

Backend:

apps.py: name the app

forms.py: create registration forms

urls.py: routes for the app

utils.py: utility functions

views.py: implementations for all the routes. Receive data from frontend, send data to the frontend, database operations.

manage.py: run the app

DB:

db.sql: database and table definitions

Frontend:

App/static/main.css: dashboard formatting

App/static/staff/piechart.css: format the pie chart in staff use cases

App/templates/booking_agent/:

base.html: base template

index.html: home page

login.html: login page

purchase.html: purchase tickets

register.html: register page

search.html: search for flights

view_commissions.html: view commission

view_top_customers.html: view corresponding top 5 customers

App/templates/customer/

base.html: base template

comment.html: comment page

index.html: home page

login.html: login page

purchase.html: purchase tickets

register.html: register page

search.html: search for flights

spending.html: view spending

App/templates/staff/

base.html: base template

index.html: home page

login.html: login page

register.html: register page

add_airplane.html: add airplane

add_airport.html: add airport

change_status.html: change flight status

compare_revenue.html: compare revenue between customer purchase and agent purchase

create_flight.html: create flight

most_frequent_customers.html: view most frequent customers

view_customers.html: view customers for each flight

view_flight_ratings.html: view ratings and comments given to each flight

view_flights_for_one_customer.html: view flights taken for specific customer

view_reports.html: view total amount of ticket sold based on ranges of dates

view_top_booking_agents.html: view top 5 booking agents based on tickets sales and amount of commission

view_top_destinations.html: view top 3 most popular destinations

main.html: generate public info page (including search flights, 3 types of login and 3 types of register entries)