PROJECT NAME

SOFTWARE DESIGN DOCUMENT

Team 08
Shivani Tamkiya
Dipal Bhandari
Aniruddha Saxena
Rahul Prajapati

INTRODUCTION

The scheduling system should be able to keep track of the CHA crew members. The system should be able to assign the correct position and number of crew members required for each type of airplane. Furthermore, the system should be able to keep track of both the estimated and actual time of takeoff/touchdown of the airplanes. The system should also be able to log in all the updates made to the schedules and these updates should be accessible for searching based on the flight number. The system will be comprised of a smartphone application with a database backend

SYSTEM OVERVIEW

The crew manager will be able to input and remove information to facilitate the flight and corresponding crew schedules from CHA to three nearby airports: Iowa City, Iowa; Evanston, Illinois; and West Lafayette, Indiana. There are three crew positions: the Captain, First Officer, and Flight Attendant. Each flight has a Captain, a First Officer and one Flight attendant for every 50 passengers. Additionally, the crew will be able to search the schedules for information on flights and working time. The system needs to have a backup and restore capabilities. Also, the guest user will be able to search the flight and track them. There are two types of aircraft at CHA and 3 types of crew members. The scheduler needs to keep track of each flight and its takeoff and landing times as well as staff and their working hours. The software will have input for all flights then generates flight numbers. The software will be considered successful when it can have this information input, update the information, retrieve the information, and backup and restore the information.

Functional Requirements

F1. System Administrator can log in with correct credentials.

F2. Place employee to establish the initial airport for a crew member

F3. Qualify pilot or co-pilot to operate an aircraft

F4. Create flight complete with flight number, aircraft, origin & destination airports, scheduled takeoff & touchdown times, and required crew members

F5. Cancel flight, which frees crew members for other flights.

F6. Change crew member on a flight

F7. Change aircraft for a flight, which cannot be done after takeoff has been set.

F8. Change estimated takeoff time, which needs to automatically update the estimated touchdown time.

F9. Maintain updates in an electronic log that can be searched by flight, crew member, airport and/or date range.

F10. Each update in the form n-MMYYYY where n is a positive integer, MM YYYY are for month and year.

F11. Set actual takeoff time, which will set the estimated touchdown time.

F12. Change estimated touchdown time to account for in-flight delays

F13. Set actual touchdown time

F14. The crew can log in with their own credentials and place themselves.

F15. Create Airport with coordinates, location and airport code.

Nonfunctional Requirements

N1. Each flight must have sufficient staffing of qualified crew members

N2. An aircraft can't be flown from CHA if it is not located at CHA

N3. There must be 30 minutes between touching down for one flight and taking off for the next flight for each aircraft.

N4. If a flight delay causes more the time between flights to be less than 30 minutes after the aircraft has landed, either the next flight needs to be delayed to allow for 30 minutes or the aircraft needs to be changed out for a different one that has been grounded for at least 30 minutes.

N5. Flights from CHA cannot use crew that are not located at CHA

N6. Employees can only work up to 8 hours a day.

N7. Employees must have a rest period of 16 hours between work days.

N8. Each airport must have full standby crew members for each type of flight

N9. The flight that lasts longer than eight hours must have at least three pilots to allow rest break in-flight.

N10. The flight that lasts longer than eight hours must have one additional flight attendant.

The primary architecture is a client-server architecture. But instead of hosting it to a local server we're using firebase tools which changes our whole architecture to a serverless architecture. A serverless architecture eliminates the need of server and hardware management by the developer to host the application, in our case the database and authentication. Instead, it hosts it on a third party service like the firebase tools. We're using the syncing based database and authentication functionality of firebase firestore.

The system is divided into UserInterfaceSubsystem, DatabaseSubsystem, ModificationSubsystem, FetchSubsystem.

FEATURES TO BE TESTED

Testing Requirement	System Requirement(s)	Short Description
1	F1,F2,N6,N7, N9,N10	adding a crew member to an existing flight
2	F1,N1, F4,F8,N3,N2, N4	adding the flight successfully
3	F1	manager login authentication
4	F1,F3,F6	Qualifying a co-pilot to pilot and adding it to a flight.
5	F1,F6,F2	changing the crew assigned to an existing flight
6	F1	Register the manager

7	F1,F9,F10	Update logs in the format n-MMYYYY
8	F1,F6,F7	cancel the flight that is on the board.
9	F6, N5, N6. N7, N8	Assign a crew member a flight who is not eligible to work
10	F4, N1, N5, N6. N7, N8	adding flight without sufficient crew member
11	F15, F1	adding flight from lincoln to lincoln
12	F1 , N8, N5, N6, N7, N8, N9, N10	flight longer than 8 hour must have 3 pilot and additional crew.
13	F5	Cancel flight and see crew availability status.
14	F11, F12, F13, N3, N4	Change the actual take off touchdown time for a flight and then changing estimated takeoff time of the following flight.

TEST ENVIRONMENT

 The resources required for testing are -

1)Office.
2)Desk.
3)Electricity.
4)Internet Connection.
5)A computer with Android studio installed
6)Android SDK
7)Git Bash to clone the project.

TEST CASES

These test cases are created based on black-box equivalence partitions. In the test cases below, most of them cover functional requirements. Particular test cases feature the system requirements, it might be more than 1 or just 1.

TEST CASE

TEST CASE NUMBER

1

COMPONENT UNDER TEST

Test invocation of adding crew and program operation on the manager input. For this, we will use addCrew method and queries to fetch the crew from the database. Authentication subsystem and Database subsystem are under.

FEATURE(S) TO BE TESTED
F2, N6, N7, N9, N10

INITIAL CONDITIONS

In order to add the crew members, the manager should be logged-in using his/her credentials, and search for a flight in which the crew is to be added.

EXPECTED BEHAVIOR

The crew member is added to the flight as expected

1. login credential of the manager (userName = dipalbhandari@gmail.com , password = bhandari)
2. logged in to the system (Successful). Dashboard layout is presented.
3. The manager taps on flight button. Flight recyclerview list appears with a search bar.
4. The manager selects the flight and tap edit. Flight details with editable fields appear.
5. The manager adds crew(Sam) and Tap Save.

6. Flight recyclerview list is presented and changes are saved.

Test Case Number

2

Component Under Test

Test invocation of adding flight and program operation on the manager input. The subsystems that will be enabled in this are authentication, modification System and database Substem. The method used in this is addFlight in flight class.

Feature(s) to be Tested

F1,N1, F4,N3,N2

Initial Conditions

In order to add the flight, the manager should log in using his/her credentials. The manager selects flight on the dashboard and the recycler view list of flight with a search bar is presented.

Expected Behavior

1. Manager taps add flight icon from the flight activity page. The editable form to add a flight is presented.
2. Manager fills the required details of the flight. takeoff place (Lincoln)
    destination (Iowa City)
    takeoff time (10 am)
    landing time(13 pm) touchdown time is less than 30 minutes error popped up and has to input the correct take off time so touchdown is greater than 30 minutes.
3. Once the sufficient required details are filled in, the Save icon is highlighted.
4. Manager hits the save icon, the recycler view flight list is presented.

Test Case Number

3

Component Under Test

Test invocation of authentication. The subsystems which will be enabled in this are authentication, modification System and database Subsytem. The method used in authentication which checks the manager credentials.

Feature(s) to be Tested

F1

Initial Conditions

The manager should have a phone with the mobile app installed and the internet on the mobile.

Expected Behavior

The manager opens the app, enters credentials and taps the button login. The dashboard with Flight and the Crew button is presented.

Input

UserName = shivani@gmail.com
Password = Tamikya

Output

Accepted
Logged into the system successfully

4

Test invocation of qualifying a co-pilot to pilot and adding it to a flight. The subsystems which are under test under this case are - Modification subsystem, fetch subsystem and database subsystem.

F3. Qualify pilot or co-pilot to operate an aircraft
F6. Change crew member on a flight.

The manager is logged in. From the dashboard, the manager selects flight from the dashboard. Recycler view list of the flight is presented manager selects the flight he wishes to edit and the editable flight details appear.

After the user selects to add a co-pilot to the flight, it will open an activity displaying all of the co-pilots available. Manager long holds the name of the co-pilot he wants to promote to pilot and taps the option to promote. and successfully adds the co-pilot as a pilot. After saving the changes, recycler view list of the flight is presented.

1. logged into the system with valid credentials(username = dipalbhandari14@gmail.com , password = Bhandari
2. select flight in the dashboard
3. edit in particular flight and make changes

1.logged into the system
2. navigates to flight page
3. successfully added copilot

Test Case Number

5

Component Under Test

Test invocation of changing the crew member and program operation on the manager input. The subsystems which will be enabled in this is authentication, modification System and database Subsystem. The method used in this, changes crewMember.
The system under test in this category is authentication, flight and database subsystem. Manager logins with his/her credential and change the crew.

Feature(s) to be Tested

F6, F2

Intial Conditions

The manager should be logged in and should select edit flight from the dashboard and select the flight (should be edited) from the recycler view list of flight and be on the editable form of an existing flight.

Expected Behavior

In the editable form, after the required fields are filled, the manager will tap on the crew member. A drop-down list of available employees will appear. The manager will select a crew from the list and save changes. The recycler view list of flight will be presented.

Input

1.UserName = anirudhh30@gmail.com
2.Password = saxena
3. Click on the crew button
4.assign crew to particular flight (Harry to flight10)

Output

1 and 2 . logged into the system

3. navigates to crew page

4. successfully adds the crew to the flight

Test Case Number

6

Component Under Test

Test invocation of registering the manger to the system with invalid credentials and program operation on the manager input. The subsystem which will be enabled in this is authentication, modification System and database Subsystem. Register method is used in this case.

Feature(s) to be Tested

F1

Initial Conditions

None

Expected Behavior

The manager inputs invalid Gmail credentials on the signup page. The system pops upon the message to register with a valid username.

Input

username = dipalbhandarigmail.com

Output

Enter your password.

User enters password and is registered to the next page.

TEST CASE NUMBER

7

COMPONENT UNDER TEST

Test invocation of the update logs in the database for the manager to check every change made in past with the database. The format should be n-MMYYYY where n is an integer. This tests fetchingsubsystem, Databasesubsystem and userinterfacesubsystem.

FEATURE(S) TO BE TESTED

F9, F10

INITIAL CONDITIONS

In order to check the update logs, the manager needs to be logged in first into the mobile application.

EXPECTED BEHAVIOR

The user will click the log button in the manager dashboard and it will jump into the log page and gives search bar to search logs with some entry.

INPUT

1. login credentials (userName = rahul@gmail.com , password = prajapati) for manager
2. Criteria to search logs into the database

OUTPUT

1. output the changes made through query

TEST CASE NUMBER

8

COMPONENT UNDER TEST

Test invocation of canceling the flight and program operation on the manager input. The subsystems which will be enabled in this are authentication, modification System and database Subsystem. cancelFlight is the method is used in this case.

FEATURE(S) TO BE TESTED

F1,F6,F7

INITIAL CONDITIONS

The manager logins with the credential, and navigate to flight in the dashboard and cancel the particular flight.

EXPECTED BEHAVIOR

Flight should not be canceled as it is already onboard.

INPUT

1. username = dipalbhandari@gamil.com
2. password = bhandari
3. click on the flight button in the dashboard.
4. Selects a flight and tap on the edit button to cancel the particular flight.

OUTPUT

1 and 2 . valid
3. navigate to flight page
4. couldn't cancel the flight

Test Case Number

9

Component Under Test

Test invocation of adding a crew member to a flight who is not eligible to work i.e. has worked for 8 hours already. This test case will test the fetch subsystem and the modification subsystem.

Feature(s) to be Tested

F6, N5, N6. N7, N8

Initial Conditions

The manager is logged in and the recycler view flight list is open. The manager selects the flight. The flight details are presented.

Expected Behavior

Manager taps the edit flight floating button.
The system presents the editable flight details.
 The user attempts to add Varsha to the Flight Attendant.
System doesn't show Varsha's name in the list.

TEST CASE NUMBER

10

COMPONENT UNDER TEST

The invocation of adding a flight without sufficient crew member at the airport and Operation of the edit flight. This tests the Modification subsystem

FEATURE(S) TO BE TESTED

F4,N1,N5,N6,N7,N8

INITIAL CONDITIONS

The manager is logged in. The manager selects flight from the dashboard to create a new flight. The editable black form is present to fill in the flight details.

EXPECTED BEHAVIOR

The manager enters flight number, destination, takeoff, touchdown time.
The manager attempts to add crew to the flight.
The system doesn't allow to save the flight and the changes are discarded.

TEST CASE NUMBER

11

COMPONENT UNDER TEST

The invocation of adding an airport to the database with coordinates, location and airport code. This test the database subsystem, user interface subsystem and modification subsystem.

FEATURE(S) TO BE TESTED

F15

INITIAL CONDITIONS

The manager is logged in. The manager selects Airport from the dashboard and selects Add airport. The editable form of adding airport appear.

EXPECTED BEHAVIOR

The manager enters valid coordinate, location and Airport code. The system checks existing records and successfully adds the new airport if no duplicate records are found.

TEST CASE NUMBER

12

COMPONENT UNDER TEST

The invocation of adding a flight of more than 8 hrs without sufficient crew member at the airport. This tests the Database subsystem, userinterface subsystem, fetchsubsystem.

FEATURE(S) TO BE TESTED

F1, N8, N5, N6, N7, N8, N9, N10

INITIAL CONDITIONS

The manager is logged in. The manager selects flight from the dashboard and selects add flight, an editable form appears.

EXPECTED BEHAVIOR

The manager fills in the required information and selects two pilots for the flight. The application shows a third pilot as a required field. Manager taps in the field, dropdown menu shows no pilot available, neither another flight attendant. Manager taps back and comes back to recycler flight list display.

Test Case Number

13

Component Under Test

The invocation of cancelling a flight and searching for the crew member at the airport to see their availability status. This tests the Database subsystem, userinterface subsystem, fetchsubsystem and modification subsystem.

Feature(s) to be Tested

F5

Initial Conditions

The manager is logged in. The manager selects flight from the dashboard and selects the desired flight.

Expected Behavior

Manager cancels the flight. The recyclerview list of other flights is presented. The manager taps back. Dashboard is presented. Manager selects crew from the dashboard. Manager searches for the crew that were assigned to the flight that was just cancelled. The crew status is available.

Test Case Number

14

Component Under Test

The invocation of placing the crew into the flight.This tests the Database subsystem, userinterface subsystem, fetchsubsystem and modification subsystem

Feature(s) to be Tested

F14

Initial Conditions

Crew logg with thier credentials , and place themselves to flight

Expected Behavior

Crew log into the system. Then he adds his first name , last name , location for the airport to place.

Test Case Number

15

Component Under Test

The convocation of setting the actual takeoff and actual touchdown time of a flight resulting in delay for the next flight. Hence changing the estimated touchdown time of the following flight. This tests the modification subsystem and database subsystem.

Feature(s) to be Tested

F11, F12, F13, N3, N4

Initial Conditions

The manager is logged in. The manager selects flight from the dashboard and selects the desired flight. The manager taps on edit flight

Expected Behavior

Manager set the actual take off and actual touchdown time. Successfully saves the changes.The recyclerview list of other flights is presented. The manager taps on the following flight and taps edit. The manager changes the estimated takeoff. The touchdown time for the flight is calculated by itself. Manager saves changes. Dashboard is presented.