

CENTRO UNIVERSITÁRIO FRASSINETTI DO RECIFE - UNIFAFIRE

Desafio de Fundamentos da Programação

Disciplina: Fundamentos da Programação

Professor: Humberto Caetano Cardoso da Silva

Alunos:

Samuel Victor de Souza Romão - 202510375 José Miguel Souza Vasconcelos Leão - 202510376 Enrico Reno Nunes da Silva - 202510322

1.0 Contexto

Neste documento, iremos relatar de forma resumidamente o processo feito para fazer o desafio(fazer um sistema de gerenciamento de restaurante) da matéria de Fundamentos da Programação do professor Humberto Caetano.

2.0 Processo Criativo

2.1 Lógica/Codificação

A parte da lógica e codificação começou com uma pequena estruturação de todo o esqueleto do sistema em si, começando a uma versão bem crua(como relatado em algumas branches) usando bibliotecas como a OS, mas encontramos dificuldades em armazenar esses dados de forma concisa e funcional, então desistimos de fazer com permanência de dados, apenas focando no funcionamento do sistema em si.

Em alguns tópicos relatados pelo professor no documento "Desafio 1.docx", encontramos mais dificuldades em fazer o código, tendo que recorrer à ajuda da IA(Copilot, DeepSeek e ChatGPT) para ajudar em algumas lógicas(não de forma exagerada), e obtemos informações muito importantes que não usamos em sala de aula, como:

ALIASING - importar módulos com nomes mais convencionais.

```
import gestao_estoque_cozinha as cozinha
import gestao_pedidos as pedidos
import gestao_pagamentos as pagamentos
```

TRY; EXCEPT - melhor gestão para validação de erros.

```
try:

opcao = int(input("Opção: "))
except ValueError:
print("Digite um número válido!")
continue
```

ESTRUTURAS DIFERENTES - para melhor experiência de interface e experiência do código.

```
input("\nPressione Enter para continuar...")
if __name__ == "__main__":
```

ESTRUTURAÇÃO DE DICIONÁRIO - melhor eficiência de dicionário.

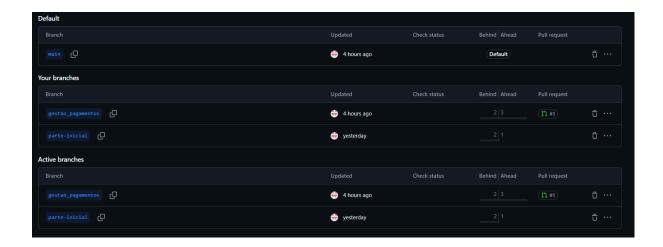
```
key=lambda item: item[1],
```

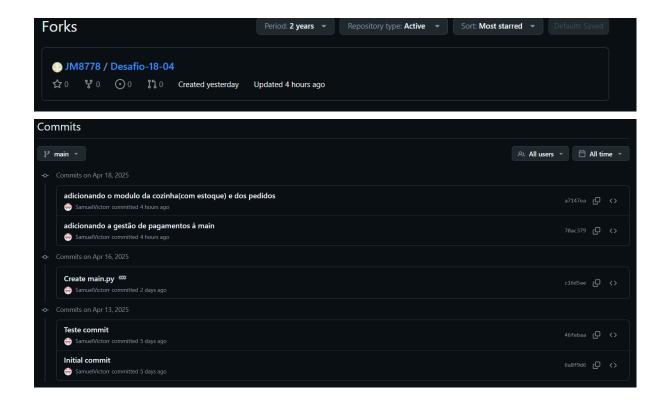
USO NÃO EXACERBADO DE BIBLIOTECAS EXTERNAS - apenas o DateTime para especificar a data de hoje para relatórios.

```
def gerar_relatorio_diario(data=None):
    if not data:
        data = datetime.now().strftime("%d/%m/%Y")
```

2.2 Versionamento

Para a utilização de versionamento, usamos o Git e GitHub, alocando quase todas as versões do sistema. Porém, também tivemos alguns problemas, como: não conseguir usar o versionamento em grupo(apenas eu dando commit ou fazendo branches, com exceção de Miguel fazendo um fork para testes). Então a organização deixou MUITO a desejar, mas fizemos o melhor que podemos.





3.0 Conclusão

Entretanto, apesar das demais dificuldades, conseguimos tirar bastante proveito dos conhecimentos adquiridos para as demais aulas e desafios futuros. Aprendizados como, estrutura modular, dicionários, tratamento de erros, versionamento e interface serão bastante importantes para o decorrer de todos que participaram desse desafio.