

# Desenvolvimento de Sistemas de Software

Licenciatura em Engenharia Informática

Departamento de Informática

Universidade do Minho

2024/2025

---

## Enunciado do Trabalho

---

José Creissac Campos  
jose.campos@di.uminho.pt

António Nestor Ribeiro  
anr@di.uminho.pt

Rafael Costa  
d13414@di.uminho.pt

## Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Objectivo do trabalho</b>	<b>1</b>
<b>3</b>	<b>Cursos, horários, unidades curriculares e alunos</b>	<b>1</b>
<b>4</b>	<b>Realização do trabalho</b>	<b>2</b>
4.1	Entrega intermédia . . . . .	2
4.2	Entrega final . . . . .	2
<b>5</b>	<b>Apresentação e discussão do trabalho</b>	<b>3</b>
<b>6</b>	<b>Avaliação</b>	<b>4</b>
<b>7</b>	<b>Grupos de Trabalho</b>	<b>4</b>

## 1 Introdução

Este documento apresenta o enunciado do trabalho prático da Unidade Curricular (UC) de Desenvolvimento de Sistemas Software para o ano lectivo 2024/2025. **Leia-o com atenção**, já que descreve, não só o sistema a desenvolver, como o processo que deve seguir para a realização do trabalho. Quaisquer dúvidas devem ser esclarecidas junto dos docentes da UC.

## 2 Objectivo do trabalho

Em cursos de grande dimensão, como é o caso da Licenciatura em Engenharia Informática (LEI), os alunos têm de ser distribuídos por turnos para viabilizar a realização das aulas. A gestão dessa distribuição por turnos é um processo complexo, uma vez que é necessário compatibilizar os turnos das diferentes Unidades Curriculares (UC) de cada ano do curso (e, eventualmente, de diferentes anos do curso), de modo a que os alunos não tenham colisões no seu horário.

Historicamente, os docentes de cada UC geriam a definição dos turnos da sua UC. Esta solução mostrou não ser escalável, criando situações em que os alunos tinham dificuldades em obter horários sem colisões e atrasando o início das aulas.

Uma forma de facilitar o processo é centralizar a criação dos turnos nas direções de curso. No entanto, essa criação é complexa e demorada, consumindo recursos, pelo que só é viável se as listas de inscritos forem conhecidas com antecedência suficiente. Além disso, tem as suas próprias desvantagens, uma vez que não permite a cada UC definir a sua política para a criação dos turnos (por exemplo, requerendo que todos os elementos de cada grupo prático frequentem o mesmo turno), nem permite aos alunos controlo sobre o seu horário.

Com este projeto pretende-se desenvolver um sistema para auxiliar a criação dos turnos práticos de um curso, dadas as suas UC, o horário e as listas de inscritos. O sistema deverá suportar diferentes algoritmos de priorização de alunos na criação dos turnos.

## 3 Cursos, horários, unidades curriculares e alunos

Para a realização deste trabalho recorra aos conhecimentos que tem sobre cursos, horários, UC e alunos. Relembre que um curso é organizado em semestres, tendo em cada semestre um conjunto de UC (algumas poderão ser UC opcionais com várias alternativas), podendo os alunos estar inscritos a UC de semestres diferentes.

Ao pensar na sua solução tenha em atenção que diferentes tipos de aula podem ter limites de alunos por turma diferentes. No caso das aulas teóricas, o limite é a capacidade da sala a que a aula está alocada no horário. No caso de aulas teórico-práticas (TP) e práticas laboratoriais (PL) o limite é definido UC a UC e poderá ser inferior ao tamanho da sala.

Tenha ainda em atenção que os alunos com estatuto especial (trabalhadores-estudantes e atletas, por exemplo) devem ter prioridade na alocação a turnos. A forma de refletir essa prioridade nos turnos depende da estratégia adoptada para a sua criação. Estratégias comumente utilizadas no passado incluem reservar alguns lugares em cada turno para esses alunos, ou pedir de antemão as suas preferências (quando feito de forma manual, apenas viável em turmas pequenas).

Para esclarecimento de eventuais dúvidas contacte os docentes de DSS.

## 4 Realização do trabalho

A concepção e desenvolvimento da aplicação deverá seguir uma abordagem baseada em modelos (suportada por UML), de acordo com o processo de entregas faseadas descrito nas aulas teóricas. A aplicação deverá ser desenvolvida utilizando uma arquitectura multi-camada e tecnologias orientadas a objectos (preferencialmente, Java). Irá ser criado um repositório no GitHub<sup>1</sup> para cada grupo, onde deverá ser mantida a versão actualizada do trabalho.

Para facilitar o processo de concepção e desenvolvimento, o trabalho será realizado em duas fases.

### 4.1 Entrega intermédia

Análise de requisitos – a entregar até às 23h59 de 19 de Outubro.

Objectivos:

- Um Modelo de Domínio com as entidades relevantes
- Um Modelo de Use Case (diagramas mais especificações do Use Case) com as funcionalidades propostas para o sistema

O resultado desta fase será sujeito a avaliação qualitativa.

### 4.2 Entrega final

Modelação conceptual e implementação da solução – a entregar até às 23h59 de 7 de dezembro (atualizações possíveis até 4 de dezembro).

---

<sup>1</sup><https://github.com>

Objectivos:

- Uma arquitectura conceptual do sistema, capaz de suportar os requisitos identificados – em particular a solução adoptada deverá garantir, tanto quanto possível, que não são cometidos erros como gerar turnos com conflitos
- Os modelos comportamentais necessários para descrever o comportamento pretendido para o sistema
- Os modelos que considere necessários à descrição da implementação do sistema
- A implementação do sistema
- Documento técnico com todos os modelos desenvolvidos (em PDF).

Pretende-se que o documento técnico sirva de apoio à análise do trabalho, pelo que **deverá ter a seguinte estrutura:**

- **Capa com identificação** da Unidade Curricular, **do grupo (com fotos dos elementos)** e o URL do **repositório do trabalho**.
- Descrição dos resultados obtidos (máximo uma página).
- Diagramas relativos à **análise de requisitos** (Modelação de Domínio, Diagramas de *Casos de Uso* e correspondentes descrições dos casos de uso).
- Diagramas relativos à **modelação conceptual da solução** proposta (Diagramas de Classe e de Sequência).
- Diagramas com a descrição da **solução efectivamente implementada** (Diagramas de Classe, de Sequência, de Componentes e de *packages*).
- Manual de utilização do sistema desenvolvido.
- Em anexo, este enunciado.

Os diagramas mencionados acima podem ser complementados com outros que considerem relevante incluir.

## 5 Apresentação e discussão do trabalho

Para a apresentação do trabalho deverão preparar uma apresentação com a duração máxima de 15 minutos. Esta apresentação deverá descrever a solução e a

abordagem seguida para a atingir, desde a análise dos cenários até a implementação e demonstração da solução final. A apresentação deverá terminar com uma análise crítica dos resultados obtidos.

Após essa apresentação, seguir-se-á um período de análise e discussão do trabalho de até 30 minutos.

## 6 Avaliação

A apresentação e discussão final do trabalho será realizada na semana de 6 a 10 de janeiro de 2025, em horários a combinar. A **presença** na discussão do trabalho é **obrigatória**.

Os pesos relativos de cada componente do trabalhos serão os seguintes:

- Modelo de domínio e análise de requisitos: 25%
- Modelação conceptual: 25%
- Modelação final e implementação: 35%
- Apresentação e discussão: 15%

A nota de cada elemento do grupo será individual, tendo em consideração a nota do trabalho e a avaliação por pares. A equipa docente reserva-se a possibilidade de ajustar as notas, em função da sua avaliação de cada elemento durante a discussão do trabalho.

## 7 Grupos de Trabalho

Os grupos de trabalho deverão obrigatoriamente ser constituídos por de 3 a 5 elementos. A definição dos grupos de trabalho será realizada no Blackboard, **terminando a 2 de outubro**.