## Qualifications

BEng Aerospace Engineer, University of Hertfordshire, 2018

## Profile

My interest in the IT industry initiated during my Engineering course at university. Many of my group projects involved utilizing programming based mathematical modelling and data handling through software such as MATLAB. This included physics based models in my rocket performance and propulsion modules. The nature of my degree allowed me to become a very analytical-minded individual and also made me very keen in pursuing a job in a technology-based environment.

My individual project was focused around the Design and evaluation of a solar powered UAV.

After University I decided to t as it allows me to implement my programming knowledge and learn further.

Technical Skills

|  |  |
| --- | --- |
| **Technology Domain** | **Technologies and Tools** |
| Programming Languages | Java, HTML, CSS, JavaScript, Python |
| IDE’s | Eclipse, Visual Studios |
| Operating Systems | Windows |
| DevOps Technologies | Git |
| Database Technologies | MySQL |
| Project Frameworks | Agile Scrum |
| Other | MATLAB |

## Experience

### QA Consulting – Consultant

#### Spring Boot Ticketing System

I was part of a team that created an enterprise project from beginning to end based around a cinema ticket ordering system. The application gave the opportunity to use spring-boot technologies with a PostgreSQL database to store and retrieve data, which would then be connected to a number of different front-end technologies.

My responsibility focused more around the front end of the application. I implemented a fully designed and integrated front-end site using the VueJS framework, which I would then connect to the Spring Boot backend. This involved working with REST API’s and connecting the two layers of the architecture through Axios, a plugin designed to deal with all POST and GET requests between the server and the front-end site.

#### Learning Management System

I was part of a group project which involved building a Learning Management system using micro-service architecture. My role was to assist in building the backend which involved the use of java in the form of Spring Boot.

In addition to working on the backend I built the frontend using React JS. During the second sprint, I was part of the tech team where I reviewed the code written by the developers and made sure the code was functioning correctly, so it could be pushed on to the feature branch on GitHub and helped other team members when they needed it.

#### Individual Project

The project involved creating a back end in Java which would be linked to a ReactJS front-end. The project was hosted on an Amazon Web Service EC2 instance, allowing for public access to the application. I utilised a Mongo database to store all of the information, which would then be retrieved through a REST API and presented to the user.

## Additional Information

|  |  |  |
| --- | --- | --- |
| * First aid at work | * Manual / Heavy lifting | * Fire Prevention training |
| * Presentation skills | * Full, clean driving license | * Health and Safety at work training |

## Hobbies/Interests

In my spare time I am a keen sports enthusiast, partaking in regular games of touch rugby, basketball, football and cricket. I also have represented my university in shooting, winning the Welsh Universities Champion 2016 in the process.

I also enjoy listening, playing and writing music. I have been playing piano for the past five years and I find the creative outlet to and composing skills. I continue to learn new technical skills.

Week 4 – Java Test Driven Development, creating tests for methods. Errors, try and catch exceptions. Hash maps, Map<keySet, valueSet> items = new HashMap<>();Solid principles,

**S** - Single responsibilities: each class does one thing

**O** -Open/Close Method is open to extension but closed to modification so its reusable without having to go back to modify it

**L** –Liskov Sub Principle Child should be perfectly substitutable for the parent class. i.e flight should be implemented at the flightful level of birds who are able to fly

**I** – Interface segregation have more, smaller interfaces rather than one interface with many methods

**D** – Dependency inversion. Depend on abstractions (Abstract class/interface) not concretions (Classes that have method bodies that output something) Output should be the same type as the input

Created an object orientated treasure hunt game which used user input, error exceptions, iteration, encapsulation,