Cloud Ready Hack

Challenge Description

Challenge 3: Let’s automate the packaging

Now is the time to select tools that you think best fit your team. You can work with GitHub Actions. Since you want to automatize the continuous integration process, you want to create the pipeline that will properly build your application, test it, create container images, and push them to the Container Registry in Azure.

You’ll want to resist the urge to deploy any resources to AKS since now you are building the integration part of your continuous delivery process. However, don’t forget to include executing the unit tests in your pipeline.

Since you will now be automating the creation of container images, it is an absolute must that any change committed to the master branch undergoes a peer review and approval process. You will first have to protect your main branch (or master, depending on GitHub default naming in your organization) with branch policy in a way that it isn’t possible to commit directly to the main branch until you’ve created a pull request which needs to be approved by at least one person other than the author himself.

Note: Although contributor level of rights are enough, while creating the Azure service principle, make sure to create an owner role for the purposes of future challenges.

Note: You should complete your challenge without any intervention to code. However, your coach will ask you to do a small code change to trigger automated pipeline execution.

Definition of done:

- The Continuous Integration (CI) pipeline is created for the web application, and each of the APIs
- Unit tests are executed as part of each CI pipeline
- Make sure that when there is a code change in one of the APIs (or the web application), ONLY that pipeline gets triggered
- Show that you’ve pushed updated versions of containers to Container Registry in Microsoft Azure
- The policy which prevents any unreviewed changes in code from being committed to the main branch is still in place.
- The code change requested by the coach is processed, and the new container image is created and pushed to the container registry
- Show your coach that you can link a GitHub Issue with the associated changes in code.
- Any new code you created (pipeline YAML files) is pushed to your team’s GitHub repository

Reading materials:

GitHub Actions

- Learn GitHub Actions - GitHub Docs
- Building and testing Node.js or Python - GitHub Docs
- About protected Branches
- About code owners
- About protected branches - require linear history
- Squash and merge your pull request commits
- What is Continuous Integration (CI)?
- What is Continuous Delivery?
- GitHub Actions - Azure CLI Action
- Build and push Docker Image to Azure Container Registry using GitHub Action