



# 1.º Projeto de Algoritmos e Estruturas de Dados

Grupo 63

Bruno Oliveira - up202208700

João Mendes – up202208783

Diogo Vieira - up202208723

# Classes Utilizadas

- Lesson :

Para guardar informações de uma aula a classe Lesson foi utilizada. Esta guarda informações sobre o início, fim, tipo e dia da semana de cada aula. De facto, podemos também verificar se duas aulas se interseitam.

- UClass :

Esta classe representa uma turma de uma determinada unidade curricular. Para além disso, oferece informação sobre as aulas da turma, o seu código, a sua unidade curricular correspondente e também detalhes sobre a mesma.

# Classes Utilizadas

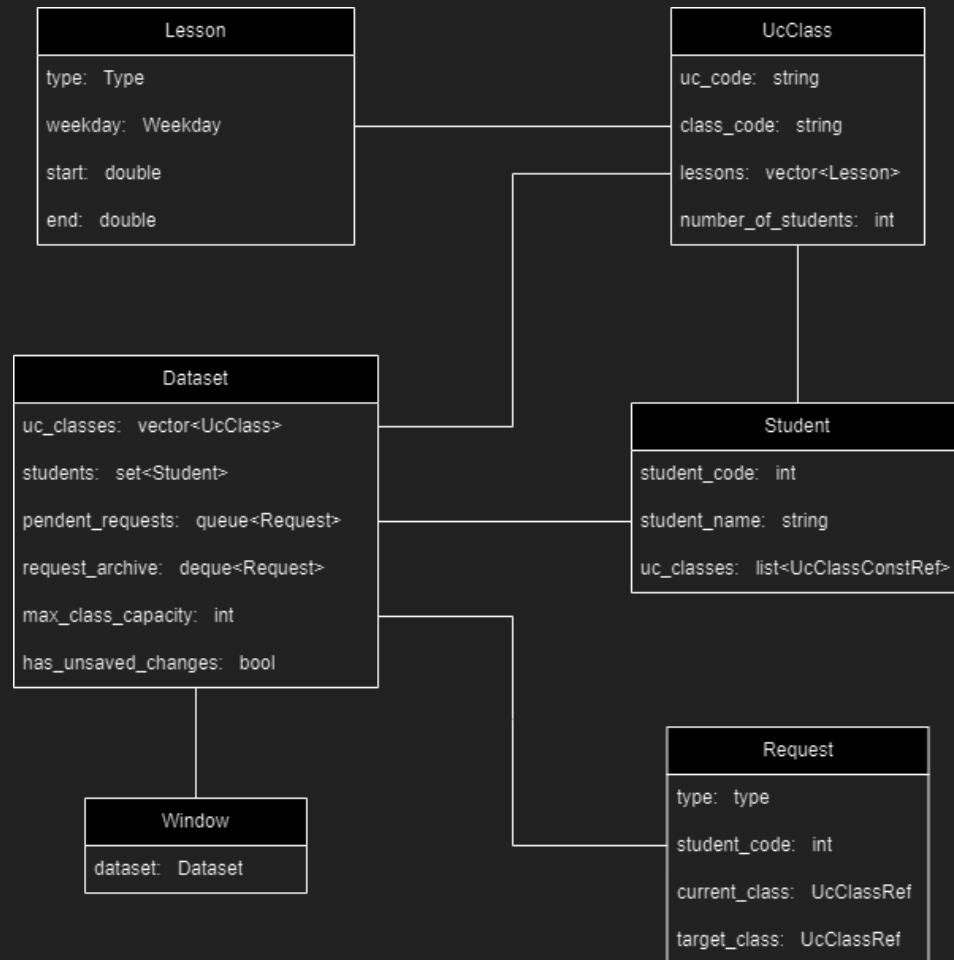
- Student :

Para guardar dados relativos a um estudante, tais como o seu nome e código UP, foi criada a classe Student. Esta permite descobrir as turmas e unidades de um determinado aluno, tal como a verificação da sua integração em qualquer turma e unidade curricular. Em adição, podemos aceder ao seu ano académico e, também, ao seu ano de admissão.

- Request :

Esta classe é uma abstração para pedidos de adição, remoção e mudança de turmas. Evidentemente, nesta conseguimos acessar ao tipo do pedido, o código UP do estudante que pretende trocar de turma, a sua turma atual e a turma para a qual este pretende mudar.

# Diagrama de Classes



# Classes Utilizadas

- Window :

A classe Window permite o utilizador interagir com o conjunto de dados. Deste modo, o utilizador pode aceder a todas as listagens implementadas, tais como todos os alunos de determinada turma, pesquisar um aluno pelo seu código UP, ... Para além disto, pode consultar o horário de um estudante em diagrama ou visualmente. Finalmente, pode também gerir os pedidos da classe Request.

- Dataset :

A classe Dataset guarda a implementação de todas as listagens e pedidos, tais como procurar um aluno pelo seu código UP, listar todas as turmas do conjunto de dados, adicionar e remover estudantes de turmas,... Em adição, é nesta classe onde estão guardadas as estruturas de dados que contêm todos os alunos, aulas, pedidos pendentes e pedidos arquivados

# Estruturas de dados usadas

- Sets (estudantes):
  - Permite pesquisas e inserções rápidas, mantendo a ordem (por código de estudante)
- Vetores ordenados (UcClasses e lições):
  - UcClasses ordenadas lexicograficamente por código de UC e código de turma e lições pela hora de início na semana
  - Não sendo modificados (em si) durante a execução do programa, permitem pesquisas binárias sem abdicar de acesso aleatório e mutabilidade dos elementos
- Lista (UcClasses de estudante):
  - A inserção e remoção de elementos pode ser feita com complexidade constante
  - É quase sempre necessário percorrer todos os elementos quando usada

# Estruturas de dados usadas

- Queue (pedidos pendente):
  - Estrutura FIFO: adequa-se ao facto de que os primeiros pedidos devem ser efetuados primeiro
- (~~Stack~~) Deque (pedidos arquivados):
  - Consegue implementar a estrutura LIFO necessária à implementação do sistema de desfazimento de pedidos de forma eficiente
  - Permite percorrer todos os elementos do contentor, para a escrita do arquivo em ficheiro

# Funcionalidades do Programa (Listagens)

- Listar todos os estudantes
- Listar todas as unidades curriculares
- Listar todas as turmas
- Listar as ocupações de todas as turmas
- Procurar turmas em unidades curriculares
- Procurar estudante pelo código UP
- Procurar estudante em unidade curricular
- Procurar estudante em turma
- Procurar estudante em Turma de Unidade Curricular
- Procurar estudante por ano Académico
- Procurar estudante por ano de Admissão
- Procurar estudante em pelo menos n Unidades Curriculares



## Funcionalidades do Programa (Horários)

- Consultar horário em diagrama de um estudante
- Consultar horário visual de um estudante
- Consultar horário em diagrama de uma turma
- Consultar horário visual de uma turma

# Funcionalidades do Programa (Pedidos)

- Adicionar aluno a uma turma
- Remover aluno de uma turma
- Trocar aluno de turma
- Desfazer um pedido

## Ordenação das Listagens

- Ordenação por ordem alfabética
- Ordenação por ordem alfabética inversa
- Ordenação por código UP Ascendente
- Ordenação por código UP Descendente
- Ordenação por número ascendente de unidades curriculares
- Ordenação por número descendente de unidades curriculares

# Funcionalidades

- Preservação das alterações do sistema:

São guardadas todas as alterações efetuadas no sistema, assim como um registo dos pedidos efetuados no sistema. Assim, o programa, ao ser iniciado novamente, receberá todas as mudanças efetuadas anteriormente.