

Plano Técnico Detalhado - Estúdio IA de Vídeos

MVP em 60 dias para o mercado brasileiro de treinamentos NRs

1. VISÃO GERAL TÉCNICA

1.1 Arquitetura do Sistema

Frontend (NextJS 14 + PWA)

├─ Interface Web Responsiva

├─ Editor Drag-and-Drop

├─ Visualizador de Vídeo

└─ Gerenciador de Assets

Backend (Node.js + Express)

├─ API Gateway

├─ Serviços de IA

├─ Processamento de Vídeo

└─ Gerenciamento de Usuários

Integrações IA

├─ Hugging Face (Geração de Vídeo)

├─ Google TTS (Narração)

├─ RouteLLM (Otimização de Custos)

└─ APIs de Avatar 3D

Infraestrutura

├─ Vercel (Deploy Frontend)

├─ Railway/Render (Backend)

├─ Cloudinary (Assets)

└─ PostgreSQL (Dados)

1.2 Tech Stack Definido

Frontend:

- NextJS 14 com App Router
- PWA com next-pwa
- TailwindCSS + Shadcn/ui
- React Flow (Editor drag-and-drop)
- Zustand (Estado global)

Backend:

- Node.js + Express
- Prisma ORM + PostgreSQL
- JWT Authentication
- Multer (Upload de arquivos)

IA e Processamento:

- Hugging Face Diffusers (LTX-Video, HunyuanVideo)
- Google Cloud TTS
- RouteLLM para otimização
- FFmpeg para processamento de vídeo

Infraestrutura:

- Vercel (Frontend + Edge Functions)
 - Railway (Backend + Database)
 - Cloudinary (CDN + Transformações)
-

2. BREAKDOWN DE SPRINTS (60 DIAS)

FASE 1: PLANEJAMENTO E SETUP (Dias 1-18)

Sprint 1 (Dias 1-6): Setup e Arquitetura

Objetivos:

- Setup completo do ambiente de desenvolvimento
- Configuração da arquitetura base
- Definição de padrões de código

Tarefas Técnicas:

- [] Setup NextJS 14 com TypeScript
- [] Configuração PWA com next-pwa
- [] Setup backend Node.js + Express
- [] Configuração Prisma + PostgreSQL
- [] Setup CI/CD (GitHub Actions)
- [] Configuração ESLint + Prettier
- [] Docker containers para desenvolvimento

Entregáveis:

- Repositórios configurados
- Ambiente de desenvolvimento funcional
- Pipeline CI/CD básico

Sprint 2 (Dias 7-12): Autenticação e UI Base

Objetivos:

- Sistema de autenticação completo
- Interface base responsiva
- Componentes fundamentais

Tarefas Técnicas:

- [] Sistema de auth JWT
- [] Páginas de login/registro
- [] Layout responsivo base
- [] Componentes UI (Shadcn/ui)
- [] Sistema de roteamento
- [] Middleware de autenticação

Entregáveis:

- Sistema de auth funcional
- Interface base responsiva
- Componentes reutilizáveis

Sprint 3 (Dias 13-18): Integrações IA Base**Objetivos:**

- Configuração das APIs de IA
- Testes de integração
- Sistema de fallback

Tarefas Técnicas:

- [] Setup Hugging Face API
- [] Configuração Google TTS
- [] Integração RouteLLM
- [] Sistema de fallback para APIs
- [] Testes de latência e custos
- [] Documentação das integrações

Entregáveis:

- APIs de IA configuradas
- Sistema de fallback funcional
- Documentação técnica

FASE 2: DESENVOLVIMENTO CORE (Dias 19-42)**Sprint 4 (Dias 19-24): Editor Drag-and-Drop****Objetivos:**

- Editor visual funcional
- Sistema de timeline
- Manipulação de elementos

Tarefas Técnicas:

- [] Implementação React Flow
- [] Sistema de timeline de vídeo
- [] Drag-and-drop de elementos
- [] Preview em tempo real
- [] Sistema de undo/redo
- [] Salvamento automático

Entregáveis:

- Editor drag-and-drop funcional
- Timeline interativa
- Sistema de preview

Sprint 5 (Dias 25-30): Geração de Vídeo IA**Objetivos:**

- Geração de vídeos com IA
- Avatares 3D falantes
- Integração com TTS

Tarefas Técnicas:

- [] Integração LTX-Video/HunyuanVideo
- [] Sistema de avatares 3D
- [] Sincronização TTS + Avatar
- [] Processamento de vídeo (FFmpeg)
- [] Sistema de filas para renderização
- [] Otimização de performance

Entregáveis:

- Geração de vídeo funcional
- Avatares 3D integrados
- Sistema de renderização

Sprint 6 (Dias 31-36): Conversão PPTX**Objetivos:**

- Upload e processamento PPTX
- Conversão para vídeo
- Narração automática

Tarefas Técnicas:

- [] Parser de arquivos PPTX
- [] Extração de conteúdo e imagens
- [] Conversão slides para frames
- [] Geração de narração automática
- [] Sincronização áudio/vídeo
- [] Sistema de templates

Entregáveis:

- Conversão PPTX funcional
- Narração automática
- Templates de vídeo

Sprint 7 (Dias 37-42): Otimizações e Qualidade**Objetivos:**

- Otimização de performance
- Qualidade de vídeo
- Sistema de cache

Tarefas Técnicas:

- [] Otimização de renderização
- [] Sistema de cache inteligente
- [] Compressão de vídeos
- [] Melhoria na qualidade dos avatares
- [] Testes de carga
- [] Monitoramento de performance

Entregáveis:

- Sistema otimizado
- Cache implementado
- Métricas de performance

FASE 3: LANÇAMENTO E REFINAMENTO (Dias 43-60)

Sprint 8 (Dias 43-48): Funcionalidades Finais

Objetivos:

- Exportação de vídeos
- Compartilhamento
- Funcionalidades de usuário

Tarefas Técnicas:

- [] Sistema de exportação MP4/HD
- [] Compartilhamento de vídeos
- [] Biblioteca de assets
- [] Sistema de templates
- [] Histórico de projetos
- [] Configurações de usuário

Entregáveis:

- Exportação funcional
- Sistema de compartilhamento
- Biblioteca de assets

Sprint 9 (Dias 49-54): Testes e Polimento

Objetivos:

- Testes completos
- Correção de bugs
- Polimento da UX

Tarefas Técnicas:

- [] Testes end-to-end
- [] Testes de usabilidade
- [] Correção de bugs críticos
- [] Polimento da interface
- [] Otimização mobile
- [] Documentação do usuário

Entregáveis:

- Sistema testado
- Bugs críticos corrigidos
- UX polida

Sprint 10 (Dias 55-60): Deploy e Lançamento

Objetivos:

- Deploy em produção
- Monitoramento
- Lançamento beta

Tarefas Técnicas:

- [] Deploy produção (Vercel + Railway)
- [] Configuração monitoramento
- [] Setup analytics
- [] Testes de produção

- [] Documentação final
- [] Lançamento beta restrito

Entregáveis:

- Sistema em produção
- Monitoramento ativo
- Beta lançado

3. ESPECIFICAÇÕES TÉCNICAS DETALHADAS

3.1 APIs de IA e Custos Estimados

Hugging Face (Geração de Vídeo)

- **Modelos:** LTX-Video, HunyuanVideo
- **Custo:** ~\$0.10-0.50 por vídeo (1-2 min)
- **Limites:** Rate limiting por API key
- **Fallback:** Múltiplos modelos disponíveis

Google TTS

- **Custo:** \$4/milhão caracteres (Standard), \$16/milhão (WaveNet)
- **Free Tier:** 4M caracteres/mês (Standard), 1M (WaveNet)
- **Idiomas:** PT-BR com sotaques regionais
- **Otimização:** Cache de áudios gerados

RouteLLM

- **Função:** Otimização de custos de IA
- **Economia:** Até 85% de redução
- **Integração:** Drop-in replacement para OpenAI
- **Configuração:** Threshold calibrado para qualidade

3.2 Arquitetura de Dados

```
-- Schema Principal
Users (id, email, name, plan, created_at)
Projects (id, user_id, name, data, status, created_at)
Videos (id, project_id, url, duration, status, metadata)
Templates (id, name, data, category, is_public)
Assets (id, user_id, type, url, metadata)
```

3.3 Sistema de Filas e Processamento

```
// Sistema de filas com Bull
const videoQueue = new Queue('video processing');

videoQueue.process('generate-video', async (job) => {
  const { projectData, userId } = job.data;

  // 1. Gerar vídeo com IA
  const videoUrl = await generateVideoWithAI(projectData);

  // 2. Processar com FFmpeg
  const processedUrl = await processVideo(videoUrl);

  // 3. Upload para CDN
  const finalUrl = await uploadToCDN(processedUrl);

  return { videoUrl: finalUrl };
});
```

3.4 PWA Configuration

```
// next.config.js
const withPWA = require('next-pwa')({
  dest: 'public',
  register: true,
  skipWaiting: true,
  disable: process.env.NODE_ENV === 'development'
});

module.exports = withPWA({
  // NextJS config
});
```

4. CRONOGRAMA DE RECURSOS

4.1 Equipe Recomendada

- **1 Tech Lead/Fullstack** (60 dias)
- **1 Frontend Developer** (42 dias - a partir do Sprint 2)
- **1 Backend Developer** (36 dias - a partir do Sprint 3)
- **1 DevOps/QA** (18 dias - últimos sprints)

4.2 Custos Estimados (60 dias)

Desenvolvimento

- Equipe: \$15,000 - \$25,000
- Infraestrutura: \$500 - \$1,000
- APIs de IA: \$1,000 - \$2,000
- **Total:** \$16,500 - \$28,000

Custos Operacionais (pós-lançamento)

- Infraestrutura: \$200-500/mês

- APIs de IA: \$500-2000/mês (baseado no uso)
- CDN: \$100-300/mês

5. RISCOS E MITIGAÇÕES

5.1 Riscos Técnicos

Risco	Probabilidade	Impacto	Mitigação
APIs de IA instáveis	Alta	Alto	Sistema de fallback, múltiplos providers
Performance de renderização	Média	Alto	Otimização progressiva, cache inteligente
Custos de IA elevados	Média	Médio	RouteLLM, limites por usuário
Complexidade do editor	Alta	Médio	MVP simplificado, iteração baseada em feedback

5.2 Plano de Contingência

- **Semana 3:** Avaliação de progresso, ajuste de escopo se necessário
- **Semana 6:** Checkpoint crítico, decisão sobre funcionalidades
- **Semana 8:** Freeze de features, foco em estabilidade

6. MÉTRICAS DE SUCESSO

6.1 Métricas Técnicas

- **Performance:** Renderização < 10s para vídeos 1-2min
- **Disponibilidade:** 99.5% uptime
- **Qualidade:** Taxa de erro < 5% na geração
- **Escalabilidade:** Suporte a 500 usuários simultâneos

6.2 Métricas de Produto

- **Adoção:** 100 usuários ativos no primeiro mês
- **Engajamento:** 70% dos usuários criam pelo menos 1 vídeo
- **Retenção:** 40% retornam na segunda semana
- **Conversão:** 20% dos usuários gratuitos migram para pago

7. PRÓXIMOS PASSOS IMEDIATOS

Semana 1 (Dias 1-7)

1. **Setup do ambiente de desenvolvimento**
2. **Configuração dos repositórios**
3. **Definição da arquitetura detalhada**
4. **Setup das contas de API (Hugging Face, Google Cloud)**
5. **Criação do backlog detalhado**

Preparação para Sprint 1

- ☐ Criar repositórios GitHub
- ☐ Setup ambiente local
- ☐ Configurar contas de API
- ☐ Definir padrões de código
- ☐ Preparar ambiente de CI/CD

Documento criado em: 30 de Agosto de 2025

Versão: 1.0

Próxima revisão: Sprint 3 (Dia 18)