

## ЛАБОРАТОРНА РОБОТА № 3-4

### Структурне моделювання. Класи

#### ВЕБ-ПЛАТФОРМА ДЛЯ ПРОДАЖУ ЗАПЧАСТИН ДЛЯ АВТО

##### 1. Визначення основних класів і їх відповіальностей

User (логічний базовий клас) - базова сутність для всіх користувачів системи.

Відповіальність: аутентифікація, зберігання профілю, визначення ролі користувача (Customer, Admin).

Атрибут	Тип	Опис
user_id	int	Унікальний ідентифікатор
username	string	Логін користувача
passwordHash	string	Хешований пароль
role	string	Роль (customer, admin)

Методи	Повертає	Опис
RegisterUser(login, password, role)	void	Реєстрація користувача
Login()	void	Вхід у систему
Logout()	void	Вихід у систему

Customer (клієнт наслідує User) - користувач, який здійснює покупки автозапчастин.

Відповіальність: перегляд каталогу товарів, оформлення замовлень, перегляд історії покупок.

Змн.	Арк.	№ докум.	Підпис	Дата	ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4		
Розроб.	Поліщук І. А.						
Перевір.	Сугоняк І. І.						
Керівник							
Н. контр.							
Зав. каф.	Вакалюк Т. А.				Літ.		
					1		
					8		
Звіт з лабораторної роботи					ФІКТ Гр. ІПЗ-22-1		

Атрибут	Тип	Опис
customer_id	int	Унікальний ідентифікатор
firstName	string	Ім'я
lastName	string	Прізвище
email	string	Електронна пошта
phone	string	Номер телефону
user_id	int	PK від User

Методи	Повертає	Опис
Register(user_id)	void	Створення профілю клієнта
EditProfile()	void	Редагування профілю
ViewOrderHistory()	List<Order>	Перегляд історії замовлень
CreateOrder(product_id, quantity)	void	Створення замовлення
CancelOrder(order_id)	void	Скасування замовлення

Admin (адміністратор наслідує User) - користувач із розширеними правами доступу.

Відповідальність: керування товарами, замовленнями, користувачами та звітами.

Атрибут	Тип	Опис
admin_id	int	Унікальний ідентифікатор
email	string	Пошта
user_id	int	PK від User

Методи	Повертає	Опис
Register(user_id)	void	Створення профілю адміністратора
ManageUsers()	void	Керування користувачами
ManageProducts()	void	Додавання, редагування, видалення товарів
ManageOrders()	void	Оновлення статусів замовлень

Змн.	Арк.	Поліщук І. А.			ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4	Арк.
		Сугоняк І. І.				2

GenerateReports()	void	Формування звітів
-------------------	------	-------------------

Product (товар) - автозапчастина, доступна для продажу.

Відповіальність: зберігання інформації про товар.

Атрибут	Тип	Опис
product_id	int	Унікальний ідентифікатор
name	string	Назва
category	string	Категорія
manufacturer	string	Виробник
price	decimal	Ціна
stock	int	Кількість на складі
description	string	Опис

Методи	Повертає	Опис
AddProduct()	void	Додати товар
EditProduct()	void	Редагувати товар
DeleteProduct(product_id)	void	Видалити товар

Order (замовлення) - оформлене клієнтом замовлення.

Відповіальність: облік замовлення, його статус та сума.

Атрибут	Тип	Опис
order_id	int	ID замовлення
customer_id	int	FK до Customer
orderDate	DateTime	Дата створення
status	string	Статус (new, paid, shipped, cancelled)
totalAmount	decimal	Загальна сума

Методи	Повертає	Опис
CreateOrder()	void	Створити замовлення
UpdateStatus()	void	Оновити статус
Cancel()	void	Скасувати замовлення

Змн.	Арк.	Поліщук І. А.			ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4	Арк.
		Сугоняк І. І.				3

OrderItem (позиція замовлення) - окремий товар у замовленні.

Відповіальність: зв'язок між товаром і замовленням.

Атрибут	Тип	Опис
orderItem_id	int	ID позиції
order_id	int	FK до Order
product_id	int	FK до Product
quantity	int	Кількість
price	decimal	Ціна за одиницю

## 2. Зв'язки між класами

### Наслідування (Generalization)

- User <|-- Customer
- User <|-- Admin

### Асоціації (Association):

- Customer "1" -- "0..\*" Order : places
- Order "1" -- "1..\*" OrderItem : contains
- OrderItem "1" -- "1" Product : refersTo
- Admin "1" -- "0..\*" Product : manages
- Admin "1" -- "0..\*" Order : manages

### Агрегація (Aggregation)

Customer o-- Order

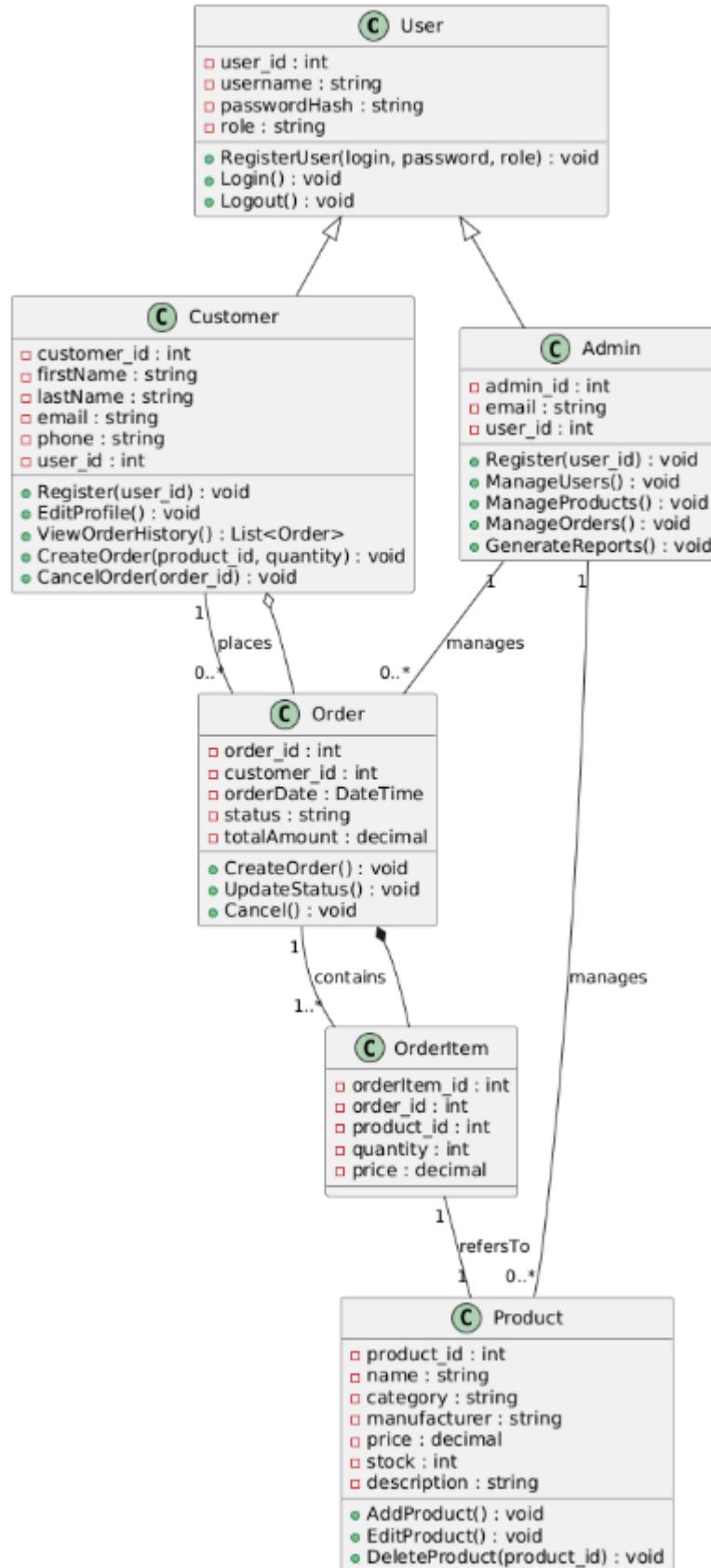
(Замовлення можуть зберігатися навіть після видалення профілю клієнта)

### Композиція (Composition)

Order \*-- OrderItem

(Видалення замовлення → видалення всіх його позицій)

		Поліщук І. А.			ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4	Арк.
		Сугоняк І. І.				
Змн.	Арк.	№ докум.	Підпис	Дата		4



		Поліщук І. А.		
		Сугоняк І. І.		
Змн.	Арк.	№ докум.	Підпис	Дата

## **Діаграма класів системи із використанням патернів GOF**

### **Factory**

Діаграма показує, як фабрики створюють об'єкти без використання умовних операторів:

- UserFactory створює різні типи користувачів (Customer, Admin).
- ProductFactory створює різні категорії товарів (Engine, Body і т.д.).

Це дозволяє централізовано керувати створенням об'єктів і легко додавати нові типи користувачів або товарів.

### **Strategy**

Діаграма демонструє гнучку зміну поведінки замовлення:

- Order делегує зміну статусу (New, Paid, Shipped) відповідній стратегії (OrderStatusStrategy).
- Система оплати (PaymentStrategy) може використовувати різні методи (CreditCardPayment, PayPalPayment).

Це дозволяє змінювати логіку без зміни базових класів.

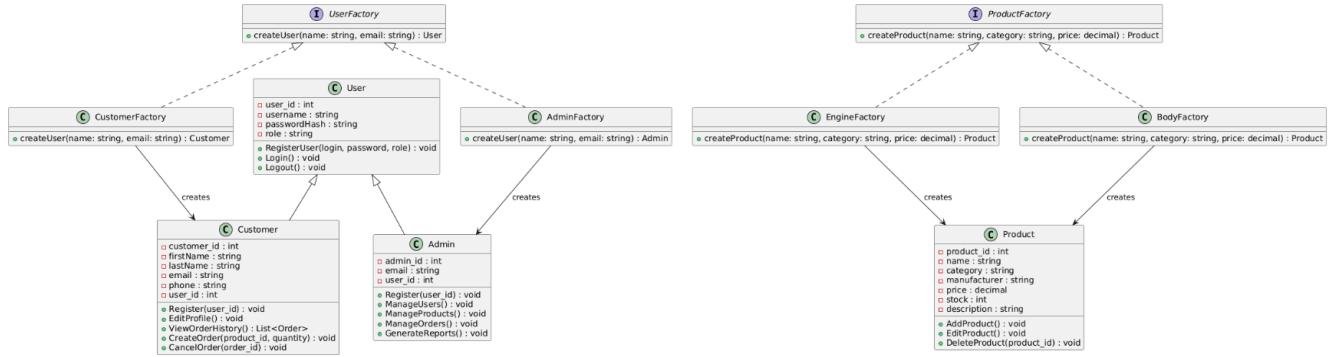
### **Observer**

Діаграма показує механізм сповіщень клієнтів про зміни замовлення:

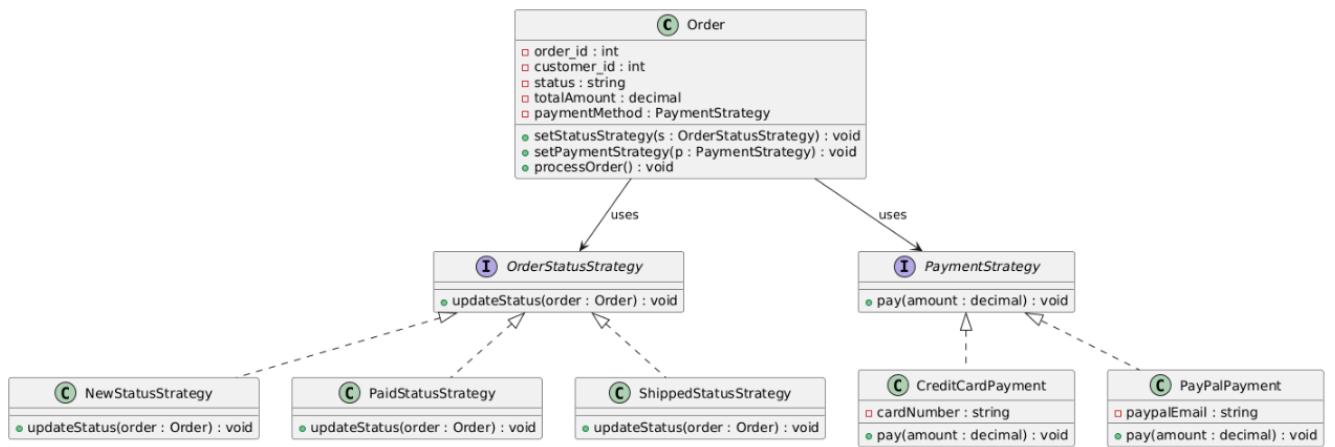
- OrderSubject зберігає список підписаних клієнтів.
- При зміні статусу викликається notifyObservers(), і всі підписані клієнти (CustomerObserver) отримують повідомлення.

Це забезпечує автоматичне інформування користувачів без прямої залежності між замовленням і клієнтом.

		Поліщук І. А.			ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4	Арк.
		Сугоняк І. І.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

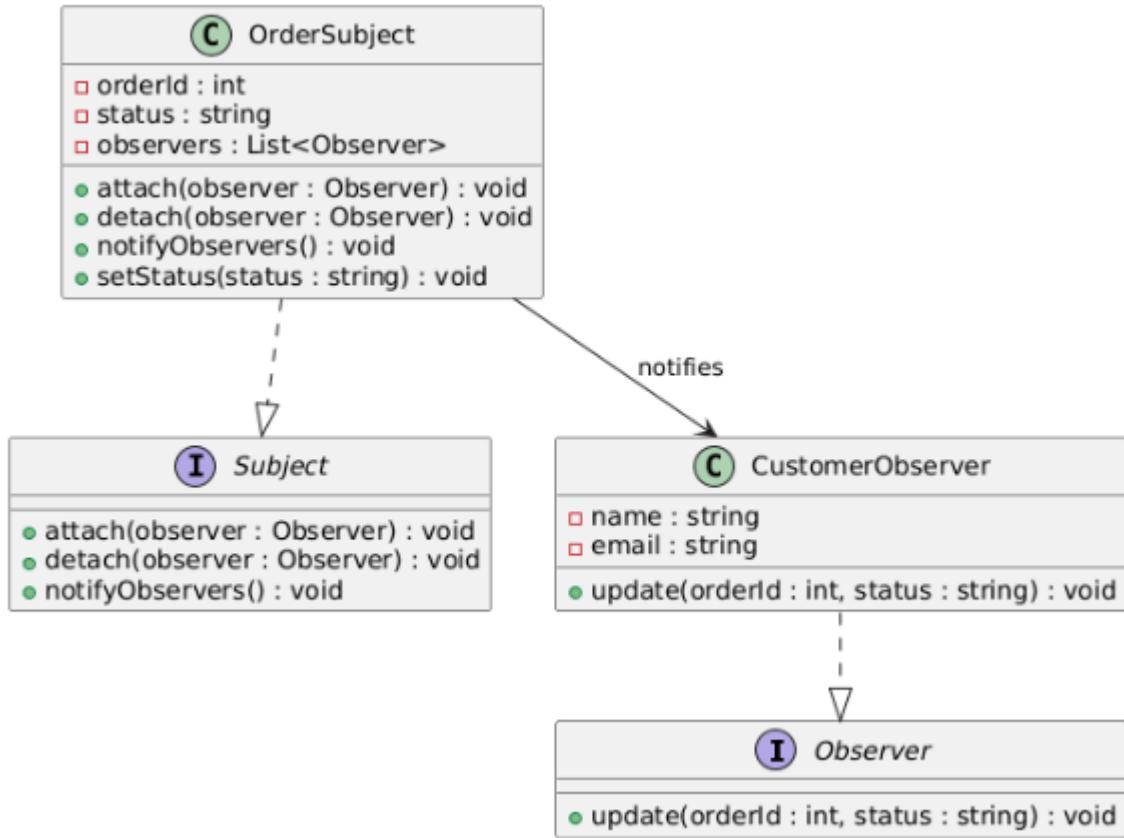


Патерн Factory



Патерн Strategy

		Поліщук І. А.						
		Сугоняк І. І.						
Змн.	Арк.	№ докум.	Підпис	Дата		ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4	Арк.	7



## Патерн Observer

		<i>Поліщук І. А.</i>				<i>Арк.</i>
		<i>Сугоняк І. І.</i>			<i>ДУ «Житомирська політехніка».25.121.24.000 – ПрЗ-4</i>	
Змн.	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		8