

## *Trabalho Final de Técnicas de Programação*

**Grupo 13:** Giulia Ferraro e Isabella Pagnoncelli

**Definição:** O objetivo do Trabalho final da cadeira de Técnicas de programação é o desenvolvimento de um sistema Web utilizando os conceitos vistos em aula. Nosso grupo decidiu utilizar como tema uma plataforma online de compra e entrega de comidas de um restaurante específico, onde o usuário terá uma variedade de petiscos, pratos prontos e sobremesas para escolher. Ao finalizar seu pedido, receberá todos os itens selecionados no endereço desejado. Além disso, o sistema contará com um design simples e prático, tornando mais fácil a utilização deste.

**Relatório:** Este relatório tem como objetivo relatar como foi o desenvolvimento do trabalho final da disciplina de Técnicas de Programas bem como descrever quais foram as dificuldades enfrentadas pelo grupo.

O objetivo deste trabalho era realizar uma aplicação Web de um restaurante simulando a realização de pedidos por um cliente utilizando os conceitos da arquitetura CLEAN. As tecnologias utilizadas no desenvolvimento foram Spring Boot com JPA, MySQL e Angular.

Na camada de Entidades não tivemos dificuldades em desenvolvê-las e nem testá-las pois é algo que já estávamos acostumadas em fazer. Enquanto estávamos desenvolvendo a camada de Casos de Uso, o Spring Boot parou de reconhecer os drivers de testes e não conseguimos mais testar nenhuma Entidade, ao utilizar a notação `@SpringBootTest` passando o path das classes, o erro desapareceu e voltamos a testar sem problemas futuros.

A camada de Interface foi a que tivemos mais dificuldade em desenvolver. Criamos nosso front-end utilizando o framework Angular e preferimos criar um design simples para nossa aplicação. Porém, como nenhuma de nós, integrantes do grupo, tínhamos muito conhecimento em Front-End, conseguimos criar um simples modelo mockando a Entidade Pratos, porém não conseguimos reproduzir o mesmo para os Pedidos. Tivemos vários erros quando tentávamos usar a mesma classe de Mocks para duas telas diferentes, e outros erros quando fazíamos classes separadas. Dessa forma, como não tínhamos muito tempo sobrando, decidimos deixar apenas a Classe de Pratos mockada.

Outro erro que encontramos no caminho, foi ao tentar criar uma conexão com o Banco MySQL, para popularmos a aplicação e testar os endpoints no PostMan. Como não encontramos nenhuma solução viável para esse problema, decidimos usar um banco em memória, entretanto, esta opção também trouxe complicações. Ao darmos os Selects com os dados dos Pratos, não tivemos nenhuma incomodação, mas ao fazer o mesmo com os Pedidos, tivemos diversos erros diferentes, acreditamos que esteja relacionado com as anotações `@JoinColumn`, mas como também não temos muito conhecimento em relação a Banco de Dados, não conseguimos resolver este contratempo.