**PROJECT REPORT**

PROBLEM DESCRIPTION:

The goal of this project was to develop an online shopping system that allows customers to view products, manage their shopping carts, and complete purchases. The system also supports administrative functions, including product management and user account management.

DISTINGUISHING FEATURES OF PROJECT:

* **User Types**: The system supports two types of users: Customers and Admins. Each type of user has distinct capabilities and permissions.
* **Shopping Cart Management**: Customers can add, remove, and view products in their shopping cart, and proceed to checkout.
* **Purchase History**: Customers can view their past purchases and details of each transaction.
* **Admin Controls**: Admins can view all products, add new products, and remove existing products from the catalog.
* **Persistent User Data**: User accounts, including shopping carts and purchase histories, are stored and loaded from a file, ensuring data persistence across sessions.

FLOW OF PROJECT:

**User Authentication**:

* Users can either create an account or log in.
* Admins can log in using predefined credentials.

**Product Viewing**:

* Both customers and admins can view the list of available products.

**Customer Activities**:

* **Add to Cart**: Customers can add products to their shopping cart.
* **Remove from Cart**: Customers can remove products from their shopping cart.
* **View Cart**: Customers can view the contents of their shopping cart.
* **Checkout**: Customers can complete their purchase, which records the transaction in their purchase history.
* **View History**: Customers can view their past purchases.

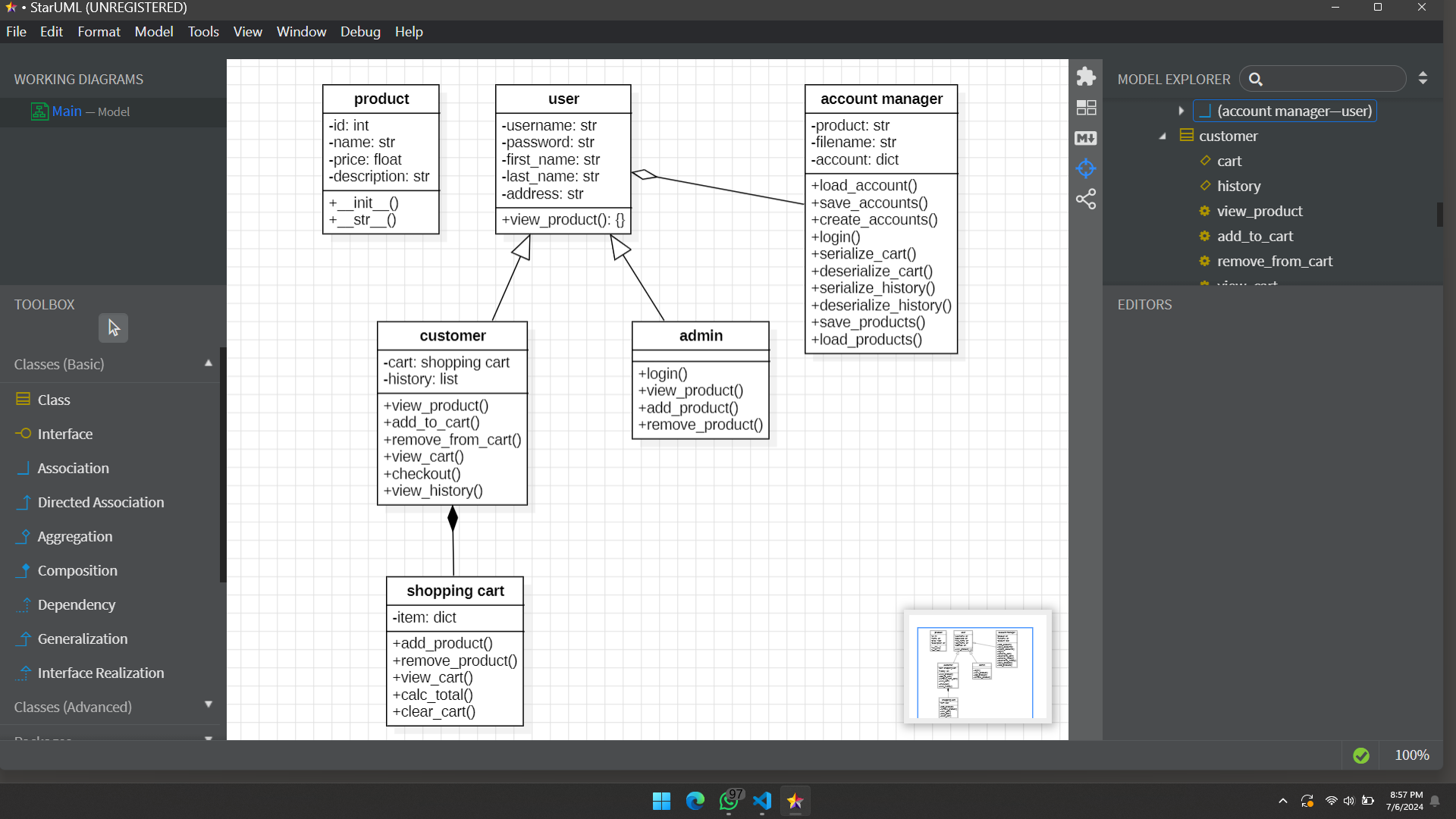
**Admin Activities**:

* **Add Product**: Admins can add new products to the catalog.
* **Remove Product**: Admins can remove products from the catalog.

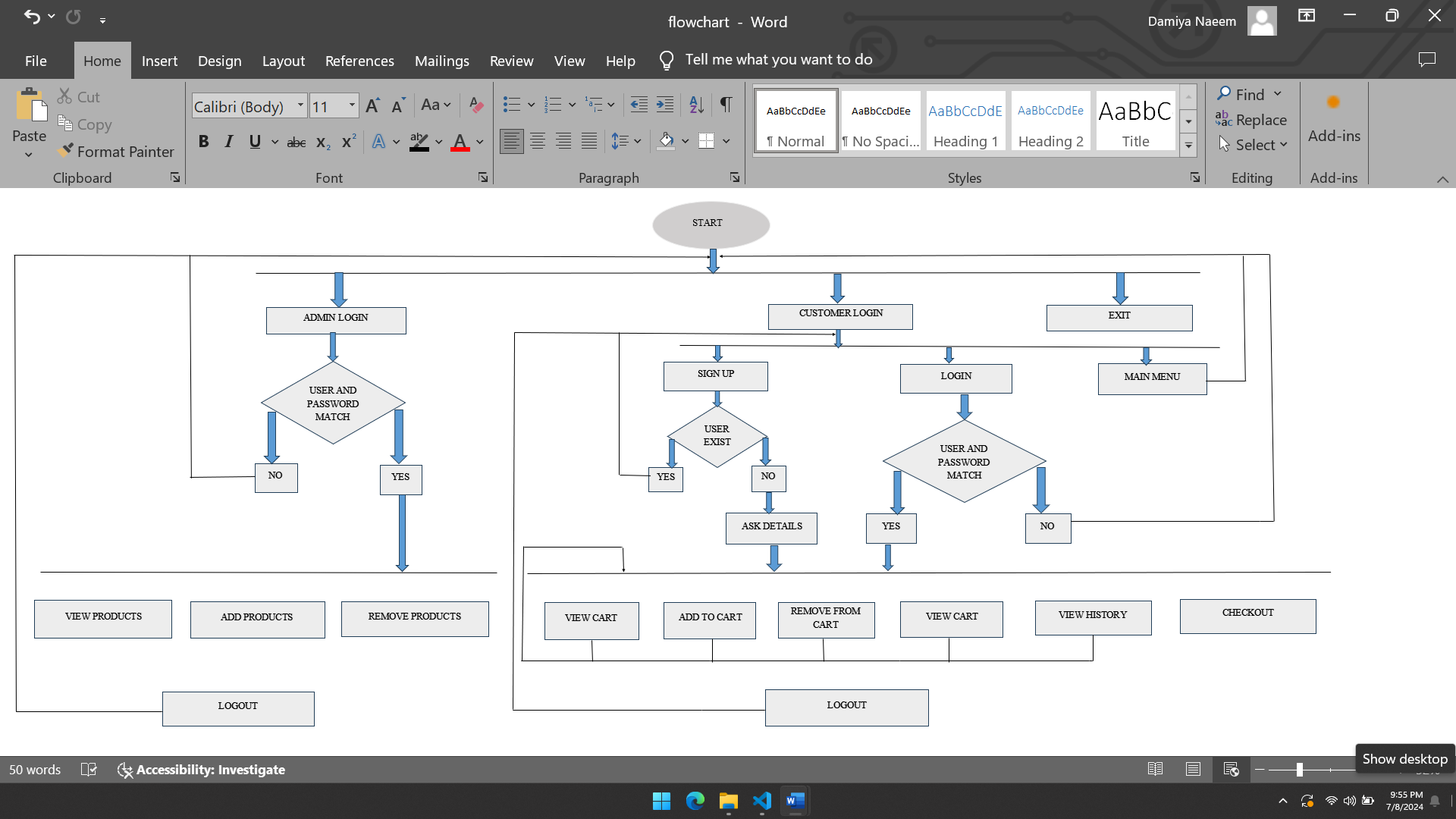
**Account Management**:

* User data, including account details, shopping cart contents, and purchase histories, are saved to a file and loaded upon starting the system.

CLASS DIAGRAM:



FLOW CHART:



MOST CHALLENGING PART WHILE WORKING ON THE PROJECT:

The most challenging part was implementing the data persistence feature. Ensuring that user data (shopping carts and purchase histories) is accurately saved to and loaded from a file requires careful handling of serialization and deserialization of objects.

NEW THING LEARNT IN PYTHON WHILE WORKING ON THE PROJECT:

One of the new things learned was handling file operations to save and load complex data structures in Python. Additionally, using abstract base classes to define common interfaces for different types of users was a valuable lesson in object-oriented programming.

INDIVIDUAL CONTRIBUTION OF EACH MEMBER IN THE PROJECT:

**Member A**: (WALEED AHMED KHAN)

* Worked on the **Customer** and **Admin** classes.
* Implemented user-specific functionalities such as viewing products, managing the shopping cart, and admin controls.
* Developed the **Account Manager** class.
* Implemented account creation, login functionalities, and data persistence features.

**Member B**: (DAMIYA NAEEM)

* Implemented the core functionality for adding and removing products from the shopping cart.
* Created the report and designed the class diagram and flow chart of the project.

FUTURE EXPANSION:

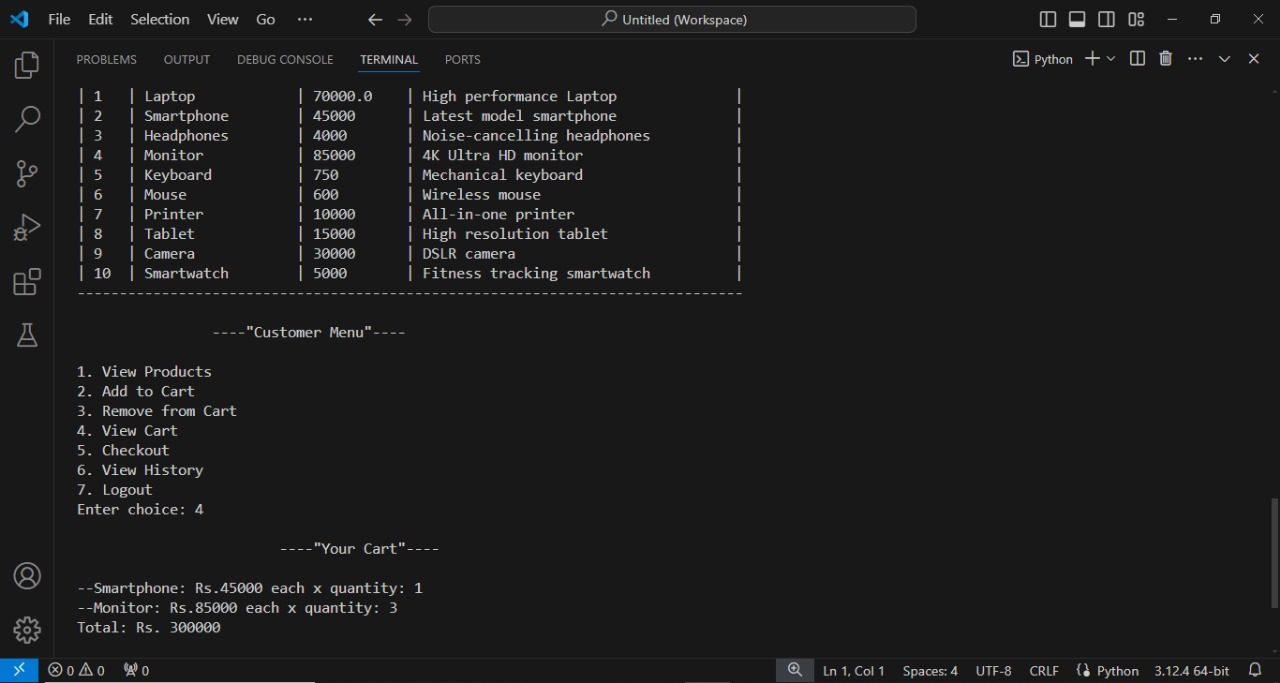
**User Interface**: Develop a graphical user interface (GUI) for better user experience.

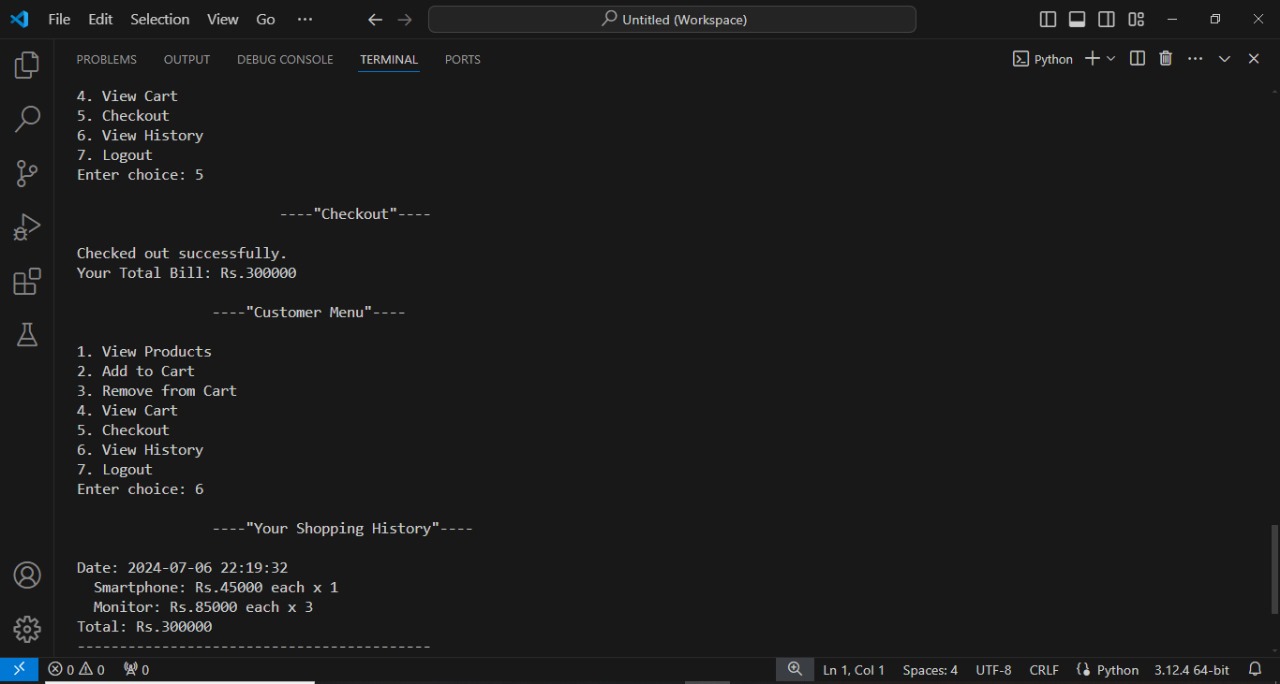
**Payment Integration**: Integrate payment gateways to handle real transactions.

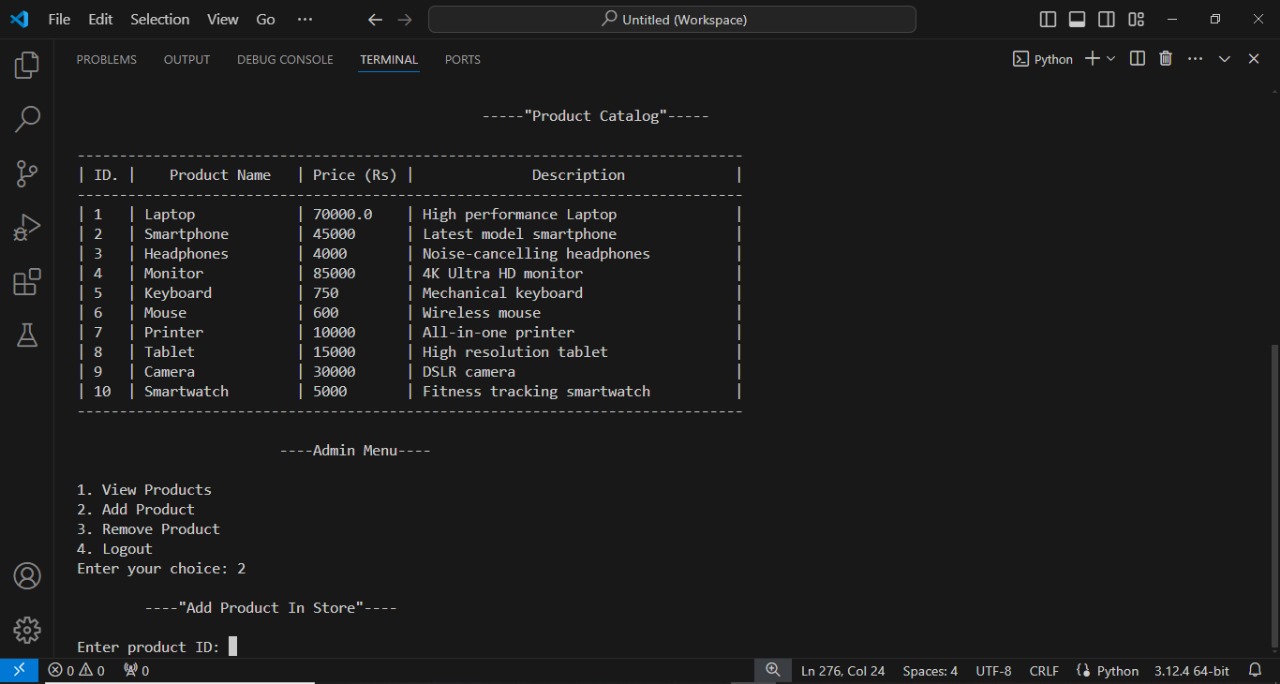
**Product Categories**: Add support for product categories and search functionality.

**User Reviews and Ratings**: Implement a system for customers to leave reviews and rate products.

TEST CASE RUNS:







ORGANIZING THE APPLICATION INTO INTERLINKED USER DEFINED CLASSES:

* **Product Class**: Represents a product in the system.
* **User Class**: An abstract base class for both Admin and Customer.
* **Admin Class**: Inherits from User and manages the product inventory.
* **Customer Class**: Inherits from User, manages a shopping cart, and maintains a purchase

history

* **Shopping Cart Class**: Manages products in a customer's cart.
* **Account Manager Class**: Manages user accounts, including loading and saving customerdata

OBJECT ORIENTED FEATURES:

* **Inheritance**: Customer and Admin classes inherit from the User class.
* **Association**: Customer class contains an instance of the Shopping Cart class.
* **Method Overriding**: view\_ products method is overridden in both Admin and Customer classes.
* **Abstract Classes**: User is an abstract base class with the abstract method view\_ products.
* **Exception Handling**: The Shopping Cart Exception class (though not used here, could be integrated for managing cart-related exceptions).