

Módulo 9: Projetos Final

Módulo 9: Projetos Final

Objetivo:

Consolidar todos os conceitos e práticas adquiridos ao longo do curso em um projeto completo e funcional, utilizando Node.js, MySQL, e boas práticas de desenvolvimento e documentação.

Escopos de Projetos Sugeridos

Projeto 1: Sistema de Gerenciamento de Eventos

Levantamento de Requisitos

- Cadastro de eventos com nome, data, local e descrição.
- Listagem de eventos com filtros por data e local.
- Cadastro e login de usuários para gestão de eventos.
- Restrição de acesso: somente organizadores podem criar/editar eventos.
- Geração de relatórios de eventos.

Definição de Tecnologia e Recursos

- **Back-end:** Node.js com Express.
- **Banco de Dados:** MySQL.
- **Autenticação:** JWT.
- **Interface:** EJS ou React (opcional).
- **Deploy:** Heroku ou Render.

Documentação Técnica

Inclua as seguintes seções:

- **Visão Geral do Sistema:** Descrição dos principais objetivos e funcionalidades do sistema.
- **Diagrama de Arquitetura:** Representação do fluxo de dados entre cliente, servidor e banco de dados.
- **Endpoints da API:** Lista de endpoints criados, com descrição, parâmetros e exemplos de requisições/respostas.

Documentação de Testes

- **Planos de Testes:** Lista de casos de teste, incluindo entradas, saídas esperadas e resultados obtidos.
 - **Ferramentas:** Testes de API com Postman ou Insomnia.
-

Projeto 2: Sistema de Controle de Estoque

Levantamento de Requisitos

- Cadastro de produtos com nome, quantidade, preço e categoria.
- Controle de entrada e saída de estoque.
- Relatórios de produtos mais vendidos e estoque baixo.
- Gestão de usuários para controle de acesso.

Definição de Tecnologia e Recursos

- **Back-end:** Node.js com Express.
- **Banco de Dados:** MySQL.
- **Autenticação:** JWT com roles (admin/usuário comum).
- **Interface:** EJS ou uma API para consumo externo.
- **Deploy:** Heroku ou DigitalOcean.

Documentação Técnica

- **Descrição do Sistema:** Objetivos e principais funcionalidades.
- **Diagrama de Relacionamento de Dados:** Estrutura das tabelas do banco de dados e suas relações.
- **Detalhamento de Endpoints:** Tabelas com métodos HTTP, descrições e exemplos.

Documentação de Testes

- **Testes Unitários:** Utilizando Jest para funções críticas.
- **Testes de Integração:** Verificar interação entre back-end e banco de dados.
- **Testes Funcionais:** Garantir que as funcionalidades atendam os requisitos definidos.

Projeto 3: Plataforma de Feedback de Produtos

Levantamento de Requisitos

- Cadastro de produtos com nome, descrição e categoria.
- Sistema de feedback com avaliações e comentários de usuários.
- Exibição de avaliações médias para cada produto.
- Moderação de feedbacks pelos administradores.

Definição de Tecnologia e Recursos

- **Back-end:** Node.js com Express.
- **Banco de Dados:** MySQL.
- **Autenticação:** JWT.
- **Interface:** EJS ou Vue.js.
- **Deploy:** Render ou AWS.

Documentação Técnica

- **Especificação do Sistema:** Objetivo, escopo e visão geral.
- **Modelagem de Dados:** Tabelas e relações (ex.: produtos, usuários, feedbacks).
- **Descrição dos Endpoints:** Inclua exemplos e descrições claras para cada endpoint.

Documentação de Testes

- **Testes de Segurança:** Validação de autenticação e autorização.
 - **Testes de Desempenho:** Análise do tempo de resposta dos endpoints mais usados.
 - **Testes Manuais:** Execução de fluxos completos no Postman.
-

Modelo de Documentação do Projeto

1. Introdução

- **Nome do Projeto:**
- **Descrição Geral:** Resumo dos objetivos e escopo do projeto.
- **Tecnologias Utilizadas:** Ferramentas e linguagens usadas no desenvolvimento.

2. Requisitos Funcionais

- Liste as funcionalidades essenciais do sistema.
 - Ex.: "Usuários podem cadastrar produtos."
 - Ex.: "Administradores podem gerar relatórios."

3. Modelagem de Dados

- Inclua um diagrama ER ou descrição textual das tabelas e suas relações.

4. Estrutura do Projeto

Organização de Pastas: Descreva como o projeto está organizado.

/controllers

/models

/routes

/views

-
- **Dependências:** Liste as principais bibliotecas usadas no projeto.

5. Endpoints da API

- Para cada endpoint, descreva:
 - Método HTTP.
 - URL.
 - Parâmetros e corpo da requisição.

- Exemplo de resposta.

6. Configuração do Ambiente

- Passos para rodar o projeto localmente:
 1. Clone o repositório.
 2. Instale as dependências com `npm install`.
 3. Configure as variáveis de ambiente.
 4. Inicie o servidor com `npm start`.

7. Documentação de Testes

- **Planos de Teste:** Inclua descrições das verificações realizadas.
- **Ferramentas Usadas:** Liste as ferramentas empregadas, como Postman, Jest ou Swagger.

8. Deploy

- **Plataforma Utilizada:** Descreva o ambiente de produção (Heroku, Render, AWS).
- **URL de Produção:** Inclua o link para acesso à aplicação.

9. Conclusão

- Resuma os principais aprendizados e desafios enfrentados.
-

Entrega Final

- Cada aluno ou grupo deve:
 1. Criar um repositório no GitHub e adicionar os professores como colaboradores.
 2. Utilizar branches auxiliares para o desenvolvimento.
 3. Criar pull requests e marcar os professores para revisão de código.
 4. Submeter a URL do repositório e a aplicação em produção para avaliação.