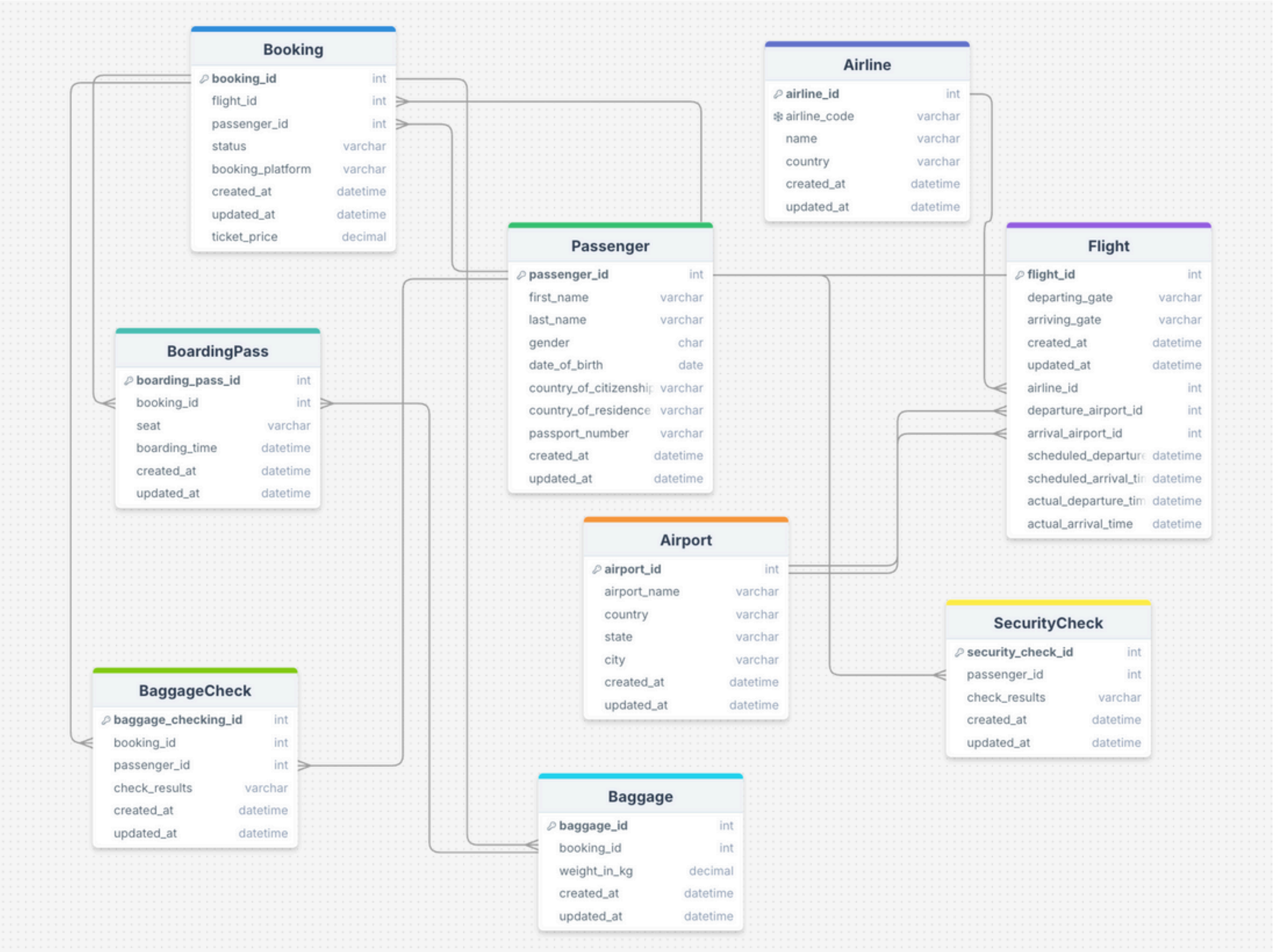

Airport → Stores airport details (name, city, country, etc.).

Airline → List of airlines with their codes and countries.

Flight → Flight information: airline, departure/arrival airports, time, gates, etc.

Passenger → Passenger data (name, date of birth, citizenship, passport, etc.).

Booking → Reservations made by passengers for flights (status, price, platform, etc.).

BookingChange → History of modifications to bookings (type of change, old/new values, timestamp, etc.).

BoardingPass → Boarding passes issued for bookings (seat assignment, boarding time, etc.).

Baggage → Registered baggage items for bookings (weight and details, etc.).

BaggageCheck → Results of baggage inspections (security clearance, results, etc.).

SecurityCheck → Security checks performed on passengers (results of screening, etc.).

Legend of Relationships:

1:1 → One-to-One relationship

1:N → One-to-Many relationship

PK → Primary Key (unique record identifier)

FK → Foreign Key (reference to another table)

Overall Explanation:

The database is designed to manage airport operations, including flights, passengers, bookings, baggage, and security checks. It consists of nine entities: **Airport, Flight, Airline, Booking, BoardingPass, Baggage, BaggageCheck, Passenger, and SecurityCheck**. Each entity has a primary key and attributes, with relationships maintained through foreign keys. The model ensures data integrity and normalization (3NF). Relationships include one-to-many (e.g., one flight has many bookings), and one-to-one (e.g., one passenger has one security check). This structure supports efficient and organized airport services.