Integrating Project (PRCO204)

Group Title: Airline System – Group 7

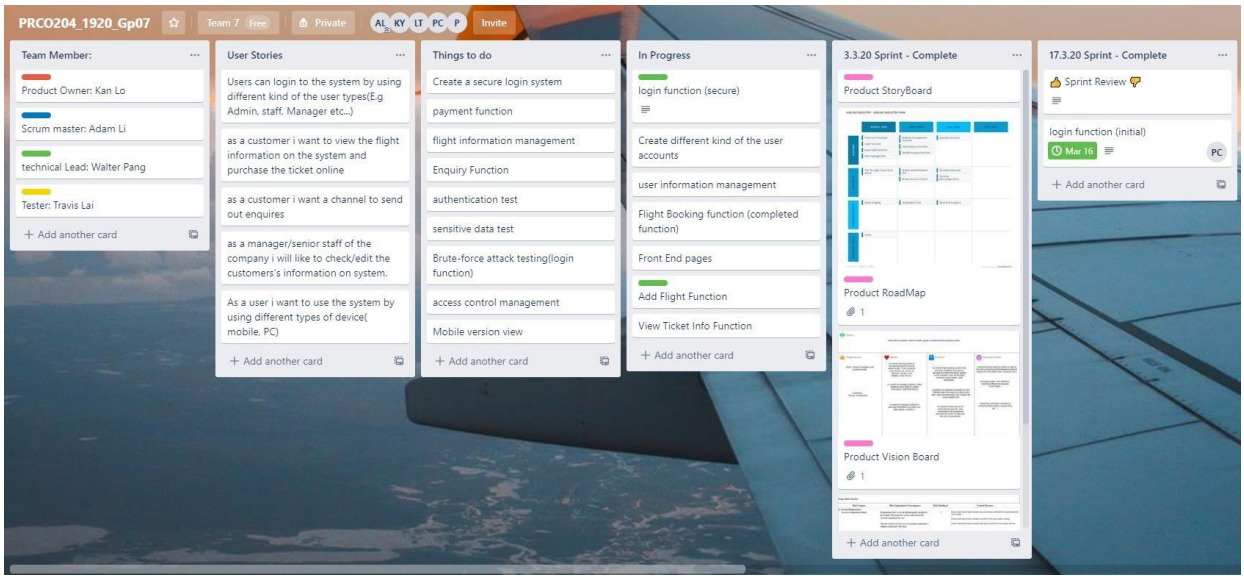
Individual Report
Lo Ho Kan

1 Task Description

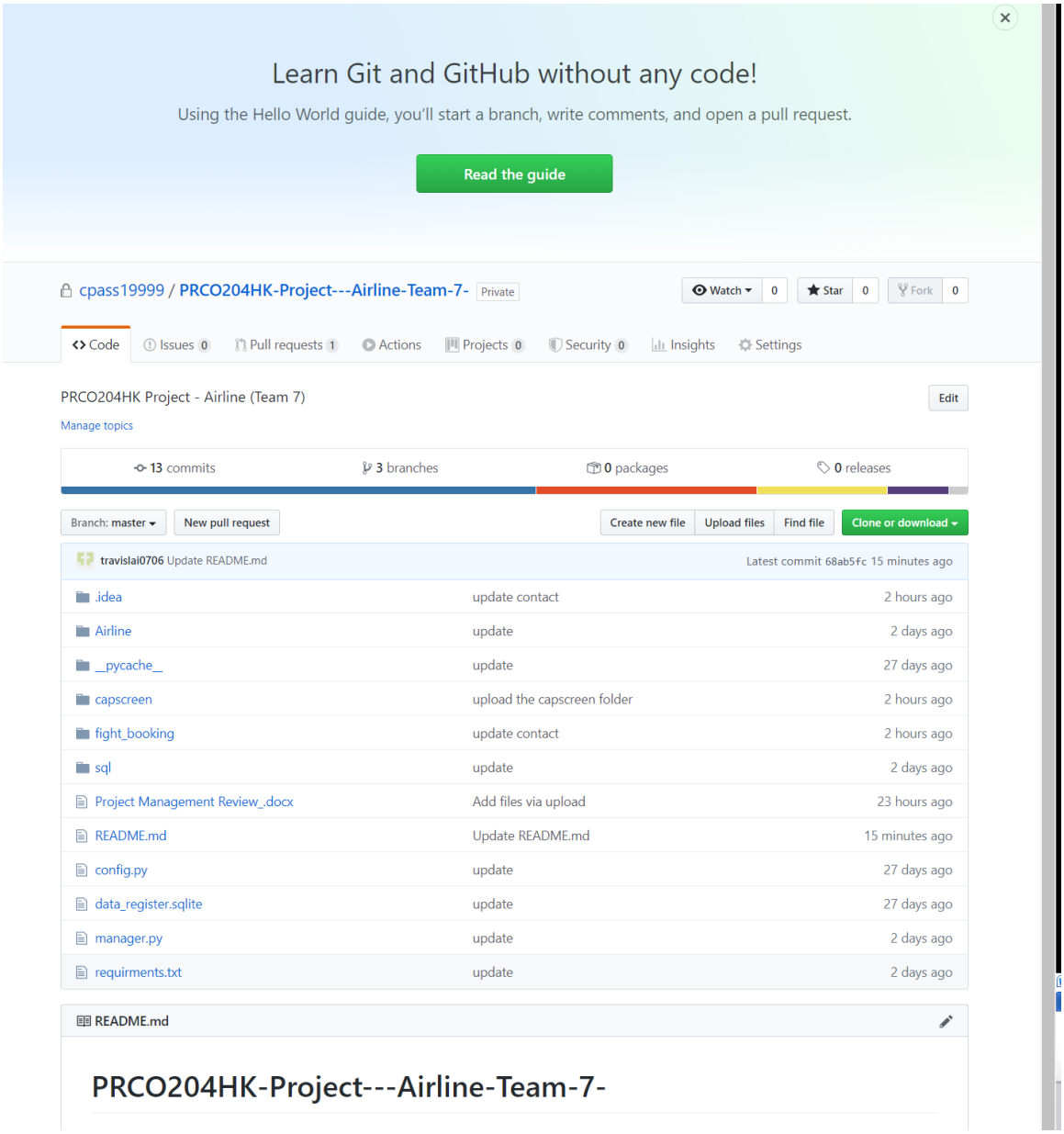
This document is a report for the individual project of this project, and the title of this project is "Airline", this project is an attempt to design and create an Airline Reservation Booking System with my team. The airline reservation Booking system is a web-based booking solution that helps in consolidating data from all airlines through the use of global distribution systems, an airline can also direct distribution works within their own reservation system, in this system will provide different function, such as "flight searching, buying airline ticket, client's information, payment's edit…etc. User account with different *role* and different rights, such as admin, and senior staff, client, so each user can be manage their data. The primary goal of this project is to help client make flight reservation. The system will be able to find all the flights available for passenger's needs and able to search all flights with a given origin and destination. The system will displayed the arrival and departure times, the flight number and passenger will decide flight reservation based on the displayed information, also after payment. And the second goal is to provide a means for the airline operators to manager the flights. The operator will be able to put up lists of passengers with their payment information and flight information.

2 Overview of the Progress

Our team have 4 members, each member have different play role, and my role in this team is Product Owner, the responsibilities of this role is to represent and communicate with customer to the development team, also to understand what the customer need so the team can be develop and for fill the requirement of the project, another responsibilities for Product Owner is to Managing the product backlog for the team, *Product Backlog* is simply a list of all things that needs to be done in the development. It can be breaks down each of the items on the list into a series of steps that helps the development team, The *Backlog* platform we have choose is call "**Trello**", it is an online application with collaboration tool that organizes your project, inside you can add image, sticky notes which note for your own and your team member to make it easy to understand how the progress for the project. After when we having conversation after meeting with customer, we list out few things to do and make a first task which to develop the web interface with few common function and scheduled the end task date on the backlog to for fill next meeting with customer, and every time there's new idea update from our team or customer, we will update the backlog so to let the progress was taken and resulted in a systematic development



Github:
GitHub is a web-based version-control and collaboration platform for software developers, we use it to store the source code for a project and track the complete history of all changes to that code, It allows developers to collaborate on a project more effectively by providing tools for managing possibly conflicting changes from multiple developers, so when we try or code completed, we will push it to GitHub to overwrite the part we have done, however it have version control in Github, and we can recovery the previous version if there somethings wrong, Our repository is set private this time, and only allow team member to access instead of set it public.



3 Challenges

Beside the play role I have in this project, I also need to help develop the program, and we have choose to use a language call "Python" to develop the web-based application for customer, either the language "python" that we use, we also have try to use C sharp at first but we change back, although we are facing many new problems in the program because it's not the language we usually use, we keep looking for the solution from online and bug fixing during the progress. The parts in this project I handle are "User" parts and "Flight Searching". In the part of "User" it have two function be need, to show User Info and User Info that can be edit, also need to develop with my team member: Travis, because one of the function he has is login function, which mean both of us share data, another is flight search, this function also the develop with other member: Adam and Walter, because they help to develop function for booking and payment.

4 Solution

The web framework we use in python is call "Flask", it is a concept of blueprints for making application components and supporting common patterns within an application or across applications. and the reason why we change to use python instead of C sharp although we are familiar with, because Flask can generate secure web content and it has consistent template tag syntax too.
In the project, we have implemented different extension for flask.

flask
flask_wtf
flask_bootstrap
flask_script
flask_bcrypt
flask_sqlalchemy
flask_migrate
flask_mail
flask_login
flask_babel
flask_markdown
flask-blueprint

The extension I need use for my User login is "**flask_login**", it can provides user session management for Flask. Also it handles the logging in, logging out, and remembering your users' sessions over extended periods of time, for the web page to communicate to the Database, I use "flask_sqlalchemy" extension, this extension is a database manipulation tool for python which can be used as standalone library to manipulate relational databases. The Database we have use is MySql, we downloaded from official site, call "MySQL Workbench", then we create a Database name call airline, in that airline, we create a schemas call airline too, the tables inside the schemas I use for my part is call "tbl_user" "tbl_flight" " airline" "airplanes""airport", for

user info part and flight searching. In most of the pages, we need form format, so we use an extension call “flask_wtf”.
It provides a simple interface with this WTForms library so we can define the form fields in our Python script and render them using an HTML template, this extension can also possible to apply validation to the WTF field, which for prevent account hacking or attacking.

First need to import Flask-Bootstrap packages, because Bootstrap into an extension that mostly consists of a blueprint named 'bootstrap', and it have a lot support to another extension, such as WTForms support for “flask_wtf”, Flask-SQLAlchemy support, also with some tamplates, then “flask_script” next, **Flask-Script** extension provides support for writing external scripts in Flask. This includes running a development server, a customised Python shell, scripts to set up the database, also we can define and add commands that can be called from the command line.

Then will be import the extension “flask_migrate” because this extension can handles SQLAlchemy database migrations for Flask applications using Alembic, The database operations are made available through the Flask command-line interface or through the Flask-Script extension

DB Connector: The service uses SQLAlchemy Object-relational Mapping (ORM) technology to communicate with the underlying service. It supports several dialects, meaning we could choose from MySQL, Oracle, PostgreSQL, SQLite and many other dialects. Also using the modularity of the framework, SQLAlchemy could be replaced with another solution for using NoSQL databases as a backend, so after we installed and run the MySQL server, we need use the ”flask_sqlalchemy” extension to communicate between the web and the Database
In this project, when user create a new account, the password will hashed while store into the database in “tbl_user” table

At the end we import “flask-blueprint”, it can help you structure your Flask application by grouping its functionality into reusable components, this can be used for have resources, such as static files, templates, and views that are associated with routes.

5 Overview

The architectural overview in Figure 1 shows both the main parts of the web application and also gives an overview of the interfaces these parts communicate through, Figure 2 shows the Database Architecture

Figure 1: Architectural overview

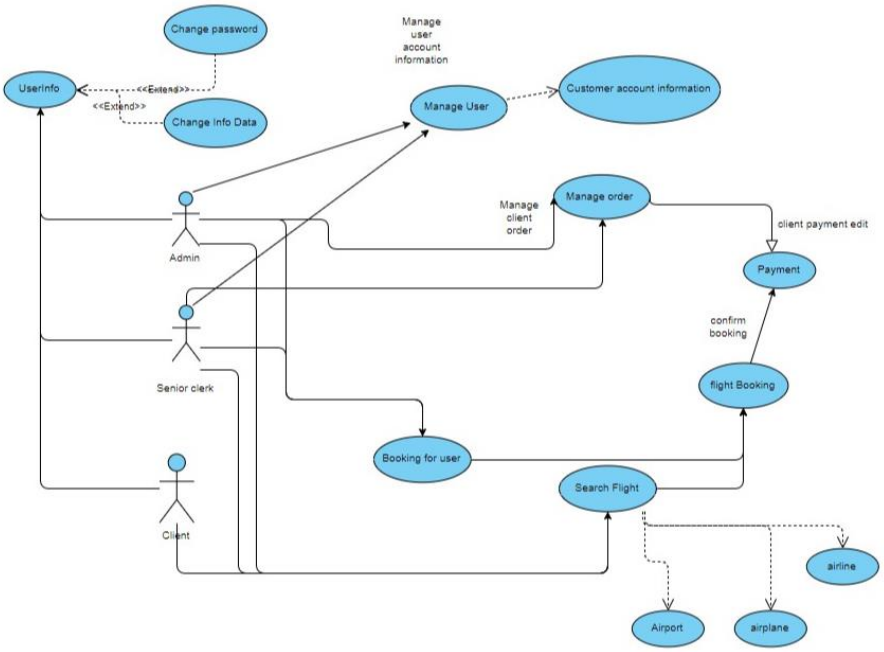
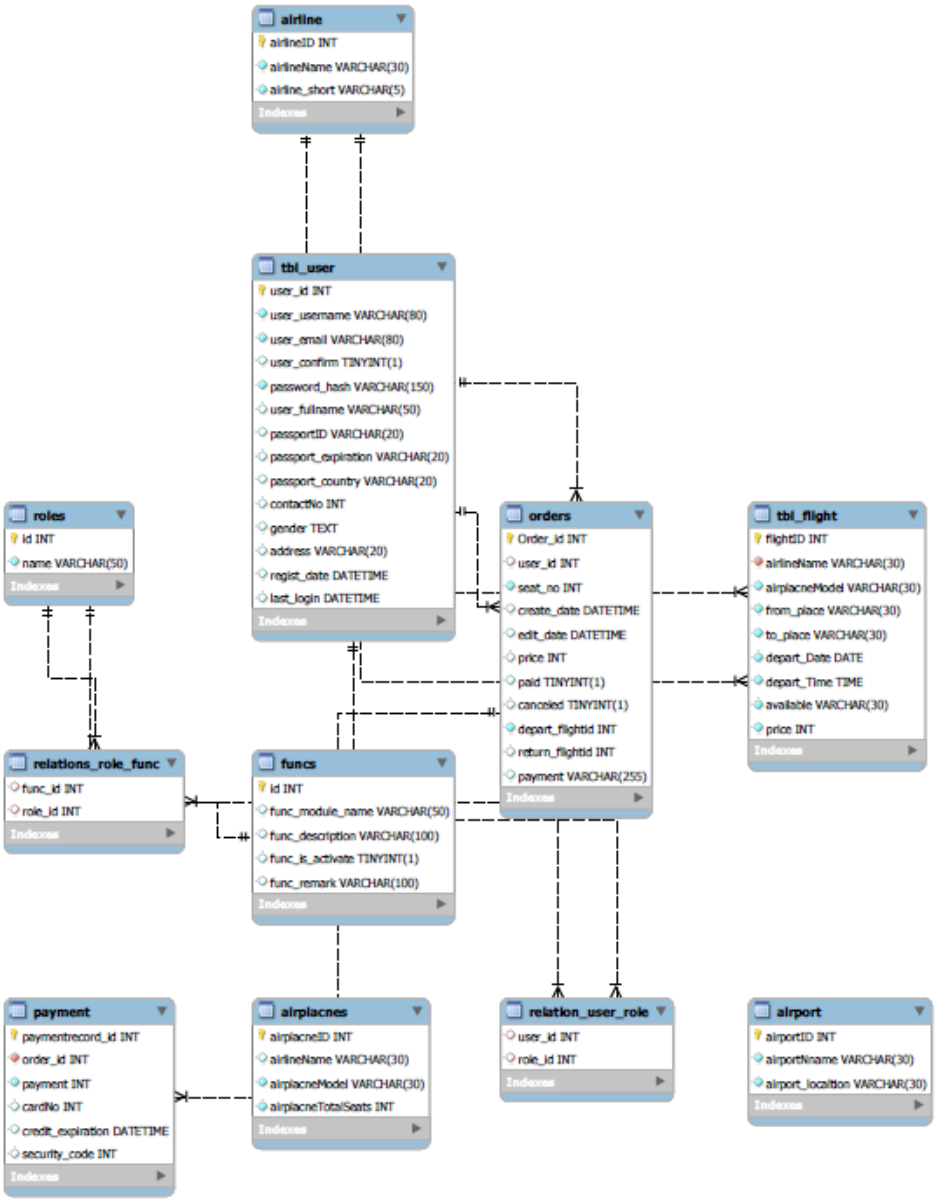


Figure 2: Database Architecture overview



Provided User info check: After user logged in, user name will be show in the webpage and different button will show by the role of current user logged-in, and if user want to their information, user can click the “UserInfo” button to check current account information, such as full name, email, and contact number, inside the “UserInfo” there’s a 2 more button, “Update personal data” which to be adding more information, like password, gender and address, “Change Password” which for user change current account password, user in this page have to type 1 more time of the old password and new password to complete the procedure.

Flight Search: Flight Searching page can be click on the menu bar where's the button" Search Flight" after logged-in, inside the flight search page, User can choose wither one-way or return for their trip traveling. Inside this page will show 4 fill-in drop box which is "from" and "To" for destination, Date for depart and return, this 4 fill-in drop box are connected to the database's table which call "airline" "airplane" and "airport", with a new request by input the correct value, it will show the available flight information: the airplane name, airplane model and airplane short-term name, once the user has select the flight, it will jump to the payment page, and in the payment page, user need to input the seat number for the flight. After everything's done which the payment have confirm, the flight information, booking date, price will all show into the "UserInfo" page.

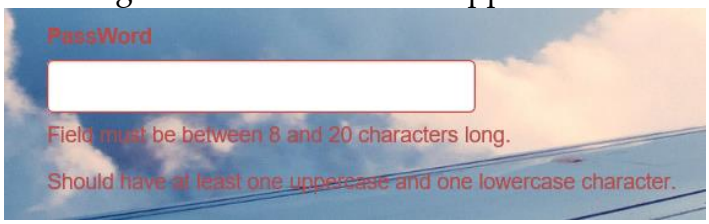
In this situation, user can not cancel the order but need other different role of user to help cancel, such as senior role and admin role, this 2 account have right to check client order, to either edit or cancel.

Booking service:

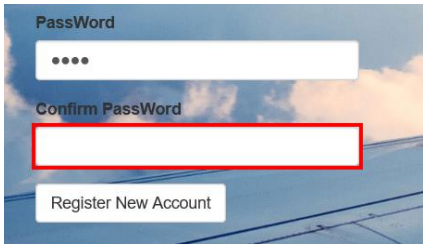
For the account which have senior role and admin role, both can help client to make a flight booking, and after logged-in either this 2 account, the menu will show the button call " Book For User", this Booking page will have 4 filled in box, and it have to input the correct data, such as User_ID, Depart_ID and Return_ID, when this 3 are correct, then press the button "add booking", the program will go search at the back of Database, matching inside the database tables call "tbl_user" for User_ID and tablet " tbl_flight" for Depart_ID and Return_ID, if it is correct, the booking data will store into a table call "orders", so the client be search the flight information by click the UserInfo button to check.

Security:

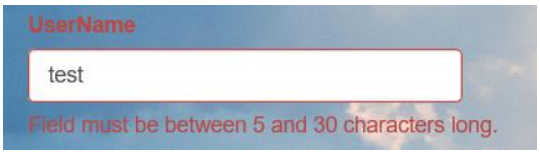
In this project, we have use some security method to prevent account hacking or attack, for create a new account, new user must insert secure password which is Strong Password Protection, user must fill in must be between 8 and 20 characters long also have at least one uppercase and one lowercase character in the field.



The second Protection Confirm Password validation, and if missed or input mismatch password, it will show red boxed field for warn.



UserName also need to be filling between 5 and 30 characters long in the field box to register a new account.



6 Summary

During the Project we have managed to design and develop an easily expandable framework and service for User to have online airline reservation booking experience. We designed it with automation in mind, minimal administrator intervention is needed.

7 Future Work

While we achieved to finish the core part of the project, which the client can be search flight and do payment, there are several extensions we would like to see in the future. The system could gain more function and more secure, like showing weather for any other location, more secure programming structure for anti-hacking or attack to our web-application or web-server, hidden Database server by splitting into few database connection.

8 Workflow Tools

To coordinate a project of this size, we needed some development and workflow tools. I used the program for python development call “PyCharm” and GitHub to collaborate, Trello for following the progress of the project, and communication and discussion with our team.

Marks allocation

Team working can cause concern amongst students. They worry that their marks will be lower if working with people who do not have the same approach to team

working than they. To account for this, students must claim their marks using the
mark claim grid shown below. This should be added as an appendix in their individual reflection. For each category, evidence must be provided such as hyperlinks to

minutes, documents and commits to the repository for the category.

Please record below the % of the category marks being claimed as justified by the evidence provided.

Marking Category	% claimed	Justification/evidence	Final mark (Please leave blank)
Process - Project Management - Objectives - V&V - Approach	25	https://github.com/cpass19999/PRCO204HK-Project---Airline-Team-7- /blob/master/Documentation/Project%20Management%20Review_.docx https://github.com/cpass19999/PRCO204HK-Project---Airline-Team-7- /blob/master/Documentation/PROC204_Group7_Group%20Report.pdf	
Product - Background - Communications - Implementation - LSEP	25	https://github.com/cpass19999/PRCO204HK-Project---Airline-Team-7- /blob/master/Documentation/minutes%20of%20the%20group%20meeting.docx https://github.com/cpass19999/PRCO204HK-Project---Airline-Team-7- /blob/master/Documentation/offical%20meeting%20minutes%20with%20the%20client.docx	