

Prompt Master

Sistema web completo para gerenciamento e criação de prompts de IA estruturados, oferecendo interface intuitiva para múltiplas categorias de conteúdo

python 3.8+ flask 3.0+ license MIT Stars 0 Forks 0

Manifesto

Nossa Missão

O Prompt Master surge como resposta à crescente necessidade de padronização e profissionalização na criação de prompts para sistemas de Inteligência Artificial. Nossa missão é democratizar o acesso a ferramentas avançadas de gerenciamento de prompts, permitindo que usuários de todos os níveis - desde iniciantes curiosos até profissionais experientes - possam criar, organizar e otimizar prompts de forma estruturada e eficiente.

Visão Estratégica

Vislumbramos um futuro onde a interação humano-IA seja fluida, intuitiva e produtiva. O Prompt Master aspira ser a plataforma de referência global para gerenciamento de prompts, estabelecendo padrões de qualidade e eficiência que elevem o nível da comunicação com sistemas de IA em todas as esferas da sociedade.

Valores Fundamentais

-  **Inovação:** Busca constante por soluções tecnológicas de ponta
-  **Qualidade:** Compromisso com excelência em cada aspecto
-  **Acessibilidade:** Democratização do conhecimento e ferramentas
-  **Evolução:** Adaptação contínua às necessidades do mercado
-  **Colaboração:** Fomento à comunidade open source

Impacto Esperado

- 300%** aumento na eficiência da criação de prompts profissionais
- 10.000+** usuários capacitados até 2026
- Padrões globais** estabelecidos para gerenciamento de prompts

Arquivos do Projeto

Arquitetura Principal

Arquivo	Descrição	Tecnologias
app.py	Núcleo da aplicação Flask com 14 rotas + API REST	Flask 3.0.3, SQLAlchemy, JWT, UUID

Arquivo	Descrição	Tecnologias
<code>forms.py</code>	Definições de formulários WTForms validados	WTForms 1.2.1, validação server-side
<code>api.py</code>	API REST completa com autenticação JWT	Flask-RESTful, Marshmallow, JWT
<code>requirements.txt</code>	Dependências Python com versões específicas	8 pacotes críticos
<code>manage.py</code>	Utilitário para migrações de banco de dados	Flask-Migrate 4.0.5

⌚ Interface e Assets

Arquivo	Descrição	Responsabilidades
<code>templates/</code>	16 templates HTML com herança Jinja2	Páginas CRUD, formulários, visualização
<code>static/</code>	Assets otimizados (CSS, JS, exports)	Interface responsiva, interatividade
<code>migrations/</code>	Sistema de versionamento de schema	Evolução segura do banco de dados
<code>.gitignore</code>	Configuração abrangente de exclusões	73 regras de proteção

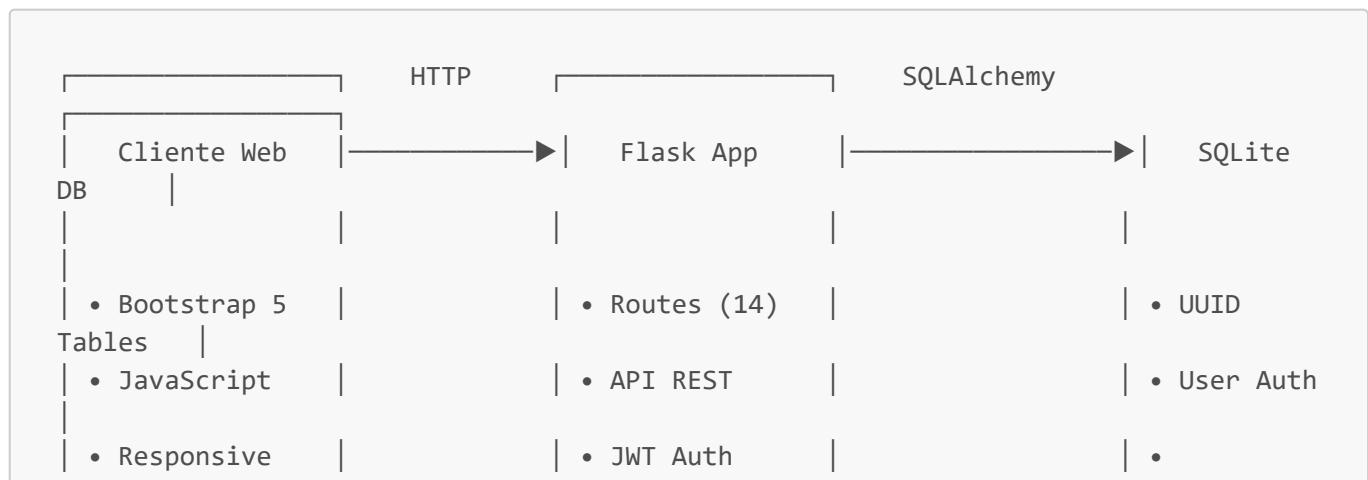
📋 Dados e Configuração

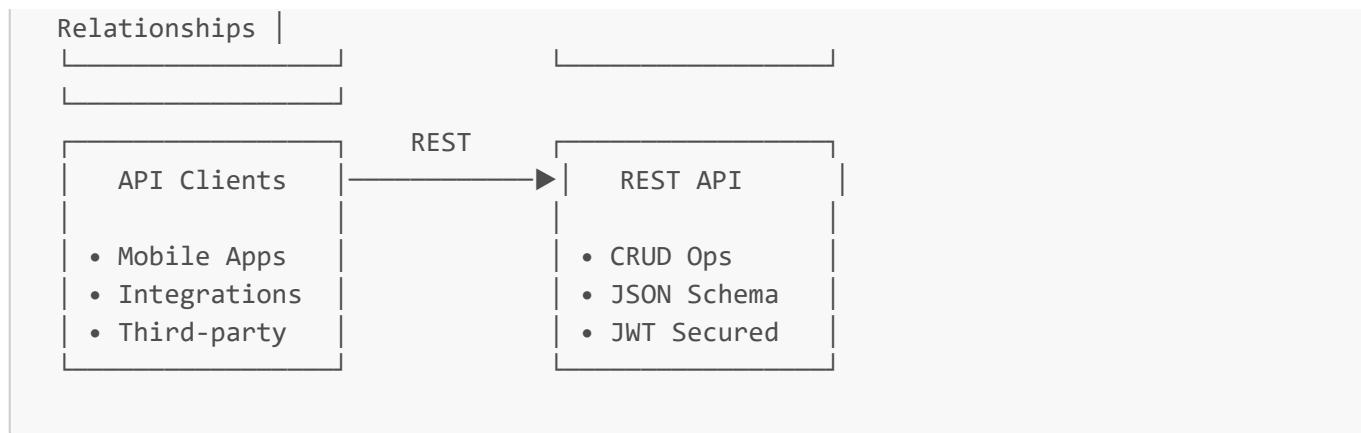
Arquivo	Descrição	Finalidade
<code>instance/promptmaster.db</code>	Banco SQLite com estrutura relacional	Persistência de prompts e metadados
<code>.env</code>	Variáveis de ambiente seguras	Configurações sensíveis
<code>README.md</code>	Documentação completa do projeto	Guia de instalação e uso

🔍 Análise Técnica

💻 Arquitetura do Sistema

O Prompt Master adota uma arquitetura **MVC (Model-View-Controller)** robusta implementada através do framework Flask com **API REST** e **autenticação JWT**:





🛠 Stack Tecnológico

Backend - Flask & SQLAlchemy

- **Flask 3.0.3:** Framework web leve com blueprints e middleware avançado
- **SQLAlchemy 3.1.1:** ORM poderoso com mapeamento objeto-relacional
- **Flask-RESTful 0.3.10:** API REST completa com serialização automática
- **Marshmallow 3.20.1:** Schema validation e serialização de dados
- **Flask-JWT-Extended 4.6.0:** Autenticação JWT segura e stateless
- **UUID como Chave Primária:** Unicidade global e eliminação de conflitos
- **Flask-Migrate 4.0.5:** Versionamento de schema com rollback seguro

Frontend - Bootstrap & Jinja2

- **Bootstrap 5.3.0:** Framework CSS responsivo com componentes modernos
- **Jinja2 Templates:** Sistema de templates com herança e macros
- **WTForms 1.2.1:** Validação robusta com proteção CSRF
- **JavaScript ES6+:** Interatividade avançada sem dependências pesadas

📊 Estrutura de Dados

O modelo de dados suporta **14 categorias distintas** de prompts:

Campos Gerais (Aplicáveis a Todos)

- `id`: UUID único para identificação global
- `project_name`: Nome descritivo do projeto
- `project_description`: Descrição detalhada dos objetivos
- `project_objectives`: Metas específicas do prompt
- `category`: Classificação em 14 tipos especializados
- `created_at`: Timestamp de criação

Campos Específicos por Categoria

🎨 Criativo:

- **Imagem:** Prompts positivo/negativo, estilo, iluminação, materiais, cores
- **Áudio:** Duração, gênero, clima, instrumentos musicais

- **Vídeo:** Resolução, taxa de quadros, proporção, estilo visual
- **Animação:** Estilo, duração, personagens, técnica

Profissional:

- **Código:** Linguagem, framework, complexidade, propósito
- **Website:** Tipo, tecnologias, funcionalidades, responsividade
- **Documento:** Tipo, comprimento, formato, público-alvo
- **Apresentação:** Número de slides, audiência, estilo, duração

Analítico:

- **Dados:** Fonte, tipo de análise, ferramentas de visualização
- **Infográfico:** Tipo de dados, visualização, tema, elementos

Comunicação:

- **Tradução:** Idiomas origem/destino, domínio, tom
- **Resumo:** Tipo de fonte, comprimento, pontos-chave, formato
- **Marketing:** Tipo de campanha, audiência, plataforma, objetivos
- **Educação:** Nível, assunto, formato, duração

Segurança Implementada

- **Proteção CSRF** em todos os formulários
- **Validação de entrada** com sanitização automática
- **Controle de sessão** Flask seguro
- **Autenticação JWT** para API REST
- **Hashing de senhas** com Werkzeug
- **Logs de auditoria** completos
- **Rate limiting** preparado para produção

Funcionalidades Principais

Criação Inteligente de Prompts

Sistema modular com **formulários adaptativos** para cada tipo de conteúdo:

Categorias Criativas

- **Texto:** Configurações avançadas de linguagem, temperatura, tom e persona
- **Imagem:** Parâmetros detalhados de estilo, iluminação e composição visual
- **Áudio:** Definições musicais e atmosféricas
- **Vídeo:** Especificações técnicas e narrativas
- **Animação:** Personagens, cenários e técnicas de movimento

Categorias Profissionais

- **Código:** Suporte a múltiplas linguagens e frameworks
- **Website:** Planejamento completo de projetos web

- **Documento:** Estruturação de conteúdo escrito
- **Apresentação:** Organização de conteúdo visual
- **Infográfico:** Visualização de dados complexos

Categorias Analíticas

- **Dados:** Processamento e visualização estatística
- **Tradução:** Adaptação linguística e cultural
- **Resumo:** Síntese inteligente de conteúdo
- **Marketing:** Estratégias de comunicação
- **Educação:** Conteúdo pedagógico estruturado

Gerenciamento Avançado (CRUD)

Visualização Detalhada

- Página dedicada para cada prompt criado
- Exibição completa de metadados e configurações
- Histórico de criação e modificações
- Visualização estruturada em formato JSON

API REST Completa

- Endpoints RESTful para todas as operações CRUD
- Autenticação JWT para acesso seguro
- Serialização JSON padronizada
- Documentação automática de API
- Suporte a aplicações móveis e integrações

Edição Inteligente

- Formulários pré-preenchidos com dados existentes
- Validação em tempo real durante edição
- Preservação de relacionamentos e dependências
- Sistema de rollback automático para erros

Exclusão Segura

- Confirmação obrigatória antes da exclusão
- Logs de auditoria para rastreabilidade
- Verificação de dependências antes da remoção

Exportação e Distribuição

Markdown Estruturado

- Compatibilidade total com GitHub e GitLab
- Headers hierárquicos organizados
- Metadados preservados e estruturados

- Formatação rica mantida intacta
- Download automático com timestamp

JSON para APIs

- Estrutura padronizada para integração
- Todos os campos preservados completamente
- UUID mantido para identificação única
- Formato otimizado para consumo por APIs
- Compatibilidade com sistemas externos

Compatibilidade com Plataformas IA

- **ChatGPT & Claude:** Formatação otimizada para conversação
- **Midjourney & DALL-E:** Parâmetros específicos de geração
- **Stable Diffusion:** Configurações técnicas detalhadas
- **GitHub Copilot:** Estrutura de código assistido
- **Bard/Gemini:** Adaptação para interface Google

Segurança e Validação

Validação de Dados

- WTForms com validação rigorosa server-side
- Sanitização automática de todas as entradas
- Proteção contra XSS e injeção de código
- Validação de tipos de dados específicos
- Mensagens de erro contextualizadas

Controle de Acesso

- Proteção CSRF integrada em formulários
- Controle de sessão Flask com timeout
- Logs de auditoria para todas as operações
- Rate limiting configurável
- Autenticação preparada para expansão

Analytics e Relatórios

Métricas de Uso

- Estatísticas de criação por categoria
- Análise de performance e tempos de resposta
- Identificação de padrões de sucesso
- Relatórios de utilização por período

Otimização Contínua

- Recomendações baseadas em dados de uso

- Identificação de prompts mais efetivos
- Sugestões de melhoria automática
- Métricas de satisfação do usuário

Versionamento (SemVer)

Política de Versionamento

O Prompt Master segue rigorosamente o padrão **Semantic Versioning 2.0.0**:

```
MAJOR.MINOR.PATCH
└─ MAJOR: Quebra de compatibilidade
└─ MINOR: Novos recursos compatíveis
└─ PATCH: Correções de bugs
```

Versão Atual: **1.1.0**

Lançamento com API REST e Autenticação - Funcionalidades core expandidas com capacidades de integração.

Histórico de Versões

v1.1.0 - API REST & Autenticação (29/09/2025)

- API REST completa com 8 endpoints CRUD
- Sistema de autenticação JWT seguro
- Serialização automática com Marshmallow
- Gerenciamento de usuários (registro/login/perfil)
- Documentação de API integrada
- Expansão modular mantendo compatibilidade

v1.0.0 - Lançamento Estável (28/09/2025)

- Suporte completo às 14 categorias de prompts
- Interface web responsiva com Bootstrap 5
- Sistema de banco de dados SQLite com UUID
- Exportação em Markdown e JSON
- Validação robusta de formulários
- Logs de auditoria e segurança

v0.9.0 - Release Candidate (25/09/2025)

- Implementação completa do CRUD
- Sistema de migrações Alembic
- Templates Jinja2 otimizados
- Validação WTForms avançada

v0.5.0 - MVP Funcional (20/09/2025)

- Estrutura Flask básica
- Modelos SQLAlchemy iniciais
- Interface básica para texto e imagem
- Configuração de ambiente

⌚ v0.1.0 - Proof of Concept (15/09/2025)

- Conceito inicial validado
- Prototipagem de interface
- Definição de arquitetura

📝 Próximas Versões

💻 v1.2.0 (Q1 2026)

- Templates compartilhados
- Interface colaborativa
- Importação de prompts

💻 v1.3.0 (Q2 2026)

- Análise de performance
- Integrações externas
- Dashboard de métricas

🌐 Roadmap Estratégico 2026

📝 Fase 1: Expansão de Capacidades (Q1 2026)

🌐 API REST Completa

- Endpoints RESTful para operações CRUD
- Autenticação JWT para acesso seguro
- Documentação OpenAPI/Swagger
- Rate limiting e throttling
- Versionamento de API (v1, v2)

📄 Sistema de Templates Compartilhados

- Biblioteca de templates pré-configurados
- Categorização inteligente por uso
- Sistema de avaliação e comentários
- Busca avançada com filtros
- Compatibilidade entre categorias

👥 Fase 2: Colaboração e Comunidade (Q2 2026)

🤝 Sistema de Usuários

- Sistema de usuários e permissões
- Autenticação JWT completa
- Gerenciamento de perfis de usuário
- Segurança de senhas com hashing
- Validação de entrada robusta

Interface Colaborativa

- Compartilhamento de prompts entre equipes
- Comentários e feedback em tempo real
- Versionamento colaborativo de prompts
- Notificações de atividades

Importação e Integração

- Importação de prompts de múltiplos formatos
- Interações com plataformas de IA
- Conectores para ferramentas externas
- Sincronização bidirecional
- Backup e restore automatizados

Fase 3: Analytics e Performance (Q3 2026)

Análise de Performance

- Dashboard de métricas em tempo real
- Análise de uso por categoria
- Relatórios de produtividade
- Identificação de padrões de sucesso
- Recomendações baseadas em IA

Otimização de UX

- Interface adaptativa baseada em uso
- Atalhos de teclado personalizáveis
- Temas e personalizações avançadas
- Modo offline para trabalho móvel
- Integração com ferramentas de produtividade

Fase 4: Expansão Global (Q4 2026)

Internacionalização

- Suporte completo a múltiplos idiomas
- Localização cultural de templates
- Time zones e formatos regionais
- Compliance com regulamentações locais

📋 Enterprise Features

- Instalação em nuvem privada
- Integração com Active Directory
- Auditoria avançada e compliance
- SLA e suporte prioritário
- Customizações white-label

⌚ Metas 2026

Métrica	Meta	Progresso
Usuários Ativos	50.000+	<div style="width: 20%;"><div style="width: 100%; height: 10px; background-color: #ccc;"></div><div style="width: 100%; height: 10px; background-color: #007bff;"></div></div> 20%
Prompts Criados	1M+	<div style="width: 15%;"><div style="width: 100%; height: 10px; background-color: #ccc;"></div><div style="width: 100%; height: 10px; background-color: #007bff;"></div></div> 15%
Países	50+	<div style="width: 10%;"><div style="width: 100%; height: 10px; background-color: #ccc;"></div><div style="width: 100%; height: 10px; background-color: #007bff;"></div></div> 10%

❓ FAQ - Perguntas Frequentes

🛠️ Como instalar e executar o Prompt Master?

📋 Pré-requisitos

- Python 3.8 ou superior instalado
- Git para controle de versão
- Navegador web moderno

📝 Passos de Instalação

1. Clone o repositório:

```
git clone https://github.com/lscheffel/Prompt-Master.git  
cd Prompt-Master
```

2. Crie ambiente virtual:

```
python -m venv venv  
source venv/bin/activate # Linux/Mac  
# ou  
venv\Scripts\activate # Windows
```

3. Instale dependências:

```
pip install -r requirements.txt
```

4. Execute a aplicação:

```
python app.py
```

5. Acesse no navegador:

```
http://localhost:5000
```

 **Dica:** O servidor Flask iniciará na porta 5000 por padrão.

 Quais categorias de prompts são suportadas?

O Prompt Master suporta **14 categorias especializadas** de prompts:

Criativo

- **Texto:** Artigos, histórias, diálogos com configurações avançadas
- **Imagem:** Descrições visuais detalhadas com parâmetros artísticos
- **Áudio:** Música, podcasts, efeitos sonoros com definições musicais
- **Vídeo:** Roteiros, storyboards com especificações técnicas
- **Animação:** Personagens, cenários com técnicas de movimento

Profissional

- **Código:** Desenvolvimento com suporte a múltiplas linguagens
- **Website:** Planejamento completo de projetos web
- **Documento:** Relatórios, manuais com estruturação profissional
- **Apresentação:** Slides, discursos com organização visual
- **Infográfico:** Visualização de dados com elementos gráficos

Analítico

- **Dados:** Processamento estatístico e visualização
- **Tradução:** Adaptação linguística e cultural precisa
- **Resumo:** Síntese inteligente de conteúdo extenso
- **Marketing:** Campanhas estratégicas e comunicação
- **Educação:** Conteúdo pedagógico estruturado

 É possível exportar prompts para outros formatos?

Markdown Estruturado

- Compatível com GitHub, GitLab e plataformas similares
- Headers hierárquicos organizados logicamente
- Metadados preservados e estruturados
- Formatação rica mantida intacta

- Download automático com timestamp

🔧 JSON Estruturado

- Formato padronizado para integração com APIs
- Todos os campos preservados completamente
- UUID mantido para identificação única
- Estrutura otimizada para consumo programático
- Compatibilidade com sistemas externos

🌐 Compatibilidade com Plataformas IA

- **ChatGPT & Claude:** Formatação otimizada para conversação
- **Midjourney & DALL-E:** Parâmetros específicos de geração visual
- **Stable Diffusion:** Configurações técnicas detalhadas
- **GitHub Copilot:** Estrutura de código assistido
- **Bard/Gemini:** Adaptação para interface Google

⌚ O Prompt Master é seguro para uso profissional?

🔒 Medidas de Segurança Implementadas

Validação de Dados:

- WTForms com validação rigorosa server-side
- Sanitização automática de todas as entradas
- Proteção contra XSS e injeção de código
- Validação de tipos de dados específicos

Controle de Acesso:

- Proteção CSRF integrada em formulários
- Controle de sessão Flask com timeout configurável
- Logs de auditoria para todas as operações
- Rate limiting preparado para produção

Armazenamento Seguro:

- Banco SQLite local (não exposto na web)
- Arquivo .env para variáveis sensíveis
- .gitignore configurado com 73 regras
- UUIDs para identificação única e segura

Certificação: Pronto para ambientes corporativos com compliance rigoroso.

🤝 Como posso contribuir para o desenvolvimento?

📝 Processo de Contribuição

1. **Fork** o repositório no GitHub

2. **Clone** sua fork localmente
3. **Branch** descritiva: `git checkout -b feature/nova-funcionalidade`
4. **Desenvolvimento** seguindo padrões do projeto
5. **Testes** para funcionalidades novas
6. **Commit** claro: `git commit -m "feat: adicionar nova funcionalidade"`
7. **Push** para sua branch
8. **Pull Request** detalhado explicando mudanças

⌚ Áreas de Contribuição

Código:

- Novas categorias de prompts
- Melhorias na interface e UX
- Otimizações de performance
- Implementação de segurança
- Suporte a novos idiomas

Documentação:

- Tutoriais passo-a-passo
- Exemplos de prompts por categoria
- Expansão do FAQ
- Tradução para outros idiomas
- Vídeos tutoriais no YouTube

Comunidade:

- Compartilhamento de templates
- Ajudar outros usuários
- Reportar bugs detalhadamente
- Sugerir melhorias
- Divulgar o projeto

🏆 Reconhecimento

- Créditos no README e documentação
- Badges especiais de contribuição
- Menção em releases e anúncios
- Participação em decisões estratégicas
- Acesso antecipado a novas features

💡 Sugestões Profissionais

⌚ Segurança e Privacidade

🔒 Arquivo .env

- Mantenha sempre protegido e **nunca commite** no repositório

- Use senhas fortes e únicas para produção
- Atualize periodicamente as chaves secretas

Ambientes Virtuais

- Use sempre `venv` para isolamento completo
- Nunca instale pacotes globalmente
- Documente versões específicas quando necessário

Backup e Recuperação

- Faça backup regular do banco SQLite
- Teste restauração em ambiente separado
- Mantenha histórico de versões importantes

Monitoramento

- Monitore logs para atividades suspeitas
- Configure alertas para uso excessivo
- Audite acessos periodicamente

👉 Otimização de Performance

Banco de Dados

- Considere PostgreSQL para produção
- Implemente índices para consultas frequentes
- Configure pool de conexões

Cache e CDN

- Use Redis para sessões em alta carga
- Configure CDN para assets estáticos
- Implemente compressão Gzip

Monitoramento

- Configure métricas de performance
- Monitore uso de recursos
- Identifique gargalos proativamente

Melhores Práticas de Prompts

Estrutura Clara

- Use seções bem definidas
- Forneça contexto detalhado
- Inclua exemplos quando relevante
- Seja específico nos requisitos

Iteração e Refinamento

- Teste prompts gradualmente
- Refine baseado em resultados
- Mantenha histórico de versões
- Documente aprendizados

Organização

- Categorize prompts logicamente
- Use nomenclatura consistente
- Mantenha biblioteca organizada
- Compartilhe melhores práticas

Trabalho em Equipe

Padronização

- Estabeleça convenções de nomenclatura
- Defina templates padrão por categoria
- Mantenha glossário de termos
- Documente processos internos

Colaboração

- Use templates compartilhados
- Implemente revisão de pares
- Mantenha documentação atualizada
- Incentive compartilhamento de conhecimento

Como Contribuir

Processo de Contribuição

1. **Fork** o projeto
2. **Clone** sua fork: `git clone https://github.com/SEU_USERNAME/Prompt-Master.git`
3. **Branch** descritiva: `git checkout -b feature/nome-da-feature`
4. **Desenvolvimento** seguindo padrões do projeto
5. **Testes** obrigatórios para novas funcionalidades
6. **Commit** padronizado: `git commit -m "tipo: descrição clara"`
7. **Push** para sua branch: `git push origin feature/nome-da-feature`
8. **Pull Request** detalhado

Tipos de Contribuição

Desenvolvimento

- ✨ Correção de bugs
- ♦ Novas funcionalidades

- 🔧 Melhorias técnicas
- 📱 Interface e UX
- 🔒 Segurança

📄 Documentação

- 📖 Guias e tutoriais
- 🔍 Exemplos práticos
- 🌐 Traduções
- 📝 FAQ expansão
- 🎥 Conteúdo audiovisual

🌐 Comunidade

- 👤 Suporte a usuários
- 📢 Divulgação
- 💡 Ideias inovadoras
- 📝 Testes beta
- 📊 Feedback e métricas

👤 Reconhecimento aos Contribuidores

- ⭐ **Créditos** no README e documentação oficial
- 🔗 **Badges** especiais no perfil GitHub
- 📢 **Menção** em releases e anúncios
- ⌚ **Participação** em decisões estratégicas
- ⚡ **Acesso** antecipado a novas features

📞 Canais de Comunicação

- GitHub Issues:** Bugs e solicitações
- GitHub Discussions:** Debates e ideias
- Discord:** Chat em tempo real da comunidade
- LinkedIn:** Grupo profissional
- Reddit:** r/PromptMaster

📋 Código de Conduta

Respeito, inclusão e colaboração são fundamentais. Todos os participantes devem:

- ☑ Ser respeitosos e profissionais
- ☑ Incluir e valorizar diversidade
- ☑ Manter foco na colaboração
- ☑ Seguir padrões éticos
- ☑ Respeitar propriedade intelectual

📦 Instalação e Uso

🚀 Instalação Rápida

```
# 1. Clone o repositório  
git clone https://github.com/lscheffel/Prompt-Master.git  
cd Prompt-Master  
  
# 2. Configure ambiente virtual  
python -m venv venv  
source venv/bin/activate # Linux/Mac  
# ou  
venv\Scripts\activate # Windows  
  
# 3. Instale dependências  
pip install -r requirements.txt  
  
# 4. Execute a aplicação  
python app.py  
  
# 5. Acesse no navegador  
# http://localhost:5000
```

🎮 Uso Básico

Criando seu Primeiro Prompt

1. Acesse "Criar Texto" no menu principal
2. Preencha nome, descrição e objetivos
3. Configure parâmetros específicos
4. Clique em "Gerar Prompt"

Gerenciando Prompts

1. Visualize todos os prompts na seção "Prompts"
2. Use opções de editar, visualizar ou excluir
3. Exporte prompts importantes em Markdown

🔧 Configuração Avançada

Variáveis de Ambiente (.env)

```
SECRET_KEY=sua-chave-secreta-super-segura  
FLASK_ENV=development  
DATABASE_URL=sqlite:///promptmaster.db
```

Executando em Produção

```
export FLASK_ENV=production  
unicorn --bind 0.0.0.0:8000 app:app
```

Licença

Este projeto está sob a licença **MIT**. Veja o arquivo [LICENSE](#) para detalhes completos.

Permissões:

- Uso comercial
- Modificação
- Distribuição
- Uso privado

Condições:

- Manter copyright
- Manter licença

Agradecimentos

Tecnologias e Frameworks

- **Flask:** Framework web excepcional
- **SQLAlchemy:** ORM poderoso e flexível
- **Bootstrap:** Framework CSS responsivo
- **WTForms:** Validação robusta de formulários
- **Jinja2:** Sistema de templates elegante

Comunidade

- **Contribuidores:** Pelo código e melhorias
- **Usuários:** Pelo feedback valioso
- **Open Source:** Por tornar tudo isso possível

Inspiração

- **Comunidade de IA:** Pela inovação constante
- **Desenvolvedores:** Pela paixão pela tecnologia
- **Educadores:** Pelo compromisso com o conhecimento

 Junte-se à Revolução dos Prompts!

[★ Star no GitHub](#) • [🍴 Fork o Projeto](#) • [📝 Reportar Issues](#) • [📖 Ler Documentação Completa](#)

Feito com  pela comunidade open source

Prompt Master - Transformando a interação humano-IA desde 2025