



Digital Skills Partnership

'Time to Go' Project Brief

A Smartphone App to Reduce Airport Drop-off Charges

Problem Context

19 of the UK's 30 busiest airports charge drivers for a 10-minute drop-off or collection, with fees ranging from £1 to £4.50

Entering the terminal drop off area to collect arriving passengers can be especially expensive if you arrive too early with some airports adding as much as £1/minute if you take any longer than 10 minutes.

It can be difficult to accurately predict aircraft arrival times. At Aberdeen, Glasgow, and Edinburgh on average 25% of flights are delayed.

Variable traffic conditions can significantly alter the duration of car journeys to the airport – less traffic than expected enroute can translate into an unpleasant charge.

Many drivers, upon arriving early, run down the clock before entering the terminal drop-off in nearby petrol stations, industrial parks, or long stay car parks. These areas are often also charged or are simply inconvenient, risking the ire of local businesses and causing a nuisance to non-airport users.

Project Objectives

Create a mobile app that when provided with the identification of an arriving flight will notify users when it is time to leave for the airport to collect an arriving passenger such that their wait time and thus drop off charges are minimised.

- Assume that the user will drive to the airport from their current location. Predict their correct departure time by taking into account the real-time flight status and local traffic conditions to calculate the commute duration relative to the expected flight arrival time.
- Host the service on the cloud. The service should perform the actual calculation and communicate the response to the app installed on the user phone. Implement a mechanism for authenticating and charging approved subscribers, incorporating personal and enterprise

subscription tiers. Provide a time limited free trial version for dissemination and demonstration purposes.

- Design a clean and simple user interface. For example, users should not need to know the exact flight code – simplify their interaction. Send the user a notification when it is time to leave for the airport – provide directions for the fastest route – give them a countdown clock.
- Design and implement a software verification and validation strategy. This is essential to ensure that the app will work as expected when deployed into the real-world. Consider test driven develop or other formal method of writing reliable code.
- Design and develop a business service model. Conduct a feasibility study backed by market research to maximise the commercial potential of the service.

Technical Stack

Your project team is distributed. You will need to identify and demonstrate usage of an appropriate collaboration and communication platform such as Slack, WhatsApp, G+, etc.

You will be provided with a private GitHub repository for software version control management. The programming languages are your choice and you may need to use more than one. However you must actively maintain the remote source code repository and use best practice for commit documentation, merging conflicts, branching, and versioning.

To achieve the project objectives you will need to utilise a number of web APIs which provide real-time flight tracking and real-time traffic information. You may also incorporate user communication and notification support using APIs provided by Twitter, WhatsApp, etc.

The service should be delivered using a client-server model. The client app should be installed on the user phone, supporting at least one of popular operating system. The server backend should be hosted on the cloud and expose a suitable API. For demonstration purposes a free hosting solution such as Azure can be used.

Project Management

Appoint a designated project manager as the project point of contact. The project manager will also be responsible for coordinating the implementation and timely delivery of the project.

Early in the project you will be required to deliver and present your project plan. You should define your Minimum Viable Service/Product and perform a MoSCoW analysis of its features and functions. Distil the project specification into work packages and tasks assigned to named team members and calculate the delivery schedule, using milestone and deliverables to identify the timeline of major project outcomes.

During project execution keep track of the amount of time spent on each task – use this information to estimate the project cost as well as to update the estimated project delivery timelines.

Investigate the use of Agile methods to evolve the project requirements as new market research information becomes available. As the time and budget are fixed the project delivery framework known as Dynamic Systems Development Method (DSDM) might be a suitable approach.

Commercialisation Plan

Produce a brief executive summary of the Project Commercialisation Plan (2-3 pages) covering the following key information:

What were the market research results - Are there any competing products or services?

What are your targeted customer segments? Did you receive feedback from end users – any conclusions? Define the market gap – how large is the serviceable addressable market and target addressable market?

What are the Unique Selling Points of your service app? What are the benefits of the app - focus on the user's perspective and requirements rather than the technical innovation. Why would somebody part with money to have access to the service provided by the app?

Summarise your marketing, sales, and revenue generation strategies – provide details about costs, pricing, and how you'll promote the app to potential customers – what is the user's return on investment?

What are the envisaged costs of operating the system –subscriptions to 3rd party services, hosting, and capital expenditure – do these scale with the number of users or are some costs fixed?

What is the break-even point to make the app profitable?

Deliverables

Each team will be required to deliver and present an interim project plan towards the end of month 1.

At the end of the project teams will present a two part pitch to the final judgement panel

- a technical demonstration including showcase video for promotion
- an investor slide-deck to justify the business case to commercialise and scale the app

Four deliverables are required before final pitches

- the source code repository on GitHub
- the showcase video
- the app commercialisation plan report
- the pitch slide-deck