

PLANO ESTRUTURAL DE EXECUÇÃO — RAWN PRO

1. Fundação do Ambiente

Objetivo: preparar um ambiente de engenharia robusto, limpo e replicável.

Ações:

- Instalar **VS Code 1.105** e habilitar módulos de IA (Copilot + MCP).
 - Criar repositório rawn-pro no GitHub com branches main, dev, feature/*.
 - Configurar .editorconfig, .gitignore, e scripts automatizados no package.json.
 - Instalar extensões padrão de engenharia:
 - *ESLint, Prettier, GitLens, Error Lens, REST Client, Docker, Thunder Client, Copilot Chat.*
 - Integrar terminal com perfis Node / Python via nova API de terminal persistente.
-

2. Inicialização Técnica do Projeto

Objetivo: criar a base Next.js moderna e escalável.

Ações:

- `npx create-next-app@latest .` (TypeScript, ESLint e App Router).
 - Configurar estrutura limpa:
 - `src/`
 - `app/`
 - `components/`
 - `hooks/`
 - `services/`
 - `store/`
 - `public/`
 - Habilitar **direnv** e variáveis de ambiente seguras.
 - Testar build e `npm run dev` com hot reload ativo.
-

3. Design System RAWN

Objetivo: implementar o visual idêntico ao WhatsApp, conforme PDFs de design.

Ações:

- Criar tokens de cor, tipografia e espaçamento.
 - Usar Tailwind + ShadCN/UI + Framer Motion.
 - Criar biblioteca local @rawn/ui com componentes reutilizáveis.
 - Reproduzir interações idênticas (chat bubbles, listas, status).
 - Garantir acessibilidade (ARIA, contraste, foco visível).
-

4. Núcleo de Autenticação e Perfil

Objetivo: implementar login e gerenciamento de usuário.

Ações:

- Tela de login → tela inicial → edição de perfil (PDF “EDITAR PERFIL — RAWN PRO”).
 - JWT + Refresh Token.
 - Mock API local para testes.
 - Integração posterior com backend real.
 - Armazenar sessão em cookies seguros e persistentes (NextAuth).
-

5. Núcleo de Mensageria (RAWN Engine)

Objetivo: construir o sistema de chat em tempo real.

Ações:

- WebSocket + Redis para streaming de mensagens.
 - Estrutura modular:
 - /chat/core (eventos e handlers)
 - /chat/ui (componentes visuais)
 - /chat/ai (orquestração do Copilot/RAWN AI)
 - Estados: online/offline, typing, recebimento, leitura.
 - Envio de mídia e documentos.
-

6. Integração com IA e System Prompt

Objetivo: conectar o Copilot/RAWN AI com a lógica definida no “SYSTEM PROMPT.pdf”.

Ações:

- Criar camada src/ai/agent.ts para chamadas a OpenAI API.
 - Implementar prompts estruturais do RAWN (engine contextual + persona).
 - Adotar “Chain of Thought View” do VS Code para debug de IA.
 - Criar painel lateral “RAWN AI Console” para logs e inspeções.
-

7. Políticas, Termos e Segurança

Objetivo: aplicar conformidade e proteção de dados.

Ações:

- Páginas estáticas para Política de Privacidade e Termos de Uso.
 - Consentimento explícito de cookies e dados.
 - Encriptação local (AES-256) para dados sensíveis.
 - Linter de segurança + auditoria npm (npm audit fix).
-

8. Observabilidade e Qualidade

Objetivo: garantir rastreabilidade, performance e estabilidade.

Ações:

- ESLint + Prettier + TypeCheck + Husky + Commitlint.
 - Testes unitários com Jest + Playwright (E2E).
 - Logs estruturados (pino) e métricas com Sentry.
 - Liveness e readiness endpoints.
-

9. Performance e Deploy

Objetivo: otimizar build e entrega contínua.

Ações:

- Pipeline CI/CD (GitHub Actions).
- Minificação, cache e edge rendering.

- Deploy na **Vercel** (prod) e **Railway** (dev).
 - Análise de bundle (next build --analyze).
-

10. Beta Fechado e Iterações

Objetivo: testar e refinar com usuários reais.

Ações:

- Testes de UX baseados em “Pesquisa de mercado e benchmark”.
 - Feedbacks armazenados no Firebase.
 - Logs de uso e gravação de sessão (Hotjar).
 - Sprint semanal de ajustes.
-

11. Go-Live e Pós-Lançamento

Objetivo: publicar, monitorar e escalar.

Ações:

- Revisão de conformidade e checklist final.
 - Tag 1.0.0 e changelog.
 - Ativação de monitoramento em tempo real (API + IA).
 - Reuniões quinzenais de performance e updates.
-

12. Estrutura de Progresso (Hermética)

Cada etapa é **bloqueante** para a próxima.

Nada é iniciado sem o término e validação da anterior.

Todo o fluxo é auditável e versionado.