

# CS261 Project Specification 2026

## Introduction

Dorset Software is a digital consultancy, providing software development services to clients across a range of industry sectors including finance, health, defence and engineering. Consultants are involved in all stages of software development, from analysis and design to implementation and testing, and with a range of leading technologies in cloud engineering, cyber security, data analytics and artificial intelligence.

Founded in 1987, Dorset Software is proud of having an extensive history of successful projects and consistent growth. Of the 150+ employees, over 85% are consultants who joined the company via the graduate developer intake, supported by a proven developer training programme. Growth targets include recruitment and training of a further 150 graduate developers over the next 3 years, located in UK offices in Poole, Oxford and Manchester.

If you are interested in joining Dorset Software, please click the following link to find out more:  
<https://www.dorsetsoftware.com/Careers>

## Problem Statement

With the global demand for air travel continuing to rise, it is increasingly important for airports to manage air traffic efficiently and optimise the movement of aircraft during both take-off and landing. Monitoring aircraft throughput is a key priority for airport managers, as it is key to maximising revenue and customer satisfaction whilst maintaining safety.

*A case study by the International Civil Aviation Organization (ICAO) found that the implementation of an advanced air traffic management system at Incheon International Airport in South Korea resulted in a 10% reduction in aircraft taxi times and a 5% increase in runway throughput (ICAO, 2016).*

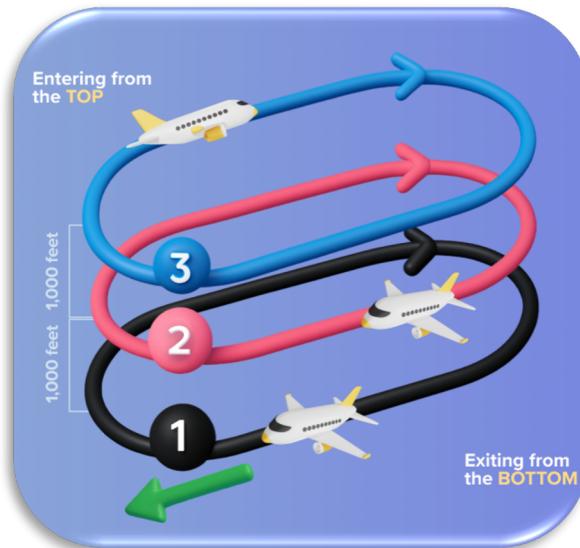
<https://startpac.com/blog/airport-operational-efficiency/>

It is essential that airport managers are able to respond effectively to situations that disrupt normal runway operations. Events such as snow, runway obstructions and equipment failure can all temporarily cease operations on runways. In these cases, the ability to model operational changes, such as converting a runway from single use (e.g. take-off only or landing only) into mixed use, is vital. This modelling helps assess the impact on delays and the prioritizations of aircraft sequencing.

# Airport Modelling Project

*This is a simulated brief, designed to be representative of a commercial software development project.*

An international airport group has requested a software solution to model aircraft travelling through their airports. This will be used to gather data on the throughput of aircraft in various runway/airport configurations and monitor scenarios such as runway closures and fuel emergencies.



*"Holding Pattern Diagram" from [Airhead](#)*

The aim of the project is to simulate the arrival and departure of aircraft at a single airport, based on several input variables and configurable parameters. The scope is limited to airplane movements and does not require any modelling of terminal activities such as baggage handling, gate assignment, or passenger movement. Helicopter movements are also outside the scope of this project.

The model needs to track aircraft arriving at and departing from the airport.

## Arrivals (Inbound)

- On entering the airport's airspace, the aircraft will land immediately if there is no other traffic to consider, otherwise it will enter a queue called a *holding pattern*<sup>1</sup> which is located near the airport.
- Aircraft in the holding pattern must maintain a minimum vertical separation for safety. Typically, aircraft will enter the holding stack at high altitude and descend as they proceed through the queue.
- When multiple aircraft are waiting to land in the holding pattern, they are selected to land in the following order:
  - Emergency – aircraft which have declared an emergency due to mechanical failure, low fuel level or passenger illness.
  - In queue order – the aircraft who joined the holding pattern first.

<sup>1</sup> [Holding \(aeronautics\) - Wikipedia](#)

## Diversions

In extreme cases where the airport is unable to provide a suitable landing slot for inbound aircraft, they will be diverted to other airports. This results in disruption for passengers and a loss of revenue for the airport and is to be avoided where possible.

## Departures (Outbound)

Unlike the holding pattern used for landings, there is no global standard for processing departures. However, for the purposes of this simulation we will assume departures take place as follows:

- Aircraft join the back of the take-off queue once they have received clearance to take-off.
- When a runway slot becomes available, the aircraft at the front of the queue will proceed to take off.

## Cancellations

In extreme cases where the airport is unable to provide a suitable take-off slot for outbound aircraft, the flights will be cancelled. This results in disruption for passengers and a loss of revenue for the airport and is to be avoided where possible.

## Runway Mode

Only a single aircraft may occupy a runway, including the immediate approach and departure zones.

Airports with multiple runways may designate runways exclusively for take-off OR landing (e.g. Heathrow), or may use a single runway in '*mixed mode*' for both take-off AND landing (e.g. Gatwick). Where a runway is used in mixed mode, runway availability must be divided between aircraft waiting to land and those waiting to take-off.

## Input Parameters

The primary input parameters for the model, including example values are:

- **Available Runways:** Between 1 and 10.
- **Inbound Flow:** 15 aircraft per hour.
- **Outbound Flow:** 15 aircraft per hour.

## Information

Data tracked in the system includes:

- **Aircraft:** Callsign (string), Operator (string), Origin, Destination, Scheduled Arrival / Departure Time from the airport, Altitude (metres above ground), Ground Speed (knots), Current Fuel Level remaining (litres), Emergency Status [None, Fuel, Mechanical Failure, Passenger Health].
- **Holding Pattern:** Aircraft in the holding pattern.
- **Take-Off Queue:** Aircraft waiting to take-off.
- **Runway:** Length (metres), Runway Number (two-digit number), Bearing (degrees), Operating Mode [Landing, Take-Off, Mixed Mode], Operational Status [Available, Runway Inspection, Snow Clearance, Equipment Failure].

*Note that aircraft that are outside the airfield boundary or stationary on the ground are not represented in the model.*

## User Controls

The user is able to configure the following:

- Runway Operational Status.
- Runway Operating Mode.

## Output

The client's primary objective is to use this software to model various airport scenarios. For a given set of input parameters, they would like to identify how the airport handles the aircraft when a single runway is closed, or when changing from dedicated take-off and landing runways to mixed-use runways.

Priorities for the software are as follows (in order of importance):

1. Model aircraft departures and take-off queue given a dedicated take-off runway:
  - a. Model variations in take-off time (see assumptions below).
  - b. Calculate the maximum number of planes in the take-off queue and the average wait time.
  - c. Calculate the maximum and average delay between scheduled arrival time and actual arrival time.
2. Model aircraft arrivals and hold queue given a dedicated landing runway:
  - a. Model variations in arrival time at the airport (see assumptions below).
  - b. Calculate the maximum number of planes in the holding pattern and the average hold time.
  - c. Calculate the maximum and average delay between scheduled arrival time and actual arrival time.
3. Model further runway configurations
  - a. Mixed-use runways (both take-off and landing).
  - b. Multiple runways in various configurations.
4. Model the impact of runway closures on the holding queue and take-off queue.
  - a. Allows operators to specify a runway closure.
  - b. Calculate the maximum number of planes that must be cancelled based on a configurable maximum wait time (default 30 minutes).
5. Model fuel usage and fuel remaining for aircraft in the holding pattern.
  - a. Calculate the maximum number of planes that must be diverted.

*The operation mode of one or more runways may be changed in response to runway closures.*

## Assumptions & Restrictions

1. **Departure Time:** It is common for aircraft to be ready for take-off slightly later or earlier than expected due to weather or delays at the airport. Assume the aircraft enter the departure queue at times normally distributed around their target departure time, with a standard deviation of 5 minutes.
2. **Arrival Time:** It is common for an aircraft to arrive at the destination airport slightly later or earlier than expected due to weather or delays in the origin airport. Assume the aircraft enter the simulation with their arrival times normally distribution around their target arrival time, with a standard deviation of 5 minutes.
3. **Vertical Aircraft Separation:** Aircraft in holding patterns must be separated vertically by at least 1000ft. This constraint removes any need to consider wake turbulence or 'jetwash'.
4. **Lateral Aircraft Separation:** When taking off and landing, there must never be more than 1 aircraft on an active runway at any moment in time. Aircraft must have left the runway zone, before another aircraft is allowed to enter the same runway.

# Optional Features

The following features are beneficial but not essential:

## **Additional Modelling Metrics**

Additional information beyond the average figures listed above. e.g. range, variance, certain percentiles etc. which explain the distribution. A formal statistical distribution is not required.

## **Real-time Model**

A real-time simulation of aircraft in the holding pattern and take-off queue, displaying key information about each aircraft.

*NB. This can be achieved in a variety of information displays (e.g. tables) without requiring the visual representation below.*

## **Visual Representation**

A graphical representation of the aircraft in the holding pattern and take-off queue.<sup>2</sup>

## **Statistical Event Modelling**

Events such as runway mechanical failure, runway inspection & aircraft emergencies are automatically generated and modelled based on likelihood.

---

<sup>2</sup> There are many ways to achieve this, but inspiration may be gained from:

<https://www.flightradar24.com>

## Constraints

To fit within the constraints of the project timescales, it is necessary to simplify the problem by imposing constraints. Software development teams are welcome to remove these constraints if they have time.

1. **Zones:** The airport can be assumed to have distinct zones which an aircraft can occupy. A single aircraft exists in EITHER a holding pattern, a take-off queue or in a runway zone. Aircraft move instantly between zones. Aircraft prior to reaching the holding pattern, after landing or after take-off do not exist in the model.
2. **Constant Speed:** There are many factors that determine the speed of an aircraft when coming into land at an airport, but we will assume every aircraft travels at a constant speed. Assume that all aircraft travel at a pre-determined fixed speed before touch-down and after initiating take-off. Aircraft descending in the holding pattern can be assumed to do so instantly.
3. **Fuel:** Aircraft entering the simulation will contain a limited amount of fuel, uniformly distributed between 20-60 minutes' worth. The aircraft must land, or be diverted elsewhere, before the fuel reaches 10 minutes' worth. The amount of fuel consumed by the aircraft will not be affected by any factors such as speed, height or weight of the aircraft. Assume all aircraft consume fuel at a constant rate.
4. **Perfect Pilots:** There is always an element of human error that can be expected in the real world. Assume pilots follow the model perfectly every time.
5. **Zero Security Threats:** While always maintaining good coding standards, this system is not considered to represent a security risk.

# Considerations

- **Quality**

The aim of this project is to create a proof of concept; therefore, it may result in a solution that is not ready for production use. However, commercial software development prioritises a useable and reliable solution which meets the key objectives, rather than feature-rich code which does not work.

- *Consider how you will reassure the client that the final version will result in a reliable, accurate solution.*
- *How long do you need to run the simulation to ensure the averages are accurate when viewing the results?*

- **Maintainability and Flexibility**

Given the considerable cost of development, most commercial software solutions are expected to provide value for many years. Systems must consider future maintainability in their design and implementation.

- *Consider what good software development practices you can use to ensure that the software will be maintainable.*

- **User Experience / User Interface**

The most technically advanced solution will be poorly received by end users if it does not consider their interactions with the system. User interfaces need to be intuitive and have appropriate and useful validation and error handling.

- *Consider how the end-user is guided to enter the input variables.*
- *Consider how you will present the output from the model to the end user. For example, does this take the form of a printable report, or a summary screen? How can the user compare multiple variations of airport configurations?*
- *What does the system do if it encounters an unexpected error or invalid input data? Is the error message user friendly, whilst still providing technical resources with information about what went wrong?*

- **The Value of Good Communication**

Neither line managers or customers typically read code and are unable to appreciate technical wizardry behind the scenes.

- *Consider how you will communicate the benefits of your solution.*

## Hints and Tips

- **Planning, Design and Review**

We are not aware of any successful software development methodology in which coding is the very first step.

- **Plan for Integration of Components**

This project has many facets, which allows multiple components to be developed in parallel. These components must be integrated to form the final simulation. It is recommended that this integration, and 'main' process logic be considered towards the start of the project.

- **Focus Your Efforts**

A fully functional and production-ready system is outside the scope of this project. Consider how to achieve maximum value from your contribution.

- **Teamwork**

A common issue which results in poorly functioning software development teams is the 'Superman Complex'<sup>3</sup>. This describes an individual within the team who believes that they are the only team member who can produce good output, intimidating other team members into marginal contributions.

*Successful software development teams depend on good teamwork!*

- **Remember the Primary Objective and Deliverables**

To create an airport simulation that achieves the functionality described in the Output section above.

---

<sup>3</sup> [https://en.wikipedia.org/wiki/Superman\\_complex](https://en.wikipedia.org/wiki/Superman_complex)