

Projeto Fullstack

Situação-Problema

Sua equipe de desenvolvimento foi contratada para criar uma aplicação **Fullstack moderna**, com foco em **segurança, acessibilidade, comunicação e interatividade**.

O cliente deseja uma solução:

- **Robusta e escalável**, preparada para crescer junto com o negócio.
- **Acessível a diferentes perfis de usuários**, incluindo pessoas com deficiência auditiva, visual e de mobilidade reduzida.
- **Segura**, garantindo proteção de dados e confiabilidade no uso.

A aplicação deverá ser desenvolvida com:

- **Frontend**: Angular 19
- **Backend**: Node.js com Express (padrão MVC)
- **Banco de Dados**: MySQL (relacional)
- **Segurança**: uso obrigatório de bcrypt, dotenv e JWT.
- **Proteção de Rotas**: Angular Guard (separação de áreas ADM e USER).

Requisitos Técnicos

1. Banco de Dados (MySQL)

- Criar **mínimo 3 tabelas relacionadas** (exemplo: Usuários, Produtos/Serviços, Pedidos).
- Entregar **modelo conceitual e lógico** (DER).
- Utilizar **chaves primárias e estrangeiras** corretamente.
- Garantir relacionamentos adequados (1:N, N:N quando necessário).

2. Backend (Node.js + Express)

- Estruturação no padrão **MVC** (Model, View, Controller).
- Uso de **arquivo .env** para variáveis sensíveis.
- Controllers com **CRUD completo** para as tabelas.
- Implementação de **login seguro** com bcrypt (criptografia) e JWT (token de autenticação).
- **CORS habilitado** para comunicação com o Angular.

3. Frontend (Angular 19)

- Implementar **rotas com guard**, separando área **ADM** e **USER**.
- Criar **mínimo de 4 páginas**, além de **Home** e **Login**, contendo:
 - Carrossel de imagens.
 - Cards dinâmicos.
 - Menus interativos.
 - Rodapé com redes sociais e informações.
- Implementar **PWA (Progressive Web App)** para instalação no celular.

4. Acessibilidade e Inclusão

Implementar **mínimo 3 recursos** entre:

- Intérprete de Libras (VLibras).
- Leitor de tela embutido.
- Controle por voz (Web Speech API).
- Modo alto contraste e fonte ampliada.

5. Comunicação e Interatividade

Implementar **mínimo 2 recursos** entre:

- Chatbot inteligente (ex: ChatGPT API, Dialogflow, IBM Watson).
- Atendimento humano via chat (Tawk.to, JivoChat).

- FAQ dinâmico com busca inteligente.
 - Botão de contato via WhatsApp/Telegram.
-

6. Segurança e Confiabilidade

- Implementar **autenticação em dois fatores (2FA)**.
 - Disponibilizar **login com redes sociais (OAuth2)**.
 - Garantir **acesso via HTTPS** (SSL configurado no projeto).
-

7. Testes Automatizados

- Testes de **persistência e carga** com **K6**.
 - Testes de **interface e rotas** com **Cypress**.
 - Geração de **relatórios automáticos** com os resultados.
-

8. Relatórios e Dashboards

Criar um **dashboard administrativo** exibindo:

- Usuários cadastrados.
 - Atividades recentes.
 - Gráficos (ex: vendas, acessos, relatórios de uso).
-

9. Recursos Avançados (mínimo 2 obrigatórios)

- Dark Mode (modo escuro).
 - Tradução automática de idiomas (i18n).
 - Leitura em voz alta (Text-to-Speech).
 - Pesquisa avançada com filtros inteligentes.
 - Recomendação personalizada baseada no perfil do usuário.
 - Gamificação (pontos, conquistas, recompensas).
-

Documentação Final

O projeto deverá ser entregue com um **relatório técnico** contendo:

- Modelo Conceitual e Lógico do Banco.
- Descrição do banco e dos endpoints.
- Manual de instalação e execução do sistema.
- Prints e logs dos testes realizados (Cypress e K6).
- Explicação dos recursos de acessibilidade implementados.