

UNIVERSIDADE FEDERAL FLUMINENSE

Daniel de Souza Zambrano Brasil

**Aplicação móvel do cidadão para busca de
medicamentos ofertados pelo Município de Niterói
e gestão do consumo próprio**

Niterói, RJ, Brasil

2022

Daniel de Souza Zambrano Brasil

**Aplicação móvel do cidadão para busca de
medicamentos ofertados pelo Município de Niterói
e gestão do consumo próprio**

Trabalho submetido ao Curso de Bacharelado em Ciência da Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação

Orientador:
Prof. Dr Flávio Luiz Seixas

Niterói, RJ, Brasil

2022

Ficha catalográfica automática - SDC/BEE
Gerada com informações fornecidas pelo autor

B823a Brasil, Daniel de Souza Zambrano
Aplicação móvel do cidadão para busca de medicamentos ofertados pelo Município de Niterói e gestão do consumo próprio / Daniel de Souza Zambrano Brasil ; Flávio Luiz Seixas, orientador. Niterói, 2022.
70 f. : il.

Trabalho de Conclusão de Curso (Graduação em Ciência da Computação)-Universidade Federal Fluminense, Instituto de Computação, Niterói, 2022.

1. Assistência farmacêutica. 2. Aplicativo móvel. 3. Medicamento. 4. Niterói (RJ). 5. Produção intelectual. I. Seixas, Flávio Luiz, orientador. II. Universidade Federal Fluminense. Instituto de Computação. III. Título.

CDD -

Daniel de Souza Zambrano Brasil

**Aplicação móvel do cidadão para busca de medicamentos ofertados pelo
Município de Niterói e gestão do consumo próprio**

Trabalho submetido ao Curso de Bacharelado em Ciência da Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação.

Aprovado em 28 de Janeiro de 2022:



Prof. Dr. Flávio Luiz Seixas
(Orientador) - UFF



Prof. Dr. André Teixeira Pontes -
UFF



Prof^ª. Dr^ª. Daniela Gorski Trevisan -
UFF

Niterói
2022

Agradecimentos

Gostaria de agradecer e dedicar este trabalho às seguintes pessoas:

Minha mãe Ana Cláudia, meu pai Marco Aurélio, minha irmã Julia e minha namorada Daiana, que sempre me apoiaram durante meu trajeto acadêmico e me incentivaram a continuar.

Meus amigos da faculdade, Gabriel Costa, Gabriel Fiuza, Felipe Brito, Marcos Paulo e Vitor Augusto, que estiveram presente e me acompanharam durante todos estes anos.

Meu orientador Flávio e professores André e Ronaldo, que me acompanharam e prestaram toda ajuda necessária para concluir essa monografia.

Meus Professores da universidade, sem os quais não teria obtido os conhecimentos necessários para chegar aonde cheguei.

Resumo

Existe um problema crescente resultante do contínuo aumento no número de drogarias comerciais e da facilidade de aquisição de medicamentos, agravado por conta da pandemia do COVID-19, em que pessoas consomem medicamentos de maneira excessiva muitas vezes sem prescrição médica, ou não têm controle ou bons cuidados com os medicamentos que possuem em casa. Esta pesquisa foi desenvolvida no âmbito de um projeto aprovado no Programa de Desenvolvimento de Projetos Aplicados (PDPA) lançado pela Prefeitura Municipal de Niterói junto à Universidade Federal Fluminense, e tem por objetivo o desenvolvimento de um aplicativo de gerenciamento de medicamentos pessoais para smartphones, com o objetivo de auxiliar o cidadão a ter melhores práticas e controle sobre quais medicamentos ele toma, e também melhor informá-lo sobre os medicamentos oferecidos gratuitamente pelo município através do Sistema Único de Saúde e onde encontrá-los. O desenvolvimento de um produto mínimo viável para que fosse testado por representantes dos usuários finais seguiu com apoio de professores da Faculdade de Farmácia que trouxeram os requisitos e auxiliaram no levantamento de funcionalidades para a aplicação. Com o desenvolvimento avançado, foi feita uma avaliação heurística com alunos da graduação de Ciência da Computação com conhecimento na área de Interface Humano-Computador. Como resultado, alguns pontos de severidade alta foram apontados, e melhorias foram realizadas para uma segunda etapa de avaliação, dessa vez através de um teste preliminar de usabilidade com alunos da Faculdade de Farmácia, que avaliaram a usabilidade do aplicativo em um questionário utilizando o método System Usability Scale, ou SUS. Os resultados foram analisados e discutidos, com a conclusão que o aplicativo tem uma boa usabilidade para usuários experientes ou não, com uma pontuação média de 89 no SUS, e que servirá de grande utilidade para os cidadãos de Niterói, facilitando o acesso à informação sobre os medicamentos da prefeitura.

Palavras-chave: Aplicativo; Assistência Farmacêutica; Estoque de medicamentos; Niterói;

Abstract

The increasing number of retail drugstores and easy access to medication, along with the Covid-19 pandemic, make people consume drugs excessively. Moreover, people usually have no control or proper management of these medications at home. In a partnership with the Applied Projects Development Program launched by the Niterói City and the Fluminense Federal University, UFF, we developed a mobile application for the management and own pharmacies. The application helps citizens have better habits and control regarding which medicine they are taking and better inform them about those offered free by the city through their pharmaceutical care program. We develop an MVP (Minimum Viable Product) with the help of professors from the Faculty of Pharmacy. They presented the requirements and helped raise the application functionalities.

Following the development cycles, we held a heuristic evaluation with Computer Science students familiar with the Human-Computer Interface subject. As a result, we implemented several improvements for the second evaluation cycle, fixing some high severity issues noted during this phase. After that, we applied a preliminary usability test involving students from the Faculty of Pharmacy. The results were analyzed and discussed, concluding that the app presents good usability to the users, experienced or not, scoring 89 points in the System Usability Scale, and will be very useful for the citizens of Niterói, easing their access to information regarding the city hall medications.

Keywords: Mobile app; Meds management; Medications companion; Niterói;

Lista de Figuras

| | | |
|----|---|----|
| 1 | Diagrama de casos de uso para os atores usuário do aplicativo e ator prefeitura | 21 |
| 2 | Diagrama de atividades - cadastro de medicamento | 23 |
| 3 | Diagrama de atividades - registro de dose de medicamento eventual | 24 |
| 4 | Arquitetura do aplicativo e comunicação com o sistema backend | 25 |
| 5 | Telas de boas vindas (esquerda) e tutorial (direita). | 32 |
| 6 | Menu drawer | 32 |
| 7 | Tela de doses do dia | 33 |
| 8 | Tela de nova dose | 33 |
| 9 | Telas de medicamentos (esquerda) e detalhe de medicamento (direita) | 36 |
| 10 | Tela de cadastro de novo medicamento | 37 |
| 11 | Tela procurar | 38 |
| 12 | Telas de medicamentos da REMUME (esquerda) e policlínicas (direita) | 39 |
| 13 | Tela de dicas de saúde | 41 |
| 14 | Distribuição de pontuações pelo método SUS | 51 |
| 15 | Pontuação SUS por participante da avaliação | 52 |
| 16 | Distribuição de respostas de participantes pelo método SUS | 53 |

Lista de Tabelas

| | | |
|----|---|----|
| 1 | Problemas enfrentados pelos cidadãos e pela prefeitura. | 17 |
| 2 | Funcionalidades propostas e comparativo com aplicativos similares. | 17 |
| 3 | Heurística: I. Visibilidade do estado do sistema | 44 |
| 4 | Heurística: II. Correspondência entre o sistema e o mundo real | 44 |
| 5 | Heurística: III. Controle e liberdade do usuário | 45 |
| 6 | Heurística: IV. Consistência e padronização | 45 |
| 7 | Heurística: V. Reconhecimento em vez de memorização | 45 |
| 8 | Heurística: VI. Flexibilidade e eficiência do uso | 46 |
| 9 | Heurística: VII. Projeto estético e minimalista | 46 |
| 10 | Heurística: VIII. Prevenção de erros | 46 |
| 11 | Heurística: IX. Suporte para reconhecer, diagnosticar e reparar erros | 46 |
| 12 | Heurística: X. Ajuda e documentação | 46 |
| 13 | Avaliação subjetiva de usabilidade - Perguntas do método SUS | 48 |

Sumário

| | | |
|----------|--|-----------|
| 1 | Introdução | 12 |
| 1.1 | Objetivo | 14 |
| 1.2 | Organização do Texto | 15 |
| 2 | Levantamento de sistemas de gerenciamento de farmácia pessoal | 16 |
| 3 | Análise do sistema do aplicativo | 18 |
| 3.1 | Cenários | 18 |
| 3.2 | Atores | 20 |
| 3.3 | Casos de Uso | 21 |
| 3.4 | Diagrama de Atividades | 22 |
| 4 | Projeto e implementação | 25 |
| 4.1 | Bibliotecas e Frameworks chaves | 26 |
| 4.1.1 | React | 26 |
| 4.1.2 | React Native | 26 |
| 4.1.3 | AsyncStorage | 27 |
| 4.1.4 | Spring e Spring Boot | 28 |
| 4.1.5 | MongoDB | 28 |
| 4.2 | Apresentação da Interface do Usuário | 29 |
| 4.2.1 | Tela Tutorial | 31 |
| 4.2.2 | Tela Home/ Doses do dia | 31 |
| 4.2.3 | Tela Medicamentos | 35 |

| | | |
|----------|--|-----------|
| 4.2.4 | Tela Procurar | 38 |
| 4.2.5 | Tela Dicas de Saúde | 40 |
| 4.3 | Serviço Backend | 41 |
| 5 | Avaliação | 43 |
| 5.1 | Avaliação Heurística | 43 |
| 5.2 | Avaliação Subjetiva de Usabilidade | 47 |
| 5.3 | Resultados | 49 |
| 5.3.1 | Avaliação Heurística | 49 |
| 5.3.2 | Avaliação Subjetiva de Usabilidade | 50 |
| 6 | Conclusão | 54 |
| | REFERÊNCIAS | 56 |
| | Apêndice A - Avaliação Heurística | 58 |
| | Apêndice B - Teste de Usabilidade | 67 |

1 Introdução

Segundo os dados disponibilizados pelo CFF ([CONSELHO FEDERAL DE FARMÁCIA, 2020](#)) em 2021, com o número de farmácias e drogarias comerciais se aproximando dos noventa mil, existe aproximadamente uma farmácia comercial para cada 2.380 brasileiros, considerando a população brasileira totalizando duzentos e treze milhões de pessoas ([IBGE, 2021](#)). O grande número destas farmácias e o fácil acesso a medicamentos, muitas vezes sem necessidade de prescrição médica, podem resultar em um consumo inadequado de medicamentos pelos cidadãos, gerando um risco sério à saúde; muitas pessoas acabam não tendo a noção da quantidade de medicamentos que consomem, comumente tomando medicamentos sem verdadeira necessidade. Muitas vezes, são medicamentos de tratamentos anteriores que sobraram ou foram adquiridos sem a prescrição correta ou até mesmo isentos de prescrição ([MASTROIANNI et al., 2011](#)).

Um fator que trouxe à tona esse comportamento por parte da população foi a pandemia causada pelo COVID-19, que, junto às inúmeras fontes de informações disponíveis na internet, algumas corretas e outras não, levou as pessoas a tomarem medicamentos por precaução para tratar a doença, porém de maneira descuidada, com medicamentos recomendados para outros fins, sem comprovação científica de eficácia para esta doença, trazendo risco para a vida de milhões de pessoas ao redor do mundo ([OLIVEIRA et al., 2021](#)).

Um outro problema relacionado ao cuidado com medicamentos está na forma sobre como ele é armazenado. Como em geral o acesso a medicamentos é facilitado, e muitas vezes o consumidor não tem conhecimento de como armazená-los e não gerencia adequadamente este estoque domiciliar, é comum encontrar medicamentos fora da validade ou acondicionados fora das especificações dos fabricantes. O consumidor acaba tendo em casa um estoque pessoal grande destes remédios, muitas vezes fora de validade ou mal condicionados devido à maneira em que foram guardados. Esta situação gera risco à população, uma vez que o medicamento vencido ou armazenado fora das condições adequadas pode gerar danos à saúde ([MASTROIANNI et al., 2011](#)), por isso é de grande

importância auxiliar a população na gestão adequadas destes estoques domiciliares, isto é, tanto no armazenamento quanto no descarte apropriado.

O Governo Federal do Brasil oferece, através do SUS (Sistema Unificado de Saúde), um sistema de saúde pública universal, em que toda a população possui o direito à saúde universal gratuita, paga pelo próprio governo, com orçamentos da União, Estado e Municípios, com serviços que variam desde atendimentos primários até procedimentos de emergência, além de vacinas e remédios. Nele, fazem parte hospitais públicos, banco de sangue, serviços de vigilância sanitária e unidades de saúde. Diferentes municípios podem ofertar, através das unidades de saúde, medicamentos gratuitos para seus cidadãos, oferecendo uma camada a mais de informações sobre cuidados e boas práticas do que quando comparado às farmácias comerciais.

A Prefeitura de Niterói também deve oferecer a seus cidadãos usuários do SUS uma relação destes medicamentos de forma gratuita, através da política nacional de assistência farmacêutica¹:

a Assistência Farmacêutica trata de um conjunto de ações voltadas à promoção, proteção e recuperação da saúde, tanto individual como coletivo, tendo o medicamento como insumo essencial e visando o acesso e ao seu uso racional. Este conjunto envolve a pesquisa, o desenvolvimento e a produção de medicamentos e insumos, bem como a sua seleção, programação, aquisição, distribuição, dispensação, garantia da qualidade dos produtos e serviços, acompanhamento e avaliação de sua utilização, na perspectiva da obtenção de resultados concretos e da melhoria da qualidade de vida da população.

A relação de medicamentos do SUS são atualizados e disponibilizados na RENAME (Relação Nacional de Medicamentos) periodicamente e classificados entre os três componentes da assistência farmacêutica: componente básico da assistência farmacêutica; componente estratégico da assistência farmacêutica e o componente especializado da assistência farmacêutica. Para o componente básico, a aquisição é responsabilidade dos Municípios, sendo os medicamentos do municípios definidos na REMUME (Relação Municipal de Medicamentos).

A política nacional de Assistência Farmacêutica busca garantir a disponibilidade destes medicamentos selecionados no tempo correto e nas quantidades necessárias para atender as necessidades da população. O conselho menciona ainda a importância da implantação de um sistema de gestão de estoque com eficiência, para que esta programação

¹Acesso em 31 de janeiro de 2022. Disponível em: https://bvsms.saude.gov.br/bvs/saudelegis/cns/2004/res0338_06_05_2004.html

utilize dados corretos, possibilitando o traço de perfil epidemiológico, consumo histórico, ajustado, a oferta de serviços, entre outros.

É papel da prefeitura cuidar da distribuição dos medicamentos para as suas unidades de saúde participantes de acordo com a necessidade de cada região. Essa distribuição, muitas vezes, não acomoda a real necessidade do cidadão, porque há casos em que o cidadão tenta buscar um medicamento em determinada unidade, mas este está em falta. Esse tipo de problema pode ocorrer tanto porque a prefeitura não oferece visibilidade de status do estoque de determinado medicamento em uma unidade de saúde para os cidadãos, como também pela possível distribuição inapropriada de seus medicamentos, uma vez que o órgão pode não ter mapeado por completo o número de retiradas ou interesse dos cidadãos em cada medicamento nas diversas unidades de saúde.

Para o cidadão, o acesso à informação sobre a presença do medicamento em uma unidade de saúde evitaria a frustração de tentar buscar o medicamento em vão, diminuiria o risco de atrasar seu tratamento por não saber onde encontrar o medicamento necessário e melhoraria sua percepção sobre o programa de assistência farmacêutica no geral, podendo então confiar na prefeitura em oferecer todas as informações necessárias para ele.

Para a prefeitura, saber onde os cidadãos estão buscando os medicamentos com e sem sucesso auxiliaria no desenvolvimento de uma melhor solução para distribuição dos mesmos. Dessa forma, os cidadãos serão melhor atendidos e seu programa mostrará ser mais eficiente.

1.1 Objetivo

O objetivo geral é desenvolver e propor uma solução tecnológica para smartphones, voltada às necessidades da prefeitura municipal e do cidadão de Niterói. Em uma primeira versão, oferecer para o Cidadão uma plataforma de controle e gerenciamento dos seus medicamentos pessoais e facilitar o acesso à informação sobre onde os medicamentos oferecidos pela Prefeitura de Niterói.

Esta monografia possui os seguintes objetivos específicos:

1. Realizar um estudo de aplicações de gestão de medicamentos existentes no mercado.
2. Elaborar cenários e diagramas de casos de uso e atividades para auxílio no levantamento de requisitos.

3. Desenvolver e apresentar um aplicativo de smartphones em um escopo de MPV, com funcionalidades de gestão de estoque de medicamentos pessoais, dicas de saúde e busca de medicamentos oferecidos e unidades de saúde presentes na política de assistência farmacêutica.
4. Realizar etapas de avaliações para medir a qualidade do aplicativo e analisar os resultados.

A elaboração e desenvolvimento do aplicativo é tema do projeto “Utilização de aplicativo de celular para otimização do sistema de distribuição de medicamentos em Niterói” aprovado no PDPA (Programa de Desenvolvimento de Projetos Aplicados), um edital lançado pela Prefeitura de Niterói em conjunto com a Fundação Euclides da Cunha e também a Universidade Federal Fluminense, em 2019. Os requisitos do aplicativo foram levantados junto à coordenação do projeto, visando representar as necessidades dos usuários finais.

1.2 Organização do Texto

Essa monografia está dividida em 6 seções. A Seção 2 apresenta um levantamento de aplicações similares. A Seção 3 descreve uma análise de alto nível do sistema, e exemplifica alguns casos de uso e fluxos de atividade que o usuário final poderá exercer. A Seção 4 abrange a parte técnica no desenvolvimento, como o uso de bibliotecas e frameworks utilizados, bem como a apresentação de telas do aplicativo. Por fim, a Seção 5 discute dois tipos de avaliações levantadas a partir do desenvolvimento, que trouxeram pontos de ajuste e melhorias no sistema.

2 Levantamento de sistemas de gerenciamento de farmácia pessoal

Essa Seção levanta algumas soluções similares usadas no gerenciamento de farmácia pessoal. Como funcionalidades básicas de um sistema de gerenciamento de medicamentos, o usuário deve ser capaz de controlar estoque e agendar as doses que deve tomar para cada medicamento. Para isto, existem diversos aplicativos no mercado, pagos e gratuitos, que oferecem estas funcionalidades de diversas maneiras. Alguns destes aplicativos foram analisados em conjunto com a equipe do projeto e contribuíram para o levantamento das funcionalidades e layout do aplicativo. Os dois aplicativos de maior influência para o desenvolvimento da aplicação foram “Lembrete de Remédios e Pílula - Medisafe” ([MEDISAFE, 2021](#)) e “Pills Time Farmácia assistente para a sua saúde” ([MOBILE CREATURES, 2021](#)).

Além disso, existem também problemas relacionados a interação entre o cidadão e prefeitura, no que diz respeito aos medicamentos oferecidos gratuitamente pelo governo. A Tabela 1 cita quatro dos problemas que o cidadão e o Prefeitura Municipal de Niterói podem enfrentar no âmbito de assistência farmacêutica e farmácia pessoal. Estes problemas foram levantados pela equipe do projeto e serviram de base para elaboração das principais funcionalidades. Para o primeiro problema, os aplicativos citados no parágrafo anterior bem como outros aplicativos disponíveis na internet propõem soluções para o usuário de alguma uma maneira com algumas funcionalidades parecidas e outras exclusivas. Os três outros itens, por se tratarem justamente da relação entre um órgão governamental e seus cidadãos, não estão presentes nestes outros aplicativos, que são voltados para um público mais amplo e de uso geral.

Tendo conhecimento desses problemas, é possível levantar soluções em formas de funcionalidades básicas que o aplicativo deve ter para que atenda as necessidades dos seus usuários finais. A Tabela 2 lista algumas dessas funcionalidades importantes, e também indica se esta funcionalidade está presente ou não em um dos aplicativos existentes citados anteriormente. Algumas das funcionalidades foram desenvolvidas e estão presentes no

Tabela 1: Problemas enfrentados pelos cidadãos e pela prefeitura.

| Visão | Problema |
|------------|--|
| Cidadão | Dificuldade na gestão de medicamentos pessoais. |
| Cidadão | Acesso dificultado à relação de medicamentos oferecidos pela prefeitura. |
| Cidadão | Insatisfação na busca de um medicamento em falta em uma unidade de saúde. |
| Prefeitura | Não ter informações sobre as tentativas de baixa de medicamentos pelos cidadãos. |

Fonte: o autor

Tabela 2: Funcionalidades propostas e comparativo com aplicativos similares.

| Funcionalidade | Aplicativo | | |
|--|------------|------------|--------------|
| | Medisafe | Pills Time | App proposto |
| Controle de estoque de medicamentos | Possui | Não possui | Possui |
| Agendamento de doses e lembretes para medicamentos cadastrados | Possui | Possui | Possui |
| Busca de medicamentos oferecidos gratuitamente pelo município | Não possui | Não possui | Possui |
| Registro de feedback de tentativa de baixa de medicamentos gratuitos | Não possui | Não possui | Previsto |

Fonte: o autor

MVP, enquanto outras estão previstas para o aplicativo final.

O levantamento dessas e outras funcionalidades nestes sistemas já existentes auxilia na elaboração dos requisitos necessários para a solução proposta, e até mesmo funcionalidades similares trazem uma melhor percepção de como aplicá-las ao aplicativo desenvolvido, uma vez que cada solução é elaborada de uma maneira diferente, portanto a análise dessas múltiplas implementações permite que pontos positivos sejam apropriados e pontos negativos descartados ou melhorados.

Outro ponto importante a ser mencionado é o escopo das funcionalidades de cada aplicativo. O aplicativo da solução tem um conjunto de funcionalidades específicas planejadas para atender ao cidadão de maneira fácil e não complicada. Os demais aplicativos citados possuem diversas outras funcionalidades que podem não ser úteis para maioria das pessoas, e acabam trazendo ao sistema uma complexidade maior, e com isso, maior dificuldade no aprendizado. Além do mais, apesar de gratuitos, esses aplicativos também possuem funcionalidades adicionais que só podem ser acessadas através de pagamento, enquanto a solução proposta trará todas as suas funcionalidades de maneira gratuita.

3 Análise do sistema do aplicativo

Essa seção aborda certas definições que são necessárias para se ter melhor entendimento de todas as entidades envolvidas no sistema, e está organizada em quatro subseções: a subseção 3.1 retrata os cenários, que ajudam a concretizar fluxos de utilização do aplicativo; a subseção 3.2 descreve os atores do sistema, que ajudarão a entender o que se espera da aplicação; a subseção 3.3 retrata os casos de uso, que junto aos cenários levantados ajudam estabelecer como os atores do sistema vão interagir com ele. Por fim, a subseção 3.4 ilustra dois diagramas de atividade que descrevem alguns fluxos de uso do aplicativo.

3.1 Cenários

Cenários são contextos que podem ser criados em cima de diversos casos de uso para compor sua descrição. Neles devem ter informações como o ator, pré condições, fluxos normais e alternativos, bem como uma pós-condição que deve ser atendida após a execução do cenário.

De acordo com Sommerville (SOMMERVILLE, 2011), os cenários auxiliam na adição de detalhes à definição geral de requisitos. Um cenário geralmente cobrirá poucas interações, e geralmente começa com um esboço de uma interação. Uma maneira de descrever cenários é através de um texto narrativo simulando de fato uma situação na vida do ator em questão, em que a motivação, sequencia de operações e resultados da interação são descritos, bem como possíveis fluxos alternativos que podem acontecer caso algo dê errado.

Para o aplicativo, foram estimados quatro cenários de uso críticos para a elaboração de um produto mínimo viável. Como a Prefeitura de Niterói é um ator que não usará o aplicativo e sim receberá os benefícios dele indiretamente, todos os atores dos cenários descritos abaixo serão, inicialmente, os cidadãos da cidade de Niterói:

1. Cadastro de Medicamento

O usuário deseja cadastrar um medicamento de um novo tratamento que está fazendo, no aplicativo. Para isso, ele abre o aplicativo, toca na aba que o leva para tela de medicamentos, e lá seleciona o botão para adicionar um novo medicamento. Com isso, ele preenche as informações do medicamento, como nome, estoque e horário das doses, caso necessário. Por fim, toca em salvar para finalizar o cadastro. Se preencher uma ou mais informações de maneira incorreta, o aplicativo deve informar qual informação foi preenchida incorretamente e instruir o usuário sobre como corrigir seu erro. Se tudo estiver preenchido corretamente, o medicamento será cadastrado com sucesso e o usuário redirecionado para a tela principal. Caso o medicamento registrado possua doses agendadas, os registros das doses aparecerão com seus respectivos horários e, para cada horário marcado, uma notificação de lembrete será agendada.

2. Buscar Policlínicas

O usuário gostaria de retirar um medicamento gratuitamente em uma unidade de saúde mas não sabe qual está mais perto de seu endereço. Ele então abre o aplicativo e toca na aba que o redireciona para tela de busca, e seleciona a opção “policlínicas”. A relação de policlínicas que oferecem medicamentos é exibida, contendo informações de contato e endereço. Para isso, o usuário precisa ter uma conexão ativa com a internet, e caso esteja sem conexão o sistema deve informá-lo para que ele tente reaver a conexão ou tentar novamente mais tarde. Caso esteja conectado e a lista de unidades de saúde apareça, ele pode tocar em um dos registros telefônicos da unidade de interesse para fazer uma chamada ou tocar no botão que o redirecionará para o aplicativo de mapa, que irá sugerir o melhor caminho até o destino.

3. Tomar dose agendada

O usuário recebeu uma notificação que o lembrou de tomar um dos medicamentos que havia cadastrado no aplicativo. Após tomar o medicamento, ele deseja marcar que sua dose foi tomada no horário correto. Ele abre o aplicativo e na tela inicial são exibidas as doses que ele deve tomar no dia, incluindo a dose referente à notificação recebida. Ele toca no registro da dose em questão e depois toca no botão “tomar”. Caso ele tenha esquecido e se atrasou na hora de marcar a dose tomada, ele pode tocar no registro da dose, tocar em “ajustar” e inserir o horário correto em que tomou a dose. Após isso, ao confirmar a dose tomada o aplicativo deve atualizar o registro de doses e também a quantidade de estoque do respectivo medicamento.

4. Tomar dose eventual

O usuário não está se sentindo bem e toma um medicamento isolado, então deseja registrar esse consumo no aplicativo. Caso ele não o medicamento cadastrado, ele deve seguir o fluxo de cadastro de medicamento e marcar no cronograma do mesmo como “tomar quando necessário”. Com o medicamento cadastrado, o usuário toca no botão “nova dose” presente na tela inicial e é redirecionado para a tela de registro de doses esporádicas, lá ele seleciona o horário, a quantidade e qual medicamento foi tomado. Após preencher essas informações, ele toca no botão “confirmar” e um novo registro aparece na tela inicial para a dose pontual tomada.

3.2 Atores

Os atores ajudam a entender o objetivo da aplicação. Eles representam quem usará de fato o sistema. Um ator é um papel que uma pessoa exerce quando interage com o sistema. Ele pode ser representado por diferentes instâncias, chamadas personas, que podem atribuir informações sobre uma pessoa fictícia que atuará no sistema. Esta persona pode receber nome, idade, emprego, e quaisquer outras características que possam ajudar a trazer uma ideia do usuário real ao sistema desenvolvido.

Para o aplicativo, dois atores podem ser mencionados, um sendo a própria entidade da Prefeitura de Niterói como um todo, que deseja coletar dados de interesse com a interação do cidadão da cidade com os medicamentos oferecidos gratuitamente, e o próprio cidadão da cidade, que de fato usará o aplicativo. A entidade da prefeitura, na verdade, será composta por funcionários públicos que extrairão os dados gerados pelos cidadãos durante o uso do aplicativo. Por outro lado, o ator "cidadão" representa todo o tipo de usuário público-alvo para o aplicativo, sendo homens e mulheres de diferentes idades e condições diferentes, que têm em comum o desejo ou necessidade de melhorar sua gestão de medicamentos pessoais e ter acesso a mais informações sobre os medicamentos da REMUME.

Para garantir maior engajamento do usuário com a aplicação, é necessário que todas as funcionalidades e ferramentas disponíveis atendam as necessidades destes atores e que façam que continuem utilizando o sistema. Em outras palavras, entender o tipo de pessoa que utilizará a aplicação auxilia na elaboração de uma interface e funcionalidades atraentes e úteis para eles, de forma a garantir que continuem utilizando o sistema após um primeiro contato. No caso da solução proposta, deve-se garantir que pessoas tanto jovens e com

certo conhecimento tecnológico quanto idosos leigos na área consigam utilizar o aplicativo de maneira contínua e sem dificuldades.

3.3 Casos de Uso

A elaboração de casos de uso também auxilia na definição de requisitos, uma vez que estes identificam os atores envolvidos em certa interação e melhor descrevem-na. A interação é então acrescentada por informações adicionais que descrevem a interação com o sistema. Essa informação adicional pode ser textual ou apresentada através de diagramas como o de sequência ou de estados da UML (do inglês *unified modeling language*) (SOMMERVILLE, 2011).

Para representar os casos de uso em si, diagramas de alto nível chamados diagramas de caso de uso são utilizados. Os casos de uso são bem semelhantes aos cenários, com algumas definições descrevendo um conjunto de cenários para apenas um caso de uso. Por exemplo, múltiplos cenários podem ser levantados para uma mesma interação, seguindo diversos fluxos diferentes, tratando diferentes exceções, e estes estariam compondo um único caso de uso. Na Figura 1 são exemplificados casos de uso que descrevem alguns requisitos funcionais do sistema:

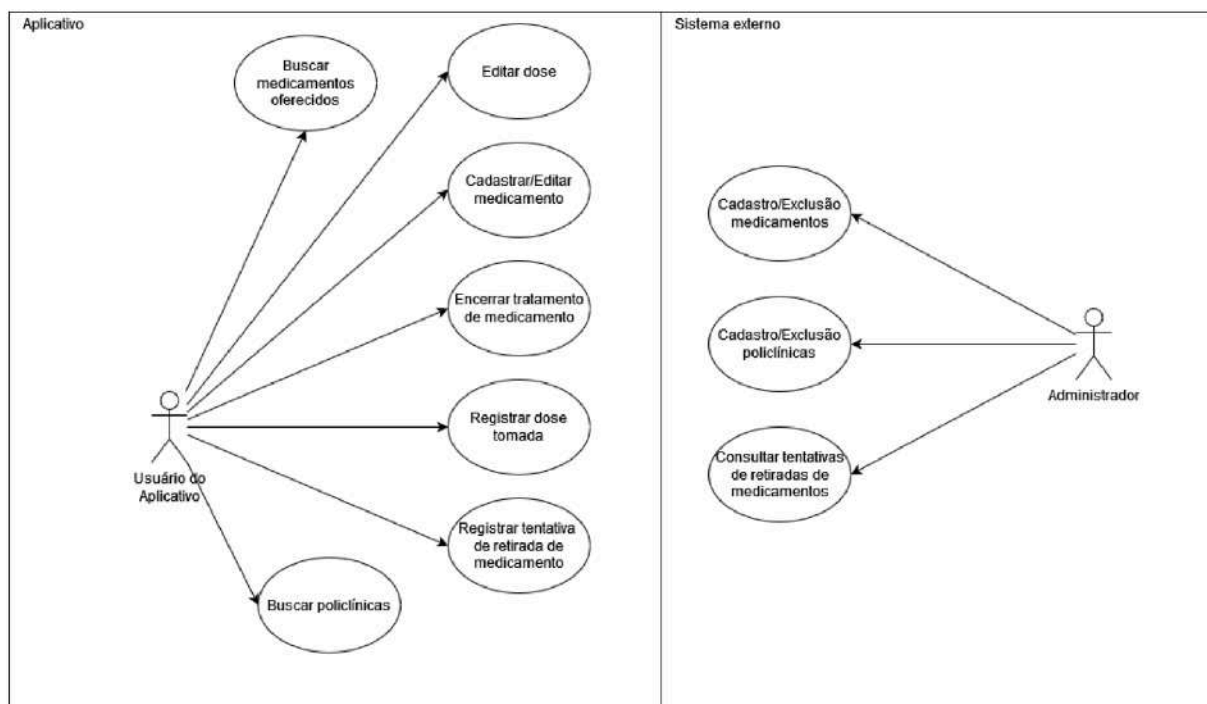


Figura 1: Diagrama de casos de uso para os atores usuário do aplicativo e ator prefeitura

Fonte: O autor

Os cenários mencionados na subseção 3.1 abordam quatro dos casos de uso principais aplicáveis ao cidadão enquanto utiliza o aplicativo. Outros casos de usos que definem funcionalidades pontuais e que podem ser mencionados são o de busca de medicamentos oferecidos, que permite que o usuário marque o início de um tratamento de um medicamento que adquiriu gratuitamente ou que registre falha na tentativa de retirada do medicamento em unidade de saúde, embora esta última funcionalidade não está dentro do escopo do produto mínimo viável. Uma outra funcionalidade elucidada pelo diagrama de casos de uso é a de encerramento de tratamento, que indica que o usuário pode interromper o tratamento cadastrado no aplicativo em qualquer momento.

Do ponto de vista externo ao aplicativo, três casos de uso principais são destacados:

1. Atualização dos medicamentos presentes na REMUME.
2. Atualização das unidades de saúde que oferecem medicamentos.
3. Consulta nas tentativas de retirada de medicamentos nas unidades de saúde (análise comportamental dos cidadãos).

3.4 Diagrama de Atividades

O diagrama de atividades é um tipo de modelo de contexto, que descreve os processos, automatizados ou humanos, em que sistemas específicos serão usados. Eles têm a finalidade de mostrar as diferentes atividades que vão compor um processo ou fluxo do sistema de uma atividade para outra (SOMMERVILLE, 2011).

No diagrama de atividades, são expostos o início e fim de um processo, atividades que devem ser realizadas, fluxos de execução de uma atividade para a outra e também um mecanismo que permite observar a coordenação entre atividades. As figuras abaixo representam diagramas de atividades com os fluxos principais de uso do aplicativo.

A Figura 2 descreve a interação de atividades presentes no cadastro de um medicamento. O fluxo inicia pelo círculo preto no canto superior esquerdo da figura, que denota o ator como sendo o cidadão de Niterói. Ele tem se aplicativo aberto, segue o caminho de telas até o cadastro do medicamento. Caso o novo medicamento tenha horário marcado para suas doses, elas são criadas com as datas correspondentes e notificações locais também são agendadas para que sejam disparadas no horário correto, indicadas pelo fluxo que termina no canto inferior direito. Caso contrário, apenas o medicamento é criado e nenhuma dose ou notificação é inserida.

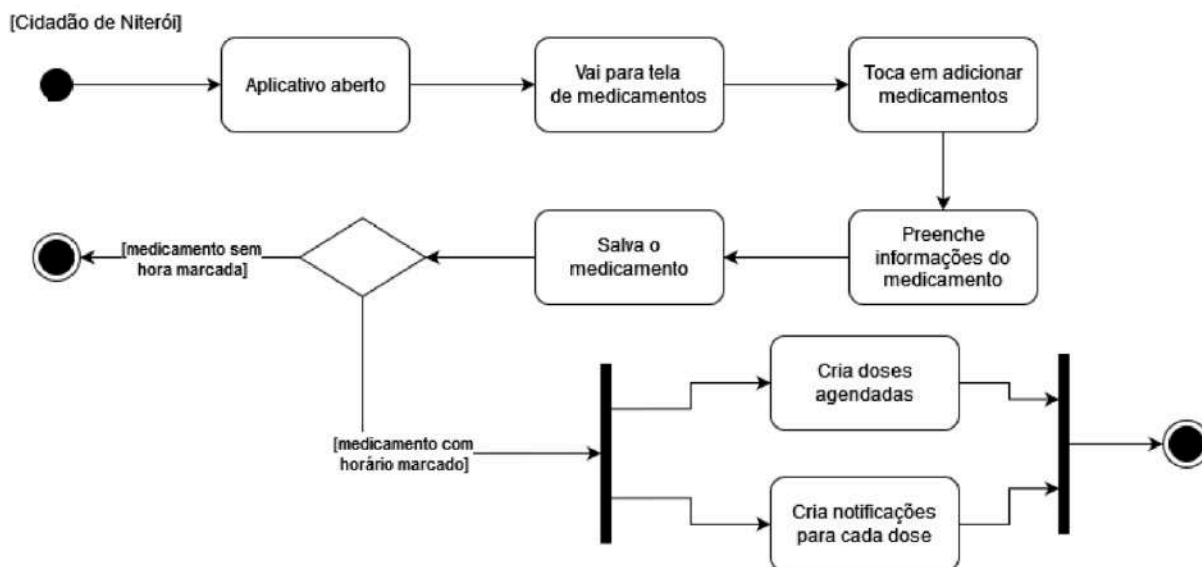


Figura 2: Diagrama de atividades - cadastro de medicamento

Fonte: O autor

A Figura 3 descreve o fluxo do usuário para registrar uma dose de um medicamento sem doses agendadas. Novamente, a sequência começa com o cidadão de Niterói tendo o aplicativo aberto, caso não possua o medicamento cadastrado previamente, é necessário seguir o fluxo de cadastro de medicamentos resumido neste diagrama. Quando o usuário possuir um medicamento cadastrado sem doses agendadas, um novo botão aparecerá na tela principal para que ele marque registros de doses deste medicamento. Essa tela é melhor descrita na subseção 4.2.2.

Estes dois diagramas fazem menção a dois fluxos principais de uso para o usuário, uma vez que as funcionalidades principais do aplicativo incluem o cadastro de medicamentos e acompanhamento de doses. A elaboração destes diagramas contribuem para o desenvolvimento, mantendo as regras de negócio coesas e navegação entre telas mais explícitas. A próxima seção entrará em mais detalhes sobre o projeto em nível técnico e a interface do usuário no aplicativo.

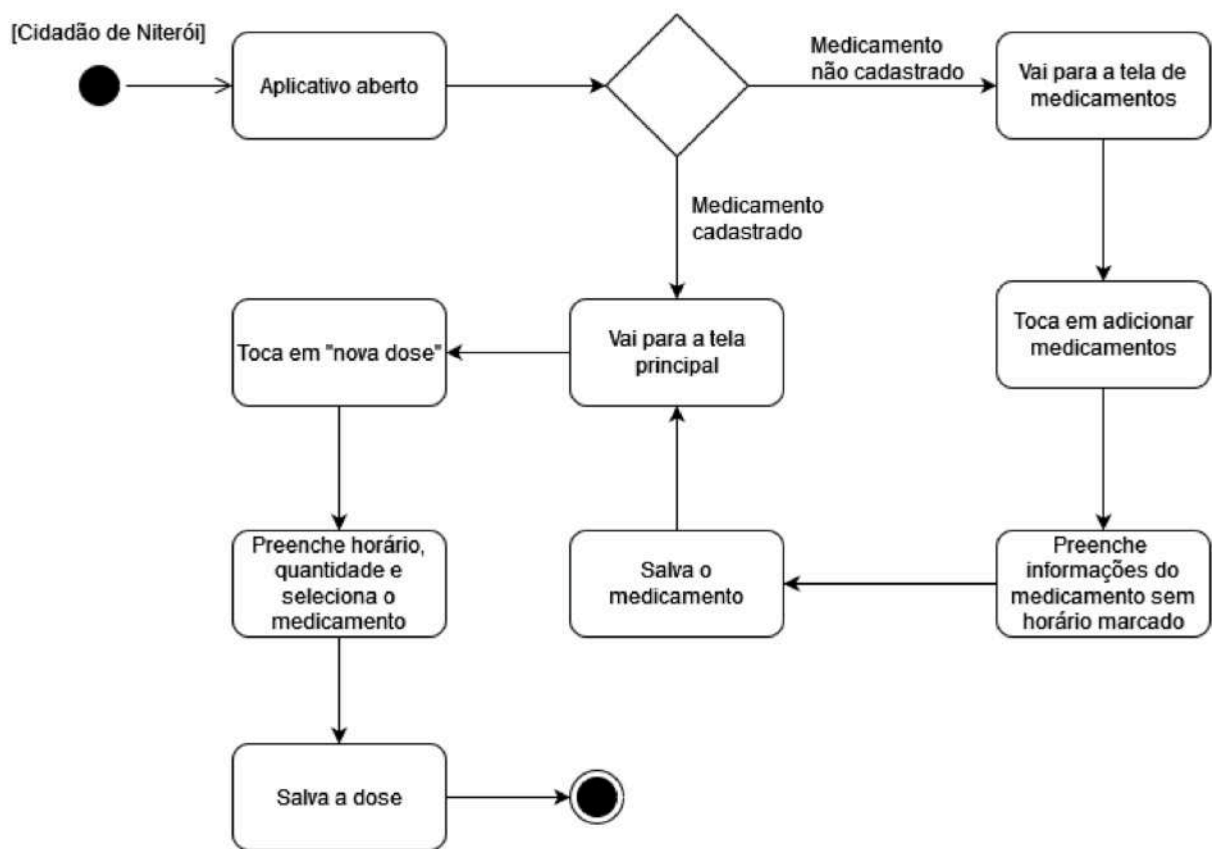


Figura 3: Diagrama de atividades - registro de dose de medicamento eventual

Fonte: O autor

4 Projeto e implementação

Esta Seção trata os componentes técnicos, ou seja, linguagem, bibliotecas e frameworks que foram utilizados durante o desenvolvimento, assim como a própria arquitetura e interface do aplicativo. O desenvolvimento do produto como um todo pode ser dividido no aplicativo para smartphones em si e um serviço de backend externo ao aplicativo que este usará para fazer consultas e trazer dados para o usuário. Inicialmente, o serviço externo contará com algumas requisições através de uma API REST que retornarão os dados referentes as policlínicas e os medicamentos oferecidos pela prefeitura. A figura 4 ilustra estes componentes e como comunicam entre si. A parte de renderização e armazenamento de dados do usuário ficam integralmente dentro do dispositivo, enquanto os dados não persistidos são consultados no serviço de backend. Posteriormente, este serviço se estenderá para registrar as experiências do usuário nas interações com as unidades de saúde em suas tentativas de retirada de medicamentos gratuitos.

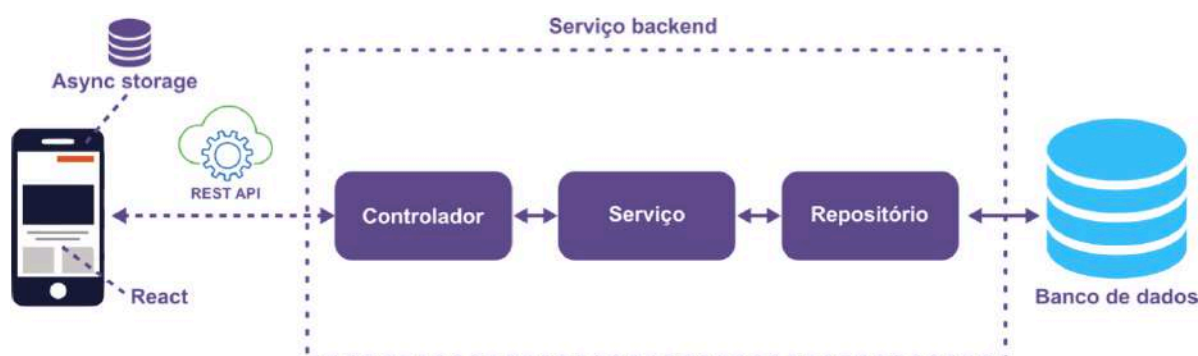


Figura 4: Arquitetura do aplicativo e comunicação com o sistema backend

Fonte: O autor

A Subseção 4.1 descreve as bibliotecas e frameworks principais utilizados para o desenvolvimento do aplicativo, enquanto a Subseção 4.2 apresenta a interface do sistema. Por fim, a Subseção 4.3 discute a implementação do serviço backend, que é o serviço online no qual o aplicativo se conectará para trazer dados não persistidos.

4.1 Bibliotecas e Frameworks chaves

4.1.1 React

O React é uma biblioteca de JavaScript para desenvolvimento front-end que busca facilitar a maneira como objetos são renderizados em tela. Os componentes React implementam um método de renderização que permite receber código com sintaxe JSX (do inglês - *JavaScript Syntax Extension*), que é uma extensão da linguagem JavaScript para estruturar componentes de renderização de maneira similar ao XML. A próxima Seção explicará que o React Native também utiliza o React como ferramenta de renderização e por isso o código é escrito em componentes, que podem ser definidos via função ou via classe. Para o aplicativo, todos os componentes foram escritos como componentes funcionais.

Os componentes definidos pelo usuário, ou seja, componentes diferentes dos pré existentes do framework podem receber como parâmetros propriedades (ou “props”) que permite a comunicação de dados de um componente pai para um componente filho, em uma espécie de comunicação direta. Para uma comunicação indireta, ou uma comunicação de componente filho para um pai, uma função do componente pai pode ser passada como um “prop” para o filho, e então este pode invocar a função passando valores presentes apenas dentro de seu escopo.

Outra entidade importante do framework React é o “State”, que permite controlar o estado de um componente de acordo com sua necessidade. Diferente dos props, que são imutáveis, um estado pode guardar um ou mais variáveis cujos valores podem ser alterados de acordo com a interação do usuário. O estado é inicializado e gerenciado pelo componente, e não passado como parâmetro. O estado de componentes pode ser gerenciado através de *Hooks*, que são funções que permitem componentes funcionais a se prenderem ao estado e ciclo de vida do React (anteriormente feito somente com componentes por classes)([FACEBOOK](#), s.d.).

4.1.2 React Native

O aplicativo foi desenvolvido na linguagem JavaScript através do *framework* React Native, que permite a utilização da biblioteca React no desenvolvimento de aplicativos móveis para os sistemas iOS e Android de forma nativa. Como na concepção do projeto foi estimado como requisito de sistema operacional apenas em Android, toda parte de projeto e implementação foi pensada apenas com este sistema operacional em mente.

O React Native é executado no thread principal de execução da aplicação, que é onde maior parte do código de aplicação é executado. Ele é responsável por despachar eventos como os de renderização para a tela, motivo pelo qual também pode ser chamado de thread IU, para interface de usuário. (GOOGLE, s.d.). Apesar de similiar ao React, o React Native utiliza componentes nativos do sistema para construção de tela.

Ao longo de todo o código da aplicação o uso de componentes, props, estados e hooks é evidente e necessário para o correto funcionamento de telas e processamento de dados. Apesar de ser um framework de *front-end*, é possível aproveitar a porção de JavaScript da aplicação para criar a lógica de cada tela e até mesmo utilizar bibliotecas que cuidam do armazenamento local e notificação para dispositivo sem necessidade de outra linguagem, servindo como uma espécie de back-end (ou lógica da aplicação e comunicação com o banco de dados) local, na porção em que não é necessária uma conexão com a internet. Isto será visto na próxima Seção.

O React Native é um framework muito utilizado atualmente e com uma comunidade de desenvolvedores cada vez maior, o que oferece para desenvolvedores uma grande variedade de bibliotecas com diferentes propósitos e funcionalidades. Diversas bibliotecas foram utilizadas no desenvolvimento do aplicativo tanto para questões de interface quanto à lógica da aplicação.

4.1.3 AsyncStorage

O AsyncStorage é uma biblioteca para React Native que funciona como um sistema assíncrono e persistente para armazenamento de dados no formato chave-valor. Foi avaliada a necessidade de utilização do banco de dados local (SQLite, no caso do Android) para armazenamento dos dados de medicamentos dos usuários, mas o AsyncStorage mostrou-se não apenas suficiente como uma maneira eficiente de armazenar e recuperar os dados e prepará-los para renderização em tela.

Como mencionado, ele funciona armazenando informação de forma textual em chaves que então são persistidas internamente no dispositivo. Esse tipo de modelo não é muito eficaz para tratar de dados muito grandes ou complexos, mas conforme os dados armazenados foram pensados, o modelo facilita a lógica de banco de dados tradicionais de consultas para trazer os objetos inteiros de uma só vez. Para isso, todos os dados salvos no AsyncStorage foram convertidos em texto de modo que pudessem ser serializados de volta para o formato JSON, formato em que os objetos são tratados durante a lógica de código.

Neste primeiro momento, a aplicação utiliza o AsyncStorage como forma de armazenamento local básica para armazenamento dos medicamentos e outros dados de configurações do usuário. Por se tratar de uma estrutura de objetos mais simples, é um substituto válido a um banco de dados embutido próprio, como o SQLite para Android. Ao carregar uma tela que necessita dos dados, é feita uma leitura no AsyncStorage com a chave referente aos medicamentos e um objeto na forma JSON é retornado contendo a lista de todos os medicamentos cadastrados pelo usuário, já apropriado para manipulação de código, uma vez que o JavaScript vai entendê-lo como objeto válido da linguagem, e também para renderização. Outros parâmetros também são salvos pelo AsyncStorage utilizando outras chaves, como configurações de filtro.

4.1.4 Spring e Spring Boot

Spring é um framework para desenvolvimento Java. Tem como um dos propósitos simplificar o desenvolvimento de aplicações, com injeção de dependência sendo um dos seus principais recursos. Spring Boot é um módulo do framework Spring que facilita o desenvolvimento de uma aplicação web. É uma solução que permite criar um programa independente com pouca ou nenhuma configuração. Estes foram usados para a criação de um serviço RESTful que se comunicará com o aplicativo para trazer dados armazenados em um banco de dados externo. Com o Spring, o foco do desenvolvimento fica na lógica de negócio para a aplicação, enquanto maior parte da estrutura é tratada pelo framework. A comunicação com banco de dados para leitura e escrita é tratada utilizando repositórios já implementados pelo framework, e é necessário apenas definir como são os dados que serão manipulados para o armazenamento e criação das APIs de consulta.

4.1.5 MongoDB

MongoDB é software de banco de dados não relacional, que trabalha orientado a documentos, sem a necessidade de uma estrutura relacional pré definida para armazenamento dos dados. É um software livre e de código aberto. O MongoDB foi o banco de dados de escolha para a aplicação por ser mais simples de se inicializar, uma vez que não há necessidade de definir estruturas para tabelas. Como a quantidade de dados a ser navegada pela aplicação não é muito grande, outros requisitos como escalabilidade ou velocidade de processamento de dados maiores não precisaram ser considerados, e na prática algum banco de dados relacional também seria uma alternativa viável.

4.2 Apresentação da Interface do Usuário

Essa Subseção entrará em mais detalhes sobre a aplicação em si, descrevendo cada tela e como componentes se comunicam entre si. Nesta versão inicial, existem onze telas no aplicativo, algumas contendo apenas informações para consulta e outras, mais importantes, com uma lógica mais robusta e mais primária ao sistema.

Houveram três fases de desenvolvimento do aplicativo. A primeira se deu junto ao levantamento de requisitos e das análises de soluções similares discutidas na Seção 2, em que a implementação seguiu de forma continuada durante até o começo da avaliação heurística, que marcou o início da segunda fase de desenvolvimento, onde pontos críticos apontados na avaliação foram priorizados para serem corrigidos ou adicionados. Logo em seguida deu-se início à terceira fase de desenvolvimento, após o teste de usabilidade, em que mais uma rodada de melhorias foi realizada no aplicativo, dessa vez levando em consideração os pontos levantados por avaliadores representantes dos usuários finais. A maior parte das questões mencionadas diz respeito à interface do aplicativo e sua interação com o usuário, portanto essas melhorias foram voltadas principalmente para a questão de apresentação do sistema.

Antes de falar sobre as telas, é importante entender os tipos de navegadores implementados para que a comunicação de telas faça mais sentido posteriormente. Existe uma biblioteca de navegação para React Native que oferece diversos tipos de navegação comumente vistos em aplicativos de smartphones, o React Navigation¹. Com ele foram utilizadas três dos tipos de navegadores mais comuns em aplicativos: *Drawer Navigator*, *Stack Navigator* e *Tab Navigator*.

Drawer Navigator: Cria uma espécie de “gaveta” disponível para ser puxada a partir do canto esquerdo de uma tela, por isso o nome em inglês *drawer*. Ao puxar o componente, que se estende até além do meio da tela, é exibido um menu de navegação, que permite o usuário acessar outras telas de maneira mais rápida, uma vez que estas podem não estar imediatamente acessíveis a partir da tela atual do usuário.

Stack Navigator: Um navegador fundamental para aplicativos, funciona criando uma pilha de telas que permite fácil retorno para o usuário. Com este, ao ser redirecionado, o usuário é enviado para uma nova tela e geralmente um botão de voltar é exibido no canto superior esquerdo, que se pressionado deve retornar para a tela

¹Documentação disponível no site oficial em <https://reactnavigation.org/docs/getting-started>.

anterior, podendo restaurar o estado original da página ou não.

Tab Navigator: Este navegador cria abas que podem ficar na parte inferior de telas, persistentemente, que permite o usuário acessar telas importantes com um único toque. É um tipo de navegador que permite manter os fluxos mais importantes de interação próximos ao usuário e sempre disponíveis quando necessário.

Conhecendo esses navegadores e suas funcionalidades, é evidente que certas telas devem estar presentes em um ou mais navegadores, e ao mesmo tempo não é necessário ou melhor para o usuário inserir todas as telas em todos os navegadores, uma vez que pode prejudicar sua experiência com tantas opções que não foram organizadas de maneira intuitiva.

Algo tão importante quanto a funcionalidade de um sistema é a maneira como este sistema é apresentado a seu utilizador. Para que se tire proveito de toda sua capacidade e funcionalidade, a aplicação precisa fornecer ao usuário um conjunto de características que combinadas proporcionam uma experiência positiva na interação dele com o sistema. Essas características definem uma qualidade de uso para a aplicação.

Jakob Nielsen, consultor e cientista da computação, conhecido por promover a usabilidade em sistemas web iniciou o movimento da “engenharia de usabilidade” para efetuar melhorias em interfaces de usuário de maneira eficiente e econômica, criando também a avaliação heurística e diversos métodos de usabilidade. Segundo ele, a facilidade de aprendizado e uso de uma interface, bem como a experiência ou satisfação do usuário em decorrência do uso estão relacionados com a **usabilidade** (NIELSEN, 1994). Para que ele utilize um sistema como um todo, a interface deve auxiliá-lo e conduzi-lo de maneira natural e coesa, e não servir como um empecilho que atrapalhe ele. Um outro critério estabelecido por Nielsen é o de **comunicabilidade**, que diz respeito a interface conseguir refletir as intenções de quem criou o sistema para o usuário, e, uma vez sabendo as intenções de design, o usuário poderá aproveitar melhor a aplicação.

A prática da utilização de elementos comuns e que estão presentes em outras aplicações traz para o usuário uma sensação de familiaridade, de modo que mesmo que ele não tenha utilizado um sistema antes, ele intuitivamente saberá como certos elementos devem ou não se comportar, e também o que esperar deles. Uma campainha, maçaneta ou um interruptor de luz não diferem tanto de um navegador, botão ou gesto de um aplicativo, isto é, implementados corretamente fazem com que o sistema se comporte da maneira que o usuário espera. Este princípio, de semelhança entre elementos, auxilia na **facilidade de**

aprendizado do usuário, que se refere ao tempo e esforço necessário para que ele aprenda a utilizar o sistema com certo desempenho e aptidão (NIELSEN, 1994). Além disso, outros mecanismos podem ser utilizados para trazer esta facilidade, como a utilização de guias de ajuda ou tutoriais que ensinam melhor como estes e outros elementos se comportam dentro do sistema. Esse mecanismo é mencionado na Seção 4.2.1, com a tela de tutorial.

Além destes, o aplicativo foi desenvolvido tendo também outros critérios em mente visando proporcionar ao usuário final uma melhor experiência. Ao longo deste capítulo, outros critérios serão mencionados em junção à exibição das telas presentes no aplicativo. Além disso, foi feita uma pequena avaliação heurística que pontuou locais de melhorias que depois foram implementadas para versão atual. Essa avaliação é discutida na Seção 5.

4.2.1 Tela Tutorial

Essa tela é a primeira tela que usuário vê ao abrir o aplicativo pela primeira vez. Uma tela de boas vindas que informa ao usuário instruções de como realizar algumas funções dentro da aplicação. Ela apresenta um layout em páginas em que o usuário desliza como em um livro para seguir pelas instruções. A Figura 5 mostra o que o usuário vê quando acessa o aplicativo pela primeira vez, assim como uma página do tutorial contendo um texto instrucional e uma curta imagem animada demonstrando uma determinada ação.

As demais páginas do tutorial seguem o padrão de imagem animada demonstrativa seguida do texto explicativo. Ao final do tutorial, um botão de confirmação é exibido para o usuário informar que entendeu as instruções, e, quando tocado, o usuário é redirecionado para a tela principal, de doses. O tutorial pode ser sempre acessado a partir do navegador “drawer” ou gaveta, mostrado na Figura 6.

4.2.2 Tela Home/ Doses do dia

A tela principal do sistema, comumente chamada de Home em outros sistemas, é a tela de doses do dia, que mostra ao usuário as doses que ele tem agendadas no dia selecionado, permite atualizar os status de cada uma ou cadastrar uma nova dose eventual. Assim como maior parte das telas, ela é composta por três seções. Na parte superior encontra-se um cabeçalho com o nome da tela e um botão de navegação no canto superior esquerdo que oferece uma alternativa a abertura do navegador de gaveta. Além deste botão, um outro botão de filtro está presente (este, no entanto, possui uma exibição condicional de

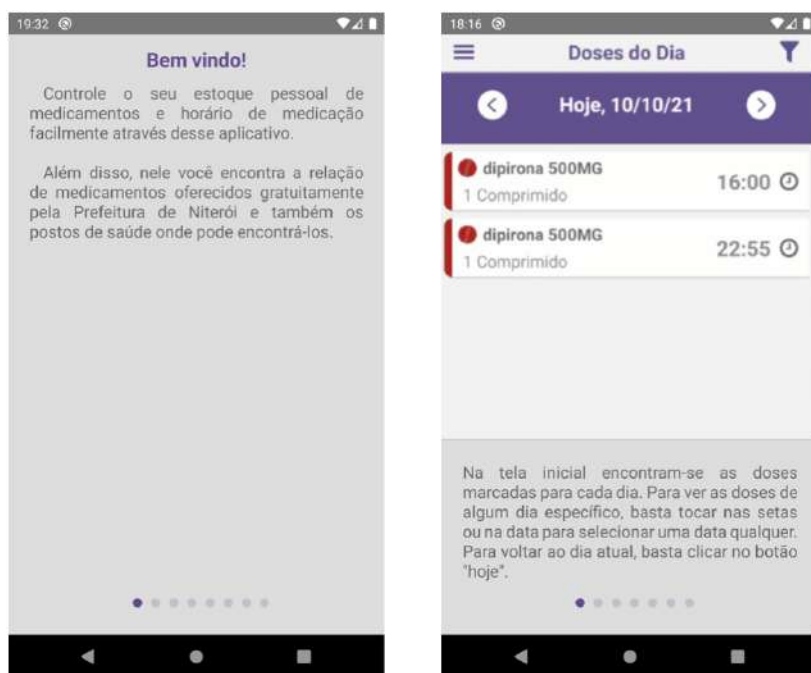


Figura 5: Telas de boas vindas (esquerda) e tutorial (direita).

Fonte: O autor

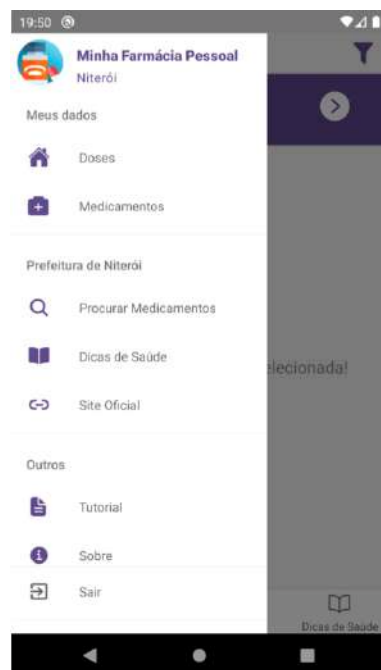


Figura 6: Menu drawer

Fonte: O autor

acordo com a necessidade em tela, sendo presente somente em telas em que há necessidade de filtrar algo), que permite o usuário escolher como suas doses serão exibidas, isto é, se exibirá todas as doses do dia, as não tomadas ou então apenas as já tomadas. Na parte de

baixo, encontra-se o navegador em abas, que permite o usuário rapidamente trocar entre telas de fluxo principal, conforme mencionado na Seção anterior. Por fim, o conteúdo da tela encontra-se no meio, preenchendo maior parte da tela.

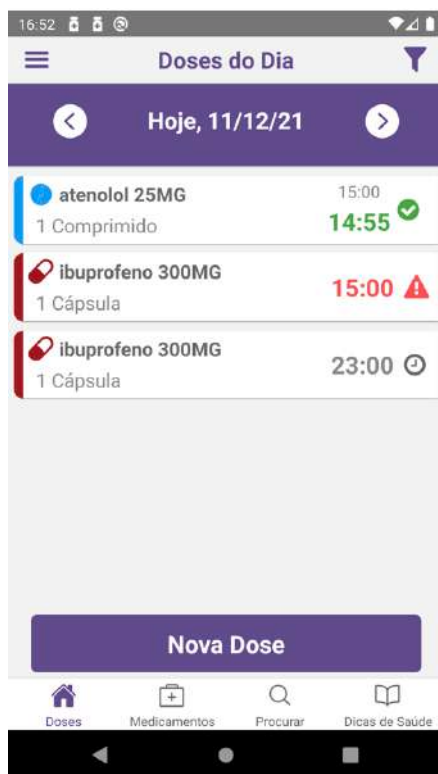


Figura 7: Tela de doses do dia

Fonte: O autor

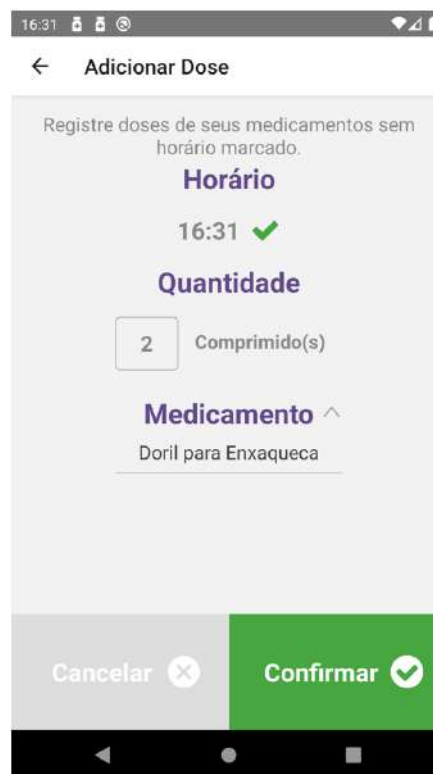


Figura 8: Tela de nova dose

Fonte: O autor

Essa tela apresenta um componente para selecionar o dia de doses a ser exibido. Este componente exibe o dia selecionado, permite o usuário a escolher uma data específica, tocar nos botões de seta para aumentar ou diminuir em um dia da data selecionada ou rapidamente retornar ao dia atual tocando no botão “hoje”. Abaixo disso, uma lista de doses de medicamentos cadastrados é exibida, informando ao usuário o nome do medicamento, com a cor e ícone selecionados durante o cadastro, a quantidade e horário a ser tomado. Quando o horário de tomar uma dose é passado e a dose não foi tomada, haverá uma atualização em tela indicando o atraso da mesma, com o texto do horário mudando para uma cor vermelha e um ícone de atenção, como um tipo de indicador de alerta para o usuário. Estes elementos podem ser observados na Figura 7. Ao tocar em um registro da lista, uma modal é exibida apresentando informações sobre esta dose, e o usuário pode confirmar a dose como tomada ou então alterar a quantidade ou horário da dose. Ao marcar uma dose como tomada, seu registro na lista será atualizado, bem como o estoque pessoal inicialmente cadastrado pelo usuário.

A construção desta tela foi pensada com objetivo de mostrar ao usuário as informações mais importantes para ele naquele momento. Consolidar todas as doses que ele tem que tomar em certo horário, que é parte do objetivo principal do aplicativo, de maneira organizada e com uma interface amigável para facilitar seu uso. Os ícones coloridos selecionados pelo usuário no cadastro do medicamento, permitindo uma fácil associação entre imagem e o medicamento, o horário em destaque no canto direito, e cada registro ordenado pelo horário da dose são elementos que vão proporcionar ao usuário menor dificuldade em realizar a tarefa básica de atualizar seus medicamentos.

Pensando em possíveis cenários reais para os usuários, no geral, o momento de tomar um medicamento, mesmo que agendado, não deveria ser uma tarefa demorada e que interrompa o usuário de fazer o que estava fazendo antes. Portanto, com uma simples notificação alertando-o do horário do medicamento, basta abrir o aplicativo, localizar o medicamento facilmente na lista, tocar para confirmar a dose e sua interação com o aplicativo está encerrada naquele momento. Um dos conceitos básicos de experiência do usuário diz respeito a aumentar eficiência e transmissão de dados, diminuindo então o tempo de interação e erros cometido pelos usuário (NIELSEN, 1994). Estes conceitos também podem ser observados na tela de registro de nova dose.

Se o usuário tiver cadastrado um medicamento do tipo eventual, ou seja, um medicamento cujas doses não sejam agendadas, o botão para adicionar doses é exibido, e quando tocado redireciona o usuário para uma tela de registro de dose eventual, ilustrada na Figura 8, onde seleciona o medicamento dentre os medicamentos de tipo eventual já cadastrados, horário e quantidade tomadas. Se o usuário preencher com um horário futuro, uma notificação será agendada para informá-lo no horário correto. Esta interação, assim como a para doses agendados, é objetiva e fácil, sempre buscando minimizar o tempo necessário para o usuário completar a tarefa. O fluxo de cadastro é resumido no diagrama de atividades apresentado na Figura 3.

De um ponto de vista técnico, pode-se fazer menção à biblioteca AsyncStorage, que buscará as configurações de filtro e todos os medicamentos cadastrados do usuário persistidos no armazenamento do dispositivo assim que o usuário abre o aplicativo ou a tela entra em foco. A abordagem de fazer a busca dos objetos persistidos toda vez que a tela entra em foco não é fundamentalmente eficiente ou rápida, mas garante que os dados sejam sempre carregados corretamente. Apesar disso, por se tratar de uma quantidade pequenas de dados, essa perda de performance é imperceptível para o usuário.

Uma abordagem mais comum utilizada junto ao React é com uso de contexto (con-

text), que permite o compartilhamento de dados entre todos os componentes da aplicação. Este método não foi implementado na versão atual da aplicação por ser um método um pouco mais complexo que precisaria de uma mudança na maneira em que a aplicação foi implementada, mas é um ponto que pode ser desenvolvido no futuro para melhor assegurar a manutenibilidade da aplicação.

4.2.3 Tela Medicamentos

A Figura 9 ilustra as telas de lista e detalhes de medicamentos. A primeira apresenta ao usuário uma lista contendo todos os seus medicamentos cadastrados. Assim como na tela principal, um filtro permite o usuário alternar a exibição entre todos os medicamentos ou apenas medicamentos cujo tratamento ainda não foi encerrado. Já na lista, cada registro faz referência a um medicamento, novamente com cor e ícone selecionados durante o cadastro, para facilitar a distinção entre medicamentos para o usuário, a quantidade atual no estoque pessoal do usuário e uma informação sobre o tratamento: se for um medicamento com doses agendadas, informa a quantidade de doses restantes, caso contrário, informa que é um medicamento a ser “tomado quando necessário”.

Um botão similar ao da tela principal é exibido nesta tela, dessa vez permitindo o usuário cadastrar um novo medicamento. Em todo o aplicativo, houve uma tentativa de manter padrões de comportamento entre telas e componentes, para trazer uma melhor facilidade de uso ao usuário. Exemplos destes padrões são as posições e aparência dos botões de cadastro de dose e o de medicamento ocupando a parte inferior da tela, ou também o filtro destes em cada tela, localizado no canto superior direito. Botões de ação confirmatória possuem cores verdes, enquanto botões de cancelar ou fluxo alternativo possuem cor cinza, de menor destaque.

Ao tocar em um registro de medicamento, a tela de detalhes é carregada, trazendo informações adicionais sobre o medicamento selecionado. Nesta tela, o usuário pode atualizar o seu estoque em casos de reposição, ou então marcar um tratamento de medicamento como finalizado. Esta ação irá apagar todas as doses futuras restantes do medicamento e cancelar as notificações agendadas.

Tocando no botão para adicionar medicamentos, o usuário é redirecionado para a tela de cadastros de medicamentos, onde inicialmente é exibido um campo de texto preenchível pedindo o nome do medicamento, que deve ser preenchido antes de poder prosseguir com o resto do cadastro. Ao começar a preencher o nome de um medicamento, o aplicativo fará uma consulta ao serviço externo para trazer ao usuário sugestões de medicamentos

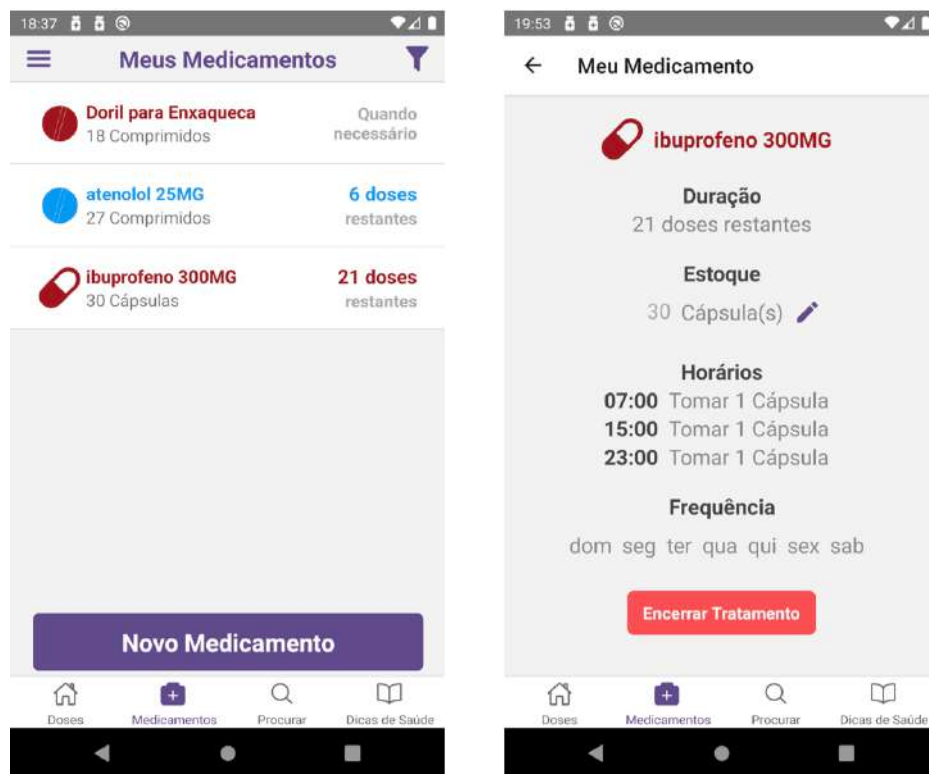


Figura 9: Telas de medicamentos (esquerda) e detalhe de medicamento (direita)

Fonte: O autor

com nomes similares, para facilitar o usuário no preenchimento. Esta relação de medicamentos foi obtida no site oficial da Anvisa ([GOVERNO FEDERAL, 2020](#)). Com o nome preenchido, outros campos do formulário são exibidos, e o usuário deve preencher informações como estoque, apresentação do medicamento (comprimido, cápsulas ou líquido), cronograma de tratamento, isto é, doses marcadas ou não, ícone e cor, além de um campo de observação para informações adicionais.

O cronograma de tratamento é composto por um botão do tipo *switch*, ou interruptor, que alterna entre valores verdadeiro e falso. Quando “desligado”, o usuário está informando que o medicamento deve ser tomado apenas quando necessário, mas quando ele ativa o botão, ele alterna para o modo de doses marcadas, e com isso, novos campos são exibidos para preenchimento. Primeiramente, um botão informando a recorrência do medicamento durante o dia, com as opções variando de uma a quatro vezes ao dia. De acordo com a opção selecionada, uma lista de horários é exibida com uma quantidade de marcações referente a seleção de recorrência. Para cada horário, o usuário pode alterar a hora e quantidade da dose do medicamento que deve ser tomada, com uma restrição impedindo ele registrar mais de uma dose do mesmo medicamento no mesmo horário.

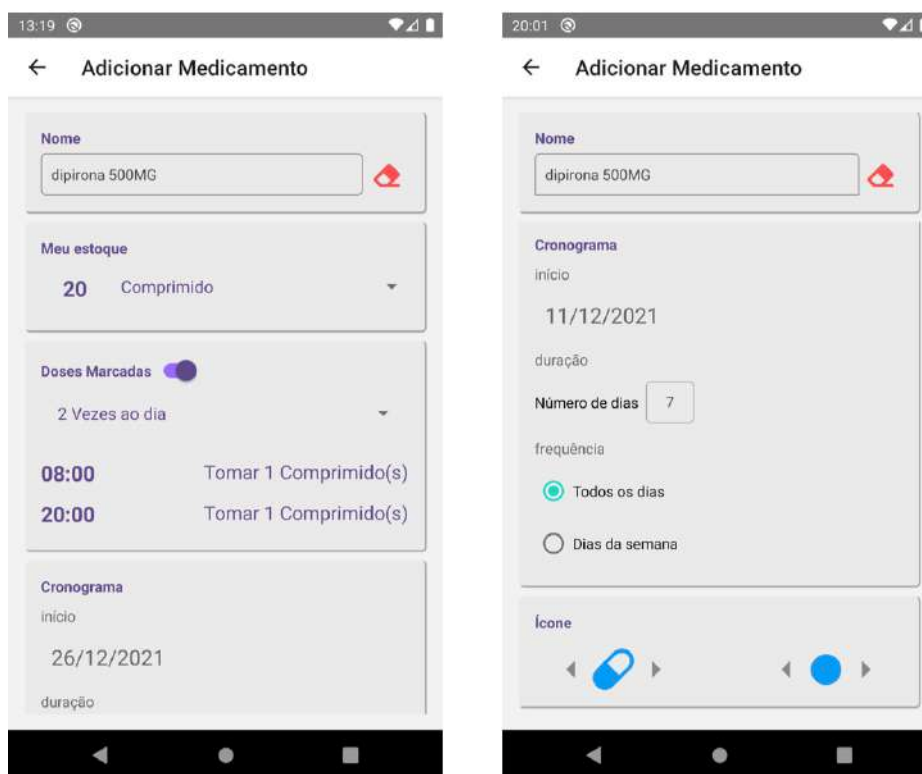


Figura 10: Tela de cadastro de novo medicamento

Fonte: O autor

Além disso, um campo de seleção de data também é exibido, este campo define a data de início do tratamento. Em seguida, um campo numérico para informar a duração em dias do tratamento e por fim a frequência, onde o usuário pode selecionar o tratamento com frequência diária ou selecionar quais dias na semana ele deve tomar. A Figura 10 mostra o conteúdo dessa tela.

Com todos os campos preenchidos corretamente, ele toca no botão “Salvar”, o aplicativo verifica se houve alguma informação necessária não preenchida e caso contrário, segue para o fluxo de armazenamento. Caso o medicamento seja de doses marcadas, ele irá criar todas as doses necessárias de acordo com as informações preenchidas no cronograma do medicamento, e também, para cada uma, irá agendar uma notificação local. Este fluxo é observado no diagrama de atividades apresentado na Figura 2, da Seção 3.

É importante observar, neste caso, que o número de doses que o usuário tomará durante ao tratamento é igual ao produto entre o número de vezes no dia que tomará uma dose pelo número total de dias, entretanto, este produto não é necessariamente a quantidade de dias que o tratamento irá durar. Para isso, é necessário uma lógica que traduza a configuração de dias da semana e horários em um dia para as doses reais que o

usuário deve tomar. Caso não seja um tratamento diário, não devem ser criadas doses para estes dias, ou caso o usuário queira começar o tratamento no dia em que cadastrou um medicamento, pode ser que um ou mais horários de doses daquele dia já haviam passado, necessitando deslocar o horário dessas doses para um dia futuro. Uma vez que essa lógica for executada e todas estas informações criadas, o sistema armazena as informações do medicamento e doses no dispositivo utilizando o AsyncStorage, novamente.

4.2.4 Tela Procurar

O objetivo dessa tela é relacionar o usuário com as informações da prefeitura. Ela permite que o usuário consulte a relação atual de medicamentos oferecidos na REMUME e as policlínicas da cidade que atualmente disponibilizam tais medicamentos de maneira gratuita ao cidadãos. Ela possui dois botões que redirecionam o usuário tanto para uma tela de medicamentos da REMUME ou para a lista das policlínicas.



Figura 11: Tela procurar

Fonte: O autor

A tela constitui em dois botões com um texto explicativo para cada um deles. O botão superior redirecionará o usuário para a tela dos medicamentos da REMUME, enquanto o outro redirecionará para tela de policlínicas cadastradas, conforme a Figura 11. Na tela de medicamentos oferecidos, ilustrada na Figura 12, o usuário é apresentado com um campo de texto para filtrar por nome, e, tocando em um registro da lista de medicamentos é

redirecionado para uma tela com mais informações sobre o medicamento. Nesta tela de detalhes, um botão permite que o usuário registre o início de um tratamento com este medicamento, redirecionando-o para tela de cadastro de medicamentos com o campo do nome do medicamento já preenchido. Para esta tela, está prevista a funcionalidade para o usuário reportar a tentativa de baixa do medicamento em uma policlínica, dado que poderá ser consultado pela Prefeitura de Niterói. Esta função ficou fora do escopo do produto mínimo viável inicial mas está prevista para uma versão futura do aplicativo, uma vez que tratará de um dos problemas abordado nas seções anteriores.

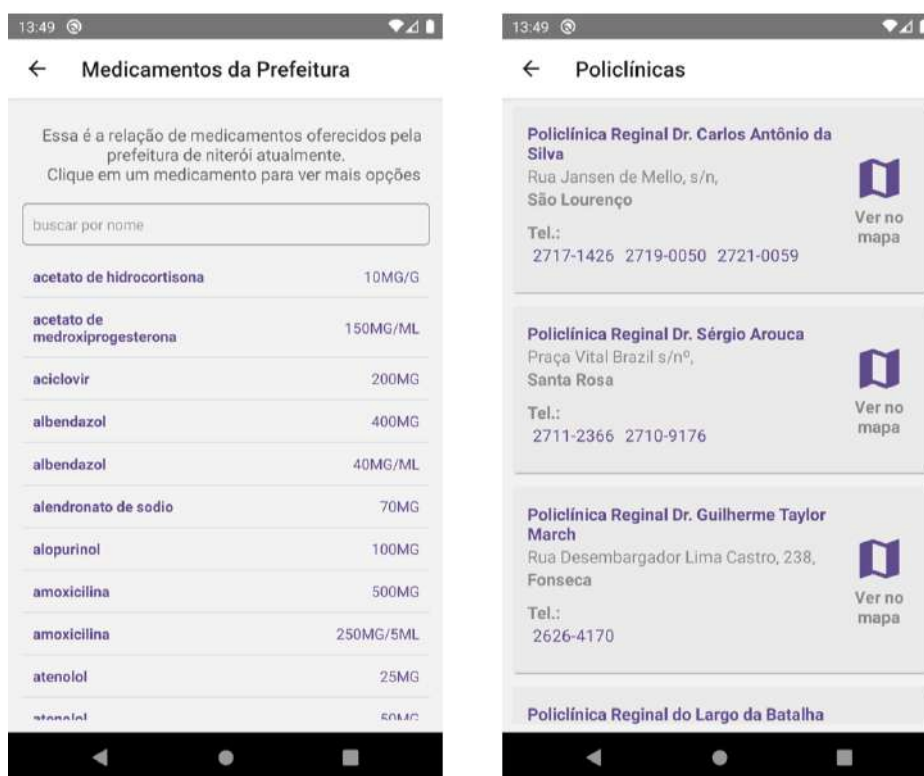


Figura 12: Telas de medicamentos da REMUME (esquerda) e policlinicas (direita)

Fonte: O autor

Já na tela de policlinicas, uma lista das atuais policlinicas com os medicamentos é exibida contendo nome, endereço, números de telefone e um botão com rótulo “ver no mapa”, também na Figura 12. Tocando em um dos números, o aplicativo de chamadas é aberto com o número preenchido para que o usuário possa ligar facilmente para a unidade de saúde, enquanto tocar no botão do mapa redireciona o usuário para um aplicativo externo de mapas com o posto de saúde indicado, permitindo ver o melhor caminho até lá.

Durante a concepção dos requisitos do sistema, houve discussões a respeito sobre quais medicamentos presentes na REMUME eram oferecidos em tais unidades de saúde. Este

relacionamento exigiria uma complexidade adicional para mostrar ao usuário a relação de cada medicamento oferecido pelas policlínicas, mas foi informado à equipe de desenvolvimento deste projeto que tal relacionamento não existe, e que as policlínicas podem oferecer todos os medicamentos da REMUME. Além disso, não é oferecido nenhum tipo de API² de consulta que permita buscar informações sobre os medicamentos atualmente em estoque em cada unidade de saúde.

Por conta disto, não é possível informar ao usuário exatamente quais medicamentos ele conseguiria encontrar em uma determinada unidade de saúde, o que pode resultar nas já existentes tentativas de retirada de medicamentos em falta, com o cidadão tendo feito a viagem em vão, além de possivelmente atrasar seu tratamento. Para mitigar este problema, está previsto a inclusão de um mecanismo onde o usuário informaria, pelo aplicativo, se obteve sucesso ou não em uma tentativa de retirada de medicamento, incluindo um motivo nos casos de falha, e essa informação poderia então ser extraída pela prefeitura para que melhor gerenciasse sua distribuição de medicamentos pela região.

4.2.5 Tela Dicas de Saúde

A Figura 13 apresenta a última tela presente no navegador de abas, que apresenta algumas dicas e recomendações de saúde para boas práticas dos usuários sobre armazenamento e consumo de medicamentos em casa. Essas recomendações foram levantadas por professores da área que fazem parte da equipe de desenvolvimento e servem como um lembrete ao usuário que deve tratar com responsabilidade seus medicamentos na sua farmácia pessoal.

Apesar de ser uma tela simples no momento de escrita desta monografia, é uma tela de absoluta importância, uma vez que ela relembra o usuário cuidados que ele deve ter a respeito de medicamentos. Conforme mencionado na Seção 1, o acesso a medicamentos no Brasil é de certa forma banalizado, e até mesmo medicamentos que deveriam precisar de prescrição médica podem ser obtidos sem apresentação da mesma, em certas farmácias, e por isso é importante trazer mais conscientização sobre o consumo de medicamentos das pessoas, que muitas vezes sequer necessitariam do fármaco.

Ao longo do tempo, esta tela poderá apresentar cada vez mais informações e se tornar um portal de informações de saúde, trazendo conteúdo que vá além também do âmbito de gestão de medicamentos pessoais. Com isso, pode-se afirmar que esta tela tem grande potencial para trazer uma genuína mudança e conscientização para os cidadãos.

²Do inglês “Application Programming Interface”, uma API permite que um sistema ou serviço se comunique com outro.

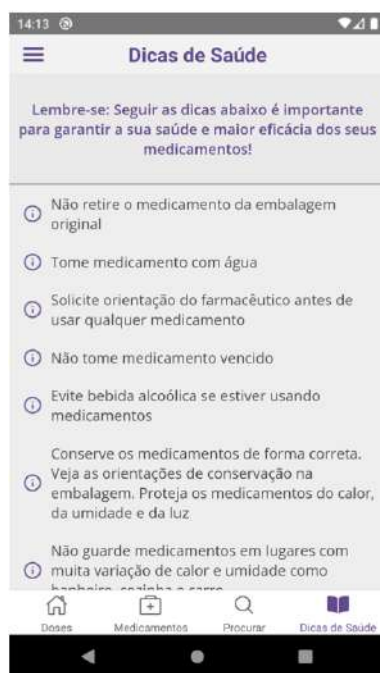


Figura 13: Tela de dicas de saúde

Fonte: O autor

4.3 Serviço Backend

Dentro do escopo de MVP do aplicativo, mencionado na Seção 1, os dados gerados pelo usuário dentro do aplicativo são somente os referentes aos medicamentos cadastrados por ele. Esses dados são persistidos localmente no próprio dispositivo do usuário. Já as informações dos medicamentos e unidades de saúde da prefeitura constituem tipos de dados que não devem ficar persistidos no dispositivos porque são dados que podem ser alterados sem controle do usuário, e por isso, precisam estar armazenados em um serviço online. Conforme mencionado na Seção 4.1, foi desenvolvido um serviço web em Java utilizando o framework Spring Boot, com o banco de dados de escolha sendo o MongoDB.

Antes de tudo, dentro do escopo mencionado existem três entidades tratadas no serviço, que são: policlínica, medicamento de Niterói e medicamento da Anvisa. Estes são os três tipos de dados externos que o usuário deseja ter acesso enquanto utiliza seu aplicativo. Com isso, tudo mencionado nessa seção levará em consideração a existência dessas três entidades.

A arquitetura do serviço web pode ser dividida em três camadas: repositórios, serviços e controladores. Os repositórios são como um intermédio entre o banco de dados e a camada de serviço da aplicação, e eles definem os métodos de busca e escrita de dados

no banco. Novamente, o Spring poupa a necessidade de escrever código excessivo através de sua estrutura, e a implementação de uma classe de interface Java representando um repositório cuidará dos métodos CRUD (do inglês *Create, Read, Update, Delete*) básicos e permite também a criação de métodos de consulta de dados sem necessidade de escrever consultas em SQL. Foi criada uma classe de interface de repositório para cada uma das entidades mencionadas.

A camada seguinte é a de serviços, camada intermediária que liga os repositórios dos controladores. Esta camada contém, tradicionalmente, a lógica de negócio da aplicação. Como o serviço apresenta inicialmente apenas métodos de consulta, a camada de serviço foi simplificada a fazer uma chamada direta aos repositórios, mas a implementação do serviço seguindo esta estrutura trará manutenibilidade para o futuro. Por fim, a camada de controladores é a camada que o aplicativo tem visão e utiliza como meio de comunicação com o serviço. Esta camada define os métodos de requisição HTTP e os caminhos que devem ser chamados para cada requisição, e assim como a camada de serviço, também foram feitas uma classe para cada entidade mencionada.

Como parte de escopo de desenvolvimento futuro, o serviço web disponibilizará de métodos que registrem os informes de tentativa de baixa de medicamento dos usuários, bem como fornecerá ao canal da Prefeitura métodos de consulta desses dados para que possam pensar em melhores estratégias de distribuição destes medicamentos, uma questão de interesse do órgão público mencionado na Seção 1.

5 Avaliação

A avaliação é um passo essencial no desenvolvimento de um sistema para melhor garantir uma aplicação com alta qualidade de uso. Ela auxilia na identificação de problemas de interface e interação que podem prejudicar a experiência do usuário durante seu uso (NIELSEN, 1994). Durante o desenvolvimento, certos pontos podem passar despercebidos tanto pelo desenvolvedor quanto para o resto da equipe do projeto que acompanha o desenvolvimento. Por esta razão, uma avaliação com pessoas externas à equipe de desenvolvimentos é imprescindível para garantia de uma melhor qualidade do produto final.

Existem diferentes maneiras de se avaliar um sistema, e foram escolhidos dos métodos de avaliação de interface humano-computador, ou IHC, para o sistema: Avaliação Heurística e Teste de Usabilidade. Essa Seção discute estas duas avaliações, seus resultados e o impacto delas para o aplicativo.

5.1 Avaliação Heurística

Nielsen e Molich (1990) mencionam a existência de quatro maneiras de se avaliar um sistema: formalmente através de análises técnicas, automaticamente por procedimentos computadorizados, empiricamente por experimentos ou heurísticamente, olhando para interface e julgando de acordo com certas regras e opiniões. Na prática, a maioria das interfaces são avaliadas de maneira heurística, pelos avaliadores, mas por conta da necessidade de certo conhecimento técnico na área, pode ser difícil fazer uma avaliação heurística correta de certas interfaces. Eles ainda concluem que avaliadores individuais apenas identificam até 51% dos problemas de interface que deveriam avaliar. Entretanto, agrupar estas avaliações individuais ocasionam em um resultado positivo, mesmo com um número pequeno de avaliadores.

Com isso em mente, foram selecionados alguns alunos dos cursos de computação para avaliar a interface, executando algumas atividades no aplicativo em seus smartphones.

Estes foram selecionados de acordo com a disponibilidade que tinham no período de avaliação. Como estes alunos atuaram como avaliadores iniciantes, sem um profundo conhecimento da área, foi desenvolvido um formulário contendo orientações, atividades planejadas e afirmações objetivas e abertas abrangendo as dez heurísticas de Nielsen, (NIELSEN, 1994), para guiá-los na avaliação, tentando reduzir o impacto da inexperiência em uma avaliação heurística formal, mas ainda assim conseguindo produzir resultados.

Os avaliadores foram orientados sobre como preencher o formulário, e receberam algumas sugestões de atividades para realizarem dentro do aplicativo, como cadastro de um medicamento, marcar uma dose como tomada e buscar os medicamentos. Estas orientações e afirmações estão disponíveis no Apêndice A. As Tabelas 3 a 12 descrevem a divisão de heurísticas e afirmações para a avaliação. É importante observar que certas heurísticas podem acabar tendo um peso maior dependendo do tipo de sistema que foi desenvolvido e de quem será o usuário final.

Tabela 3: Heurística: I. Visibilidade do estado do sistema

| Item | Afirmação |
|------|---|
| 1 | O usuário é informado sobre o que está acontecendo no sistema por meio de feedback e exibição das informações de forma clara, concisa e apropriada (Por exemplo, depois de o usuário cadastrar um medicamento o usuário recebe alguma confirmação de que o medicamento foi adicionado). |
| 2 | Componentes selecionados são destacados, enquanto componentes desabilitados são “acinzentados” ou omitidos (Por exemplo, botões, texto selecionado, ícones, caixa de seleção, menus, etc.) |
| 3 | Informação crítica e contextual, como a gestão das doses marcadas, por exemplo, são priorizadas e visíveis. |

Fonte: O autor

Tabela 4: Heurística: II. Correspondência entre o sistema e o mundo real

| Item | Afirmação |
|------|---|
| 4 | Os rótulos dos controles são consistentes com as suas ações (Por exemplo, o rótulo “Procurar” leva para a tela esperada) |
| 5 | A informação aparece em uma ordem lógica e natural. |
| 6 | As opções e rótulos de menu podem ser compreendidos rapidamente. |
| 7 | Ícones, imagens e outros componentes de interface são concretos e familiares. |
| 8 | As cores usadas correspondem as expectativas comuns sobre o código de cores. (Por exemplo, a cor vermelha é usada para indicar a prioridade máxima do atendimento). |

Fonte: O autor

Tabela 5: Heurística: III. Controle e liberdade do usuário

| Item | Afirmação |
|------|--|
| 9 | Os usuários podem avançar e retroceder na navegação do sistema. |
| 10 | Os usuários podem salvar tarefas no meio da execução para continuar futuramente (Por exemplo, salvar um formulário parcialmente preenchido). |

Fonte: O autor

Tabela 6: Heurística: IV. Consistência e padronização

| Item | Afirmação |
|------|--|
| 11 | Os elementos da interface seguem uma terminologia padrão. |
| 12 | A navegação do sistema e o layout das telas são consistentes (Por exemplo, menus, botões de confirmação e áreas de texto principais sempre aparecem na mesma posição nas telas). |
| 13 | O estilo da interface é consistente nas telas do sistema (Por exemplo, o mesmo conjunto de cores e formas é usado em todas as telas). |

Fonte: O autor

Tabela 7: Heurística: V. Reconhecimento em vez de memorização

| Item | Afirmação |
|------|--|
| 14 | Todas as informações necessárias para o usuário realizar as tarefas são visíveis e/ou fáceis de encontrar (Por exemplo, quando o usuário está preenchendo o cadastro de medicamento ele consegue visualizar as informações dele na mesma tela e não precisa memorizá-las). |
| 15 | O sistema fornece todas as informações necessárias (Por exemplo, quando o usuário está preenchendo algum formulário o sistema provê todos os dados que ele precisa). |
| 16 | Os menus são balanceados, nem muito profundos nem largos demais (Por exemplo, o menu não tem muitos itens ou muitos níveis). |

Fonte: O autor

Apesar de todas serem pertinentes, algumas das heurísticas implementadas acabam sendo mais evidentes que outras. Por exemplo, a última heurística, sobre ajuda e documentação, se aplica ao sistema mais no quesito da tela de tutorial presente na introdução do aplicativo, enquanto as heurísticas mencionadas nas Tabelas 6 e 9 podem ser observadas em todas as telas do aplicativo.

O processo de avaliação por inspeção do uso da interface por parte dos avaliadores durou uma semana e depois cada avaliador preencheu seu formulário. Cada pergunta tinha como opção de respostas valores em um intervalo de 0 a 4, com o primeiro representando

Tabela 8: Heurística: VI. Flexibilidade e eficiência do uso

| Item | Afirmação |
|------|--|
| 17 | As principais funcionalidades do sistema são fáceis de acessar. (Por exemplo, as funcionalidades principais podem ser alcançadas com poucos cliques) |
| 18 | Funcionalidades e tópicos relacionados estão próximos uns dos outros. |
| 19 | A entrada de dados e a navegação são mínimas o suficiente. |
| 20 | A necessidade de mudar de tela é evitada (Por exemplo, todo o conteúdo exibido cabe em uma tela). |

Fonte: O autor

Tabela 9: Heurística: VII. Projeto estético e minimalista

| Item | Afirmação |
|------|---|
| 21 | O design visual é atraente. |
| 22 | As telas têm um design “clean” apresentando somente informações e componentes importantes. |
| 23 | O conteúdo da tela é sempre visível completamente e não coberto por outros componentes de interface |
| 24 | O uso do espaço disponível da tela é adequado |

Fonte: O autor

Tabela 10: Heurística: VIII. Prevenção de erros

| Item | Afirmação |
|------|--|
| 25 | Os campos do formulário aceitam apenas o tipo de caractere correspondente (Por exemplo, o campo quantidade aceita somente valores inteiros). |
| 26 | As opções do menu são lógicas e compreensíveis. |

Fonte: O autor

Tabela 11: Heurística: IX. Suporte para reconhecer, diagnosticar e reparar erros

| Item | Afirmação |
|------|--|
| 27 | O sistema oferece revisão das informações para evitar possíveis erros antes da conclusão do atendimento. |
| 28 | Os botões têm uma distância adequada entre si para evitar que o usuário pressione o botão errado. |

Fonte: O autor

Tabela 12: Heurística: X. Ajuda e documentação

| Item | Afirmação |
|------|---|
| 29 | O sistema oferece suporte para eventuais dúvidas ou maiores esclarecimentos |

Fonte: O autor

nível de severidade 0, ou seja, não apresentou problemas, e o último severidade 4, necessita correção imediata. Ainda mais, uma opção “não se aplica”, quando o item avaliado em questão não se aplicava ao contexto do sistema.

Além das perguntas apresentadas nas tabelas, campos discursivos também foram disponibilizados para cada uma das heurísticas para que cada avaliador pudesse adicionar comentários pertinentes com sugestões ou exemplificando os pontos com necessidade de melhoria imediata. Alguns destes comentários são abordados na Subseção 5.3.1.

5.2 Avaliação Subjetiva de Usabilidade

Um teste de usabilidade visa estimar a usabilidade de um sistema de acordo com as experiências do seu público-alvo. Diferentes objetivos da avaliação determinarão critérios de usabilidades diferentes que devem ser medidos. Em um teste de usabilidade, os usuários avaliadores realizam o teste realizando certas tarefas, em ambiente controlado, e durante essas seções dados são coletados sobre o desempenho dos participantes (BARBOSA et al., 2021). Entretanto, em razão da pandemia causada pelo Covid-19 e pelo tempo disponível para o projeto, não foi possível realizar um teste de usabilidade desta maneira.

Para contornar estas limitações, foi adotado um método de avaliação subjetiva desenvolvido por John Brooke, e que é um dos métodos de avaliação de usabilidade mais populares, que quantifica a usabilidade baseado em algumas afirmações subjetivas sobre a experiência do usuário interagindo com o sistema (BROOKE, 1995). Este é o método SUS (do Inglês “*System Usability Scale*”), que apesar de não reproduzir todas as etapas esperadas de um teste de usabilidade formal, atinge resultados satisfatórios e consegue inferir se um sistema possui boa usabilidade ou não.

A avaliação consiste em dez perguntas objetivas, em que o usuário deve responder em uma escala de 1 a 5, com os pesos variando entre “Discordo completamente” para o 1 e “Concordo completamente” para 5. As dez perguntas são oferecidas em certa ordem de modo que as perguntas de índice ímpar possuem afirmações de conotação positiva sobre o sistema, enquanto perguntas de índice par apresentam afirmações de conotação negativa. A Tabela 13 descreve as perguntas apresentadas no formulário de avaliação. As orientações escritas na avaliação heurística também foram inseridas neste formulário.

O período de avaliação e preenchimento do formulário foi de 10/12/2021 a 20/12/2021 e contou com a participação de 32 pessoas, alunos do curso de farmácia. Estes alunos naturalmente possuem um conhecimento maior na área de saúde, mas dentre os parti-

participantes da avaliação encontravam-se aqueles com maior e menor conhecimento técnico, que poderiam ter nenhuma ou muitas dificuldades ao utilizar o aplicativo.

Como o aplicativo foi desenvolvido tendo em mente, primeiramente, apenas o sistema operacional Android, algumas pessoas não poderiam participar ativamente do teste. Para contornar esta situação, foi gravado um vídeo apresentação que passa por todos os fluxos do aplicativo, explicando cada funcionalidade. Apesar de não ser a solução ideal e substituir o uso verdadeiro do aplicativo, o vídeo contribuiu para um maior engajamento de pessoas. Ao todo, dezenove destas pessoas não puderam baixar e testar o aplicativo e utilizaram somente o vídeo como critério de avaliação. Este vídeo também serviu como uma orientação a mais para os usuários, uma vez que todos os fluxos principais foram percorridos na apresentação.

Por fim, um outro fator que atrapalhou na coleta de dados de usuários que melhor representariam usuários finais foi a dificuldade em acesso ao aplicativo. Por conta da política de testes fechados da Google na Play Store, o acesso a aplicativos para testes internos é mais complexo, tendo a necessidade de incluir os avaliadores em uma lista de e-mail específica para que estes então pudessem baixar o aplicativo. Essas etapas a mais dificultariam o teste do aplicativo para pessoas mais leigas e de maior idade, que não conseguiriam o acesso ao aplicativo em uma única etapa. Apesar destes contratempos, os resultados obtidos foram promissores sobre a usabilidade do sistema.

Tabela 13: Avaliação subjetiva de usabilidade - Perguntas do método SUS

| Item | Afirmação |
|------|---|
| 1 | Eu acho que gostaria de usar esse sistema com frequência. |
| 2 | Eu acho o sistema desnecessariamente complexo. |
| 3 | Eu achei o sistema fácil de usar. |
| 4 | Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o sistema. |
| 5 | Eu acho que as várias funções do sistema estão muito bem integradas. |
| 6 | Eu acho que o sistema apresenta muita inconsistência. |
| 7 | Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente. |
| 8 | Eu achei o sistema atrapalhado de usar. |
| 9 | Eu me senti confiante ao usar o sistema. |
| 10 | Eu precisei aprender várias coisas novas antes de conseguir usar o sistema. |

Fonte: [BROOKE \(1995\)](#) (Adaptado)

5.3 Resultados

Nesta seção são apresentados os resultados obtidos pelas avaliações discutidas neste capítulo. A Subseção 5.3.1 discute os resultados da avaliação heurística, em que também são mencionados pontos de melhorias observados durante a avaliação que foram prontamente desenvolvidos e apropriados ao aplicativo, enquanto a Subseção 5.3.2 trata os resultados obtidos na avaliação subjetiva de usabilidade.

5.3.1 Avaliação Heurística

O processo de avaliação durou uma semana, entre os dias 13/09/2021 e 20/09/2021, e contou com a participação de cinco avaliadores, de sete convidados. Durante este período, estes avaliadores tiveram acesso a uma mesma versão do aplicativo, que já passou por diversas mudanças desde então.

No critério “I. Visibilidade do estado do sistema”, os níveis de severidade ficaram em sua maioria entre 0 e 1, indicando nível de aceitação favorável, com um usuário mencionando severidade 3 no item 1, descrito na Tabela 3. Já na heurística “II. Correspondência entre o sistema e o mundo real”, o item 7, a respeito da familiaridade entre os componentes de interface, recebeu mais indicativos de severidade média e alta, com 60% das respostas indicando uma severidade 2 ou maior. Este foi um dos principais pontos de melhoria que foi levado para a segunda fase de desenvolvimento, mencionada na Subseção 4.2, onde foi priorizada uma padronização maior de componentes e também maior familiaridade entre os mesmos.

Outros locais de atenção que merecem destaque são os itens 10, 12, 13, 14 e 29, em que níveis de severidade 3 ou 4 foram mencionados. O primeiro destacou a falta de um mecanismo de recuperação no cadastro de medicamento em caso de falha do dispositivo no meio do processo. Essa questão foi levantada com a equipe do projeto que julgou não se aplicar no contexto do aplicativo. A equipe concluiu que é importante o usuário preencher os dados de um novo medicamento de uma só vez para melhor garantir que todos os dados tenham sido preenchidos corretamente, e interromper o processo para continuar em outra hora pode afetar a acurácia destas informações. Já os pontos 12 e 13 criticam a consistência e padronização de elementos e telas, principalmente um número de botões com aparências distintas pelo aplicativo. O item 14, da heurística de “reconhecimento em vez de memorização” trouxe comentários pertinentes a alguns ícones cuja funcionalidade não era de fácil percepção ou alguns elementos visuais com tamanho pequenos, trazendo

dificuldade para o usuário em seguir com a ação.

Por fim, o item 29, da heurística “ajuda e documentação” teve o feedback com maiores níveis de severidade da avaliação, com o principal comentário sendo sobre a visualização do tutorial, que na versão avaliada, só aparecia uma vez para o usuário, quando ele acessava o aplicativo pela primeira vez. Este foi outro ponto cuja correção foi feita prontamente, e agora nas versões posteriores à avaliação do aplicativo o usuário consegue acessar a tela de tutorial quando desejar. Além disso, foram feitos uns ajustes no tutorial trazendo informações melhores explicadas e separadas. Esta melhoria é apresentada na Subseção [4.2.1](#).

Além dos itens pertinentes às heurísticas de Nielsen, os avaliadores também apontaram alguns defeitos que identificaram durante o uso do aplicativo, que foram corrigidos para a versão seguinte. O formulário para a avaliação e as respostas objetivas e discursivas estão disponíveis no Apêndice [A](#).

5.3.2 Avaliação Subjetiva de Usabilidade

Para atribuir uma nota para essa avaliação, é necessário realizar um cálculo de acordo com a resposta de cada um, se atentando para a paridade do índice da afirmação. Segundo Texeira, a média limite de pontuação para um sistema com boa usabilidade é de 68 pontos. Isso quer dizer que um sistema com pontuação inferior a este valor apresenta problemas de usabilidade ([TEXEIRA, 2015](#)).

Com os resultados obtidos da avaliação, todas as pontuações individuais do System Usability Score ficaram acima deste limite, com o menor resultado pontuando 70. A média obtida entre todos os participantes foi de 89, uma pontuação bem acima do limite estipulado, sugerindo que tanto a primeira etapa de desenvolvimento seguindo uma metodologia de levantamento de requisitos e comparações com outros aplicativos do mercado, quanto a primeira etapa de avaliação com a avaliação heurística, podem ter contribuído significativamente na melhoria de usabilidade, com esta última apontando problemas que foram corrigidos antes de iniciar esta avaliação.

É importante acrescentar que os usuários que utilizaram e de fato conseguiram ter a experiência de interação com o sistema tiveram uma noção melhor da usabilidade do que aqueles que se basearam apenas no vídeo gravado por um usuário experiente no sistema. Por conta disso, foram analisadas também as pontuações obtidas pelo grupo que utilizou o aplicativo e o que apenas se baseou no vídeo, para se ter ideia de quão divergente suas

respostas seriam.

O diagrama de caixa apresentado na Figura 14 mostra esta distribuição de nota obtida pela avaliação dos usuários, que utilizaram ou não o aplicativo. Neste diagrama é possível notar que quem utilizou o aplicativo teve uma divergência maior de opiniões, mesmo sendo um número menor de pessoas. Ainda assim, a pontuação média de usabilidade obtida foi de aproximadamente 84,6 pontos, ainda acima da média considera boa. Dentre os que apenas assistiram o vídeo, a pontuação ficou em torno de 92,1, confirmando que um vídeo gravado por um usuário experiente pode causar uma impressão pouco diferente da realidade.

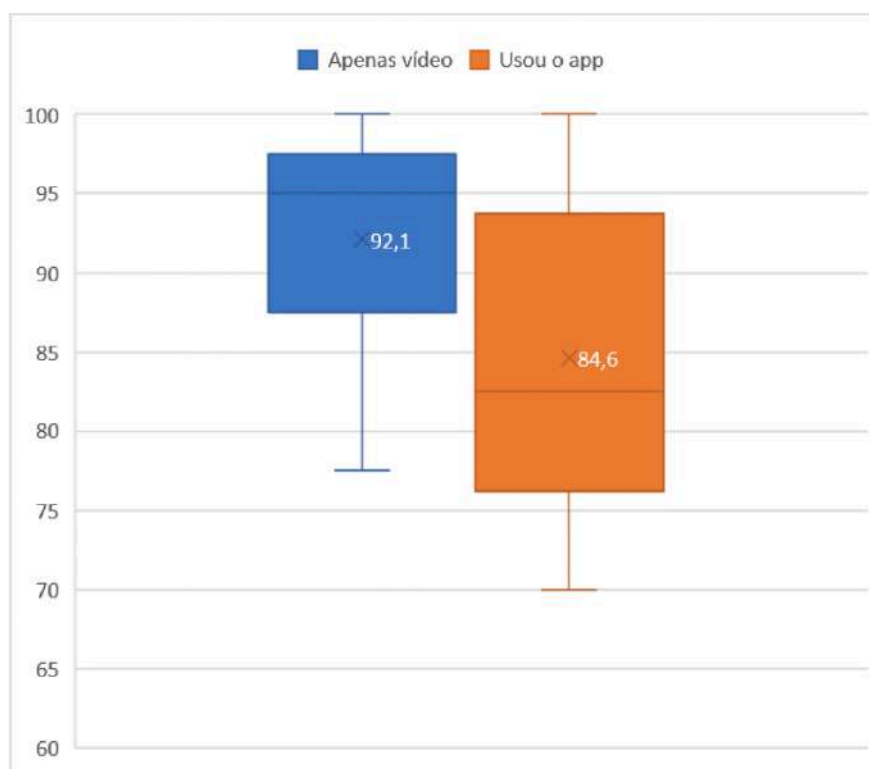


Figura 14: Distribuição de pontuações pelo método SUS

Fonte: O autor

A Figura 15 mostra a pontuação SUS obtida por cada usuário. Apesar de circunstâncias diferentes, os dois grupos de avaliação retornaram pontuações próximas, e isso pode sugerir que a utilização de um vídeo de apresentação, com certos ajustes, pode ser um forte substituto aos métodos de avaliação em situações onde não é possível disponibilizar o sistema aos usuários. Por fim, a Figura 16 ilustra a distribuição de respostas dos participantes para cada pergunta da avaliação. O formato do formulário está disponível no Apêndice B.

Além das perguntas objetivas, também foram deixados espaços de texto discursivo

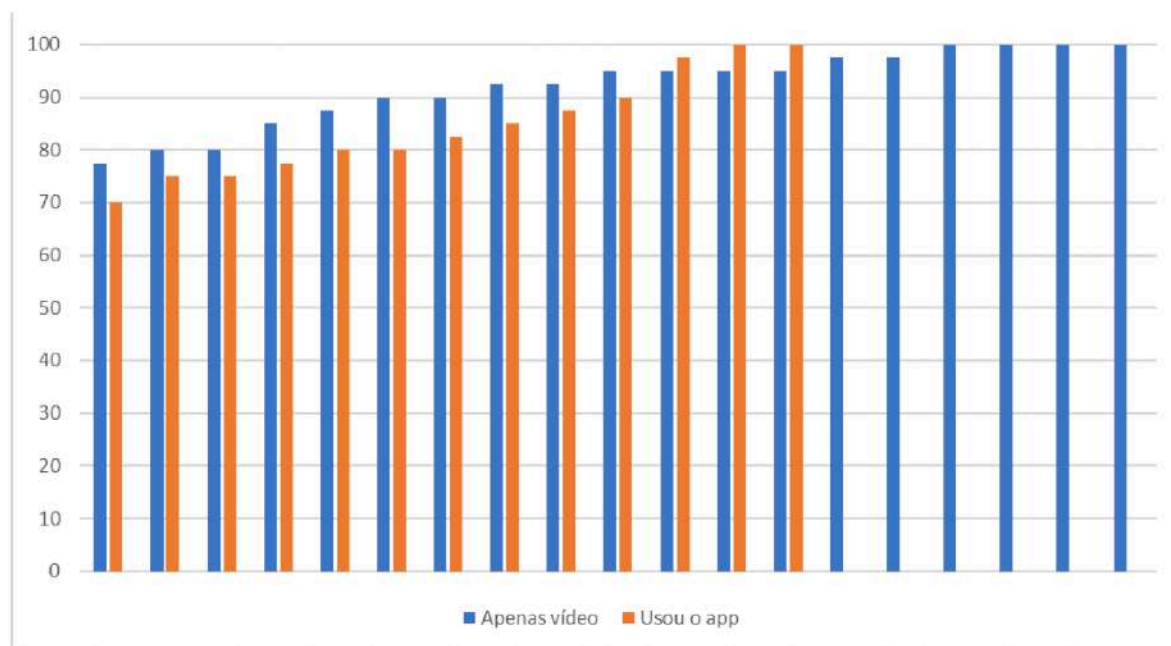


Figura 15: Pontuação SUS por participante da avaliação

Fonte: O autor

onde participantes podiam apontar pontos positivos que lhe chamaram atenção, assim como pontos de melhoria que julgaram ser melhor informar. Dentre os pontos positivos, houveram menções à funcionalidade de horário das doses, de que seria útil para o usuário no caso de precisar tomar múltiplos medicamentos em um dia. Um outro ponto elogiado foi o acesso à informação sobre os medicamentos oferecidos gratuitamente e as policlínicas que oferecem, com respectivos endereços e telefones. Por fim, houve um comentário apreciando a tela de dicas de saúde e um comentário mais amplo referente a usabilidade, em que o participante comentou que “apesar de parecer um sistema bem técnico, só ter um mínimo de conhecimento sobre medicamentos é o suficiente para usar”. Esse tipo de comentário ajuda a confirmar que as melhorias levantadas durante a avaliação heurística bem como a metodologia de desenvolvimento e levantamentos de requisitos em todo o projeto foram aplicadas corretamente.

Nos pontos de melhoria, uma crítica foi levantada a respeito do usuário não conseguir alterar mais informações sobre o medicamento cadastrado. Na versão disponibilizada para os testes, só é possível alterar a quantidade atual de estoque do medicamento, quando na verdade há interesse na possibilidade de alterar horário das doses ou frequência do tratamento. Esse tipo de melhoria já havia sido mencionado após a avaliação heurística e deve ser implementada no futuro. Uma outra sugestão menciona a ausência de serviço de consulta de bulário no aplicativo, e oferecer ao usuário informações sobre como o

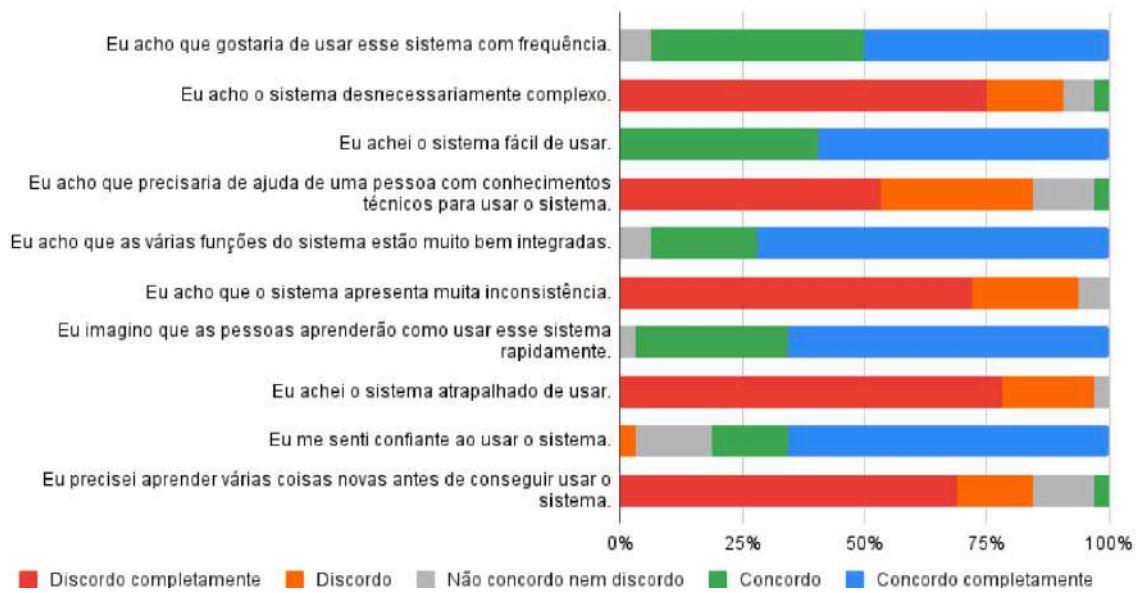


Figura 16: Distribuição de respostas de participantes pelo método SUS

Fonte: O autor

medicamento atua seria vantajoso para ele.

6 Conclusão

É de grande importância ter organização e maturidade quando o assunto é saúde. Ter controle dos medicamentos presentes em casa, saber armazená-los e consumi-los da maneira correta contribui na diminuição de riscos à saúde. Este trabalho teve como objetivo trazer uma plataforma em dispositivo móvel que permite ao usuário ter controle destes medicamentos, agendar seus tratamentos de maneira correta e, especialmente para cidadãos da cidade de Niterói, trazer informação a respeito dos medicamentos oferecidos gratuitamente pela política de assistência farmacêutica da prefeitura. Esse trabalho é o início de uma parceria entre a Universidade Federal Fluminense e a Prefeitura de Niterói através do Programa de Desenvolvimento de Projetos Aplicados, e buscará também trazer benefícios a longo prazo para a Prefeitura, que poderá coletar os dados fornecidos pelos cidadãos para que possa desenvolver ainda mais sua política de distribuição de medicamentos e melhorar esse contato entre as unidades de saúde do município e os pacientes.

Para isso, foram levantados os requisitos com a equipe de desenvolvimento pensando em quais funcionalidades trazer para o aplicativo de modo a trazer essa facilidade para o usuário, tendo em mente também características visuais que melhorassem sua experiência de uso.

Um produto mínimo viável foi desenvolvido durante este trabalho para apresentar uma primeira versão da solução para a Prefeitura, e ao longo da parceria é previsto que o desenvolvimento continue até que o aplicativo esteja pronto para lançamento na Play Store. Para qualificar o produto desenvolvido, foram feitas duas etapas de avaliação que trouxeram pontos de melhorias que foram implementadas posteriormente, a primeira sendo a avaliação heurística, que trouxe alguns pontos de severidade alta como falta de familiaridade entre os componentes de interface ou falta de ajuda e documentação presentes no sistema. Já na segunda etapa de avaliação, com uma avaliação subjetiva de usabilidade, os resultados apontaram uma média de 89 pontos no método SUS, indicando uma boa usabilidade para o aplicativo.

Apesar de estar ainda em fase de desenvolvimento, pela definição de produto mínimo viável, o trabalho desenvolvido já poderia ser utilizado pelas pessoas, mesmo que certas funcionalidades ainda estejam por vir. Ainda assim, o produto criado poderá trazer sérios benefícios para os usuários bem como prover mudanças nos hábitos daqueles que poderiam melhorar a maneira como lidam com os medicamentos em casa. Para isso, é importante providenciar ao usuário ferramentas que mantenham o engajamento dele com a aplicação além de oferecer estes benefícios de maneira imediata, como as informações dos medicamentos da prefeitura e a ferramenta de gestão de farmácia pessoal, para que ele tenha esta percepção de valor do aplicativo.

Para o futuro, é proposto a continuação do desenvolvimento trazendo correções e melhorias para que o produto seja eventualmente liberado em ambiente de produção. Com isso, a coleta de dados a partir do feedback do usuário conforme ele tenta buscar os medicamentos nas unidades de saúde, registrado através do aplicativo poderá indicar importantes funcionalidades que podem ser agregadas ao aplicativo, e ainda subsidiar ações políticas do município que necessitem dos dados relativos ao consumo e distribuição de medicamentos.

REFERÊNCIAS

BARBOSA, Simone Diniz Junqueira et al. **Interação Humano-Computador e Experiência do Usuário**. [S. l.]: Autopublicação, 2021.

BROOKE, John. SUS: A quick and dirty usability scale. **Usability Eval. Ind.**, v. 189, nov. 1995.

CONSELHO FEDERAL DE FARMÁCIA. **Dados 2020**. [S. l.: s. n.], 2020. Atualizado em 20 de abril de 2021. Acesso em 09 de outubro de 2021. Disponível em: <<https://www.cff.org.br/pagina.php?id=801&titulo=Boletins>>.

FACEBOOK. **Learn the Basics**. [S. l.: s. n.], s.d. Acesso em 21 de novembro de 2021 no website da documentação oficial do React Native. Disponível em: <<https://reactnative.dev/docs/getting-started>>.

GOOGLE. [S. l.: s. n.], s.d. Acesso em 21 de novembro de 2021 no website da documentação oficial do Desenvolvimento para Android. Disponível em: <<https://developer.android.com/guide/components/processes-and-threads.html#Threads>>.

GOVERNO FEDERAL. **Lista de medicamentos de referência**. [S. l.: s. n.], 2020. Acesso em 02 de dezembro de 2021, atualizado em 26 de novembro de 2021. Disponível em: <<https://www.gov.br/anvisa/pt-br/setorregulado/regularizacao/medicamentos/medicamentos-de-referencia/lista-de-medicamentos-de-referencia>>.

IBGE. **Projeção da população do Brasil e das Unidades de Federação**. [S. l.: s. n.], 2021. Acesso em 09 de outubro de 2021. Disponível em: <<https://www.ibge.gov.br/apps/populacao/projecao/index.html>>.

MASTROIANNI, PC et al. Estoque doméstico e uso de medicamentos em uma população cadastrada na estratégia saúde da família no Brasil, 2011.

MEDISAFE. **Lembrete de Remédios e Pílula - Medisafe**. [S. l.: s. n.], 2021. Acesso de março a outubro de 2021. Disponível em: <<https://play.google.com/store/apps/details?id=com.medisafe.android.client>>.

- MOBILE CREATURES. **Pill Time Farmácia assistente para a sua saúde**. [S. l.: s. n.], 2021. Acesso de março a julho de 2021. Disponível em: <<https://play.google.com/store/apps/details?id=mobilecreatures.pillstime>>.
- NIELSEN, Jakob. **Usability Engineering**. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994.
- NIELSEN, Jakob; MOLICH, Rolf. **Heuristic Evaluation of User Interfaces**, 1990.
- OLIVEIRA, João Victor Lopes et al. A automedicação no período de pandemia de COVID-19: Revisão integrativa. v. 10, n. 3, mar. 2021. Disponível em: <<https://rsdjournal.org/index.php/rsd/article/view/13762>>.
- SOMMERVILLE, Ian. **Engenharia de Software 9ª Edição**. [S. l.]: Pearson, 2011.
- TEXEIRA, Fabricio. **O que é o SUS (System Usability Scale) e como usá-lo em seu site**. [S. l.: s. n.], 2015. Acesso em dezembro 2021. Disponível em: <<https://brasil.uxdesign.cc/o-que-%C3%A9-o-sus-system-usability-scale-e-como-us%C3%A1-lo-em-seu-site-6d63224481c8>>.

APÊNDICE A - Avaliação Heurística

Sugestões de Fluxos

- Ler as instruções de boa vinda do aplicativo e verificar se todas as informações ficaram claras

- Buscar medicamentos oferecidos pela prefeitura e as policlínicas participantes
 1. Abra o aplicativo e vá na aba "Procurar"
 2. Verifique a presença das informações desejadas.

- Cadastre um novo medicamento.
 1. Abra o aplicativo e vá na aba "Medicamentos"
 2. Clique no botão de adicionar e cadastre um novo medicamento com informações de uso
 3. Verifique o cadastro realizado na aba medicamentos, tocando no medicamento para abrir mais informações.
 4. Se for o caso, verifique a existência de doses com horário marcado na aba "Home".

- Marcar dose agendada como tomada
 1. Abra o aplicativo e vá na aba "Home"
 2. Se houver um medicamento no dia atual, marcar como tomado deslizando o item para a esquerda ou pressionando na dose para marcar dados adicionais.

- Marcar uma dose esporádica
 1. Abra o aplicativo e vá na aba "Home"
 2. Para marcar uma dose não agendada, clicar no ícone "+" e registrar uma nova dose.
 3. Verificar o registro novo na aba "Home"

E outros fluxos que lhe interessar

I. Visibilidade do estado do sistema

1. O usuário é informado sobre o que está acontecendo no sistema por meio de feedback e exibição das informações de forma clara, concisa e apropriada (Por exemplo, depois de o usuário cadastrar um medicamento o usuário recebe alguma confirmação de que o medicamento foi adicionado).

2. Componentes selecionados são destacados, enquanto componentes desabilitados são “acinzentados” ou omitidos (Por exemplo, botões, texto selecionado, ícones, caixa de seleção, menus, etc.) *

3. Informação crítica e contextual, como a gestão das doses marcadas, por exemplo, são priorizadas e visíveis. *

II. Correspondência entre o sistema e o mundo real

4. Os rótulos dos controles são consistentes com as suas ações (Por exemplo, o rótulo “Procurar” leva para a tela esperada) *

5. A informação aparece em uma ordem lógica e natural. *

6. As opções e rótulos de menu podem ser compreendidos rapidamente. *

7. 7. Ícones, imagens e outros componentes de interface são concretos e familiares. *

8. 8. As cores usadas correspondem as expectativas comuns sobre o código de cores. (Por exemplo, a cor vermelha é usada para indicar a prioridade máxima do atendimento). *

9. Sugestões

III. Controle e liberdade do usuário

10. 9. Os usuários podem avançar e retroceder na navegação do sistema *

11. 10. Os usuários podem salvar tarefas no meio da execução para continuar futuramente (Por exemplo, salvar um formulário parcialmente preenchido). *

12. Sugestões

IV. Consistência e padronização

13. 11. Os elementos da interface seguem uma terminologia padrão. *

14. 12. A navegação do sistema e o layout das telas são consistentes (Por exemplo, menus, botões de confirmação e áreas de texto principais sempre aparecem na mesma posição nas telas). *

15. 13. O estilo da interface é consistente nas telas do sistema (Por exemplo, o mesmo conjunto de cores e formas é usado em todas as telas). *

16. Sugestões

V. Reconhecimento em vez de memorização

17. 14. Todas as informações necessárias para o usuário realizar as tarefas são visíveis e/ou fáceis de encontrar (Por exemplo, quando o usuário está preenchendo o cadastro de medicamento ele consegue visualizar as informações dele na mesma tela e não precisa memorizá-las). *

18. 15. O sistema fornece todas as informações necessárias (Por exemplo, quando o usuário está preenchendo algum formulário o sistema provê todos os dados que ele precisa). *

19. 16. Os menus são balanceados, nem muito profundos nem largos demais (Por exemplo, o menu não tem muitos itens ou muitos níveis). *

20. Sugestões

VI. Flexibilidade e eficiência do uso

21. 17. As principais funcionalidades do sistema são fáceis de acessar. (Por exemplo, as funcionalidades principais podem ser alcançadas com poucos cliques) *

22. 18. Funcionalidades e tópicos relacionados estão próximos uns dos outros. *

23. 19. A entrada de dados e a navegação são mínimas o suficiente. *

24. 20. A necessidade de mudar de tela é evitada (Por exemplo, todo o conteúdo exibido cabe em uma tela). *

25. Sugestões

VII. Projeto estético e minimalista

26. 21. O design visual é atraente. *

27. 22. As telas têm um design “clean” apresentando somente informações e componentes importantes. *

28. 23. O conteúdo da tela é sempre visível completamente e não coberto por outros componentes de interface *

29. 24. O uso do espaço disponível da tela é adequado *

30. Sugestões

VIII. Prevenção de erros

31. 25. Os campos do formulário aceitam apenas o tipo de caractere correspondente (Por exemplo, o campo quantidade aceita somente valores inteiros). *

32. 26. As opções do menu são lógicas e compreensíveis. *

33. Sugestões

IX. Suporte para reconhecer, diagnosticar e reparar erros

34. 27. O sistema oferece revisão das informações para evitar possíveis erros antes da conclusão do atendimento. *

35. Sugestões

36. 28. Os botões têm uma distância adequada entre si para evitar que o usuário pressione o botão errado. *

X. Ajuda e documentação

37. 29. O sistema oferece suporte para eventuais dúvidas ou maiores esclarecimentos *

38. Sugestões

39. Defeitos

40. Sugestões

41. Comentários

Este conteúdo não foi criado nem aprovado pelo Google.

Google Formulários

APÊNDICE B - Teste de Usabilidade

9. Eu me senti confiante ao usar o sistema. *

Marcar apenas uma oval.

1 2 3 4 5

Discordo completamente Concordo completamente

10. Eu precisei aprender várias coisas novas antes de conseguir usar o sistema. *

Marcar apenas uma oval.

1 2 3 4 5

Discordo completamente Concordo completamente

11. Alguns pontos positivos que chamaram a atenção e você gostaria de mencionar? Obrigado!

12. Notou alguns pontos de melhoria que gostaria de me informar? Obrigado!
