M.Sc. DATA SCIENCE

Project Report
on

Virtual Wardrobe App
(Java Application

–

Implementing all the concepts of java)

Submitted by
Angeline A
M.Sc. Data Science B
2348409

APRIL 2024

CHRIST (Deemed to be University)
Yeshwanthpur Campus
Bangalore, Karnataka

TABLE OF CONTENTS

ABSTRACT

- This report provides an in-depth analysis of the design and implementation of a Java-based online shopping application, emphasizing its adherence to object-oriented programming principles. The application offers a user-friendly interface for browsing, selecting, and purchasing a wide range of clothing and footwear items for men, women, and kids. Key functionalities include user authentication, browsing through wardrobe categories, adding items to the shopping cart, finalizing purchases, and optional premium membership upgrades.

- The report focuses on elucidating how various Java concepts such as encapsulation, inheritance, polymorphism, and interface implementation have been employed to architect the application's design and functionality. Additionally, it explores the application's adherence to best practices in software engineering, including modularity, scalability, and code reusability. Insights into the application's architecture, data flow, and user interaction mechanisms are provided to offer a comprehensive understanding of its inner workings.

- Moreover, the report discusses the significance of the Java ecosystem in facilitating the development of robust and scalable applications, leveraging its extensive libraries, frameworks, and tools. Furthermore, it highlights the role of object-oriented design principles in promoting code organization, maintainability, and extensibility, fostering a structured and efficient development process.

- The findings presented in this report serve to underscore the importance of leveraging object-oriented programming paradigms and Java's features to build versatile and resilient software solutions. Through a detailed examination of the online shopping application's design and implementation, valuable insights into effective software development practices and methodologies are elucidated, providing a roadmap for aspiring developers and practitioners in the field.

INTRODUCTION

- In today's digital age, online shopping has become increasingly prevalent, offering consumers convenience, accessibility, and a diverse range of products at their fingertips. With the proliferation of e-commerce platforms, the demand for robust and user-friendly online shopping applications continues to grow. Java, as a versatile and widely-used programming language, provides an excellent foundation for developing such applications, owing to its strong support for object-oriented programming (OOP) principles and rich ecosystem of libraries and frameworks.

- This report delves into the design and implementation of a Java-based online shopping application that embodies the core principles of OOP and leverages Java's features to deliver a seamless and intuitive shopping experience. The application caters to users of all demographics, offering a comprehensive selection of clothing and footwear items for men, women, and kids. Users can browse through various categories, add items to their shopping cart, finalize purchases, and optionally opt for premium membership to unlock additional benefits.

- The development of this online shopping application serves as a testament to the efficacy of Java in building sophisticated and scalable software solutions. By adhering to established software engineering practices and leveraging Java's robust features, the application demonstrates the versatility and adaptability of the language in addressing complex real-world requirements. Throughout this report, we will explore the intricacies of the application's design, delve into the implementation details, and analyze the underlying Java concepts employed to realize its functionality.

- Through a comprehensive examination of the online shopping application, this report aims to shed light on the pivotal role of Java in modern software development and provide valuable insights into effective OOP practices and methodologies. By elucidating the design decisions, architectural considerations, and implementation strategies employed in the development process, this report aims to equip readers with a deeper understanding of Java-based application development and inspire further exploration and innovation in the field.

SAMPLE CODES

1. Generics

```java
public class ShoppingCart implements Iterable<Item> {
    private List<Item> items;
    private boolean cardUsed;
    private String cardHolderName;
    private String cardNumber;
    private String doorNo;
    private String street;
    private String city;
    private String state;
    private String country;
    private int pin;
```

2. Collections

```java
package com.EliteThreads.shoppingapp;
import java.util.Map;
import java.util.HashMap;
```

```java
public class WomenWardrobe {
    private Map<WomenItemType, List<Item>> wardrobe;

    private ShoppingCart cart;

    public WomenWardrobe(ShoppingCart cart) {
        this.cart = cart;
        this.wardrobe = new HashMap<>(); // Initialize the wardrobe map
        initializeWardrobe();

    }
```

3. Interface

```java
package com.EliteThreads.shoppingapp.model;

import java.util.Map;
import java.util.HashMap;

public interface AccountManagement {
    Map<String, String> userCredentials = new HashMap<>();

    void signUp(String username, String gender, int age, String pas

    boolean login(String username, String password);
}
```

4. Inheritance

```java
public class Admin extends User {
    public Admin(String username, String gender, int age) {
        super(username, gender, age);
    }
}
```

5. Polymorphism

```
@Override
    public void browseWardrobe() {
        // Implementation specific to browsing men's wardrobe
```

6. Abstraction

```
    public abstract void browseWardrobe();
```

7. Encapsulation

```java
package com.EliteThreads.shoppingapp.model;

public class Item {
    private String name;
    private double price;

    public Item(String name, double price) {
        this.name = name;
        this.price = price;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public double getPrice() {
        return price;
    }

    public void setPrice(double price) {
        this.price = price;
    }
}
```

8. Multithreading

```java
Thread paymentThread = new Thread(() -> {
    System.out.println("=== Payment processing... ===");
    // Perform payment processing
    // Simulate payment processing time
    try {
        Thread.sleep(3000); // Simulate payment processing time of 3 seconds
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    System.out.println("=== Payment completed successfully ===");

    // Update UI or perform any necessary actions after payment completion
    System.out.println("=== Your order placed successfully ===");
    System.out.println("=== Your order will be delivered on: " + generateDeliveryDate() + " ===\n");

    // Store card details for premium membership
    cart.setCardUsed(true);
    cart.setCardHolderName(cardHolderName);
    cart.setCardNumber(cardNumber);
    updatePremiumMembership();
});

// Start the payment thread
```

OUTPUT

```
D:\JAVA Programs\Elite Thread Online Wardrobe>java Main.java
====================================================
=== Welcome to ELITE THREADS Online Shopping App! ===

  _      __       _
  \ \    / /  _ | |                         
   \ \/\/ / _ \ |/ __ \ \  _ __ ___    _\ \ 
    \ V V / _) | (_) | |   | '   `  _\ _ )
     \_/\_/ \___|\___/|_| |_| |_| |_|\__|

==== Please Sign up if you are new user ====
1. Sign Up
2. Login
3. Exit
Enter your choice: 1
==== Sign Up ====
Enter username: Angeline
Enter gender: Female
Enter age: 21
Enter password: elite
Confirm password: elite
Account successfully created.

==== Login ====
Enter username: Angeline
Enter password: elite
========= ELITE THREADS =========
1. Browse Wardrobe
2. View Cart
3. Finalize Products
4. Update Premium Membership
5. Logout
Enter your choice: 1
==== ELITE WARDROBE =====
```

```
==== PARTNERS FROM LEADING BRANDS ====
==== CLOTHING ====
Zara, H&M, Forever 21, Gap, Nike, Adidas, Levi's, Ralph Lauren, Tommy Hilfiger, Calvin Klein

==== JEWLLERY ====
Tiffany & Co., Pandora, Swarovski, Cartier, David Yurman, Alex and Ani, Kendra Scott, Bulgari, Chopard, Van Cleef & Arpels

==== ACCESSOREIES ====
Coach, Michael Kors, Kate Spade, Louis Vuitton, Gucci, Fossil, Titan, , Ray-Ban, Oakley, Prada, Hermès

==== FOOTWEAR ====
Nike, Adidas, Puma, Reebok, Converse, Vans, Timberland, Dr. Martens, Birkenstock, Skechers
```

```
======= EXCITING COLLECTIONS AHEAD ======

1. MEN
2. WOMEN
3. KIDS
4. Go back to previous options

Enter your choice: 2
==== Elevate your elegance, Embrace your unique allure ====

==== WOMEN ====

1. FOOTWEAR
2. ACCESSORIES
3. ETHNIC
4. JEWELLERY
5. WESTERN
6. FORMALS
7. View Cart
8. Go back to previous options

3
===== Available items =====
1. Traditional Saree - Kanjivaram Silk - Red - $150.0
2. Anarkali Suit - Designer - Blue - $200.0
3. Lehenga - Embroidered - Green - $180.0
4. Salwar Kameez - Cotton - Pink - $160.0
5. Kurti - Floral Print - Yellow - $100.0
6. Churidar - Silk - Orange - $170.0
7. Saree - Georgette - Purple - $120.0
8. Ethnic Gown - Sequin - Silver - $250.0
9. Patiala Suit - Punjabi - Maroon - $190.0
10. Bridal Lehenga - Velvet - Gold - $350.0
11. Go back to Women Wardrobe

4
Salwar Kameez - Cotton - Pink added to cart.
```

```
==== WOMEN ====

1. FOOTWEAR
2. ACCESSORIES
3. ETHNIC
4. JEWELLERY
5. WESTERN
6. FORMALS
7. View Cart
8. Go back to previous options

4
===== Available items =====
1. Ring - Diamond - Solitaire - Gold - $300.0
2. Pendant - Gemstone - Ruby - Silver - $250.0
3. Bracelet - Pearl - Beaded - White - $180.0
4. Earrings - Hoops - Diamond - Rose Gold - $220.0
5. Necklace - Emerald - Choker - Gold - $400.0
6. Anklet - Silver - Traditional - Red - $120.0
7. Bangles - Glass - Bangles - Green - $80.0
8. Brooch - Floral - Brooch - Blue - $150.0
9. Toe Ring - Silver - Toe Ring - Pink - $50.0
10. Hairpin - Rhinestone - Hairpin - Purple - $40.0
11. Go back to Women Wardrobe

2
Pendant - Gemstone - Ruby - Silver added to cart.
```

```
==== WOMEN ====

1. FOOTWEAR
2. ACCESSORIES
3. ETHNIC
4. JEWELLERY
5. WESTERN
6. FORMALS
7. View Cart
8. Go back to previous options


7
==== Items in Cart: ====
Salwar Kameez - Cotton - Pink - $160.0

Pendant - Gemstone - Ruby - Silver - $250.0

==== WOMEN ====

1. FOOTWEAR
2. ACCESSORIES
3. ETHNIC
4. JEWELLERY
5. WESTERN
6. FORMALS
7. View Cart
8. Go back to previous options


8
======== ELITE THREADS ========
1. Browse Wardrobe
2. View Cart
3. Finalize Products
4. Update Premium Membership
5. Logout
Enter your choice: _
```

```
======== ELITE THREADS ========
1. Browse Wardrobe
2. View Cart
3. Finalize Products
4. Update Premium Membership
5. Logout
Enter your choice: 2
==== Items in Cart: ====
Salwar Kameez - Cotton - Pink - $160.0

Pendant - Gemstone - Ruby - Silver - $250.0
```

```
======== ELITE THREADS ========
1. Browse Wardrobe
2. View Cart
3. Finalize Products
4. Update Premium Membership
5. Logout
Enter your choice: 3
==== Items in Cart: ====
==== Items in Cart: ====
Salwar Kameez - Cotton - Pink - $160.0

Pendant - Gemstone - Ruby - Silver - $250.0
```

```
Do you want to continue with the placement of order? (yes/no)
yes
==== Your total bill: ====
Total Amount: $410.0
GST: $73.8
Delivery Charges: $10.0
Final Amount (Including GST and Delivery Charges): $493.8
Do you want to proceed with the payment? (yes/no)
yes
=== Please provide your address details: ===
Door No: 12
Street: 5th Street
City: Bangalore
State: Karnataka
Country: India
PIN: 56
=== Choose payment option: ===
1. Online
2. Cash on Delivery

Enter your choice: 1
=== Enter card details ===
Card Holder Name: Angeline
Card Number (12 digits): 123456789012
CVV: 124
=== Payment completed successfully ===
=== Your order placed successfully ===
=== Your order will be delivered on: 4 MAY (Saturday) ===
```

```
========================================
==== Invoice ====
Products Purchased:
Salwar Kameez - Cotton - Pink - $160.0
Pendant - Gemstone - Ruby - Silver - $250.0
Total Amount: $410.0
GST: $73.8
Delivery Charges: $10.0
Final Amount (Including GST and Delivery Charges): $493.8
Payment Method: Online
Address:
Door No: 12
Street: 5th Street
City: Bangalore
State: Karnataka
Country: India
PIN: 56
Delivery Date: 4 MAY (Saturday)
========================================
```

```
=== Rate us! Your stars guide our way! ===
Rate our app (1-5):
4
Thank you for your rating! We'll continue to enhance our app.
```

```
======== ELITE THREADS ========
1. Browse Wardrobe
2. View Cart
3. Finalize Products
4. Update Premium Membership
5. Logout
Enter your choice: 5
Logging out...
==== Please Sign up if you are new user ====
1. Sign Up
2. Login
3. Exit
Enter your choice: 3
=== THANK YOU, VISIT AGAIN ===
** FOLLOW US ON:**
** INSTAGRAM: @elitethreads
** FACEBOOK: @elitethreadsbrd
** OFFICIAL WEBSITE: **
** @elitethreads.com
********Exiting********
```

INSIGHTS

Encapsulation:

- Encapsulation is a fundamental principle of object-oriented programming that involves bundling data (attributes) and methods (behaviors) that operate on the data into a single unit, known as a class. It hides the internal state of objects from external access and modification, ensuring data integrity and promoting abstraction.
- In the online shopping application:
    - Private access modifiers are used to restrict direct access to the internal state of classes. For example, instance variables like items, doorNo, street, etc., in the ShoppingCart class are declared as private to encapsulate the state of the shopping cart.
    - Accessor methods (getters) and mutator methods (setters) are provided to manipulate the internal state of objects indirectly, following the principle of encapsulation. These methods allow controlled access to the private attributes, enabling validation and ensuring consistent state transitions.

Inheritance:

- Inheritance is a mechanism in Java that allows a class (subclass or derived class) to inherit attributes and methods from another class (superclass or base class). It promotes code reuse, extensibility, and modularity by enabling the creation of specialized classes that inherit common functionalities from a shared ancestor.
- In the online shopping application:
    - Inheritance is utilized to create specialized wardrobes for different user categories, such as MenWardrobe, WomenWardrobe, and KidsWardrobe, which inherit common functionalities from the Wardrobe class.
    - Common attributes and methods related to browsing, adding items, and viewing the cart are defined in the Wardrobe class, promoting code reuse and ensuring consistency across specialized wardrobes.

Polymorphism:

- Polymorphism, a core concept in object-oriented programming, allows objects of different types to be treated as objects of a common superclass, enabling dynamic method dispatch and runtime method binding. It enhances flexibility, extensibility, and modifiability in software systems.
- In the online shopping application:
    - Polymorphism is demonstrated through method overriding, where subclasses provide specific implementations of methods defined in their superclass.
    - For example, the browseWardrobe() method is overridden in subclasses like MenWardrobe, WomenWardrobe, and KidsWardrobe to cater to the unique browsing requirements of each wardrobe type. At runtime, the appropriate method implementation is dynamically selected based on the type of wardrobe being accessed, promoting flexibility and code maintainability.

Interface Implementation:

- Interfaces in Java define contracts that specify a set of methods that implementing classes must override. They promote loose coupling, code consistency, and interoperability by providing a common interface for unrelated classes to adhere to.
- In the online shopping application:
  - The AccountManagement interface defines methods for user authentication, including signUp() and login(). Concrete classes like ShoppingApp implement these methods to provide authentication functionality.By adhering to the AccountManagement interface, classes ensure adherence to a common contract, enabling interchangeable usage and promoting code extensibility.
  - This allows for easy substitution of different implementations of authentication logic without affecting the overall functionality of the application.

Abstraction:

- Abstraction is the process of hiding complex implementation details and exposing only essential features of an object or system. It helps in managing complexity, promoting modularity, and facilitating code comprehension and maintenance.
- In the online shopping application:
  - Abstraction is achieved through the definition of classes, where each class encapsulates specific functionalities related to shopping, user management, and product browsing.
  - The application abstracts away intricate implementation details of shopping cart management, user authentication, and product categorization, providing users with a simplified and intuitive interface for shopping online.

By effectively implementing these Java concepts, the online shopping application demonstrates robustness, maintainability, and extensibility, ensuring a seamless user experience while facilitating code reusability, modularity, and scalability.

CONCLUSION

In conclusion, the development of the Java-based online shopping application exemplifies the effective utilization of object-oriented principles and design patterns to create a robust, scalable, and user-friendly system. The application seamlessly integrates various Java concepts, such as encapsulation, inheritance, polymorphism, interface implementation, and abstraction, to deliver a cohesive and feature-rich shopping experience. The following points summarize the key takeaways and insights gained from this project:

- **Object-Oriented Design Principles:** The application adheres to fundamental object-oriented design principles, including encapsulation, inheritance, and polymorphism, to promote code reusability, maintainability, and extensibility. By encapsulating data and behaviors within classes, leveraging inheritance for code reuse, and embracing polymorphism for dynamic method dispatch, the application achieves a high level of flexibility and modularity.

- **Interface-based Programming:** Interface implementation facilitates loose coupling and flexibility in the system architecture. By defining interfaces for user authentication and contract adherence, the application decouples implementation details from client code, enabling interchangeable usage and promoting code extensibility. This allows for seamless integration of different authentication mechanisms and enhances the overall scalability of the system.

- **Abstraction and Modularity:** The application leverages abstraction to hide complex implementation details and expose essential functionalities to users. By abstracting away intricate internal workings, the application provides users with a simplified and intuitive interface for browsing products, managing their shopping carts, and finalizing purchases. This abstraction enhances code comprehension, maintenance, and scalability
.

- **User Experience and Interaction:** The application prioritizes user experience by offering a user-friendly interface, seamless navigation, and intuitive workflows. Users can effortlessly browse through product categories, add items to their carts, and finalize purchases with minimal friction. Clear prompts, informative messages, and error handling mechanisms enhance user interaction and ensure a positive shopping experience.

- **Scalability and Maintainability:** The modular architecture and adherence to object-oriented principles facilitate scalability and maintainability in the long run. The application's design promotes code reuse, extensibility, and modifiability, allowing for easy integration of new features, enhancements, and optimizations. Additionally, the separation of concerns and abstraction of functionalities simplify maintenance tasks and support future growth and evolution.

- **Continuous Improvement and Feedback:** The development of the online shopping application is an iterative process, with a focus on continuous improvement and feedback incorporation. User feedback, performance metrics, and market trends inform ongoing

enhancements and feature updates, ensuring that the application remains relevant, competitive, and aligned with user expectations.

- **Future Directions and Expansion:** Looking ahead, the online shopping application has significant potential for expansion and diversification. Future directions may include the integration of advanced personalization algorithms, recommendation engines, and social features to enhance user engagement and satisfaction. Additionally, partnerships with third-party vendors, adoption of emerging technologies, and expansion into new markets could further propel the application's growth and success.

In conclusion, the Java-based online shopping application represents a culmination of best practices in software engineering, user experience design, and business innovation. By leveraging object-oriented principles, interface-based programming, and abstraction, the application delivers a seamless and enjoyable shopping experience while laying the foundation for future growth and evolution in the dynamic e-commerce landscape.

REFERENCES

[1] Book- https://github.com/EbookFoundation/free-programming-books/blob/main/books/free-programming-books-langs.md

[2] GitHub link - https://devfreebooks.github.io/java/

[3] Generics - https://www.tutorialspoint.com/java/java_generics.htm

[4] Great Learning - https://www.mygreatlearning.com/blog/oops-concepts-in-java/

[5] Oracle Java Documentation - https://docs.oracle.com/en/java/

[6] Java Tutorials - https://docs.oracle.com/javase/tutorial/