



Relatório do projeto de Programação II

Instituto Politécnico da Lusófona

Nº de aluno: a22100540

Aluno: Felipe Silva

Ano letivo: 2021/22

Turma: 1º ano - GSI



Índice de imagens

Figura 1 - Tabela Notion.....	5
Figura 2 - Menu importação	6
Figura 3 - Manipular objetos em listas.....	7
Figura 4 - Função classe apagar	7
Figura 5 - Servidor Base de Dados	8
Figura 6 – phpMyAdmin	8
Figura 7 - Git	9
Figura 8 - Git Terminal	9



Índice

Introdução	4
Desenvolvimento do código	5
Estrutura do projeto	6
Desafios e Dificuldades	7
Base de dados	8
Sistema de controle de versões	9
Conclusão.....	10
Agradecimentos	11
Material de apoio	11



Introdução

Neste relatório sobre o projeto de programação II, vou apresentar os métodos, as dificuldades, assim como os problemas e desafios que foram surgindo durante o desenvolvimento do código do projeto.

No manual, encontrará todas as indicações para executar e experimentar o código, sem problemas. Foi testado em “MacOS” e “Ubuntu”.

Na documentação, estão as explicações de algumas partes do código que achei conveniente e interessante de mostrar .



Desenvolvimento do código

O projeto foi realizado utilizando o IDE “PyCharm” e foi testado em “MacOS” e “Ubuntu”.

Na fase inicial do projeto, comecei por estudar e interpretar o enunciado.

Para organizar e colocar por etapas os processos do trabalho, criei uma tabela para organizar melhor.

1) Criar Disciplina	Feito.	
2) Listar Disciplinas	Feito.	
3) Eliminar Disciplina	Feito.	
4) Criar Aluno	Feito.	Implementar funções
5) Inscrever Aluno	Feito.	
6) Listar Aluno	Feito.	
7) Eliminar Aluno	revisar.	Precisa de revisão.
8) Listar Alunos inscritos numa dada disciplina	Feito.	
9) Criar Professor	Feito.	
10) Adicionar professor a uma disciplina	Feito.	
11) Importar alunos de um ficheiro	Feito.	Verificar erros.
12) lista de professores para cada disciplina	Feito.	
Testar base de dados	Feito.	
Testar em projeto em outros ambientes.		Linux Feito.
		Falta Windows

Figura 1 - Tabela Notion

Utilizei o programa “Notion” para criar uma tabela e ir anotando o que tinha de fazer e implementar.



Estrutura do projeto

Segui todas as indicações do projeto e todas as regras foram respeitadas.

Foram criadas 3 classes, para “Alunos”, “Professor” e “Disciplina”.

Para além das regras, foi implementado validações como:

- Verificar se existe disciplinas com o mesmo nome;
- Validar espaços, “Enter” e caracteres especiais;
- Validar números e letras, em caso de “string” ou “int”;
- Verificar se existe professores e alunos com o mesmo número;
- Idade entre 18 a 100 anos;

Foi adicionado mais uma opção do menu para mostrar todas as disciplinas e professores inscritos.

Como só existe um professor para cada disciplina, todas as informações são apresentadas, nome do professor ou sem professor.

Na opção de importação, foi criado um menu para importar dados de disciplinas, alunos e professores que são enviados diretamente para a base de dados. E uma segunda opção para adicionar alunos à uma disciplina a escolhida pelo o utilizador.

```
def menu_importar():#opcoes do menu de importacao
    print("\n11 - Importar alunos de um ficheiro")
    print("\n1 - Importar ficheiro (disciplinas, alunos e professores)\n")
    print("2 - Importar alunos para uma disciplina\n")
```

Figura 2 - Menu importação



Desafios e Dificuldades

O maior desafio do projeto foi manipular objetos dentro de listas e listas dentro de listas.

```
for aluno in lista_alunos:
    if nomeAluno == aluno.nome_aluno: #valida
        try:
            lista_alunos.pop(i)

            for inscrito in lista_discs: #valida
                inscrito.apagar(i)
```

Figura 3 - Manipular objetos em listas

Foi utilizado funções entre classes para manipular atributos e métodos dentro das listas.

```
def apagar(self, x): #apagar alunos de todas as disciplinas que estiverem inscritos
    for a in self.listaAlunos:
        self.listaAlunos.pop(x)
        break
    pass
```

Figura 4 - Função classe apagar



Base de dados

Para a base de dados foi utilizado o “XAMPP” e o “phpmyadmin” para verificar os valores guardados.

Foi usado “MariaDB” para a sintaxe e o tipo de servidor.

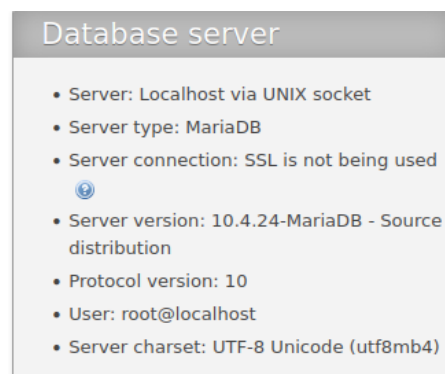


Figura 5 - Servidor Base de Dados

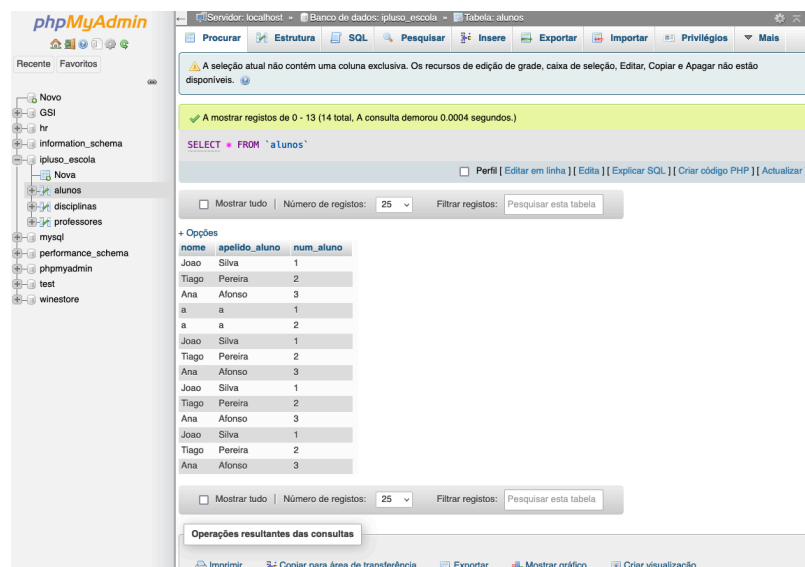


Figura 6 – phpMyAdmin



Sistema de controle de versões

Para o sistema de controle de versões foi utilizado o “Git” tanto no terminal como no próprio IDE, de forma verificar versões anteriores sempre que acontecia algum problema em outra parte do código.

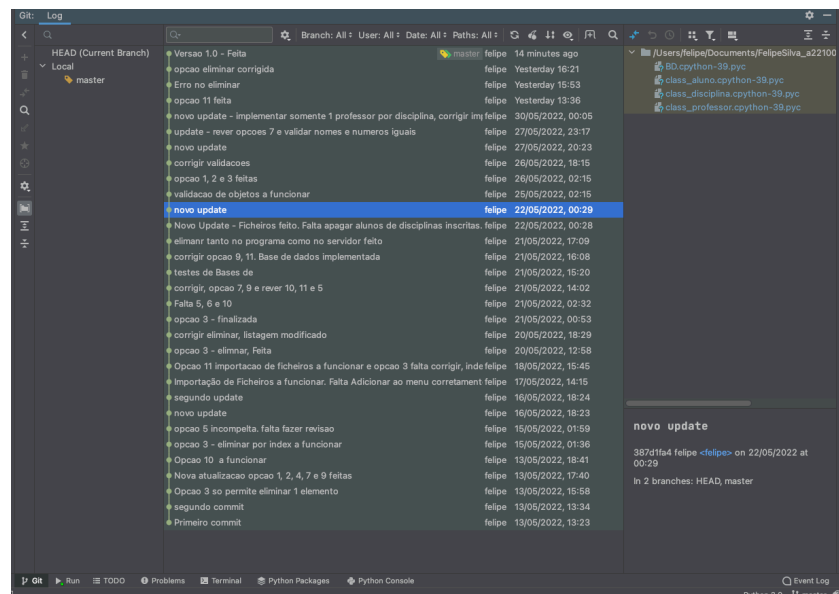


Figura 7 - Git

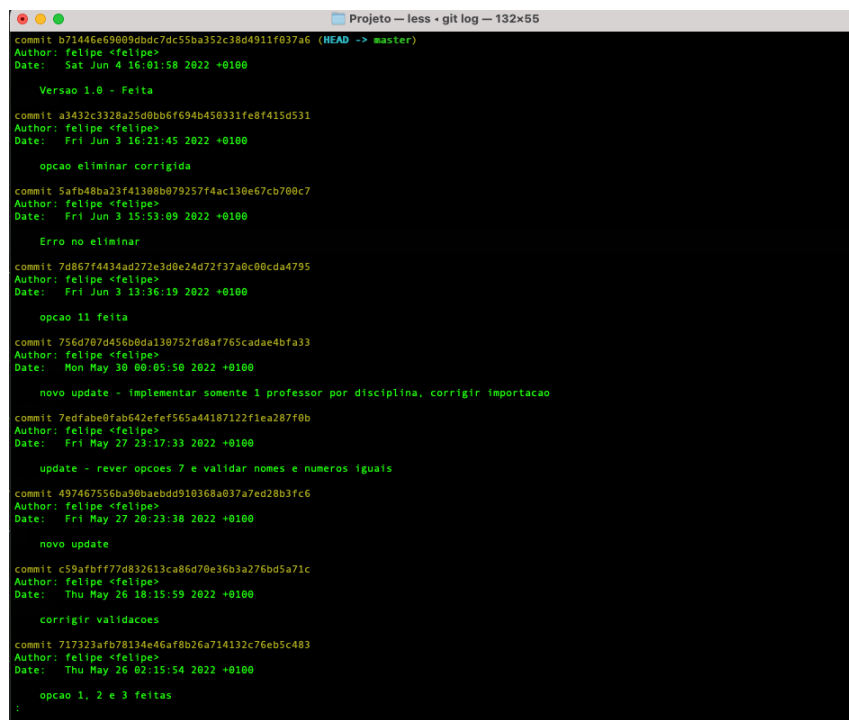


Figura 8 - Git Terminal



Conclusão

Foi um projeto muito desafiante de realizar e bastante formidável para consolidar, aprender e colocar em práticas métodos e técnicas que foram lecionadas durante o semestre.

A integração com a unidade curricular de Base de Dados facilitou a implementação, desenvolvimento do servidor base de dados e para colocar em prática um sistema de gerenciamento de sistemas de informação.

Com este projeto, fiquei com ainda mais interesse em continuar a procurar e a estudar mais, tanto “Python” como base de dados.



Agradecimentos

Agradeço aos meus colegas pelo o apoio quando necessário e ao professor pela a disponibilidade e a disposição para esclarecer dúvidas e disponibilizar todo o material de apoio e dicas para melhorar.

Material de apoio

<https://pt.stackoverflow.com/>

<https://moodle.ensinolusofona.pt/>

<https://docs.python.org/3/>

<https://mariadb.com/resources/blog/>