
<Group 07>

<QUẢN LÝ BÁN VÉ CHUYẾN BAY>
Software Architecture Document

Version <1.0>

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

Revision History

Date	Version	Description	Author
<14/05/2025>	<1.0>	Thực hiện viết Software Architecture Document	Hà Đức Huy Trần Gia Hào Nguyễn Minh Hưng Nguyễn Tấn Hưng Nguyễn Lê Anh Phúc

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

Table of Contents

1.	Introduction	4
2.	Architectural Goals and Constraints	5
3.	Use-Case Model	6
4.	Logical View	7
4.1	Frontend Layer:	8
4.1.1.	Component: Authentication:	8
4.1.2.	Component: Airport:	8
4.1.3.	Component: Flight:	9
4.1.4.	Component: Ticket:	9
4.1.5.	Component: Regulation:	10
4.1.6.	Component: Report:	10
4.2	Backend Layer:	11
4.2.1.	Component: Authentication API	11
4.2.2.	Component: Airport API	12
4.2.3.	Component: Flight API	12
4.2.4.	Component: Tickets API	13
4.2.5.	Component: Regulations API	13
4.2.6.	Component: Report API	14
5.	Class Diagrams:	14
6.	DB design:	15
7.	Deployment:	17
8.	Implementation View:	17

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

Software Architecture Document

1. Introduction

• Purpose

Tài liệu Software Architecture Document này mô tả tổng thể kiến trúc của hệ thống **Quản lý bán vé chuyến bay**, hỗ trợ nhân viên và khách hàng thực hiện các thao tác liên quan đến việc tìm kiếm, đặt vé, quản lý và xuất vé các chuyến bay. Tài liệu cung cấp cái nhìn toàn cảnh về các thành phần chính của hệ thống, cách chúng tương tác và công nghệ được sử dụng. Đây là cơ sở để triển khai, kiểm thử, bảo trì và mở rộng hệ thống trong tương lai.

• Scope

Hệ thống cung cấp các chức năng chính sau:

- **Tìm kiếm chuyến bay** theo điểm đi, điểm đến, ngày bay dành cho khách hàng và nhân viên.
- **Quản lý chuyến bay**: Nhân viên có thể thêm, sửa, xóa chuyến bay.
- **Quản lý phiếu đặt chỗ**: Thêm, sửa, xóa, tra cứu phiếu đặt chỗ; chuyển phiếu thành vé chuyến bay.
- **Quản lý vé chuyến bay**: Sửa, xóa, tra cứu vé đã xuất.
- **Xác thực tài khoản nhân viên**: Nhân viên có thể đăng nhập, đăng xuất.
- **Báo cáo doanh thu**: Nhân viên có thể xem doanh thu theo tháng và năm.
- **Quản lý quy định hệ thống**: Nhân viên có quyền thay đổi một số quy định về đặt chỗ, thời gian giữ vé,...

• Software Architecture Style

Hệ thống áp dụng kiến trúc **MVC (Model - View – Controller)** với các yêu cầu:

- **Frontend**: Xây dựng bằng ReactJS, xử lý giao diện người dùng, gửi yêu cầu đến backend qua API.
- **Backend**: Xây dựng bằng FastAPI (Python), xử lý nghiệp vụ, xác thực, kết nối cơ sở dữ liệu.
- **Database**: PostgreSQL, lưu trữ thông tin về chuyến bay, phiếu đặt chỗ, vé, quy định,...

• Technologies and Component Interaction

- Frontend ↔ Backend: Giao tiếp qua RESTful API (HTTP/HTTPS, dữ liệu dạng JSON).
- Backend ↔ Database: Giao tiếp qua ORM (Django ORM hoặc SQLAlchemy).
- Backend tích hợp với dịch vụ email và thanh toán qua HTTPS API nếu cần mở rộng sau này.

• Definitions, Acronyms, and Abbreviations

Từ viết tắt	Giải nghĩa
MVC	Model-View-Controller
API	Application Programming Interface
DB	Database
ORM	Object-Relational Mapping
HTTP/HTTPS	Giao thức truyền tải siêu văn bản có bảo mật
JSON	Định dạng trao đổi dữ liệu

• References: Tài liệu lý thuyết trên lớp

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

- **Overview: Phần còn lại của tài liệu gồm:**
 - Architectural Goals and Constraints
 - Use-case Models
 - Logical View
 - Deployment and Implementation View (Dành cho PA4)

2. Architectural Goals and Constraints

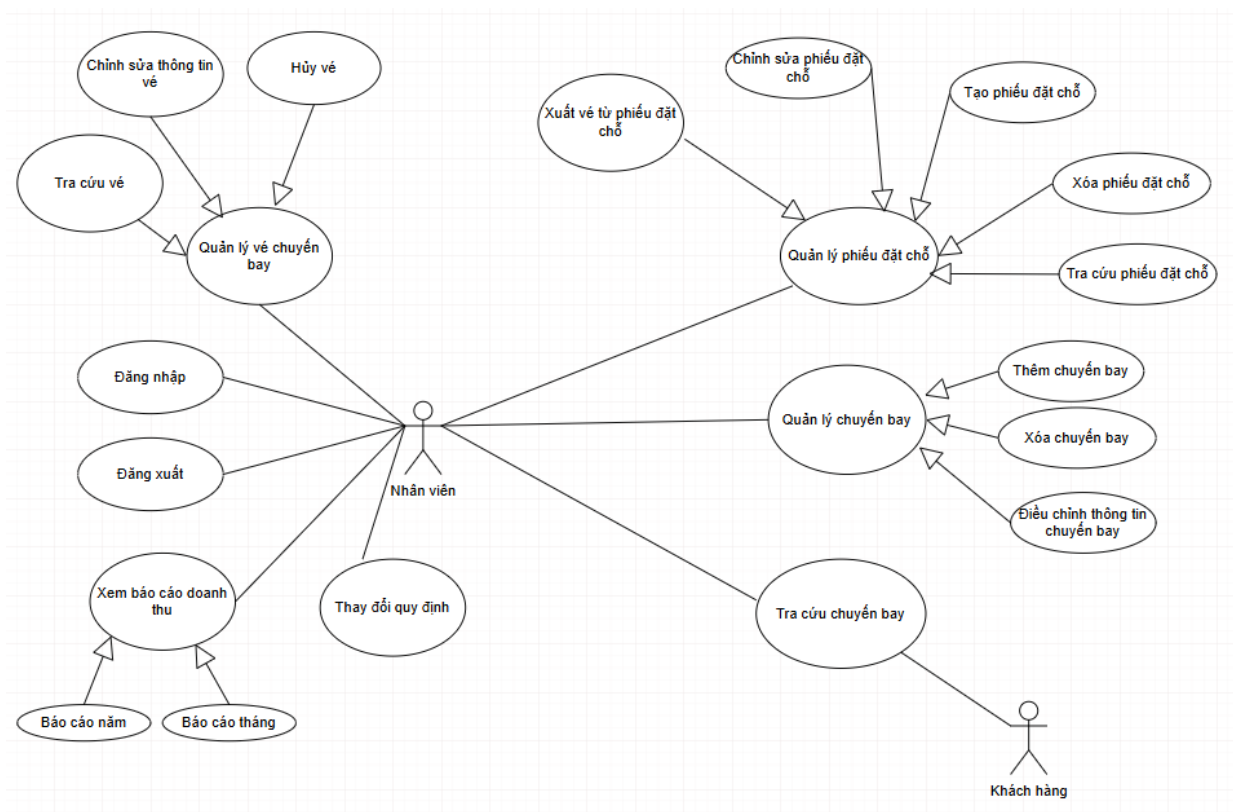
- **Architectural Goals**
 - **Scalability (Khả năng mở rộng)**
 - Hệ thống phải có khả năng xử lý tốt khi số lượng người dùng, chuyến bay, giao dịch tăng lên.
 - Hệ thống cần được thiết kế linh hoạt để dễ dàng bảo trì, cập nhật và mở rộng trong tương lai.
 - Việc tích hợp các tính năng mới phải được thực hiện một cách dễ dàng mà không làm gián đoạn hoạt động kinh doanh
 - **Security (Bảo mật)**
 - Hệ thống quản lý bán vé chuyến bay phải đảm bảo an toàn cho tài khoản của hãng, đại lý bay; thông tin cá nhân của hành khách, bao gồm thông tin đăng nhập, thông tin đặt vé,
 - Tất cả dữ liệu nhạy cảm cần được mã hóa trước khi lưu trữ và truyền tải để ngăn chặn truy cập trái phép.
 - Phân quyền rõ ràng cho nhân viên và khách hàng.
 - **Availability (Tính khả dụng cao)**
 - Hệ thống nên hỗ trợ cơ chế phát hiện lỗi và phục hồi như retry, failover.
 - Có thể triển khai các thành phần dư thừa (redundant) để đảm bảo tính liên tục khi gặp sự cố.
 - **Reusability (Tái sử dụng thành phần)**
 - Chức năng xử lý tìm kiếm, quản lý chuyến bay, quản lý vé được xây dựng thành các module có thể tái sử dụng cho các nền tảng khác (web...).
 - **Maintainability (Dễ bảo trì)**
 - Mỗi chức năng chính (quản lý chuyến bay, đặt vé, báo cáo, cấu hình...) được tách thành module riêng biệt.
 - Tuân theo mô hình kiến trúc MVC giúp dễ thay đổi logic nghiệp vụ mà không ảnh hưởng đến giao diện hoặc cơ sở dữ liệu.
 - **Performance (Hiệu năng)**
 - Tối ưu truy vấn cơ sở dữ liệu và sử dụng cơ chế caching để giảm thời gian phản hồi khi thao tác trên hệ thống và giúp hệ thống duy trì hiệu suất ổn định
 - Hệ thống phải đảm bảo khả năng hoạt động trơn tru, xử lý nhanh chóng các yêu cầu từ khách hàng và nhân viên, bao gồm tìm kiếm chuyến bay, đặt vé, thanh toán...
 - Ưu tiên sử dụng các thành phần có độ “thô” lớn hơn để giảm giao tiếp giữa các lớp.
 - **Safety (Tính an toàn hệ thống)**
 - Các quy trình ảnh hưởng đến doanh thu (xuất vé, chỉnh sửa vé) cần ghi log và giới hạn truy cập.
 - Các chức năng liên quan đến quy định hệ thống cần xác thực kỹ lưỡng.

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

• Architectural Constraints

- **Ngôn ngữ lập trình:** Frontend sử dụng JavaScript (ReactJS), backend sử dụng Python (FastAPI).
- **Tuân thủ theo mô hình MVC.**
- **Giao tiếp RESTful API,** chuẩn dữ liệu JSON.
- **Sử dụng PostgreSQL làm hệ quản trị cơ sở dữ liệu quan hệ.**
- **Hệ thống không sử dụng giải pháp cloud hoặc dịch vụ bên ngoài ở giai đoạn đầu.**
- **Thời gian phát triển giới hạn trong 10 tuần, đội phát triển gồm 5 người.**
- **Giao diện thân thiện, tương thích trên trình duyệt desktop trước, mobile sau.**

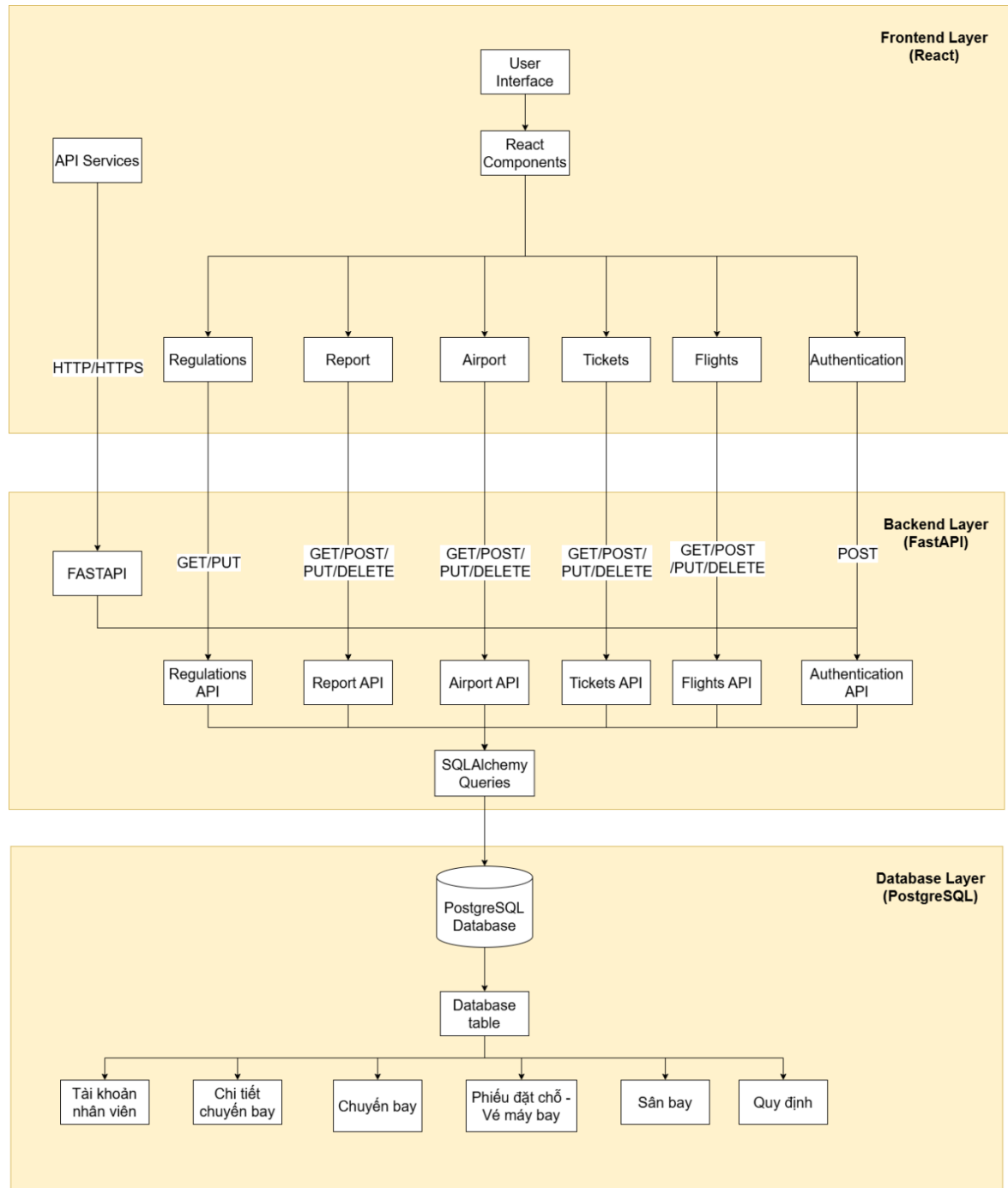
3. Use-Case Model



Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

4. Logical View

Hệ thống Quản lý Bán vé Chuyến bay được thiết kế với ba thành phần chính: Giao diện người dùng (Frontend), Backend và Cơ sở dữ liệu (Database). Hệ thống sử dụng FastAPI để xây dựng các API RESTful ở phía Backend và PostgreSQL để quản lý dữ liệu trong Cơ sở dữ liệu, với việc giao tiếp được thực hiện qua giao thức HTTP/HTTPS giữa Frontend và Backend, và các truy vấn SQLAlchemy giữa Backend và Database.



Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

4.1 Frontend Layer:

Nhiệm vụ

- Hiển thị giao diện người dùng cho nhân viên để quản lý chuyến bay, vé, đặt chỗ, báo cáo và quy định.
- Hiển thị nội dung động (kết quả tra cứu chuyến bay, cài đặt quy định) từ các endpoint FastAPI.
- Thu thập đầu vào từ người dùng (thông tin đăng nhập, tiêu chí tra cứu chuyến bay) và gửi yêu cầu HTTP đến Backend.

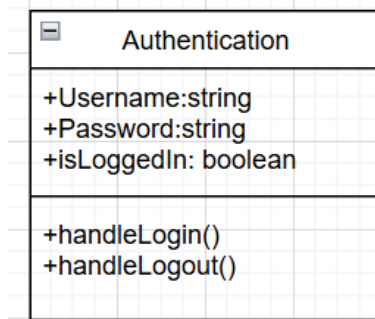
Dịch vụ cung cấp:

- Giao diện đăng nhập để xác thực nhân viên.
- Giao diện quản lý sân bay để xem và chỉnh sửa thông tin sân bay.
- Giao diện tra cứu và quản lý chuyến bay để tra cứu và quản lý thông tin chuyến bay.
- Giao diện quản lý vé để xử lý vé và đặt chỗ.
- Giao diện quản lý quy định để cập nhật tham số hệ thống.
- Giao diện tạo báo cáo để xem báo cáo doanh thu và hiệu suất nhân viên.

Cách thức giao tiếp:

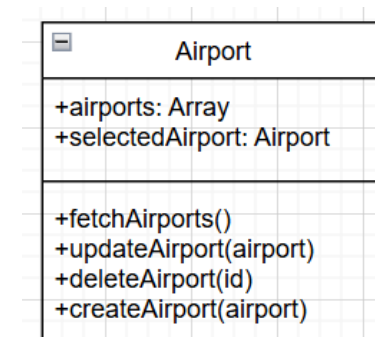
- Gửi yêu cầu **HTTP/HTTPS** đến Backend: FastAPI qua các endpoint RESTful (ví dụ: /api/v1/flights/search).
- Nhận phản hồi JSON từ Backend để cập nhật giao diện.

4.1.1. Component: Authentication:



- **Định nghĩa:** Xử lý đăng nhập và đăng xuất nhân viên.
- **Thuộc tính:**
 - username: string: Tên đăng nhập người dùng.
 - password: string: Mật khẩu người dùng.
 - isLoggedIn: bool: Trạng thái đăng nhập.
- **Phương thức:**
 - handleLogin(): Xử lý đăng nhập.
 - handleLogout(): Xử lý đăng xuất.

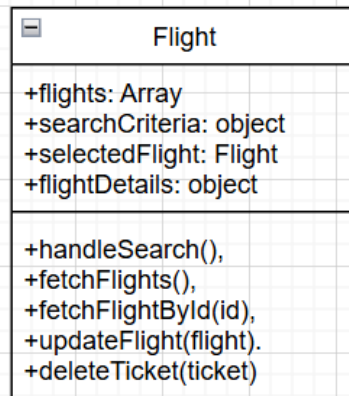
4.1.2. Component: Airport:



Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

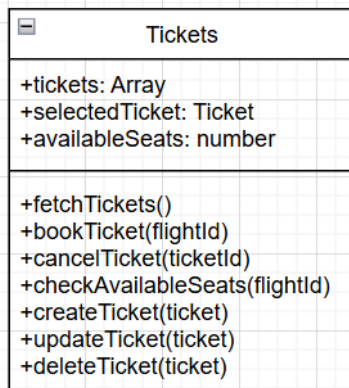
- **Định nghĩa:** Quản lý và chỉnh sửa thông tin sân bay.
- **Thuộc tính:**
 - airports: Array<Airport>: Danh sách sân bay.
 - selectedAirport: Airport: Sân bay được chọn.
- **Phương thức:**
 - fetchAirports(): Lấy danh sách sân bay.
 - updateAirport(airport): Cập nhật thông tin sân bay.
 - deleteAirport(id): Xóa sân bay.
 - createAirport(airport): Tạo sân bay mới.

4.1.3. Component: Flight:



- **Định nghĩa:** Tra cứu và quản lý thông tin chuyến bay.
- **Thuộc tính:**
 - flights: Array<Flight>: Danh sách chuyến bay.
 - searchCriteria: object: Tiêu chí tra cứu.
 - selectedFlight: Flight: Chuyến bay được chọn.
 - flightDetails: object: Chi tiết chuyến bay.
- **Phương thức:**
 - handleSearch(): Xử lý tra cứu chuyến bay.
 - fetchFlights(): Lấy danh sách chuyến bay.
 - fetchFlightById(id): Lấy chi tiết chuyến bay.
 - updateFlight(flight): Cập nhật thông tin chuyến bay.
 - deleteFlight(id): Xóa chuyến bay.

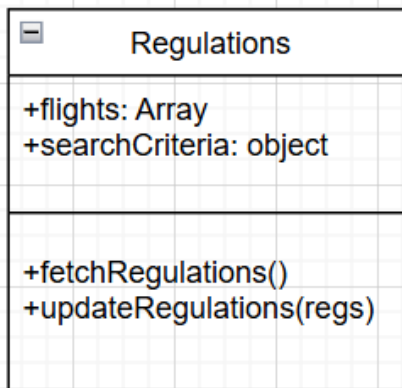
4.1.4. Component: Ticket:



Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

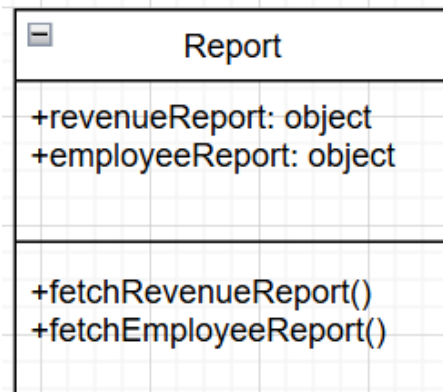
- **Định nghĩa:** Xử lý vé, đặt chỗ và kiểm tra ghế trống.
- **Thuộc tính:**
 - tickets: Array<Ticket>: Danh sách vé.
 - selectedTicket: Ticket: Vé được chọn.
 - availableSeats: number: Số ghế trống.
- **Phương thức:**
 - fetchTickets(): Lấy danh sách vé.
 - bookTicket(flightId): Đặt vé cho chuyến bay.
 - cancelTicket(ticketId): Hủy vé đã đặt.
 - checkAvailableSeats(flightId): Kiểm tra ghế trống.
 - createTicket(ticket): Tạo vé mới.
 - updateTicket(id, ticket): Cập nhật thông tin vé.
 - deleteTicket(id): Xóa vé.

4.1.5. Component: Regulation:



- **Định nghĩa:** Quản lý và cập nhật quy định hệ thống.
- **Thuộc tính:**
 - regulations: object: Quy định hiện tại.
- **Phương thức:**
 - fetchRegulations(): Lấy quy định hệ thống.
 - updateRegulations(regs): Cập nhật quy định.

4.1.6. Component: Report:



Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

- **Định nghĩa:** Tạo và hiển thị báo cáo doanh thu, hiệu suất nhân viên.
- **Thuộc tính:**
 - revenueReport: object: Báo cáo doanh thu.
 - employeeReport: object: Báo cáo hiệu suất nhân viên.
- **Phương thức:**
 - fetchRevenueReport(): Lấy báo cáo doanh thu.
 - fetchEmployeeReport(): Lấy báo cáo hiệu suất.

4.2 Backend Layer:

Nhiệm vụ

- Xử lý yêu cầu HTTP từ Frontend bằng các tuyến FastAPI và thực hiện logic nghiệp vụ
- Kiểm tra đầu vào và thực thi quy tắc nghiệp vụ (ví dụ: kiểm tra chuyến bay)
- Phối hợp với Database qua SQLAlchemy để lấy hoặc cập nhật dữ liệu

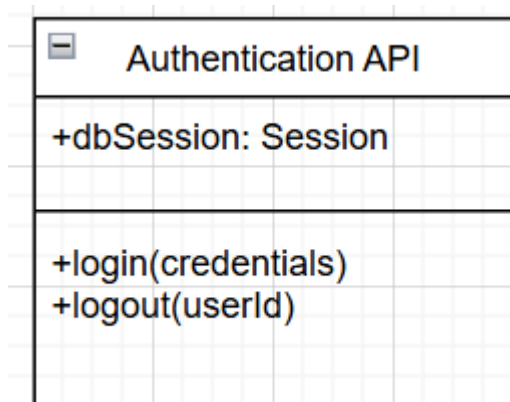
Dịch vụ cung cấp

- API Xác thực: Quản lý đăng nhập và đăng xuất
- API Sân bay: Quản lý CRUD cho sân bay
- API Chuyến bay: Quản lý CRUD và tra cứu chuyến bay
- API Vé: Quản lý CRUD, đặt vé, hủy vé, kiểm tra ghế trống
- API Quy định: Quản lý cập nhật quy định hệ thống
- API Báo cáo: Tạo báo cáo doanh thu và hiệu suất nhân viên

Giao tiếp

- Nhận yêu cầu HTTP/HTTPS từ Frontend qua các tuyến FastAPI (ví dụ: /api/v1/flights)
- Gửi truy vấn SQLAlchemy đến Database: PostgreSQL (ví dụ: SELECT * FROM ChuyenBay) và trả về phản hồi JSON

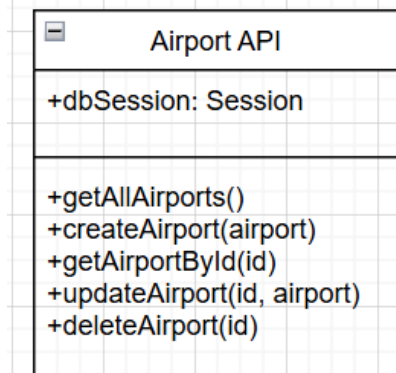
4.2.1. Component: Authentication API



- **Định nghĩa:** Quản lý xác thực và đăng xuất nhân viên
- **Thuộc tính:**
 - dbSession: Session (phiên SQLAlchemy truy vấn dữ liệu)
- **Phương thức:**
 - login(credentials): Xác thực đăng nhập nhân viên
 - logout(userId): Đăng xuất nhân viên

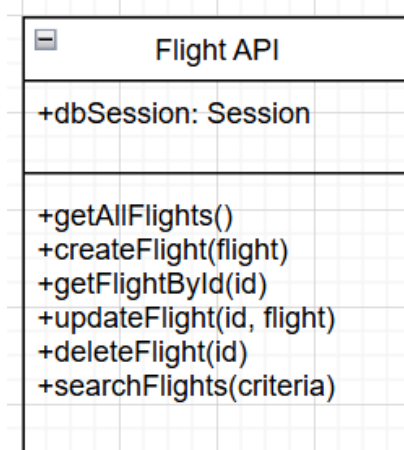
Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

4.2.2. Component: Airport API



- Định nghĩa: Quản lý CRUD cho sân bay
- Thuộc tính:
 - dbSession: Session
- Phương thức:
 - getAllAirports(): Lấy danh sách sân bay
 - createAirport(airport): Tạo sân bay mới
 - getAirportById(id): Lấy chi tiết sân bay
 - updateAirport(id, airport): Cập nhật sân bay
 - deleteAirport(id): Xóa sân bay

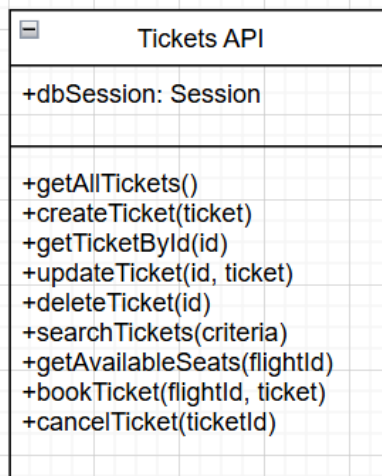
4.2.3. Component: Flight API



- Định nghĩa: Quản lý CRUD và tra cứu chuyến bay
- Thuộc tính:
 - dbSession: Session
- Phương thức:
 - getAllFlights(): Lấy danh sách chuyến bay
 - createFlight(flight): Tạo chuyến bay mới
 - getFlightById(id): Lấy chi tiết chuyến bay
 - updateFlight(id, flight): Cập nhật chuyến bay
 - deleteFlight(id): Xóa chuyến bay
 - searchFlights(criteria): Tìm kiếm chuyến bay

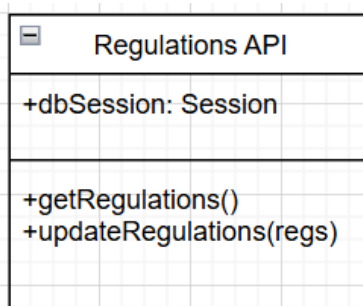
Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

4.2.4. Component: Tickets API



- Định nghĩa: Quản lý vé, đặt chỗ và hủy vé
- Thuộc tính:
 - dbSession: Session
- Phương thức:
 - getAllTickets(): Lấy danh sách vé
 - createTicket(ticket): Tạo vé mới
 - getTicketById(id): Lấy chi tiết vé
 - updateTicket(id, ticket): Cập nhật vé
 - deleteTicket(id): Xóa vé
 - searchTickets(criteria): Tìm kiếm vé
 - getAvailableSeats(flightId): Kiểm tra ghế trống
 - bookTicket(flightId, ticket): Đặt vé cho chuyến bay
 - cancelTicket(ticketId): Hủy vé đã đặt

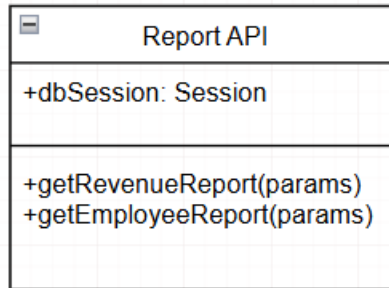
4.2.5. Component: Regulations API



- Định nghĩa: Quản lý quy định hệ thống
- Thuộc tính:
 - dbSession: Session
- Phương thức:
 - getRegulations(): Lấy quy định hiện tại
 - updateRegulations(regs): Cập nhật quy định hệ thống

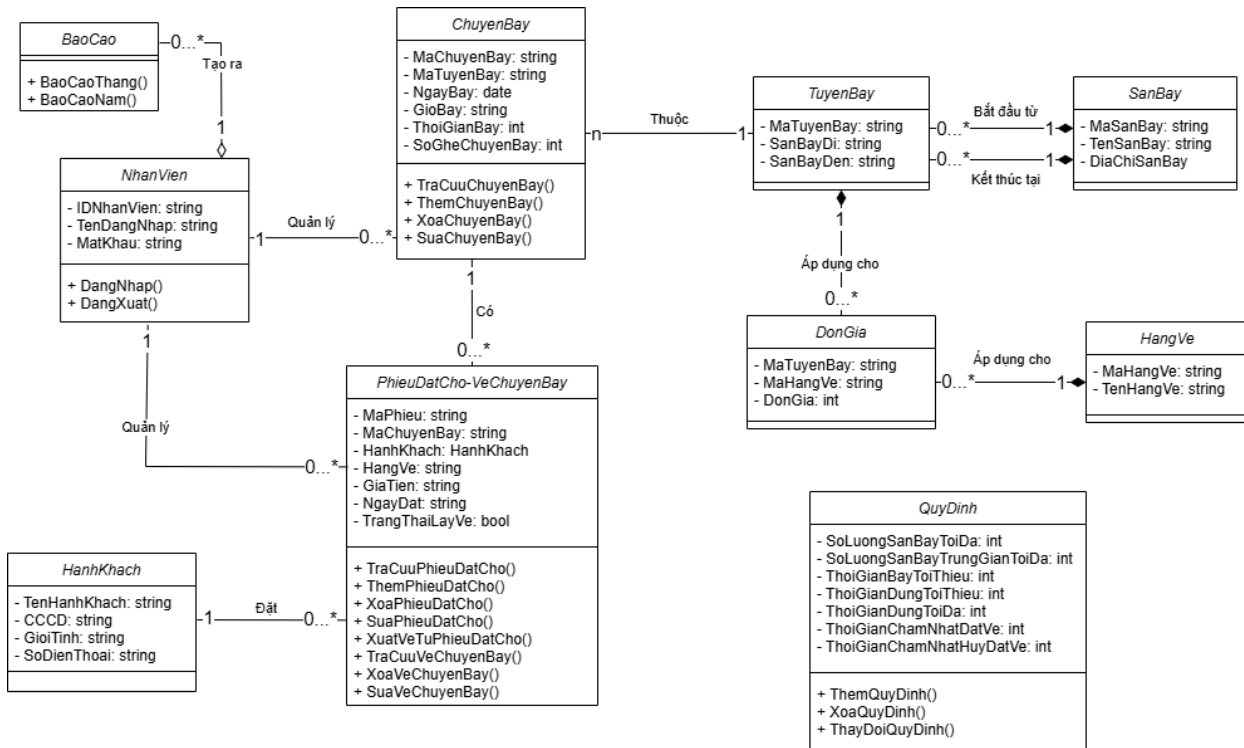
Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

4.2.6. Component: Report API



- Định nghĩa: Tạo báo cáo doanh thu và hiệu suất
- Thuộc tính:
 - dbSession: Session
- Phương thức:
 - getRevenueReport(params): Lấy báo cáo doanh thu
 - getEmployeeReport(params): Lấy báo cáo hiệu suất

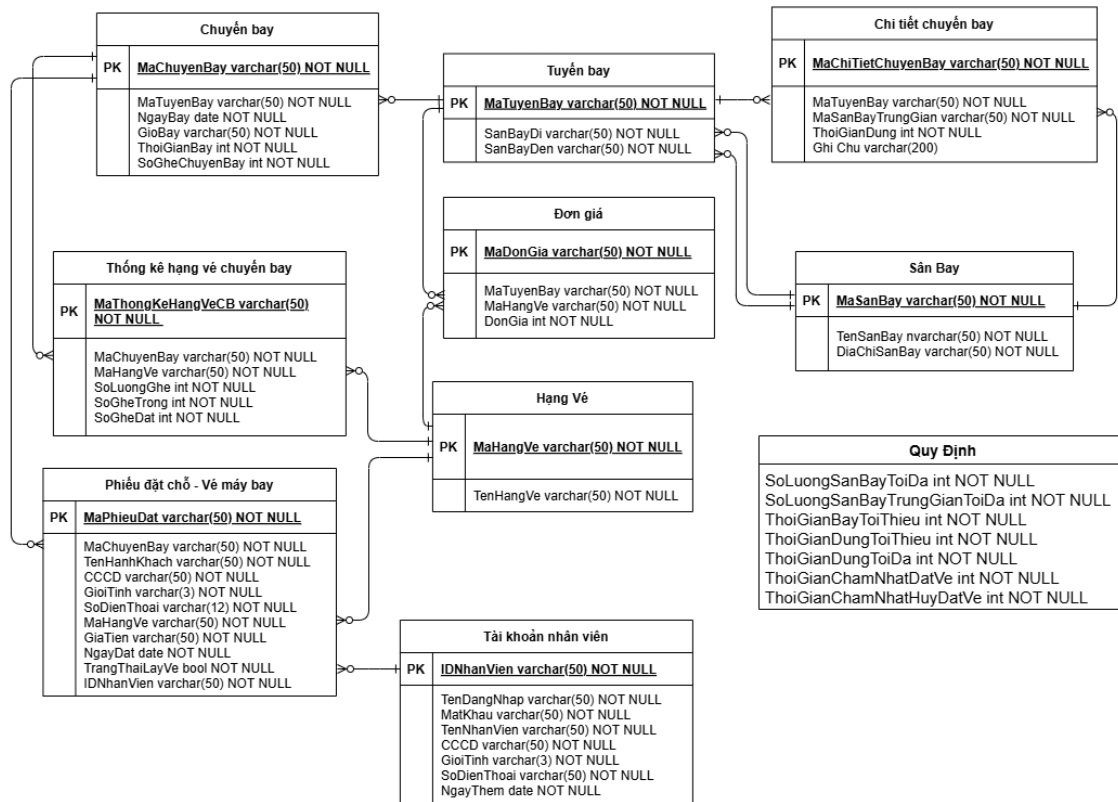
5. Class Diagrams:



Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

6. DB design:

Dưới đây là thiết kế cơ sở dữ liệu của hệ thống Quản lý Bán vé Chuyến bay, được trình bày dưới dạng sơ đồ ERD. Mô hình thể hiện rõ các thực thể (entities), thuộc tính (attributes) và các mối quan hệ (relationships) giữa chúng.



1. Thực thể và thuộc tính

- **Chuyến bay (ChuyenBay)**
 - MaChuyenBay (PK): Mã chuyến bay
 - MaTuyenBay: Mã tuyến bay
 - NgayBay: Ngày bay
 - GioBay: Giờ bay
 - ThoiGianBay: Thời gian bay
 - SoGheChuyenBay: Số ghế trong chuyến bay đó
- **Tuyến bay (TuyenBay)**
 - MaTuyenBay (PK): Mã tuyến bay
 - SanBayDi: Sân bay đi
 - SanBayDen: Sân bay đến
- **Sân bay (SanBay)**
 - MaSanBay (PK): Mã sân bay
 - TenSanBay: Tên sân bay
 - DiaChi: Địa chỉ sân bay
- **Chi tiết chuyến bay (ChiTietChuyenBay)**
 - MaChiTietChuyenBay (PK): Mã chi tiết chuyến bay
 - MaTuyenBay: Mã tuyến bay
 - MaSanBayTrungGian: Mã sân bay trung gian
 - ThoiGianDung: Thời gian dừng
 - GhiChu: Ghi chú
- **Hạng vé (HangVe)**

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

- MaHangVe (PK): Mã hạng vé
- TenHangVe: Tên hạng vé
- **Thông kê hạng vé chuyến bay (ThongKeHangVeChuyenBay):** Nhằm biết được số lượng ghế, số lượng ghế trống và số lượng ghế đặt của một hạng vé trên một chuyến bay
 - MaThongKeHangVeCB (PK): Mã thông kê cho từng hạng vé trong 1 chuyến bay
 - MaChuyenBay: Mã chuyến bay
 - MaHangVe: Mã của hạng vé xuất hiện trong chuyến bay
 - SoGheTrong: Số ghế trống ở hạng vé đó trên chuyến bay
 - SoGheDat: Số ghế đã đặt của hạng vé đó trên chuyến bay
- **Đơn giá (DonGia)**
 - MaDonGia (PK): Mã đơn giá
 - MaTuyenBay: Mã tuyến bay
 - MaHangVe: Mã hạng vé
 - GiaTien: Giá tiền tương ứng với hạng vé ở tuyến bay đó
- **Phiếu đặt chỗ - Vé máy bay (PhieuDatCho_VeMayBay)**
 - MaPhieuDat (PK): Mã đặt vé
 - MaChuyenBay: Mã chuyến bay
 - TenHanhKhach: Tên hành khách
 - CCCD: Căn cước công dân
 - GioiTinh: Giới tính của khách hàng
 - SoDienThoai: Số điện thoại
 - MaHangVe: Mã hạng vé
 - GiaTien: Giá tiền
 - NgayDat: Ngày đặt
 - TrangThaiLayVe: Trạng thái lấy vé (true: Vé chuyến bay/false: Phiếu đặt chỗ)
 - IDNhanVien: Mã nhân viên thực hiện
- **Tài khoản nhân viên (TaiKhoanNhanVien)**
 - IDNhanVien (PK): Mã nhân viên
 - TenDangNhap: Tên đăng nhập
 - MatKhau: Mật khẩu
 - TenNhanVien: Tên nhân viên
 - CCCD: Căn cước công dân
 - GioiTinh: Giới tính nhân viên
 - SoDienThoai: Số điện thoại
 - NgayThem: Ngày thêm vào hệ thống
- **Quy định (QuyDinh):** Là bảng các tham số quy định của hệ thống, gồm:
 - SoLuongSanBayToiDa
 - SoLuongSanBayTrungGianToiDa
 - ThoiGianBayToiThieu, ToiDa
 - ThoiGianDungToiThieu, ToiDa
 - ThoiGianChamNhatDatVe
 - ThoiGianChamNhatHuyDatVe

2. Các mối quan hệ chính

- **ChuyenBay ↔ TuyenBay:** Mỗi chuyến bay thuộc một tuyến bay.
- **TuyenBay ↔ SanBay:** Tuyến bay gồm sân bay đi và sân bay đến.
- **TuyenBay ↔ ChiTietChuyenBay:** Một tuyến bay có thể có nhiều sân bay trung gian.
- **ChuyenBay ↔ ThongKeHangVeChuyenBay:** Mỗi chuyến bay có các thông kê cho nhiều hạng vé khác nhau trong chuyến bay đó.
- **TuyenBay + HangVe ↔ DonGia:** Mỗi tuyến bay và hạng vé có giá riêng.
- **ChuyenBay + HangVe ↔ PhieuDatCho_VeMayBay:** Mỗi phiếu đặt chỗ/vé chuyến bay liên quan đến một chuyến bay và một hạng vé.
- **PhieuDatCho_VeMayBay ↔ NhanVien:** Mỗi phiếu đặt chỗ/vé chuyến bay do một nhân viên thực hiện.

Quản lý bán vé chuyến bay	Version: <1.0>
Software Architecture Document	Date: <14/05/2025>
PA3 – Group 07 – Software Architecture Document	

7. Deployment:

<Phần này sẽ trình bày ở PA4>.

8. Implementation View:

<Phần này sẽ trình bày ở PA4>.