Project 1: Flight Booking System

1 UML Model

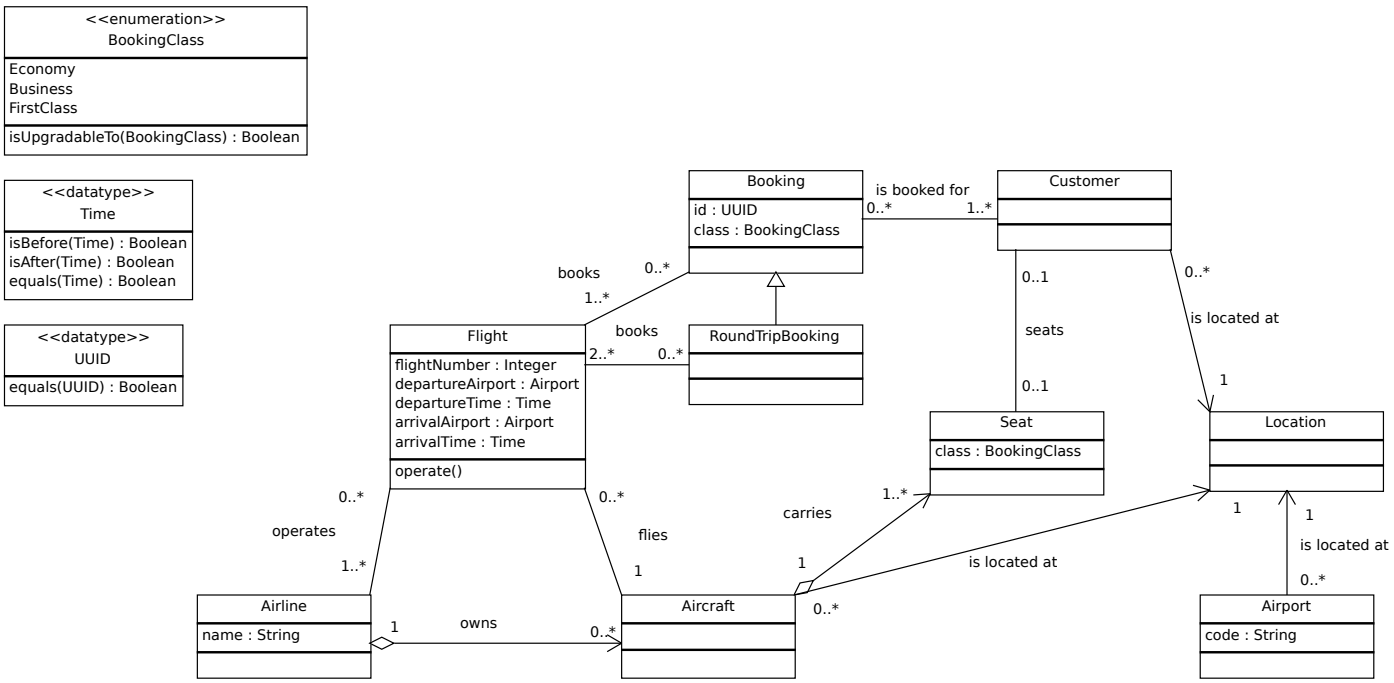

Figure 1: The UML model for the flight booking system

2 Additional Constraints

The following constraints are not expressible in the UML class diagram, but need to be upheld.

- Each `Flight`'s `departureAirport` and `arrivalAirport` must not match.
- Each subsequent `Flight`'s `departureAirport` in a `Booking` must match the preceding `Flight`'s `arrivalAirport`.
- Each `Flight` in a `Booking` must have a `departureTime` that `isBefore` its `arrivalTime`.
- Each subsequent `Flight`'s `departureTime` in a `Booking` `isAfter` the previous `Flight`'s `arrivalTime`.
- Each `Customer` associated with a `Booking` is tied to a `Seat` in each of the `Booking`'s `Flight`'s `Aircraft`, whose `class` either matches the `Booking`'s `class` or for which the `Booking`'s `class` `isUpgradableTo` the `Seat`'s `class`.
- `Economy` `isUpgradableTo` `Business` is true.
- `Business` `isUpgradableTo` `FirstClass` is true.
- Any other combination of `BookingClasses` with `isUpgradableTo` is false.
- A `Booking` may not be tied to the same `Customer` more than once.

- A `Customer` cannot be tied to a `Booking` whose `Flights`' `departureTime` `isBefore` and `arrivalTime` `isAfter` the `departureTime` of a `Flight` from another `Booking` tied to the same customer.
- An `Airplane` can only be tied to `Flights` whose `departureTimes` and `arrivalTimes` do not intersect.
- In a `RoundTripBooking`, the first `Flight`'s `departureAirport` must match the last `Flight`'s `arrivalAirport`.
- The `id` of every `Booking` must not be `equal` to any other `Booking`'s.
- The `name` of every `Airline` must not be `equal` to any other `Airline`'s.
- The `code` of every `Airport` must not be `equal` to any other `Airport`'s.
- The `flightNumber` of every `Flight` must not be `equal` to any other `Flight`'s whose `departureTimes` and `arrivalTimes` overlap.
- The `Airplane`'s `Location` must be equal to the current `Airport`'s `Location` whenever the `Airplane` is not in flight.