**Gerardo Andres Almaguer Ramos**

# Business Case: Coffee Shop Order System

## Overview

The Coffee Shop Order System aims to provide a flexible and dynamic way for customers to customize their coffee orders. Utilizing the Decorator Pattern, the system allows customers to select various base coffee types and add multiple optional ingredients or add-ons, each of which can modify the final price and description of the coffee.

## Key Components

1. **Coffee Base Classes**: These represent the different types of coffee available in the shop, such as Espresso, Latte, Cappuccino, etc. Each type has a base cost and a description.
2. **Decorator Classes**: These allow customers to add various add-ons to their coffee, such as milk, sugar, flavored syrups, and whipped cream. Each decorator adds its own cost to the base coffee and modifies the description to reflect the chosen add-ons.
3. **Cost Calculation**: The system calculates the total cost based on the selected coffee type and any add-ons. Each time an add-on is added, it contributes to the overall cost, and the description is updated accordingly.

## Business Objectives

- **Customization**: Provide customers the ability to create their own unique coffee blends by choosing from a variety of add-ons.
- **Dynamic Pricing**: Automatically calculate the total price based on the selected coffee and add-ons, ensuring transparency for customers.
- **User-Friendly Experience**: Simplify the ordering process by allowing customers to visually see their choices and understand how each option affects the total price.

## Functional Requirements

1. **Coffee Types**: The system should support various types of coffee:
    - Espresso
    - Americano
    - Latte
    - Cappuccino
    - Mocha
2. **Add-Ons**: The following add-ons should be available for selection:
    - Milk
    - Sugar
    - Vanilla Shot

    - Whipped Cream
3. **Ordering Process**:
    - Customers select a base coffee type.
    - Customers can add one or more add-ons to their coffee.
    - The system calculates the total cost dynamically as add-ons are selected.
    - The complete order summary, including the description and total cost, should be displayed to the customer.
4. **Unit Testing**: Implement unit tests to ensure that the cost calculations and descriptions are accurate, aiming for a minimum of 85% code coverage.

**Technical Implementation**

- **Decorator Pattern**: The core design pattern used in this application is the Decorator Pattern, which allows for flexible extension of coffee objects by adding additional responsibilities (add-ons) without modifying the existing code structure.
- **Object-Oriented Programming**: The application is built using principles of object-oriented programming, making it easy to maintain and extend as new coffee types or add-ons are added in the future.

**Example Use Case**

1. A customer walks into the coffee shop and decides to order a **Latte**.
2. They choose to add **Milk** and **Double Sugar** to their Latte.
3. The system calculates the total cost:
    - Base cost for Latte: $3.00
    - Cost for Milk: $0.50
    - Cost for two servings of Sugar: $0.20 + $0.20
    - **Total Cost**: $3.90
4. The order summary displayed to the customer would read:
    - **Order**: "Latte, Milk, Sugar, Sugar"
    - **Total**: $3.90

**Benefits**

- **Increased Customer Satisfaction**: Customers can create their preferred drinks, enhancing their overall experience.
- **Flexibility in Pricing**: The system can adapt easily to changes in pricing for coffee types or add-ons, allowing for responsive business strategies.
- **Scalability**: New coffee types or add-ons can be added to the system with minimal effort, ensuring the business can grow its offerings.

## Conclusion

The Coffee Shop Order System provides a practical application of the Decorator Pattern, facilitating a customizable coffee ordering experience that benefits both customers and the business. By using this design pattern, the coffee shop can efficiently manage a variety of orders while maintaining clear and accurate pricing structures.