**AIRLINE TICKET RESERVATION**

1. **Introduction**

In today’s airline industry users or customers can reserve seat or book flight as long as they are connected to internet. The innovation of technology has made travelling in the air easier for customers with airline reservation system. Thousands of people flock the airline industry these days so that they can arrive their destination within a short period of time. Through this airline reservation system, customers can reserve seat, book flight and set the date if their departure and arrival using the internet. Customers can choose the place they want to go here in the Philippines. This system contributed to fundamental changes in the structure of the industry. The places available in this system are Bacolod, Batanes, Cebu, Davao, and Palawan. In the database, the flight schedule on different routes and aircraft seating capacity is stored. Along with their deposits and their accounts. The system ensures the security of the customers. The airline reservation system take pride in monitoring, administration and maintenance of the database environment using MySQLite.

1. **Objectives**

The objectives of this system are:

1. To help the customers in reserving air tickets through online system
2. To implement the airline reservation system to the best satisfaction of the customer
3. Use database MySQLite to facilitate this process

The normal process of this system is to fill the data and then process the data which used to cause a lot of inconvenience to both the administrator and customers. In using databases, the reservation of airline tickets will be facilitated. The airline reservation system can be presented in views of the different involved with it. These are the administrator and customer.

1. **Sections**
2. **Administrator Page**

The administrator is the one responsible for managing and monitoring the system. The administrator must do the following:

* Check available flights
* By choosing the displayed flight number, the administrator may now be able to view the flight details of the customer
* Check the number of reservations per flight
* Schedule of all flights
* List and status of passengers

1. **Client Page**

The customer needs to know the following information:

* Availability of flights to the destination from the departure
* Reservation status for a particular flight
* Ticket price
* Different booking counters available and their addresses
* Status if it is confirmed or still in waiting list

1. **Graphical User Interface**

**Customer Sign Up**

**Dialog**

**Flight Reservation Page**

**Main Window**

1. **Use Case Diagram**

**A picture containing text, map

Description generated with very high confidence**

For the Use Case Diagram of the system, the whole system is seen. It contains use cases and actors outside the box. The system name is Airline Ticket Reservation. As you can see, the ovals represent the system’s function. These functions are search for the available flight, find matching flight, reserve ticket, and login/sign up site. The actor of the system is the customer. The customer is the one who uses the functions except for finding the matching flight. Search engine is the one responsible to find a matching flight.

1. **Test Case Diagram**
2. **Partial Codes**
3. **Sprint 1**

* Admin.py

# -\*- coding: utf-8 -\*-

# Form implementation generated from reading ui file 'C:\Airline Reservation System\admin.ui'

#

# Created by: PyQt5 UI code generator 5.9.2

#

# WARNING! All changes made in this file will be lost!

from PyQt5 import QtCore, QtGui, QtWidgets

class Ui\_MainWindow(object):

def setupUi(self, MainWindow):

MainWindow.setObjectName("MainWindow")

MainWindow.resize(870, 667)

self.centralwidget = QtWidgets.QWidget(MainWindow)

self.centralwidget.setObjectName("centralwidget")

self.gbFlightDetails = QtWidgets.QGroupBox(self.centralwidget)

self.gbFlightDetails.setGeometry(QtCore.QRect(20, 20, 821, 621))

font = QtGui.QFont()

font.setPointSize(14)

self.gbFlightDetails.setFont(font)

self.gbFlightDetails.setObjectName("gbFlightDetails")

self.calendarWidget = QtWidgets.QCalendarWidget(self.gbFlightDetails)

self.calendarWidget.setGeometry(QtCore.QRect(70, 50, 351, 231))

1. **Sprint 2**
2. **Sprint 3**
3. **Errors Encountered**

|  |  |
| --- | --- |
| Date | Error |
| 9/20/18 | * Failed to upload the document (Gantt Chart) under the Documentation folder in the Github. The file was already reuploaded inside the folder. |
| 9/21/18 | * The title of our project did not match with the system name in the use case diagram. |

1. **Recommendation**
2. **Conclusion**