

Descrição da Infraestrutura de Implantação

Data: 22 de Junho de 2025

Versão: 1.0 (Rascunho Inicial)

Autor: Henrique Valente Lima (211055380)

1. Introdução

Este documento descreve a infraestrutura de TI proposta para a implantação e operação do Sistema de Gestão de Feiras em um ambiente de produção. O objetivo é delinear os requisitos de hardware, software e serviços necessários para garantir a disponibilidade, segurança, desempenho e manutenibilidade da aplicação.

2. Ambiente de Produção Proposto

Para a implantação do Sistema de Gestão de Feiras, propõe-se uma arquitetura web padrão, utilizando um servidor único (que pode ser uma máquina virtual) para hospedar tanto a aplicação quanto o banco de dados, adequado para um projeto de pequeno a médio porte.

2.1. Hardware (Servidor)

- **Tipo:** Máquina Virtual (VM) em um provedor de nuvem (ex: AWS EC2, Google Cloud Compute Engine, DigitalOcean Droplet) ou um servidor dedicado virtualizado. A escolha por uma VM oferece flexibilidade e escalabilidade futura.
- **Especificações Mínimas:**
 - **CPU:** 2 vCPUs (processadores virtuais), para garantir capacidade de processamento para as requisições web e operações de banco de dados.
 - **RAM:** 4 GB (Gigabytes), suficiente para a aplicação Flask, o SGBD PostgreSQL e o sistema operacional base.
 - **Disco:** 50 GB SSD (Solid State Drive), para desempenho rápido de leitura/escrita e espaço suficiente para o sistema operacional, código-fonte, dados do banco de dados e logs.

2.2. Software Essencial

- **Sistema Operacional (SO):** Ubuntu Server 22.04 LTS (Long Term Support). Escolhido por sua robustez, vasta comunidade, bom suporte e estabilidade para ambientes de servidor.
- **Linguagem de Programação:** Python 3.9+ (a versão específica utilizada no desenvolvimento do protótipo).
- **Servidor WSGI (Web Server Gateway Interface):** **Gunicorn**. Será utilizado para servir a aplicação Flask em produção. Ele lida com o paralelismo e a comunicação entre o servidor web e a aplicação Python.
- **Servidor Web (Proxy Reverso):** **Nginx**. Funcionará como um proxy reverso na frente do Gunicorn. O Nginx será responsável por:
 - Receber as requisições HTTP/HTTPS dos usuários.
 - Servir arquivos estáticos (CSS, JavaScript, imagens) diretamente, de forma mais eficiente.

- Passar as requisições dinâmicas (para as rotas Flask) para o Gunicorn.
 - Gerenciar certificados SSL/TLS para comunicação segura (HTTPS).
- **Sistema de Gerenciamento de Banco de Dados (SGBD): PostgreSQL 14+.** Selecionado por sua confiabilidade, integridade de dados, recursos avançados e capacidade de escalabilidade.

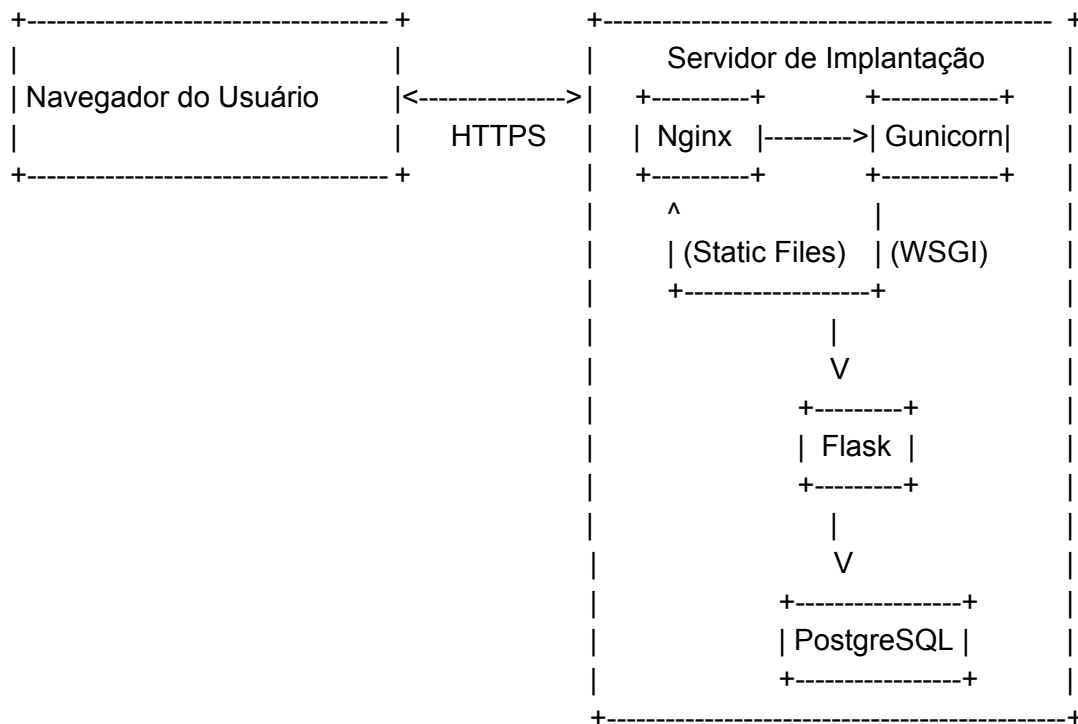
2.3. Serviços e Ferramentas Complementares

- **Controle de Versão: Git.** O código-fonte será gerenciado em um repositório Git (ex: GitHub, GitLab, Bitbucket), facilitando o controle de mudanças e a implantação.
- **Gerenciador de Pacotes Python: pip.** Utilizado para instalar e gerenciar as dependências do projeto (`requirements.txt`).
- **Gerenciador de Processos: systemd** (nativo do Ubuntu). Será configurado para gerenciar os processos do Gunicorn, garantindo que a aplicação inicie automaticamente ao reiniciar o servidor e que seja reiniciada em caso de falha.
- **Segurança:**
 - **Certificado SSL/TLS:** Essencial para criptografar a comunicação entre os clientes e o servidor (HTTPS), protegendo dados sensíveis. Pode ser obtido gratuitamente via Let's Encrypt.
 - **Firewall:** Configuração de regras de firewall (ex: UFW - Uncomplicated Firewall no Ubuntu) para permitir apenas o tráfego essencial (portas 80 e 443 para Nginx, e a porta do PostgreSQL apenas para o servidor da aplicação).
 - **Credenciais Seguras:** Variáveis de ambiente ou um sistema de gerenciamento de segredos para armazenar informações sensíveis (chaves de API, senhas de banco de dados) fora do código-fonte.
- **Monitoramento e Logs (Opcional, mas recomendado para produção):**
 - Configuração de coleta de logs da aplicação Flask, Gunicorn e Nginx para depuração e auditoria.
 - Ferramentas básicas de monitoramento de sistema (CPU, RAM, disco) para identificar gargalos de performance.

3. Topologia da Implantação

O fluxo de requisições será o seguinte:

1. O **Navegador do Usuário** faz uma requisição HTTP/HTTPS para o domínio do sistema.
2. O **Nginx** (operando na porta 80/443 do servidor) recebe a requisição.
 - Se for um arquivo estático (CSS, JS), o Nginx o serve diretamente.
 - Se for uma requisição dinâmica (para a aplicação Flask), o Nginx atua como proxy reverso e a encaminha para o **Gunicorn**.
3. O **Gunicorn** recebe a requisição do Nginx, processa-a através da aplicação **Flask**.
4. A aplicação **Flask** interage com o **PostgreSQL** (normalmente em uma porta interna do servidor, acessível apenas pelo Flask) para armazenar ou recuperar dados.
5. A resposta é enviada de volta pelo Flask ao Gunicorn, depois ao Nginx, e finalmente ao navegador do usuário.



4. Processo de Implantação (Visão Geral)

O processo de implantação envolverá os seguintes passos gerais:

1. **Provisionamento do Servidor:** Criação da VM no provedor de nuvem e instalação do SO (Ubuntu Server).
2. **Configuração Inicial do Servidor:** Atualização do sistema, criação de usuários com permissões limitadas, configuração de firewall (UFW).
3. **Instalação de Dependências:** Instalação de Python, PostgreSQL, Nginx e Gunicorn.
4. **Configuração do PostgreSQL:** Criação do banco de dados, usuário e permissões específicas para a aplicação.
5. **Criação do Ambiente Virtual:** No servidor, clonagem do repositório Git e criação/ativação do ambiente virtual.
6. **Instalação das Dependências Python:** `pip install -r requirements.txt`.
7. **Configuração da Aplicação Flask:** Ajuste de variáveis de ambiente para produção (URI do banco de dados, SECRET_KEY).
8. **Configuração do Gunicorn:** Criação de um serviço `systemd` para o Gunicorn.
9. **Configuração do Nginx:** Criação de um arquivo de configuração de site para proxy reverso e servir arquivos estáticos.
10. **Configuração SSL/TLS:** Obtenção e instalação de certificado SSL (ex: via Certbot para Let's Encrypt).
11. **Reinício de Serviços:** Início/reinício do Nginx e Gunicorn.

5. Considerações de Segurança e Manutenção

- **Atualizações:** Manutenção regular do SO e das dependências de software para aplicar patches de segurança.
- **Backups:** Implementação de rotinas de backup diárias/semanais do banco de dados e do código-fonte.
- **Monitoramento:** Configuração de alertas para uso excessivo de recursos ou falhas na aplicação.
- **Logs:** Análise regular dos logs para identificar problemas e atividades suspeitas.
- **Acesso Restrito:** Acesso SSH ao servidor limitado por chaves e apenas para usuários autorizados.