

# Teste Técnico Front-end

## Projeto: Área do Aluno (Plataforma de Cursos)

### 1. Contexto

O objetivo deste desafio é avaliar sua capacidade de construir uma interface moderna e bem estruturada, com foco em:

- Arquitetura de front-end
- Organização de projeto
- Qualidade de código
- UI/UX
- Boas práticas de versionamento (Git)

Você deverá desenvolver uma **Área do Aluno**, onde o usuário pode visualizar seus cursos e assistir aulas.

Importante:

- Não é necessário backend real.
- Todos os dados devem ser mockados (JSON/local).
- O foco está na estrutura, organização e funcionamento no front-end.

### 2. Stack Obrigatória

- Nuxt 4
- Tailwind CSS
- Gerenciador de pacotes: pnpm ou npm
- SSG obrigatório (Static Site Generation)

O projeto deve funcionar corretamente como site estático gerado pelo Nuxt.

### 3. Entrega Obrigatória

- Repositório público no GitHub
- Enviar o link do repositório para avaliação
- Projeto rodando em modo dev
- Build estático funcionando via generate (SSG)

Observação importante:

O histórico do Git será avaliado. Evite enviar apenas um commit final.

## 4. Requisitos do Produto

### 4.1. Página "Meus Cursos" (Listagem)

Criar uma tela que liste os cursos do usuário contendo:

- Card do curso com:
  - Título
  - Descrição curta
  - Imagem de capa
  - Progresso (%)
  - Quantidade de aulas
- Ações:
  - "Continuar"
  - "Ver detalhes"

Filtros:

- Busca por título  
ou
- Filtro por status:
  - Em andamento
  - Concluído
  - Não iniciado

Os dados devem ser mockados (ex.: data/, content/, server/api mockado ou estrutura equivalente organizada).

### 4.2. Página do Curso (Detalhe + Aula)

Ao acessar um curso, a tela deve conter:

- Player de vídeo em destaque (YouTube embed ou player simples)
- Sidebar ou lista de aulas
- Troca de conteúdo do player ao clicar em uma aula
- Indicação visual:
  - Aula selecionada
  - Aulas concluídas (mock)

Conteúdo mínimo do curso:

- Nome do curso
- Descrição
- Pelo menos 8 aulas
- Duração (mock)

- Status das aulas

### 4.3. Materiais Extras

Adicionar suporte a pelo menos um material complementar:

- Link para PDF (abrindo em nova aba)  
ou
- Seção "Materiais" com links (PDF, docs etc.)

Tudo pode ser mockado.

## 5. Requisitos Técnicos (Importante)

### SSG Obrigatório

O projeto deve:

- Estar configurado para output estático no Nuxt 4
- Gerar corretamente as rotas no modo estático
- Funcionar após o build estático

Obrigatório incluir no README:

- Como rodar o projeto
- Como gerar o build estático (npm run generate ou pnpm generate)
- Como visualizar o resultado gerado

## 6. Organização e Arquitetura

Esperamos encontrar:

- Componentes reutilizáveis (exemplo):
  - CourseCard
  - LessonList
  - VideoPlayer
  - ProgressBar
- Separação clara entre:
  - Dados mockados
  - Interface
  - Composables (se aplicável)
- Estrutura de pastas coerente:

- components/
- pages/
- composables/
- types/
- data/
- utils/

## 7. Qualidade

- TypeScript recomendado
- Código legível e organizado
- Responsabilidade única por componente
- Consistência de padrões

## 8. UI/UX

- Responsivo (desktop e mobile)
- Estados de interface:
  - Empty state
  - Skeleton ou loading simulado
- Capricho visual utilizando Tailwind CSS

## 9. Git (Será Avaliado)

Critérios:

- Commits pequenos, claros e bem descritos
- Histórico mostrando construção incremental
- Uso de branches e PR é bem-vindo (não obrigatório)
- Evitar envio de commit único final

## 10. Pontos Extras (Não Obrigatório)

PWA

Implementação básica de PWA gera pontos adicionais:

- Manifest configurado
- Ícones
- Cache básico/offline funcional

- README explicando como testar

Não é obrigatório, mas agrega valor.

## 11. Critérios de Avaliação

- Organização e arquitetura do projeto
- Componentização e reuso
- Qualidade do código
- UI/UX e atenção a detalhes
- Funcionamento correto do SSG
- Histórico e organização do Git
- Extras implementados (se houver)

## 12. Forma de Entrega

- Repositório público no GitHub
- Envio do link do repositório
- README contendo:
  - Como rodar
  - Como gerar versão estática
  - Breve explicação das decisões técnicas
  - Extras implementados (se aplicável)