

Guia de Manutenção - Sistema de Gerenciamento para Empresa de Instalação de Tile

Data: 06/06/2025

Versão: 1.0

Sumário

1. [Introdução](#)
2. [Manutenção do Servidor](#)
3. [Manutenção do Banco de Dados](#)
4. [Manutenção do Backend](#)
5. [Manutenção do Frontend](#)
6. [Backup e Recuperação](#)
7. [Monitoramento](#)
8. [Atualizações](#)
9. [Solução de Problemas Comuns](#)
10. [Contatos de Suporte](#)

1. Introdução

Este guia de manutenção fornece instruções detalhadas para manter o Sistema de Gerenciamento para Empresa de Instalação de Tile em funcionamento adequado. Ele abrange tarefas de manutenção regulares, procedimentos de backup, monitoramento e solução de problemas.

1.1 Visão Geral do Sistema

O sistema é composto por:

- **Backend:** API RESTful desenvolvida em Python com Flask
- **Frontend:** Interface de usuário desenvolvida em React
- **Banco de Dados:** PostgreSQL (produção) ou SQLite (desenvolvimento)
- **Servidor Web:** Nginx para servir o frontend e fazer proxy para o backend

1.2 Requisitos de Manutenção

Para realizar a manutenção do sistema, você precisará:

- Acesso SSH ao servidor
- Credenciais de administrador do banco de dados
- Conhecimento básico de Linux, Python e JavaScript
- Acesso ao repositório de código-fonte

2. Manutenção do Servidor

2.1 Verificação de Recursos

Monitore regularmente os recursos do servidor para garantir que o sistema esteja funcionando adequadamente:

```
# Verificar uso de CPU e memória
top

# Verificar espaço em disco
df -h

# Verificar uso de inodes
df -i

# Verificar processos em execução
ps aux | grep -E 'gunicorn|nginx'
```

2.2 Atualizações do Sistema Operacional

Mantenha o sistema operacional atualizado para garantir segurança e estabilidade:

```
# Atualizar lista de pacotes
sudo apt update

# Instalar atualizações
sudo apt upgrade -y

# Reiniciar o servidor (se necessário)
sudo reboot
```

2.3 Limpeza de Arquivos Temporários

Limpe regularmente arquivos temporários para liberar espaço em disco:

```
# Limpar arquivos temporários
sudo find /tmp -type f -atime +7 -delete

# Limpar logs antigos
sudo find /var/log -name "*.gz" -type f -mtime +30 -delete
```

2.4 Verificação de Logs do Sistema

Verifique regularmente os logs do sistema para identificar problemas:

```
# Verificar logs do sistema
sudo journalctl -xe

# Verificar logs do Nginx
sudo tail -f /var/log/nginx/error.log

# Verificar logs do Gunicorn
sudo tail -f /var/log/tile-system/backend.log
```

3. Manutenção do Banco de Dados

3.1 Backup do Banco de Dados

Realize backups regulares do banco de dados:

```
# Backup diário do banco de dados
pg_dump -U postgres tile_system > /backup/tile_system_$(date +%Y%m%d).sql

# Compactar o backup
gzip /backup/tile_system_$(date +%Y%m%d).sql

# Remover backups antigos (mais de 30 dias)
find /backup -name "tile_system_*.sql.gz" -type f -mtime +30 -delete
```

3.2 Verificação de Integridade

Verifique regularmente a integridade do banco de dados:

```
# Conectar ao banco de dados
psql -U postgres tile_system
```

```
# Verificar tabelas
\d

# Verificar integridade
VACUUM ANALYZE;

# Verificar estatísticas
SELECT relname, last_vacuum, last_analyze FROM
pg_stat_user_tables;
```

3.3 Otimização de Desempenho

Otimize regularmente o desempenho do banco de dados:

```
-- Reconstruir índices
REINDEX DATABASE tile_system;

-- Analisar consultas lentas
SELECT * FROM pg_stat_activity WHERE state = 'active' ORDER BY
query_start ASC;

-- Verificar tamanho das tabelas
SELECT
    relname AS table_name,
    pg_size_pretty(pg_total_relation_size(relid)) AS total_size
FROM pg_catalog.pg_statio_user_tables
ORDER BY pg_total_relation_size(relid) DESC;
```

3.4 Monitoramento de Crescimento

Monitore o crescimento do banco de dados para planejar a capacidade:

```
-- Verificar tamanho do banco de dados
SELECT pg_size_pretty(pg_database_size('tile_system'));

-- Verificar crescimento das tabelas
SELECT
    relname AS table_name,
    n_tup_ins AS inserts,
    n_tup_upd AS updates,
    n_tup_del AS deletes
FROM pg_stat_user_tables
ORDER BY n_tup_ins DESC;
```

4. Manutenção do Backend

4.1 Verificação de Logs

Verifique regularmente os logs do backend para identificar problemas:

```
# Verificar logs do backend  
tail -f /var/log/tile-system/backend.log
```

4.2 Reiniciar Serviços

Reinicie os serviços do backend quando necessário:

```
# Reiniciar o serviço do backend  
sudo systemctl restart tile-system-backend  
  
# Verificar status do serviço  
sudo systemctl status tile-system-backend
```

4.3 Atualização de Dependências

Mantenha as dependências do backend atualizadas:

```
# Ativar o ambiente virtual  
cd /path/to/tile-system/backend  
source venv/bin/activate  
  
# Atualizar pip  
pip install --upgrade pip  
  
# Listar dependências desatualizadas  
pip list --outdated  
  
# Atualizar dependências  
pip install --upgrade -r requirements.txt  
  
# Desativar o ambiente virtual  
deactivate
```

4.4 Verificação de Segurança

Verifique regularmente vulnerabilidades de segurança:

```
# Verificar vulnerabilidades nas dependências
pip install safety
safety check -r requirements.txt
```

5. Manutenção do Frontend

5.1 Verificação de Logs

Verifique regularmente os logs do frontend para identificar problemas:

```
# Verificar logs do Nginx (frontend)
tail -f /var/log/nginx/access.log
tail -f /var/log/nginx/error.log
```

5.2 Atualização de Dependências

Mantenha as dependências do frontend atualizadas:

```
# Navegar até o diretório do frontend
cd /path/to/tile-system/frontend

# Verificar dependências desatualizadas
pnpm outdated

# Atualizar dependências
pnpm update

# Reconstruir a aplicação
pnpm build
```

5.3 Otimização de Desempenho

Otimize regularmente o desempenho do frontend:

```
# Verificar tamanho dos arquivos
du -sh /path/to/tile-system/frontend/dist/*

# Verificar configuração do Nginx
sudo nginx -t

# Reiniciar o Nginx
sudo systemctl restart nginx
```

5.4 Verificação de Compatibilidade

Verifique regularmente a compatibilidade com navegadores:

```
# Instalar o Lighthouse
npm install -g lighthouse

# Executar o Lighthouse
lighthouse https://seu-dominio.com --view
```

6. Backup e Recuperação

6.1 Estratégia de Backup

Implemente uma estratégia de backup abrangente:

1. **Backup Diário:** Backup completo do banco de dados e arquivos de upload
2. **Backup Semanal:** Backup completo do sistema, incluindo código-fonte
3. **Backup Mensal:** Backup completo do sistema, armazenado em local externo

6.2 Procedimento de Backup

```
# Criar diretório de backup
mkdir -p /backup/$(date +%Y%m%d)

# Backup do banco de dados
pg_dump -U postgres tile_system > /backup/$(date +%Y%m%d)/
database.sql

# Backup dos arquivos de upload
tar -czf /backup/$(date +%Y%m%d)/uploads.tar.gz /path/to/tile-
system/uploads

# Backup do código-fonte
tar -czf /backup/$(date +%Y%m%d)/source.tar.gz /path/to/tile-
system

# Compactar tudo
tar -czf /backup/tile-system-backup_$(date +%Y%m%d).tar.gz /
backup/$(date +%Y%m%d)

# Remover diretório temporário
rm -rf /backup/$(date +%Y%m%d)
```

6.3 Procedimento de Recuperação

```
# Descompactar o backup
tar -xzf /backup/tile-system-backup_20250606.tar.gz -C /tmp

# Restaurar o banco de dados
psql -U postgres -c "DROP DATABASE IF EXISTS tile_system;"
psql -U postgres -c "CREATE DATABASE tile_system;"
psql -U postgres tile_system < /tmp/20250606/database.sql

# Restaurar os arquivos de upload
rm -rf /path/to/tile-system/uploads
tar -xzf /tmp/20250606/uploads.tar.gz -C /

# Restaurar o código-fonte (se necessário)
tar -xzf /tmp/20250606/source.tar.gz -C /tmp
rsync -av --delete /tmp/path/to/tile-system/ /path/to/tile-system/

# Limpar
rm -rf /tmp/20250606
```

6.4 Teste de Recuperação

Teste regularmente o procedimento de recuperação para garantir que os backups estão funcionando corretamente:

```
# Criar um ambiente de teste
mkdir -p /tmp/tile-system-test

# Restaurar o backup no ambiente de teste
tar -xzf /backup/tile-system-backup_$(date +%Y%m%d).tar.gz -C /tmp/tile-system-test

# Verificar os arquivos
ls -la /tmp/tile-system-test

# Limpar
rm -rf /tmp/tile-system-test
```

7. Monitoramento

7.1 Monitoramento de Recursos

Monitore regularmente os recursos do sistema:


```
# Instalar o Glances
pip install glances

# Executar o Glances
glances
```

7.2 Monitoramento de Disponibilidade

Configure o monitoramento de disponibilidade para ser alertado em caso de problemas:

```
# Verificar se o backend está respondendo
curl -I http://localhost:5000/api/health

# Verificar se o frontend está respondendo
curl -I http://localhost
```

7.3 Monitoramento de Logs

Configure o monitoramento de logs para ser alertado em caso de erros:

```
# Monitorar logs de erro
grep -i "error\|exception\|fail" /var/log/tile-system/
backend.log
```

7.4 Alertas

Configure alertas para ser notificado em caso de problemas:

- Uso de CPU acima de 80% por mais de 5 minutos
- Uso de memória acima de 90%
- Espaço em disco abaixo de 10%
- Serviços parados (Nginx, Gunicorn)
- Erros 5xx no backend ou frontend

8. Atualizações

8.1 Atualização do Sistema

Para atualizar o sistema para uma nova versão:

```
# Fazer backup antes da atualização
./deploy.sh backup

# Atualizar o código-fonte
cd /path/to/tile-system
git pull

# Atualizar o backend
cd backend
source venv/bin/activate
pip install -r requirements.txt
flask db upgrade
deactivate

# Atualizar o frontend
cd ../frontend
pnpm install
pnpm build

# Reiniciar os serviços
sudo systemctl restart tile-system-backend
sudo systemctl restart nginx
```

8.2 Rollback

Em caso de problemas após a atualização, faça o rollback para a versão anterior:

```
# Restaurar o código-fonte
cd /path/to/tile-system
git reset --hard HEAD~1

# Restaurar o banco de dados
psql -U postgres tile_system < /backup/
tile_system_before_update.sql

# Reconstruir o frontend
cd frontend
pnpm install
pnpm build

# Reiniciar os serviços
sudo systemctl restart tile-system-backend
sudo systemctl restart nginx
```

8.3 Verificação Pós-Atualização

Após a atualização, verifique se o sistema está funcionando corretamente:

```
# Verificar status dos serviços
sudo systemctl status tile-system-backend
sudo systemctl status nginx

# Verificar logs
tail -f /var/log/tile-system/backend.log
tail -f /var/log/nginx/error.log

# Verificar API
curl -I http://localhost:5000/api/health

# Verificar frontend
curl -I http://localhost
```

9. Solução de Problemas Comuns

9.1 Backend Não Inicia

Se o backend não iniciar:

1. Verifique os logs: `bash tail -f /var/log/tile-system/backend.log`
2. Verifique se o ambiente virtual está ativado: `bash cd /path/to/tile-system/backend source venv/bin/activate`
3. Verifique se todas as dependências estão instaladas: `bash pip install -r requirements.txt`
4. Verifique se o banco de dados está acessível: `bash psql -U postgres -h localhost -d tile_system -c "SELECT 1;"`
5. Verifique se as variáveis de ambiente estão configuradas corretamente: `bash cat .env`

9.2 Frontend Não Carrega

Se o frontend não carregar:

1. Verifique os logs do Nginx: `bash tail -f /var/log/nginx/error.log`
2. Verifique se os arquivos estáticos estão presentes: `bash ls -la /path/to/tile-system/frontend/dist`
3. Verifique se o Nginx está configurado corretamente: `bash sudo nginx -t`

4. Verifique se o Nginx está em execução: `bash sudo systemctl status nginx`

5. Verifique se o backend está acessível:

```
bash curl -I http://localhost:5000/api/health
```

9.3 Problemas de Banco de Dados

Se houver problemas com o banco de dados:

1. Verifique se o PostgreSQL está em execução: `bash sudo systemctl status postgresql`

2. Verifique se o banco de dados existe: `bash psql -U postgres -c "\l" | grep tile_system`

3. Verifique se as tabelas existem: `bash psql -U postgres -d tile_system -c "\dt"`

4. Verifique se há conexões ativas: `bash psql -U postgres -c "SELECT * FROM pg_stat_activity WHERE datname = 'tile_system';"`

5. Verifique se há bloqueios: `bash psql -U postgres -d tile_system -c "SELECT * FROM pg_locks l JOIN pg_class c ON l.relation = c.oid WHERE c.relkind = 'r';"`

9.4 Problemas de Desempenho

Se houver problemas de desempenho:

1. Verifique o uso de recursos: `bash top`

2. Verifique as consultas lentas: `bash psql -U postgres -d tile_system -c "SELECT * FROM pg_stat_activity WHERE state = 'active' ORDER BY query_start ASC;"`

3. Verifique o tamanho do banco de dados: `bash psql -U postgres -d tile_system -c "SELECT pg_size_pretty(pg_database_size('tile_system'));"`

4. Verifique o tamanho das tabelas: `bash psql -U postgres -d tile_system -c "SELECT relname AS table_name, pg_size_pretty(pg_total_relation_size(relid)) AS total_size FROM`

```
pg_catalog.pg_statio_user_tables ORDER BY  
pg_total_relation_size(relid) DESC;"
```

5. Verifique o cache do Nginx: `bash sudo tail -f /var/log/nginx/
access.log | grep -v "HIT"`

10. Contatos de Suporte

Em caso de problemas que não conseguir resolver, entre em contato com o suporte técnico:

- **Email:** suporte@tilessystem.com
- **Telefone:** (555) 123-4567
- **Horário de Atendimento:** Segunda a Sexta, das 8h às 18h

10.1 Informações Necessárias para o Suporte

Ao entrar em contato com o suporte, tenha as seguintes informações em mãos:

1. Versão do sistema
2. Descrição detalhada do problema
3. Logs relevantes
4. Capturas de tela (se aplicável)
5. Passos para reproduzir o problema

10.2 Níveis de Severidade

Os problemas são classificados nos seguintes níveis de severidade:

- **Crítico:** Sistema completamente indisponível
- **Alto:** Funcionalidade principal comprometida
- **Médio:** Funcionalidade secundária comprometida
- **Baixo:** Problema cosmético ou de usabilidade

O tempo de resposta varia de acordo com o nível de severidade:

- **Crítico:** 1 hora
- **Alto:** 4 horas
- **Médio:** 24 horas
- **Baixo:** 48 horas