



HEY, PRONTOS PARA O DESAFIO FINAL?

Muito bem agora com o front-end concluído do primeiro desafio, iremos construir uma API para que possamos ter o controle interno sobre as informações de nosso sistema Radar

Endpoints a ser criados na API:

- Produtos (id, nome, descricao, valor, qtd_estoque)
- Clientes (id, nome, telefone, email, cpf, cep, logradouro, numero, bairro, cidade, estado, complemento)
- Pedidos (id, cliente_id, valor_total, data)
- PedidosProdutos (id, pedido_id, produto_id, valor, quantidade)
- Campanhas (id, nome, descricao, data, url_foto_prateleira)
- Lojas (id, nome, observacao, cep, logradouro, numero, bairro, cidade, estado, complemento, latitude, longitude)
- PosicoesProdutos (id, campanha_id, posicao_x, posicao_y)

O sistema será desenvolvido em ASP.NET Core API C#

Funcionalidades extra:

- Autenticação JWT na API
- Testes de unidade e funcionalidades
- Utilizar API do Google Maps para mostrar os pontos da loja no mapa
- Criar tela para organização dos produtos, podendo arrastar e soltar os produtos posicionando os mesmos na prateleira (Objetivo: instruir o lojista, destacando os produtos que tem mais saída)

TECNOLOGIAS UTILIZADAS

- .NET API C#
- Testes com NUnit (opcional)
- Banco de dados MySQL
- Rodar a aplicação no localhost
- Criar um front-end do projeto em Angular

SUGESTÃO DE ORGANIZAÇÃO DO PROJETO

Elaboração do Kanban com definição dos entregáveis



1. Elaboração do kanban (sugestão de utilização: Trello, Jira, etc)
2. Criação do backlog
3. Detalhamento descritivo das tarefas da squad dentro dos seus cards (e não apenas com títulos genéricos no card)
4. Formatação do kanban padrão "to do, doing, done"
5. Definição de data de entrega das tarefas nos cards
6. Definição de responsável pelo card ou checklist de completude
7. Priorização dos cards (ex: tags com cores para maior relevância ou com títulos descritivos para nível de importância na priorização)

CRITÉRIOS DE AVALIAÇÃO

Itens mínimos para entrega

- Organização do projeto (Kanban no Trello, Jira ou outra ferramenta)
- Modelagem de dados
- API funcionando para cadastros e consultas
- Login na API
- Front-end em Angular
- Repositório do GIT
- Fluidez na apresentação