Air Traffic Controller

Submitted in partial fulfilment of the requirements for Object Oriented Modelling & Design during 7th semester of

Bachelor of Technology
in
Computer Science & Engineering

Submitted by :

01FB16ECS253 : Pradeep
01FB16ECS290 : Rahul Sharma
01FB16ECS303 : Ritabhash Das

Under the guidance of

Kiran P
Professor

2019

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100ft Ring Road, Bengaluru – 560 085, Karnataka, India

Air Traffic Controller

Requirements Specification

Introduction

The case study for air traffic controller provided actions necessary for the controller to handle in order to guide the aircraft safely into the airspace. The case includes divisions of ATC, ground control, radar controller, weather station each of which are responsible for handling separate functionality. The ATC is responsible to guiding the aircraft on land, as well as navigate the aircraft while it is the airspace. The ATC also plays crucial responsibility of providing landing and takeoff clearance to the aircraft while taking a flight. The ATC also monitors aircraft flight path and navigates aircraft on correct path to prevent probable collision.

Functional Requirements

1. Identify Aircraft:

- Adding flight: In order to add a new flight in the flight database the ATC must provide a functionality to take necessary information related to the aircraft and make a new object in the database.
- Deleting flight: In order to delete a flight in the flight database the user must search a flight with its flight number. On successful hit, the user can choose between update/delete. On choosing delete option the ATC system would delete the concerned flight from the database.
- Updating flight: In order to update a flight in the flight database the user must search a flight with its flight number. On successful hit, the user can choose between update/delete. On choosing update option the ATC system would update the concerned flight from the database.

2. Show airspace:

- The Pilot would be provided with an interface in his cockpit that would show the minimum safe amplitude from ground.
- If the height goes below the safe level the ATC would inform the pilot to set its height to a safe level.

3. Locate track:

- ATCS should issue departure instruction to pilots which contain runway designator, route to be flown, initial level etc.
- It should issue push back and/or start up instruction to aircraft.
- It should give instruction to aircraft to reach holding point of runway. Similarly, an arrival aircraft is instructed by this unit from runway to the parking gate.

4. Separation detection:

- ATCS should check for potential conflicts. For this ATC may first select the check conflict->predict conflict. Then he/she is prompted for ID's of the concerned aircrafts. The ATC may select from the list of ID's in the database. The system computes the distance between the two aircrafts and issues an alarm signal on unsafe configuration of the airspace.
- ATCS should issue an alarm signal to the ATC for potential conflict. The system would give several options to the ATC for the resolution of the conflict from which some are listed here:
  - altitude amendments
  - horizontal route amendment
  - speed changes

The ATC will select one of the options to send one of the two pilots a message update and the other pilot would be informed of the situation.

5. Display flight path:

The system should display the total flight path from start to destination mentioning altitude and GPS co-ordinates.

6. Display topology:

The cockpit would be equipped with a GPS database of the terrain to provide terrain warnings.

7. Process user command:

ATCS should be able to respond the user command.

ATCS should allow digital messages to be sent between controllers and pilots, avoiding the need to use radiotelephony.

8. Show runway queue:

The ATC would assign a sequence no. to an aircraft which is ready for departure with the help of the ATC software. The ATC would open the Departure interface of the software system. He would select assign runway option. The software would then prompt for necessary details related to the aircraft and automatically generate a sequence number. This sequence number would then be assigned to the concerned aircraft.

Non-Functional Requirements

- The system shall allow access at any time to all civil and military actors according to their defined roles.
- The system shall prevent unauthorized access to the system.
- The system shall be adaptable to changes to the Airport Authority procedures.

- The system shall accommodate the local Government policy.
- All the communications between aircraft and tower must be recorded and standardized and for this reason we need a database for our system. Here two databases will be used and will be maintained separately. They may interact with one another, but their usage and effect will be separate.
- Anyone to work on the system or interact with the system must login first with a username and password and also a thumb impression for complete security in a very delicate environment.
- The software should have good performance. In order to ensure good performance of the AATC software the hardware must be compatible with all the sub systems of the software. The system should be designed and implemented as such that it could handle errors and exceptions efficiently. In case of system failure, there must be a backup system to resume the operations without much time overhead as for the AATC system time is a major constraint.