Phase II Schema and Unhandled Constraints
Project Team 31
Minyoung Cho, Madison O'Carroll, Rithvik Rajavelu, Jeremy Vach

RELATIONAL SCHEMA

Airline = (airlineID, revenue)

Location = (locID)

Airport = (airportID, city, state, location[fk1])
fk1: location → location.locID

Person = (personID, firstName, lastName, occupant[fk2])
fk2: occupant → location.locID NOT NULL

Leg = (legID, distance, departure[fk3], arrival[fk4])
fk3: departure → airport.airportID NOT NULL

fk4: arrival → airport.airportID NOT NULL

Route = (routeID)

Airplane = (airlineID[fk5], tail_num, speed, seat_cap, locationID[fk6], plane_type, num_engines, num_props)
fk5: airlineID → airline.airlineID NOT NULL

fk6: locationID → location.locID

Flight = (flightID, routeID[fk7], support_airline, support_tail[fk8], progress, status, next_time)
fk7: routeID → route.routeID NOT NULL

fk8: airlineID, support_tail → airplane.airlineID, airplane.tail_num

Ticket = (ticketID, cost, deplane_at[fk9], customer[fk10], carrier[fk11])
fk9: deplane_at → airport.airportID NOT NULL

fk10: customer → person.personID NOT NULL

fk11: carrier → flight.flightID NOT NULL

Seat = (ticketID[fk12], seat)
fk12: ticketID → ticket.ticketID

Contain = (routeID[fk13], legID[fk14], sequence)

fk13: routeID → route.routeID, routeID NOT NULL

fk14: legID → leg.legID NOT NULL

Pilot = (taxID, personID[fk15], experience, airlinesID, files[fk16])

fk15: personID → person.personID

fk16: airlinesID, files → airplane.airlineID, airplane.tail_num

Passenger = (personID[fk17], miles)

fk17: personID → person.personID

License = (personID[fk18], License-Name)

fk18: personID → person.personID

UNHANDLED CONSTRAINTS

1. Ensure an airplane's tail number is not empty.
2. Ensure pilot has sufficient experience to fly a plane
3. Ensure that plane speed is reasonable for the type of plane
4. Ensure that the seat capacity of a plane is reasonable for the type of plane
5. Ensure a pilots tax number is correctly formatted xxx-xx-xxxx
6. Ensure that a flight is not filled beyond seat capacity
7. Ensure that an airport's city is in the correct state, and vice versa.