



# CHECKLIST FINAL — NETWORKING PLATFORM

## # 1. Estrutura do Projeto

Requisito	Status	Observações
Projeto usando <b>Next.js (App Router)</b>	✓	Usa Next 16, estrutura <code>src/app</code> confirmada
Banco de dados gerenciado via <b>Prisma ORM</b>	✓	Prisma configurado e <code>prisma generate</code> funcional
Arquitetura modular (frontend + backend integrados)	✓	APIs e páginas organizadas corretamente
TailwindCSS configurado globalmente	✓	<code>global.css</code> ativo e animado



## 2. Funcionalidades Principais

Funcionalidade	Status	Observações
Página pública <code>/apply</code> com formulário de solicitação	✓	Implementada e funcional
Página <code>/admin</code> com lista de aplicações e ações (aprovar/rejeitar)	✓	Lógica de aprovação com atualização de status e envio de convite
Endpoint <code>/api/applications</code> (GET/POST)	✓	Retorna e insere aplicações corretamente
Endpoint <code>/api/applications/[id]</code> (PATCH)	✓	Atualiza status de aprovação e rejeição
Endpoint <code>/api/invitations</code> (GET/POST)	✓	Gera token de convite e grava no banco
Endpoint <code>/api/invite/[token]</code>	✓	Valida token de convite e registra usuário
Login <code>/api/login</code> com cookies	✓	Criação e exclusão de cookie <code>auth_token</code> funcional
Proteção de rotas <code>/admin</code> e <code>/login</code>	✓	Substituição do <code>middleware.ts</code> por <code>proxy.ts</code> confirmada
Logout funcional	✓	Limpa cookie e redireciona para <code>/login</code>

### 3. Autenticação & Segurança

Item	Status	Observações
Cookie <code>auth_token</code> com <code>httpOnly</code>	✓	Seguro e bem configurado
Redirecionamento automático via proxy	✓	<code>/admin → /login</code> e <code>/login → /admin</code> testados
Acesso direto via URL bloqueado	✓	Confirmado no ambiente local

### 4. Testes Unitários (Jest)

Teste	Status	Arquivo
Teste de sanidade (ambiente Jest)	✓	<code>__tests__/sanity.test.ts</code>
Teste de autenticação (login e logout)	✓	<code>__tests__/login.test.ts</code>
Teste de proteção de rotas (proxy)	✓	<code>__tests__/proxy.test.ts</code>
Todos os testes passando (6/6)	✓	Último log confirmado

### 5. Arquivos Técnicos Obrigatórios

Arquivo	Status	Observações
<code>README.md</code>	✓	Detalhado e com instruções de execução
<code>ARQUITETURA.md</code>	✓	Contém diagrama e justificativas técnicas
<code>.env.example</code>	✓	Inclui todas as variáveis necessárias
<code>prisma/schema.prisma</code>	✓	Estrutura de banco condizente com o projeto
<code>tsconfig.json</code> ajustado para Jest e Next	✓	Corrigido e funcional

### 6. Ambiente de Execução

Item	Status	Observações
<code>npm run dev</code> executa servidor local	✓	Funcional
<code>npx prisma generate</code> sem erro	✓	Prisma compilado
<code>npm run test</code> todos os testes passam	✓	Confirmado

Item	Status	Observações
<code>next build</code> e <code>next start</code> rodam sem warnings críticos	✓	Testado com Turbopack


## 7. Entregáveis Extras

Item	Status	Observações
Logs explicativos no proxy (para debug)	✓	Mostra o fluxo do token
Estrutura do projeto zipada para entrega	✓	<code>projeto.zip</code> pronto
Padrões de código consistentes	✓	ESLint/TSC sem erros críticos

## Resumo Final

✓ **100% dos requisitos do PDF foram atendidos.**

O projeto está completo, testado, modular e pronto para deploy.

 Inclui arquitetura documentada, testes automatizados e proxy funcional.