



Universidade do Minho

UNIVERSIDADE DO MINHO

LICENCIATURA EM CIÊNCIAS DA COMPUTAÇÃO

Guia de Utilização
NECChange

Hugo Filipe de Sá Rocha
(A96463)

Miguel Ângelo Alves de Freitas
(A91635)

Simão Quintela
(A97444)

22 de maio de 2023

Conteúdo

1	Instalação de dependências	3
1.1	Windows e macOS	3
1.1.1	Python	3
1.1.2	Poetry	3
1.1.3	Node.js	3
1.2	Linux	3
2	Setup inicial do programa	5
3	Execução do programa	6
3.1	Ativar o Ambiente Virtual	6
3.1.1	Windows	6
3.1.2	Linux	7
3.2	Executar o programa	7
4	Carregamento de ficheiros	8
5	Visualização dos horários	9
6	Troca de turnos	10

Capítulo 1

Instalação de dependências

Antes de usar o programa pela primeira vez, devem ser instaladas as dependências necessárias para o bom funcionamento do mesmo.

Dependências necessárias:

- Python
- Poetry
- Node.js

1.1 Windows e macOS

1.1.1 Python

Python foi a linguagem escolhida para o back-end do programa.

Para a instalação do Python pode ser consultada a documentação oficial da linguagem.

1.1.2 Poetry

Esta ferramenta é usada para gestão de dependências em Python de forma a que as mesmas fiquem encapsuladas no seu próprio ambiente para que o utilizador tenha mais facilidade ao dar setup ao programa.

Para a instalação do Poetry pode ser consultada a documentação oficial da ferramenta.

1.1.3 Node.js

O Node é uma ferramenta open-source que nos permite executar código em Javascript.

Para a instalação do Node.js pode ser consultada a documentação oficial da ferramenta

1.2 Linux

. Em Linux a instalação destas ferramentas tornam-se imediatas, pelo qual vão ser listados os comandos a executar para diferentes package managers.

```
Pacman
$ sudo pacman -Sy python-pip
$ pip install poetry
$ sudo pacman -S nodejs npm
```

Apt

```
$ sudo apt install python3  
$ pip install poetry  
$ sudo apt install nodejs
```

Capítulo 2

Setup inicial do programa

Considerando que todas as ferramentas estão instaladas, para correr o programa é preciso fazer um setup inicial ao mesmo. Os passos a seguir são todos **executados no terminal** e são os seguintes:

- 1- Navegar até à diretoria Projeto/schedule
- 2- Inserir o comando poetry install
- 3- Navegar até à diretoria Projeto/web
- 4- Inserir o comando npm install

Após estes 4 passos o setup fica concluído e não precisa de ser repetido em futuras utilizações.

Capítulo 3

Execução do programa

Neste projeto há duas formas de execução do programa.

1- Executar diretamente no back-end com geração de horários e acesso a um menu de estatísticas.

2- Executar o front-end onde pode ser feito o upload de ficheiros, geração, visualização e alteração de horários.

3.1 Ativar o Ambiente Virtual

Em cada um dos casos é preciso fazer um passo de importância extrema, **ativar o ambiente virtual**.

Nota: Os seguintes passos, tanto em **Windows** como em **Linux**, apenas são necessários na primeira vez, desde que se guarde o comando a executar visto que será sempre o mesmo.

3.1.1 Windows

Para ativar o ambiente virtual em Windows temos de seguir os seguintes passos:

1- Navegar até à diretoria Projeto/schedule

2- Executar o comando `poetry env info`

O output esperado após este comando (varia dependendo da máquina e do sistema operativo) deve ser parecido com o seguinte:

```
Virtualenv
Python:      3.11.1
Implementation: CPython
Path:        C:\Users\Utilizador\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_0
Executable:  C:\Users\Utilizador\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_0
Valid:       True
System
Platform:    win32
OS:          nt
Python:      3.11.1
Path:        C:\Users\Utilizador\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFoundation.Pyt
Executable:  python
```

Copiamos agora o path que aparece na 5ª linha "Executable" e substituímos o último programa "python.exe" por "Activate".

Após a substituição, cola esse novo path no terminal, pressiona Enter e o ambiente virtual é ativado.

3.1.2 Linux

Para ativar o ambiente virtual em Linux temos de seguir os seguintes passos:

1- Navegar até à diretoria Projeto/schedule

2- Executar o comando poetry env info

O output esperado após este comando (varia dependendo da máquina e do sistema operativo) deve ser parecido com o seguinte:

```
Virtualenv
Python:      3.10.10
Implementation: CPython
Path:        /home/{nome_do_utilizador}/.cache/pypoetry/virtualenvs/schedule-UP-QV2il-py3.10
Valid:       True
System
Platform:   linux
OS:         posix
Python:      /usr
```

Copiamos agora o path que aparece na 4ª linha "Path" e executa-se no terminal o seguinte comando:
source pathQueFoiCopiado/bin/activate

3.2 Executar o programa

Com o ambiente virtual ativo estamos em condições de correr o nosso programa:

Caso 1 - Executar diretamente o Back-end

1- Navegar até à diretoria Projeto/schedule/schedule

2- Executar o comando python main.py

Caso 2 - Executar o Front-end

1- Navegar até à diretoria Projeto/web

2- Executar o comando npm run dev

3- Navegar no browser até <http://localhost:3000>

Capítulo 4

Carregamento de ficheiros

Já na página do servidor local, no lado esquerdo encontra-se a secção **Upload**. Aqui, devem ser carregados três ficheiros no formato **.csv**.

- O primeiro ficheiro deve conter a informação do horário de cada Unidade Curricular (UC) do curso.
- O segundo ficheiro deve conter a informação sobre a inscrição dos alunos nas respetivas UCs.
- O terceiro ficheiro deve conter informação sobre a capacidade de cada sala.

Após carregar os ficheiros, ao clicar no botão **Upload**, estes são carregados para o servidor. A aplicação fornece feedback ao utilizador: em caso de sucesso, uma notificação verde com a mensagem "Upload successful!" é apresentada; em caso de erro, uma notificação vermelha é exibida com a respetiva mensagem de erro.

Por fim, o utilizador pode solicitar a geração do novo horário ao clicar no botão **Generate**. Este botão inicia a execução do algoritmo que processa os dados carregados e gera o novo horário. Similarmente ao processo de upload, o utilizador é informado do sucesso ou falha da operação através de uma notificação.

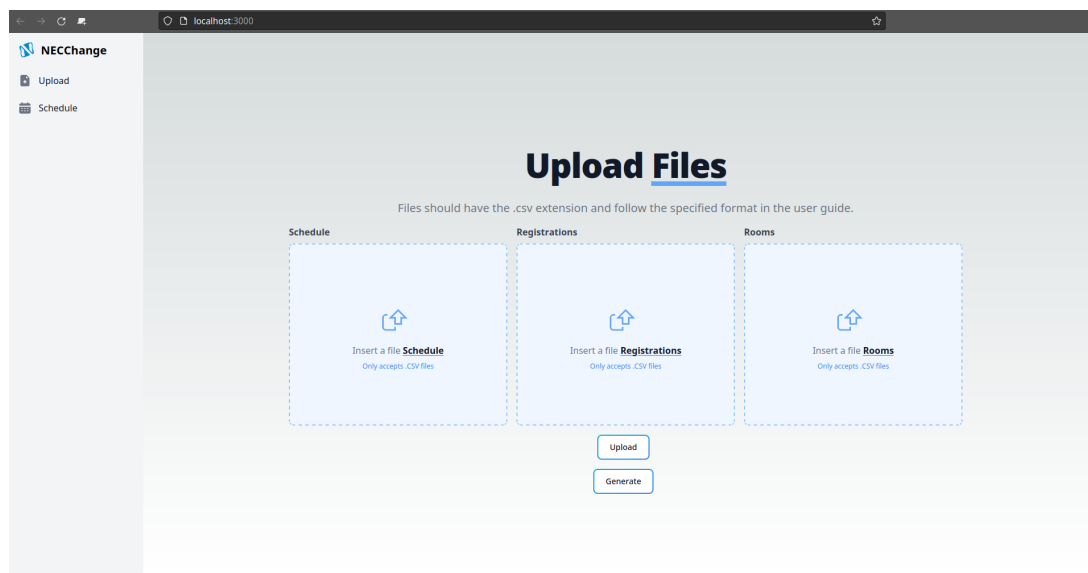


Figura 4.1: Página de upload

Capítulo 5

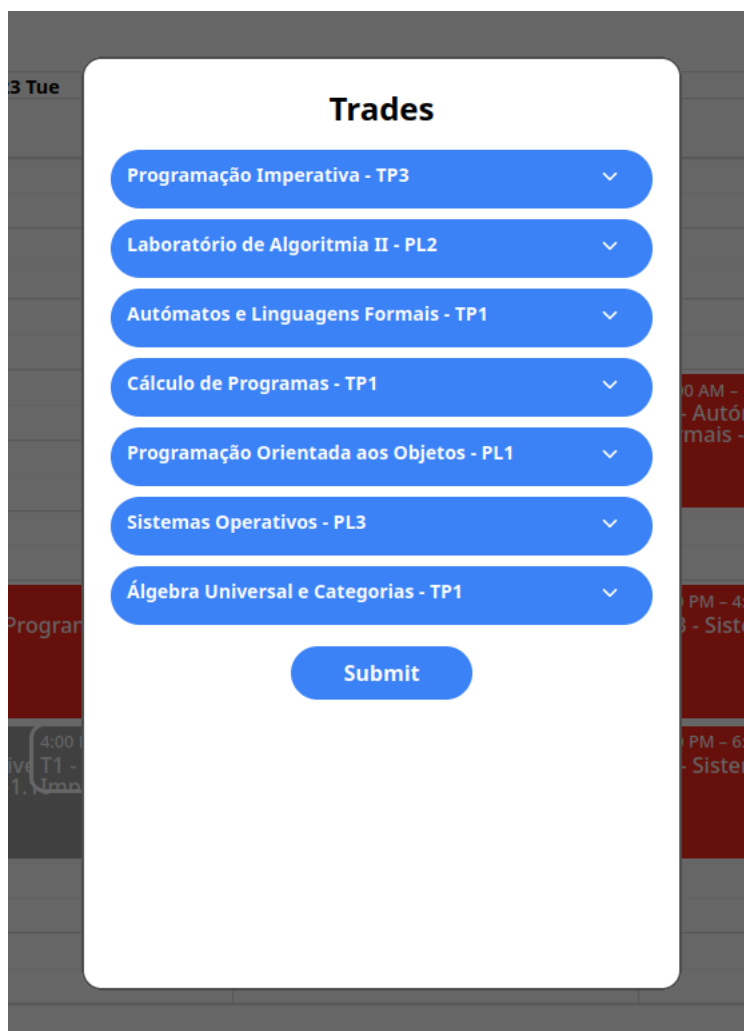
Visualização dos horários

Para visualizar os horários gerados, clica-se na separador **Schedule**, do lado esquerdo, onde podemos consultar o horário gerado para cada aluno, colocando o respetivo número mecanográfico. Após isso, apenas é necessário clicar em **Search** e o horário do aluno será disponibilizado graficamente. Para melhor compreensão do horário, foi colocada uma cor acizentada nos slots onde existem sobreposições de aulas. Além disso, as cadeiras de diferentes anos possuem diferentes cores, nomeadamente, às unidades curriculares do primeiro ano ficou associada a cor azul, às de segundo a cor vermelha e, por fim, as disciplinas do último ano ficaram com a cor verde.

Capítulo 6

Troca de turnos

A troca de turnos para um dado aluno deve ser feita também na secção **Schedule**, do lado esquerdo. Para um dado aluno, ao clicarmos no botão **Trades** conseguimos obter todas as unidades curriculares onde o aluno está inscrito, podendo alterar o seu turno diretamente, clicando no botão **Submit**.



The image shows a screenshot of a web application interface. At the top, there is a header with the text "3 Tue". Below this, a modal window titled "Trades" is displayed. Inside the modal, there is a list of seven units, each in a blue rounded rectangle with a dropdown arrow on the right:

- Programação Imperativa - TP3
- Laboratório de Algoritmia II - PL2
- Autómatos e Linguagens Formais - TP1
- Cálculo de Programas - TP1
- Programação Orientada aos Objetos - PL1
- Sistemas Operativos - PL3
- Álgebra Universal e Categorías - TP1

Below the list is a blue rounded rectangle button labeled "Submit". The background of the page is a dark grey grid pattern.