The meaning of color:
red: translated relational schema
blue: discription of entity
yellow: FDs of the schema
purple: description of each relationship and its cardinality
green: fix of suggestions for the previous stages

Our FDs show that all the relational schema above are already in BCNF form,
because the LHS of all FDs are all super keys. Therefore, they are automatically in 3NF form.

description:
The Airlines entity represents different airline companies.
It stores information like the unique IATA code and the airline's name.

Operate

Airlines

IATA_airline
(2 character, e.g. UA)

name_airline

Serve(
IATA_airport: VARCHAR(3) [PK] [FK to Airports.IATA_airport],
IATA_airline: VARCHAR(2) [PK] [FK to Airlines.IATA_airline]
)

FDs: IATA_airport, IATA_airline --> IATA_airport, IATA_airline

Airlines(
IATA_airline: VARCHAR(2) [PK],
name_airline: VARCHAR(255)
)

FDs: IATA_airline --> name_airline

Serve

assumption:
For a given flight, it can only be
operated by one airline. For a
given airline, it can operate 0 or many
flights.

assumption:
For a given airline, it can be served by
0 or many airports. For a given airport,
it can serve 0 or many airlines.

Scheduled_departure
Departure_delay
Destination_airport
Scheduled_arrival
Arrival_delay
Cancelled
Cancellation_reason
Air_system_delay
Security_delay
Airline_delay
Late_aircraft_delay
Weather_delay
Diverted
Arrival_time
Schedule_time
Air_time
Distance
Departure_time

description:
Flights entity represents individual flight instances,
storing data like flight number, associated airline, origin airport,
destination airport, and various times and statuses related to the flight.

Flights

Year
Day
Month
Flight_number

description:
The Airports entity holds data regarding various airports,
storing attributes such as the IATA code, airport name, city,
state, country, latitude, and longitude.

Depart

Airports

IATA_airport
(3 character, e.g. ORD)

name_airport
city
state
longitude
atitude

assumption:
For a given flight, it can only
depart from one airport. For a
given airport, there could be
many flight departure.

Airports(
IATA_airport: VARCHAR(3) [PK],
name_airport: VARCHAR(255),
city: VARCHAR(255),
state: VARCHAR(2),
longitude: REAL,
altitude: REAL
)

FDs: IATA_airport --> name_airport, city, state, longitude, atitude

FDs: Year, Month, Day, Flight_number, IATA_airline
--> Origin_airport, ..., Departure_time

Flights(
Year: INTEGER [PK],
Month: INTEGER [PK],
Day: INTEGER [PK],
Flight_number: INTEGER [PK],
IATA_airline: VARCHAR(2) [PK] [FK to Airlines.IATA_airline],
Origin_airport: VARCHAR(3) [FK to Airports.IATA_airport],
Scheduled_departure: INTEGER,
Departure_delay: INTEGER,
Destination_airport: VARCHAR(3),
Scheduled_arrival: INTEGER,
Arrival_delay: INTEGER,
Cancelled: BOOL,
Cancellation_reason: CHAR,
Air_system_delay: INTEGER,
Security_delay: INTEGER,
Airline_delay: INTEGER,
Late_aircraft_delay: INTEGER,
Weather_delay: INTEGER,
Diverted: BOOL,
Arrival_time: INTEGER,
Schedule_time: INTEGER,
Air_time: INTEGER,
Distance: INTEGER,
Departure_time: INTEGER
)

description:
The Feedback entity contains feedback submitted by
users about particular flights, storing the feedback content along
with identifiers for the user and the flight in question.

description:
Users entity refers to the individuals who utilize the application.
It stores their unique user ID, username, password, and email address.

Relate

Feedbacks

feedbackId
content

Send

Users

Fix all suggestions for the previous stages:
delete: user can delete their own feedback

userId
username
password
email

assumption:
For a given flight, it can have 0 or
many feedbacks. For a given
feedback, it can only be related
to one flight.

assumption:
For a given feedback, it can only be
sent by one user. For a given user,
it can send 0 or many feedbacks.

Feedbacks(varchar(255)
feedbackID: INTEGER [PK],
content: VARCHAR(255),
userId: INTEGER [FK to Users.userId],
Year: INTEGER [FK to Flights.Year],
Month: INTEGER [FK to Flights.Month],
Day: INTEGER [FK to Flights.Day],
Flight_number: INTEGER [FK to Flights.Flight_number],
IATA_airline: VARCHAR(2) [FK to Flights.IATA_airline]
)

Users(
userId: INTEGER [PK] varchar(255)
username: VARCHAR(255),
email: VARCHAR(255),
password: VARCHAR(255)
)

FDs: userId --> username, password, email

FDs:feedbackId -> content, userId, Year, Month, Day, Flight_number, IATA_airline