

ESCOLA POLITÉCNICA				
CURSO: CIÊNCIA DE DADOS E INTELIGÊNCIA ARTIFICIAL				
COMPONENTE CURRICULAR: BANCOS DE DADOS RELACIONAIS				CÓDIGO: 12514
CARGA HORÁRIA TOTAL DO COMPONENTE CURRICULAR:	CH Teórica 76	CH Prática 38	CH Autônoma 50	CH Extensão 45
DOCENTE(S): FELIPE CAVALARO				
NOME DOS ALUNO (A)S: Amanda Oliveira Piton Giulia Feitosa De Souza Giulia Monteiro Garrido Mateus Andrade Antezana Willian David Gomes Da Silva Lana				

RESUMO

A falta de informação centralizada sobre os hemocentros da região de Campinas e a baixa qualidade dos sites existentes dificultam o acesso dos doadores a dados relevantes e atualizados, resultando em baixa adesão às campanhas de doação de sangue. O projeto HemoCamp tem como objetivo desenvolver uma plataforma web que centralize informações dos hemocentros e registre histórico de doações, além de coletar preferências de horários para auxiliar no planejamento de campanhas. A metodologia adotada seguiu sprints quinzenais utilizando Kanban, com desenvolvimento do backend em Python/Flask, banco de dados MySQL hospedado no Railway, frontend responsivo em HTML/CSS/JavaScript e sistema de autenticação com criptografia bcrypt. Como resultados, foram implementadas 11 tabelas no banco de dados, 8 requisitos funcionais, API RESTful completa, totalizando um sistema funcional e seguro. Conclui-se que o HemoCamp contribui significativamente para melhorar a comunicação entre hemocentros e doadores, alinhando-se aos ODS 3, 9 e 17 da ONU, com potencial de aumentar o número de doadores regulares, reduzir a escassez de sangue nos hospitais e fortalecer a conscientização sobre a importância da doação contínua, promovendo impacto social positivo e salvando vidas através da tecnologia.

Palavras-chave: Doação de sangue. Sistemas web. Saúde pública. Integração de dados.

1. IDENTIFICAÇÃO

A doação de sangue é um pilar fundamental para a manutenção dos sistemas de saúde pública e privada, sendo vital para cirurgias, tratamento de doenças crônicas e atendimentos

de urgência. No entanto, o Brasil enfrenta um desafio contínuo para manter os estoques de sangue em níveis seguros. Segundo dados recentes do Ministério da Saúde, apenas cerca de 1,6% da população brasileira doa sangue regularmente (MINISTÉRIO DA SAÚDE, 2024). Embora este índice esteja dentro da recomendação da Organização Mundial da Saúde (OMS), que estipula uma taxa entre 1% e 3%, o país opera frequentemente no limite, com hemocentros a enfrentar baixas críticas em períodos sazonais, como festas de fim de ano e inverno (AGÊNCIA BRASIL, 2023).

Na região de Campinas, observa-se uma problemática específica: a assimetria de informação. A falta de uma plataforma centralizada que agregue dados sobre localização, horários de funcionamento e níveis de estoques dos diversos hemocentros dificulta o acesso dos potenciais doadores. Muitos cidadãos deixam de realizar a doação por desconhecimento dos requisitos básicos ou por não receberem notificações oportunas sobre campanhas de coleta e emergências. O projeto HemoCamp surge como uma resposta tecnológica a esta lacuna, visando mitigar a ineficiência comunicacional que contribui para a escassez de hemoderivados.

O projeto HemoCamp alinha-se diretamente com a Agenda 2030 da ONU, contribuindo para três Objetivos de Desenvolvimento Sustentável:

- ODS 3 (Saúde e Bem-Estar): Ao facilitar o processo de doação e aumentar a captação de doadores, o sistema contribui diretamente para a redução da mortalidade e para o acesso a tratamentos de saúde essenciais que dependem de transfusões.
- ODS 9 (Indústria, Inovação e Infraestrutura): A solução propõe uma inovação tecnológica na gestão da saúde, modernizando a infraestrutura de comunicação entre os hemocentros e a sociedade através de um sistema digital eficiente.
- ODS 17 (Parcerias e Meios de Implementação): O projeto fomenta a cooperação entre diferentes entidades, incluindo a comunidade acadêmica, instituições governamentais e hospitais privados, fortalecendo a rede de solidariedade necessária para a saúde pública.

O território de atuação inicial do projeto é a Região Metropolitana de Campinas (RMC). O público beneficiado abrange, primariamente, toda a população que necessita ou poderá

necessitar de transfusões sanguíneas, o que torna a quantificação exata desafiadora, dado o caráter universal da saúde. No entanto, o impacto direto recai sobre os doadores voluntários da região e os pacientes atendidos pela rede hospitalar.

Economicamente, o projeto impacta o Primeiro Setor (instituições públicas como o Hemocentro da Unicamp e hospitais do SUS), auxiliando na otimização de recursos públicos através de uma gestão de estoques. Simultaneamente, beneficia o Segundo Setor (hospitais e clínicas privadas), que dependem da regularidade no abastecimento de sangue para a realização de procedimentos eletivos e de emergência, reduzindo custos operacionais associados ao cancelamento de cirurgias por falta de insumos.

Objetivos Específicos e Desenvolvimento Técnico

O objetivo central do HemoCamp é desenvolver uma aplicação web responsiva que centralize as informações dos bancos de sangue e gerencie o fluxo de doações. Para tal, a aplicação foi desenvolvida utilizando a linguagem Python 3.12 com o microframework Flask para o backend, garantindo uma arquitetura em camadas baseada no padrão MTV (Model-Template-View), seguindo as convenções do Flask. A persistência dos dados é realizada através de um banco de dados relacional MySQL 8.0, hospedado na nuvem (plataforma Railway), enquanto o frontend foi construído com HTML5, CSS3 e JavaScript, priorizando a acessibilidade e a usabilidade.

Os objetivos específicos do projeto incluem:

- Desenvolver uma plataforma web que reúna dados essenciais sobre os hemocentros da região de Campinas, como localização, horários de funcionamento e requisitos para doação.
- Implementar um sistema de agendamento online para organizar a demanda e reduzir filas nos postos de coleta.
- Permitir o cadastro e login seguros, diferenciando perfis de doadores (validação via

CPF) e colaboradores de hemocentros (validação via CNPJ), com uso de criptografia para proteção de dados.

- Oferecer ferramentas para que os hemocentros possam gerenciar seus estoques de sangue em tempo real (identificando níveis críticos) e criar campanhas de doação direcionadas.
- Registrar o histórico de doações dos usuários e calcular automaticamente, via regras de negócio no banco de dados, a data permitida para a próxima doação.

A aplicação estrutura-se através das seguintes telas e funcionalidades principais:

- Tela Inicial (*index.html*): Página inicial, ou home, do site. Nela é possível navegar para diferentes abas.
- Tela de Login e Cadastro (*login.html*, *cadastro.html*): Permite o registro seguro de doadores (com validação de CPF) e colaboradores de hemocentros (com validação de CNPJ), utilizando criptografia bcrypt para proteção de credenciais.
- Mapeamento de Hemocentros (*Hemocentros.html*): Interface que lista e localiza as unidades de coleta parceiras, exibindo endereços e horários de funcionamento.
- Agendamento de Doações (*agendamento.html*): Funcionalidade que permite ao usuário escolher datas e horários disponíveis para o agendamento de sua doação, permite o usuário selecionar o motivo da doação, o que pretende doar e adicionar alguma observação.
- Gestão de Campanhas (*campanha.html*): Área dedicada à divulgação de campanhas ativas, permitindo aos hemocentros alertar sobre tipos sanguíneos críticos e necessidades urgentes; usuários doadores conseguem acompanhar quais campanhas estão ativas na região.

- Perfil do Usuário (*Perfil.html*): Painel pessoal que reúne as informações principais de cada usuário. Nele, os doadores podem visualizar seus dados, acompanhar doações agendadas e configurar suas preferências dentro do sistema. No caso dos colaboradores, a mesma interface inclui as funcionalidades administrativas, como o cadastro e a gestão de campanhas de doação.
- Notícias (*Noticias.html*): Página estática para futura implementação de notícias sobre doações de sangue, eventos e mais.

2. DESCRIÇÃO E ANÁLISE DAS ATIVIDADES

Arquitetura Geral e Tecnologias Utilizadas

A aplicação foi construída seguindo uma arquitetura baseada no padrão MVC, garantindo separação clara entre apresentação, regras de negócio e persistência. O backend foi desenvolvido em Python 3.12 utilizando o microframework Flask, complementado com Blueprints para modularização das rotas, um recurso essencial para organizar funcionalidades como cadastro de usuários, agendamento, campanhas e estoque.

O armazenamento persistente foi realizado em MySQL 8.0, hospedado na Railway, aproveitando suas propriedades ACID e suporte a queries complexas. A integração com o banco ocorreu por meio do mysql-connector-python, possibilitando o uso de transações, views, constraints e stored procedures.

No frontend, HTML5, CSS3 e JavaScript foram utilizados para criar interfaces responsivas. A opção por CSS puro em vez de frameworks visuais foi estratégica para manter baixo peso da aplicação e maior controle estético. A integração entre frontend e backend ocorreu através de chamadas assíncronas via Fetch API.

O projeto utilizou ainda Git e GitHub para versionamento e Jira para organização das tarefas usando metodologias ágeis, permitindo ciclos quinzenais de entrega e revisão.

A organização do repositório foi planejada para separar claramente as responsabilidades do frontend, backend e documentação, favorecendo manutenção, escalabilidade e colaboração entre membros da equipe. A estrutura final ficou definida conforme abaixo:

```
app/
|__ pycache/
|__ back/          # Núcleo do backend da aplicação
|   |__ models/    # Classes e scripts responsáveis pelo acesso ao banco de dados
|   |__ routes/    # Blueprints e rotas que implementam as funcionalidades
|   |__ utils/     # Funções auxiliares, validações e serviços de apoio
|__ config/
|__ front/
|   |__ static/    # Arquivos estáticos usados pelo frontend
|   |   |__ scripts/ # Arquivos JavaScript
|   |   |__ styles/ # Estilos CSS e componentes visuais
|   |   |__ imagens/
|   |__ templates/ # Estruturas HTML
docs/
|__ diagrams/     # Diagramas MER e Modelo Lógico
|__ prototype/    # Protótipo inicial do projeto
|__ reports/      # Relatórios já entregues
|__ assets

README.md
```

Essa organização permite que o backend e o frontend evoluam de forma independente, mantendo ainda uma estrutura limpa e padronizada para todos os membros do grupo. Os módulos do backend foram segmentados em models, routes e utils para refletir a lógica da arquitetura, enquanto o frontend permanece isolado em front, facilitando testes, refatorações e futuras expansões. A pasta docs concentra toda a documentação técnica, garantindo rastreabilidade das decisões tomadas ao longo do desenvolvimento.

A execução do projeto foi organizada de forma colaborativa, com divisão clara de tarefas entre os integrantes. Os alunos Giulia Feitosa e Willian se encarregaram do frontend, com foco no HTML e CSS. O aluno Mateus cuidou do JavaScript do projeto. O backend foi desenvolvido pela Amanda. A criação e gerenciamento do banco de dados foi encarregado à Giulia Monteiro.

Atividades realizadas

DATA	ATIVIDADE DESENVOLVIDA	NÚMERO DE HORAS
19/08/2025 a 26/08/2025	Levantamento de requisitos, definição das regras de negócio (Doador vs. Hemocentro) e modelagem conceitual e lógica do banco de dados	6
26/08/2025 a 09/09/2025	Desenvolvimento das interfaces Front-End: Telas de Login, Cadastro, Home e navegação responsiva (HTML/CSS)	8
09/09/2025 a 23/09/2025	Desenvolvimento do Back-End: Criação dos models e implementação da autenticação	8
23/09/2025 a 07/10/2025	Criação de tabelas, views e stored procedures	8
07/10/2025 a 30/10/2025	Integração entre Frontend e Backend via JavaScript (Fetch API) e testes iniciais de fluxo de dados	8
30/10/2025 a 13/11/2025	Ajustes de layout, refatoração de código para segurança (Auth/Bcrypt) e implementação de funcionalidades de estoque	6
13/11/2025 a 27/11/2025	Realização de testes finais, correção de bugs e elaboração da documentação técnica final	9
Total		53

- **Extração de Requisitos e Modelagem do Banco:** O trabalho teve início com a estruturação das regras de negócio e a criação do modelo conceitual e lógico. Foram definidas inicialmente 9 entidades essenciais, todas normalizadas até a 3FN para eliminar redundâncias e garantir consistência. Essa fase resultou na elaboração do DER e do modelo lógico.

- **Desenvolvimento das Interfaces Frontend:** As primeiras telas do sistema (login, cadastro e página inicial) foram implementadas com design responsivo e navegação consistente. Para garantir uma experiência fluida em diferentes dispositivos, foram aplicadas media queries e padrões mobile-first.
- **Implementação do Backend e Autenticação:** Em seguida, foi estruturada a camada de backend. As tabelas foram representadas por classes Python (models), e as rotas receberam validações completas. A autenticação de usuários utilizou hashing bcrypt com criptografia bcrypt com cost factor 12, equilibrando segurança e desempenho. Perfis distintos (doador e colaborador) foram incorporados com validação de CPF e CNPJ.
- **Construção de Tabelas, Views e Stored Procedures:** Nesta etapa, todo o banco foi implementado fisicamente no MySQL. Além da adição de tabelas para estoque e autenticação de tokens, foram criadas views estratégicas:
 - *vw_campañas_ativas*
 - *vw_hemocentros_estatísticas*
 - *vw_histórico_doacoes_completo*
 - *vw_próximos_agendamentos*Além disso, também criamos algumas stored procedures:
 - *sp_atualizar_quantidade_campaña*
 - *sp_calcular_proxima_doacao*
 - *sp_listar_campañas_ativas*
 - *sp_relatorio_doacoes_periodo*
 - *sp_verificar_pode_doar*O conjunto de views e stored procedures foi criado para futuramente diminuirmos a complexidade do backend e centralizar as regras de negócio, mas ainda não estão implementados na aplicação.
- **Integração Frontend–Backend:** Com o backend pronto, iniciou-se a integração via Fetch API, garantindo fluxo completo de dados entre formulários, banco e interface.

Foram implementadas rotas RESTful para CRUD de usuários, agendamentos, campanhas e estoques (ex: "O cadastro de usuário envia dados via POST para /api/usuarios, que valida o CPF, aplica bcrypt na senha e retorna um token JWT").

- **Refatorações, Segurança e Módulo de Estoque:** Após os testes iniciais, foram realizados ajustes de layout, melhorias de segurança contra SQL Injection e padronização de headers. O módulo de gerenciamento de estoque foi finalizado com classificação automática dos níveis (Crítico, Baixo e Normal).
- **Testes Finais e Documentação:** Por fim, foram conduzidos testes exploratórios de todos os fluxos críticos: cadastro, login, agendamento, consulta de campanhas e gestão por colaboradores. A documentação técnica foi elaborada, incluindo README, diagramas, instruções de execução e evidências da aplicação funcional.

3. AVALIAÇÃO DAS ATIVIDADES DESENVOLVIDAS

Cumprimento dos objetivos

A análise final do desenvolvimento do sistema HemoCamp evidencia que os objetivos estabelecidos na fase de concepção foram alcançados de maneira consistente e satisfatória. A solução entregue configura uma aplicação funcional de ponta a ponta, cobrindo integralmente o ciclo de gerenciamento da doação de sangue: cadastro e autenticação segura de doadores, colaboradores e hemocentros; agendamento estruturado de coletas; monitoramento do histórico de doações; gestão operacional de campanhas; e controle atualizado do estoque sanguíneo.

Do ponto de vista técnico, o sistema apresenta 11 tabelas implementadas, estruturadas com chaves primárias, estrangeiras, índices, constraints e visões SQL que ampliam a capacidade analítica da plataforma. Além disso, foram atendidos 8 requisitos funcionais principais, garantindo que as funcionalidades essenciais fossem plenamente entregues, testadas e integradas.

No que se refere à camada de serviços, o backend foi estruturado como uma API RESTful robusta, contendo mais de 15 endpoints organizados em rotas modulares, o que assegura escalabilidade, padronização no tratamento de dados e facilidade de manutenção. A comunicação entre frontend e backend ocorre de maneira eficiente, com respostas consistentes e adequadas às operações executadas pelo usuário.

Em síntese, o HemoCamp demonstra maturidade tecnológica e aderência ao problema real que se propõe a solucionar. A arquitetura implementada, aliada à modelagem cuidadosa e à completude dos requisitos, confirma a solidez do produto final entregue.

Pontos fortes da aplicação

Um ponto forte é a modularidade do código, alcançada através do uso de Blueprints no framework Flask. Esta estrutura permitiu uma separação clara de responsabilidades (rotas de usuário, agendamento, hemocentro), facilitando a manutenção e a escalabilidade futura do sistema. Adicionalmente, a interface responsiva, desenvolvida com CSS nativo, assegura a acessibilidade da plataforma em diferentes dispositivos, cumprindo um requisito não funcional essencial para o público-alvo.

Pontos a melhorar e dificuldades

Apesar do sucesso funcional, a aplicação apresenta limitações que denotam oportunidades de melhoria técnica. O ponto mais crítico é a ausência de testes automatizados abrangentes. Embora tenham sido realizados testes unitários e exploratórios manuais, a falta de uma suíte de testes automatizados (utilizando ferramentas como pytest) aumenta o risco de regressões em futuras atualizações e limita a garantia de qualidade de software a longo prazo.

Outra fragilidade identificada refere-se à infraestrutura de deploy. O projeto, embora funcional em ambiente de desenvolvimento e hospedagem gratuita (Railway), ainda não conta com um processo de containerização (Docker) ou pipelines de Integração Contínua/Entrega Contínua (CI/CD). Isso dificulta a portabilidade da aplicação entre diferentes provedores de nuvem e a automação do processo de implantação em escala produtiva. Por fim, funcionalidades de engajamento passivo, como o sistema de notificações automáticas por e-mail para

campanhas, foram mapeadas no banco de dados, mas permaneceram como implementações futuras, o que reduz parcialmente a capacidade imediata de retenção de doadores da plataforma.

Implementações Futuras

Embora o sistema HemoCamp tenha alcançado seus objetivos fundamentais e entregue uma solução funcional e integrada, há um conjunto significativo de aprimoramentos planejados para elevar a maturidade tecnológica do projeto, ampliar sua escalabilidade e fortalecer sua utilidade prática. As implementações futuras propostas incluem:

- **Sistema de notificações por e-mail e SMS**

A criação de um módulo automatizado de envio de notificações permitirá o disparo de lembretes de agendamento, confirmação de doações e avisos de campanhas críticas, fortalecendo o engajamento do doador. A solução pode utilizar serviços como SendGrid, Twilio ou integrações com provedores nacionais.

- **Integração com APIs de geolocalização**

A adição de geolocalização permitirá que o doador identifique o hemocentro mais próximo, visualize rotas e obtenha estimativas de deslocamento. A integração pode ocorrer via Google Maps API, OpenStreetMap (Nominatim) ou plataformas equivalentes.

- **Dashboard analítico para gestores**

Propõe-se o desenvolvimento de um painel administrativo com indicadores-chave (KPIs), gráficos e relatórios gerados em tempo real. Isso permitirá análise do volume de doações, desempenho de campanhas, níveis de estoque e comportamento dos usuários, tornando a plataforma uma ferramenta estratégica para tomada de decisão.

- **Aplicativo móvel nativo (Android/iOS)**

Um app dedicado ampliaria o alcance do HemoCamp e facilitaria o acesso dos doadores às funcionalidades do sistema. A implementação pode ser realizada em

Flutter, React Native ou Kotlin/Swift para maior integração com recursos do dispositivo (push notifications, biometria, localização).

- **Testes automatizados com Pytest**

A implementação de testes unitários, de integração e de regressão com Pytest aumentará a confiabilidade do backend e facilitará futuras expansões. Isso também estabelece boas práticas de engenharia de software e melhora a cobertura de testes.

- **Containerização com Docker**

Containerizar o backend, o banco de dados e serviços auxiliares padronizaria o ambiente de desenvolvimento, simplificaria a implantação e reduziria riscos de incompatibilidade. A criação de um docker-compose.yml permitiria orquestrar toda a aplicação localmente ou em produção.

- **Pipeline de CI/CD**

A criação de pipelines de integração contínua e entrega contínua possibilitará a automação de testes, validação de versionamento, build e deploy da aplicação. Plataformas como GitHub Actions, GitLab CI ou Jenkins podem ser empregadas, aumentando a robustez do ciclo de desenvolvimento.

4. REFERÊNCIAS BIBLIOGRÁFICAS

AGÊNCIA BRASIL. Quatorze em cada mil brasileiros são doadores regulares de sangue. 2023. Disponível em: <https://agenciabrasil.ebc.com.br/saude/noticia/2023-06/quatorze-em-cada-mil-brasileiros-sao-doadores-regulares-de-sangue>. Acesso em: 27 nov. 2025.

Até março deste ano, foram realizadas mais de 25 mil doações no Pará. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/noticias-para-os-estados/para/2024/junho/ate-marco-deste-ano-foram-realizadas-mais-de-25-mil-doacoes-no-pará>>. Acesso em: 08 out. 2025.

BARBOZA, Stephanie Ingrid Souza; COSTA, Francisco José da. Marketing social para doação de sangue: análise da predisposição de novos doadores. **Cadernos de Saúde Pública**, v. 30, p. 1463-1474, 2014.

CARLESSO, Leticia; SANTOS, Cristiane Ferreira dos; GUIMARÃES, Rosane de Fátima da Silva; SILVA, Suzel Lima da; VIERO, Viviani; VIEIRA, Silvani Vargas; GIRARDON-PERLINI, Nara Marilene Oliveira. Estratégias implementadas em hemocentros para aumento da doação de sangue. **Revista Brasileira em Promoção da Saúde**, [S. I.], v. 30, n. 2, 2017. DOI: 10.5020/18061230. 2017.p213. Disponível em: <https://ojs.unifor.br/RBPS/article/view/5873>. Acesso em: 9 out. 2025.

MINISTÉRIO DA SAÚDE. Doação de sangue: dados e indicadores. Brasília: Ministério da Saúde, 2024. Disponível em: <https://www.gov.br/saude/pt-br/composicao/saes/doacao-de-sangue>. Acesso em: 27 nov. 2025.

MESQUITA, N. F.; VAZQUEZ, A. C. S.; DUARTE, M. D. L. C.; da SILVA, D. G.; de MATTOS, L. G. Dificuldades e estratégias relacionadas com a doação de sangue em um serviço de hemoterapia. **Rev Rene**, Fortaleza, v. 22, n. 1, p. e70830, 2021. Disponível em: <https://enfispo.es/servlet/articulo?codigo=8080795>. Acesso em: 8 out. 2025.

MOURA, Aldilene Sobreira de; MOREIRA, Camila Teixeira; MACHADO, Caroline Antero; VASCONCELOS NETO, José Ananias; MACHADO, Maria de Fátima Antero Sousa. Doador de sangue habitual e fidelizado: fatores motivacionais de adesão ao programa. **Revista Brasileira em Promoção da Saúde**, [S. I.], v. 19, n. 2, p. 61–67, 2012. DOI: 10.5020/963. Disponível em: <https://ojs.unifor.br/RBPS/article/view/963>. Acesso em: 8 out. 2025.

REDAÇÃO. **População estimada da RMC supera 3,3 milhões de habitantes e Campinas bate 1,2 milhão**. Disponível em: <<https://horacampinas.com.br/ibge-populacao-estimada-damc-supera-33-milhoes-de-habitantes-e-campinas-bate-12-milhao/>>. Acesso em: 9 out. 2025.

TOLLER, A.; MORO, C.; DORNELLES, Érico; ZANINI, K.; BRUSCHI, M.; HAIGERT, V.; FLECK, J. Campanha de Incentivo à Doação de Sangue. **Disciplinarum Scientia | Saúde**, Santa Maria (RS, Brasil), v. 3, n. 1, p. 73–88, 2016. DOI: 10.37777/853. Disponível em: <https://periodicos.ufn.edu.br/index.php/disciplinarumS/article/view/853>. Acesso em: 6 out. 2025.

5. EVIDÊNCIAS

O código de nosso projeto está organizado no seguinte GitHub:

<https://github.com/0-manda/Hemocentro>

Diagramas:

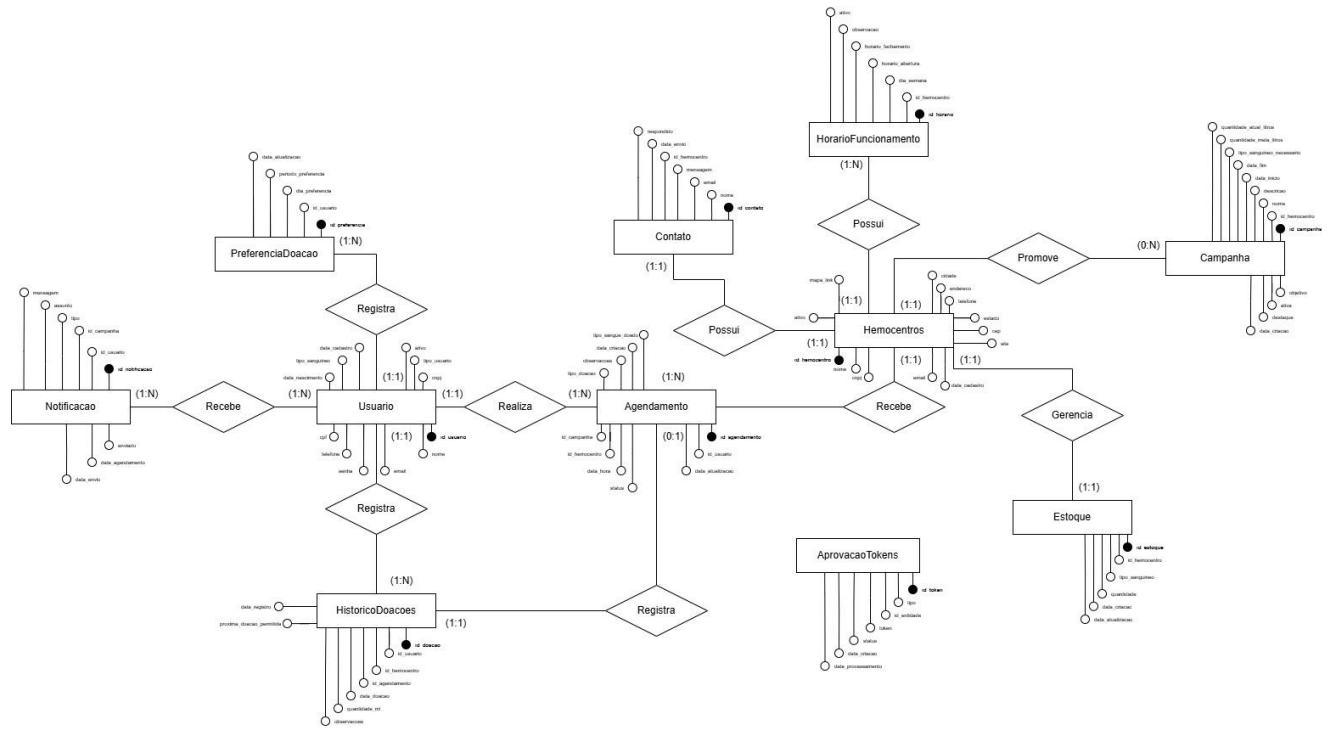


Figura 1: Modelo Conceitual.

Fonte: Elaborado pelos autores (2025).

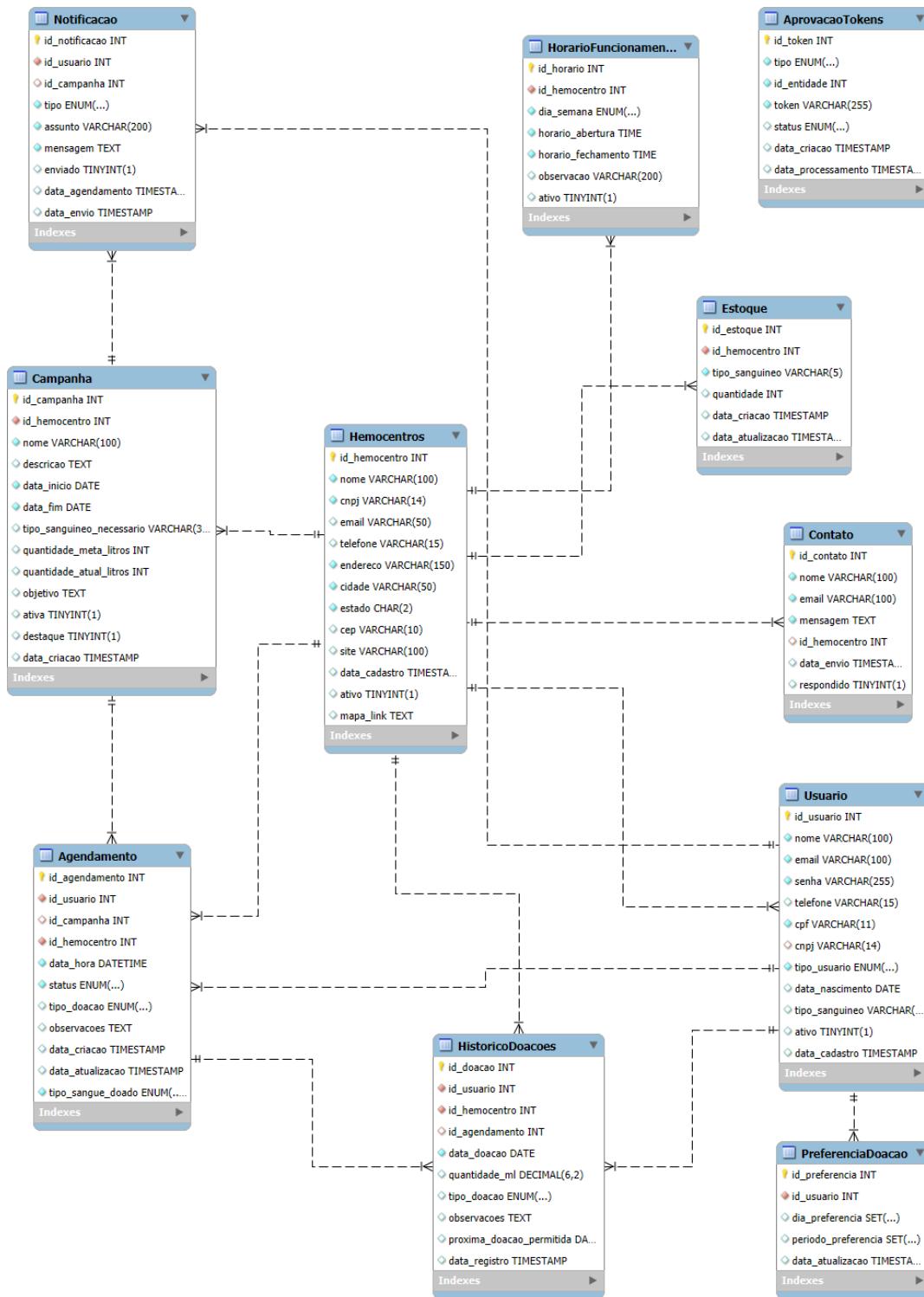


Figura 2: Modelo Lógico.

Fonte: Elaborado pelos autores (2025).

Diagrama da arquitetura:

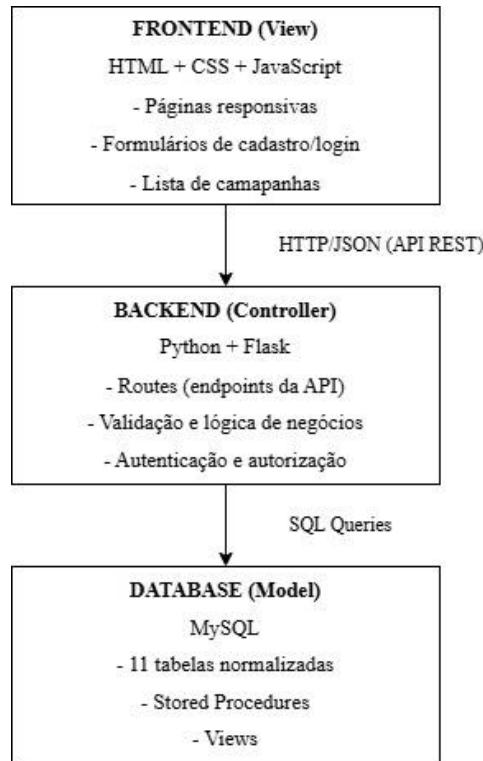


Figura 3: Diagrama de arquitetura.

Fonte: Elaborado pelos autores (2025).

SQL de criação das tabelas:

```

CREATE TABLE Usuario (
    `id_usuario` int NOT NULL AUTO_INCREMENT,
    `nome` varchar(100) NOT NULL,
    `email` varchar(100) NOT NULL,
    `senha` varchar(255) NOT NULL,
    `telefone` varchar(15) DEFAULT NULL,
    `cpf` varchar(11) NOT NULL,
    `cnpj` varchar(14) DEFAULT NULL,
    `tipo_usuario` enum('doador', 'colaborador') NOT NULL DEFAULT 'doador',
    `data_nascimento` date DEFAULT NULL,
    `tipo_sanguineo` varchar(5) DEFAULT NULL,
    `ativo` tinyint(1) DEFAULT '1',
    `data_cadastro` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
)

```

```

PRIMARY KEY (id_usuario),
UNIQUE KEY email (email),
UNIQUE KEY cpf (cpf),
UNIQUE KEY cnpj (cnpj),
KEY idx_cpf (cpf),
KEY idx_usuario_cnpj (cnpj),
CONSTRAINT fk_usuario_hemocentro FOREIGN KEY (cnpj) REFERENCES
Hemocentros (cnpj)

);

```

```

CREATE TABLE Hemocentros (

`id_hemocentro` int NOT NULL AUTO_INCREMENT,
`nome` varchar(100) NOT NULL,
`cnpj` varchar(14) NOT NULL,
`email` varchar(50) DEFAULT NULL,
`telefone` varchar(15) DEFAULT NULL,
`endereco` varchar(150) NOT NULL,
`cidade` varchar(50) NOT NULL DEFAULT 'Campinas',
`estado` char(2) NOT NULL DEFAULT 'SP',
`cep` varchar(10) DEFAULT NULL,
`site` varchar(100) DEFAULT NULL,
`data_cadastro` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
`ativo` tinyint(1) DEFAULT '0',
`mapa_link` text,
PRIMARY KEY (`id_hemocentro`),
UNIQUE KEY `cnpj` (`cnpj`),
KEY `idx_cidade` (`cidade`),
KEY `idx_cnpj` (`cnpj`)

);

```

```

CREATE TABLE HorarioFuncionamento (
    `id_horario` int NOT NULL AUTO_INCREMENT,
    `id_hemocentro` int NOT NULL,
    `dia_semana` enum('segunda','terca','quarta','quinta','sexta','sabado','domingo') NOT
    NULL,
    `horario_abertura` time NOT NULL,
    `horario_fechamento` time NOT NULL,
    `observacao` varchar(200) DEFAULT NULL,
    `ativo` tinyint(1) DEFAULT '1',
    PRIMARY KEY (`id_horario`),
    UNIQUE KEY `unique_horario` (`id_hemocentro`, `dia_semana`),
    CONSTRAINT `HorarioFuncionamento_ibfk_1` FOREIGN KEY (`id_hemocentro`)
    REFERENCES Hemocentros (`id_hemocentro`) ON DELETE CASCADE
);


```

```

CREATE TABLE Campanha (
    `id_campaña` int NOT NULL AUTO_INCREMENT,
    `id_hemocentro` int NOT NULL,
    `nome` varchar(100) NOT NULL,
    `descricao` text,
    `data_inicio` date NOT NULL,
    `data_fim` date NOT NULL,
    `tipo_sanguíneo_necessário` varchar(30) DEFAULT NULL,
    `quantidade_meta_litros` int DEFAULT '0',
    `quantidade_atual_litros` int DEFAULT '0',
    `objetivo` text, ativa tinyint(1) DEFAULT '1',
    `destaque` tinyint(1) DEFAULT '0',
    `data_criacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP,

```

```

        PRIMARY KEY (`id_campanha`),
        KEY `idx_ativa` (`ativa`),
        KEY `idx_datas` (`data_inicio`, `data_fim`),
        KEY `idx_hemocentro` (`id_hemocentro`),
        CONSTRAINT `Campanha_ibfk_1` FOREIGN KEY (`id_hemocentro`) REFERENCES
        Hemocentros (`id_hemocentro`) ON DELETE CASCADE

);

```

```

CREATE TABLE Agendamento (
    `id_agendamento` int NOT NULL AUTO_INCREMENT,
    `id_usuario` int NOT NULL,
    `id_campanha` int DEFAULT NULL,
    `id_hemocentro` int NOT NULL,
    `data_hora` datetime NOT NULL,
    `status` enum('pendente','confirmado','realizado','cancelado','nao_compareceu') NOT
    NULL DEFAULT 'pendente',
    `tipo_doacao` enum('espontanea','campanha','primeira_vez') DEFAULT 'espontanea',
    `observacoes` text,
    `data_criacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
    `data_atualizacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON
    UPDATE CURRENT_TIMESTAMP,
    `tipo_sangue_doador` enum('sangue_total','plaquetas','plasma','aferese') NOT NULL
    DEFAULT 'sangue_total',
    PRIMARY KEY (`id_agendamento`),
    KEY `id_campanha` (`id_campanha`),
    KEY `id_hemocentro` (`id_hemocentro`),
    KEY `idx_usuario` (`id_usuario`),
    KEY `idx_data_hora` (`data_hora`),
    KEY `idx_status` (`status`),
    CONSTRAINT `Agendamento_ibfk_1` FOREIGN KEY (`id_usuario`) REFERENCES

```

```

        Usuario (`id_usuario`) ON DELETE CASCADE,
        CONSTRAINT `Agendamento_ibfk_2` FOREIGN KEY (`id_campanha`)
        REFERENCES Campanha (`id_campanha`) ON DELETE SET NULL,
        CONSTRAINT `Agendamento_ibfk_3` FOREIGN KEY (`id_hemocentro`)
        REFERENCES Hemocentros (`id_hemocentro`) ON DELETE CASCADE
    );

```

```

CREATE TABLE HistoricoDoacoes (
    `id_doacao` int NOT NULL AUTO_INCREMENT,
    `id_usuario` int NOT NULL,
    `id_hemocentro` int NOT NULL,
    `id_agendamento` int DEFAULT NULL,
    `data_doacao` date NOT NULL,
    `quantidade_ml` decimal(6,2) DEFAULT '450.00',
    `tipo_doacao` enum('sangue_total', 'plaquetas', 'plasma', 'aferese') DEFAULT
    'sangue_total',
    `observacoes` text,
    `proxima_doacao_permitida` date DEFAULT NULL,
    `data_registro` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
    PRIMARY KEY (`id_doacao`),
    KEY `id_agendamento` (`id_agendamento`),
    KEY `idx_usuario` (`id_usuario`),
    KEY `idx_data_doacao` (`data_doacao`),
    KEY `idx_hemocentro` (`id_hemocentro`),
    CONSTRAINT `HistoricoDoacoes_ibfk_1` FOREIGN KEY (`id_usuario`)
    REFERENCES Usuario (`id_usuario`) ON DELETE CASCADE,
    CONSTRAINT `HistoricoDoacoes_ibfk_2` FOREIGN KEY (`id_hemocentro`)
    REFERENCES Hemocentros (`id_hemocentro`) ON DELETE CASCADE,
    CONSTRAINT `HistoricoDoacoes_ibfk_3` FOREIGN KEY (`id_agendamento`)
    REFERENCES Agendamento (`id_agendamento`) ON DELETE SET NULL
)

```

```
);
```

```
CREATE TABLE Notificacao (
    `id_notificacao` int NOT NULL AUTO_INCREMENT,
    `id_usuario` int NOT NULL,
    `id_campaña` int DEFAULT NULL,
    `tipo` enum('campanha', 'agendamento', 'lembrete', 'confirmacao', 'urgente', 'geral')
        NOT NULL,
    `assunto` varchar(200) NOT NULL,
    `mensagem` text NOT NULL,
    `enviado` tinyint(1) DEFAULT '0',
    `data_agendamento` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
    `data_envio` timestamp NULL DEFAULT NULL,
    PRIMARY KEY (`id_notificacao`),
    KEY `id_campaña` (`id_campaña`),
    KEY `idx_enviado` (`enviado`),
    KEY `idx_tipo` (`tipo`),
    KEY `idx_usuario` (`id_usuario`),
    CONSTRAINT `Notificacao_ibfk_1` FOREIGN KEY (`id_usuario`) REFERENCES
    Usuario (`id_usuario`) ON DELETE CASCADE,
    CONSTRAINT `Notificacao_ibfk_2` FOREIGN KEY (`id_campaña`) REFERENCES
    Campanha (`id_campaña`) ON DELETE SET NULL
);
```

```
CREATE TABLE PreferenciaDoacao (
```

```
    `id_preferencia` int NOT NULL AUTO_INCREMENT,
    `id_usuario` int NOT NULL,
    `dia_preferencia` set('segunda','terça','quarta','quinta','sexta','sábado','domingo')
```

```
        DEFAULT NULL,  
        `periodo_preferencia` set('manha','tarde','noite') DEFAULT NULL,  
        `data_atualizacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON  
        UPDATE CURRENT_TIMESTAMP,  
        PRIMARY KEY (`id_preferencia`),  
        UNIQUE KEY `unique_usuario` (`id_usuario`),  
        CONSTRAINT `PreferenciaDoacao_ibfk_1` FOREIGN KEY (`id_usuario`)  
        REFERENCES Usuario (`id_usuario`) ON DELETE CASCADE  
  
);
```

```
CREATE TABLE Contato (  
  
        `id_contato` int NOT NULL AUTO_INCREMENT,  
        `nome` varchar(100) NOT NULL,  
        `email` varchar(100) NOT NULL,  
        `mensagem` text NOT NULL,  
        `id_hemocentro` int DEFAULT NULL,  
        `data_envio` timestamp NULL DEFAULT CURRENT_TIMESTAMP,  
        `respondido` tinyint(1) DEFAULT '0',  
        PRIMARY KEY (`id_contato`),  
        KEY `id_hemocentro` (`id_hemocentro`),  
        CONSTRAINT `Contato_ibfk_1` FOREIGN KEY (`id_hemocentro`) REFERENCES  
        Hemocentros (`id_hemocentro`) ON DELETE SET NULL  
  
);
```

```
CREATE TABLE Estoque (  
  
        `id_estoque` int NOT NULL AUTO_INCREMENT,  
        `id_hemocentro` int NOT NULL,  
        `tipo_sanguineo` varchar(5) NOT NULL,  
        `quantidade` int DEFAULT '0',
```

```

`data_criacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
`data_atualizacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
PRIMARY KEY (`id_estoque`),
UNIQUE KEY `unique_estoque` (`id_hemocentro`, `tipo_sanguineo`),
KEY `idx_hemocentro` (`id_hemocentro`),
KEY `idx_tipo_sanguineo` (`tipo_sanguineo`),
CONSTRAINT `Estoque_ibfk_1` FOREIGN KEY (`id_hemocentro`) REFERENCES
Hemocentros (`id_hemocentro`) ON DELETE CASCADE
);

```

```

CREATE TABLE AprovacaoTokens (
`id_token` int NOT NULL AUTO_INCREMENT,
`tipo` enum('doador', 'colaborador', 'hemocentro') NOT NULL,
`id_entidade` int NOT NULL,
`token` varchar(255) NOT NULL,
`status` enum('pendente', 'aprovado', 'rejeitado') DEFAULT 'pendente',
`data_criacao` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
`data_processamento` timestamp NULL DEFAULT NULL,
PRIMARY KEY (`id_token`),
UNIQUE KEY `token` (`token`),
KEY `idx_token` (`token`),
KEY `idx_status` (`status`),
KEY `idx_entidade` (`tipo`, `id_entidade`)
);

```

SQL das inserções iniciais

Inicialmente inserimos dados fictícios para visualização do funcionamento da aplicação.

INSERT INTO Usuario VALUES

```
(1, 'João Pedro Silva', 'joao.silva@email.com',
'$2b$12$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 98765-4321', '00000000000', NULL, 'doador',
'1990-05-15', 'O+', 1, '2025-11-22 20:58:07'),  
  
(2, 'Maria Santos Oliveira', 'maria.santos@email.com',
'$2b$12$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 99876-5432', '11111111111', NULL, 'doador',
'1985-08-20', 'A+', 1, '2025-11-22 20:58:07'),  
  
(3, 'Pedro Costa Lima', 'pedro.lima@email.com',
'$2b$12$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 97654-3210', '22222222222', NULL, 'doador',
'1995-12-10', 'B-', 1, '2025-11-22 20:58:07'),  
  
(4, 'Ana Paula Souza', 'ana.souza@email.com',
'$2b$12$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 96543-2109', '33333333333', NULL, 'doador',
'1992-03-25', 'O-', 1, '2025-11-22 20:58:07'),  
  
(5, 'Carlos Eduardo Alves', 'carlos.alves@email.com',
'$2b$12$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 95432-1098', '44444444444', NULL, 'doador',
'1988-11-08', 'AB+', 1, '2025-11-22 20:58:07'),  
  
(6, 'Juliana Martins', 'juliana.martins@email.com',
'$2b$12$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 94321-0987', '55555555555', NULL, 'doador',
'1993-07-18', 'A-', 1, '2025-11-22 20:58:07'),
```

(7, 'Roberto Fernandes', 'roberto.fernandes@email.com',
'\$2b\$12\$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 93210-9876', '666666666666', NULL, 'doador',
'1987-09-30', 'B+', 1, '2025-11-22 20:58:07'),

(8, 'Fernanda Costa', 'fernanda.costa@email.com',
'\$2b\$12\$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 92109-8765', '777777777777', NULL, 'doador',
'1991-01-14', 'O+', 1, '2025-11-22 20:58:07'),

(9, 'Lucas Ribeiro', 'lucas.ribeiro@email.com',
'\$2b\$12\$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 91098-7654', '88888888888', NULL, 'doador',
'1994-06-22', 'A+', 1, '2025-11-22 20:58:07'),

(10, 'Beatriz Mendes', 'beatriz.mendes@email.com',
'\$2b\$12\$LQv3c1yqBWVHxkd0LHAkCOYz6TtxMQJqhN8/LewY5NU7qfLq8GF/i',
'(19) 90987-6543', '99999999999', NULL, 'doador',
'1989-04-05', 'AB-', 1, '2025-11-22 20:58:07');

INSERT INTO Hemocentros VALUES

(1, 'Hemocentro Unicamp', '0000000000000000', ' contato@hemocentro.unicamp.br', '(19)
3521-8740', 'Rua Carlos Chagas, 480 - Cidade Universitária', 'Campinas', 'SP', '13083-878',
'https://www.hemocentro.unicamp.br', '2025-11-22 20:57:52', 1, NULL);

INSERT INTO HorarioFuncionamento VALUES

(1, 1, 'segunda', '07:30:00', '17:00:00', 'Doação de sangue total e plaquetas', 1),
(2, 1, 'terca', '07:30:00', '17:00:00', 'Doação de sangue total e plaquetas', 1),
(3, 1, 'quarta', '07:30:00', '17:00:00', 'Doação de sangue total e plaquetas', 1),
(4, 1, 'quinta', '07:30:00', '17:00:00', 'Doação de sangue total e plaquetas', 1),

(5, 1, 'sexta', '07:30:00', '17:00:00', 'Doação de sangue total e plaquetas', 1),
(6, 1, 'sabado', '08:00:00', '12:00:00', 'Apenas sangue total', 1);

INSERT INTO Campanha VALUES

(1, 1, 'Junho Vermelho 2025', 'Campanha nacional de conscientização sobre a importância da doação de sangue. Durante todo o mês de junho, realizamos ações especiais e buscamos novos doadores para reforçar nossos estoques.', '2025-06-01', '2026-01-26', 'Todos', 500, 237, 'Aumentar o estoque de sangue para o período de inverno, quando as doações costumam cair significativamente.', 1, 1, '2025-11-22 20:58:06'),
(2, 1, 'Doação Urgente - Tipo O Negativo', 'URGENTE: Precisamos de doadores tipo O- para atender demanda hospitalar crítica. O tipo O- é doador universal e salva vidas em emergências.', '2025-10-20', '2026-01-26', 'O-', 50, 32, 'Suprir a demanda urgente de sangue O- para cirurgias de emergência e pacientes politraumatizados.', 1, 1, '2025-11-22 20:58:06');

INSERT INTO Agendamento VALUES

(1, 1, 1, 1, '2025-11-05 09:00:00', 'confirmado', 'campanha', 'Primeira doação do doador', '2025-11-22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),
(2, 2, 1, 1, '2025-11-06 10:30:00', 'confirmado', 'campanha', NULL, '2025-11-22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),
(3, 3, 2, 1, '2025-11-07 14:00:00', 'pendente', 'campanha', 'Doador tipo O-', '2025-11-22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),
(4, 4, 2, 1, '2025-11-08 08:30:00', 'confirmado', 'campanha', 'Doador tipo O-', '2025-11-22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),
(6, 6, NULL, 1, '2025-11-10 15:00:00', 'pendente', 'espontanea', NULL, '2025-11-22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),
(9, 9, 1, 1, '2025-10-28 10:00:00', 'realizado', 'campanha', NULL, '2025-11-22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),
(11, 1, NULL, 1, '2025-08-15 09:00:00', 'realizado', 'espontanea', 'Doação anterior', '2025-11-

```
22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),  
(12, 2, NULL, 1, '2025-07-10 11:00:00', 'realizado', 'espontanea', 'Doação anterior', '2025-11-  
22 20:58:12', '2025-11-22 20:58:12', 'sangue_total'),  
(13, 1, 1, 1, '2025-11-05 09:00:00', 'confirmado', 'campanha', 'Primeira doação do doador',  
'2025-11-22 20:59:42', '2025-11-22 20:59:42', 'sangue_total'),  
(14, 2, 1, 1, '2025-11-06 10:30:00', 'confirmado', 'campanha', NULL, '2025-11-22 20:59:42',  
'2025-11-22 20:59:42', 'sangue_total'),  
(15, 3, 2, 1, '2025-11-07 14:00:00', 'pendente', 'campanha', 'Doador tipo O-', '2025-11-22  
20:59:42', '2025-11-22 20:59:42', 'sangue_total'),  
(16, 4, 2, 1, '2025-11-08 08:30:00', 'confirmado', 'campanha', 'Doador tipo O-', '2025-11-22  
20:59:42', '2025-11-22 20:59:42', 'sangue_total'),  
(18, 6, NULL, 1, '2025-11-10 15:00:00', 'pendente', 'espontanea', NULL, '2025-11-22  
20:59:42', '2025-11-22 20:59:42', 'sangue_total');
```

INSERT INTO HistoricoDoacoes VALUES

```
(1, 1, 1, 11, '2025-08-15', 450.00, 'sangue_total', 'Doação realizada com sucesso', '2025-10-  
14', '2025-11-22 20:59:44'),  
(2, 2, 1, 12, '2025-07-10', 450.00, 'sangue_total', 'Doação realizada com sucesso', '2025-09-  
08', '2025-11-22 20:59:44'),  
(4, 9, 1, 9, '2025-10-28', 450.00, 'sangue_total', 'Doação realizada com sucesso', '2025-12-  
27', '2025-11-22 20:59:44'),  
(6, 3, 1, NULL, '2025-09-05', 450.00, 'sangue_total', 'Doação espontânea', '2025-11-04',  
'2025-11-22 20:59:44'),  
(7, 4, 1, NULL, '2025-08-20', 450.00, 'sangue_total', 'Doação espontânea', '2025-10-19',  
'2025-11-22 20:59:44'),  
(9, 6, 1, NULL, '2025-05-10', 450.00, 'sangue_total', 'Doação realizada', '2025-07-09', '2025-  
11-22 20:59:44');
```

INSERT INTO Notificacao VALUES

(1, 1, 1, 'confirmacao', 'Agendamento Confirmado', 'Seu agendamento para doação no dia 05/11/2025 às 09:00 foi confirmado. Hemocentro Unicamp.', 1, '2025-11-22 20:59:49', '2025-10-30 10:00:00'),
(2, 2, 1, 'lembrete', 'Lembrete de Doação', 'Lembramos que você tem um agendamento amanhã às 10:30 no Hemocentro Unicamp.', 1, '2025-11-22 20:59:49', '2025-11-05 18:00:00'),
(3, 3, 2, 'urgente', 'Doação Urgente - Tipo O-', 'Precisamos urgentemente de sua doação! Tipo O- em falta crítica.', 1, '2025-11-22 20:59:49', '2025-10-25 09:00:00'),
(4, 4, 2, 'confirmacao', 'Agendamento Confirmado', 'Seu agendamento foi confirmado para 08/11/2025 às 08:30.', 1, '2025-11-22 20:59:49', '2025-10-28 14:00:00'),
(5, 5, 1, 'campanha', 'Nova Campanha: Junho Vermelho', 'Participe da campanha Junho Vermelho! Doe sangue e salve vidas.', 1, '2025-11-22 20:59:49', '2025-06-01 08:00:00'),
(6, 6, NULL, 'geral', 'Bem-vindo ao HemoCamp!', 'Obrigado por se cadastrar! Você pode agendar sua doação a qualquer momento.', 1, '2025-11-22 20:59:49', '2025-10-20 12:00:00'),
(8, 8, NULL, 'geral', 'Primeira Doação - Orientações', 'Confira as orientações para sua primeira doação de sangue.', 1, '2025-11-22 20:59:49', '2025-10-22 15:00:00');

INSERT INTO PreferenciaDoacao VALUES

(1, 1, 'segunda,quarta,sexta', 'manha', '2025-11-22 20:59:47'),
(2, 2, 'terca,quinta', 'tarde', '2025-11-22 20:59:47'),
(3, 3, 'sabado', 'manha', '2025-11-22 20:59:47'),
(4, 4, 'segunda,quarta', 'manha,tarde', '2025-11-22 20:59:47'),
(5, 5, 'terca,quinta,sexta', 'tarde', '2025-11-22 20:59:47'),
(6, 6, 'segunda,sexta', 'manha', '2025-11-22 20:59:47'),
(7, 7, 'quarta', 'tarde,noite', '2025-11-22 20:59:47'),
(8, 8, 'terca,quinta', 'manha,tarde', '2025-11-22 20:59:47'),
(9, 9, 'segunda,quarta,sexta', 'manha', '2025-11-22 20:59:47'),
(10, 10, 'sabado', 'manha', '2025-11-22 20:59:47');

```
INSERT INTO Contato VALUES
```

```
(1, 'Amanda Oliveira', 'amanda.oliveira@email.com', 'Gostaria de saber quais são os  
requisitos para doar sangue pela primeira vez.', 1, '2025-11-22 20:59:54', 1),  
(2, 'Gabriel Santos', 'gabriel.santos@email.com', 'Posso doar sangue se tiver tomado vacina  
na semana passada?', 1, '2025-11-22 20:59:54', 1),  
(3, 'Larissa Costa', 'larissa.costa@email.com', 'Qual o horário de funcionamento aos  
sábados?', 2, '2025-11-22 20:59:54', 1),  
(4, 'Rafael Lima', 'rafael.lima@email.com', 'Preciso agendar ou posso ir direto ao  
hemocentro?', 3, '2025-11-22 20:59:54', 0),  
(5, 'Camila Rodrigues', 'camila.rodrigues@email.com', 'Vocês aceitam doação de  
plaquetas?', 1, '2025-11-22 20:59:54', 0),  
(6, 'Bruno Almeida', 'bruno.almeida@email.com', 'Quanto tempo devo esperar entre uma  
doação e outra?', NULL, '2025-11-22 20:59:54', 1);
```

```
INSERT INTO Estoque VALUES
```

```
(1, 1, 'A+', 45, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(2, 1, 'A-', 12, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(3, 1, 'B+', 28, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(4, 1, 'B-', 8, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(5, 1, 'AB+', 15, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(6, 1, 'AB-', 5, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(7, 1, 'O+', 52, '2025-11-25 22:30:48', '2025-11-25 22:30:48'),  
(8, 1, 'O-', 18, '2025-11-25 22:30:48', '2025-11-25 22:30:48');
```

SQL de criação das views:

```
CREATE VIEW vw_campanhas_ativas AS  
SELECT
```

```

c.id_campanha,
c.nome AS campanha_nome,
c.descricao,
c.data_inicio,
c.data_fim,
c.tipo_sanguineo_necessario,
c.quantidade_meta_litros,
c.quantidade_atual_litros,
ROUND((c.quantidade_atual_litros / c.quantidade_meta_litros * 100), 1) AS
percentual_atingido,
DATEDIFF(c.data_fim, CURDATE()) AS dias_restantes,
c.destaque,
h.id_hemocentro,
h.nome AS hemocentro_nome,
h.endereco,
h.cidade,
h.telefone,
h.email

```

```

FROM Campanha c INNER JOIN Hemocentros h ON c.id_hemocentro = h.id_hemocentro
WHERE c.ativa = TRUE AND c.data_fim >= CURDATE()
ORDER BY c.destaque DESC, c.data_inicio ASC;

```

```
CREATE VIEW vw_historico_doacoes_completo AS
```

```
SELECT
```

```

hd.id_doacao,
hd.data_doacao,
hd.quantidade_ml,
hd.tipo_doacao,
hd.proxima_doacao_permitida,
u.id_usuario,
u.nome AS doador_nome,
u.email AS doador_email,
u.tipo_sanguineo,
h.id_hemocentro,
h.nome AS hemocentro_nome,
h.cidade

```

```

FROM HistoricoDoacoes hd INNER JOIN Usuario u ON hd.id_usuario = u.id_usuario INNER
JOIN Hemocentros h ON hd.id_hemocentro = h.id_hemocentro
ORDER BY hd.data_doacao DESC;

```

```
CREATE VIEW vw_proximos_agendamentos AS
```

```
SELECT
```

```
a.id_agendamento,  
a.data_hora,  
a.status,  
a.tipo_doacao,  
u.nome AS doador_nome,  
u.email AS doador_email,  
u.telefone AS doador_telefone,  
u.tipo_sanguineo,  
h.nome AS hemocentro_nome,  
h.endereco,  
h.telefone AS hemocentro_telefone,  
c.nome AS campanha_nome
```

```
FROM Agendamento a INNER JOIN Usuario u ON a.id_usuario = u.id_usuario INNER JOIN  
Hemocentros h ON a.id_hemocentro = h.id_hemocentro LEFT JOIN Campanha c ON  
a.id_campanha = c.id_campanha  
WHERE a.data_hora >= NOW() AND a.status IN ('pendente', 'confirmado')  
ORDER BY a.data_hora ASC;
```

```
CREATE VIEW vw_hemocentros_estatisticas AS
```

```
SELECT
```

```
h.id_hemocentro,  
h.nome,  
h.endereco,  
h.cidade,  
h.telefone,  
h.email,  
h.site,  
COUNT(DISTINCT hd.id_doacao) AS total_doacoes,  
COUNT(DISTINCT c.id_campanha) AS total_campañas,  
COUNT(DISTINCT CASE WHEN c.ativa = TRUE AND c.data_fim >= CURDATE()  
THEN c.id_campanha END ) AS campañas_ativas
```

```
FROM Hemocentros h LEFT JOIN HistoricoDoacoes hd ON h.id_hemocentro =  
hd.id_hemocentro LEFT JOIN Campanha c ON h.id_hemocentro = c.id_hemocentro  
WHERE h.ativo = TRUE  
GROUP BY
```

```
h.id_hemocentro,  
h.nome,  
h.endereco,
```

```
h.cidade,  
h.telefone,  
h.email,  
h.site;
```

SQL das stored procedures:

```
DELIMITER //  
CREATE PROCEDURE sp_atualizar_quantidade_campanha(  
    IN p_id_campanha INT  
)  
BEGIN  
    DECLARE v_total_ml DECIMAL(10,2);  
    DECLARE v_total_litros INT;  
  
    -- Calcular total de ml doados para esta campanha  
    SELECT COALESCE(SUM(hd.quantidade_ml), 0)  
    INTO v_total_ml  
    FROM HistoricoDoacoes hd  
    INNER JOIN Agendamento a ON hd.id_agendamento = a.id_agendamento  
    WHERE a.id_campanha = p_id_campanha  
        AND a.status = 'realizado';  
  
    -- Converter ml para litros  
    SET v_total_litros = FLOOR(v_total_ml / 1000);  
  
    -- Atualizar campanha  
    UPDATE Campanha  
    SET quantidade_atual_litros = v_total_litros  
    WHERE id_campanha = p_id_campanha;  
END //  
DELIMITER ;  
  
DELIMITER //  
CREATE PROCEDURE sp_calcular_proxima_doacao(  
    IN p_data_doacao DATE,  
    IN p_tipo_doacao VARCHAR(20),  
    OUT p_proxima_data DATE  
)  
BEGIN  
    DECLARE v_dias_espera INT;  
  
    -- Sangue total: 60 dias  
    -- Plaquetas: 48 horas (2 dias)  
    -- Plasma: 48 horas (2 dias)  
  
    IF p_tipo_doacao = 'sangue_total' THEN
```

```

        SET v_dias_espera = 60;
ELSEIF p_tipo_doacao = 'plaquetas' THEN
        SET v_dias_espera = 2;
ELSE
        SET v_dias_espera = 2;
END IF;

        SET p_proxima_data = DATE_ADD(p_data_doacao, INTERVAL v_dias_espera DAY);
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE sp_listar_campanhas_ativas()
BEGIN
SELECT
    c.id_campaña,
    c.nome,
    c.descrição,
    c.data_inicio,
    c.data_fim,
    c.tipo_sanguíneo_necessário,
    c.quantidade_meta_litros,
    c.quantidade_atual_litros,
    ROUND((c.quantidade_atual_litros / c.quantidade_meta_litros * 100), 1) AS percentual_atingido,
    DATEDIFF(c.data_fim, CURDATE()) AS dias_restantes,
    c.destaque,
    h.nome AS hemocentro_nome,
    h.endereço AS hemocentro_endereço,
    h.telefone AS hemocentro_telefone
FROM Campanha c
INNER JOIN Hemocentros h ON c.id_hemocentro = h.id_hemocentro
WHERE c.ativa = TRUE
    AND c.data_fim >= CURDATE()
ORDER BY c.destaque DESC, c.data_inicio ASC;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE sp_relatorio_doacoes_periodo(
    IN p_data_inicio DATE,
    IN p_data_fim DATE
)
BEGIN
SELECT
    h.nome AS hemocentro,
    COUNT(hd.id_doacao) AS total_doacoes,
    SUM(hd.quantidade_ml) AS total_ml,
    ROUND(SUM(hd.quantidade_ml) / 1000, 2) AS total_litros,
    COUNT(DISTINCT hd.id_usuario) AS doadores_unicos
FROM Hemocentros h

```

```

LEFT JOIN HistoricoDoacoes hd ON h.id_hemocentro = hd.id_hemocentro
    AND hd.data_doacao BETWEEN p_data_inicio AND p_data_fim
WHERE h.ativo = TRUE
GROUP BY h.id_hemocentro, h.nome
ORDER BY total_doacoes DESC;
END //
DELIMITER ;

DELIMITER //
CREATE PROCEDURE sp_verificar_pode_doar(
    IN p_id_usuario INT,
    OUT p_pode_doar BOOLEAN,
    OUT p_mensagem VARCHAR(200)
)
BEGIN
    DECLARE v_ultima_doacao DATE;
    DECLARE v_proxima_permitida DATE;

    -- Buscar última doação
    SELECT MAX(data_doacao), MAX(proxima_doacao_permitida)
    INTO v_ultima_doacao, v_proxima_permitida
    FROM HistoricoDoacoes
    WHERE id_usuario = p_id_usuario;

    -- Se nunca doou, pode doar
    IF v_ultima_doacao IS NULL THEN
        SET p_pode_doar = TRUE;
        SET p_mensagem = 'Você pode doar! Esta será sua primeira doação。';
    -- Se já passou do período de espera
    ELSEIF CURDATE() >= v_proxima_permitida THEN
        SET p_pode_doar = TRUE;
        SET p_mensagem = 'Você pode doar novamente!';
    -- Se ainda está no período de espera
    ELSE
        SET p_pode_doar = FALSE;
        SET p_mensagem = CONCAT('Você poderá doar novamente a partir de ',
        DATE_FORMAT(v_proxima_permitida, '%d/%m/%Y'));
    END IF;
END //
DELIMITER ;

```

Captura de telas da aplicação final:



Figura 4: Tela de início.

Fonte: Elaborado pelos autores (2025).

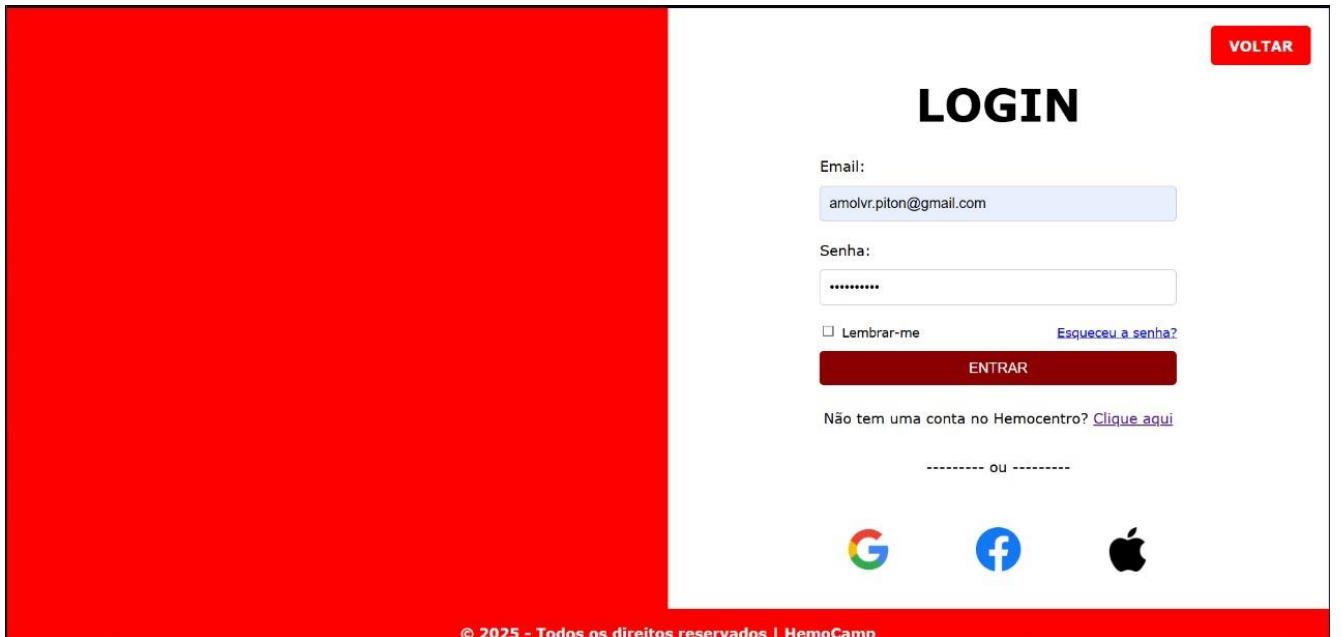


Figura 5: Tela de login.

Fonte: Elaborado pelos autores (2025).



Figura 6: Tela do usuário colaborador.

Fonte: Elaborado pelos autores (2025).



Figura 7: Tela das campanhas cadastradas.

Fonte: Elaborado pelos autores (2025).



Figura 8: Tela da parte inferior das campanhas cadastradas.

Fonte: Elaborado pelos autores (2025).

The screenshot shows a red header bar at the top. Below it, there is a white form titled "Área Administrativa" (Administrative Area) in red. The form is for creating a new campaign and includes fields for "Nome da Campanha" (Campaign Name), "Descrição" (Description), "Tipo Sanguíneo Necessário" (Required Blood Type), "Meta (litros)" (Target (liters)), "Objetivo" (Objective), "Objetivo central da Campanha" (Central Objective of the Campaign), "Início" (Start Date), "Fim" (End Date), and a "CRIAR" (Create) button. A red footer bar at the bottom contains the copyright notice "© 2025 - Todos os direitos reservados | HemoCamp".

Figura 9: Tela do cadastro das campanhas.

Fonte: Elaborado pelos autores (2025).

The screenshot shows a web page titled "Hemocentros disponíveis". It lists two registered hemocenters:

- Hemocentro Caravelas do Mar**
 - Email: caravelas@caravelas.com
 - Telefone: 11970904623
 - Localização: Av. Almeida Garret, 1953, Taquaral, Campinas - SP, CEP: 13087291
 - A note: **⚠ Horários não cadastrados**
 - Buttons: Visitar, Agendar
- Hemocentro PUC**
 - Email: am.olvrnw@gmail.com
 - Telefone: 19987654321
 - Localização: Rua das Acácias, 745 - Jardim Primavera, Campinas - SP, CEP: 13080260

Figura 10: Tela dos hemocentros cadastrados.

Fonte: Elaborado pelos autores (2025).

This screenshot shows the detailed information for the "Hemocentro Unicamp" entry, located at the bottom of the previous page's content.

Hemocentro Unicamp

Email: ProjetoHemoCamp@gmail.com
Telefone: (19) 3521-8740
Localização: Rua Carlos Chagas, 480 - Cidade Universitária, Campinas - SP, CEP: 13083-878
Site: <https://www.hemocentro.unicamp.br>

⌚ Horários de Funcionamento:

Segunda-feira: 07:30:00 - 17:00:00
Terça-feira: 07:30:00 - 17:00:00
Quarta-feira: 07:30:00 - 17:00:00
[+ Ver todos os horários](#)

FECHADO

Buttons: Visitar, Agendar

At the bottom of the page, there is a red footer bar with the text: © 2025 - Todos os direitos reservados | HemoCamp

Figura 11: Tela da parte inferior dos hemocentros cadastrados.

Fonte: Elaborado pelos autores (2025).



Figura 12: Tela de contato.

Fonte: Elaborado pelos autores (2025).



Figura 13: Tela das notícias.

Fonte: Elaborado pelos autores (2025).

HOME HEMOCENTRO CONTATO SOBRE NOTICIAS CAMPANHA USUARIO

NOSSA HISTÓRIA:

O problema inicial:

Em 2025, percebemos que encontrar um hemocentro para doar sangue em Campinas era uma tarefa confusa e trabalhosa. Os sites não eram práticos, as informações sobre doação eram difíceis de achar e muitos hemocentros não conseguiam alcançar novos doadores.

Nossa solução:

Figura 14: Tela da nossa história.

Fonte: Elaborado pelos autores (2025).

VOLTAR

CADASTRO DE HEMOCENTRO

Nome do Hemocentro:

Nome fantasia ou razão social

CNPJ:

Apenas números (14 dígitos)

Email:

contato@hemocentro.com

Telefone:

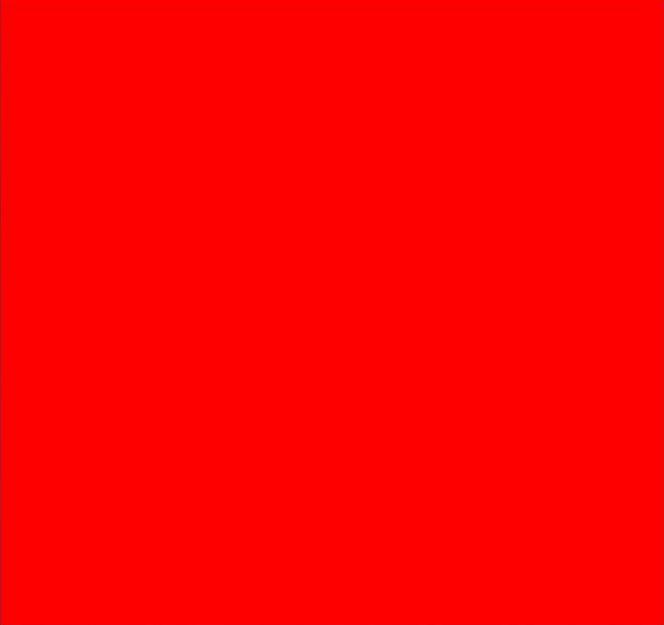
(XX) XXXXX-XXXX

Endereço:

Rua, número e bairro

Figura 15: Tela de cadastro dos hemocentros.

Fonte: Elaborado pelos autores (2025).



VOLTAR

CADASTRO

Tipo de Usuário:

Doador

Doador: Voluntário que deseja doar o sangue
Colaborador: Funcionário de um hemocentro

Nome:

Afonso Fonseca

CPF:

32189043283

Sexo

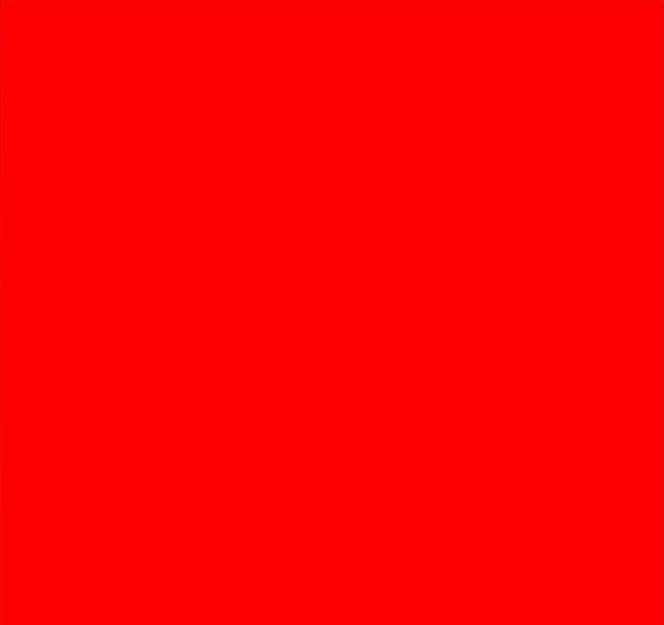
Mulher

Email:

Afonseca@gmail.com

Figura 16: Tela de cadastro dos doadores.

Fonte: Elaborado pelos autores (2025).



VOLTAR

CADASTRO

Tipo de Usuário:

Colaborador

Doador: Voluntário que deseja doar o sangue
Colaborador: Funcionário de um hemocentro

Nome:

Afonso Fonseca

CPF:

83929049382

CNPJ:

83929049382940

Sexo

Mulher

Email:

Figura 17: Tela de cadastro dos colaboradores dos hemocentros.

Fonte: Elaborado pelos autores (2025).

Campanhas Cadastradas

Sua doação vale mais que mil palavras. Doe sangue, doe vida!

Doação Urgente - Tipo O Negativo ★

Descrição: URGENTE: Precisamos de doadores tipo O- para atender demanda hospitalar crítica. O tipo O- é doador universal e salva vidas em emergências.

Período: 19/10/2025 até 25/01/2026

Tipo Sanguíneo: O-

Objetivo: Suprir a demanda urgente de sangue O- para cirurgias de emergência e pacientes politraumatizados.

Meta: 50L | **Atual:** 32L (64%)

Status: ✓ Ativa

Hemocentro: Hemocentro Unicamp - Campinas/SP

Junho Vermelho 2025 ★

Descrição: Campanha nacional de conscientização sobre a importância da doação de sangue. Durante todo o mês de junho, realizamos ações especiais e buscamos novos doadores para reforçar nossos estoques.

Período: 01/06/2025 até 30/06/2025

[Editar Campanha](#)

Figura 18: Tela de campanhas cadastradas (visão do colaborador).

Fonte: Elaborado pelos autores (2025).