**Project summary document:**

**Overview:**

Message from the developer: I was slow to get this assignment started as I had not been properly taught how to run a flask app or do the other things required to get the first webpage running, combined with the time constraints of having a full-time job and taking other courses. This has resulted in me focusing on the main functionalities of the website such as all the main pages and features, but details like filtering flights by time or displaying tables in alphabetical order have been skipped to save time. Additionally, I have not had sufficient time to test all the routing and sessions, there for the website can be fragile and prone to crashing, nor have I had time to figure out how to use python anywhere or GitHub, so I am not sure how my code will run once handed in, please be patient with it and if you run into errors, it helps to logout first before going back into the website.

**Assumptions:**

The website was basically designed and built on the assumption that it would be better to use an agile methodology and try build the main functions of the website as quickly as possible before coming back and refining details, bug testing/fixing ect but in the end I failed to build the last of the main functions and certainly did not have time to clean up details.

**Design decisions:**

My main two challenges were lack of skill and time, my design choices were often based around completing as much work as, as fast as possible in the simplest way for me to understand. This result in me often creating new templates and rewriting code when I should have been nesting templates and creating functions to reuse. I was initially challenged by my lack of skills and experience as I was learning how routes and different functions worked as I built the website however my challenges changed over time as my skills increased rapidly and I become more confident, but I was also challenged by the increasing complicity of the website and how different pieces operated together. My main python file was also starting to get quite large as a result of me not creating and using functions if I had more time I probably would have considering creating new python files to manage different parts of the website. Additionally I regret not having more time to finish off the website as I felt I now have the skills to complete all of the requirements laid out but was greatly constrained by the time I had available.

**GetCursor**

Function is used to connect user to the database, seemed easier to use a function that re-writing the code every time.

**getID**

Believe this code is meant to be used for created new flights or adding new passengers, the function is never actually called because the database seemed to create new id’s already

**@app.route("/")**

home page, needs work, currently does not serve any real purpose other than being the first page the user should land on

**@app.route("/admin")**

Serves as the login page for employees, also checks to see if the user is a manger or normal member of staff and creates a session accordingly, currently there are no permission differences between employee types as I did not complete the flight adding/editing section. Many of the admin features can be accessed via the URL as I ran out of time to figure out how to implement post method for these pages and session control

**@app.route("/adminLoggedIn")**

once the employee has logged in this is the page they will land on, which gives them a range of options of data to view/edit. If I were to redo this, I would run it from the same template used to create the user login page, but as I was figuring out how to do a lot of this work for the first time as I went there were plenty of similar inefficiencies

**@app.route("/flightlist")**

SQL Query combines multiple tables to get a list of all the flights with departing and arriving airports, and seating availability, creates links that allow you to view each individual flights manifest

**@app.route("/passengerdetails")**

Creates a list of all the passengers and their relevant details, creates links to view information about each individual user and allows you to edit their profile and flight details

**@app.route("/flightdetails")**

The purpose of this route is to display the flight manifest for individual flights, this can be accessed via the flight list.

**@app.route("/userprofile")**

This is the admins view of an individual passenger and can be access via the passenger list, ideally this would have just been created from the user template with changed permissions ect, but I felt it would be easier/fast to create a new page given my time constraints. The page shows the passengers personal and flight details and allows for details to be changes and flights to be cancelled. (I ran out of time to build in features to book flights)

**@app.route("/Arrivals & Depatures/")**

Allows the user to select an airport from which to display incoming and out going flights. I ran out of time to filter flights by time, but I think it serves its function well.

**@app.route("/bookings")**

Allows the user to delete flights, but not book flights due to development time constraints. Can be access either view the user and their own profile page, or an admin going through the userprofile route

**@app.route("/register")**

Allows users to create new accounts which they can use to login with

**@app.route("/editpersonal")**

Allows users to edit their own details, is accessed from the users profile page

**@app.route("/login")**

Allows the user to either login or create an account, unfortunately this is one of the things I created first and actually populates the user page and the sql query and relevant function should be moved under the user route

**@app.route("/user")**

Creates the users personal profile page

**@app.route("/logout")**

Ends all sessions and returns user to the login page