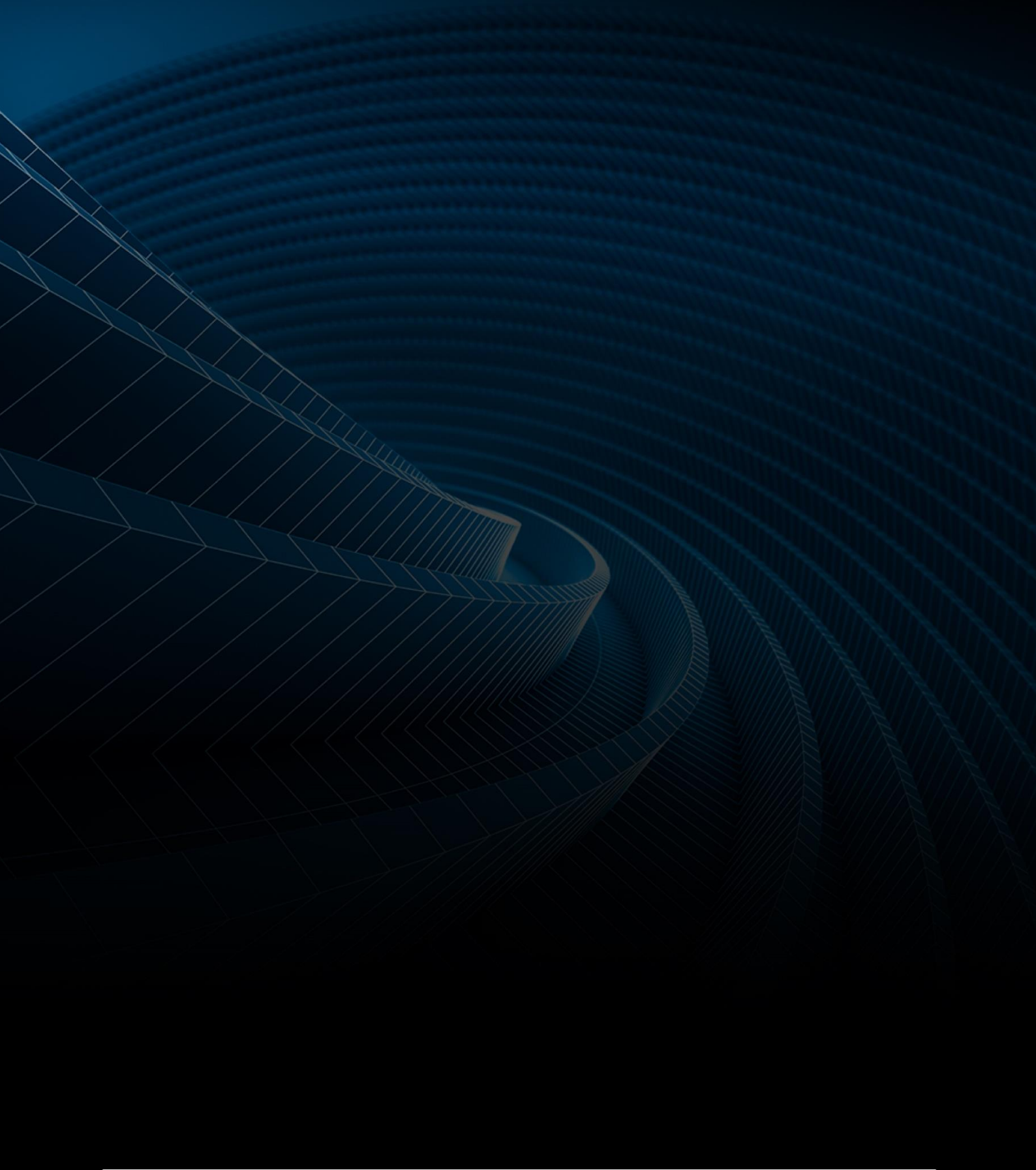


POSTECH



Tech Challenge

Tech Challenge é o projeto da fase que engloba os conhecimentos obtidos em todas as disciplinas dela. Esta é uma atividade que, a princípio, deve ser desenvolvida em grupo. É importante atentar-se ao prazo de entrega, uma vez que essa atividade é obrigatória e vale 90% da nota de todas as disciplinas da fase.

O problema

A **FIAP Cloud Games (FCG)** segue sua evolução! Após construir a base do sistema na primeira fase, precisamos garantir que a plataforma seja escalável, confiável e monitorável. Agora, nosso foco será a automação do deploy, a containerização da aplicação e o monitoramento da infraestrutura.

O desafio desta fase foi estruturado para aplicar os conhecimentos adquiridos nas disciplinas que vimos, como CI/CD, Docker, Azure DevOps, AWS e ferramentas de monitoramento.

Desafio

Agora que a FIAP conseguiu identificar o valor no MVP construído anteriormente, precisamos lançar a primeira versão para os alunos e alunas. Isso está atrelado diretamente a sua disponibilidade, escalabilidade e resiliência, já que a FGC é uma plataforma muito aguardada; então, esperamos que o acesso seja muito grande durante sua estreia e a cada lançamento de jogo ou funcionalidade.

A FIAP sabe que essa não será a versão final, mas, de qualquer forma, quer reduzir o impacto de problemas relacionados a performance, uma vez que os alunos e alunas têm pouco tempo disponível para jogar e aprender dentro da plataforma devido ao dia a dia corrido.

Pensando nisso, agora, precisamos garantir que os deploys e testes sejam feitos de forma automática e, para não limitar o acesso, devemos criar uma infraestrutura na cloud para suportar o alto número de usuários.

Funcionalidades obrigatórias

Garantir escalabilidade e resiliência da aplicação:

- Escolher uma infraestrutura que suporte o alto número de alunos e alunas.

Dockerizar a aplicação:

- Criar uma imagem Docker simples e pequena para facilitar novos deploys.

Monitorar a aplicação:

- Garantir métricas para entender possíveis problemas com falta de recurso(s) e compreender o comportamento da aplicação.

Arquitetura:

- Para essa fase, seguiremos com um monolito para facilitar o desenvolvimento ágil e focar na implementação na cloud.

Requisitos técnicos

Configurar CI/CD para automatizar a entrega do conteúdo:

- CI: pipeline que deverá ser executada na abertura de PR/Commit.
- CD: pipeline que deverá ser executada quando o merge ocorrer na branch principal. Essa branch representa a versão mais atualizada da FCG.
- **(Opcional)** Multistage: se utilizar uma pipeline Multistage, ela será considerada a união das pipelines CI/CD; neste caso, uma pipeline basta.

Dockerização:

- Criar uma Dockerfile para a elaboração de imagem do FCG relacionada à publicação na cloud.
- Enviar e armazenar uma imagem Docker em algum repositório (ex.: Dockerhub, ECR, ACR).

Publicar aplicação na cloud:

- A aplicação deve ser atualizada por meio da pipeline.
 - Fica livre a escolha da provedora de cloud:
 - AWS (Free tier ou AWS Academy).
 - Azure cloud (licença de estudante).
 - Qualquer outra cloud de preferência.

Monitoramento:

- Utilizar alguma Stack de monitoramento para coletar métricas da aplicação a fim de garantir que a infraestrutura não esteja sofrendo com alto tráfego.
 - A escolha da Stack é livre (não é necessário utilizar todas as opções a seguir, uma ou mais basta):
 - Prometheus.
 - Zabbix.
 - Datadog.
 - New relic.
 - Grafana.

Entregáveis da Fase 2

- **Vídeo** demonstrando todos os pontos com até **15 minutos**. Ele pode ser em grupo ou individual (um integrante do grupo grava ou é possível se dividir entre si e apresentar).
 - O projeto deve rodar em uma cloud de sua escolha e apresentar os requisitos anteriores.
 - Se o requisito técnico estiver com a flag **(Opcional)**, isso significa que caso ele não seja implementado não descontaremos pontos.
 - Para o requisito das pipelines, não é necessário mostrar a execução no vídeo: os checks verdes já deixam explícito seu sucesso.
 - Mostrar rapidamente a monitoração da aplicação.

Tech Challenge

- **Código-fonte** no repositório (público ou privado), incluindo:
 - APIs conforme requisitos.
 - Arquivo de Pipeline CI (testes) escrita.
 - Arquivo de Pipeline CD (deploy) escrita.
 - Arquivo Dockerfile simples escrito.
 - README.md completo com instruções de uso e objetivos.
- **Relatório de entrega (PDF ou TXT) - esse arquivo deve ser postado na data da entrega e conter:**
 - Nome do grupo.
 - Participantes e usernames no Discord.
 - Link da documentação.
 - Link do(s) repositório(s).
 - Link do vídeo salvo no Youtube ou lugar de sua preferência.

Lembramos que caso você tenha qualquer dúvida, é só nos chamar no Discord!

The background is a dark blue gradient. On the left side, there is a series of concentric circles that create a tunnel-like perspective. Overlaid on this are several layers of a grid pattern, with lines that are slightly offset from each other, giving a 3D effect. The text 'POSTECH' is centered in the middle of the image.

POSTECH