



UNIVERSIDADE FEDERAL DO PARANÁ

GIOVANNI PISSAIA

SAÚDE DO BUGRE: APLICATIVO MÓVEL PARA AUXILIAR NA COLETA DE
DADOS EPIDEMIOLÓGICOS DA UNIDADE BÁSICA DE SAÚDE PRESIDENTE
TANCREDO NEVES

CURITIBA

2021

GIOVANNI PISSAIA

SAÚDE DO BUGRE: APLICATIVO MÓVEL PARA AUXILIAR NA COLETA DE
DADOS EPIDEMIOLÓGICOS DA UNIDADE BÁSICA DE SAÚDE PRESIDENTE
TANCREDO NEVES

Trabalho de Conclusão de Curso apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, como requisito parcial à obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. João Eugenio Marynowski

CURITIBA

2021



UNIVERSIDADE FEDERAL DO PARANÁ

ATA DE REUNIÃO

TERMO DE APROVAÇÃO

GIOVANNI PISSAIA

SAÚDE DO BUGRE: APLICATIVO MÓVEL PARA AUXILIAR NA COLETA DE DADOS EPIDEMIOLÓGICOS DA UNIDADE BÁSICA DE SAÚDE PRESIDENTE TANCREDO NEVES

Monografia aprovada como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Prof. João Eugenio Marynowski

Orientador – SEPT/UFPR

Prof. Dieval Guizelini

SEPT/UFPR

Prof.a Sandramara Scandelari Kusano de Paula Soares

SEPT/UFPR

Curitiba, 05 de Abril de 2021.



Documento assinado eletronicamente por **JOAO EUGENIO MARYNOWSKI, PROFESSOR DO MAGISTERIO SUPERIOR**, em 09/04/2021, às 11:21, conforme art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **SANDRAMARA SCANDELARI KUSANO DE PAULA SOARES, PROFESSOR ENSINO BAS/TEC/TECNOL**, em 09/04/2021, às 15:49, conforme art. 1º, III, "b", da Lei 11.419/2006.



Documento assinado eletronicamente por **DIEVAL GUIZELINI, PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 15/04/2021, às 15:39, conforme art. 1º, III, "b", da Lei 11.419/2006.



A autenticidade do documento pode ser conferida [aqui](#) informando o código verificador **3416895** e o código CRC **4F022379**.

Quem anda no trilho é trem de ferro.

Sou água que corre entre pedras:

- liberdade caça jeito.

(MANOEL DE BARROS, 2001, p. 32)

As palavras existem para nos servir, não para mandar em nós.

(RICHARD DAWKINS, 1986, p.18)

RESUMO

Unidades Básicas de Saúde (UBS) são a porta de entrada principal dos usuários do SUS, tendo como objetivo atender 80% da população, evitando assim a sobrecarga de hospitais. Entretanto, o desconhecimento do perfil epidemiológico de uma comunidade pela UBS impede ou limita a ação de profissionais da área médica na promoção e na proteção da saúde da população. O objetivo deste trabalho é oferecer à unidade Tancredo Neves, em Balsa Nova/PR, um aplicativo mobile capaz de cadastrar famílias, manter questionário de saúde, gravar a localização geográfica das residências através de GPS, já que muitas localidades não tem logradouro e número, subsidiar a criação de mapas com o perfil epidemiológico da população, possibilitar a melhor gestão da área de atuação dos Agentes Comunitários de Saúde, criar listas de visitas em função de agravos clínicos e sincronizar dados quando houver rede móvel disponível. Através da plataforma Android e da linguagem de programação Java foi criado um aplicativo customizado para o contexto da referida unidade de saúde. Em 30 dias de operação, o sistema ajudou a coletar informações de forma padronizada de 892 moradores da região, através de 7 agentes comunitários que estiveram em campo. Além disso, subsidiou o trabalho de conclusão de curso de pós-graduação do profissional médico cliente deste trabalho. Melhorias na sincronização com o servidor, a adição de recurso para a geração de uma rota de visitas, a possibilidade de acrescentar novas perguntas ao questionário de saúde e o desenvolvimento de uma interface web para visualizar os dados no mapa são oportunidades para trabalhos futuros.

Palavras-chave: Aplicativo para Unidade Básica de Saúde. Levantamento de dados epidemiológicos. Sistema para área médica. Sistema de informação para agentes de saúde pública.

ABSTRACT

Basic Health Units (BHU) are a main gateway for SUS users, aiming to serve 80% of the population, thus avoiding the overload of hospitals. However, the lack of knowledge of the epidemiological profile of a community by BHU prevents or limits the action of medical professionals in promoting and protecting the health of the population. The objective of this work is to offer to the Tancredo Neves unit, in Balsa Nova / PR, a mobile application capable of registering families, maintaining a health questionnaire, recording the geographic location of homes using GPS, since many locations do not have a street address and number, subsidize the creation of maps with the epidemiological profile of the population, enable better management of the area of activity of Community Health Agents, create lists of visits due to clinical conditions and synchronize data when a mobile network is available. Through the Android platform and the Java programming language, an application customized for the context of the health unit was created. In 30 days of operation, the hierarchical system to collect information in a standardized manner from 892 residents of the region, through 7 agents involved who were in the field. In addition, it supported the work of completing the graduate course of the medical professional client of this work. Improvements in the synchronization with the server, the addition of a resource for the generation of a visit route, a possibility to add new questions to the health questionnaire and the development of a web interface to view the data on the map are opportunities for future work.

Keywords: Application for Basic Health Unit. Survey of epidemiological data. System for the medical field. Information system for public health agents.

LISTA DE FIGURAS

FIGURA 1 - WBS	27
FIGURA 2 - TRELLO.....	28
FIGURA 3 - ETAPAS DO PROJETO	28
FIGURA 4 - TELA PRINCIPAL AJUSTÁVEL	34
FIGURA 5 - ARQUITETURA DO SOFTWARE	37
FIGURA 6 - TELA DE LOGIN.....	39
FIGURA 7 - TELA DE CADASTRO.....	39
FIGURA 8 - TELA PRINCIPAL.....	40
FIGURA 9 - CADASTRO DE FAMÍLIA.....	41
FIGURA 10 - CADASTRO DOS INTEGRANTES DA FAMÍLIA.....	41
FIGURA 11 - TELA DE CADASTRO DE MORADOR	42
FIGURA 12 - UMA FAMÍLIA COM DOIS MORADORES	42
FIGURA 13 - FILTRO DE FAMÍLIAS.....	43
FIGURA 14 - MENU DA TELA PRINCIPAL	43
FIGURA 15 - COMPARTILHAR ARQUIVO.....	44
FIGURA 16 - MORADOR MARCADO PARA VISITA.....	45
FIGURA 17 - POSIÇÃO DA FAMÍLIA NO MAPA	45
FIGURA 18 - SINCRONIZAR FAMÍLIAS.....	46
FIGURA 19 - GESTÃO DE USUÁRIOS	47
FIGURA 20 - GERENCIAR VISITAS.....	47
FIGURA 21 - CHECK-IN	48
FIGURA 22 - EDITAR COORDENADAS	49
FIGURA 23 - TELA DE CADASTRO DE USUÁRIOS	59
FIGURA 24 - TELA DE LOGIN.....	61
FIGURA 25 - TELA DE RECUPERAÇÃO DE SENHA.....	62
FIGURA 26 - TELA DE CADASTRO DE FAMÍLIAS.....	64
FIGURA 27 - TELA DE FAMÍLIAS CADASTRADAS.....	65
FIGURA 28 - TELA DE CADASTRO DE MORADORES	67
FIGURA 29 - TELA DE VISUALIZAÇÃO DE MORADORES	69
FIGURA 30 - TELA DE PRONTUÁRIO DO PACIENTE.....	71
FIGURA 31 - TELA DE PESQUISA	72
FIGURA 32 - TELA DE VISUALIZAÇÃO AGENDA DE VISITAS.....	74

FIGURA 33 - TELA DE CHECK-IN DE VISITAS.....	75
FIGURA 34 - TELA DE GERENCIAR USUÁRIOS.....	76
FIGURA 35 - TELA DE CONFIGURAÇÕES	78
FIGURA 36 - DIAGRAMA DE CASOS DE USO	57
FIGURA 37 - DIAGRAMA DE CLASSES	79
FIGURA 38 - DIAGRAMA LÓGICO-RELACIONAL.....	80
FIGURA 39 - LOGIN	81
FIGURA 40 – CADASTRAR-SE.....	81
FIGURA 41 – RECUPERAR SENHA	82
FIGURA 42 – CADASTRAR FAMÍLIA.....	82
FIGURA 43 – CADASTRAR MORADOR	83
FIGURA 44 – SINCRONIZAR FAMÍLIAS.....	83
FIGURA 45 – SINCRONIZAR VISITAS	84

LISTA DE TABELAS

TABELA 1 – SOFTWARES SEMELHANTES	22
TABELA 2 – PAPÉIS DOS INTEGRANTES.....	26
TABELA 3 – DESCRIÇÃO DOS ATORES E CASOS DE USO	58

LISTA DE ABREVIATURAS OU SIGLAS

UBS	- Unidade Básica de Saúde
SUS	- Sistema Único de Saúde
ACS	- Agente Comunitário de Saúde
GPS	- Global Position System
CEP	- Código de Endereçamento Postal
RAM	- Random Access Memory
SSD	- Solid State Drives
URL	- Uniform Resource Locator
WBS	- Work Breakdown Structure
CSV	- Comma Separated Values
RF	- Requisito Funcional
RNF	- Requisito Não Funcional
TADS	- Tecnologia em Análise e Desenvolvimento de Sistemas
SGBD	- Data Base Management System
ADM	- Administrador
UFPR	- Universidade Federal do Paraná
DV	- Data View

SUMÁRIO

1 INTRODUÇÃO	16
1.1 JUSTIFICATIVA	17
1.2 OBJETIVOS	18
1.2.1 Objetivo geral	18
1.2.2 Objetivos específicos.....	18
1.3 ESTRUTURA DO DOCUMENTO.....	19
2 FUNDAMENTAÇÃO TEÓRICA	20
2.1 SISTEMAS SEMELHANTES.....	20
2.1.1 e-SUS Território	20
2.1.2 Cidade Saudável ACS.....	21
2.1.3 ACS - Agente Comunitário de Saúde	21
2.1.4 ACS - CONSIDERAÇÕES	22
2.2 FUNDAMENTAÇÃO TEÓRICA DAS TECNOLOGIAS.....	23
2.2.1 A linguagem de programação Java	23
2.2.2 Android.....	23
2.2.3 Desenvolvimento Ágil	24
2.2.4 Trello	25
2.3 CONSIDERAÇÕES	25
3 MATERIAIS E MÉTODOS	26
3.1 EQUIPAMENTO	26
3.2 GERENCIAMENTO DO PROJETO	26
3.3 PLANO DE ATIVIDADES	26
3.4 PLANEJAMENTO DAS TAREFAS.....	28
3.5 LINGUAGEM DE PROGRAMAÇÃO	29
3.6 REQUISITOS FUNCIONAIS	29
3.7 REQUISITOS NÃO-FUNCIONAIS	29
3.8 PROTOTIPAÇÃO DE TELAS.....	30
3.9 DESENVOLVIMENTO DO PROJETO	30
3.9.1 Sprint 1	31
3.9.2 Sprint 2	31
3.9.3 Sprint 3	31
3.9.4 Sprint 4	32

3.9.5 Sprint 5.....	32
3.9.6 Sprint 6.....	32
3.9.7 Sprint 7.....	33
3.9.8 Sprint 8.....	33
3.9.9 Sprint 9.....	33
3.9.10 Sprint 10.....	33
3.9.11 Sprint 11.....	35
3.9.12 Sprint 12.....	35
3.9.13 Sprint 13.....	35
3.9.14 Sprint 14.....	35
3.9.15 Sprint 15.....	36
3.10 DISCUSSÃO.....	36
4 APRESENTAÇÃO DO APLICATIVO.....	37
4.1 ARQUITETURA DO SOFTWARE.....	37
4.2 MANUAL DO WEBSERVICE.....	38
4.3 MANUAL DO APLICATIVO.....	38
4.3.1 Demonstração do funcionamento.....	38
4.4 CONSIDERAÇÕES.....	49
5 CONSIDERAÇÕES FINAIS.....	50
5.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS.....	51
REFERÊNCIAS BIBLIOGRÁFICAS.....	52
APÊNDICE 1 – TENTATIVA DE CONTATO COM E-SUS TERRITÓRIO.....	53
APÊNDICE 2 – DIAGRAMA DE CASOS DE USO.....	57
APÊNDICE 3 – ESPECIFICAÇÃO DE CASO DE USO.....	59
APÊNDICE 4 – DIAGRAMA DE CLASSES.....	79
APÊNDICE 5 – DIAGRAMA LÓGICO-RELACIONAL.....	80
APÊNDICE 6 – DIAGRAMA DE SEQUÊNCIA.....	81
APÊNDICE 7 – TESTE DE CRITÉRIOS ERGONÔMICOS.....	85

1 INTRODUÇÃO

A Unidade Básica de Saúde Presidente Tancredo Neves (UBS) está localizada no distrito do Bugre, na cidade de Balsa Nova, Paraná, há 50km da capital paranaense. O propósito principal de uma UBS é disponibilizar à população atendimento especializado em postos de saúde instalados nos bairros da cidade, a fim de reduzir o fluxo de pacientes nos grandes hospitais (C S R PRESIDENTE TANCREDO NEVES, 2020).

Nesta UBS (também chamadas de Unidade de Saúde - US) há diversos profissionais, entre eles 1) Agentes Comunitários de Saúde (ACS), os quais tem como rotina visitar as famílias, cadastrar seus moradores, atualizar seus dados de saúde a cada visita e, idealmente, aplicar questionários para delineamento epidemiológico posterior; 2) Enfermeiros, que atuam como coordenadores dos ACS, desempenham papéis auxiliares na administração da US, além da atividade típica de enfermagem; e 3) Médicos, que fazem atendimentos para pessoas que comparecem à US e também atendimentos em domicílio, quando constatada a necessidade através do levantamento trazido pelos ACS ou por solicitação direta de moradores da comunidade.

A UBS usa o sistema e-SUS AB, o qual foi desenvolvido pelo Ministério da Saúde. Neste sistema, é possível triar pacientes, agendar consultas, cadastrar usuários, registrar prontuários, entre outras rotinas clínicas e administrativas no âmbito do posto de saúde.

Para registros de atendimentos realizados pelos agentes fora da Unidade de Saúde, foi lançado em 2017 um sistema móvel chamado e-SUS AB Território, desenhado para ser usado apenas em *tablets*. O sistema permite salvar informações *off-line*, possibilitando a sincronização posterior com os servidores do e-SUS AB (integrando as informações); o registro de famílias e de seus moradores, bem como dados de saúde predefinidos; e mostra relatórios básicos com descrição estatística das visitas. Por outro lado, o sistema não permite ajustes necessários à realidade local, tais como incluir um questionário de saúde com perguntas pertinentes à comunidade específica; não permite marcar a posição geográfica das famílias com o auxílio de GPS (muitos pacientes residem em locais sem endereço definido); não é voltado para o médico, portanto não há a funcionalidade que permite escrever no prontuário do paciente; não possibilita gerar uma lista de visitas.

A UBS implantou este sistema móvel, mas foi abandonado meses depois após alegações dos ACS que o software falhava na sincronização e que a usabilidade era 'burocrática' (o aplicativo não permitia avançar se o telefone do paciente não fosse cadastrado. Ocorre que nem mesmo todo o território é coberto pelas redes de telefonia (ou é de baixa qualidade) e nem mesmo todos os pacientes tem condições de ter um telefone, reforçando a falta de customização do software). A nota do aplicativo na Google Play, 2,6, sugere que o software ainda precisa ser melhorado e as reclamações presentes nas primeiras páginas estão relacionadas com a falha da sincronização (acessado em 08/08/2020).

1.1 JUSTIFICATIVA

Atualmente, os Agentes Comunitários de Saúde não aplicam questionários específicos de saúde, o que implica no desconhecimento do perfil epidemiológico da população atendida pela referida Unidade Básica de Saúde. Além disso, a cidade de Balsa Nova é predominantemente rural, não dispendo de endereçamento domiciliar típico, com nome de rua, número da casa, CEP, etc, em todas as suas ruas, dificultando a localização de famílias a serem visitadas. Por exemplo, mesmo que o ACS saia a campo fazer o registro dos domicílios da região, muitas vezes tem dificuldade de repassar aos profissionais da Unidade de Saúde a localização de eventuais pacientes que precisam de atendimento médico domiciliar.

A coordenação da equipe de Agentes Comunitários de Saúde é prejudicada pelo desconhecimento do total de domicílios da região, já que não tem condições de subdividir o território entre os ACS e otimizar o processo de levantamento de dados.

Há dificuldades enfrentadas também pelos médicos, como 1) conhecer o perfil epidemiológico da região (desconhecendo os problemas de saúde da população, dificulta-se a promoção de campanhas focadas na prevenção de doenças) e 2) saber quais pacientes necessitam de atendimento domiciliar, bem como obter o endereço desses pacientes (muitas vezes, o médico necessita do acompanhamento do ACS à visita do paciente, já que só este conhece a localização).

1.2 OBJETIVOS

1.2.1 Objetivo geral

O objetivo do trabalho é desenvolver e implementar um sistema informatizado para coleta de dados socioeconômicos e epidemiológicos da população atendida pela Unidade Básica de Saúde.

1.2.2 Objetivos específicos

- Viabilizar a coleta de informações de saúde dos moradores a fim de fornecer subsídios para traçar o perfil epidemiológico da Unidade Básica de Saúde Presidente Tancredo Neves (UBS).
- Permitir o cadastro das famílias visitadas pelos agentes de saúde.
- Possibilitar a marcação das coordenadas geográficas da moradia da família, bem como rotina de *check-in*.
- Permitir aos agentes de saúde chegar ao paciente cadastrado orientado por GPS, já que a região atendida pela UBS nem sempre tem endereçamento padrão.
- Gerar lista de visitas aos moradores, possibilitando a visualização do seu endereço no mapa.
- Permitir os registros das visitas no dispositivo móvel dos Agentes de Saúde - já que a cobertura da rede móvel é escassa na região - para, ao encontrar rede móvel disponível, sincronizar os dados com o servidor.
- Possibilitar o upload/download das informações registradas.

1.3 ESTRUTURA DO DOCUMENTO

O propósito deste documento é detalhar o desenvolvimento do projeto, desde o problema levantado até a solução em software apresentada.

O próximo capítulo tem como objetivo apresentar a fundamentação teórica que apoia a justificativa para demandar esforços para a solução do problema. Também são elencados os aplicativos semelhantes disponíveis no mercado bem como as suas limitações ante às necessidades do cliente.

O capítulo 3 visa trazer os materiais e métodos utilizados, como os equipamentos, a metodologia empregada, como foi feito o planejamento da demanda, quais foram os requisitos do sistema e como se deu o desenvolvimento do projeto.

O capítulo 4 é dedicado à apresentação do software. É mostrada a arquitetura do sistema, bem como os manuais para a configuração do webservice e do aplicativo mobile. O manual apresenta todas as telas do sistema e detalha todas as funcionalidades.

O capítulo 5 é reservado às considerações finais. Os objetivos são retomados e avaliados os êxitos e os insucessos do projeto. Aponta, ainda, direções para empreender novos esforços em melhorias e oportunidades futuras.

Após os capítulos citados, estão presentes neste documentos os apêndices que contém alguns artefatos criados e demais informações relevantes ao projeto.

2 FUNDAMENTAÇÃO TEÓRICA

No Brasil, é o Sistema Único de Saúde (SUS) quem integra todas as ações e serviços públicos de saúde. A gestão é realizada em conjunto pela federação, estados e municípios e o DATASUS é o departamento de informática do SUS. Ele é responsável, entre outras coisas, pela criação de softwares para gerenciamento das Unidades Básicas de Saúde, os quais são oferecidos gratuitamente para as prefeituras. A implantação é realizada pelas prefeituras através de seus próprios departamentos de informática (MINISTÉRIO DA SAÚDE, 2020).

O e-SUS Atenção Primária é uma estratégia do Departamento de Saúde da Família que visa integrar as informações da Atenção Básica em nível nacional. O software e-SUS AB é usado nas Unidades Básicas de Saúde para registros de ações realizadas em pacientes que comparecem à Unidade. Já o e-SUS Território é um aplicativo *mobile* destinado aos profissionais de saúde que vão a campo coletar informações sobre a saúde da população (SAPS, 2020).

Apesar de todas as cidades do país terem semelhanças entre si, observa-se características que são únicas em algumas regiões, às quais requerem novas funcionalidade ou customizações que não são possíveis ou fácil de requerer junto ao DATASUS. A geolocalização de famílias é imperativa em cidades onde não há ruas com logradouro, número e CEP e o método de coleta de dados também difere de cidade para cidade, já que algumas cidades não dispõem de cobertura de internet móvel adequada.

2.1 SISTEMAS SEMELHANTES

Tanto no Ministério da Saúde quanto na Google Play, que é o repositório oficial de aplicativos *Android*, encontram-se *softwares* que facilitam a gestão de informações da UBS. Porém, como já mencionado, as peculiaridades de cada município demandam funcionalidades específicas.

2.1.1 e-SUS Território

O e-SUS Território é desenvolvido pelo DATASUS, que é o Departamento de Informática do Ministério da Saúde. Ele tem o objetivo de facilitar o processo de

trabalho dos agentes de saúde em campo. As principais características do aplicativos são:

- Cadastro de famílias;
- Cadastro de moradores;
- Cadastro de condições sanitárias da moradia;
- Cadastro de condições socioeconômicas;
- Registro de algumas condições de saúde;
- Registro das informações coletadas no dispositivo;
- Envio de informações coletadas para o servidor da prefeitura que gerencia os demais sistemas necessários para viabilizar essa troca;

2.1.2 Cidade Saudável ACS

Aplicativo desenvolvido pela Sysvale Softgroup Ltda e é um sistema pago. Pode ser adquirido pelas prefeituras para coletar dados em paralelo ao sistema e-SUS AB, já que este não se conecta aos servidores do SUS. As principais funcionalidades deste sistema são:

- Realização de Cadastro Individual, Cadastro Domiciliar, Visita Domiciliar, entre outras fichas utilizadas com frequência;
- Envio de fichas cadastradas ao conectar-se com a internet;
- Obtenção de coordenadas para geolocalização de pacientes;
- Análise estatística de dados.

2.1.3 ACS - Agente Comunitário de Saúde

É um aplicativo desenvolvido pela FGMAISS e funciona integrado ao sistema do Governo Federal. As características descritas pelos desenvolvedores são:

- Integra com o módulo ACS implantado no Município Administrador;
- Útil para realizar as Fichas de Cadastros Domiciliar e Territorial, Cadastro Individual, Visitas Domiciliares, Ficha de Atividade Coletiva e Marcadores de Consumo Alimentar;
- Permite o acompanhamento de gestantes, pré-natal ou puerpério, hipertensos (HA), diabéticos (DIA), pessoas com tuberculose ou

hanseníase, crianças, acamados, domiciliados, idosos, bem como contribuir nas campanhas de saúde do seu município.

2.1.4 CONSIDERAÇÕES

Nos softwares apresentados, uma ou mais das seguintes características estão ausentes ou o custo financeiro não é compatível com os interesses da UBS:

- Não é possível alterar o questionário de saúde adicionando perguntas específicas;
- Não registra coordenadas geográficas;
- É dependente do e-SUS AB;
- Não permite que o gestor da UBS crie uma lista de visitas para o ACS.

Na TABELA 1 se fez um resumo do comparativo entre os softwares semelhantes pesquisados com as funcionalidades propostas pelo Saúde do Bugre.

TABELA 1 – SOFTWARES SEMELHANTES

	Manter famílias	Questionário de saúde customizado	Permite geo-localização	Faz lista de visitas	Necessário e-SUS AB	Pago
e-SUS Território	Sim	Não	Não	Não	Sim	Não
Cidade Saudável ACS	Sim	Não	Não	Não	Não	Sim
ACS - Agente Comunitário de Saúde	Sim	Não	Sim	Não	Sim	Não
Saúde do Bugre	Sim	Sim	Sim	Sim	Não	Não

FONTE: O autor (2021).

O comparativo mostra que o Saúde do Bugre é o único que permite customizar as perguntas do questionário de saúde e fazer uma lista de visitas. Além disso, não depende do e-SUS AB para funcionar, não tem custo financeiro para a UBS e permite registrar a geolocalização das famílias.

2.2 FUNDAMENTAÇÃO TEÓRICA DAS TECNOLOGIAS

O desenvolvimento de aplicativos requer o uso de inúmeras tecnologias, tanto em hardware quanto em software. A partir da combinação de um conjunto de técnicas e ferramentas pode-se produzir sistemas de informação para qualquer setor da sociedade. É recorrente o uso de tecnologias para planejar o desenvolvimento de sistemas, para nortear a engenharia de software bem como para a aplicação correta do domínio técnico das linguagens de programação.

2.2.1 A linguagem de programação Java

A Java é uma linguagem de programação desenvolvida na década de 90, orientada a objetos, independente de plataforma, segura e com uma vasto conjunto de bibliotecas. É a plataforma de desenvolvimento mais adotada por companhias e desenvolvedores (ORACLE, 2021).

A plataforma Java é a tecnologia que abarca todos os componentes necessários para produzir programas de computador. Ela é formada pela Java Virtual Machine (JVM) e pela Java Application Programming Interface (API), a qual é uma coleção de bibliotecas com funções prontas para serem usadas pelos programadores (ORACLE, 2021).

Para iniciar o desenvolvimento basta escrever um código-fonte respeitando a sintaxe Java e salvá-lo em um arquivo com a extensão .java. Na sequência, este arquivo é compilado pelo programa javac e convertido em um arquivo com a extensão .class. É este arquivo que tem os bytecodes que serão interpretados e executados pela Máquina Virtual Java (JVM) (DEV MEDIA, 2021).

2.2.2 Android

O Android é um sistema operacional iniciado em 2003 por uma empresa norte-americana de mesmo nome. Em 2005, foi comprada pela Google e, em 2007, ocorreu o lançamento do projeto Android, que tinha como objetivo desenvolver um sistema para dispositivos móveis sob padrão aberto e construído sobre o kernel do Linux.

Este projeto é liderado pela Google, mas faz parte de um consórcio de empresas que inclui Sony, Samsung, operadoras de telefonia e fabricantes de dispositivos.

O primeiro aparelho comercializado com Android foi lançado em 2008 nos Estados Unidos. Desde então, sucessivas atualizações foram lançadas e hoje detém 75% do mercado de smartphones do mundo. (GUIMARÃES, 2021).

Hoje são mais de 2,5 bilhões de dispositivos que rodam Android. É um sistema de código aberto, em que empresas de tecnologias e desenvolvedores podem contribuir com o sistema. Está presente em uma variedade de dispositivos, tanto baratos quanto caros, o que confere a ele uma presença ampla no mercado. Empresas que embarcam o Android em seus dispositivos, podem customizá-lo de acordo com os seus interesses. Além disso, o Android está integrado com os serviços do Google, o que melhora a produtividade.

2.2.3 Desenvolvimento Ágil

Profissionais veteranos da área de desenvolvimento de software, reuniram-se para discutir formas de melhorar o desenvolvimento de seus projetos. Cada qual com sua bagagem e contribuição, acabaram por concordar que havia um conjunto de princípios que parecia ter sempre sido respeitado quando os projetos davam certo (DESENVOLVIMENTO ÁGIL, 2020).

A partir da discussão desse encontro, foi criado o Manifesto Ágil, que oferece uma abordagem para se desenvolver software baseados nos princípios a seguir:

- Indivíduos e interação entre eles mais que processos e ferramentas;
- Software em funcionamento mais que documentação abrangente;
- Colaboração com o cliente mais que negociação de contratos;
- Responder a mudanças mais que seguir um plano (MANIFESTO ÁGIL, 2000).

Norteados pelo Manifesto Ágil, inúmeras metodologias ágeis foram lançadas desde então. Uma delas, a Scrum, teve origem em um artigo japonês que resumia as 10 melhores práticas em empresas japonesas. A partir da leitura deste material, os desenvolvedores Jeff Sutherland e Ken Schwaber formalizaram o Scrum para o desenvolvimento de software a partir de suas experiências.

Nesta metodologia, o projeto é dividido em Sprints, que é um período de tempo. Dentro desse período, um conjunto de atividades deve ser executada. Várias Sprints são planejadas para que um projeto chegue ao fim.

Todas as funcionalidades a serem implementadas são mantidas em um lista conhecida como Product Backlog. Entre os papéis desempenhados pela equipe, há o do Product Onwner, que é o cliente do projeto. É ele quem prioriza os itens que constam na lista. Tipicamente, em todo início da manhã de cada dia da Sprint, ocorre o Daily Scrum coordenado pelo Scrum Master, que é o facilitador da equipe. O Daily Scrum é uma breve conversa entre a equipe a fim de disseminar o conhecimento do que foi produzido no dia anterior, identificar impedimentos e priorizar o que será executado no dia que começa. Como se trata de uma abordagem iterativa, a cada nova Sprint o ciclo recomeça (DESENVOLVIMENTO ÁGIL, 2020).

2.2.4 Trello

O Trello é um software online que implementa o paradigma Kanban. Nele, os projetos são representado por quadros, os quais contém listas com várias tarefas. Cada tarefa é um card, o qual pode ser movido entre as listas, de modo a alterar o seu progresso.

Este sistema pode ser usado tanto para uso pessoal como para o trabalho. Entre os exemplos de usos, inclui o gerenciamento de projetos de software, planejamento de aulas, gerenciamento de escritórios, entre outros (TRELLO, 2021).

2.3 CONSIDERAÇÕES

Neste capítulo apresentaram-se os sistemas de informação oferecidos pelo Ministério da Saúde aos municípios, bem como o seu caráter generalista e, portanto, nem sempre adequado à todas as cidades. Além disso, referenciaram-se aplicativos existentes e similares ao objetivo deste trabalho e foram abordadas as tecnologias que dão suporte ao planejamento e à Engenharia de Software.

3 MATERIAIS E MÉTODOS

Para o desenvolvimento deste projeto, foram utilizados os recursos da metodologia ágil bem como a utilização de ferramentas gerenciais e técnicas para controlar o ritmo das entregas e manter o controle dos artefatos, que estão descritas em detalhes nas subseções a seguir.

3.1 EQUIPAMENTO

Para desenvolver este trabalho foi utilizado um notebook Samsung Series 5, processador Intel i7, 8GB de memória RAM, SSD com capacidade de 240GB, com sistema operacional Windows 10 e com os seguintes aplicativos instalados: Android Studio 4.0, MariaDB 10.4, Eclipse Oxygen (release 4.7.3a), Office 2016 e Microsoft Teams.

3.2 GERENCIAMENTO DO PROJETO

Para orientar a entrega de todos os artefatos de software para o cliente, foi adotado o framework *Scrum*. Desta metodologia, foram utilizados alguns artefatos, como *Sprint*, *Sprint Backlog*, *Product Backlog*, *Product Owner* e *Scrum Master*. A distribuição dos papéis pelos integrantes do projeto está resumida na TABELA 2.

TABELA 2 – PAPÉIS DOS INTEGRANTES

Papel	Responsável
<i>Product Owner</i>	Equipe de saúde
<i>Scrum Master</i>	Giovanni Pissaia

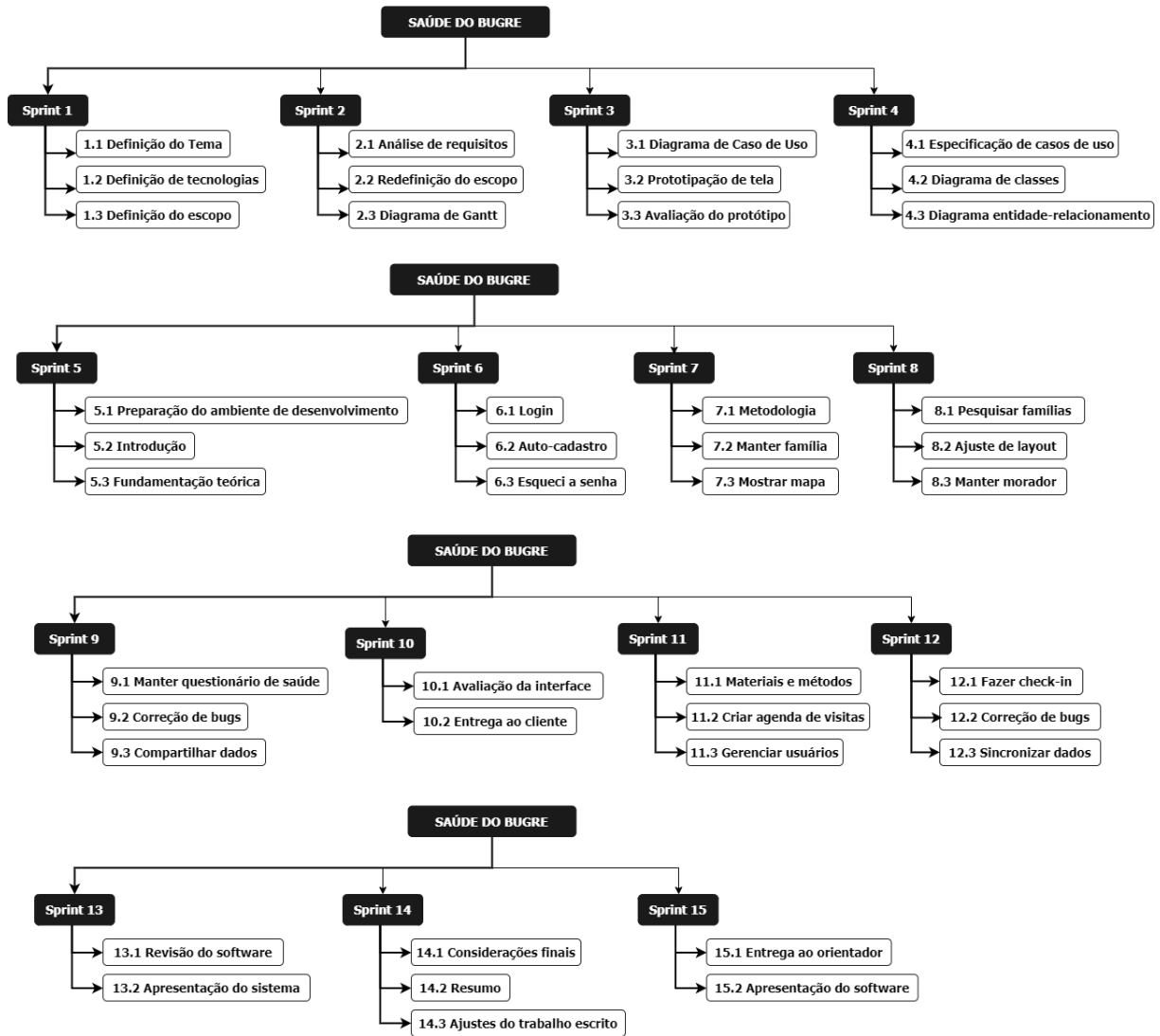
FONTE: O autor (2021).

3.3 PLANO DE ATIVIDADES

Este projeto foi dividido em atividades menores, visando o melhor gerenciamento do projeto. Dentro do framework *Scrum*, as atividades são entregues em Sprints e estas tem um período que é definido pelo usuário. No presente projeto, o tempo de cada Sprint é de uma semana. Estas atividades estão ilustradas na WBS

(Work Breakdown Structure) (FIGURA 1) e a descrição de cada Sprint está na seção 3.9.

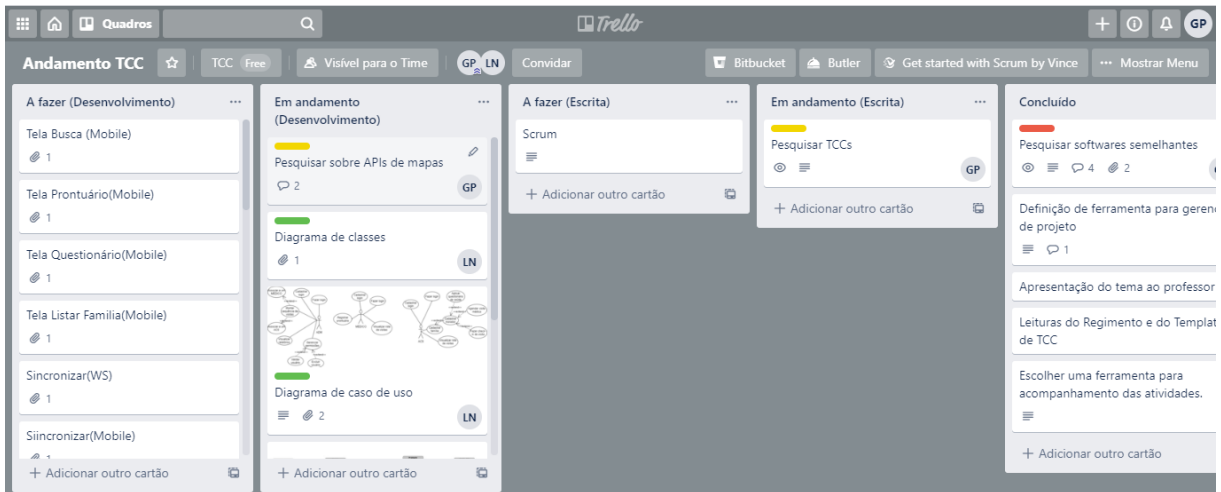
FIGURA 1 - WBS



FONTE: O autor (2021).

No aplicativo Trello, o *Backlog* foi dividido em *Sprints* e cada Sprint foi representada em um card (FIGURA 2).

FIGURA 2 - TRELLO

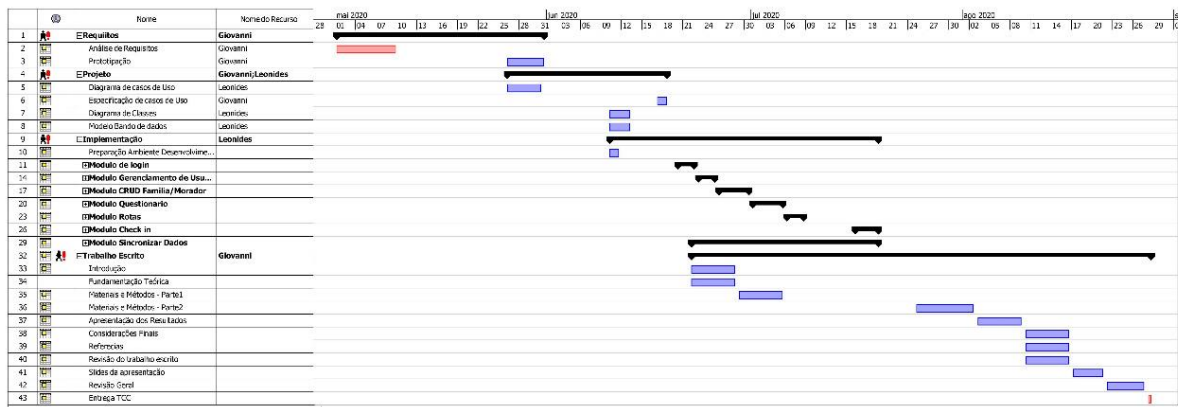


FONTE: O autor (2021).

3.4 PLANEJAMENTO DAS TAREFAS

Para o planejamento das tarefas junto ao professor-orientador, a fim de conferir estimativas de datas para cada uma das atividades prescritas nas Sprints, também foi adotado o Gráfico Gantt para o planejamento geral deste trabalho (FIGURA 3).

FIGURA 3 - ETAPAS DO PROJETO



FONTE: O autor (2021).

3.5 LINGUAGEM DE PROGRAMAÇÃO

Como foi constatado que todos os agentes de saúde da UBS possuíam celulares com o sistema operacional *Android*, a linguagem de programação mais adequada para a realização do projeto foi a Java.

3.6 REQUISITOS FUNCIONAIS

A partir das conversas com os clientes, foi possível mapear as necessidades e definir os requisitos funcionais do sistema:

- RF01 – possibilitar o cadastro de usuários novos;
- RF02 – possibilitar uma forma de realizar login;
- RF03 – possibilitar ao usuário fazer logout do sistema;
- RF04 – possibilitar a recuperação de senha;
- RF05 – possibilitar o cadastro de famílias;
- RF06 – possibilitar a visualização das famílias cadastradas;
- RF07 – possibilitar o cadastro de moradores;
- RF08 – possibilitar a visualização dos moradores cadastrados;
- RF09 – possibilitar a atualização do prontuário de pacientes;
- RF10 – possibilitar a pesquisa de famílias e moradores;
- RF11 – possibilitar aos administradores criar agenda de visitas;
- RF12 – possibilitar aos usuários ver a agenda de visitas;
- RF13 – possibilitar a realização do check-in nas famílias visitadas;
- RF14 – possibilitar o envio de dados ao servidor;
- RF15 – possibilitar ao administrador gerenciar usuários;
- RF16 – possibilitar exportar os dados no formato csv;
- RF17 – possibilitar ao usuário gerenciar as redes que deseja enviar seus

dados;

3.7 REQUISITOS NÃO-FUNCIONAIS

Os Requisitos não funcionais são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança,

disponibilidade, manutenção e tecnologias envolvidas. Estes requisitos dizem respeito a como as funcionalidades serão entregues ao usuário do software.

RNF01 – O sistema precisa ser desenvolvido para *Android*;

RNF02 – O sistema deve armazenar os dados no celular do usuário e permitir sincronizar com um servidor;

3.8 PROTOTIPAÇÃO DE TELAS

O esboço das telas (APÊNDICE 1) e os diagramas foram feitos através do sistema *draw.io*, acessado através da *url* de mesmo nome. A prototipação de telas é um meio para facilitar o entendimento dos requisitos do software, bem como lançar luz sobre os principais conceitos e funcionalidades.

3.9 DESENVOLVIMENTO DO PROJETO

O tema deste projeto foi apresentado ao professor-orientador após a reunião com o cliente. Este informou sobre suas necessidades e a dificuldade de conseguir um aplicativo customizado para a realidade da UBS.

Tendo em vista que já existia uma aplicação semelhante desenvolvida pelo Ministério da Saúde, o professor-orientador sugeriu que entrássemos em contato com os desenvolvedores a fim de verificarmos a possibilidade de trabalharmos em uma nova *branch* do projeto, inserindo as funcionalidades requeridas pelo cliente, mas não obtivemos respostas nas três tentativas de contato que fizemos (APÊNDICE 2).

O projeto foi iniciado em dupla mas, por motivos ainda não revelados, um dos integrantes abandonou o trabalho e não entrou mais em contato. Ele participou da elaboração inicial de alguns artefatos, como o diagrama de casos de uso e do rascunho da modelagem do banco de dados, ambos modificados em etapas posteriores.

Nesta seção serão descritos os 15 Sprints planejados. O tempo definido para cada Sprint foi de uma semana.

3.9.1 Sprint 1

Inicialmente, agendou-se uma reunião via WhatsApp entre os integrantes da equipe para a definição do tema do trabalho bem como as tecnologias e conhecimento necessários para a sua realização. Estabeleceu-se que o escopo do projeto seria um aplicativo *mobile*, um aplicativo web e um *webservice*.

3.9.2 Sprint 2

Agendou-se uma reunião com o cliente para conhecer mais detalhadamente as suas necessidades e expectativas. Um fluxo do processo foi rascunhado pelo cliente e, a partir dele, foram estabelecidos os requisitos de software. O sistema deveria ser composto de um aplicativo mobile e um aplicativo web – o qual apresentaria gráficos e relatórios para uma melhor gestão das pessoas e das informações coletadas pelo aplicativo mobile. Entretanto, devido às limitações de tempo e de conhecimento, o escopo do projeto reduziu-se, sendo excluído o aplicativo web.

A partir dessas informações e do escopo definido, foi elaborado o diagrama de Gantt para marcar as etapas do projeto e atribuir responsáveis para as tarefas.

3.9.3 Sprint 3

Na tentativa de validar os requisitos do sistema, desenvolveu-se o diagrama de casos de uso (APÊNDICE 3). Com ele, evidenciou-se os atores principais, as funcionalidades atribuídas e as relações entre si. Os atores iniciais eram médicos, enfermeiros e agentes comunitários de saúde, mas uma reanálise do negócio estabeleceu os atores administrador (ADM) e agente comunitário de saúde (ACS), unificando médicos e enfermeiros como administradores.

Além do diagrama de caso de uso, foi apresentado ao cliente os protótipos das telas. Imprimiu-se em papel e recortou-se as imagens que representavam a interface, e a partir disso, uma interação entre o cliente e os desenvolvedores simulou o comportamento do sistema e permitiu-se definir adequadamente cada conceito conversado na Sprint anterior.

3.9.4 Sprint 4

Com o êxito da simulação do funcionamento do software, iniciou-se a especificação dos casos de uso (APÊNDICE 1). O caso de uso busca detalhar o que o sistema deve fazer, passo a passo, para produzir um resultado.

Para estabelecer a estrutura e a relação entre as classes, elaborou-se o diagrama de classes (APÊNDICE 4). E para estruturar o software em relação à persistência de dados, o diagrama lógico-relacional também foi produzido nesta Sprint (APÊNDICE 5). A sequência do processo está representado no APÊNDICE 6.

3.9.5 Sprint 5

Na sprint 5, já estabelecidos o que deveria ser feito, iniciou-se os preparativos do ambiente de desenvolvimento. Para o desenvolvimento do aplicativo mobile foi instalado o Android Studio e para o desenvolvimento do webservice foi instalado o Eclipse. Além desses, foi instalado e criado o banco de dados usando o SGBD MariaDB.

Em paralelo à codificação do sistema, o trabalho escrito começou a ser produzido. A Resolução nº 01/18 (TADS, 2020), que estabelece as normas e critérios para a elaboração e avaliação do trabalho de conclusão de curso, foi baixada e estudada. Todas as obrigações foram aplicadas ao trabalho escrito, e aquelas facultativas, debatidas com o professor-orientador.

Iniciou-se uma pesquisa bibliográfica para fundamentar conceitos-chave do projeto. Várias fontes foram consultadas e anotadas, e aqueles pertinentes, reservadas.

As seções “Introdução” e “Fundamentação Teórica” foram adicionadas ao trabalho, seguindo o layout disponível no site do TADS (TADS 2020).

3.9.6 Sprint 6

Com os diagramas de classe e de banco de dados finalizados, iniciou-se a codificação das funcionalidades que envolvem o acesso do usuário ao sistema. A tela de login, auto-cadastro e de recuperação de senha foram programadas, bem como o script de banco de dados referentes à estas funcionalidades.

3.9.7 Sprint 7

Na Sprint 7, retomou-se o trabalho escrito, dando-se início à seção de Materiais e Métodos. Citaram-se os equipamentos e os softwares utilizados e iniciou-se a escrita do procedimento que remete à Engenharia de Software utilizada no projeto.

Na mesma semana, a tela principal do sistema, que é a tela que lista e cadastra famílias, foi desenvolvida. Nesta fase, concluíram-se as possibilidades de incluir, alterar e excluir famílias. Devido à complexidade do desenvolvimento de um software, foram feitos alguns ajustes em relação ao modelo inicial, adicionando imagens, textos e a funcionalidade de filtrar a lista de famílias.

3.9.8 Sprint 8

Na Sprint 8, adicionou-se o campo de pesquisa, o qual permite filtrar famílias. Se encontrados o conjunto de caracteres informados pelo usuário, as famílias permanecem na lista. A função “adicionar morador” foi implementada, desde a tela que lista os moradores de uma dada família até as operações com o banco de dados.

3.9.9 Sprint 9

Entregou-se o compartilhamento dos dados coletados, que é a funcionalidade que processa os dados coletados e gera uma tabela única que pode ser enviada aos gestores. Ainda nesta Sprint desenvolveu-se o questionário de saúde e o adicionou na tela de cadastro de moradores. Verificou-se o funcionamento geral do aplicativo, desde o cadastro de uma família, passando pelo cadastro de moradores até gerar o arquivo que reunia todos os dados coletados. Falhas foram corrigidas.

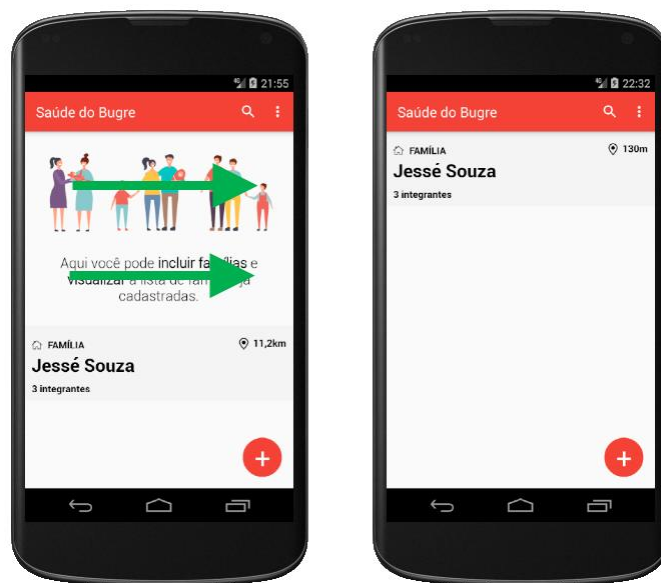
3.9.10 Sprint 10

A conclusão das funcionalidades descritas na Sprint 9 foram as mais aguardadas pelo cliente porque encerravam um ciclo importante de entrega. Além da inspeção do software, um teste para avaliar critérios ergonômicos (APÊNDICE 7) foi

aplicado na Sprint 10. O desenvolvedor aplicou o teste no médico-administrador da UBS.

A partir do resultado do teste, observou-se a necessidade de instruir o usuário em como cadastrar uma família, um morador e para que servia as telas principais. Adicionou-se, portanto, uma mensagem nas telas de Famílias e Moradores. Porém, isso gerou inconveniências para alguns usuários que possuíam dispositivos onde a tela era comparativamente pequena em relação aos demais. Foram necessários mais alguns ajustes de layout. A imagem e o texto explicativo da tela de famílias deixavam pouco espaço para as informações da lista. Para resolver esse problema, foi adicionado um recurso que permite ocultar tanto a imagem quanto o texto explicativo pressionando o dedo e deslizando para a esquerda (FIGURA 4).

FIGURA 4 - TELA PRINCIPAL AJUSTÁVEL



FONTE: O autor (2021).

Enviou-se o aplicativo através do WhatsApp para o administrador da UBS bem como orientações básicas de como proceder com a instalação (conforme consta no Capítulo 4).

3.9.11 Sprint 11

Na semana 11, adicionaram-se as funcionalidades “Criar lista de visitas” e “Gerenciar usuários”. Ambos os recursos fazem parte do menu da tela inicial e são acessíveis somente aos administradores.

Em paralelo ao desenvolvimento do software, a seção de “Materiais e Métodos” foi atualizada com as informações das Sprints.

3.9.12 Sprint 12

O aplicativo precisa funcionar off-line durante a coleta de dados e online somente para enviar os dados para o servidor, necessitando criar um banco de dados no dispositivo e outro no servidor. Para a sincronização entre esses bancos de dados não encontrou-se um padrão de projeto, o que justificou uma adaptação na tabela “familia” no servidor para receber e individualizar por usuário. Foram adicionados os campos “id_familia_app” e “id_usuario” para a correta identificação da origem.

Nesta mesma Sprint, a última funcionalidade foi adicionada ao software: o check-in. Após, uma nova inspeção do código foi realizada e foram corrigidos bugs de sincronização.

3.9.13 Sprint 13

O software foi revisado e testado. O teste consistiu em cadastrar-se, logar-se e recuperar senha; cadastrar, editar e apagar família e morador; aprovar e recusar novos usuários; gerar lista de visitas; e sincronizar dados. Após esse procedimento, o software foi apresentado ao professor-orientador.

3.9.14 Sprint 14

Com o término das atividades do desenvolvimento do software, os esforços foram direcionados para o trabalho escrito. O capítulo dedicado às considerações finais foi redigido e, com o trabalho escrito terminado, adicionou-se o resumo ao texto.

Iniciou-se a revisão do trabalho escrito, a adesão do projeto às regras da Resolução N.º 01/18 e os procedimentos para a entrega do pré-projeto.

3.9.15 Sprint 15

A última Sprint dedicou-se à entrega para o professor-orientador do pré-projeto e do desenvolvimento do material para a apresentação da defesa do projeto.

3.10 DISCUSSÃO

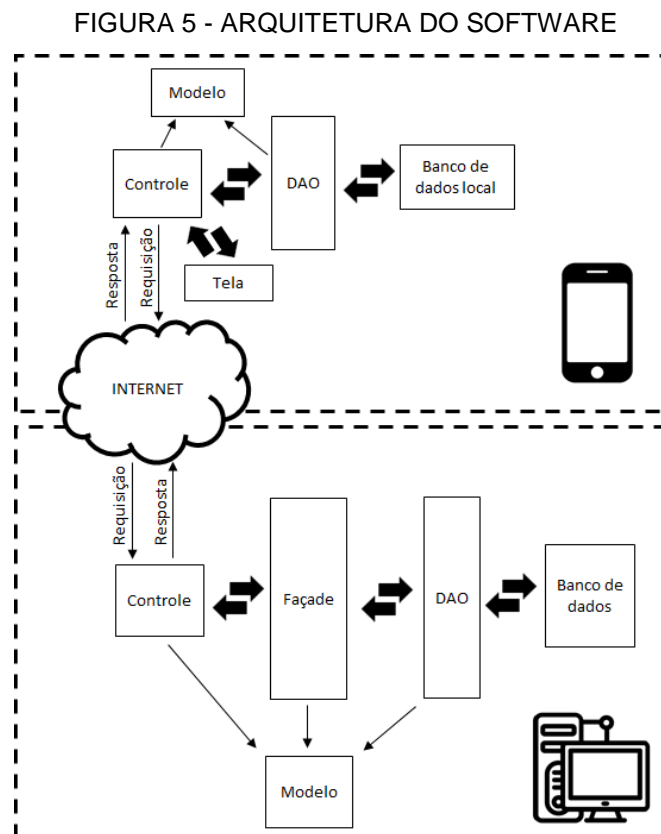
A metodologia ágil foi usada neste projeto, com o emprego adaptado do framework Scrum. Para o desempenho do projeto o único equipamento necessário foi um notebook com hardware padrão. Explicou-se sobre o plano de atividades, composto de Sprints semanais, sobre o controle do backlog com o uso do Trello e sobre o planejamento das tarefas, atribuindo responsabilidades e determinando os períodos de entrega no calendário. Os requisitos funcionais e não funcionais foram detalhados e, a partir deles, elaborou-se a prototipagem das telas, as quais foram apresentadas e discutidas com o cliente. Por fim, seguiu-se com a explicação detalhada do passo a passo do desenvolvimento do projeto, comentando-se sobre cada Sprint. A seguir, será apresentado o software construído.

4 APRESENTAÇÃO DO APLICATIVO

O software desenvolvido neste projeto é composto por dois sistemas: um mobile e, outro, um webservice. O aplicativo mobile é disponibilizado para o sistema operacional Android e o webservice necessita de um servidor com suporte à linguagem de programação Java e um servidor de aplicação GlassFish.

4.1 ARQUITETURA DO SOFTWARE

Os sistemas interagem entre si através da internet. O usuário mobile pode fazer uma requisição que é enviada para o servidor, que a recebe e interage com o modelo de negócio. Na sequência, o pedido é encaminhado para a Façade, que simplifica a interface com o cliente, omitindo a complexidade do sistema. A Façade, então, recebe a requisição e encaminha para a camada que centraliza todo o acesso ao banco de dados, a DAO. É ela que acessa e garante a persistência dos dados. Após o processamento da requisição, o Controle devolve a resposta ao cliente. O processo está modelado conforme a FIGURA 5.



FONTE: O autor (2021).

4.2 MANUAL DO WEBSERVICE

O webservice é a solução que faz a integração do sistema, o qual recebe, processa e persiste os dados do usuário do aplicativo mobile.

Para a instalação do webservice, escolher um servidor e abrir o arquivo `saude-bugre-ws.jar`, configurar a classe `Servidor.class` com as configurações desejadas e executar o arquivo `tabelas.sql` no SGBD MariaDB 10.4.13.

O arquivo `apk`, que contém o aplicativo mobile também precisa ser configurado e, por isso, compilado novamente. É necessário apontar o servidor nas classes que fazem conexão com o servidor.

4.3 MANUAL DO APLICATIVO

O usuário precisa ter um smartphone Android (versão 4.4 ou superior) e habilitar o sistema para que seja permitida a instalação de aplicativos externos à Google Play. Para fazer isso, basta ir na opção “Configurar” e habilitar a opção que permite a instalação de aplicativos de terceiros (detalhes dependem da versão instalada e são facilmente encontrados no Google).

Satisfeitas as exigências do parágrafo anterior, o usuário deverá receber o arquivo `saude_bugre.apk` do administrador do sistema e, ao tocá-lo, iniciará a instalação.

4.3.1 Demonstração do funcionamento

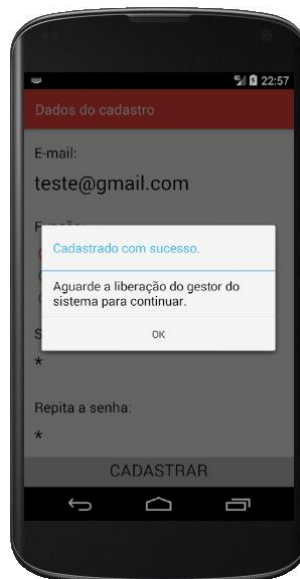
A tela de *login* (FIGURA 6) permite ao usuário entrar no sistema, se informados o seu *e-mail* e a senha. Caso seja o primeiro acesso, o usuário deverá cadastrar-se, clicando no botão “Não sou cadastrado” e preencher todos os campos requisitados. Ao fazê-lo, receberá a mensagem “Aguarde a liberação do gestor do sistema” (FIGURA 7).

FIGURA 6 - TELA DE LOGIN



FONTE: O autor (2021).

FIGURA 7 - TELA DE CADASTRO



FONTE: O autor (2021).

Caso o usuário já tenha cadastrado mas tenha esquecido a senha, é possível recuperá-la tocando no botão “Recuperar senha”. Um senha provisória então é encaminhada para o e-mail informado, o qual deve estar cadastrado previamente. Após o recebimento da senha, o usuário faz login e é direcionado para uma nova tela, que o obriga a criar uma nova senha.

Tendo as credenciais para cadastro e após a liberação pelo gestor do sistema, o usuário poderá acessar a tela principal do aplicativo (FIGURA 8). Nela, é possível visualizar as famílias cadastradas e adicionar famílias tocando no ícone vermelho que contém o sinal de “+”.

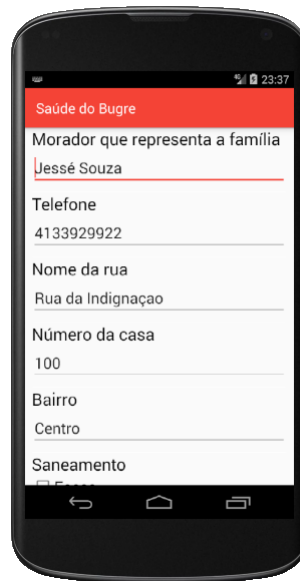
FIGURA 8 - TELA PRINCIPAL



FONTE: O autor (2021).

A tela de cadastro de famílias requer o preenchimento dos campos mostrados na FIGURA 9. Após tocar em “Salvar”, no final da tela, o usuário será direcionado para tela de moradores (FIGURA 10), a qual permite a visualização de seus moradores bem como a adição de novos membros, ao clicar no ícone vermelho que contém o sinal de “+”.

FIGURA 9 - CADASTRO DE FAMÍLIA



FONTE: O autor (2021).

FIGURA 10 - CADASTRO DOS INTEGRANTES DA FAMÍLIA



FONTE: O autor (2021).

Após tocar no ícone vermelho, a tela de cadastro de moradores é mostrada (FIGURA 11). Ao final da tela, há um botão para salvar as informações, retornando à tela anterior.

FIGURA 11 - TELA DE CADASTRO DE MORADOR



FONTE: O autor (2021).

Tela que representa uma família com dois membros (ou moradores) cadastrados (FIGURA 12).

FIGURA 12 - UMA FAMÍLIA COM DOIS MORADORES



FONTE: O autor (2021).

Ao tocar no botão “Voltar”, há a opção de filtrar a lista, representada pelo ícone da “Lupa”. Ao digitar o nome parcial de uma família, todas as famílias que tiverem aquele conjunto de caracteres serão trazidas para o topo da lista (FIGURA 13).

FIGURA 13 - FILTRO DE FAMÍLIAS



FONTE: O autor (2021).

A tela principal contém um menu no canto superior direito, representado pelo ícone com três pontos na vertical. Ao tocá-lo, as opções “Compartilhar”, “Ordenar A-Z”, “Sincronizar visitas”, “Sincronizar famílias”, “Gerenciar usuários” e “Criar lista de visitas” são mostradas, desde que o perfil do usuário seja “ADM” (FIGURA 14). Caso contrário, as opções “Gerenciar usuários” e “Criar lista de visitas” não estarão visíveis.

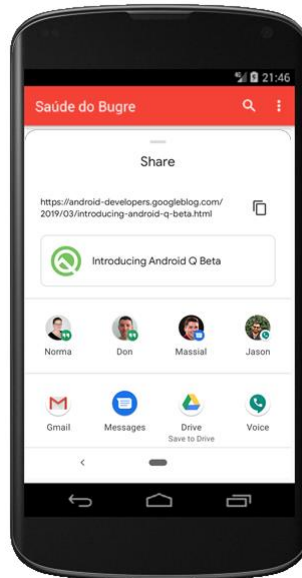
FIGURA 14 - MENU DA TELA PRINCIPAL



FONTE: O autor (2021).

A opção “Compartilhar” gera um arquivo CSV e abre o Android Sharesheet (FIGURA 15), para que o usuário escolha com quem ou com qual aplicativo deseja compartilhar a informação gerada.

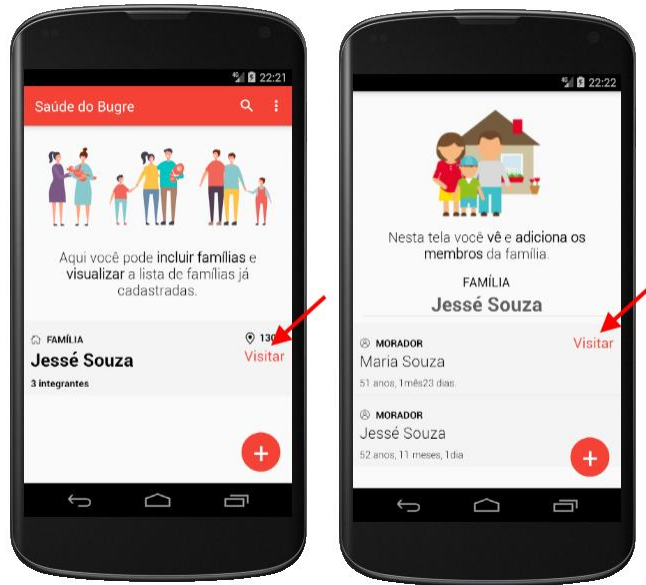
FIGURA 15 - COMPARTILHAR ARQUIVO



FONTE: Adaptado de https://developer.android.com/images/training/sharing/sharesheet_api29.png (2021).

A opção “Ordenar A-Z” simplesmente ordena a lista de famílias em ordem alfabética. A opção “Sincronizar visitas” verifica no servidor se há moradores marcados pelo administrador para serem visitados. Se houver, a família e o morador recebem a marcação “Visitar” (FIGURA 16).

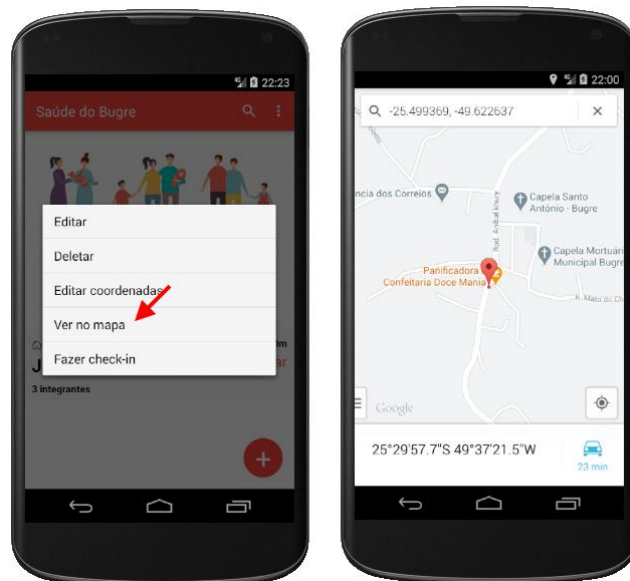
FIGURA 16 - MORADOR MARCADO PARA VISITA



FONTE: O autor (2021).

Neste cenário, onde há a marcação, o usuário pode querer saber a posição geográfica do morador a ser visitado. Para isso, basta tocar na família e segurar até aparecer o menu de opções. Nele, a opção “Ver no mapa” abre o Google Maps com o alfinete no endereço da família (FIGURA 17).

FIGURA 17 - POSIÇÃO DA FAMÍLIA NO MAPA



FONTE: O autor (2021).

A funcionalidade “Sincronizar família” faz o upload de todas as famílias cadastradas para o servidor. Se a sincronização tiver êxito, a mensagem apresentada será “Dados sincronizados” (FIGURA 18).

FIGURA 18 - SINCRONIZAR FAMÍLIAS



FONTE: O autor (2021).

A opção “Gerenciar usuários” permite aos usuários com perfil de administrador aceitar novos usuários, recusar usuários e atribuir um perfil, se administrador ou normal. Para fazer isso, basta um toque longo no usuário mostrado na lista e escolher a opção desejada. Para alterar, segue-se o mesmo fluxo (FIGURA 19).

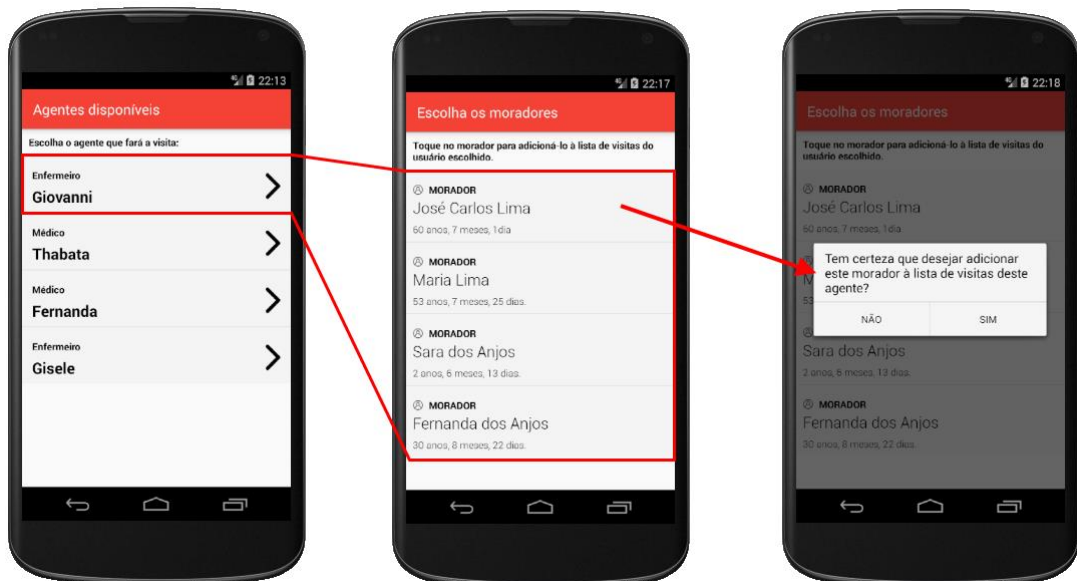
FIGURA 19 - GESTÃO DE USUÁRIOS



FONTE: O autor (2021).

O administrador poderá escolher quais moradores devem ter a visita de um ACS. Neste cenário, primeiro o usuário com perfil administrador escolherá um agente (ACS, médico ou enfermeiro) e, na sequência, marcará os moradores a serem visitados (FIGURA 20).

FIGURA 20 - GERENCIAR VISITAS



FONTE: O autor (2021).

Apesar das demandas pontuais por visitas, as famílias devem ser visitadas periodicamente. Nestes casos, toda vez que uma família for visitada deve-se realizar o “Check-in”, que é a atribuição de um motivo para a visita. Realizando um toque longo na família desejada, a lista de opções é mostrada. Ao escolher “Fazer check-in” uma nova tela é aberta (FIGURA 21). Nela, o usuário escolhe um dos motivos e toca em “Salvar”.

FIGURA 21 - CHECK-IN



FONTE: O autor (2021).

A opção “Editar coordenadas” da família é usada nos casos onde a família foi cadastrada sem ter havido a presença do ACS no local ou para correção da localização. Em ambos os casos, o usuário poderá pegar a localização atual se o usuário estiver na casa do morador ou poderá editar os campos de latitude e longitude (FIGURA 22).

FIGURA 22 - EDITAR COORDENADAS



FONTE: O autor (2021).

4.4 CONSIDERAÇÕES

Neste capítulo foi apresentado a arquitetura que norteou o desenvolvimento dos sistemas. O sistema mobile persiste os dados localmente e pode persisti-los no servidor. Para usar os artefatos deste projeto, o usuário precisa ter um smartphone com o sistema operacional Android e ter um serviço de internet disponível para persistir os dados no servidor, se desejar. As telas do aplicativo foram apresentadas em detalhes, simulando a maioria das funcionalidades descritas na análise de requisitos. O próximo e último capítulo trata das considerações e das possibilidades de melhorias deste projeto.

5 CONSIDERAÇÕES FINAIS

A oportunidade de pôr em prática conhecimentos adquiridos ao longo do curso com um cliente real ajuda o estudante a vislumbrar o que o mercado exige. A importância do planejamento, das etapas de desenvolvimento, da Engenharia de Software e do nível de domínio técnico sobre a linguagem de programação ficam evidentes. O aperfeiçoamento pessoal deve ser contínuo para produzir aplicações o mais úteis possíveis para a realidade de cada cliente.

Todos os objetivos específicos foram trabalhados e apresentados em forma de funcionalidades. Mesmo a sincronização de dados, que ofereceu dificuldades maiores, foi parcialmente atendida. Uma pesquisa mais refinada e mais dispendiosa temporalmente poderá encontrar um padrão de projeto mais adequado à situação do cliente.

Os prazos estabelecidos no planejamento não puderam ser seguidos, tendo o projeto atrasado em 4 meses. Tanto a desistência de um membro da equipe quanto ao agravamento da pandemia de Covid-19 exigiram mudanças consideráveis na rotina do autor, afetando o ritmo das entregas.

Devido às restrições burocráticas da UBS, não foi possível até a presente data a operacionalização do servidor. Entretanto, a equipe de saúde está usando o aplicativo, coletando e atualizando dados diariamente e enviando-os no formato CSV para o gestor da UBS, o qual consolida os dados e gera informações.

Em um mês de funcionamento, foram coletadas informações de 892 pessoas da comunidade local. Dentre outras possibilidades, pela primeira vez foi possível plotar no mapa a localização de cada morador, contribuindo para o delineamento dos limites de atuação de cada ACS e da própria UBS dentro da geografia da cidade. Além disso, pode-se observar em um mapa regiões onde há maior incidência de falta de saneamento básico, em que região concentra-se o maior número de fumantes ou aonde moram os pacientes acamados, contribuindo com informações relevantes para a tomada de decisão dos gestores públicos, para o foco de campanhas de promoção de saúde e para a logística do deslocamento da equipe médica, respectivamente.

Além deste trabalho contribuir para a aquisição de experiência e conhecimento técnico do autor, a viabilização da coleta dos dados pela UBS subsidiou o trabalho de conclusão de curso de pós-graduação de outro estudante da UFPR, a

médica cliente deste projeto. Isso ajuda a devolver para a sociedade o investimento público e aproxima a comunidade da universidade.

5.1 RECOMENDAÇÕES PARA TRABALHOS FUTUROS

O questionário de saúde neste momento é fixo, sendo incluído diretamente no banco de dados. Como a demanda de dados epidemiológicos pode mudar ao longo do tempo, pode ser útil adicionar uma funcionalidade que permita ao administrador adicionar ou excluir perguntas do questionário.

Outra possibilidade de melhoria consiste em transformar a lista de visitas em uma rota de visitas, otimizando os custos de deslocamento até o paciente.

Estudos mais aprofundados sobre o uso de programação paralela no Android possibilitaria um método melhorado de sincronização com o servidor, evitando ter que sincronizar visitas e famílias separadamente. Esse conhecimento contribui para tornar o aplicativo independente de rede móvel durante a coleta de dados, não necessitando de rede para atualizar a família visitada.

Uma interface *web* poderia ser útil para o coordenador das equipes de ACS, possibilitando visualizar no mapa as famílias atendidas, moradores acamados, área de atuação de cada ACS, entre outras funcionalidade que remetam a relatórios e supervisão de equipe. Atualmente isso é feito usando as ferramentas do Google, após o gestor reunir os dados enviados pelos ACS.

REFERÊNCIAS BIBLIOGRÁFICAS

AGENTE COMUNITÁRIO DE SAÚDE. Disponível em:

<https://play.google.com/store/apps/details?id=com.fgmaiss.acs>. Acesso em: 18 jun. 2020.

C S R PRESIDENTE TANCREDO NEVES. Disponível em:

<http://www.ubsbrasil.org/sobre/c-s-r-presidente-tancredo-neves>. Acesso em: 01 jul. 2020.

CIDADE SAUDÁVEL ACS. Disponível em:

<https://play.google.com/store/apps/details?id=com.sysvale.syssus>. Acesso em: 18 jun. 2020.

DESENVOLVIMENTO ÁGIL. Disponível em:

<https://www.desenvolvimentoagil.com.br/scrum/>. Acesso em: 01 jul. 2020.

DEVMEDIA. Disponível em: <https://www.devmedia.com.br/introducao-a-tecnologia-java/25899>. Acesso em: 07 abr. 2021.

E-SUS TERRITÓRIO. Disponível em:

<https://play.google.com/store/apps/details?id=br.gov.saude.acs>. Acesso em: 18 jun. 2020.

GUIMARÃES, G. Disponível em:

http://www.dsc.ufcg.edu.br/~pet/jornal/agosto2013/materias/historia_da_computacao.html#:~:text=O%20projeto%20Android%20est%C3%A1%20ligado,telefonia%20e%20fabricantes%20de%20dispositivos.&text=O%20primeiro%20aparelho%20com%20o,de%202008%20nos%20Estados%20Unidos. Acesso em: 07 abr. 2021.

MINISTÉRIO DA SAÚDE. Disponível em: <http://www.saude.gov.br/sistema-unico-de-saude>. Acesso em: 20 jul. 2020.

ORACLE. Disponível em: <https://www.oracle.com/br/java/>. Acesso em: 07 abr. 2021.

SECRETÁRIA DE ATENÇÃO PRIMÁRIA À SAÚDE (SAPS). Disponível em:

<https://aps.saude.gov.br/ape/esus>. Acesso em: 15 jul. 2020.

SOARES, S.S.K.P. Modelos para avaliação de Materiais Didáticos. Apostila elaborada para a disciplina de Práticas em Docência do Programa de Pós-Graduação em Bioinformática da Universidade Federal do Paraná. Curitiba, 2020.

TADS. Disponível em: <https://www.tads.ufpr.br/mod/page/view.php?id=8>. Acesso em: 20 dez. 2020.

TRELLO. Disponível em: <https://help.trello.com/article/708-what-is-trello>. Acesso em: 07 abr. 2021.

APÊNDICE 1 – TENTATIVA DE CONTATO COM E-SUS TERRITÓRIO

Repositório e-SUS Território

Giovanni Pissaia <giovanni@ufpr.br>

Qui, 28/05/2020 08:53

Para: esusab.nti@saude.gov.br <esusab.nti@saude.gov.br>

Cc: Leonides Ribeiro Neto <leonides.neto@ufpr.br>

Bom dia

Sou estudante da UFPR, do curso de Análise e Desenvolvimento de Sistemas, e conheci o sistema e-SUS Território através da Unidade de Saúde de Balsa Nova/PR.

Lá, eles apontaram algumas funcionalidades que poderiam ser acrescentadas ao aplicativo para torná-lo mais adequado às necessidades locais, como a possibilidade de registrar as coordenadas do local da moradia da família (como é uma cidade rural, muitos endereços não tem CEP, número, etc, dificultando a localização posterior) e a possibilidade de o gestor montar uma rota de visitas para os agentes de campo, entre outras customizações.

Tendo em vista a oportunidade acima, bem como a conclusão da minha graduação, a qual requer a realização de um Trabalho de Conclusão de Curso, iniciei uma conversa com o meu orientador e ele sugeriu que trabalhássemos implementando essas customizações em uma branch do projeto do e-SUS Território, com finalidades educacionais.

O código do sistema está disponível em algum repositório para acesso público?

Att,
Giovanni.

1shQoT5Un+6pywPS/nLnj1nCGpR1/i6CCykZW7VBSkSn9d1AwKqUtG/4b2icJ6fg1ZvpPKK9jI8m44/F4Q1dYP32t9yrAjFbX
Kc40AG6NylKPa+Qcd754PWP5YUQBfYf+/D+v0SufaHQYz4Q==
ARC-Authentication-Results: i=1; mx.microsoft.com 1; spf=pass
smtp.mailfrom=ufpr.br; dmarc=pass action=none header.from=ufpr.br; dkim=pass
header.d=ufpr.br; arc=none
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed;
d=ufprbr0.onmicrosoft.com; s=selector2-ufprbr0-onmicrosoft-com;
h=From:Date:Subject:Message-ID:Content-Type:MIME-Version:X-MS-Exchange-SenderADCheck;
bh=Mw/OR0/1CnhZG29Zjd24DHdLc7rhK4I/hj3DjeIToYI=;

b=W+Gyia8NSdRIYwMk8zweoRkJkhp2yt9T+rhAFI+QguZb5Jn3inZ7q4YFJWU0njbIPIOF8DDAPuQQj6DKCm7+tPPYv7C7HhXoU
eafb3Cy15CY0rb1Lh+SJzN7fEr6IPL6UvvAJ/bDQhLizrTS3aCrbpcIpdI7w/g71sH2W8yAsW8=
Received: from FR1P152MB0534.LAMP152.PROD.OUTLOOK.COM (2603:10d6:200:16::17)
by FR1P152MB3020.LAMP152.PROD.OUTLOOK.COM (2603:10d6:203:22::11) with
Microsoft SMTP Server (version=TLS1_2,
cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384) id 15.20.3045.19; Thu, 28 May
2020 11:53:08 +0000
Received: from FR1P152MB0534.LAMP152.PROD.OUTLOOK.COM
([fe80::6c45:ba4b:713b:9820]) by FR1P152MB0534.LAMP152.PROD.OUTLOOK.COM
([fe80::6c45:ba4b:713b:9820%12]) with mapi id 15.20.3045.018; Thu, 28 May
2020 11:53:08 +0000
From: Giovanni Pissaiá <giovanni@ufpr.br>
To: "esusab.nti@saude.gov.br" <esusab.nti@saude.gov.br>
CC: Leonides Ribeiro Neto <leonides.neto@ufpr.br>
Subject: =?iso-8859-1?Q?Reposit=F3rio_e-SUS_Territ=F3rio?=
Thread-Topic: =?iso-8859-1?Q?Reposit=F3rio_e-SUS_Territ=F3rio?=
Thread-Index: AQHWN0ZwUMB3da6Uj0KY2+nFMdsOqw==
Date: Thu, 28 May 2020 11:53:08 +0000
Message-ID: <FR1P152MB05340E23EF7783AA0C6FD266CA8E0@FR1P152MB0534.LAMP152.PROD.OUTLOOK.COM>
Accept-Language: en-US
Content-Language: pt-BR
X-MS-Has-Attach:
X-MS-TNEF-Correlator:
x-originating-ip: [200.175.195.223]
x-ms-publictraffictype: Email
x-ms-office365-filtering-correlation-id: 5917df1b-90ee-4d35-cbd9-08d802fdb09c
x-ms-trafficdiagnostics: FR1P152MB3020:
x-ms-exchange-transport-forked: True
x-microsoft-antispam-prvs:
<FR1P152MB3020A2FFAC4EE8A1ED599804CA8E0@FR1P152MB3020.LAMP152.PROD.OUTLOOK.COM>
x-ms-oob-tlc-oobclassifiers: OLM:1284;
x-forefront-prvs: 0417A3FFD2
x-ms-exchange-senderadcheck: 1
x-microsoft-antispam: BCL:0;
x-microsoft-antispam-message-info:
MgWG/s1Y/65NLqnxWiBbjkmh8zLVGnQQCX07dFQbcimuBnWxURG1dA/RfBMcMHVoa1PH5dRRodoESBryL4I6S3rSjuj1t32KD
/izzuI6PEXDRcx7at9FDdp8PF2tcnN2BqYubAHS0SiSUjY1k4Hw1p/VnzmI66tk0Iu9Z5ZqH0t0XozS0EULqYS5a78nP2xG4Pdq
0Jh2KatYap1e2pwX6vCn9AIVxDwblxHy7qxxLZGRVIFjPtigjy13/hGz+s3SKqik0toyD3tKqtTj7w5dkzqd8xKmb6dFC5S1ZRa
qZRmpKyZRkdndj3/fUKQhIw5UbyMChuK7Z4AG69Y/jNag==
x-forefront-antispam-report:
CIP:255.255.255.255;CTRY:;LANG:pt;SCL:1;SRV:;IPV:NLI;SFV:NSPM;H:FR1P152MB0534.LAMP152.PROD.OUTLOOK.
COM;PTR:;CAT:NONE;SFTY:;SFS:(136003)(376002)(396003)(39850400004)(366004)(346002)(8936002)
(566030002)(52536014)(107886003)(6916009)(55016002)(9686003)(478600001)(71200400001)(86362001)
(4326008)(316002)(786003)(186003)(26005)(33656002)(7696005)(6506007)(224303003)(2906002)(66476007)
(91956017)(76116006)(66946007)(66446008)(66556008)(64756008);DIR:OUT;SFP:1101;
x-ms-exchange-antispam-messagedata:
CeUh5LY81MA1Th5zLiwULPftgcQuA61FiQEbnBmNRDv5FbaU6/FTEIh6Lk5BSfXtsX+0S/rHL6n3Pdown8svCD5Lz4mUZdbbfQ
z6wUZDt/817/mBIBA3yAthXNXREvqPFenJP+cIFkoBec5fhSgeZhcxcv13PumAc55FkEmw8F8X5gWqbf1Mj7i6B1SrH/NrydcL4
W3wa7eZ6CbcjCVYRctHDPhebDX4VLLDIwtDwWjXd6MzYlnX1EAKk913gTz6VUcstlneFgbUFRtI/nqKvSvZ2bpztVEZNo3NnE3
K8AD0PzPaxodfS/g1IOwezj2kdKsH3uE62u1aU7MV9CM6IuZXYeA0V1LCY0X9qeTeFAFV1sJEVRM+oaQ178I0+KpV/fnB/kaLR
hPK+krPIYmSWPXe0ZD5n/zw43yPf3u+uUvoL+6s86wY0outa7t2dwMMv1Qw94WA2w4Vzu9UV7pY0f17ymeDXsEqAG2Cg7vPjeZ
Ea85yi2r1zIEpbf4e
Content-Type: multipart/alternative;
boundary=" _000_FR1P152MB05340E23EF7783AA0C6FD266CA8E0FR1P152MB0534LAMP_ "
MIME-Version: 1.0
X-MS-Exchange-CrossTenant-Network-Message-Id: 5917df1b-90ee-4d35-cbd9-08d802fdb09c
X-MS-Exchange-CrossTenant-originalarrivaltime: 28 May 2020 11:53:08.3723
(UTC)
X-MS-Exchange-CrossTenant-fromentityheader: Hosted
X-MS-Exchange-CrossTenant-id: c37b37a3-e9e2-42f9-bc67-4b9b738e1df0
X-MS-Exchange-CrossTenant-mailboxtype: HOSTED
X-MS-Exchange-CrossTenant-userprincipalname:
1hB0AJ+ki5FJea0KgyFiX/wiGtKuYziSZQbnvaevrkwJPHJisrBpBmn15HK0oZyGyGxuvW+1sPsbET8tRLlW==
X-MS-Exchange-Transport-CrossTenantHeadersStamped: FR1P152MB3020

Authentication-Results: symauth.service.identifiier; spf=pass; senderid=pass; dkim=pass
header.d=ufprbr0.onmicrosoft.com header.s=selector2-ufprbr0-onmicrosoft-com header.v=1
header.a=rsa-sha256

X-Brightmail-Tracker: H4sIAAAAAAAAAA11UfUwTzxmVbu2R+X00Cj8qKCs4sYQy3QuqFmcMw7RbMvc/pizibhz3Nra

QrHXaifBwbpMLyKK4seMXwhzCp1TRGFwxywiA0Fcp07UIAImfG06Z2DGWPZe74p1/7x53ud5
7/d73t/75GiSa4jS04LbKThyeZtBraVsre99MHPp0fas1/69mDH3wtNG1UK0p0qZuAyt0L6Z
Ldgs6wRHxoJPteahprNk3v5kd9+QtRB1TN6CImlg58C+gV9UW5CW5tg+BJerD1HSBrFHSag4
ckctb1oo2D04i5A3xxH0HrpJShuKPU1Cz9nDlFyGjAB/Y0Cpdh9B8SdeJLVrs69CR/1v1IRj
2UwYqCrW5JhkZ0H5w1G1hGPY+bc516uS27wFj38qRzI2QnmR15QwxU6HuuHRIM+wK6Hv4PXg
ectGwUilH5BrxkN5zXFCvh4LFeFbSRnr0L8noJLxNLjTXK0cSQL/oSik4/fhVtNT5UwaVLU9
xD5pJfMg4MuXaRsUjXyhmX4ZSr6fJ90J0FD/Rkk4B8Sq33qdCFNet2uAYgX10QGfXVY28GSsh
axspsR2l7wuzLWM7X0lsCGKgjYbmb3spmTfCrV2lahnPgKNlg6SMZ8LegI8K5w8jTSvic9yz
jSLvyhaMJvs642pHNckBuZD4FVGHKm8u9yGwRoYohT3Ynswp+HVi fo4PWmCoG0qkjE1cbU9
09/Mi+ZVDpdNEA2xzIq5mGbG6NUum9WgZ4bnYTzmjM0V1os2wYkTapjCHPu5NYuLH9NE15hn
+cxid4mrXA6bDwFN4rKl2q7gstl8foHgsMvNfGgyTRnimd2L67M41sQ7Basg5AmOkErQgh9a
T9MGYD6aj9tH0wST4P7cYnOGzuCP3R1YYcOV0OMk5tyjtWuLlWIM/0Ss1zEsj5c/r9vgo6U
Fq0PmegofANrpjQYMY/PES0mpX8M45mB2agQG+ydwAXKdrkQGdY3iZ1wCQ8rLiSN79mCDiB6
e/+BiYR9Z30Zt06pwOvflDI6Iq0clwvPFftxTKP0HqxUxuzKHZuHPo7xXMP3mhQmSjB0iYz1
Me6rC+NfuNIInMxut+HUSwtTxxkL/ID9K0scwKCIigovCD5VjcSpzUPRuFE3/qeHUwILPyS7W
B800cICRzCGuSQNJcqs63wxPo4ZScPkBIUMTg+YZCHHvriw4Suyixq15yJkeIuzv0P4H9Gj
ghM72kjob01Vwej+Bg3sLPHQ8PzY6CRoHTkXA7/6R2KheJNPBx3fXIiDzk2VCVB96qYe/D+c
mQpP2jZMh1OHT74CNV1PUuFgoDYd6puuG2HHw2r8G/Pfex1avZ5M+L2uZh7cvnt5IfzjGv4C
56/tXAqd1wo/hL7C65/A5qGqFXCyv4iHhsA2HprPe3g46w0T4K/b/jU0E0ET1NXxVUPTU7e
GZ6m2qZWKU0Kq6TptERYIXJcms40Bd0kS0Mnoi9EFSkPbt/1boicE9nTpbuX/M7X06offeU
nL1aMGsqvbtTq31acTHRHcuvvFegsEzLFagcE5957dFo+I3I1Jvdt/9oL4tt+JjLmBjQDcwf
TjjRb3Sv9U5NM9ffiHDVLTbLXk491r281X1Qpym4Ef1j3IMajnu725S+tbQi5YvhrItfqq2U
a0ZnpZEokf8PU+8zpNYGAAA=

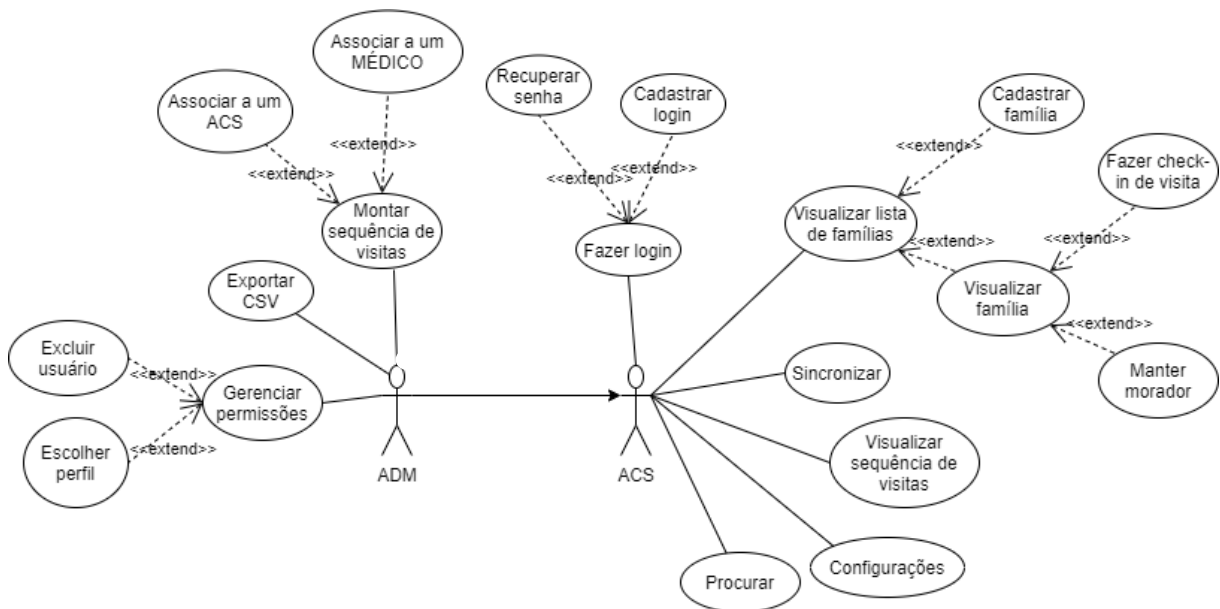
Return-Path: giovanni@ufpr.br

Esta mensagem pode conter informação confidencial e/ou privilegiada. Se você não for o destinatário ou a pessoa autorizada a receber esta mensagem, não pode usar, copiar ou divulgar as informações nela contidas ou tomar qualquer ação baseada nessas informações. Se você recebeu esta mensagem por engano, por favor avise imediatamente o remetente, respondendo o e-mail e em seguida apague-o.

APÊNDICE 2 – DIAGRAMA DE CASOS DE USO

A FIGURA 23 retrata os atores principais, as funcionalidades atribuídas e as relações entre si. Os atores iniciais eram médicos, enfermeiros e agentes comunitários de saúde, mas uma reanálise do negócio estabeleceu os atores administrador (ADM) e agente comunitário de saúde (ACS), unificando médicos e enfermeiros como administradores. A descrição completa de cada caso de uso está resumida na TABELA 3.

FIGURA 23 - DIAGRAMA DE CASOS DE USO



FONTE: O autor (2021).

TABELA 3 – DESCRIÇÃO DOS ATORES E CASOS DE USO

ATOR	DESCRIÇÃO
ADM	É o ator que administra o sistema e que representa usuários médicos ou enfermeiros.
ACS	É o ator que representa o Agente Comunitário de Saúde, o qual tem restrições em relação ao sistema.

CASO DE USO	DESCRIÇÃO
Fazer login	Permite que o usuário acesse o sistema.
Cadastrar login	Este caso de uso oferece ao usuário um meio para o auto cadastro.
Recuperar senha	Permite ao usuário recuperar a senha.
Visualizar lista de famílias	Mostra ao usuário a lista de famílias por ele cadastradas.
Cadastrar famílias	Permite cadastrar novas famílias.
Visualizar família	Mostra a lista de moradores da família selecionada.
Fazer check-in de visita	Permite ao usuário indicar um motivo de visita ao visitar uma família.
Manter morador	É o caso de uso que permite adicionar, atualizar, visualizar e excluir moradores.
Sincronizar	Oferece ao usuário a possibilidade de sincronizar os dados com o servidor.
Visualizar sequência de visitas	Permite ao usuário visualizar a lista de visitas feita pelo administrador.
Procurar	Permite ao usuário procurar uma família.
Montar sequência de visitas	Permite ao administrador montar uma sequência de visitas para um ACS.
Associar a um médico	Permite associar um médico ao morador que se pretende visitar
Associar a um ACS	Permite associar um ACS ao morador que se pretende visitar
Exportar CSV	Possibilita exportar os dados no formato CSV.
Gerenciar permissões	É o caso de uso que permite o gerenciamento das solicitações de acesso ao sistema.
Excluir usuário	Permite ao administrador excluir um usuário.
Escolher perfil	Permite ao administrador associar um usuário a um perfil.

FONTE: O autor (2021).

APÊNDICE 3 – ESPECIFICAÇÃO DE CASO DE USO

RF01 – Cadastrar usuários

Descrição

Este caso de uso serve para cadastrar usuários.

Data View

DV1 - Tela de cadastro de usuários.

FIGURA 24 - TELA DE CADASTRO DE USUÁRIOS

O diagrama mostra uma interface de usuário para o cadastro de usuários. O formulário é dividido em seções por campos de entrada:

- CADASTRO** (título da seção)
- Nome completo**: campo de texto com o valor "Nome".
- Profissão**: campo de seleção com o valor "Escolha" e uma seta para baixo.
- E-mail**: campo de texto com o valor "seuemail@qualquer.com".
- Senha**: campo de texto com o valor "*****".
- Repetir senha**: campo de texto com o valor "*****".
- CADASTRAR** (botão de ação).

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega o combo com as profissões
2. O sistema apresenta a tela (DV1)
3. O usuário insere o nome
4. O usuário escolhe a profissão
5. O usuário insere o e-mail
6. O usuário insere a senha
7. O usuário insere a confirmação de senha
8. O usuário pressiona o botão Cadastrar (E1)
9. O sistema mostra a mensagem “Cadastro realizado”
10. O caso de uso é encerrado.

Fluxos Alternativos

Nenhum.

Fluxos de Exceção

E1. Campos obrigatórios

1. O sistema emite a mensagem “Todos os campos são obrigatórios”

RF02 – Fazer login

Descrição

Este caso de uso serve para fazer login.

Data View

DV2 - Tela de login

FIGURA 25 - TELA DE LOGIN

LOGIN

E-mail
seuemail@qualquer.com

Senha

[Esqueci a senha](#)
[Não tenho cadastro](#)

ENTRAR

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV2)
2. O usuário insere o e-mail
3. O usuário insere a senha
4. O usuário pressiona o botão Entrar (E1)(A1)(A2)
5. O caso de uso é encerrado.

Fluxos Alternativos

A1. Link 'Esqueci a senha' pressionado

1. O sistema carrega a tela RF04 – Recuperar senha

A2. Link 'Não tenho cadastro' pressionado

1. O sistema carrega a tela RF01 – Cadastrar usuários

Fluxos de Exceção

E1. E-mail ou senha incorretos

1. O sistema verifica os dados no banco de dados
2. O sistema emite a mensagem “Usuário ou senha incorretos”

RF04 – Recuperar senha

Descrição

Este caso de uso serve para recuperar senha.

Data View

DV4 - Tela de Recuperação de senha

FIGURA 26 - TELA DE RECUPERAÇÃO DE SENHA

O diagrama mostra a interface de usuário para a recuperação de senha. No topo, há uma barra rotulada 'LOGIN'. Abaixo, o campo 'E-mail' contém o endereço 'seuemail@qualquer.com'. No rodapé, há um botão rotulado 'ENVIAR'.

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV4)
2. O usuário insere o e-mail
3. O usuário pressiona o botão Entrar (E1)

4. O sistema carrega a tela RF02 – Fazer login
5. O sistema mostra a mensagem “Senha enviada por e-mail”
6. O caso de uso é encerrado.

Fluxos de Exceção

E1. Campo obrigatório

1. O sistema verifica que o campo é obrigatório
2. O sistema emite a mensagem “O e-mail deve ser informado”

RF05 – Cadastrar famílias

Descrição

Este caso de uso serve para cadastrar de famílias.

Data View

DV5 - Tela de Cadastro de famílias

FIGURA 27 - TELA DE CADASTRO DE FAMÍLIAS

FAMÍLIA

Nome da família

Gravar localização

latitude; longitude

< demais perguntas serão carregadas dinamicamente, como questões de saneamento, vizinho de plantação, etc >

SALVAR

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega as perguntas cadastradas no banco de dados
2. O sistema apresenta a tela (DV5)
3. O usuário insere o nome da família
4. O usuário pressiona o botão de localização
5. O sistema preenche as coordenadas geográficas
6. O usuário preenche o questionário
7. O usuário pressiona o botão Salvar (E1)
8. O sistema carrega a tela RF06 – Visualizar famílias cadastradas
9. O sistema mostra a mensagem “Família cadastrada”
10. O caso de uso é encerrado.

Fluxos de Exceção

E1. Nome da família é obrigatório

1. O sistema verifica que o um nome de família é obrigatório
2. O sistema emite a mensagem “O nome da família é obrigatório”

RF06 – Visualizar famílias cadastradas

Descrição

Este caso de uso serve para visualizar as famílias cadastradas.

Data View

DV6 - Tela de Visualização das famílias cadastradas

FIGURA 28 - TELA DE FAMÍLIAS CADASTRADAS



FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega as famílias cadastradas
2. O sistema apresenta a tela (DV6)
3. O usuário pressiona o botão Adicionar nova família (A1) (A2) (A3) (A4) (A5) (A6)
4. O sistema carrega a tela RF05 – Cadastrar famílias
5. O caso de uso é encerrado

Fluxos Alternativos

A1. Botão “Procurar” pressionado

1. O sistema carrega a tela RF10 – Pesquisa

A2. Botão “Sincronizar” pressionado

2. O sistema sincroniza os dados locais com o banco de dados remoto

A3. Botão “Lista de visitas” pressionado

3. O sistema carrega a tela RF11 – Criar agenda de visitas

A4. Botão “Usuários” pressionado

4. O sistema carrega a tela RF15 – Gerenciar usuários

A1. Botão “Exportar CSV” pressionado

5. O sistema carrega a tela RF16 – Exportar CSV

A1. Botão “Configurações” pressionado

6. O sistema carrega a tela RF17 – Configurações

RF07 – Cadastrar moradores

Descrição

Este caso de uso serve para cadastrar moradores

Data View

DV7 - Tela de Cadastro de moradores

FIGURA 29 - TELA DE CADASTRO DE MORADORES

MORADOR

Informações pessoais

Nome
Raquel Cavali

Data de nascimento
01 03 1967

Telefone
(41) 99155 3598

Estado Civil
 Solteiro Casado

< as demais perguntas serão carregadas dinamicamente >

SALVAR

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega as perguntas cadastradas no banco de dados
2. O sistema apresenta a tela (DV7)
3. O usuário insere o nome
4. O usuário insere a data de nascimento

5. O usuário insere o telefone
6. O usuário preenche o questionário
7. O usuário pressiona o botão Salvar (E1) (E2) (A1)
8. O sistema carrega volta para a tela anterior
9. O sistema mostra a mensagem “Morador cadastrado”
10. O caso de uso é encerrado.

Fluxos Alternativos

A1. Botão “Editar” pressionado

1. O sistema permite a edição do morador cadastrado

Fluxos de Exceção

E1. Nome do morador é obrigatório

1. O sistema verifica que o nome do morador é obrigatório
2. O sistema emite a mensagem “Nome do morador é obrigatório”

E2. Data de nascimento do morador é obrigatório

1. O sistema verifica que a data de nascimento do morador é obrigatória
2. O sistema emite a mensagem “Data de nascimento do morador é obrigatória”

RF08 – Visualizar moradores cadastrados

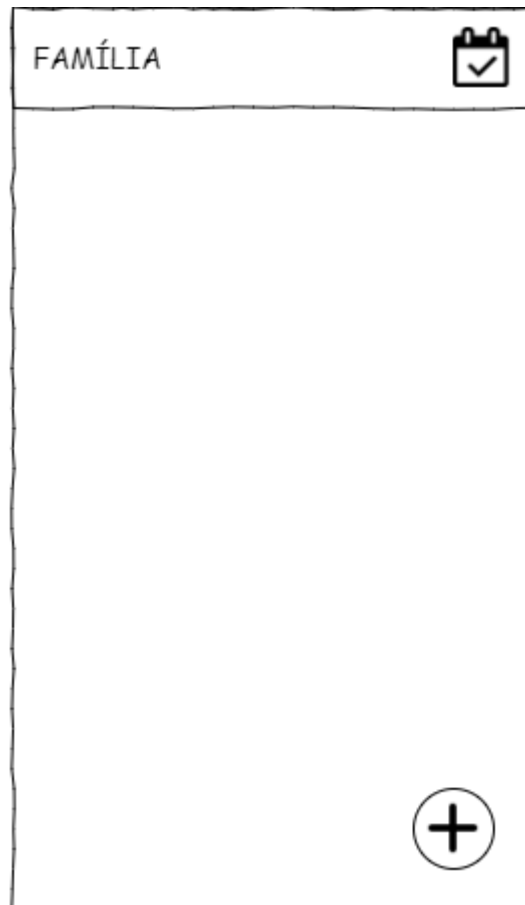
Descrição

Este caso de uso serve para visualizar os moradores cadastrados.

Data View

DV8 - Visualizar moradores cadastrados

FIGURA 30 - TELA DE VISUALIZAÇÃO DE MORADORES



FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega os moradores cadastrados
2. O sistema apresenta a tela (DV8)
3. O usuário pressiona o botão Adicionar novo morador (A1)
4. O sistema carrega a tela RF07 – Cadastrar moradores
5. O caso de uso é encerrado

Fluxos Alternativos

A1. Botão “Check-in” pressionado

1. O sistema carrega a tela RF07 – Fazer check-in

Fluxos de Exceção

E1. CPF Inválido

1. O sistema verifica que o CPF é inválido (R1)
2. O sistema emite a mensagem “CPF Inválido”

Regras de Negócio

R1. O sistema

RF09 – Prontuário do paciente

Descrição

Este caso de uso serve para registrar prontuário

Data View

DV9 - Tela de Prontuário

FIGURA 31 - TELA DE PRONTUÁRIO DO PACIENTE

PRONTUÁRIO

Anotações

SALVAR

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV9)
2. O usuário insere as anotações
3. O usuário pressiona o botão Salvar
4. O sistema volta para a tela anterior
5. O caso de uso é encerrado

RF10 – Pesquisa

Descrição

Este caso de uso serve para realizar pesquisa.

Data View

DV10 - Tela de Pesquisa

FIGURA 32 - TELA DE PESQUISA

PROCURAR

Procurar por:

Família Morador

Digite o nome:

BUSCAR

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV10)
2. O usuário seleciona "Família"
3. O usuário digita o nome da família
4. O usuário pressiona o botão Buscar (E1) (A1)
5. O sistema carrega a tela RF06 – Visualizar famílias cadastradas
6. O caso de uso é encerrado

Fluxos Alternativos

A1. Opção “Morador” selecionada

1. O sistema carrega a tela RF07 – Cadastrar moradores

Fluxos de Exceção

E1. Pesquisa inválida

1. O sistema verifica que o termo de busca não retornou nenhum resultado
2. O sistema emite a mensagem “Não encontrado”

RF11 – Visualizar agenda de visitas

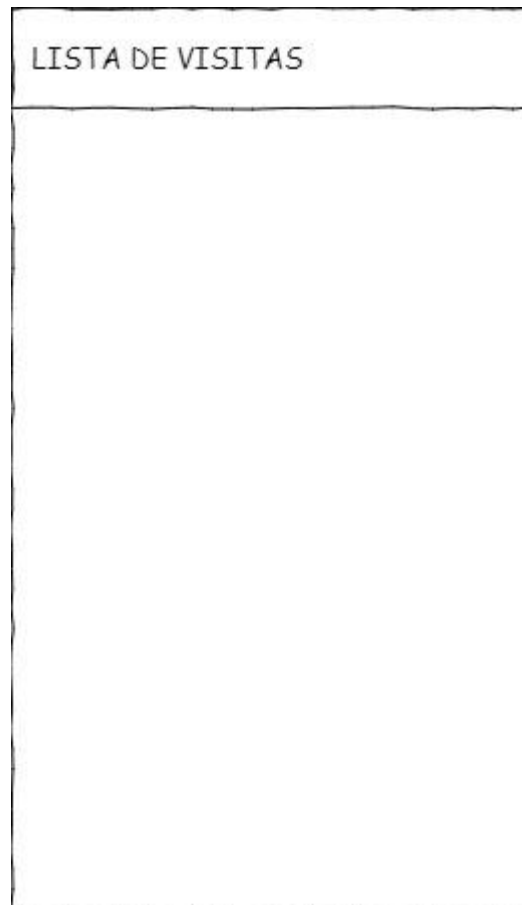
Descrição

Este caso de uso serve para visualização agenda de visitas

Data View

DV11 - Tela de Visualização agenda de visitas

FIGURA 33 - TELA DE VISUALIZAÇÃO AGENDA DE VISITAS



FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega a lista de moradores
2. O sistema apresenta a tela (DV11)

RF12 – Check-in de visitas

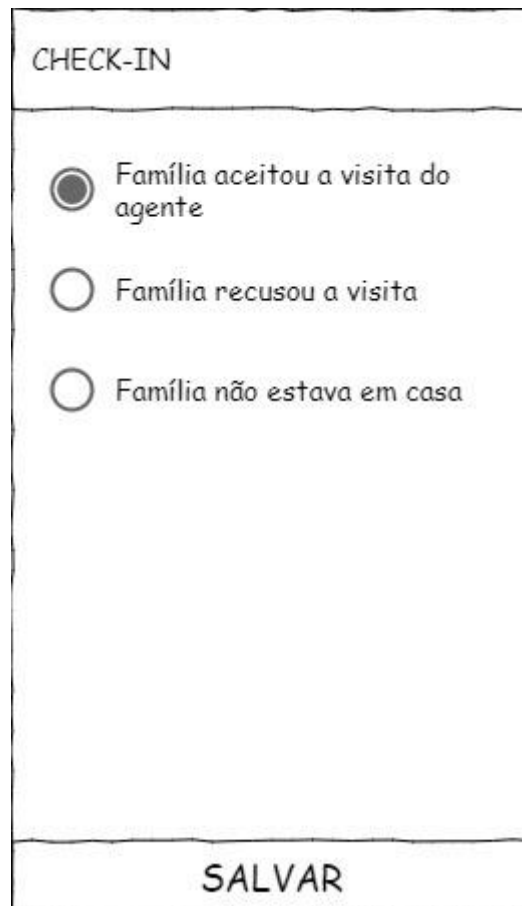
Descrição

Este caso de uso serve para fazer check-in de visitas

Data View

DV12 - Tela de Check-in de visitas

FIGURA 34 - TELA DE CHECK-IN DE VISITAS



A interface de usuário para o check-in de visitas, apresentada em um formato de tela vertical. No topo, há um cabeçalho com o texto "CHECK-IN". Abaixo dele, há três opções de seleção, cada uma com um botão de rádio à esquerda e o texto da opção à direita. A primeira opção, "Família aceitou a visita do agente", está selecionada, com o botão de rádio preenchido. As outras duas opções, "Família recusou a visita" e "Família não estava em casa", não estão selecionadas, com botões de rádio vazios. Na base da tela, há um botão retangular com o texto "SALVAR" em letras maiúsculas.

FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema apresenta a tela (DV12)
2. O usuário seleciona a opção desejada (E1)
3. O usuário pressiona Salvar
4. O caso de uso é encerrado
5. O usuário volta para tela anterior

Fluxos de Exceção

- E1. Selecionar uma das opções

1. O sistema verifica que nenhuma opção foi selecionada
2. O sistema emite a mensagem “Selecione um motivo”

RF13 – Gerenciar usuários

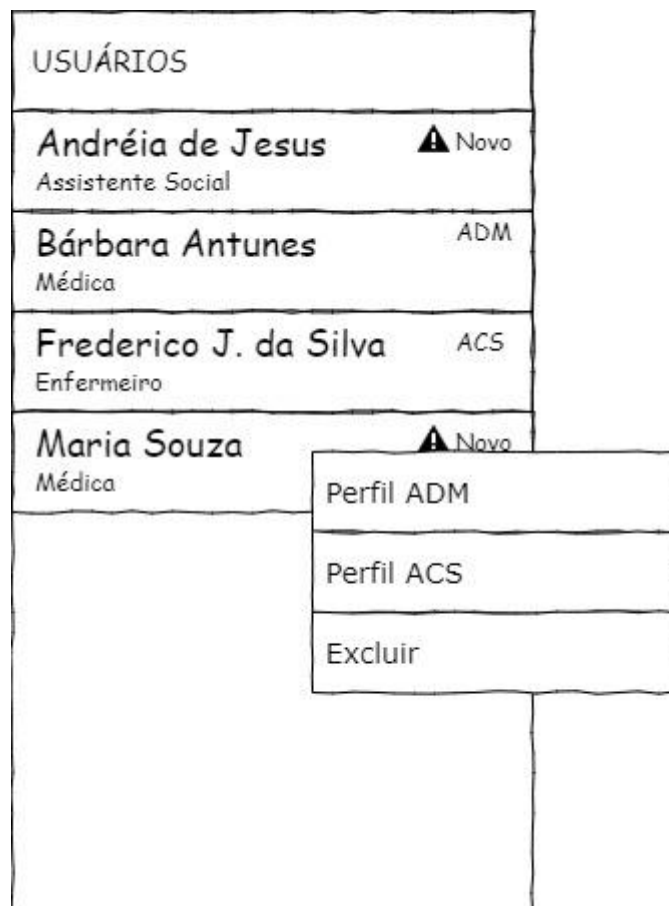
Descrição

Este caso de uso serve para gerenciar usuários

Data View

DV13 - Tela de Gerenciar usuários

FIGURA 35 - TELA DE GERENCIAR USUÁRIOS



FONTE: O autor (2021).

Ator Primário

Usuário

Fluxo de Eventos Principal

1. O sistema carrega os usuários cadastrados
2. O sistema apresenta a tela (DV13)
3. O usuário seleciona um usuário cadastrado
4. O usuário pressiona a opção “Perfil ADM” (A1) (A2)
5. O sistema grava a opção Perfil ADM
6. O sistema recarrega os usuários cadastrados

Fluxos Alternativos

A1. Opção “Perfil ACS” pressionado

1. O sistema grava a opção selecionada

A2. Opção “Excluir” pressionado

1. O sistema exclui o usuário selecionado

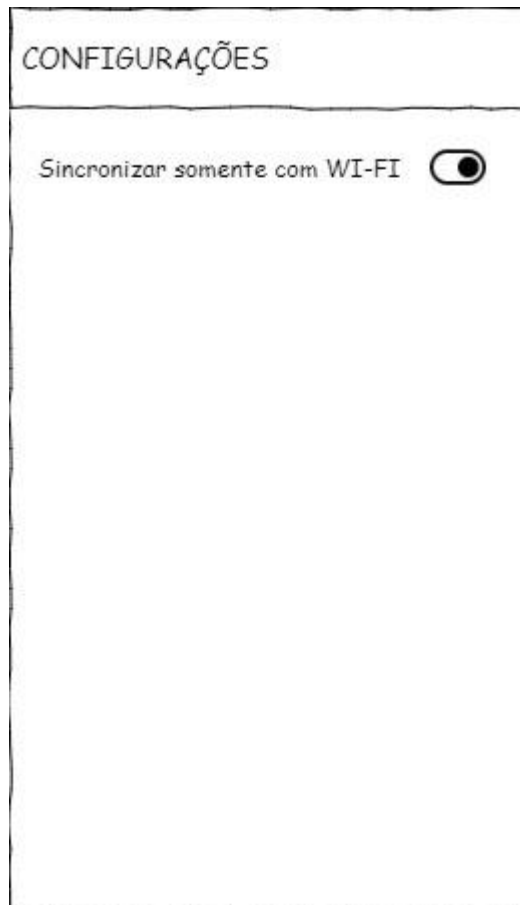
RF14 – Configurações**Descrição**

Este caso de uso serve para configurar preferências do usuário.

Data View

DV14 - Tela de Configurações

FIGURA 36 - TELA DE CONFIGURAÇÕES



FONTE: O autor (2021).

Ator Primário

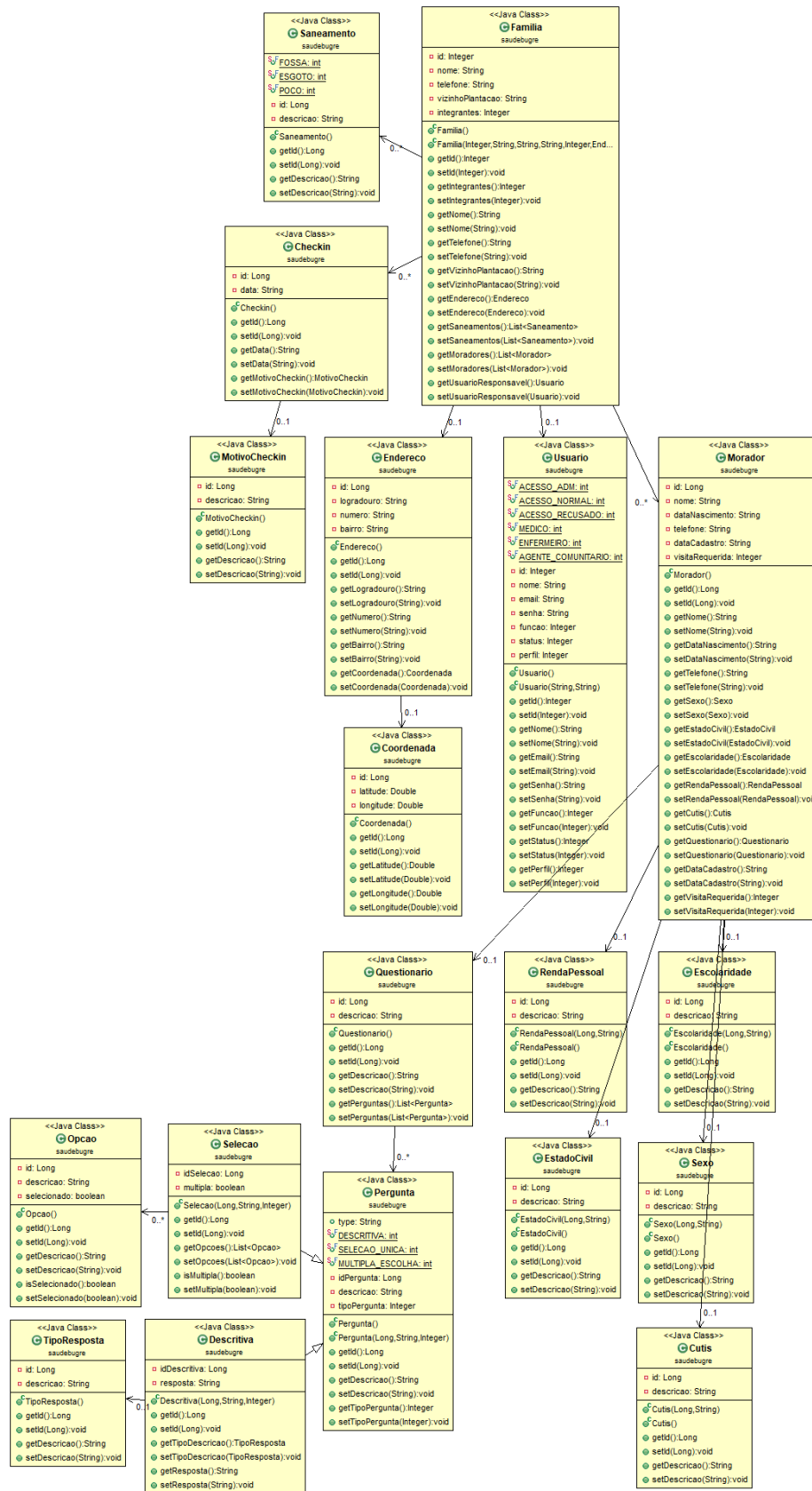
Usuário

Fluxo de Eventos Principal

1. O sistema carrega a tela (DV14)
2. O usuário seleciona "Sincronizar somente com WI-FI"

APÊNDICE 4 – DIAGRAMA DE CLASSES

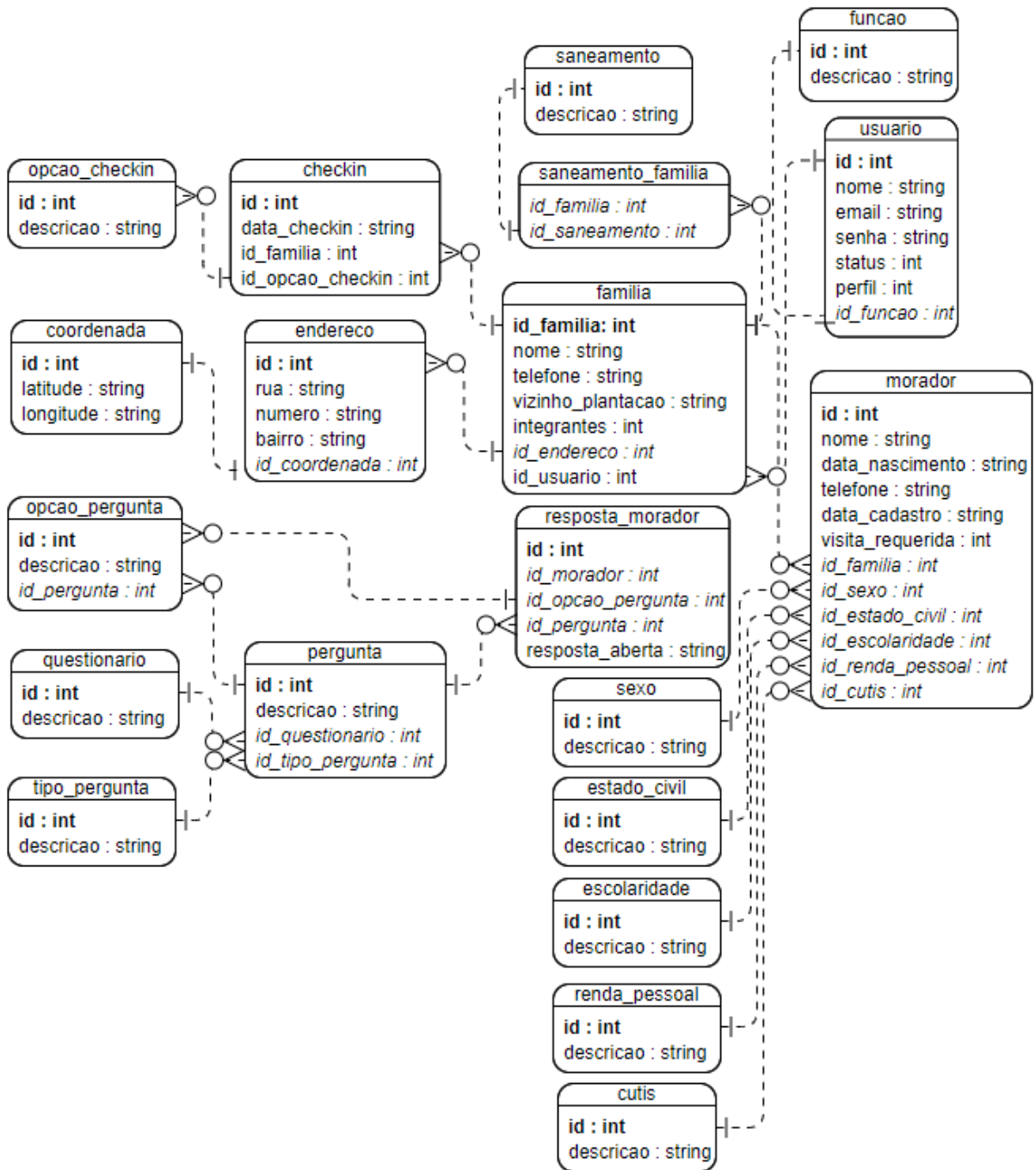
FIGURA 37 - DIAGRAMA DE CLASSES



FONTE: O autor (2021).

APÊNDICE 5 – DIAGRAMA LÓGICO-RELACIONAL

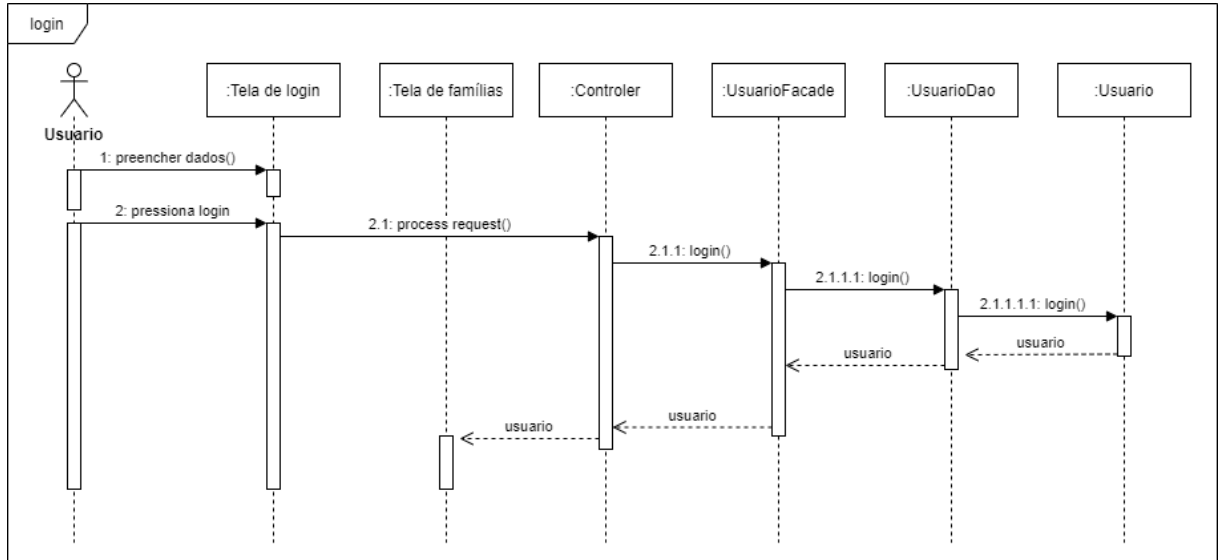
FIGURA 38 - DIAGRAMA LÓGICO-RELACIONAL



FONTE: O autor (2021).

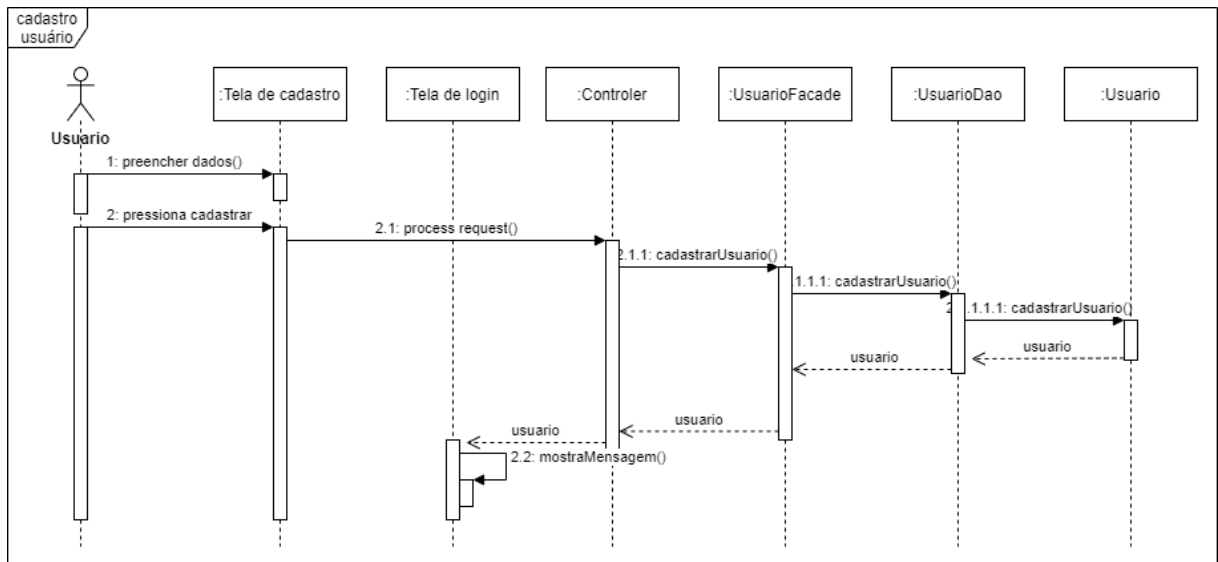
APÊNDICE 6 – DIAGRAMA DE SEQUÊNCIA

FIGURA 39 - LOGIN



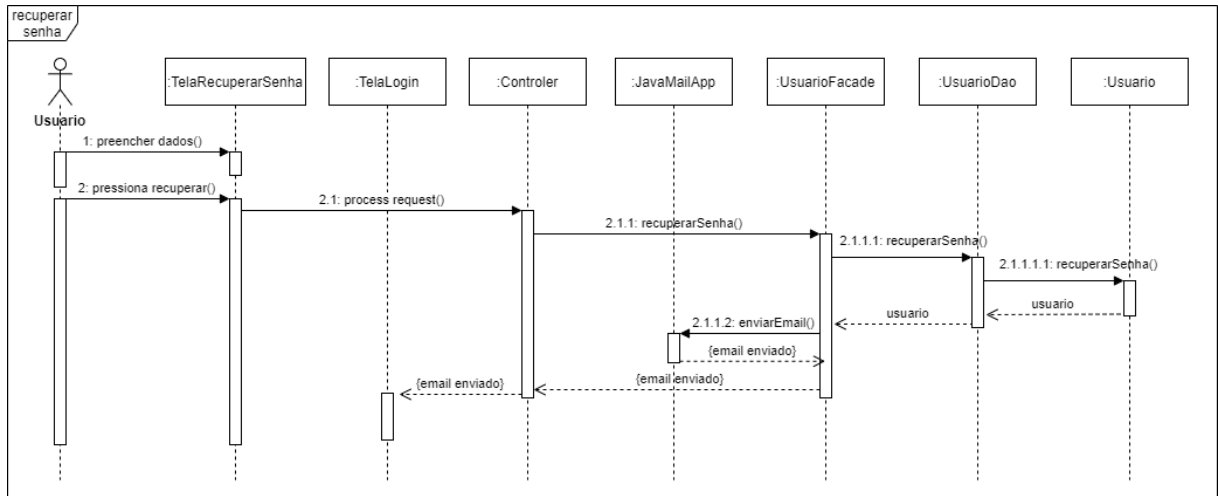
FONTE: O autor (2021).

FIGURA 40 – CADASTRAR-SE



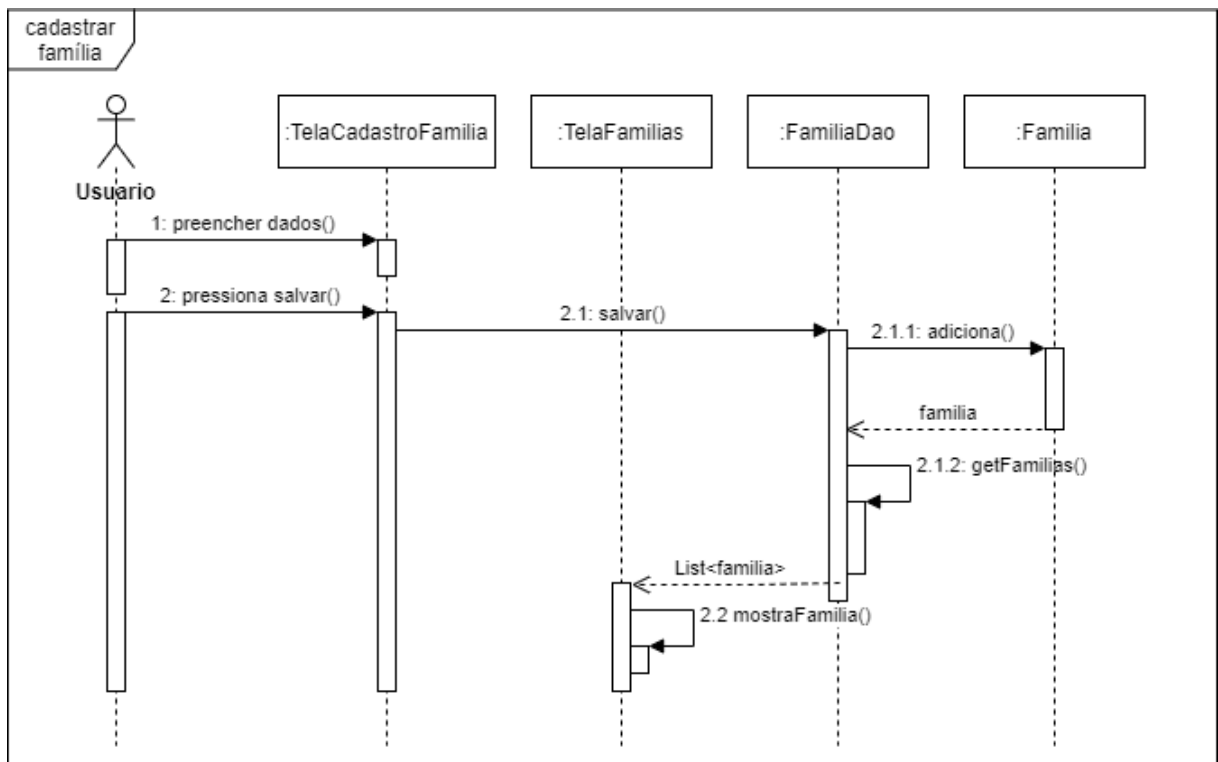
FONTE: O autor (2021).

FIGURA 41 – RECUPERAR SENHA



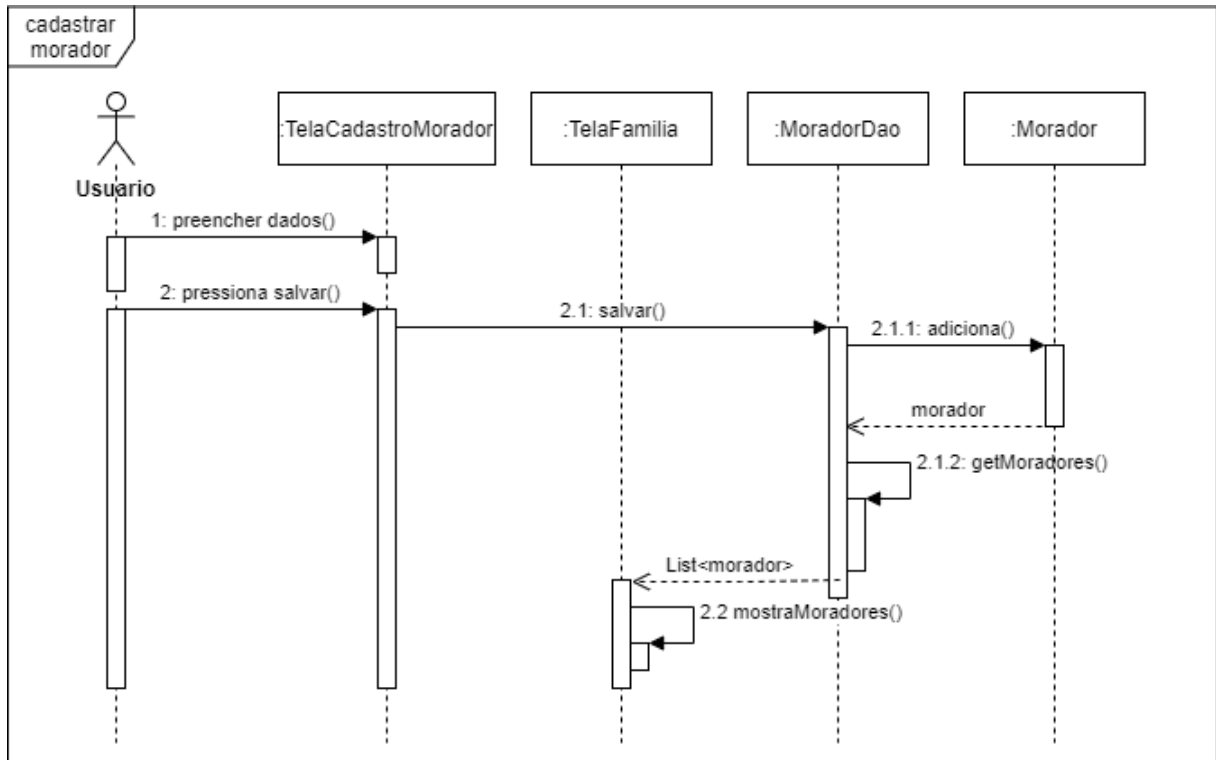
FONTE: O autor (2021).

FIGURA 42 – CADASTRAR FAMÍLIA



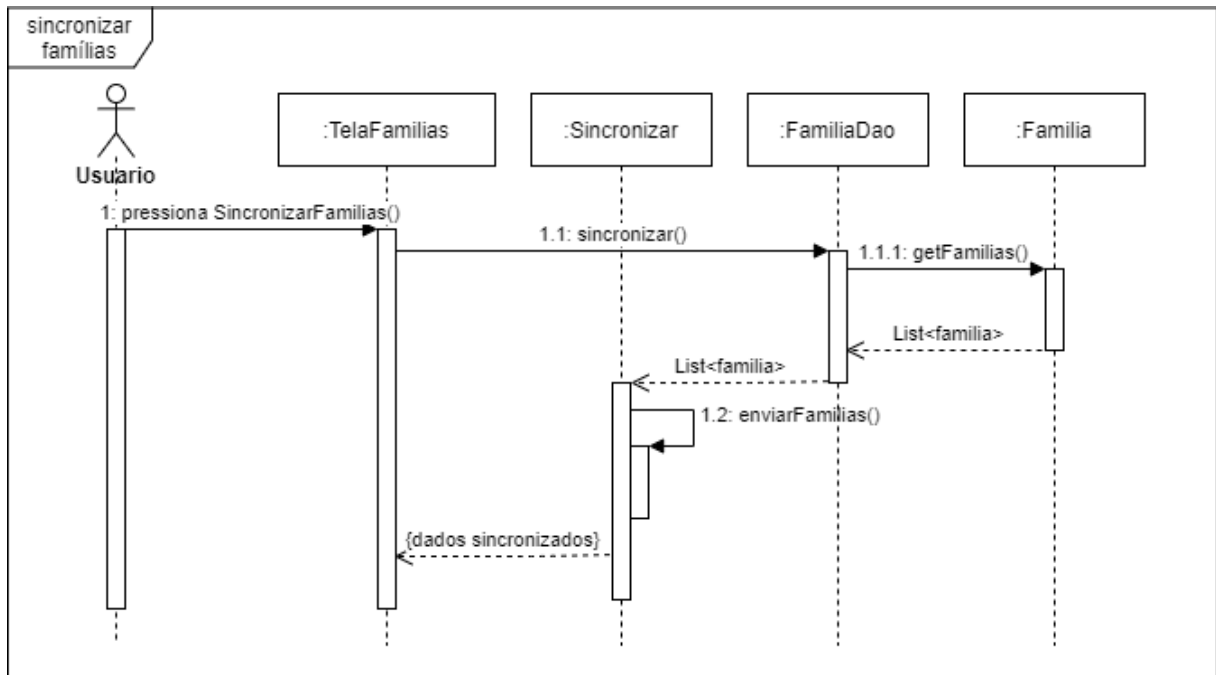
FONTE: O autor (2021).

FIGURA 43 – CADASTRAR MORADOR



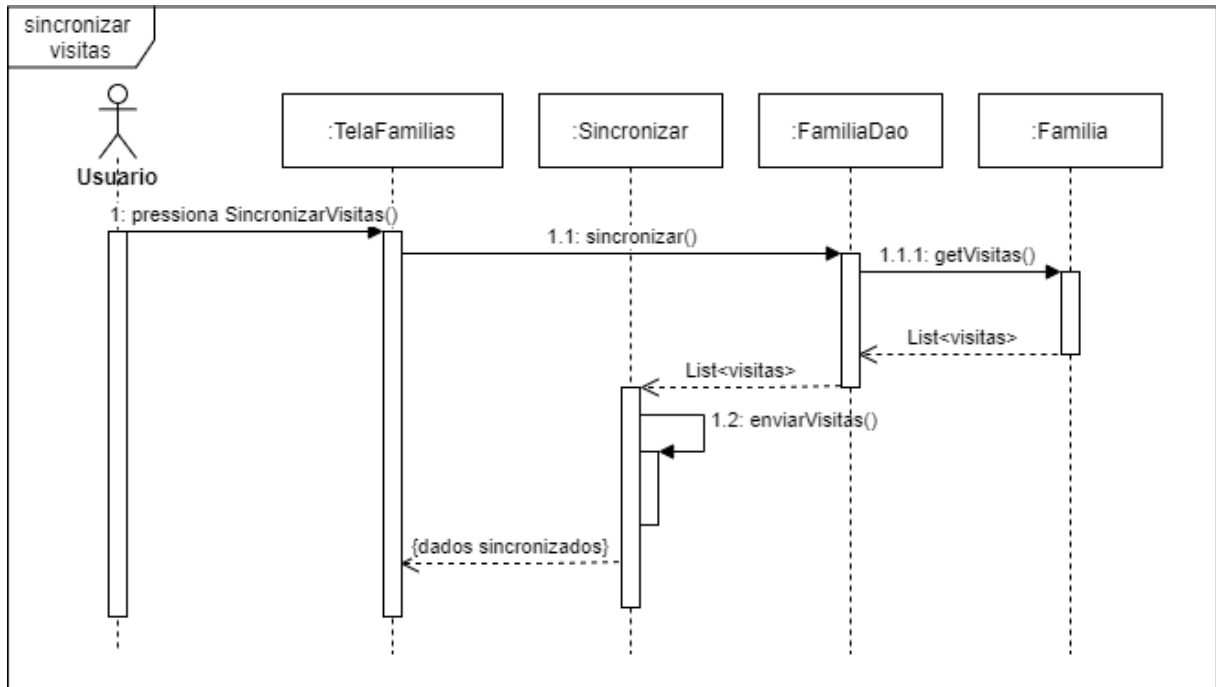
FONTE: O autor (2021).

FIGURA 44 – SINCRONIZAR FAMÍLIAS



FONTE: O autor (2021).

FIGURA 45 – SINCRONIZAR VISITAS



FONTE: O autor (2021).

APÊNDICE 7 – TESTE DE CRITÉRIOS ERGONÔMICOS

1. As informações apresentadas são claras para o usuário?

Atende Atende parcialmente Não atende Não se aplica

2. O sistema foi concebido a fim de estar próximo de como o usuário coleta dados no mundo real?

Atende Atende parcialmente Não atende Não se aplica

3. Toda mudança no aplicativo ocorre somente após o usuário iniciar uma ação?

Atende Atende parcialmente Não atende Não se aplica

4. Os botões e símbolos apresentados no aplicativo são coerentes com o contexto?

Atende Atende parcialmente Não atende Não se aplica

5. O aplicativo oferece ao usuário dicas de como ele deve entrar com os dados?

Atende Atende parcialmente Não atende Não se aplica

6. As instruções de como o usuário deve proceder para registrar uma família são claras?

Atende Atende parcialmente Não atende Não se aplica

7. O aplicativo oferece recursos para ajustar-se às necessidades do usuário?

Atende Atende parcialmente Não atende Não se aplica

8. A aparência do aplicativo facilita as ações do usuário?

Atende Atende parcialmente Não atende Não se aplica

9. Suporte para usuário reconhecer, diagnosticar e recuperar erros

Atende Atende parcialmente Não atende Não se aplica

10. O aplicativo mostra informações que auxiliam o usuário em suas ações?

Atende Atende parcialmente Não atende Não se aplica