**E-Commerce Shopping Cart System**

***A Report Submitted in***

***Partial Fulfilment for***

***award of Master of Integrated Technology***

***3rd Semester***

**In**

**OOTs Using JAVA (BMICSE0352)**

**COMPUTER SCIENCE & ENGINEERING**

**By**

**Deepak Agrawal (2301330100516)**

**Abhishek Kumar (2301330100503)**

**Gopal Kumar (2301330100521)**

**Under the Supervision of**

**MS. ADITEE MATTOO**

**Assistant Professor**



**Department of Computer Science & Engineering**

**School of Computer Science and Information Technology**

**NOIDA INSTITUTE OF ENGINEERING AND TECHNOLOGY**

**GREATER NOIDA**

**(An Autonomous Institute)**

**Affiliated to**

**DR. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW**

**DECLARATION**

I hereby declare that the work presented in this report was carried out by me. I have not submitted the matter embodied in this report for the award of any degree or diploma of any other University or Institute.

**Team Leader:**

Deepak Agrawal (Roll No.: 2301330100516)

*Deepak Agrawal*

*(Candidate’s Signature)*

**Team Members:**

*1.) Abhishek Kumar (Roll No.: 2301330100503)*

*Abhishek Kumar*

*(Candidate’s Signature)*

*2.) Gopal Kumar (Roll No.: 2301330100521)*

*Gopal Kumar*

*(Candidate’s Signature)*

**CERTIFICATE**

Certified that **Deepak Agarwal (2301330100516), Abhishek Kumar (2301330100503) and Gopal Kumar (2301330100521)** has carried out the project work presented in this Project Report in partial fulfillment of the requirements for the award of **Master of Integrated technology, 3rd Sem** from **Noida Institute of Engineering & Technology** under my supervision.

Signature: *Aditee Mattoo*  Signature:*Kumud Saxena*

Ms. Aditee Mattoo Dr. Kumud Saxena

Assistant Professor HOD

NIET, Gr. Noida Department of CSE

NIET, Gr. Noida

**ACKNOWLEDGEMENT**

I would like to express my heartfelt gratitude to Ms. Aditee Mattoo for her invaluable guidance, unwavering support, and constant supervision throughout the development of this project. Her expertise and thoughtful insights have been pivotal in shaping the project and overcoming challenges at every stage. I am truly inspired by her commitment to excellence and her ability to provide constructive feedback, which has significantly enhanced the quality of this work.

I extend my deepest thanks and appreciation to our esteemed HOD, Dr. Kumud Saxena, and Deputy HOD, Ms. Aditee Mattoo, for their encouragement, leadership, and continuous motivation. Their visionary guidance and belief in the potential of this project have been instrumental in its successful completion. The environment of encouragement and intellectual stimulation they have created has allowed me to grow both academically and personally.

I am also profoundly grateful to my peers for their collaborative spirit, valuable suggestions, and unwavering support during this journey. Their perspectives and willingness to share knowledge have been a source of inspiration and learning.

Lastly, I would like to thank everyone who has directly or indirectly contributed to the successful completion of this project. Whether through their expertise, advice, or assistance, their contributions have enriched this endeavor. To all those who have stood by me, offered their time, or provided resources, I owe a debt of gratitude for their generosity and encouragement.

**ABSTRACT**

**The “E-Commerce Shopping Cart System” is a desktop application designed to replicate the core features of an online shopping platform. Developed using Java Swing for its graphical user interface (GUI), the system offers users an intuitive experience for browsing products, adding them to a shopping cart, and completing the checkout process.**

**Key features include product management, cart management, and order processing. The Product Catalogue displays a list of available items with details such as name, price, and description, allowing users to browse and select products. The Shopping Cart module facilitates adding, removing, and viewing selected items while dynamically calculating the total cost. During checkout, users can finalize their purchase, and the system generates a comprehensive order summary.**

**This project employs a modular design, separating the backend logic from the frontend interface to ensure scalability and maintainability. It leverages object-oriented programming principles like encapsulation and inheritance, enabling efficient data handling and smooth functionality.**

**The system serves as a practical demonstration of Java’s capabilities in creating functional and user-friendly software. Future extensions could include user authentication, payment gateway simulations, and database integration for data persistence, enhancing its real-world applicability.**

**TABLE OF CONTENTS**

**Page No.**

Declaration i

Certificate ii

Acknowledgement iii

Abstract iv

**CHAPTER 1: INTRODUCTION 1**

1.1 Overview 1

1.2 Objective And Scope 1

**CHPATER 2: LITERATURE REVIEW 2**

2.1 Overview of Existing Systems 2

2.2 Limitations of Existing Systems 2

**CHAPTER 3: REQUIREMENTS 3**

3.1 Functional Requirements 3

3.2 Non-Functional Requirements 3

3.3 Software and Hardware Requirements 3

**CHAPTER 4: SYSTEM DESGIN AND IMPLEMENTATION 4**

4.1 System Architecture (Flow diagram) 4

4.2 GUI Design 4

4.3 Database Design 4

4.4 Implementation Details 4

**CHAPTER 5: TESTING 5**

5.1 Testing Methodology 5

5.2 Test Cases and Results 6

**CHAPTER 6: CONCLUSION AND FUTURE WORK 7**

6.1 Conclusion 7

6.2 Future Work 7

**APPENDIX 13**

**INTRODUCTION**

#### **Overview**

The “E-Commerce Shopping Cart System” is a desktop application developed to simulate the primary functionalities of an online shopping platform. It provides users with a graphical interface to browse products, add items to a shopping cart, and complete transactions. The system focuses on delivering a seamless shopping experience with efficient backend processing.

#### **1.2 Objective and Scope**

**Objective**:

To design and implement a user-friendly eCommerce system with essential features such as product management, shopping cart functionality, and order processing.

**Scope**:

The project is limited to a desktop-based environment, simulating an online store's functionality. It excludes advanced integrations like live payment gateways but lays a foundation for extensions such as user authentication and persistent storage.

**LITERATURE REVIEW**

#### **2.1 Overview of Existing Systems**

Existing E-Commerce platforms like Amazon and eBay offer comprehensive shopping features, including product browsing, user accounts, and payment integrations. Smaller systems focus on specific niches or simpler implementations.

#### **2.2 Limitations of Existing Systems**

* High development complexity for beginners.
* Dependency on internet connectivity for most systems.
* Limited standalone applications for local environments.
* Complexity in integrating GUI with backend logic for small-scale applications.

**REQUIREMENTS**

#### **3.1 Functional Requirements**

1. Display a catalogue of products with names, descriptions, and prices.

2. Allow users to add products to a shopping cart.

3. Enable users to view, update, and remove items in the cart.

4. Dynamically calculate the total cost of items in the cart.

5. Generate an order summary upon checkout.

#### **3.2 Non-Functional Requirements**.

1. The GUI must be intuitive and responsive.

2. System performance should allow real-time cart updates.

3. Modular design for easy scalability and maintenance.

4. Compatibility with Java 17 and higher for cross-platform use.

#### **Software and Hardware Requirements**

* **Software Requirements:**
  1. JDK 17 or above.
  2. Eclipse IDE.
  3. Java Swing for GUI development.
* **Hardware Requirements:**
  1. Minimum 2 GB RAM.
  2. 2 GHz processor.
  3. 500 MB storage space for project files.

**SYSTEM DESIGN AND IMPLEMENTATION**

#### **4.1 System Architecture**

The system architecture includes the following steps:

1. **User Interaction Layer:** User interacts with GUI to browse and select products.
2. **Application Logic Layer:** Backend processes user actions like adding products to the cart.
3. **Data Layer:** Handles temporary storage of product and cart data.

#### **4.2 GUI Design**

The GUI includes:

1. Product catalogue with list view.
2. Shopping cart view with add/remove/update functionality.
3. Checkout page summarizing selected items and total cost.

**4.3 Database Design**

The system does not use a database in its current iteration. All data is handled temporarily using in-memory structures like lists and maps.

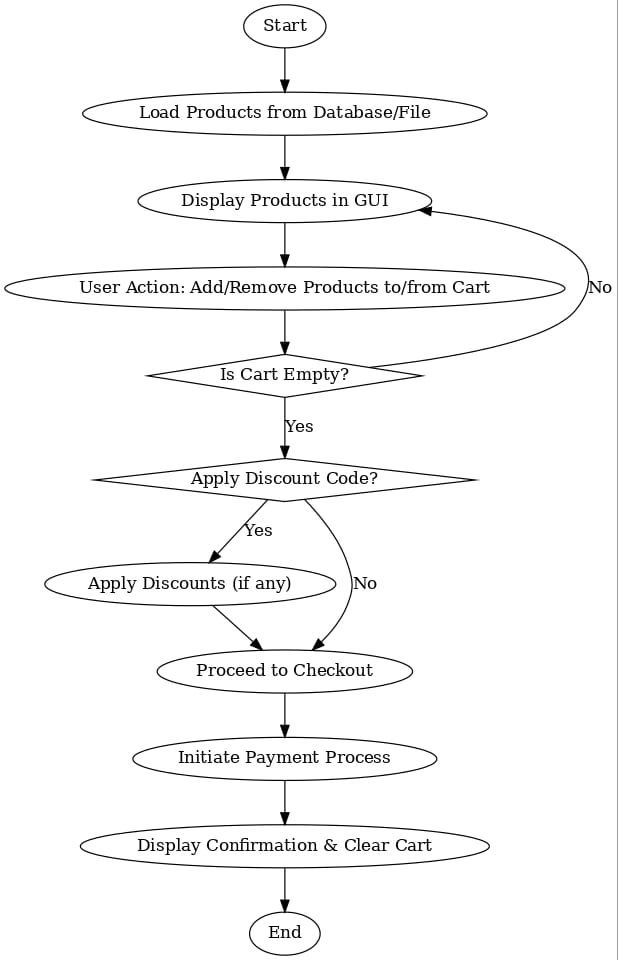
#### **4.4 Implementation Details**

* Developed using Java Swing for GUI.
* Implements object-oriented principles like encapsulation and inheritance.
* Modular code structure with distinct classes for products, cart, and the GUI.

**TESTING**

#### **5.1 Testing Methodology**

The system was tested using black box testing to validate functionality and ensure a smooth user experience. Unit testing was performed on individual components, including the shopping cart and order processing logic.

****

**5.2 Test Cases and Results**

| **Test Case**  **ID** | **Description** | **Input** | **Expected Output** | **Result** |
| --- | --- | --- | --- | --- |
| TC01 | Add product to cart | Product ID: P001 | Product added  to cart | Pass |
| TC02 | Remove product  from cart | Product ID: P002 | Product removed  from cart | Pass |
| TC03 | View total cost | Cart items: 2 | Correct total  displayed | Pass |

**CONCLUSION AND FUTURE WORK**

#### **6.1 Conclusion**

* The “eCommerce Shopping Cart System” successfully replicates core functionalities of online shopping platforms in a desktop environment. It serves as a learning tool for understanding the integration of Java Swing with backend logic, offering scalability and usability.

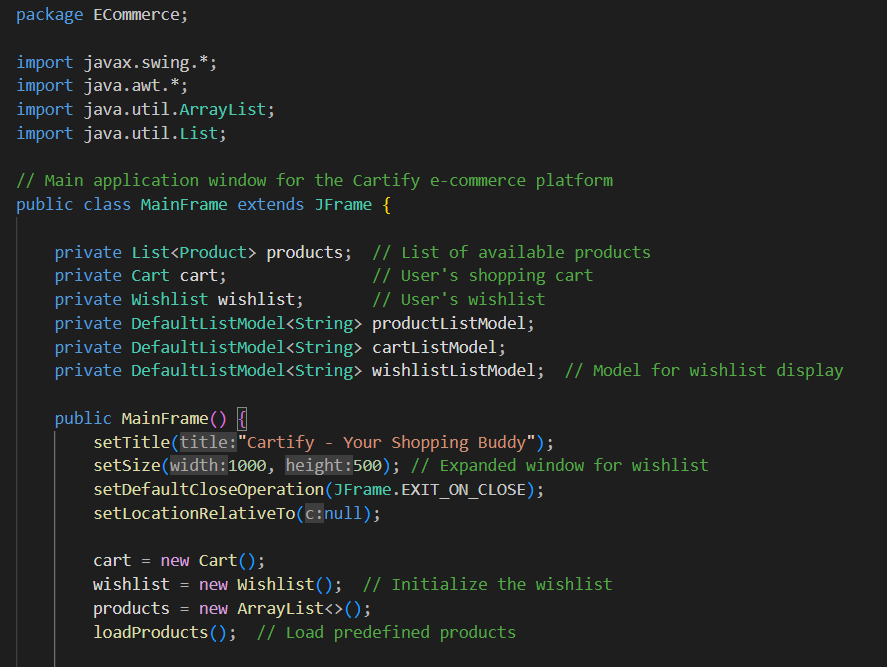
#### **6.2 Future Work**

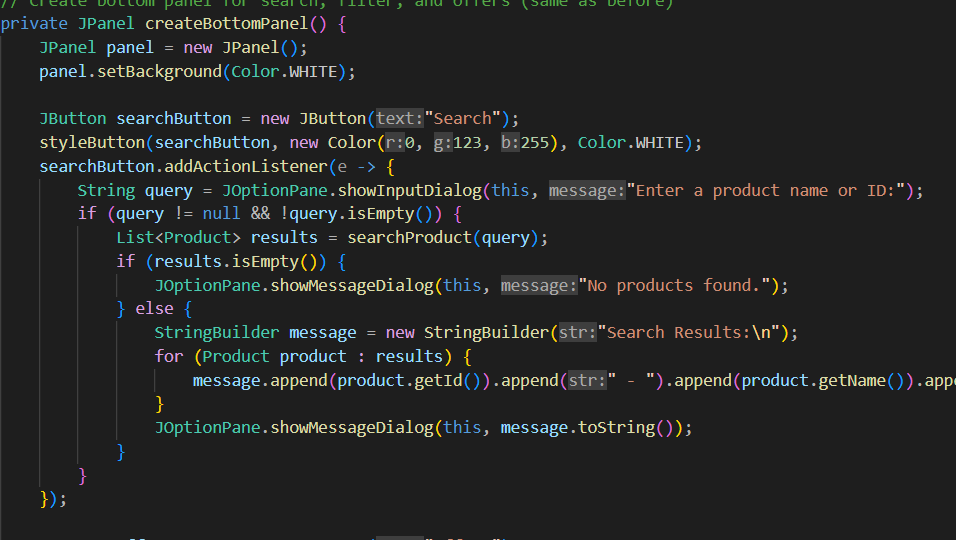
Future improvements can include:

1. Integration of user authentication.
2. Persistent storage using databases.
3. Implementation of payment gateway simulations.
4. Deployment as a web-based application for broader accessibility.

**APPENDIX**

**Source Code**

****

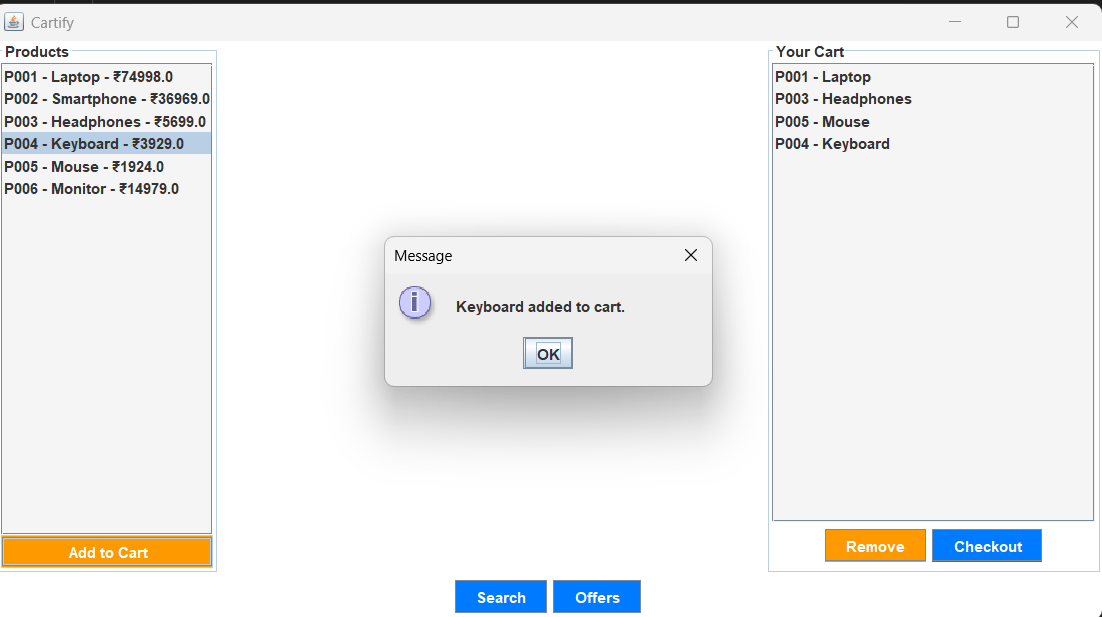
****

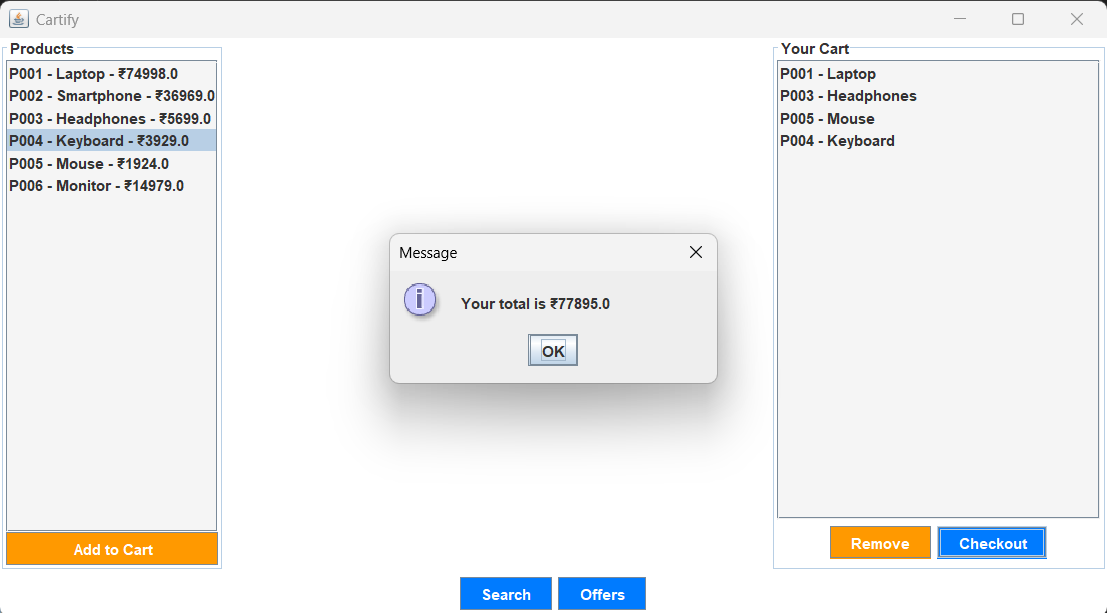
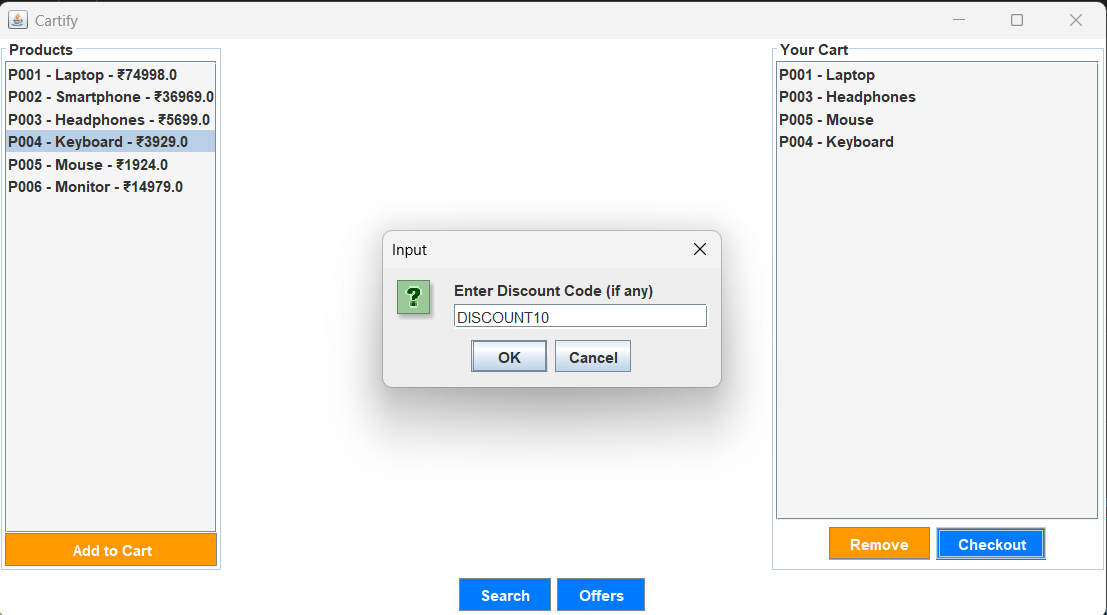
**A screen shot of a computer program

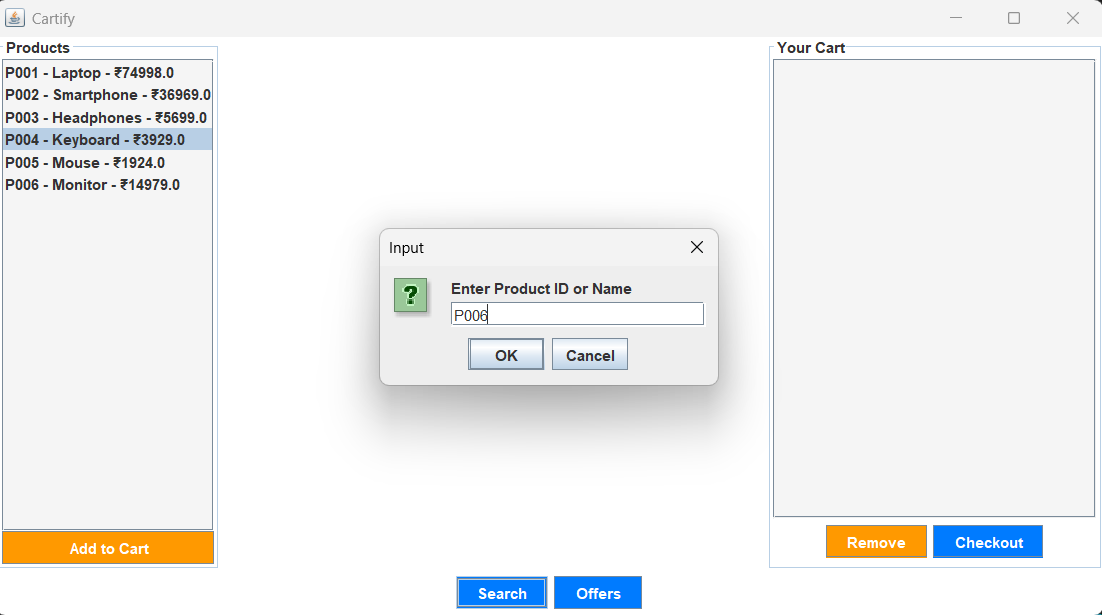
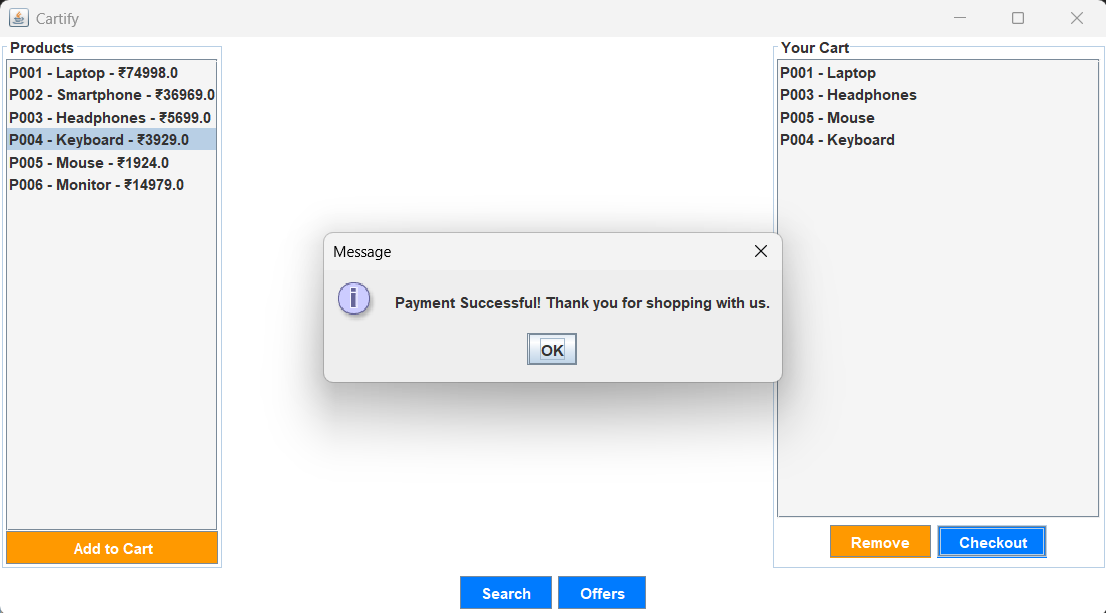
Description automatically generated**

**Final Output**

**A screenshot of a computer

Description automatically generated**

****

****

#### **References**

1. Java documentation for Swing components: <https://docs.oracle.com/javase/tutorial/uiswing/>
2. Tutorials on event-driven programming in Java: <https://www.tutorialspoint.com/java/java_event_handling.htm>