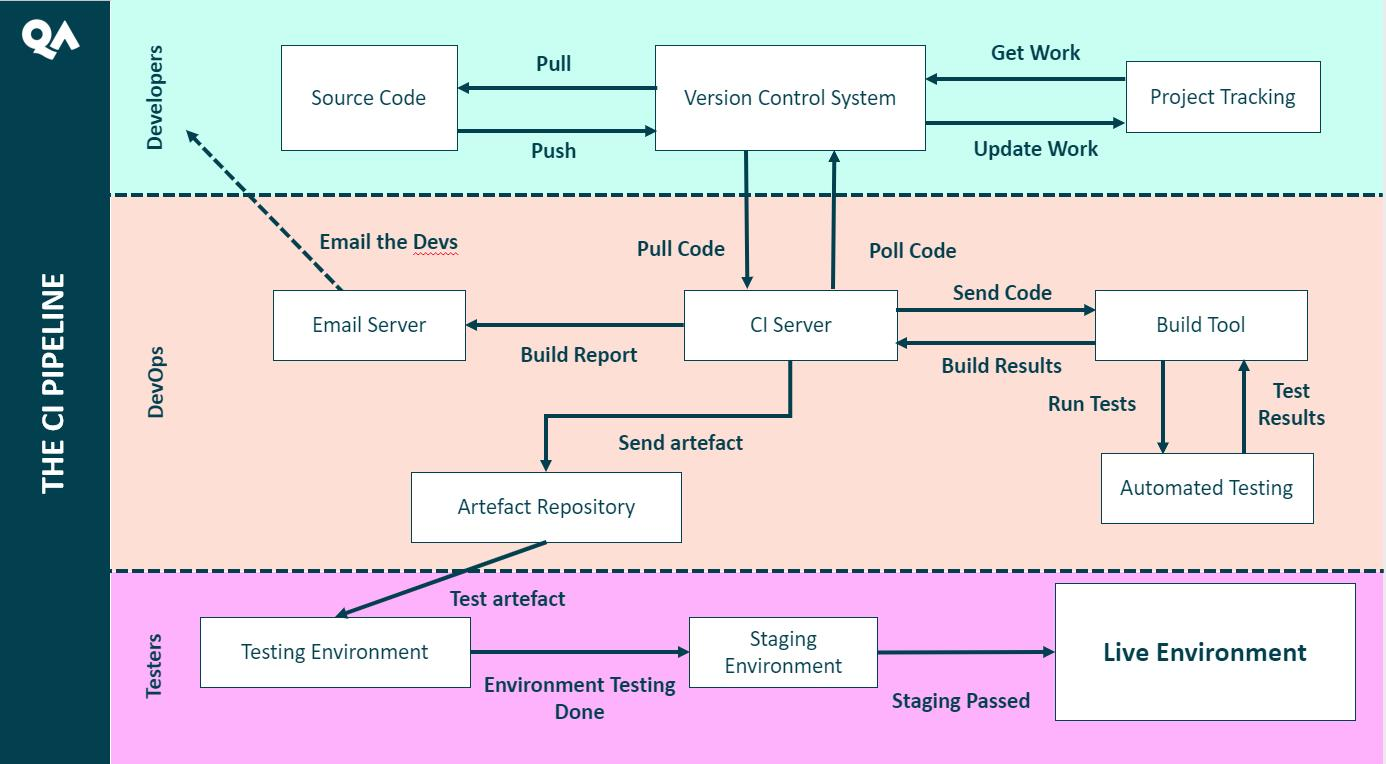
*Introduction to DevOps* Apply Activity

### Activity Outline

This activity is designed to contextualise the DevOps theory you have been learning about through the design of a continuous integration pipeline.

### Your Task



**CI Pipeline Flowchart** - *implementations of CI pipelines vary per organisation and project, but they tend to adhere to a structure like this.*

Your employer will specify a web application - real or theoretical - that they are developing. They want to implement a fully-automated system that will integrate newly-generated code into a shared code repository, allowing for multiple developers within the team to work on development at the same time, without worrying about code conflicts. The code should then be automatically built, tested, stored and run within a dedicated testing environment to allow for user-testing and stress-testing.

You will be tasked to design a **Continuous Integration** (CI) pipeline based on the specification provided to you by your employer. It should look similar to the pipeline outlined in Figure 1. You must:

* Detail each of the technologies used for the following sections of this pipeline, adhering to the requirements of the employer:
  + Project Tracking
  + Source Code
  + Version Control System
  + CI Server
  + Build Tool (if applicable)
  + Automated Testing
  + Email/Notification Server
  + Artefact Repository
  + Testing Environment
* Discuss in detail how developers will interact with this pipeline to integrate new code into the codebase
  + With particular regard to the use of Agile project planning, Project Tracking software and Version Control tools
* Develop a plan based on Agile principles
  + Prioritise your work based on MoSCoW
  + Create a product backlog using a kanban-style project planning tool such as Trello or a GitHub Project Board
  + The first two **sprints** for this project (should encompass about 2 weeks of work)
* Present your findings to your employer

The employer may or may not provide you with specific technologies or approaches to use for each section of the pipeline. Either way, you must research common tools for each section and discuss their benefits and drawbacks. If the employer has not specified a tool for a given section, you must choose one and justify your choice. e.g.: a CI server (or build server) could leverage Jenkins or CircleCI - research the benefits and detractors for both and use them to justify your final decision.

### BEFORE YOU START

Once your employer provides the details of the application they are developing and the requirements of their CI pipeline, you should prepare a series of questions to help clarify the needs of your pipeline. Consider how you will approach your employer – how do they prefer to be contacted? How should you present yourself?

For your presentation, you must consider your audience - is the client from a technical background? Do they have any familiarity with the technologies you will be discussing? There may be multiple audience members for your presentation - consider who the least technically-minded individual will be and tailor your explanations to their knowledge level (for example someone from the sales department).

Don’t be too worried about getting your timeframes right in your planning. You haven’t worked with these technologies before, so it would be unreasonable to expect you to have a perfect understanding of how much effort setting up, for example, a CI server would be! The purpose of this project is for you to illustrate what you’ve learnt about agile project planning, the technologies involved in a CI pipeline and how they fit together. Average timeframe for different sections of your pipeline might be something you could ask your client about.