

# utad

UNIVERSIDADE  
DE TRÁS-OS-MONTES  
E ALTO DOURO

2025



Parte 1

# RELATORIO

Laboratório de Planeamento e Desenvolvimento de Software

**Grupo 6**

AI80990 Bruno Alves

AI82339 Filipe Silva

AI79012 Liane Duarte

AI81311 Pedro Braz

# Índice

Introdução .....	2
Objetivos etapa 1 .....	2
Modelos de Dados .....	2
Atributos implementados: .....	3
Mockups das Interfaces.....	4
Conclusão .....	10

# Introdução

Este relatório diz respeito à **Etapa 1** do Trabalho Prático da unidade curricular *Laboratório de Planeamento e Desenvolvimento de Software*, do curso de Licenciatura em Engenharia Informática.

Nesta fase inicial, partindo dos requisitos, casos de uso e modelo conceptual já definidos no protocolo fornecido pelos docentes, foi realizado o planeamento da aplicação a desenvolver. O trabalho consistiu na criação dos **mockups de todas as interfaces (views)** necessárias e na implementação da camada **Model**, utilizando a linguagem **C#** no ambiente **Visual Studio 2022**, com recurso à framework **.NET 9.0** e seguindo o padrão **MVVM**.

Esta etapa tem como objetivo estabelecer a base visual e estrutural do sistema, preparando os componentes fundamentais para as fases seguintes do desenvolvimento.

## Objetivos etapa 1

A **Etapa 1** do trabalho prático teve como principal objetivo o **planeamento inicial da aplicação**, com base na descrição funcional e nos requisitos já fornecidos no protocolo. Esta fase teve como metas:

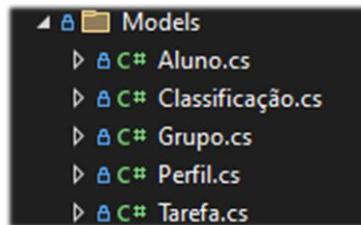
- Desenvolver os **mockups** de todas as interfaces (views) necessárias para as funcionalidades descritas;
- Estruturar a camada **Model** da aplicação em C#, refletindo as entidades e relacionamentos definidos no diagrama de classes;

Estes objetivos visam assegurar que a aplicação a ser desenvolvida esteja bem planeada desde o início, facilitando o desenvolvimento modular e organizado nas etapas seguintes.

## Modelos de Dados

Com base no **diagrama de classes fornecido no protocolo**, foram implementadas as classes que constituem a camada **Model** da aplicação. Os seguintes ficheiros foram criados na pasta *Models*: **Aluno.cs** , **Classificação.cs**, **Grupo.cs** , **Perfil.cs** e **Tarefa.cs**.

Estas classes modelam os principais elementos da aplicação e servem de base para o armazenamento e manipulação dos dados ao longo das restantes camadas do padrão MVVM.



## Atributos implementados:

Classe Aluno:

```
public class Aluno
{
    0 references
    public int Numero { get; set; } // Número de identificação do aluno
    0 references
    public string Nome { get; set; } // Nome completo do aluno
    0 references
    public string Email { get; set; } // Endereço de email do aluno
}
```

Classe Classificação.cs:

```
public class Classificacao
{
    0 references
    public int Alunoid { get; set; } // Número do aluno a que se refere a classificação
    0 references
    public int TarefaId { get; set; } // Identificador da tarefa avaliada
    0 references
    public double Valor { get; set; } // Valor da nota atribuída
}
```

Classe Grupo.cs:

```
public class Grupo
{
    0 references
    public int Id { get; set; } // Identificador único do grupo
    0 references
    public string Nome { get; set; } // Nome ou designação do grupo
    0 references
    public List<Aluno> Alunos { get; set; } // Lista de alunos pertencentes ao grupo
}
```

Classe Perfil.cs:

```
public class Perfil
{
    0 references
    public string Nome { get; set; } // Nome do utilizador (professor)
    0 references
    public string Email { get; set; } // Endereço de email do utilizador
    0 references
    public string Fotografia { get; set; } // Caminho ou URL da fotografia do utilizador
}
```

### Classe Tarefa.cs:

```
public class Tarefa
{
    0 references
    public int Id { get; set; }          // Identificador único da tarefa
    0 references
    public string Titulo { get; set; }    // Título da tarefa
    0 references
    public string Descricao { get; set; } // Descrição opcional da tarefa
    0 references
    public DateTime DataInicio { get; set; } // Data e hora de início da tarefa
    0 references
    public DateTime DataTermino { get; set; } // Data e hora de término da tarefa
    0 references
    public double Peso { get; set; }       // Peso da tarefa na avaliação final
}
```

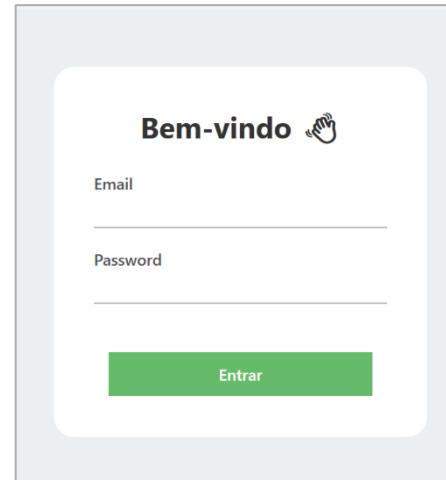
## Mockups das Interfaces

As interfaces da aplicação foram planeadas com base no **diagrama de casos de uso** disponibilizado no protocolo (Anexo 1). Cada **caso de uso** representa uma funcionalidade principal da aplicação e serviu de base para o desenho dos **mockups das views**.

O objetivo desta etapa foi definir visualmente a estrutura da aplicação, garantindo que todas as funcionalidades essenciais — como a gestão de alunos, grupos, tarefas, classificações, perfil e visualização de resultados — estejam representadas de forma clara e funcional.

### loginWindow.xaml – Ecrã de Login

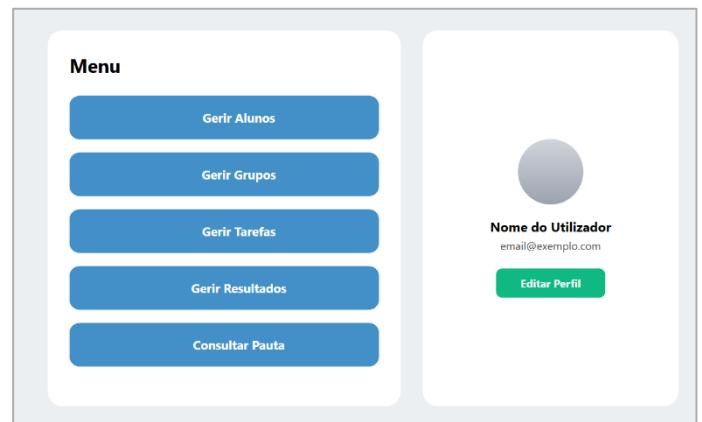
Esta é a interface inicial da aplicação, onde o utilizador (professor) insere os seus dados para aceder ao sistema. Embora esta view não estivesse explicitamente descrita no protocolo, decidimos incluí-la por ser um elemento básico e comum em praticamente todas as aplicações que envolvem interação com um utilizador autenticado. A sua implementação melhora a segurança e a estrutura geral da aplicação.



## Dashboard.xaml – Painel Principal

Esta interface funciona como o painel principal da aplicação, disponibilizando atalhos e navegação para as principais funcionalidades: gestão de alunos, grupos, tarefas, classificações, consulta de pauta e perfil.

Apesar de não estar mencionada de forma explícita no protocolo, considerámos a sua implementação essencial para garantir uma melhor organização e navegação dentro da aplicação, facilitando a experiência do utilizador.



## Gerir\_alunos.xaml – Gestão de Alunos

Esta interface permite visualizar, adicionar, editar e remover alunos da aplicação, através de um sistema de gestão com operações CRUD. Além disso, foi implementada a funcionalidade de **importar** uma lista de alunos a partir de ficheiros .csv ou .xlsx, conforme previsto no protocolo. Como melhoria adicional, foi também incluída a opção de **exportar** a lista de alunos para ficheiro, facilitando o backup e o uso dos dados noutras contextos. Esta funcionalidade não estava prevista no protocolo, mas considerámos que acrescenta valor à aplicação e melhora a experiência do utilizador.

Gerir Alunos		
		Pesquisar
<b>Lista de Alunos</b>		
80990	Bruno Braz	Bruno.Braz@email.com
80993	Filipe Duarte	Filipe.Duarte@email.com
		<a href="#">Editar</a> <a href="#">Excluir</a>
		<a href="#">Editar</a> <a href="#">Excluir</a>
<a href="#">Adicionar Aluno</a> <a href="#">Importar</a> <a href="#">Exportar</a> <a href="#">Menu</a>		

### **Novo\_aluno.xaml – Adicionar Aluno**

Interface específica para a criação de um novo aluno, com campos para preencher o número, nome e email.

The screenshot shows a form titled "Novo Aluno +". It contains three input fields: "Número", "Nome", and "Email", each with a corresponding text input below it. Below the fields are two buttons: a green "Adicionar" button and a red "Cancelar" button.

### **Gerir\_Grupos.xaml – Gestão de Grupos**

Permite consultar e gerir os grupos existentes, bem como ver os alunos associados a cada grupo.

The screenshot shows a table titled "Gestão de Grupos". The columns are "Grupo", "Alunos", and "Ações". The data rows are:

Grupo	Alunos	Ações
Grupo 1	João, Ana, Pedro	<a href="#">Editar</a> <a href="#">Eliminar</a>
Grupo 2	Carla, Tiago	<a href="#">Editar</a> <a href="#">Eliminar</a>
Grupo 3	Filipe, Ricardo	<a href="#">Editar</a> <a href="#">Eliminar</a>

At the bottom are two buttons: "Voltar ao Menu" and "Criar Grupo".

## **editar\_Grupo.xaml – Edição de Grupo**

Página dedicada à edição do nome de um grupo e à gestão da lista de alunos pertencentes ao mesmo e da criação de novos grupos.

**Criar/Editar Grupo**

Nome do grupo

**Alunos no grupo**

81234	João Silva	<input type="button" value="Remover"/>
51236	Filipe Costa	<input type="button" value="Remover"/>

**Alunos disponíveis**

81235	Ana Martins	<input type="button" value="Adicionar"/>
81237	Sara Gomes	<input type="button" value="Adicionar"/>
81238	Ricardo Nunes	<input type="button" value="Adicionar"/>

## **Gerir\_Tarefas.xaml – Gestão de Tarefas**

Interface onde se podem criar, editar e eliminar tarefas de avaliação, definindo a designação, datas e peso na nota final.

Título	Início	Fim	Peso	Editar	Eliminar
Tarefa 1	22-04-2024	20-24	30	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
Tarefa 2	24-04-2024	25-24	20	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
Tarefa 3	26-04-2024	27-24	25	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>
Tarefa 4	28-04-2024	29-24	25	<input type="button" value="Editar"/>	<input type="button" value="Eliminar"/>

**Nova Tarefa**

Título

Descrição

Início

Fim

Peso

## **Gerir\_Resultados.xaml – Gestão de Classificações**

Permite registar ou editar classificações por tarefa, associadas a cada grupo.

### **Gestão de Resultados**

Selecionar Grupo: **Grupo 1**    Selecionar Tarefa: **Tarefa 1**

**Grupo Selecionado:** Grupo 1  
**Membros:** João Silva, Ana Martins, Pedro Costa  
**Tarefa Selecionada:** Tarefa 1 - Análise de Requisitos (Peso: 25%)

Número	Nome do Aluno	Classificação	Ações
81234	João Silva	15	<b>Guardar</b>
81235	Ana Martins	14	<b>Guardar</b>
			▼

Atribuir mesma nota a todos:  **Aplicar a Todos** **Voltar ao Menu**

## **Consultar\_Pauta.xaml – Consulta da Pauta**

Mostra a pauta final dos alunos, com as notas obtidas em cada tarefa e a média final ponderada.

### **Pauta de Avaliação**

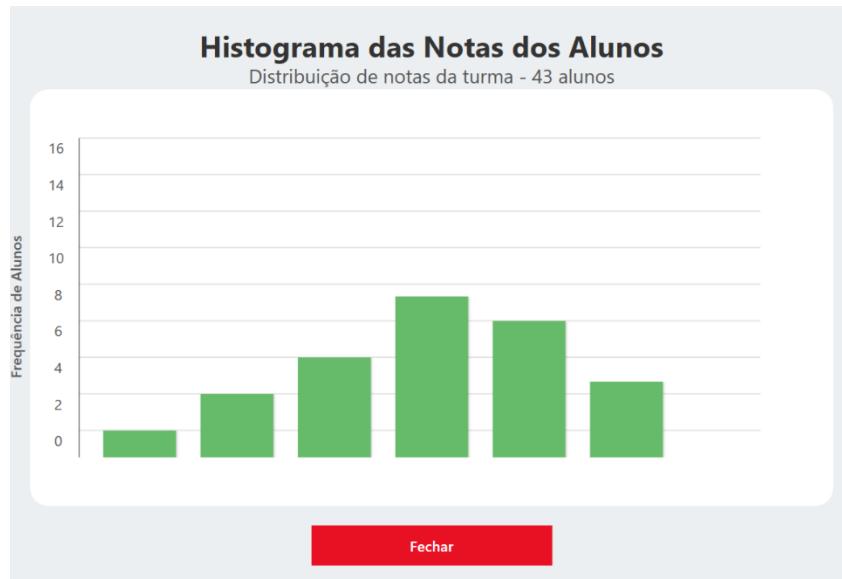
Pesquisar aluno: \_\_\_\_\_ Procurar

Número	Nome	Tarefa 1	Tarefa 2	Tarefa 3	Tarefa 4	Nota Final
20210001	João Silva	15	14	16	18	15,8
20210002	Ana Martins	12	17	15	14	14,4
20210003	Marco Pereira	10	11	13	12	11,8
20210004	Cláudia Sousa	8	9	12	10	9,8

Total de Alunos: 4    Média da Turma: 12,95 Voltar ao Menu Ver Histograma

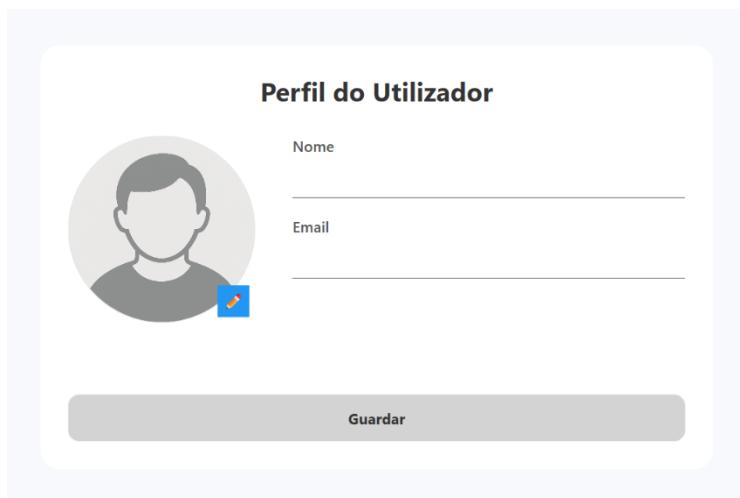
## **histograma.xaml – Histograma de Notas**

Apresenta um gráfico com a distribuição das classificações de todos os alunos, permitindo visualizar o desempenho geral da turma.



### **Gerir\_Perfil.xaml – Gestão de Perfil**

Interface específica para a edição do perfil, com campos para substituir o nome, email, e foto.



### Conclusão

A Etapa 1 permitiu estabelecer as bases estruturais e visuais da aplicação, com a criação dos modelos de dados e dos mockups das principais interfaces. Com base no protocolo fornecido, foram implementadas as classes que representam os elementos essenciais da aplicação e desenvolvidos os mockups alinhados com os casos de uso definidos.

Foram ainda incluídas algumas funcionalidades adicionais, como a view de login e a exportação de dados, com o objetivo de melhorar a experiência do utilizador e reforçar a robustez da aplicação. Esta etapa garante uma base sólida para o desenvolvimento das camadas **View** e **ViewModel** na próxima fase do projeto.



