

# Contributing

- [1 Guia de Contribuição - FACIN\\_IA\\_D\\_P\\_N\\_1](#)
  - [1.1 🍷 Contribuindo para FACIN\\_IA\\_D\\_P\\_N\\_1](#)
  - [1.2 📋 Antes de Começar](#)
  - [1.3 🛠️ Setup de Desenvolvimento](#)
  - [1.4 ✅ Padrões de Código](#)
    - [1.4.1 Formatação](#)
    - [1.4.2 Verificação](#)
    - [1.4.3 Testes](#)
  - [1.5 📄 Commits](#)
    - [1.5.1 Mensagens de Commit](#)
    - [1.5.2 Tipos](#)
    - [1.5.3 Exemplo](#)
  - [1.6 🧪 Testes](#)
    - [1.6.1 Localização](#)
    - [1.6.2 Template](#)
    - [1.6.3 Cobertura](#)
  - [1.7 📖 Documentação](#)
    - [1.7.1 Docstrings](#)
    - [1.7.2 Atualizar Documentação](#)
  - [1.8 🔄 Pull Request](#)
    - [1.8.1 Checklist](#)
    - [1.8.2 Descrição PR](#)
  - [1.9 🐛 Reportando Bugs](#)
    - [1.9.1 Título Descritivo](#)
    - [1.9.2 Informações Necessárias](#)
    - [1.9.3 Template](#)
  - [1.10 ✨ Sugestões de Feature](#)
    - [1.10.1 Título Descritivo](#)
    - [1.10.2 Informação Necessária](#)
  - [1.11 📏 Convenções](#)
    - [1.11.1 Python Style](#)
    - [1.11.2 Nomenclatura](#)
    - [1.11.3 Imports](#)
  - [1.12 🚀 Processo de Merge](#)
  - [1.13 📞 Suporte](#)
  - [1.14 📄 Licença](#)

## 1 Guia de Contribuição - FACIN\_IA\_D\_P\_N\_1

### 1.1 🍷 Contribuindo para FACIN\_IA\_D\_P\_N\_1

Obrigado por se interessar em contribuir! Este guia descreve nossos padrões e processos.

---

### 1.2 📋 Antes de Começar

- Faça um Fork do repositório
  - Clone seu fork: `git clone https://github.com/seu-usuario/FACIN_IA_D_P_N_1.git`
  - Crie uma branch: `git checkout -b feature/sua-feature`
- 

## 1.3 Setup de Desenvolvimento

```
# Criar ambiente virtual
python -m venv .venv

# Ativar
.venv\Scripts\activate # Windows
source .venv/bin/activate # macOS/Linux

# Instalar dependências
pip install -r requirements.txt
pip install -r requirements-dev.txt

# Instalar pre-commit hooks
pre-commit install
```

---

## 1.4 Padrões de Código

### 1.4.1 Formatação

```
# Black
black . --line-length=100

# isort
isort . --profile black
```

### 1.4.2 Verificação

```
# Flake8
flake8 .

# mypy
mypy . --ignore-missing-imports

# pylint
pylint **/*.py
```

### 1.4.3 Testes

```
# Executar testes
pytest -v

# Com cobertura
pytest --cov=. --cov-report=html
```

---

## 1.5 Commits

### 1.5.1 Mensagens de Commit

tipo(escopo): descrição breve

Descrição mais detalhada se necessário.

Fixes #123

### 1.5.2 Tipos

- feat: Nova feature
- fix: Correção de bug
- docs: Documentação
- style: Formatação
- refactor: Refatoração
- test: Testes
- chore: Manutenção

### 1.5.3 Exemplo

feat(agents): adiciona novo memory agent

- Implementa armazenamento de memória persistente
- Adiciona embeddings semânticos
- Integra com LangGraph

Fixes #456

---

## 1.6 Testes

### 1.6.1 Localização

Testes devem estar em tests/ com padrão test\_\*.py

### 1.6.2 Template

```
import pytest
from app import function_to_test

class TestFunctionToTest:
    def test_success_case(self):
        result = function_to_test("input")
        assert result == "expected"

    def test_error_case(self):
        with pytest.raises(ValueError):
            function_to_test("invalid")
```

### 1.6.3 Cobertura

Mínimo 70% obrigatório na CI/CD

---

## 1.7 Documentação

### 1.7.1 Docstrings

```
def execute_query(query: str) -> str:
    """Executa query SQL no banco de dados.

    Args:
        query: Comando SQL válido

    Returns:
        Resultado da query em JSON

    Raises:
        DatabaseError: Se query falhar

    Example:
        >>> result = execute_query("SELECT * FROM servidores")
        """
```

## 1.7.2 Atualizar Documentação

- docs/README.md: Overview do projeto
- docs/API.md: Referência de APIs
- docs/ARCHITECTURE.md: Arquitetura
- errors/ERRORS\_LOG.md: Erros encontrados

## 1.8 Pull Request

### 1.8.1 Checklist

- ☐ Tests passam localmente
- ☐ Código formatado (Black)
- ☐ Imports organizados (isort)
- ☐ Sem linting issues (Flake8)
- ☐ Type hints adicionados (mypy)
- ☐ Documentação atualizada
- ☐ ERRORS\_LOG.md atualizado (se aplicável)

### 1.8.2 Descrição PR

#### ## Descrição

Breve descrição das mudanças

#### ## Tipo de Mudança

- [ ] Bug fix
- [ ] Nova feature
- [ ] Breaking change
- [ ] Documentação

#### ## Como Testar

Passos para reproduzir/testar

#### ## Screenshots (se aplicável)

#### ## Checklist

- [x] Código formatado
- [x] Testes passam
- [x] Documentação atualizada

## 1.9 🐛 Reportando Bugs

### 1.9.1 Título Descritivo

"Database Agent falha com queries complexas" ao invés de "Erro"

### 1.9.2 Informações Necessárias

- Versão Python
- Versão do FACIN\_IA\_D\_P\_N\_1
- Stack trace completo
- Passos para reproduzir
- Comportamento esperado vs atual

### 1.9.3 Template

**\*\*Descrição do Bug\*\***

[Descrição clara do problema]

**\*\*Passos para Reproduzir\*\***

1. [Passo 1]

2. [Passo 2]

**\*\*Comportamento Esperado\*\***

[O que deveria acontecer]

**\*\*Comportamento Atual\*\***

[O que realmente aconteceu]

**\*\*Environment\*\***

- Python: 3.12
- OS: Windows/Linux/macOS
- FACIN\_IA\_D\_P\_N\_1 Version: 1.0.0

---

## 1.10 ✨ Sugestões de Feature

### 1.10.1 Título Descritivo

"Adicionar suporte a PostgreSQL" ao invés de "Melhorar banco de dados"

### 1.10.2 Informação Necessária

- Caso de uso
- Benefício esperado
- Possibilidade de implementação

---

## 1.11 📖 Convenções

### 1.11.1 Python Style

Seguir [PEP 8](#)

### 1.11.2 Nomenclatura

- Funções/métodos: snake\_case
- Classes: PascalCase
- Constantes: UPPER\_CASE
- Variáveis privadas: \_private\_var

### 1.11.3 Imports

*# Ordem: stdlib, third-party, local*

```
import os
```

```
import sys
```

```
from typing import List
```

```
from langchain import ...
```

```
from streamlit import ...
```

```
from app import execute_query
```

---

## 1.12 Processo de Merge

1. Submeta PR com descrição clara
  2. CI/CD valida automaticamente
  3. Review code (mínimo 1 aprovação)
  4. Merge automático quando aprovado
  5. GitHub Actions faz deploy
- 

## 1.13 Suporte

- **Issues:** [GitHub Issues](#)
  - **Discussões:** [GitHub Discussions](#)
  - **Email:** gut.passos@gmail.com
- 

## 1.14 Licença

Ao contribuir, você concorda que suas mudanças serão licenciadas sob a mesma licença MIT do projeto.

---

**Versão:** 1.0.0

**Data:** 27/02/2026