. Exception Handling
    i. Create appropriate user defined exceptions

IMPLEMENTATION

The application needs to be implemented with a layered-architecture.

1. The layers that need to implemented are
    a. Presentation Layer
    b. Business Layer
    c. Persistence Layer
    d. Database Layer

2. The applications should be implemented in a loosely-coupled manner
    a. The presentation layer should connect to the service layer through interfaces
    b. The business layer should connect to the persistence layer through interfaces.
    c. The database layer would be implemented using collections, which would be swapped later to a database.

3. The package structure to be implemented is
    a. com.powerbank.ui
    b. com.powerbank.service

c. com.powerbank.dal
d. com.powerbank.model
e. com.powerbank.exception
f. com.powerbank.util

4. Create required classes, interface and exceptions.

RULES AND CONVENTIONS TO BE FOLLOWED WHILE DEVELOPING THE APPLICATION

1. The application architecture should use the appropriate layers.
2. Exceptions to be handled in different layers.
3. The naming conventions and coding standards should be followed.
4. The code should be readable and have proper comments.
5. Code should be written taking reusability into consideration.