Arbnor Aljilji

Abstract

Documented evidence of Arbnor Aljilji implementing and demonstrating the KSBs.

Dev ops engineering apprenticeship

T. Rowe Price Portfolio

Contents

[Introduction to myself, the team, and projects 2](#_Toc96438764)

[Health and Safety 4](#_Toc96438765)

[Data Protection 4](#_Toc96438766)

[Minimum KSBs met in Assessment 2: 5](#_Toc96438767)

[K3, K6, K9, K18, K19, K20, K21, K22, K23, K24, K25, S1, S2, S4, S8, S13, S16, S21, B1, B2, B4 5](#_Toc96438768)

[Project Developer Desktop Workshop 6](#_Toc96438769)

# Introduction to myself, the team, and projects

My name is Arbnor Aljilji, I’m 28 and I’m part of the DevOps Engineering Apprenticeship at T. Rowe Price. I came from a HR background interested in tech and studied BTEC IT in college. However, I graduated in BA Hons HRM. I’ve always had an interest in tech itself but wasn’t sure where exactly. I narrowed down different tech pathways and found that DevOps engineering was something that interested me, especially with the cloud ever-growing.

I’m part of the Developer Services Group (DSG) in the London team. We are a team of 9 Senior Software Engineers, 1 Product owner, and between us there are specialist skills as SDETs, SREs, and developers. Initially, the team looks after developers who create apps, and our mission is to enable the developers to move as quickly as possible and enable them to push cleaner code at great frequency, and higher volume. Through automation around development tool chain, release management, software quality, builds, and developments, we aim to achieve these goals. Unity is an in-house product designed and looked after by the DSG developers to allow other developers to deploy their application to the cloud, at a much quicker pace.

There are a few projects and tasks that I have worked on which I will mention in the portfolio, but my main KSB were met in my Unity ChatBot project as it was covered from the beginning to production environment. Other projects came from hackathon ideas such as a user-friendly front-end app for users who can look after their own VDIs. Updating deploy config files to update resources to meet developer needs. The main project is the Unity ChatBot, which currently has met its MVP state. Further features will involve being able to deploy to the cloud with a push of a message without needing to access our internal Unity Deploy app that does that for us.

T. Rowe Price week schedule differs from the usual technological diary. However, I have 2 fixed meetings are and different tasks throughout the day:

- Every Monday morning at 9.30am I have an hour meeting with my line manager to discuss my goals for this week, and what I have achieved last week. We either note it down paper, confluence or write a Jira ticket

- Every Friday morning at 10am I have a team meeting with the DSG team to discuss what support issues we have had in the Linux Developer Desktops (extra responsibility in looking after the Linux VDIs), or any other discussion such as projects etc.

- Studying AWS SAA, NodeJS, Linux, Dev Tools, and attend other meetings required or optional

The work is assigned via tickets in Service-Now or Jira and the team and I work through them. For example, if a client has an idea for a product we own, they would create a question/user request on AskTrusty (similar Stack Overflow concept) and then a ‘feature’ ticket is raised for the team of the app to implement a new feature to the application. Another way of receiving tickets is when our clients have issues with their LDD, and the DSG team supports them from 9am-2pm, in which support is then handed over to the US team. Unity Folio, which is an internal app designed to make the SDLC a smoother operation for many developers. This app allows developers to upload their app into their cloud by themselves and can be done via pipeline or the UI. The focus is to improve the automation of the procedure, to minimize the work time to upload your work and automate any procedures into deploying their apps and maintaining them. Lastly, my project sits under one main ticket ‘Unity Boy’ with sub-tickets underneath to define each task I will be working on monthly.

**Notice**: TRP doesn’t really follow the standard technological structure many people follow; we have our own way of working which works best for us. However, different cultures work differently as I will highlight throughout the portfolio. When working with the US team, I noticed we done sprints, however, in the UK I rarely needed to do sprints.

---- Give examples above -----

# Health and Safety

Signed off the health and safety training at T. Rowe Price and Makers.

# Data Protection

There are confidentiality regulations and policies I will follow when uploading work, however, due to the nature of my examples provided, there will be no confidential information uploaded.

Minimum KSBs met in Assessment 2:

## K3, K6, K9, K18, K19, K20, K21, K22, K23, K24, K25, S1, S2, S4, S8, S13, S16, S21, B1, B2, B4

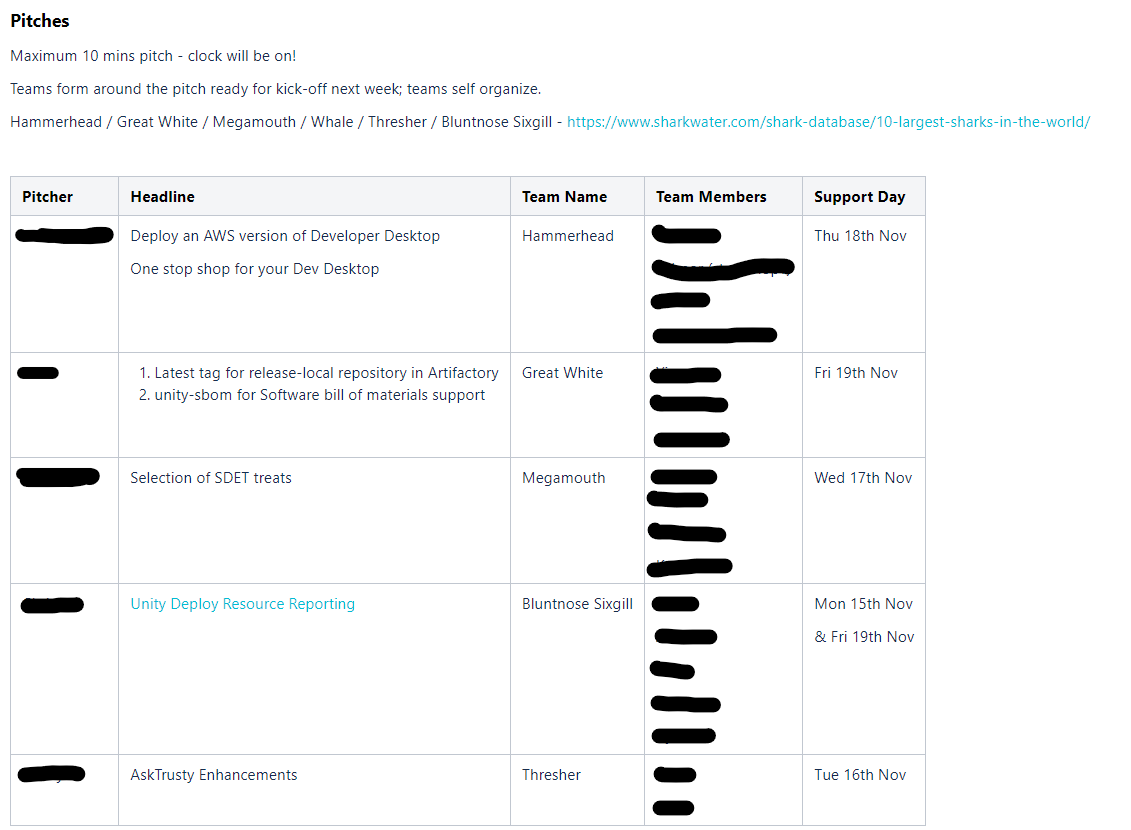
Additional KSB met in Assessment 2:

# Project Developer Desktop Workshop

**KSBs acknowledged**: K2, K10, K20, S2, S4, S13, S16, S17, S21, B1, B3

**S2** Example - **Situation**: Working with US team, different time zone and working culture compared to UK team. **Task**: Work together to make the MVP of the project. Plan it and execute. **Action:** Rescheduled my diary to fit their need, as I was the only UK based member. Shift pattern changed and worked the way they did, doing sprints etc. **Result:** I was able to work with my team more often due to working hours meeting their needs. I managed to do majority of the pages requested and complete majority of my tasks within the time frame. It was presented and teams were happy with my progress.

Within TRP, I have opportunities to work with different teams on different projects. I had the choice of picking between several projects that caught my interest. The listed projects were put on confluence and the leaders of the projects were able to present their ideas briefly on confluence.

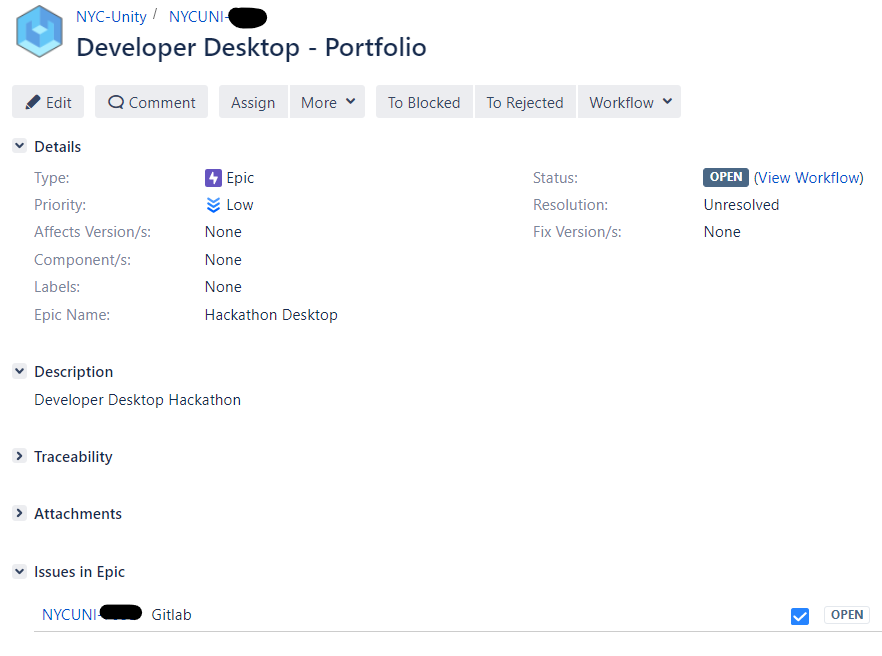


*Figure 1 – Project Overviews*

I attended the zoom meeting which the leaders were presenting the ideas and I added my name to the Team that I wanted to work in. I chose to work in the Hammerhead team as it was an interesting topic which would allow me to learn various new skills such as HTML coding.

**Time adaption (S2)**

(***S2***) The first thing I did was adapt to the changes of the team’s timing as they were in the US, so I decided to start working from 11am-8pm which meant 6am-3pm for the US team, which was enough time to work with the team, and they could then carry on with the remaining task throughout the day.



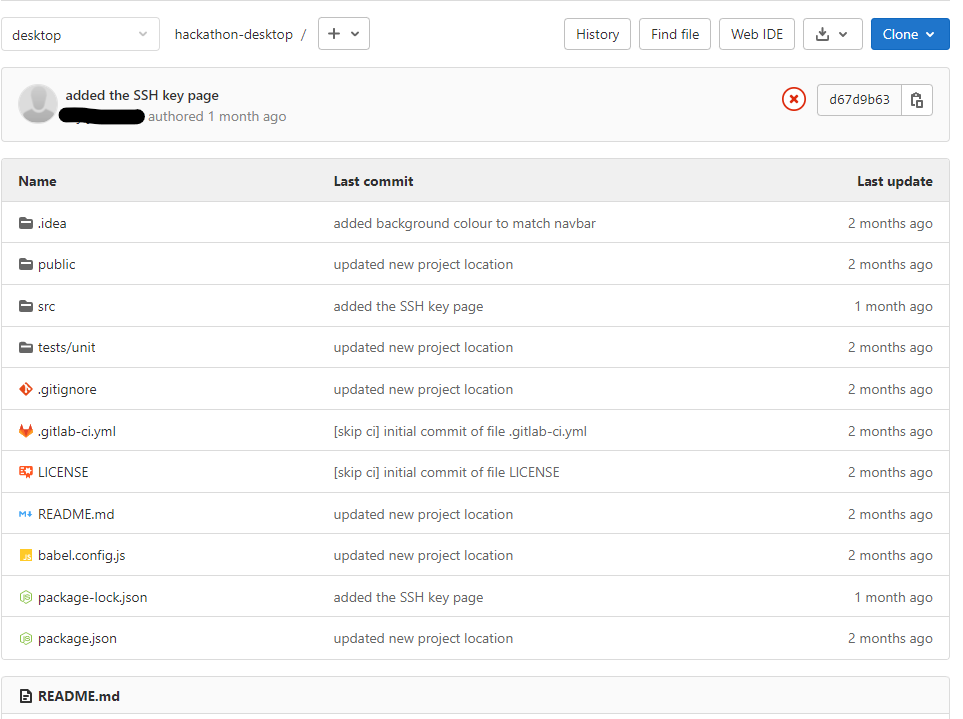
*Figure 1.1 - Jira ticket for Developer Desktop Project*

**Solo first day**

The following week of starting the project, I unfortunately had to work by myself on the first day due to the US team being on holiday on Monday. I took the initiative of setting up the project on gitlab so the team can clone the project the next day.

Above is the ticket for the project. As it was the first day of the project, I had to plan by myself how we will tackle the design of the architecture of the project. As I was not experienced in Front End development, I set up a meeting with an UX expert on what framework to use. The design of the project was that I use ReactJS as it will be a single page that renders (***S4***).

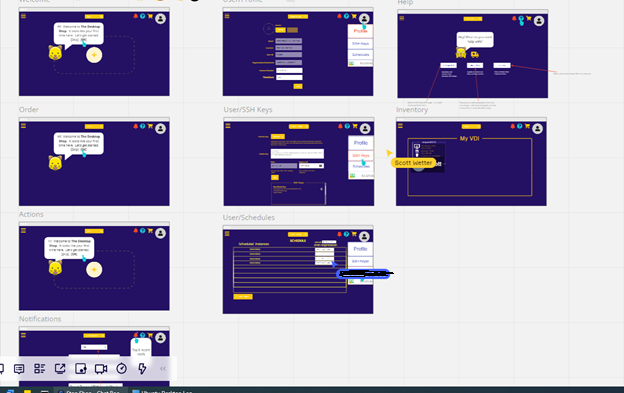
I planned the project of what framework we will use, what the project will look like (mocking), and what language to use. After the initial planning, I took the rest of the evening to learn HTML and CSS as it was my first time trying to do front-end developing. I took a quick course on codecademy and Udemy (***S16***) as it taught me the basics of front-end coding. This gave me an idea of how the elements in HTML work such as <div> etc and what framework to use (**React**). This initiative of taking lead in the project helped me set up the initial git lab repo so the team can clone it the next following day after the holiday.



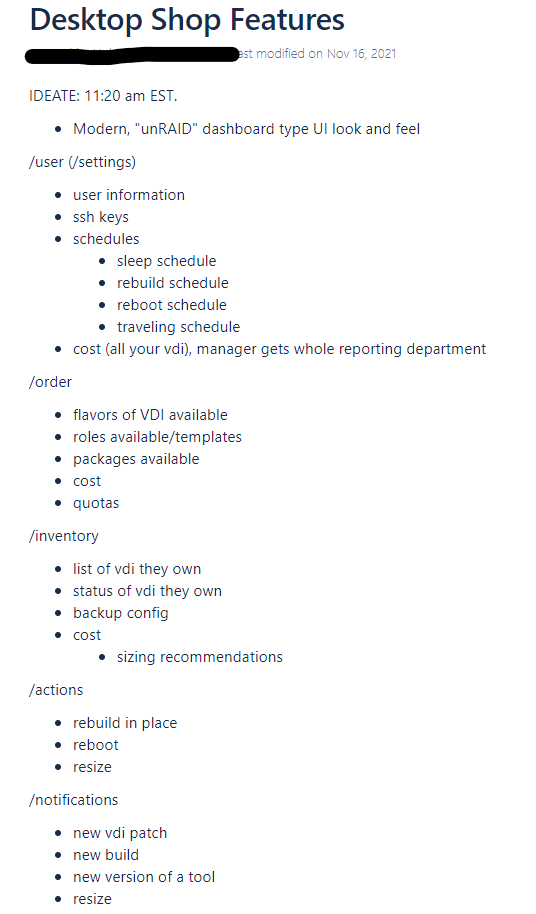
*Figure 1.2 Gitlab Repo (****K2****)*

Source control helps maintaining the project in one centralized place, where I can view the history of the project, and I could add a branch and work on my feature without disrupting my colleagues working on theirs. With small code changes, I can push for it to be reviewed and merged.

**Whiteboarding using Miro/confluence**



*Figure 1.3 Mocking up the single-page application using Miro*

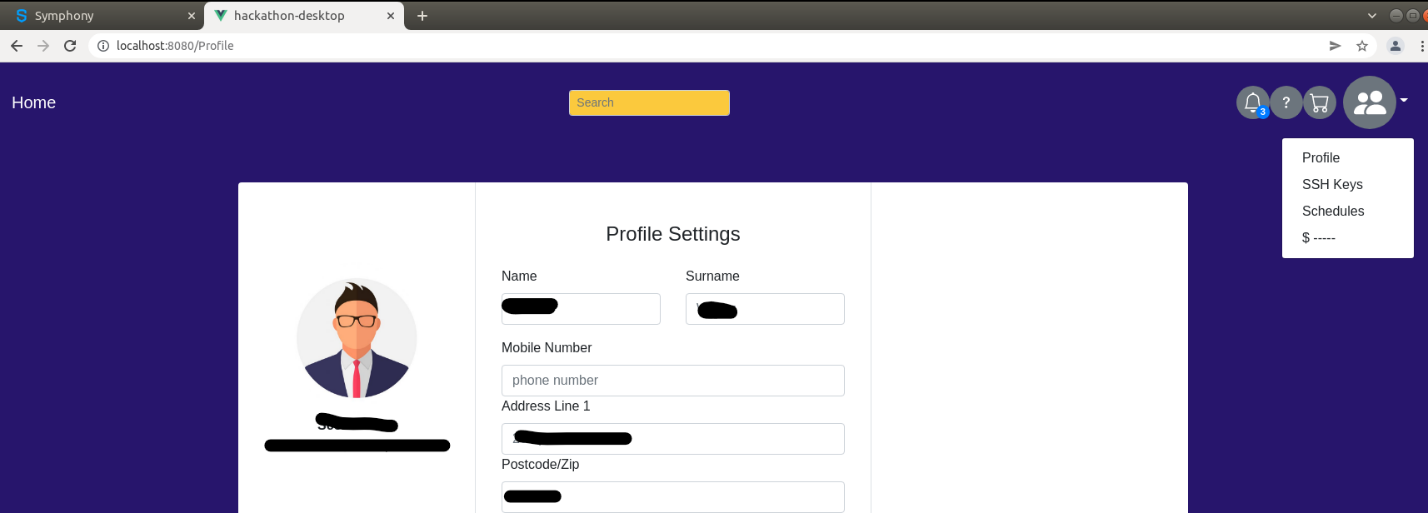


*Figure 1.4 - Project planning of the different routes and features of the project*

* Throughout the project planning on what the MVP of the product should look like and contain, the team and myself used Miro to mock-up the pages and whiteboard, and confluence for note taking (**S21**).
* Used client feedback to design the MVP and future features for the project (**K10**)

After the mock-up, it was time to code in JavaScript, HTML, and CSS (**S17**). I would sometimes pair up with a colleague and code together for the HTML/CSS (**K20**) as we were both learning, but we could also discuss the design of the page together while it’s changing. I really enjoy pair programming as we can take turn to drive while other researches, or figures out the issue (**B1**)

Below are the pages I designed using the mock-up, with slight adjustment:



*Figure 1.5 – Profile Page template*

Above is the profile page mock-up I coded using JavaScript, HTML, and CSS. The JS does the routing of the pages, while the HTML presents it and CSS styles it. As I was acting as project lead on this project, and specifically for this page, I was making sure I fixed whatever issue my code had raised. This led to continual improvement and seeing pages develop quicker, and team bonding more (**B3**). The success of this project gave us a small template to use, for the upcoming future I have with the project.