Contract CO2:	registerFlight
Operation:	registerFlight(number: String, source: Airport, destination: Airport, departureTime: String, arrivalTime: String, user: User)
Cross Reference:	Use case: Add Flights
Preconditions:	● An Airport or Airline admin is logged into the system.
Postconditions:	● An instance of Flights flights was created. ● An association from Airport source to flights was created. ● An association from Airport destination to flights was created. ● String departureTime and String arrivalTime was updated in an instance of Flight flight. ● Flight was added to flights.

Contract CO2:	verifyFlight
Operation:	verifyFlight(flight: Flight)
Cross Reference:	Use case: Add Flights
Preconditions:	● An Airport or Airline admin is logged into the system.
Postconditions:	● Value of true was returned if the Airport has enough Aircraft in its fleet, and departureTime and arrivalTime are unique for Source and Destination airports respectively. ● Value of false was returned if the Airport does not have enough Aircraft in its fleet, or departureTime and arrivalTime are not unique for Source and Destination airports respectively.