

# Архітектурний опис системи бронювання авіаквитків

## Мета проєкту:

- Система призначена для автоматизації процесу бронювання авіаквитків.
- Вона забезпечує повний цикл роботи: від реєстрації користувача та створення бронювання до оплати та видачі квитка.
- Адміністратор має можливість керувати рейсами та оновлювати розклад.

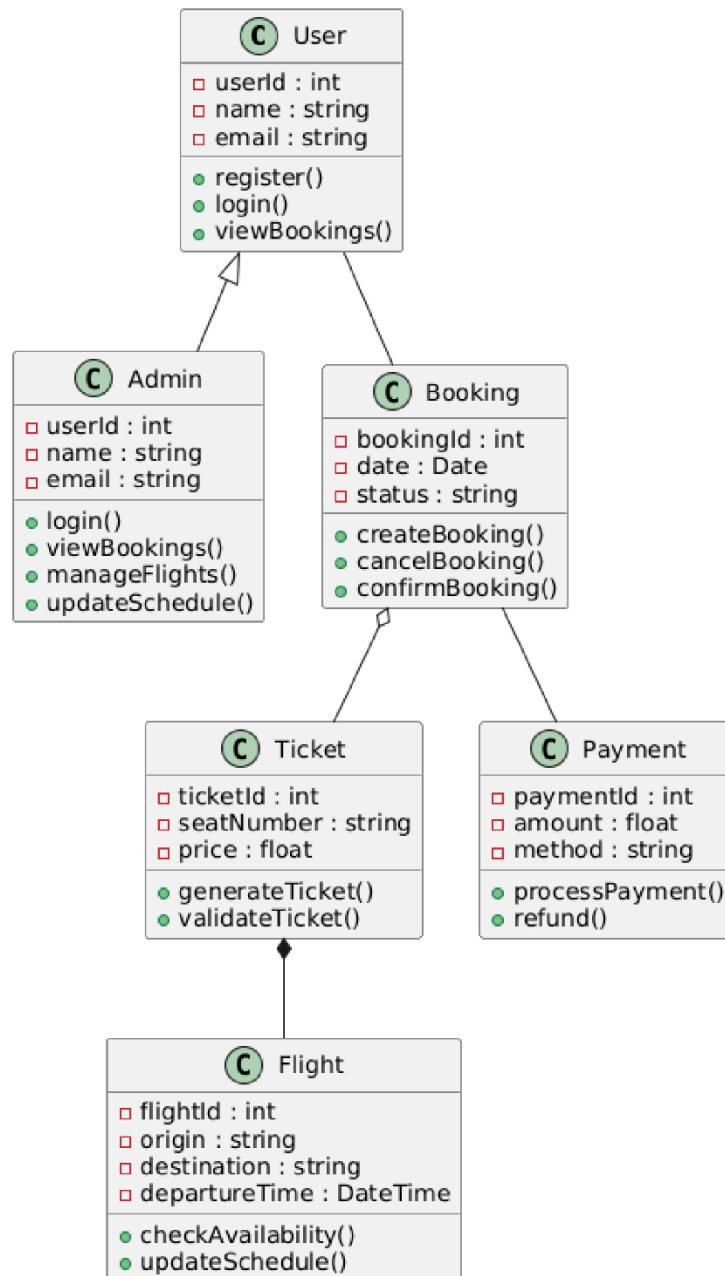
## Тип архітектури:

- **Шарова (Layered Architecture)** з реалізацією принципу клієнт-сервер.
- **Presentation Layer (UI/Client)** - інтерфейс для користувачів та адміністраторів.
- **Business Logic Layer** - сервіси для бронювання, керування рейсами, платежів та генерації квитків.
- **Data Access Layer** - база даних для збереження інформації про користувачів, рейси, бронювання, квитки та платежі.

## Основні компоненти:

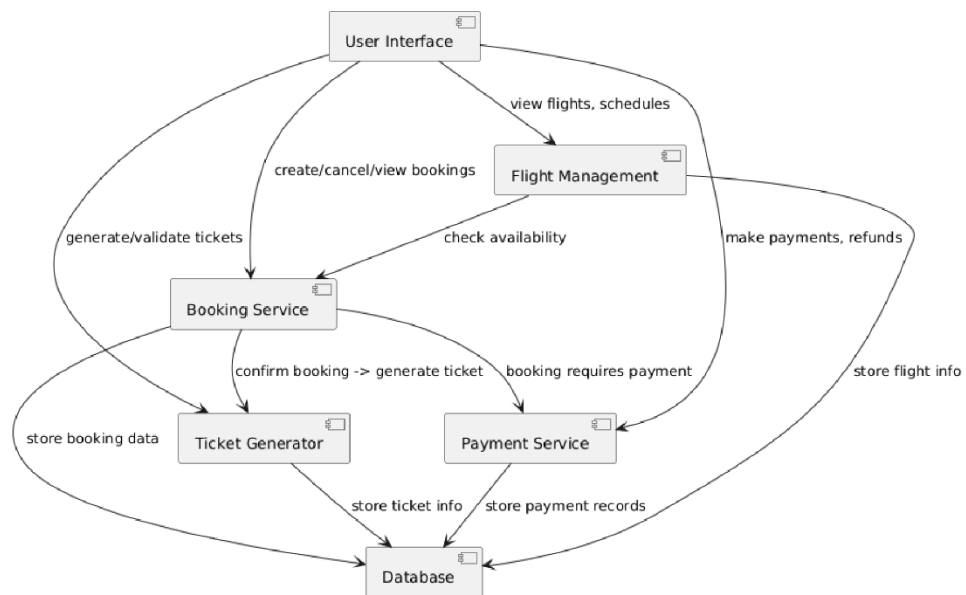
- **User Interface** - взаємодія з користувачем.
- **Booking Service** - створення, підтвердження та скасування бронювань.
- **Flight Management** - управління рейсами та розкладом.
- **Payment Service** - обробка платежів та повернень.
- **Ticket Generator** - генерація та перевірка квитків.
- **Database** - централізоване сховище даних.

## UML-діаграми:



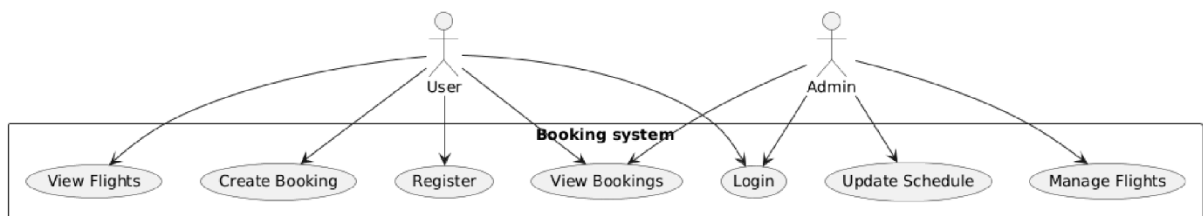
Class diagram

Component Diagram of booking flights



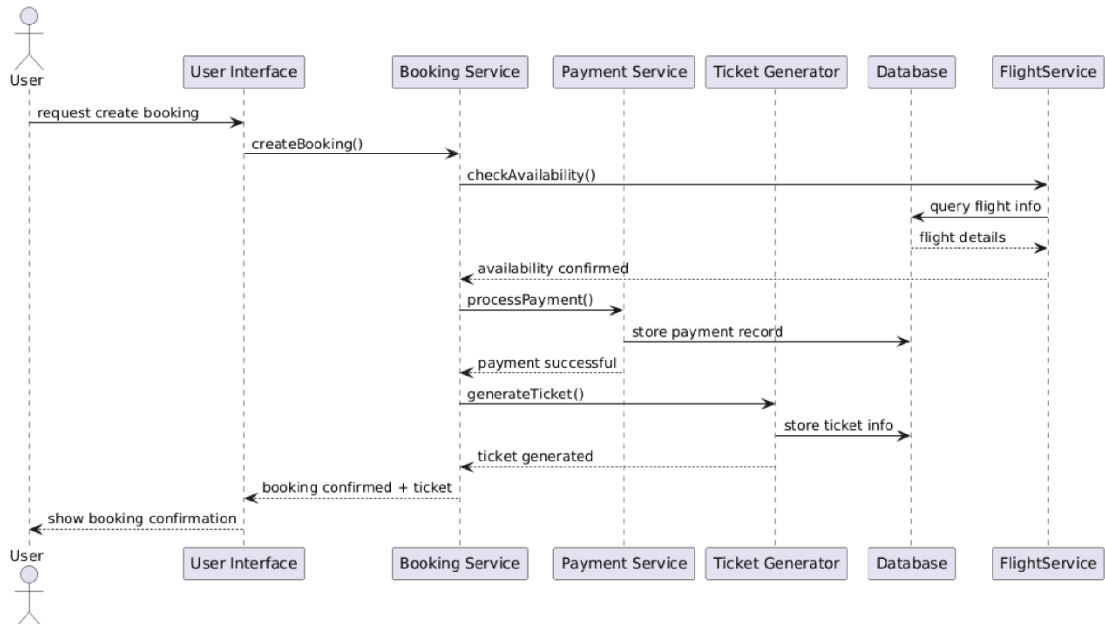
Component diagram

Use Case Diagram of create booking



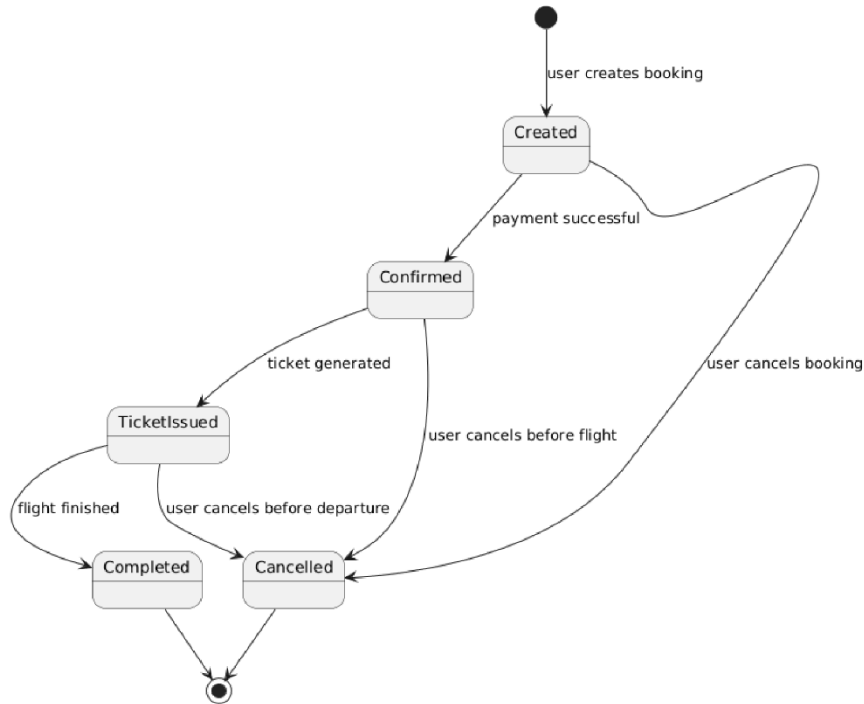
Use-case diagram

Sequence Diagram of create booking



Sequence diagram

### State Machine Diagram - Booking



State-machine diagram

### Короткі пояснення:

- ❖ **Class Diagram** - показує модель предметної області та зв'язки між сутностями.
- ❖ **Component Diagram** - відображає архітектурні компоненти та їхню взаємодію.
- ❖ **Use Case Diagram** - визначає ролі акторів та їхні можливості.
- ❖ **Sequence Diagram** - деталізує кроки виконання одного сценарію (створення бронювання).
- ❖ **State Machine Diagram** - описує зміни станів об'єкта бронювання протягом його життєвого циклу.