

Justificativa

Demonstrar, por meio da condução de um projeto prático, as habilidades e competências adquiridas no decorrer da disciplina. Neste trabalho, os alunos serão desafiados a criar uma aplicação web completa. O *back-end* será uma Web API em C#, com acesso a dados persistidos em SQLite e manipulados por meio do Entity Framework (EF). O *front-end* será desenvolvido em Angular.

Objetivo Geral

O objetivo é desenvolver operações CRUD (Create, Read, Update, Delete) para diferentes entidades de dados e implementar regras de negócio específicas. O trabalho será realizado em equipes de, no mínimo três integrantes humanos, com o Chat GPT atuando como um membro extra para auxiliar no desenvolvimento dos CRUDs básicos.

Tema

A equipe terá a liberdade de escolher o tema para a aplicação.

Objetivos Específicos

1. **Definição do Tema:** A equipe deve escolher o tema da aplicação, que deve envolver a manipulação de dados de maneira significativa. O tema escolhido deve ser aprovado pelo professor.
2. **Modelagem de Dados:** A equipe deve projetar a estrutura do banco de dados SQLite, incluindo todas as tabelas necessárias para a aplicação. Isso deve ser feito em colaboração com o ChatGPT para garantir uma modelagem adequada. Pode ser utilizado um SGBD como o MySQL, MS SQL Server etc, desde que aprovado pelo professor.
3. **Documentação:** Juntamente com o código da aplicação, deve ser produzida uma breve documentação contendo um resumo do projeto, lista das principais funcionalidades, diagrama de classes UML e relatório de atividades desenvolvidas com auxílio do ChatGPT (explicando como aconteceu).
4. **Desenvolvimento dos CRUDs Básicos:** Cada integrante da equipe deve ser responsável por desenvolver CRUDs básicos para duas entidades específicas relacionadas ao tema escolhido. O ChatGPT está disponível para auxiliar na codificação inicial dos CRUDs.
5. **Implementação das Regras de Negócio:** A equipe deve identificar as regras de negócio mais importantes relacionadas ao tema da aplicação e implementá-las. Isso inclui validações, cálculos ou lógicas específicas do domínio.
6. **Testes e Correções:** A equipe deve realizar testes para garantir que a aplicação funcione corretamente. Qualquer bug ou problema deve ser corrigido.

Requisitos da Aplicação

1. Utilizar alguma ferramenta para compartilhamento do código;
2. O sistema deve conter pelo menos duas entidades/modelos/tabelas para cada integrantes da equipe, ou seja, um CRUD de uma entidade por integrante, sendo que o ChatGPT é considerado um dos integrantes;
3. Todas as entidades devem conter a inserção, remoção, alteração e listagem das informações em banco de dados;
4. Uma das entidades deve ter relacionamento com algo já cadastrado no banco. Exemplo: supondo um sistema de e-commerce, primeiro fazemos o cadastro de uma categoria, para depois cadastrar o produto, fazendo o vínculo com uma categoria já existente;
5. O sistema não poderá apenas ter apenas cadastros básicos, ou seja, a partir dos inputs no sistema, os dados devem ser aproveitados para uma funcionalidade que dê sentido ao aplicativo. Exemplo: após os produtos e usuário cadastrado no sistema, o usuário deve ser capaz de gerar um carrinho de compras, para depois gerar uma venda no sistema;
6. A aplicação back-end deve utilizar C#, Web API, Entity Framework (EF) e SQLite.

Equipes

As equipes devem conter preferencialmente entre 3 e 5 integrantes humanos. Formações com menos de 3 integrantes humanos não serão aceitas. Formações com mais de 5 integrantes humanos devem ser validadas pelo professor, sendo que o escopo do projeto e o volume dos artefatos produzidos será cobrado de acordo com a quantidade de integrantes.

Entregas / Avaliações

O projeto será avaliado sob duas perspectivas: 1) pelo projeto entregue em grupo; e 2) pela defesa oral individual. A pontuação será distribuída entre as avaliações A1 e A2 conforme o quadro a seguir:

Av.	Modalidade	Conteúdo	Valor
A2	Projeto (código)	<i>Back-end</i> C# com todas as entidades, CRUD envolvendo models e controllers	1,0
A2	Prova oral	Três perguntas individuais sobre o código <i>back-end</i> desenvolvido	3,0
A2	Projeto (código)	<i>Front-end</i> Angular com todas as views	1,0
A1	Prova oral	Cinco perguntas individuais sobre toda a aplicação desenvolvida	5,0

As datas de entregas/avaliações estão disponíveis no plano de ensino da disciplina no Blackboard.

Considerações finais

Situações não previstas neste documento serão tratadas em comum acordo entre professor e estudantes.