

**VITOR FERREIRA DE LIMA - 22553370**  
**BOOTCAMP**  
**PORTFOLIOHUB**

## Entrega inicial

Criei um perfil do GitHub para armazenar meus projetos e repositórios para que pudesse dividir com pessoas que possam contribuir. Para isso, foi necessário tornar o meu perfil público. Dentro do meu perfil, eu tenho 9 repositórios, sendo de jogos, e-commerce, e criei um projeto durante o desenvolvimento do trabalho de uma página web para apresentar meu currículo, entre outros projetos que eu já tinha criado através de cursos ministrados pela plataforma Alura.

## Entrega intermediaria

Dando sequência, também criei um perfil no LinkedIn, plataforma utilizada como perfil profissional de trabalho, onde foi possível que eu colocasse minhas experiências profissionais, formação acadêmicas, meus certificados de outros cursos e foi possível fazer a integração entre as duas plataformas já citadas, o Github e o Linkedin. Dentro do Github foram executadas algumas ações, como a criação dos `read.me` dos projetos e repositórios criados, executei algumas alterações dentro dos repositórios no perfil do Github dos meus colegas de sala, o chamado fork, foi necessário fazer também a interação/interação do Git+Github, para isso tive que executar alguns comandos.

Sendo eles:

- **Comandos para interagir o Git com seu Github:**

```
git config --global user.name "Seu Nome de Usuário": Faz o  
Git entender o nome da pessoa que está alterando os arquivos.  
git config --global user.email "Seue-mail de Usuário": Faz  
o Git entender o gmail da pessoa que está alterando os arquivos.  
git init.  
git branch -M main: Cria uma raiz chamada "main".
```

- **Comandos para puxar e salvar seus repositórios do Github para o VScode:**

```
git clone https://link do seu repositorio no github: Clona  
o seu repositório para o visual - code.  
git branch develop: Cria uma raiz de desenvolvimento.  
git switch develop: Troca da raiz principal para a de  
desenvolvimento.  
git add seu_arquivo.algo: Adiciona o arquivo à área de espera  
(prepara pra commit).
```

`git add .`: Faz a mesma coisa que o de cima, mas com todos os arquivos e pastas.

`git commit -m "Exemplo de texto para salvar as mudanças"`: Cria um commit com a mensagem, salvando as mudanças preparadas.

`git push origin develop`: Envia os commits da branch develop para o pré repositório remoto.

`git switch main`: Necessário trocar para (main).

`git merge develop`: Mescla as mudanças da branch develop na branch atual (main).

`git push origin main`: Envia as mudanças da branch main para o repositório remoto.

## Entrega final

Dando continuidade ao desenvolvimento do trabalho, foi implementada uma nova ferramenta chamada Gemini, criada pela Google. Utilizando-a, pôde melhorar e deixar mais profissional todos os readme dos meus projetos, pude também atualizar e deixar melhor aparente a página web que criei para divulgação do meu currículum e utilizei para organizar melhor o documento final do trabalho.

- **LinkedIn:** [www.linkedin.com/in/vitor-ferreira-de-lima-400615198](https://www.linkedin.com/in/vitor-ferreira-de-lima-400615198)
- **GitHub:** <https://github.com/vitorf1>
- **YouTube:**
  - Entrega intermediária: <https://youtu.be/fm6gCze7og>
  - Entrega final: <https://youtu.be/pT9vV3GMzCI>