

Doar Vida: Um aplicativo *mobile* para facilitar a doação de sangue.

Matheus M. da Costa Silva, Fernando Sambinelli

Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Campus Hortolândia (IFSP) Avenida Thereza Ana Cecon Breda, s/n, Vila São Pedro - Hortolândia – SP – Brasil

matheusm537@gmail.com, sambinelli@ifsp.edu.br

Abstract. *The number of blood donors in Brazil has not yet reached the levels expected by the Ministry of Health, and incentives for blood donation need to be carried out continually to maintain active donors. This work demonstrates the steps and the results of developing a smartphone application with the aim of facilitating Brazilians' access to information about blood collection centers. The main programming languages and technologies used in the development of the application were JavaScript, with the React Native framework, Node.js, and PostgreSQL. The developed application offers users the ability to set reminders for donations, locate blood collection centers and track their donation history.*

Resumo. *A quantidade de doadores de sangue no Brasil ainda não atingiu os níveis esperados pelo Ministério da Saúde, e incentivos à doação de sangue precisam ser realizados constantemente para manter os doadores ativos. Este trabalho demonstra as etapas e os resultados do desenvolvimento de uma aplicação para smartphones, com o objetivo de facilitar o acesso dos brasileiros às informações de postos de coleta de sangue. As principais linguagens de programação e tecnologias utilizadas no desenvolvimento do aplicativo foram JavaScript, com o framework React Native, Node.js e PostgreSQL. O aplicativo desenvolvido oferece aos usuários a possibilidade de registrar lembretes para as doações, localizar postos de coleta de sangue e acompanhar o histórico de doações realizadas.*

1. Introdução e Motivação

A quantidade de doadores de sangue no Brasil se aproxima de 1,6% da população, e muitos postos de coleta estão com queda no número de doações de sangue, conforme apurado pelo Governo do Brasil, no ano de 2020 (Governo do Brasil, 2021).

O sangue doado é muito importante em tratamento, e a falta dele pode levar a complicações. Entre as pessoas que precisam do sangue doado, estão vítimas de acidentes, pessoas em tratamento de câncer, pessoas com anemias crônicas, além das que passaram por cirurgias e por complicações decorrentes da dengue e da febre amarela (Governo do Brasil, 2021).

Segundo Rodrigues (2020), a Agência Brasil do Ministério da Saúde estimou que, devido à pandemia do vírus Covid-19, as doações de sangue tiveram quedas de 15% a 20% em comparação ao ano de 2019. Ademais, não houve casos de desabastecimentos registrados ao longo de 2020, mas, em algumas situações, o Ministério da Saúde precisou acionar o plano nacional de contingência e realizar a transferência de bolsas de sangue.

Vinculada ao Hospital das Clínicas da Faculdade de Medicina da Universidade de São Paulo (USP) e responsável por abastecer mais de 100 instituições de saúde da rede pública paulista, a Fundação Pró-Sangue coletou, em 2020, 108.707 bolsas de sangue (Rodrigues; Alex, 2020). A Tabela 1, apresenta os resultados da coleta de bolsas de sangue que demonstram a diminuição na quantidade de doações de sangue realizadas no estado de São Paulo. O mesmo cenário se repete para outras regiões do país.

Conforme divulgado pela Secretaria de Saúde do estado de São Paulo (Cultura, 2023), o número de doações no estado apresentou queda em 2020, registrando 771.819 procedimentos de coleta, o que representa uma redução de 7,9% em comparação com o ano de 2019, que contou com 711.549 procedimentos. Em 2021, houve um aumento, chegando a 718.253, representando menos de 1,0% em relação ao ano anterior. Já em 2022, o total foi de 703.430, refletindo uma queda de 3,1% em comparação com 2021.

Com base nessas informações, percebemos a possibilidade de contribuir para o aumento do número de doações com tecnologia. Durante a elaboração da solução tecnológica, foi necessário avaliar a melhor plataforma para o desenvolvimento da aplicação. Para isso, foram levantadas as ferramentas que os brasileiros mais utilizam para acessar a internet.

Conforme divulgado pelo Governo do Brasil, também em 2021, os resultados da pesquisa realizada pelo Instituto Brasileiro de Geografia e Estatística (IBGE), no ano de 2019, mostram que 82,7% dos domicílios nacionais possuem acesso à internet e que o celular continua sendo a principal ferramenta de acesso a ela; 99,5% dos domicílios com acesso à internet tinham celular; 45,1%, computador; 31,7%, televisão; e 12,0%, tablet.

A partir desses dados, surge o aplicativo Doar Vida, que foi desenvolvido como uma solução para facilitar o acesso a postos de coleta doação de sangue e fidelizar os doadores, sendo uma ponte de comunicação entre os postos de coleta, seu banco de usuário cadastrados e os usuários do aplicativo Doar Vida. Com foco em atingir o maior número de pessoas, o desenvolvimento da aplicação é para smartphones.

Tabela 1. Resultado por ano da coleta de bolsas de sangue da Fundação Pró-Sangue (Rodrigues, Alex 2020).

Ano	Coleta de bolsas de sangue (Fundação Pró-Sangue)
2015	131.068
2016	124.063
2017	123.851
2018	118.997
2019	114.050
2020	108.707

2. Trabalhos correlatos

Nesta seção, serão apresentados os aplicativos mais populares nas lojas *Play Store* e *App Store*, que possuem semelhanças com o aplicativo desenvolvido.

2.1 COLSAN - Doe Sangue, Doe Vidas

Desenvolvido pela *Conte Smart Technology*, o aplicativo Doe sangue, Doe Vidas oferece ao usuário a utilização da geolocalização para encontrar os postos de coleta mais próximos, o agendamento pelo aplicativo e a carteirinha digital de doação (Vidas, 2020).

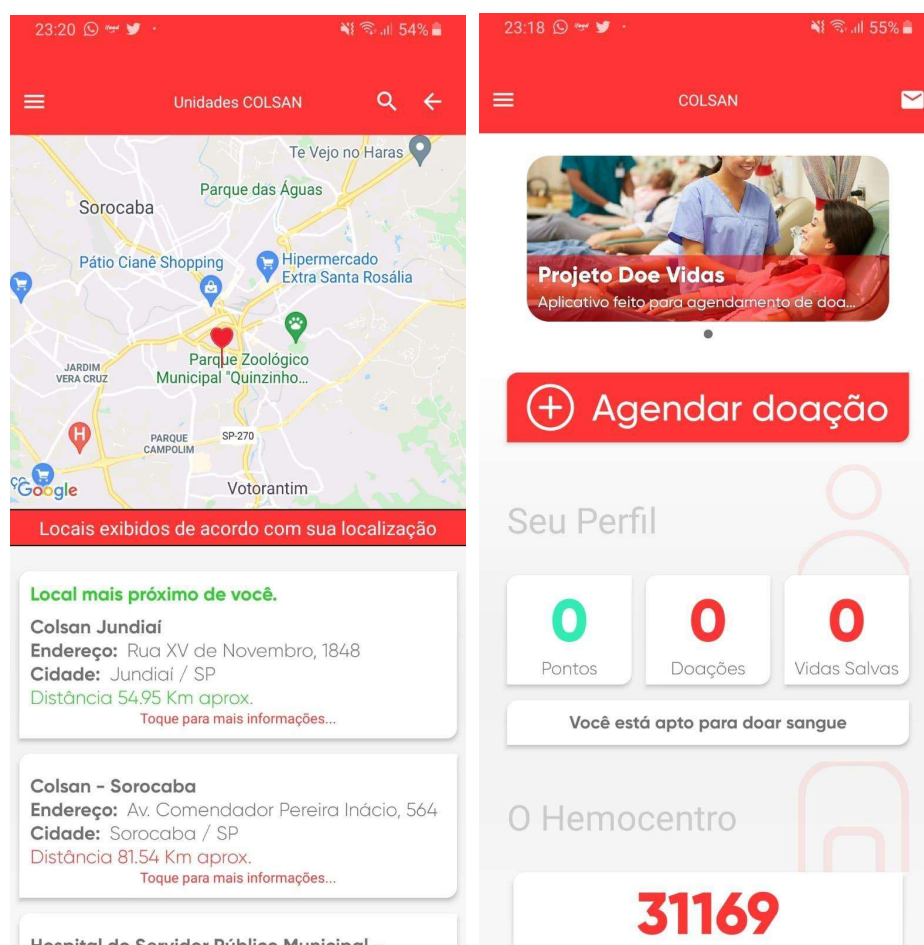


Figura 1. Interfaces do aplicativo Doe Sangue, Doe Vidas, a esquerda a tela de localização de pontos de doação e a direita a tela principal com informações sobre doações realizadas e botões de funcionalidade (2022).

2.2 Hemovida

Disponibilizado pelo Governo do Brasil, o aplicativo Hemovida possibilita que o usuário visualize em lista os postos de coleta próximos, utilizando a geolocalização (Hemovida, 2020).

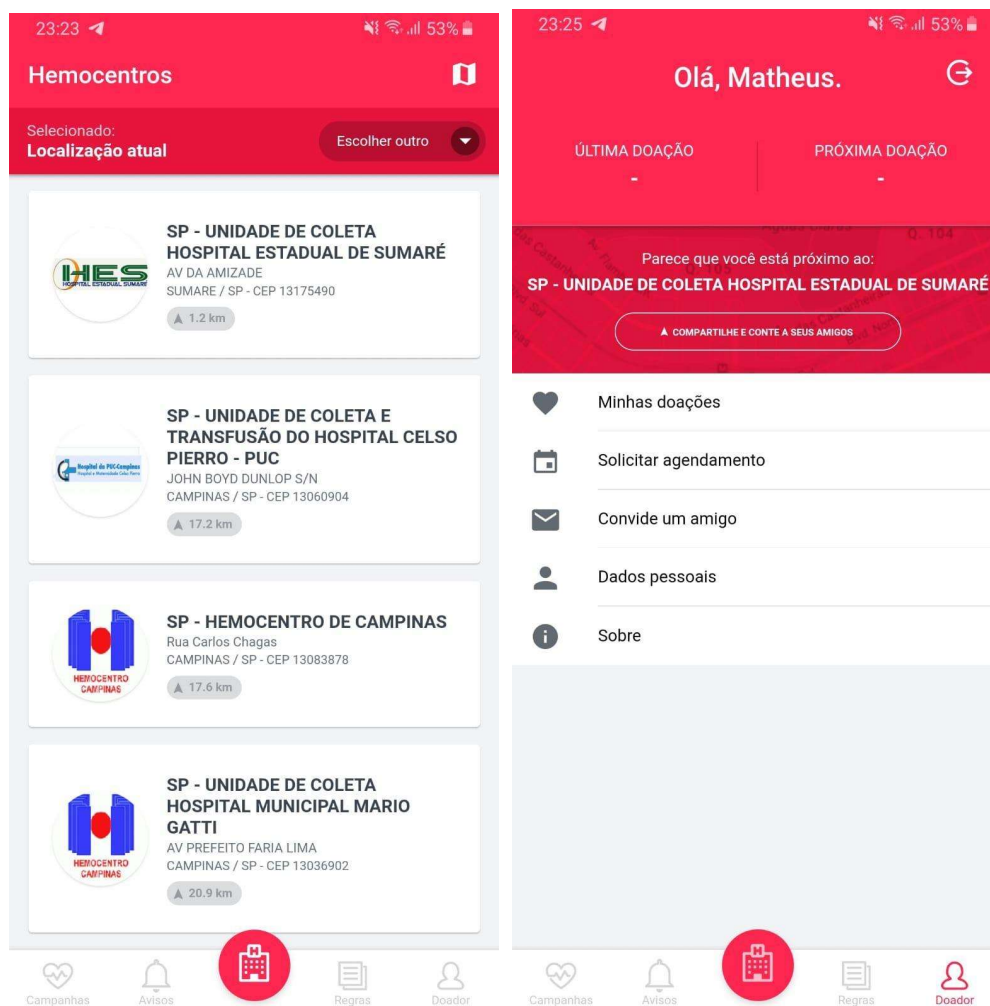


Figura 2. Interface do aplicativo Hemovida, a esquerda a tela de localização de pontos de doação e a direita a tela principal com botões de funcionalidade (2022).

2.3 Partiu Doar Sangue

Desenvolvido pela PDS Tecnologia, o aplicativo Partiu Doar Sangue possibilita que o usuário cadastre pedidos de doação, que são divulgados para outros usuários do aplicativo, e utilize a localização para encontrar áreas para doação de sangue (Partiu Doar Sangue, 2018).

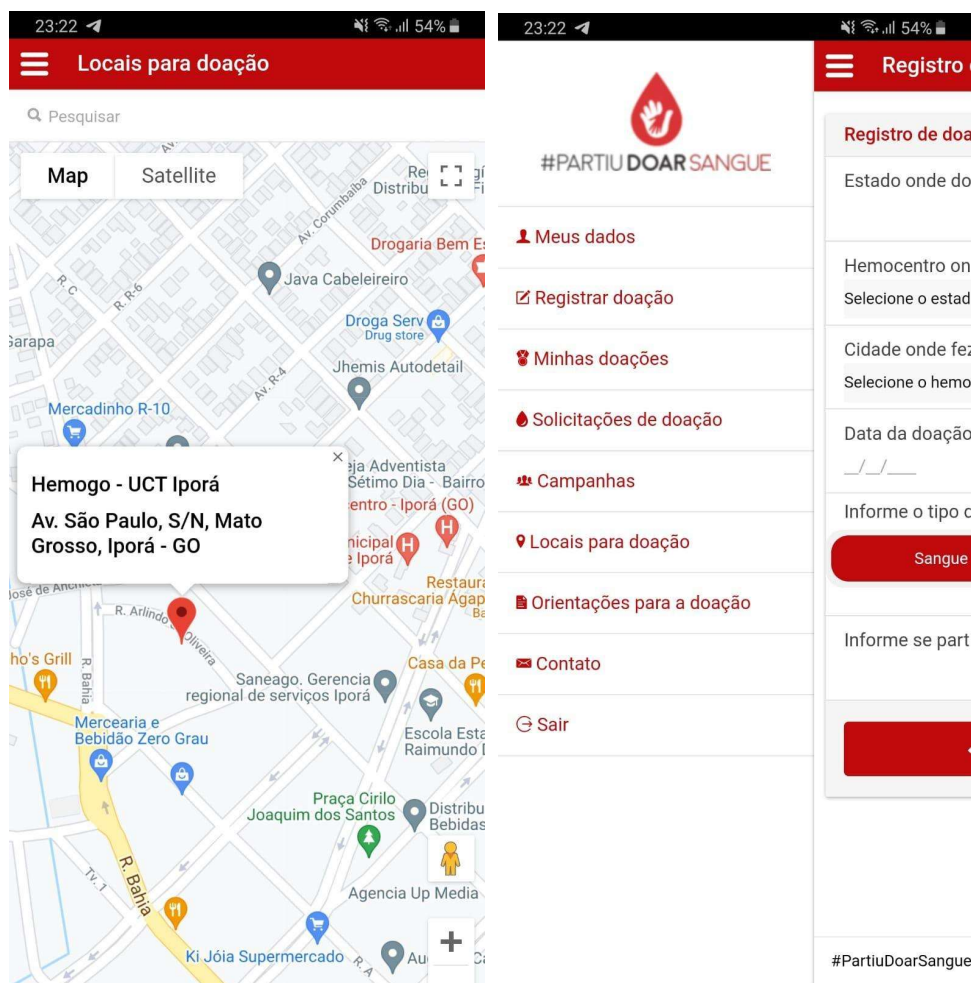


Figura 3. Interface do aplicativo Partiu Doar Sangue, a esquerda a tela de localização de pontos de doação e a direita a tela principal com botões de funcionalidade (2022).

2.4 Comparação entre os aplicativos similares e o Doar Vida

A Tabela 2 exibe uma comparação das funcionalidades entre o aplicativo Doar Vida e seus correlatos. A distinção do aplicativo Doar Vida reside na capacidade de permitir que os usuários registrem doações realizadas em qualquer posto de coleta, não se restringindo ao cadastro exclusivo de doações nos postos listados na aplicação. Em contraste, os aplicativos que carecem de integração com bancos de sangue não oferecem a facilidade de agendamento totalmente online pelo aplicativo.

Tabela 2. Comparação entre as funcionalidades do aplicativo Doar Vida e aplicativos similares (Autoria própria, 2022).

Funcionalidades	Registrar doações	Agendamento pelo App	Informações sobre o posto de coleta	Base de dados nacional	Usa localização para encontrar postos	Aplicativo para <i>IOS</i>	Aplicativo para <i>Android</i>
Doar Vida	✓	✗	✓	✓	✓	✓	✓
Partiu doar Sangue	✓	✗	✗	✓	✓	✗	✓
Hemovida	✗	✗	✓	✗	✓	✓	✓
Doe Sangue, Doe vidas	✗	✓	✓	✗	✓	✓	✓

3. Referencial Teórico

Nesta seção, apresentar-se-ão os referenciais teóricos dos principais conceitos utilizados no desenvolvimento da aplicação.

3.1 Modelo de desenvolvimento incremental

De acordo com Pressman (2011, p. 61), o modelo incremental é a junção de processos lineares e paralelos, e cada sequência linear gerará incrementos. A Figura 4, abaixo, apresenta os processos para entrega de incrementos.

O modelo de desenvolvimento incremental é descrito como um processo que "intercala as atividades de especificação, desenvolvimento e validação. O desenvolvimento como uma série de versões (incrementos), com cada uma delas acrescentado funcionalidade à versão anterior" (Sommerville, 2011, p. 21).

Segundo Pressman (2011, p. 62), quando se utiliza um modelo incremental, frequentemente, o primeiro incremento é um produto essencial. Os requisitos básicos são atendidos, porém, outros recursos da aplicação não são entregues. Esse produto essencial é utilizado pelo cliente ou passa por uma avaliação detalhada.

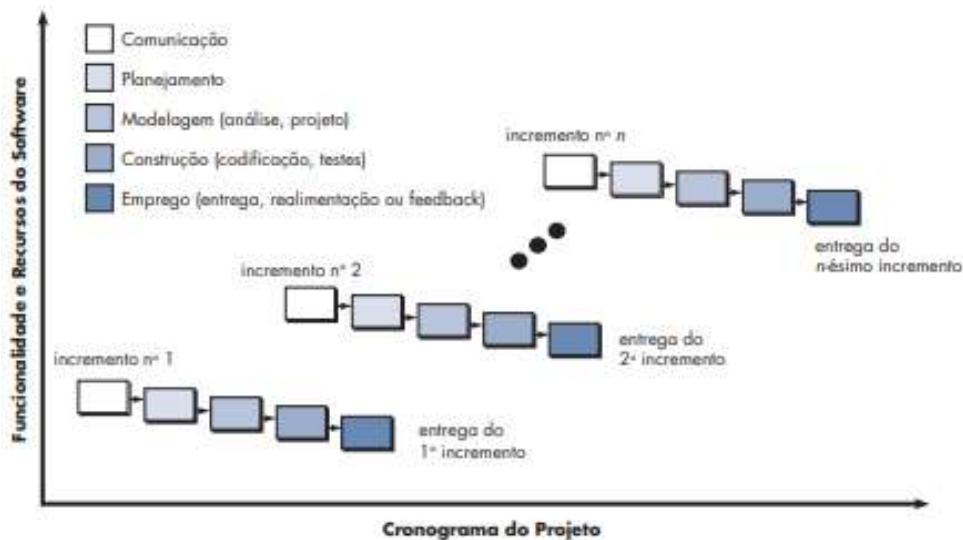


Figura 4. Diagrama do modelo de desenvolvimento incremental (Pressman, 2011, p. 61).

3.2 Requisitos.

Conforme Sommerville (2011, p. 58), os requisitos de um sistema são as descrições dos serviços que ele deve prestar e as restrições à sua operação. Esses requisitos refletem as necessidades, dos clientes, de um sistema que atenda a um determinado propósito.

A respeito da definição de requisitos de usuário e requisitos de sistema, respectivamente, há as seguintes definições: requisitos de usuário são declarações, em uma linguagem natural, somada a diagramas, dos serviços que se espera que o sistema forneça para os usuários e das limitações sob as quais ele deve operar. Já os requisitos de sistema são descrições mais detalhadas das funções, dos serviços e das restrições operacionais do sistema de *software*.

3.3 Arquitetura

De acordo com Pressman (2011, p. 230), "arquitetura" pode ser definida como a estrutura ou estruturas do sistema, abrangendo os componentes de *software*, as propriedades externamente visíveis e a relação entre eles.

Logo, a arquitetura não é o *software*, mas é uma representação que possibilita analisar o projeto, os requisitos declarados, objetivando reduzir riscos que são associados à construção do *software*.

Um componente de *software* pode ser um módulo de programa ou uma classe orientada a objetos, a bancos de dados e a *middleware*, que possibilita a configuração de uma rede de clientes e servidores.

3.4 Prototipação

Após a elucidação dos requisitos, pode ser realizado um projeto rápido para testá-los, desenvolvendo, então, o *layout* da interface, os formatos de exibição na tela, etc. Conforme descreve Pressman (p. 63), esse projeto rápido leva à construção de um

protótipo, que será utilizado para avaliar os requisitos e fornecer retornos que servirão para aprimorar os requisitos.

Segundo Sommerville (2011, p. 30), um protótipo é uma versão inicial de um sistema de *software*, usado para demonstrar conceitos, experimentar opções de projeto e descobrir mais sobre o problema e suas possíveis soluções.

No processo de desenvolvimento de *software*, um protótipo ajuda na elicitação de requisitos, na validação de requisitos do sistema, no teste de soluções e na redução de custos no desenvolvimento de soluções incorretas.

4. Plataformas e ferramentas de desenvolvimento

Nesta seção, expor-se-ão as principais ferramentas e as plataformas a que o aplicativo Doar Vida foi implementado.

4.1 Plataforma

O aplicativo Doar Vida foi elaborado para os sistemas operacionais Android e iOS, empregando a biblioteca React Native para garantir compatibilidade do código com ambas as plataformas. Essa escolha estratégica se baseia no fato de que Android e iOS são os dois sistemas operacionais de dispositivos móveis mais amplamente utilizados, conforme indicado por dados do StatCounter em 2021. (StatCounter, 2021).

4.2 JavaScript

JavaScript é uma linguagem de programação de alto nível, que permite implementar itens complexos em páginas *web*, desenvolver páginas que realizam tarefas, além de apresentar informações estáticas. A linguagem é a terceira camada das tecnologias padrões da *web*, sendo as outras duas HTML e CSS (Mozilla, 2021).

4.3 React Native

React Native é uma biblioteca *JavaScript* para desenvolvimento de aplicações *mobile*, que renderiza os componentes para a interface de usuário nativa do dispositivo. Os componentes do *React Native* interagem com as *Application Programming Interfaces* (APIs) nativas. Dessa forma, é possível manter uma base de código para diferentes plataformas, como *Android* e *IOS*.

Os componentes do *React* envolvem o código nativo e interagem com as APIs nativas, o que permite que o processo de desenvolvimento das equipes seja mais rápido (Facebook, 2021).

4.4 API Rest

De acordo com o descrito pela Amazon (2023), uma *Application Programming Interface Representational State Transfer* (API REST) é uma API que implementa a arquitetura REST, que foi criada inicialmente como uma diretriz para gerenciar a comunicação em uma rede complexa, como a internet. Ela possibilita a comunicação confiável e de alta performance em escala. Alguns dos princípios dessa arquitetura são: interface uniforme; ausência de estado; sistema em camadas; capacidade de armazenamento e código sob demanda.

4.5 Node.js

É uma tecnologia para execução de código *Javascript* fora de navegadores. Sua execução é *single-thread*, pois apenas uma *thread* é responsável por executar o código *Javascript* da aplicação (Lenon, 2018).

Em um servidor *web*, utilizando linguagens tradicionais, para cada requisição recebida, é criada uma nova *thread* para tratá-la. A cada requisição, serão demandados recursos computacionais (memória RAM, por exemplo) para a criação dessa nova *thread*. Uma vez que esses recursos são limitados, as *threads* não serão criadas infinitamente e, quando esse limite for atingido, as novas requisições terão que esperar a liberação desses recursos alocados para serem tratadas (Lenon, 2018).

Apenas uma *thread* é executada no Node.js; ela é chamada de *Event Loop* e é tratada como um evento. O *Event Loop* fica em execução, aguardando eventos para tratar. Para cada requisição, um novo evento é criado (Lenon, 2018). O Node.js consegue realizar as entradas e saídas não-bloqueantes, pois as operações são assíncronas e não bloqueiam a *thread*. A figura 5, apresenta a comparação de um servidor *web* tradicional e um Node.js. Como um ambiente de execução JavaScript assíncrono orientado a eventos, o Node.js é projetado para desenvolvimento de aplicações escaláveis de rede (Foundation, 2021).

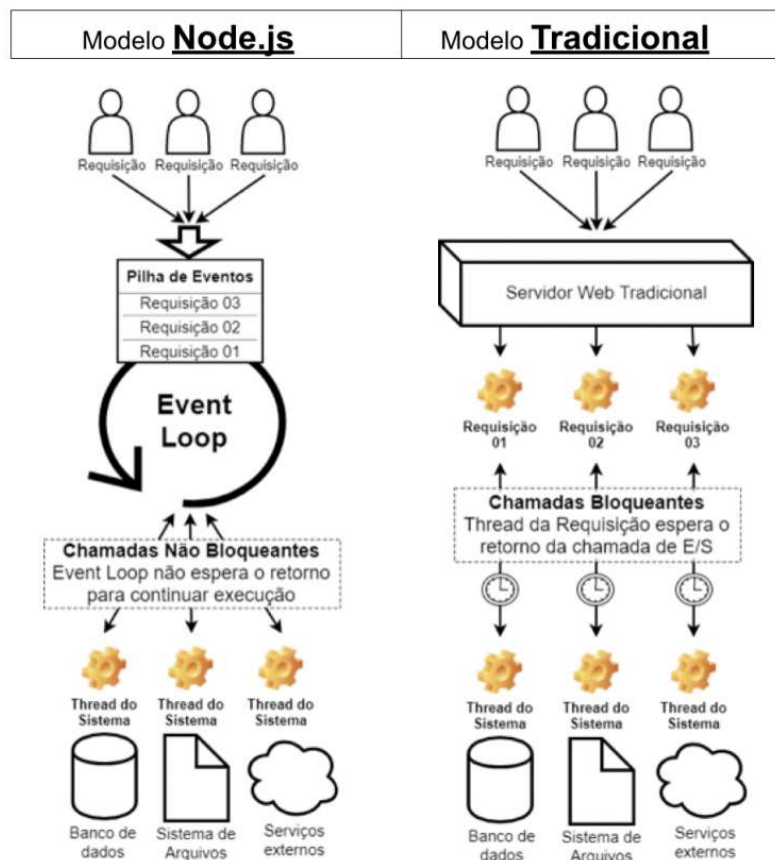


Figura 5. Comparação do do funcionamento de um servidor Node.js e servidor web tradicional (Lenon, 2018).

4.6 Visual Studio Code

Visual Studio Code é um editor de código desenvolvido pela *Microsoft* (2021), que possui funcionalidades simples, como editar código, dar suporte a diversas linguagens de programação, ser um terminal de comandos integrado e ter controle de versão. Porém, pode ser muito customizável com a utilização de extensões.

4.7 Expo

Expo é um *framework* e uma plataforma para aplicações *React Native*. Age em conjunto a ferramentas e serviços que ajudam a desenvolver, construir, implantar e iterar rapidamente em aplicativos *IOS*, *Android* e *Web*, utilizando a mesma base de código. (Expo, 2021).

5. Metodologia

O projeto se iniciou com a análise de requisitos, baseando-se nos aplicativos apresentados na seção de trabalho correlatos e na pesquisa sobre o processo de doação de sangue em postos de coleta, para definir os critérios de segurança, a divulgação de informações de campanhas e os dados sensíveis que são necessários.

A partir dos resultados, são elaborados os protótipos de interfaces do aplicativo, a implementação da aplicação e os testes manuais das interfaces que serão utilizadas por usuários.

No decorrer do desenvolvimento, a abordagem de separar as histórias de usuários, que serviram como base para a criação dos incrementos no sistema. Iniciando na implementação da API e banco de dados, estabelecendo a estrutura e lógica necessárias para suportar as funcionalidades do aplicativo. Em seguida, o desenvolvimento do aplicativo *Doar Vida*. Durante o processo, foram realizados testes para cada funcionalidade, identificando e corrigindo quaisquer problemas que surgiram. Mantendo uma abordagem iterativa, atualizando os requisitos conforme necessário para refletir as mudanças e melhorias implementadas.

6. Desenvolvimento

Nesta seção, serão revelados os resultados do desenvolvimento do aplicativo *Doar Vida*.

6.1 Elucidação de requisitos

A elucidação de requisitos de usuário que tem o perfil de cadastro como "doador" se deu por meio da análise de aplicativos citados nos trabalhos correlatos e de *feedbacks* nas lojas de aplicativos *Play Store*. Com isso, foram criados os seguintes casos de uso (Figura 6):

1. Manter doações: o usuário registra as doações realizadas para que seja exibida a regularidade das doações em forma de Gamificação;
2. Localizar postos de coleta: o usuário acessa um mapa que destaca os postos de coleta próximos de sua localização;
3. Manter conta do usuário: o usuário edita as informações da sua conta, como nome, *e-mail*, senha, tipo sanguíneo;

4. Manter agendamentos: o usuário consegue cadastrar os agendamentos de doações futuras para que seja lembrado quando estiver próximo da data de doação;
5. Enviar *feedback*: o usuário consegue enviar informações para o desenvolvedor;
6. Visualizar campanhas: o usuário consegue visualizar uma lista de campanhas de doação de sangue com informações detalhadas, como forma de agendamento, período de duração, local e horário de funcionamento.

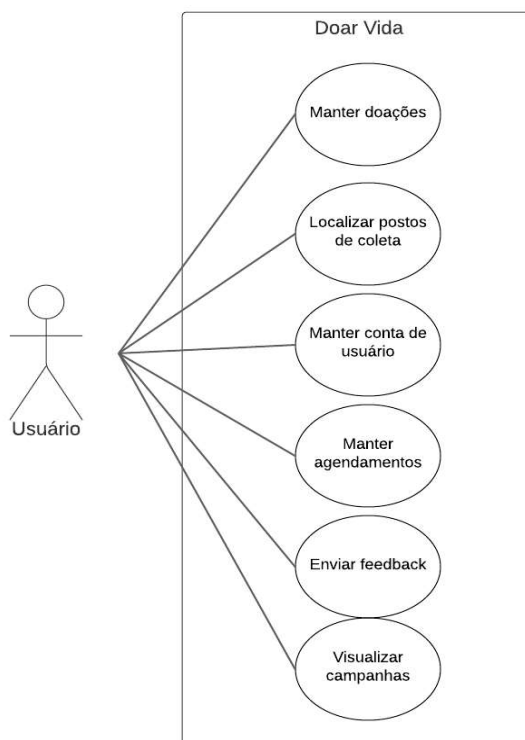


Figura 6. Diagrama de casos de uso do aplicativo Doar Vida (Autoria própria, 2022).

6.2 Histórias de usuário

Conforme descrito por Rehkopf (2023), uma história de usuário é a menor unidade de trabalho em uma estrutura ágil; é um objetivo final na perspectiva do usuário do *software*. Uma história de usuário é, assim, uma explicação informal do que o usuário deseja e tem o objetivo de expor o valor que será entregue ao cliente. A Figura 7, abaixo, demonstra as histórias de usuário (apêndice A): pesquisar postos de coleta de sangue, realizar *login* e seus critérios de aceitação.

Pesquisar postos de coleta de sangue

Como um doador eu desejo localizar postos de coleta de sangue próximos à minha localização para visualizar as informações do local.

Critérios de aceite:

- Deverá exibir em destaque no mapa os postos de coleta próximos da localização atual.
- Ao tocar em algum ponto no mapa deverá abrir uma tela com detalhes de suas informações.

Realizar Login

Como um doador eu desejo realizar login no aplicativo Doar Vida para visualizar minhas doações

Critérios de aceite:

- Deverá validar se o usuário está cadastrado no banco de dados com e-mail e senha fornecidos.
- Validar se o campo de e-mail está recebendo um e-mail.
- O campo senha deverá possuir uma máscara para ocultar os valores digitados.
- Ao pressionar o botão entrar deve realizar login no aplicativo caso os valores de e-mail e senha sejam válidos e redirecionar para a tela inicial do aplicativo.
- Ao pressionar o botão “Criar conta” deverá redirecionar para a tela de cadastro.

Figura 7. Histórias de usuários de pesquisar postos de coleta e realizar login (Autoria própria, 2023).

6.3 Prototipação

Foram construídos protótipos das histórias de usuário do aplicativo por meio de *wireframes*, com base nos requisitos elucidados. Os protótipos da Figura 8, a seguir, implementa a história de usuário “realizar login”; já o protótipo da Figura 9, abaixo, implementa a história de usuário “pesquisar postos de coleta de sangue” ambos refletem as histórias de usuário descritas acima. Na interface de busca de postos de coleta e a página de detalhes de um posto de coleta foi escolhida a navegação por um mapa, para que o usuário possa identificar os locais que estão mais próximos de sua localização atual.



Figura 8. Wireframe da história de usuário realizar login (Autoria própria, 2022).

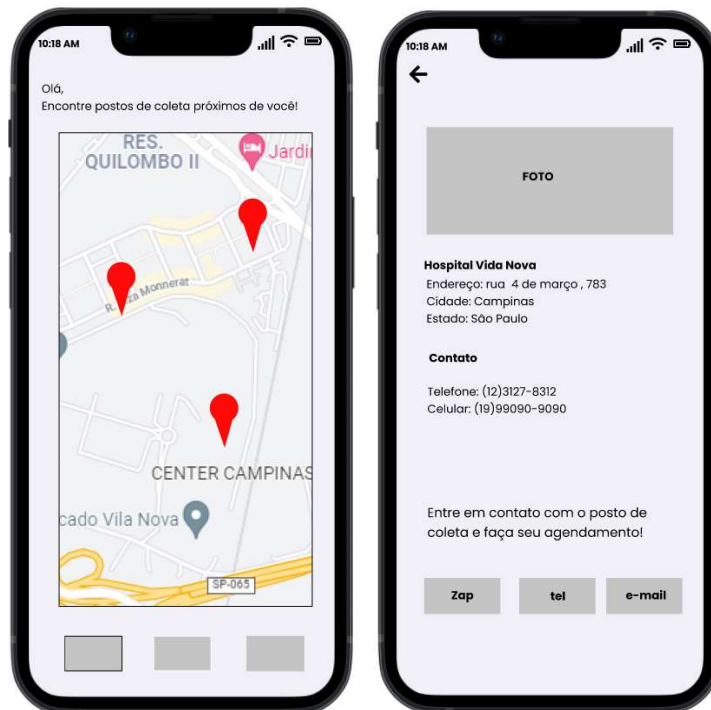


Figura 9. Wireframes da história: pesquisar postos de coleta de sangue (idem, 2022).

6.4 Modelagem do Banco de Dados

Consoante Pressman (2011, p. 163), se houver a necessidade de utilizar um banco de dados, poderá ser criado um modelo de dados como parte da modelagem de requisitos. O diagrama Entidade-Relacionamento (*entity-relationship diagram*) trata das questões e representa todos os objetos de dados introduzidos, armazenados, transformados e produzidos em uma aplicação.

Na aplicação Doar Vida, foram definidas quatro entidades: usuário, doação, postos de doação e agendamento. Um usuário pode registrar no sistema uma doação ou um agendamento; o agendamento pode estar vinculado a um posto de doação; e uma doação pode estar vinculada a um posto de doação. Existe o cenário em que o posto de doação não está registrado no sistema e, por isso, não há vínculo de doação ou agendamento com um posto de doação. A Figura 10, a seguir, ilustra a modelagem realizada para o banco de dados da aplicação.

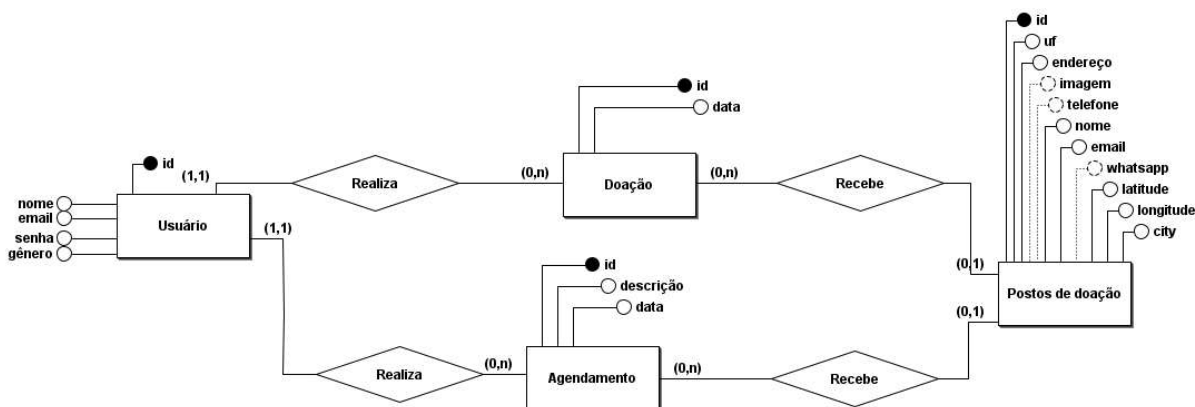


Figura 10. Diagrama Entidade-Relacionamento do projeto (Autoria própria, 2023).

6.5 Definição de modelo de arquitetura

Na etapa de desenvolvimento do *software*, foi desenhada a arquitetura da solução técnica, como mostrado na Figura 11. Após isso, foi iniciada a etapa de criação dos artefatos de código, seguindo a seguinte ordem de desenvolvimento: banco de dados, API e aplicação *mobile*.

Conforme a Figura 11, o servidor e o banco de dados são armazenados em um provedor em nuvem que, através do protocolo HTTP, responde às requisições dos clientes, as quais são instâncias da aplicação *mobile* Doar Vida em dispositivos móveis. Dessa forma, é definida a estrutura de uma API *Restful*, que atende a aplicações externas, as quais recebem respostas em formato JSON.

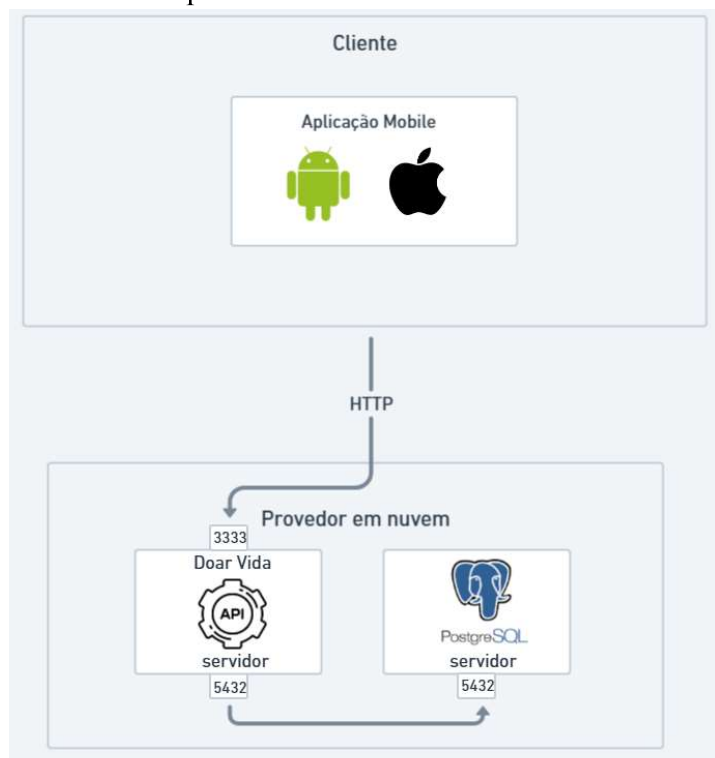


Figura 11. Diagrama da arquitetura do projeto (Autoria própria, 2022).

6.6 API Rest

Conforme apresentado na Figura 11, a comunicação feita entre o cliente (Aplicação *Mobile*) e o servidor (API) se dá por requisições no protocolo HTTP. Na tabela 3, são apresentadas as rotas implementadas na API em *Node.js* que são utilizadas para o aplicativo trocar informações com o servidor.

Tabela 3. Rotas da API Restful utilizada pelo aplicativo Doar Vida. (Autoria própria, 2022).

URI	Verbo HTTP	Descrição.
/points	POST	Criar ponto de coleta.
/points/{id}	GET	Listar um ponto de coleta.
/points?city&uf	GET	Listar todos os pontos de coleta por cidade.
/schedule	POST	Criar agendamento.
/schedule/{id}	GET	Listar todos os agendamentos de um usuário.
/schedules?user_id&date	GET	Listar os agendamentos futuros.
/donation	POST	Criar doação.
/donations/{id}	GET	Listar doações de um usuário.
/user/update	PATCH	Atualizar informações do usuário.
/user/update/password	PATCH	Atualizar senha do usuário.
/user	POST	Criar usuário.
/user/sign	POST	Realizar autenticação de um usuário.
/user/{id}	GET	Listar um usuário.

6.7 Funcionalidades da aplicação

Nesta seção, apresentaremos o resultado final das principais funcionalidades do aplicativo. Não foram implementados os casos de uso: visualizar campanhas e enviar *feedback* estão listados como trabalhos futuros. O sistema opera por meio de requisições à API, e as decisões lógicas, assim como a apresentação de informações, são conduzidas com base nos dados recebidos como retorno dessas requisições.

6.7.2 Página Principal

De acordo com a Figura 14, abaixo, na página principal, o usuário recebe um resumo das doações que já realizou nos últimos 12 meses e o total de todas as doações, a meta de doação dos últimos 12 meses é definida a partir do sexo biológico do usuário. De acordo com a Biblioteca Virtual em Saúde (2023) mulheres podem doar sangue a cada intervalo de 90 dias, podendo fazer até 3 doações por ano, homens podem fazer até 4 doações por ano, aguardando 60 dias de intervalo, o cálculo da quantidade de impacto do usuário é baseado na quantidade de doações registradas, uma doação pode impactar em média 4 pessoas. É possível acessar outras funcionalidades do aplicativo por essa tela, como registrar doação, criar lembretes e visualizar o histórico de doações nos botões da parte de cima da tela e procurar postos de coleta, retornar a página principal e acessar o perfil do usuário pelos botões na parte inferior da tela.

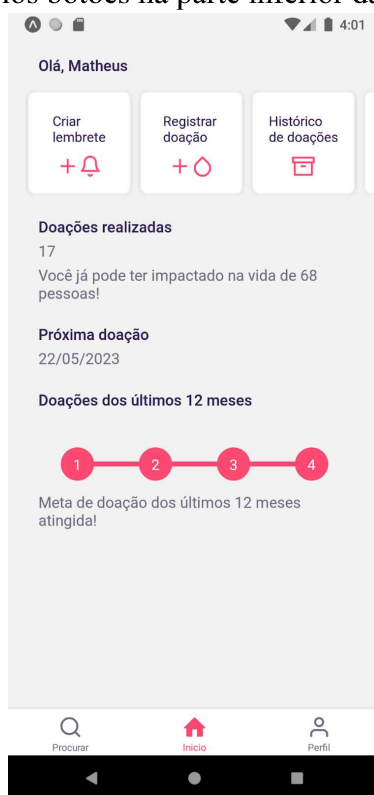


Figura 13. História de usuário: visualizar status do doador (Autoria própria, 2023).

6.7.3 Encontrar postos de coleta de sangue no mapa

Como pode ser visto na Figura 15, é possível localizar os postos de coleta de sangue próximos através da tela "procurar" representado pela lupa no canto inferior esquerdo. Ao tocar em algum posto de coleta, é apresentada a tela com detalhes, para que o usuário possa entrar em contato com o posto de coleta de sangue ou ter mais informações sobre a localização. É feita a utilização do sistema padrão de mapas do dispositivo do usuário, os postos de coleta são retornados por uma consulta na API e adicionados através da latitude e longitude no mapa. O trecho de código na Figura 16 realiza a inserção de cada posto de coleta dentro do componente de mapa através da latitude e longitude.

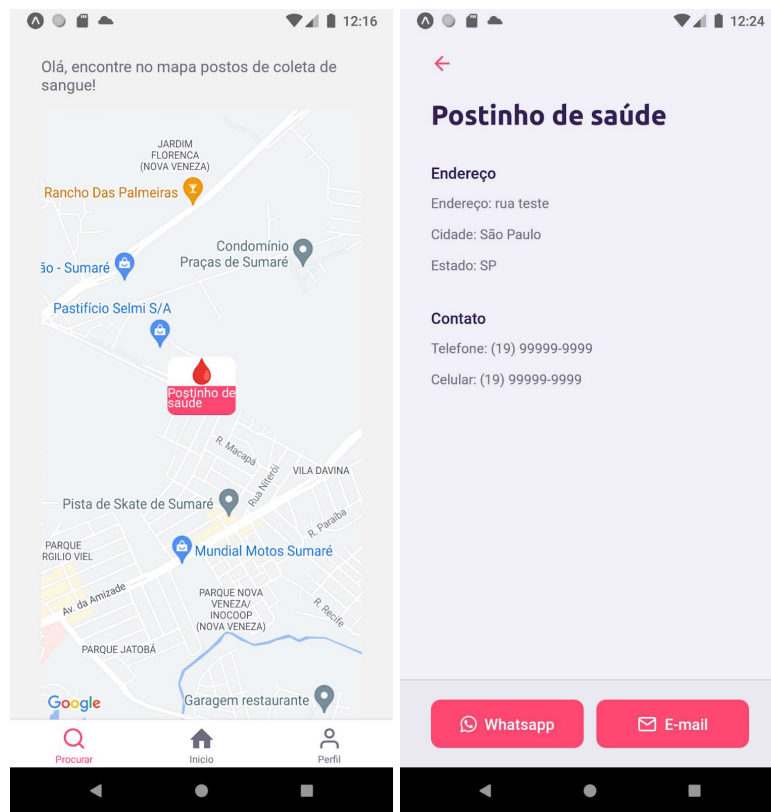


Figura 15. História de usuário: pesquisar postos de coleta de sangue (Autoria própria, 2023).

```

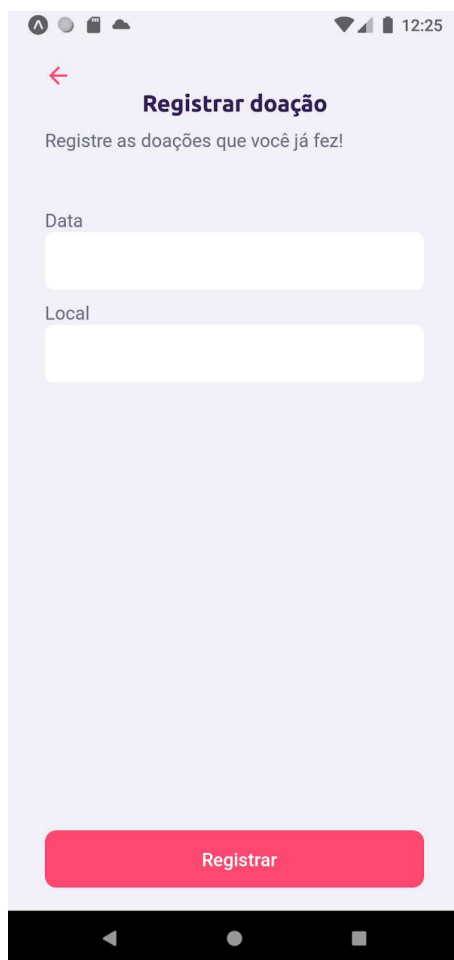
<MapView
  style={styles.map}
  loadingEnabled={initialPosition[0] === 0}
  initialRegion={{
    latitude: initialPosition[0],
    longitude: initialPosition[1],
    latitudeDelta: 0.014,
    longitudeDelta: 0.014,
  }}
>
  {points.map((point) => (
    <Marker
      key={String(point.id)}
      style={styles.mapMarker}
      onPress={() => handleNavigateToDetail(point)}
      coordinate={{
        latitude: point.latitude,
        longitude: point.longitude,
      }}
    >
      <View style={styles.mapMarkerContainer}>
        <Image
          style={styles.mapMarkerImage}
          source={require('../assets/gota-de-sangue.png')}
        />
        <Text style={styles.mapMarkerTitle}>{point.name}</Text>
      </View>
    </Marker>
  ))}
</MapView>

```

Figura 16. Trecho de código do aplicativo Doar Vida que realiza a inserção de uma lista de postos de coleta em um componente de mapa *React Native* (Autoria própria, 2023).

6.7.4 Registrar doação

Consoante a Figura 17, a seguir, o usuário pode registrar as doações realizadas acessando a funcionalidade "registrar doação", na tela principal do aplicativo, informando a data e o local em que a doação foi realizada.



A imagem mostra a interface de usuário para registrar uma doação. No topo, há um ícone de seta para voltar e o título "Registrar doação". Abaixo do título, há uma instrução: "Registre as doações que você já fez!". O formulário possui dois campos de entrada: "Data" e "Local". No rodapé, há um botão laranja com o texto "Registrar".

Figura 17. História de usuário: registrar doação (Autoria própria, 2023).

6.7.5 Visualizar histórico de doações

Como visto na Figura 18, o usuário pode consultar o histórico de doações realizadas, acessando a funcionalidade "histórico de doações", na tela principal.

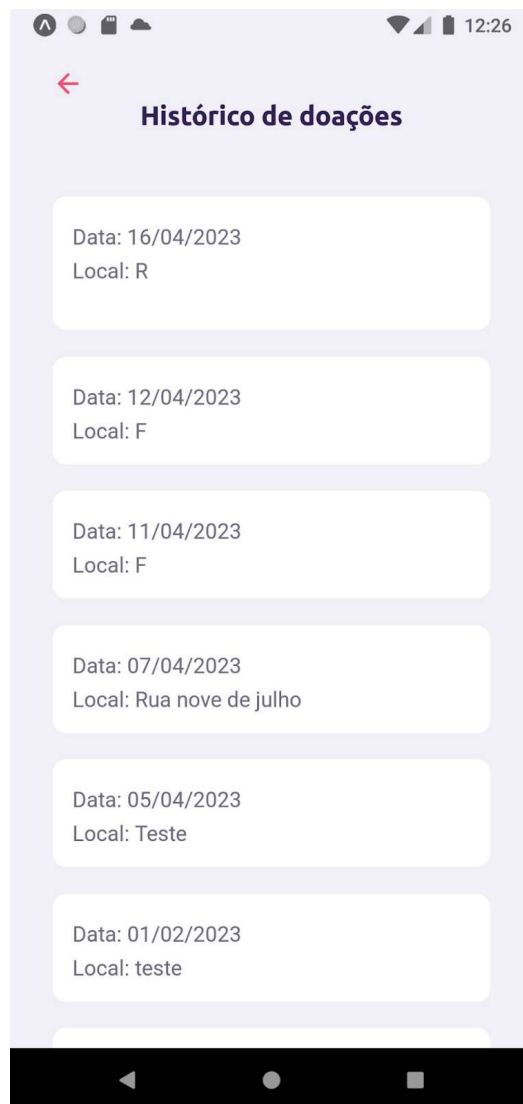


Figura 18. História de usuário: visualizar doações (Autoria própria, 2023).

6.6 Artefatos e código fonte

A fim de tornar acessível o conteúdo produzido, todos os artefatos de *softwares* desenvolvidos nesse trabalho, inclusive o código fonte, foram disponibilizados no serviço em nuvem *GitHub* na *url*: <https://github.com/Marques537>.

7. Conclusão

Com o objetivo de facilitar o acesso a postos de coleta de doação de sangue, engajar os doadores, e considerando os *smartphones* a plataforma de acesso mais utilizada pelos brasileiros, foi construído o aplicativo *Doar Vida*, para servir como um aplicativo para encontrar postos de coleta de doação de sangue e acompanhar o progresso das doações feitas pelo usuário. Com essa aplicação, busca-se colaborar com o crescimento do número de doações de sangue.

Assim, o aplicativo foi parcialmente desenvolvido, contendo as principais funcionalidades para atingir o objetivo da aplicação, como: registrar lembrete da doação, localizar postos de coleta próximos e acompanhar histórico de doações.

Como trabalhos futuros, podem ser criadas algumas funcionalidades que não foram implementadas no período de desenvolvimento, como: enviar *feedback* para que o usuário possa descrever sua experiência com o aplicativo; visualizar campanhas de doação de sangue divulgadas por postos de coleta; *login* com redes sociais; permitir editar todas as informações do cadastro do usuário; incrementar testes unitários para API e a aplicação *mobile*. Para incrementar a solução, são sugeridas a criação de uma aplicação exclusiva para os postos de coleta serem cadastrados, a disponibilização de opções de adicionar campanhas de doação de sangue, exibindo-as no aplicativo Doar Vida, e a publicação do aplicativo em lojas virtuais, como *AppStore* e *Google Play Store*.

Desse modo, com o desenvolvimento desse trabalho, foi possível aplicar diversos conhecimentos adquiridos em várias disciplinas ao decorrer do Curso Superior de Análise e Desenvolvimento de Sistemas. Dentre elas, estão: Arquitetura de Software; Engenharia de Software e Banco de Dados I e II; Linguagem de Programação I e II; Desenvolvimento Web. Ademais, foram obtidos diversos conhecimentos relacionados a construções de APIs REST e à programação em multiplataforma para dispositivos móveis, com a utilização da biblioteca *React Native*.

8. Referências bibliográficas

- Amazon Web Services, Inc (2023). O que é API RESTful? <https://aws.amazon.com/pt/what-is/restful-api/> [Online Acessado em 04 de Junho de 2023].
- Biblioteca Virtual em Saúde (2023). Conheça o passo a passo da doação de sangue. <https://bvsmms.saude.gov.br/conheca-o-passo-a-passo-da-doacao-de-sangue>. [Online; Acessado em 05 de Dezembro de 2023].
- Cultura (2023). Governo de SP se preocupa com queda nas doações de sangue em 2023. https://cultura.uol.com.br/noticias/59458_governo-de-sp-se-preocupa-com-queda-nas-doacoes-de-sangue-em-2023.html [Online Acessado em 12 de Novembro de 2023].
- Expo (2021). Introduction to Expo. <https://docs.expo.dev/index.html/>. [Online Acessado em 28 de Julho de 2021]
- Facebook (2021). React Native. <https://reactnative.dev/> [Online Acessado em 28 de Julho de 2021].
- Governo do Brasil (2021a). Doação de sangue é necessária para abastecer estoques em todo País. Governo Do Brasil. <https://www.gov.br/pt-br/noticias/assistencia-social/2020/08/doacao-de-sangue-e-necessaria-para-abastecer-estoques-em-todo-pais> [Online; Acessado em 21 de Março de 2021].
- Governo do Brasil (2021b). Pesquisa mostra que 82,7% dos domicílios brasileiros têm acesso à internet. <https://www.gov.br/mcom/pt-br/noticias/2021/abril/pesquisa-mostra-que-82-7-dos-do>

- micilios-brasileiros-tem-acesso-a-internet/. [Online; Acessado em 20 de Julho de 2021].
- Hemovida(2020). Hemovida. <https://mobilems.saude.gov.br/hemovida/>. [Online Acessado em 10 de Julho de 2021].
- Lenon (2018). Node.js. <https://www.opus-software.com.br/node-js/>. [Online Acessado em 15 de Julho de 2021].
- Microsoft Corporation (2021a). Visual Studio Code. <https://code.visualstudio.com/> [Online; Acessado em 20 de Junho 2021].
- Microsoft Corporation (2021b). Download Windows 10. <https://www.microsoft.com/en-gb/software-download/windows10>. [Online; Acessado em 20 de Junho de 2021].
- Mozilla (2021). O que é JavaScript. https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/What_is_JavaScript [Online Acessado em 28 de Julho de 2021].
- Openjsf Foundation (2021). NodeJs. <https://nodejs.org/en/>. [Online; Acessado em 25 de Junho de 2021].
- Partiu Doar Sangue (2018). Partiu Doar Sangue. <https://partiudoarsangue.com.br/#/>. [Online, acessado em 10 de Julho de 2021]
- Pressman, R.(2011) Engenharia de software: Modelos de Processo. 7. ed. Porto Alegre: AMGH Editora Ltda.
- Rehkopf, Max.,(2023). Histórias de usuários com exemplos e um template. <https://www.atlassian.com/br/agile/project-management/user-stories>. [Online; Acessado em 12 de Agosto de 2023].
- Rodrigues, A.(2020). Queda na doação de sangue devido à pandemia preocupa hemocentros. <https://agenciabrasil.ebc.com.br/saude/noticia/2021-01/queda-na-doacao-de-sangue-devido-pandeia-preocupa-hemocentros/>. [Online; Acessado em 20 de Julho de 2021].
- Sommerville, I. (2011). Engenharia de Software. Pearson, nona edição.
- StatCounter (2021). Mobile Operating System Market Share Worldwide. <https://gs.statcounter.com/os-market-share/mobile/worldwide> [Online; Acessado em 25 de Junho de 2021].
- Vidas, D.(2020). Doe Vidas. <https://www.doevidas.com.br/>. [Online; Acessado em 10 de Julho de 2021].

APÊNDICE A - Histórias de usuário.

1. Pesquisar postos de coleta de sangue

1. Como um doador, eu desejo localizar postos de coleta de sangue próximos à minha localização para visualizar as informações do local.
2. Critérios de aceite:
 - Deverá exibir, em destaque no mapa, os postos de coleta próximos da localização atual.
 - Ao tocar em algum ponto no mapa, deverá abrir uma tela com detalhes de suas informações.

2. Realizar *Login*

1. Como um doador, eu desejo realizar *login* no aplicativo Doar Vida para visualizar minhas doações.
2. Critérios de aceite:
 - Deverá validar se o usuário está cadastrado no banco de dados com *e-mail* e senha fornecidos.
 - Validar se o campo de *e-mail* está recebendo um *e-mail*.
 - O campo "senha" deverá possuir uma máscara para ocultar os valores digitados.
 - Ao pressionar o botão "entrar", deve-se realizar *login* no aplicativo caso os valores de *e-mail* e senha sejam válidos e redirecionar para a tela inicial do aplicativo.
 - Ao pressionar o botão "criar conta", deverá redirecionar para a tela de cadastro.

3. Criar conta

1. Como um doador, eu desejo criar uma conta para acessar o aplicativo Doar Vida.
2. Critérios de aceite:
 - Deve haver campos para o usuário preencher nome, *e-mail*, senha e confirmar.
 - Validar se o campo de *e-mail* está recebendo um *e-mail*.
 - Validar se os campos senha e confirmar senha estão preenchidos com os mesmos valores.
 - O campo senha e confirmar devem possuir uma máscara para ocultar os valores digitados.
 - Ao pressionar o botão "cadastrar", o formulário deve ser submetido e redirecionado para a tela de *login*.

4. Visualizar status do doador

1. Como um doador, eu desejo visualizar um resumo a respeito do status das minhas doações, desejo visualizar a data da próxima doação, total de doações e quantas doações fiz nos últimos doze meses.
2. Critérios de aceite:
 - Deve haver a data da próxima doação do usuário; caso não possua

- agendamento, deve informar que não possui um agendamento.
- Deve mostrar o total de doações registradas no aplicativo Doar Vida.
- Deve mostrar quantas doações foram feitas nos últimos doze meses.
- Deve ser a tela principal do aplicativo.
- Deve apresentar o menu para navegar para as outras tela do aplicativo.

5.Registrar lembrete

1. Como um doador, eu desejo cadastrar um lembrete para que eu possa saber quando faria outra doação de sangue.
2. Critérios de aceite:
 - Deve haver campos para registrar data e horário, descrição e local de doação.
 - O campo de descrição é opcional; os campos de data, de horário e de local de doação são obrigatórios.
 - Após o registro ser concluído, deverá exibir uma mensagem de sucesso.

6.Registrar doação

1. Como um doador, eu desejo cadastrar as doações de sangue que já realizei.
2. Critérios de aceite:
 - Deve haver campos para registrar a data e o local de doação.
 - Após o registro ser concluído, deverá exibir uma mensagem de sucesso.

7.Visualizar doações

1. Como um doador, eu desejo visualizar o histórico das doações de sangue que já realizei.
2. Critérios de aceite:
 - Deverá exibir para o usuário todas as doações registradas no aplicativo.

8.Ver perfil

1. Como um doador de sangue, eu desejo visualizar as informações no meu perfil para ter a possibilidade de editar minhas informações.
2. Critérios de aceite:
 - Deve ser possível editar dados pessoais, como senha e nome completo.
 - Deve aparecer uma notificação na tela após a edição de determinado campo.

Documento Digitalizado Público

Artigo de TCC - Versão Final

Assunto: Artigo de TCC - Versão Final
Assinado por: Fernando Sambinelli
Tipo do Documento: Formulário
Situação: Finalizado
Nível de Acesso: Público
Tipo do Conferência: Documento Digital

Documento assinado eletronicamente por:

- **Fernando Sambinelli, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 19/12/2023 20:51:17.

Este documento foi armazenado no SUAP em 19/12/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1530189

Código de Autenticação: eee63ce5ff

