## Faculdade das Ciências Exatas e da Engenharia

Mestrado em Engenharia Informática



Sistemas distribuídos

## Projeto Prático - Fluxo de trabalho de CI/CD para aplicações na nuvem com AKS e K3s

**Docentes:**

Docente: Karolina Baras

**Trabalho Realizado por:** Gonçalo de Castro - 2084515 José Sousa - 2027617 Rúben Silva - 2037517

Funchal, 5 de janeiro de 2022

# Índice

|  |  |  |
| --- | --- | --- |
| [**1 Resumo**](#_bookmark0) |  | **2** |
| [**2 Introdução**](#_bookmark1) |  | **3** |
| [**3 Descrição das atividades**](#_bookmark2)  [3.1 Fase 1](#_bookmark3) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | .4 | **4** |
| [**4 Análise de resultados**](#_bookmark4) |  | **5** |
| [**5 Conclusão e trabalho futuro**](#_bookmark5) |  | **6** |

1. **Resumo**

No âmbito da unidade curricular de Sistemas distribuídos foi-nos pro- posto a realização de um projeto usando a tecnologia Docker para publi- car uma aplicação previamente criada e construir um fluxo de trabalho de integração e entrega contínua (CI/CD, Continuous Integration/Continuous Delivery). Numa primeira fase criou-se o fluxo de trabalho de integração e entrega contínua, e numa segunda fase criou-se um cluster local.

Para a criação deste fluxo de trabalho usou-se varias tecnologias e ferramentas. Para a alteração de código, relativo ao fluxo de trabalho, utilizou-se o Visual Studio Code, para a criação de contentores utilizou-se o Docker, para o repositório de código foi utilizado o Github, tendo sido usado o Github Actions para automatizar a criação da imagem a partir do Dockerfile e posterior publicação no Azure Container Registry, tendo este sido usado em conjunto com o Azure Kubernetes Service para criar os Clusters para correr a aplicação.

Já na segunda fase foram utilizados dois raspberry pi’s com o intuito de construir um cluster local, onde um deles seria considerado o “master” e o outro o “slave”.

# Introdução

Atualmente, existem vários modelos e técnicas para desenvolver apli- cações, existindo varias dificuldades como por exemplo o controlo das dependências de uma aplicação, ou a estabilidade da aplicação indepen- dentemente da maquina em que se encontra a ser executada.

Existe no entanto uma forma que permite atualizações das aplicações sem ser necessário cancelar a sua execução na totalidade (devido à mo- dularidade), e a sua execução sem depender do sistema operativo da maquina em que se encontram a ser executadas, sendo isto possível a partir da conteinerização.

Neste projeto utiliza-se a plataforma Docker, para criar a imagem da aplicação para posteriormente criar os contentores, sendo usado em con- junção com o Kubernetes, permitindo a criação e gestão de um cluster.

# Descrição das atividades

* 1. **Fase 1**

Nesta fase foi realizado todo o processo base do fluxo de trabalho, tendo sido escolhida a aplicação de teste (aplicação Todo App, utilizada durante a aula). Após a escolha da aplicação criou-se o ficheiro do Work- flow e o ficheiro DockerFile.

1. Fase 2

Nesta fase foi realizado todo o procedimento da fase 1 onde em vez da utilização do azure foram utilizandos dois raspberry pi’s com a intuição de criar um cluster local. Para isso começamos por configurar os raspberries onde um ficaria como “slave”, com o ip 10.2.15.155, e o outro como “master”, com o ip 10.2.15.156. Após isso

# Análise de resultados

Análise dos resultados 2 a 4 Paginas

# Conclusão e trabalho futuro

Com este trabalho, colocou-se em prática e aprofundou-se os concei- tos lecionados na unidade curricular, nomeadamente os conceitos relati- vos à conteinerização, assim como os relativos à utilização do Docker, do Kubernetes, do Azure DevOps e do K3s e Raspberry Pi.

Obteve-se também o conhecimento básico relativo à utilização do Git- hub Actions que nos permitiu a automatização do fluxo de trabalho.

Relativamente ao trabalho futuro, considera-se importante a automati- zação total do fluxo de trabalho a partir do momento em que é efetuada uma alteração no repositório (assim que for realizado um push ou um pull request), de forma que o Workflow criado, realize todos os passos neces- sários tendo em conta se é necessário parar a execução da aplicação ou não para ser realizada a correção ou atualização de um ou mais modulos dessa aplicação.