DevOps - Team Project

# Introduction

The aim of the project is to give you time to explore and practice aspects of DevOps which interest you, or which you want to look into more deeply.

You can use as the basis of your work either the existing QA Quiz App, or you could write your own (probably simple) application (web-front end, API back-end or a mixture of the two). See below for a suggestion of one such application.

You'll need to think about how you intend to split the work up between members of your team, and how you'll share source code between yourselves (if necessary).

See page 2 for a list of suggested DevOps tools and practices you could try...

# Goal

Towards the end of the day, we'll re-group and each team will briefly present a summary of their work:

* What you set out to do
* What things you tried
* What worked well, or was easier than expected
* What was troublesome, or harder than expected

# Possible application: Atomic Clock

If you want to build your own application as part of your DevOps project, consider this idea...

* API server which exposes an end-point for getting the current time
* Your API actually calls another web-service to obtain the current time (e.g. http://worldtimeapi.org/)
* The benefit of having your API sit in-between the requests is that you now have control over the format of the request and the response - you can do whatever manipulation you want
* You could develop the back-end API in whatever language and framework you like (e.g. NodeJS/Express, Java/SpringBoot, C# ASP.NET, Python/Flask etc)
* Web front-end to display information about the current time (any framework, e.g. JQuery, AngularJS, Angular, React, Vue - or maybe a server-side technology such as ASP.NET MVC)
* Web front-end makes use of the back-end API you wrote, to get the time
* Website may offer additional features, such as letting the user pick the country / location they want to see the time for

# Ideas and suggestions for DevOps tools and practices to try...

* Investigate command-line interface (CLI) approaches to automating some aspects of DevOps (e.g. can you set up a new AWS Amplify deployment environment just through the command line, and not use the web UI at all?)
* Send out alerts via text or email when something goes wrong with the deployed site or API
* Investigate using GitHub Actions as a CI/CD pipeline
* Try deploying the application to a different cloud provider, e.g. Azure or Google. Can you also provision an automated CI/CD pipeline to go from source code -> live automatically?
* Can you instrument your API code so that significant events are logged somehow, and made available for analysis (e.g. how many quizzes get started but not finished, what's the average number of players per quiz, what's the average duration for a quiz, etc)
* Can you package and deploy the application using Docker containers?
* Can you write some automated web-based end-to-end tests, that use both the web front-end and API back-end together to test realistic user journeys (e.g. a quiz being hosted in one browser and played by players in two other browsers)
* Can you roll out changes to the back-end nodes using other deployment strategies, such as blue/green, rolling percentage, etc?
* Can you secure the back-end API so it can only be called from the correct web front-end URL, and not from another web site?
* Can you protect the back-end API from denial-of-service attacks?
* Could you use Jenkins with some plug-ins to automate the build and deployment (CI/CD) pipeline
* Anything else you want to try...

# Guidelines

1. Agree with your team first the direction you want to take, who's going to do what, and how you'll organise your work (e.g. Kanban / very short SCRUM cycle, etc)
2. Focus on minimum viable product - don't spend the whole time writing code but never releasing it
3. Make sure everyone in your team always has something to do, and support each other - work on the same thing together if it helps
4. If you've done some of these things before at work, try things you're less familiar with
5. Keep brief notes as you go along, to act as a reminder when you summarise your project at the end of the day