

# Relatório BD – Meta 1

## Membros:

- Fábio Campos Fernandes - [uc2023230805@student.uc.pt](mailto:uc2023230805@student.uc.pt)
- João Vítor Alves - [uc2016122878@student.uc.pt](mailto:uc2016122878@student.uc.pt)
- Pedro Gonçalves Vaz – [uc2020225812@student.uc.pt](mailto:uc2020225812@student.uc.pt)

## Descrição do projeto

O projeto tem como objetivo desenvolver um sistema de gestão universitária que centralize as atividades académicas e extracurriculares da instituição. Este sistema visa organizar e integrar os dados dos alunos, professores e funcionários, facilitando o acesso, a atualização e o controlo de toda a informação.

O sistema permite o registo de utilizadores, onde cada pessoa é identificada de forma única com os seus dados pessoais, tais como nome, email e data de nascimento. Cada pessoa é classificada de acordo com o seu papel – por exemplo, aluno, professor ou funcionário – o que evita duplicações e garante uma identificação clara. A abordagem utilizada é a de herança, em que a entidade “Pessoa” serve de base para as restantes entidades especializadas.

Entre as funcionalidades principais, encontram-se:

- **Matrícula de Alunos:** O sistema permite que os alunos se inscrevam em cursos e programas de grau. Durante o processo de matrícula, são verificadas as condições, como a disponibilidade de vagas e os pré-requisitos necessários para a inscrição.
- **Gestão de Cursos e Edições:** Cada curso é identificado por um código único e pode ter várias edições, definidas por ano, semestre e capacidade. O sistema diferencia entre aulas teóricas e práticas, possibilitando a definição de horários e locais específicos para cada turma.
- **Inscrição em Atividades Extracurriculares:** Os alunos podem registrar a sua participação em clubes, desportos e outras atividades. Ao inscrever-se, o sistema atualiza automaticamente a conta financeira do aluno com as taxas correspondentes, usando triggers para garantir que todas as alterações sejam efetuadas de forma consistente.
- **Lançamento de Notas e Controlo de Presença:** Os professores têm a possibilidade de registrar as notas dos alunos, enquanto o sistema atualiza os registos de presença. Estas informações são utilizadas para calcular a média académica e monitorizar o desempenho dos alunos.

- **Gestão Financeira:** A conta financeira de cada aluno regista as mensalidades, taxas e outras cobranças. As operações financeiras, como a inscrição ou o cancelamento de uma atividade, desencadeiam atualizações automáticas na conta, mantendo os registos sempre atualizados.

O projeto realça também a importância do controlo das transações e dos conflitos de concorrência, garantindo que todas as operações sejam atómicas e que os problemas de concorrência, como as condições de corrida e as atualizações perdidas, sejam devidamente tratados. Além disso, são implementadas estratégias de segurança para proteger os dados e garantir a integridade do sistema.

Resumindo, o sistema de gestão universitária desenvolvido neste projeto abrange todas as áreas essenciais da administração académica e financeira, proporcionando uma solução integrada que apoia as atividades educativas e extracurriculares de forma eficiente e segura.

## Transações

Uma transação é uma sequência de operações (ex.: leitura, escrita, atualização) que devem ser tratadas como uma única unidade lógica. Ou todas as operações são concluídas com sucesso, ou nenhuma é aplicada, garantindo a consistência e integridade dos dados.

### Transações no projeto:

- **Inscrição numa disciplina.** - Verificar pré-requisitos, confirmar vagas e registar aulas escolhidas. Caso algum destes pontos falhe é necessário cancelar a transação.
- **Submissão de notas** - É necessário atualizar as médias académicas. Caso haja valores inválidos, como ID inexistente ou notas negativas, cancela-se a transação.
- **Inscrição de um aluno num curso** - Associamos o aluno a um curso e geramos automaticamente uma dívida nova, usando um trigger.

## Conflitos de concorrência

Conflitos de concorrência ocorrem quando duas ou mais transações tentam aceder ou modificar os mesmos dados ao mesmo tempo, gerando resultados inconsistentes.

## Possíveis conflitos no projeto:

- **Duas inscrições na última vaga de uma disciplina** – A solução é dar lock à edição de uma disciplina usando o FOR UPDATE durante a transação.
- **Atualizações de notas para o mesmo aluno simultaneamente** – A solução vai passar pelo uso de um lock de linha (SELECT ... FOR UPDATE) para garantir que os dados são apenas modificados por uma transação de cada vez.

## Plano de desenvolvimento

Durante a fase inicial do projeto, o grupo optou por uma abordagem colaborativa sem divisão formal de tarefas. Para estruturar o trabalho, foram realizadas 5 reuniões de *brainstorming* ao longo de duas semanas, utilizámos a plataforma *Discord* como meio de comunicação principal. Estas sessões permitiram:

- Alinhar a compreensão dos requisitos funcionais descritos no Anexo A;
- Definir prioridades com base na complexidade e nas dependências entre funcionalidades;
- Promover a participação equitativa de todos os membros na tomada de decisões.

## Modelagem Inicial das Entidades

O desenvolvimento começou pela criação das entidades mais diretas, seguindo a ordem sugerida no enunciado. Por exemplo:

- **Entidade "Student"**: Foi priorizada por ser central para funcionalidades como inscrições, registos financeiros e atividades extracurriculares.
- **Entidades Relacionadas**: Como “Degree”, “ExtraActivity” e “Financial Account”, foram criadas em paralelo para estabelecer relações e restrições de integridade desde o início.

Este processo iterativo garantiu que a equipa construísse uma base sólida para o modelo de dados, facilitando a integração de funcionalidades mais complexas (ex.: triggers, transações) em fases posteriores. A ausência de divisão rígida de tarefas permitiu a troca de ideias, enquanto as reuniões regulares mantiveram o progresso alinhado com os prazos estabelecidos.

