**Software Engineering Apprenticeship Level 4**

**Department for Work and Pensions**

**Alpha Mackie**

**Apprentice Software Engineer**

About Me

I am currently doing my Software Engineering Apprenticeship with the Department for Work and Pensions (DWP), in the Digital section known as the Intelligent Automation Garage (IAG). I have been placed in a SCRUM team; there are four of us in the team and we work on various automation projects as they are allocated.

Prior to starting my apprenticeship, I had completed a Master degree in Networing and Information Security with Business, got a job at the DWP as an Employment and Support Allowance Work Capability Assessment (ESA WCA) case manager. This was completely unrelated to what I studied but I saw the opportunity as a stepping stone to moving into the Digital section of the department.

Before starting my apprenticeship I had very little programming or coding experience. I had only started teaching myself how to code by learning HTML, CSS and JavaScript as I was very fascinated by the way Web pages were built and wanted to know more and build a website myself and I have a strong desire to learn new things.

About The Organisation I Work For

The IAG is involved in Robotic Process Automation (RPA), building software robots that can manipulate the execution of repetitive processes involving some existing software applications (Legacy Applications). Automating the repetitive processes and allow DWP agents to engage in much valued tasks. RPA promises an easy fast and cost-saving solution to increase the efficiency of the department.

The IAG uses Uipath Studio as a software platform of choice which provides configured tools and automation library to build a robot. We apply VB.NET and C# programming languages to develop new functions and generate reusable contents for libraries to our own benefit. We also use Uipath Orchestrator which is a web app server that coordinates the robots.

I was encouraged to undertake the Uipath online academy training as well as a one-week intensive course where I learned the IAG standards used when building software robots; the final products are usually attended and unattended robots.

My Working Environment

At the Moment I am working from home and communicate with my team via tools such as: Teams; which my team uses regularly for our daily Stand-ups as well as the overall IAG weekly communications. We use Slack for Teams wide collaborative work, sharing of documents, screening sharing discussions and for setting up various groups and Forums. We also use Skype for department wide and Civil Service wide communications and most importantly to communicate with external stakeholders.

The Methodology

Agile and a bit about how it works

The IAG uses an Agile software development methodology. This methodology is very flexible and evolving. It stems from the idea of breaking up large amounts of work into smaller pieces. This gives product managers, developers, and any stakeholder a better understanding of the work.

IAG version of the Project Lifecycle

Compared to the conventional project lifecycle, the IAG has its version the project lifecycle as follows:

* Triage
* Discovery
* Detailed Design
* Alpha
* Private Beta
* Public Beta
* Live Service

A picture containing screenshot

Description automatically generated

*Figure :. AIG version of project lifecycle*

Triage

This is the initial stage of the project lifecycle during which a business case is completed by the business and submitted through to the Digital and Automation Telephony Efficiencies (DATE) team. The DATE team then review the submission and make a decision as to whether to approve or reject it, based on the potential business value and benefits.

Discovery

In this phase the IAG team will assign a Business Analyst (BA) to hold a few discovery sessions with business to gather key information for the IAG to determine whether the project is feasible. Specific questions are answered, and site visits arranged, and a discovery deck is created to be presented to the stakeholders.

Detailed Design

This phase is only possible when the discovery deck has been signed off and the stakeholders are in agreement. During Detailed Design, the IAG delivery team (BA and Scrum team) work with subject matter experts (SME) from the business side to fully understand the business process and document a solution in the form of a low-level design to-be process using Microsoft Visio. User Stories are also one of the outputs from the Detailed Design.

Alpha

The Detailed Design is signed off, the delivery lead sets up epics, sprint planning sessions and a tester is assigned to the team.

Private Beta

In this phase the code is ready, and the business provides cases to be processed. Scenario testing is also done at this stage.

\*Please note that due to the current Covid19 situation, The IAG department has being undertaking a lot of covid related projects that demanded a qiuick turn around. Therefore these projects have not gone according to the traditional way of planning and developing software projects.

Public Beta

The software robots are scaled up to maximum capacity for testing over the period of Public Beta with live cases provided by the business. Once business have verified all test outputs, the next stage is for the IAG development team to handover the project to the Live Service team.

Live Service

In this phase is where you have the live service management team that is responsible for looking after the Robots in terms of maintenance. There is a dedicated team called the Robotics Operation Centre (ROC) team.

The Design Process and Development Standards

As mentioned earlier, Uipath Sudio is the main tool used at the IAG for all our automations. These automations have to be created, adhering to certain development standards.

When building processes in Uipath, the design is split into three processes:

1. The Load Work – this can be automating something like loading each row of data from an excel into a queue.
2. Perform Work – retrieving the above queue item and processing to completion with either a successful outcome, a business exception or as an application.
3. Reporting – this process is aimed at providing useful management Information on the work completed by the robot. The Data includes various information such as cases processed, business and application exceptions and reasons, processing time per case as well as start and end processing times.

This is reflected in the components that I am building, and each process has three levels:

1. State Machine.
2. Process Workflow.
3. Process Logic and Automation.

The three processes above are built using components and each component is made up of ‘Sequences’, which contain several Activities; ‘Activities’, which are blocks whose functions can be modified by a written code or in some cases, their default values can be used. You also have an ‘Invoke’ Activity which is used to reference other components. Figure 2a below is an ‘If Activity’ block that has another Activity called a ‘Throw’. This block is commonly found in a component when formulating conditional statements. Figure 2b is a Test harness component built for testing other components to make sure they perform the logic or automation they are built for. Within the test harness component are several ‘Activities’ that make up the components such as:

* A Sequence
* An Assign
* Message Boxes
* An Invoke

A screenshot of a cell phone

Description automatically generated

*Figure 2a. An If block Activity with a Throw Activity in it.*

A screenshot of a computer

Description automatically generated

*Figure 2b. A test harness Component with Activities such as message box, an Invoke etc.*

Below is a list of projects that I will be including in this portfolio:

1. CPS Faster Payment project:

* Data input automation
* Implementing the functionality to check for a bank holiday and the next working day
* Test harness for unit testing components of Uipath version Upgrade

1. User Interface and Database.
2. Letter of Entitlement (LOE).
3. Carers Allowance SMS.
4. Universal Credit (UC) Bundle Builder.