



ANTONIO MENEGHETTI FACULDADE - AMF
CURSO DE SISTEMAS DE INFORMAÇÃO

WESLEY ANJOS DOS SANTOS

APLICATIVO PARA O SERVIÇO DE ATENÇÃO DOMICILIAR

RESTINGA SECA/RS

2020

WESLEY ANJOS DOS SANTOS

APLICATIVO PARA O SERVIÇO DE ATENÇÃO DOMICILIAR

Trabalho de Conclusão do Curso de Sistemas de Informação
Orientador: Profº. Dr. Jonas Bulegon Gassen

RESTINGA SECA/RS

2020

AGRADECIMENTOS

Quero agradecer a toda minha família, em especial minha mãe, Sandra Nunes dos Anjos, meus avós, Valdir Nunes dos Anjos e Terezinha Deloni de Paula dos Anjos, ao meu tio, Jorge Luiz Nunes dos Anjos, e ao meu padrasto, Alaerte Fernandes da Silva, por sempre acreditarem em mim, me apoiarem em minhas escolhas, por toda educação que me foi proporcionada e fazerem de tudo para que eu pudesse realizar essa trajetória e tornar esse momento possível.

A minha melhor amiga, Caroline Antunes, por ter me ajudado, incentivado e não deixado eu desistir durante este período.

Aos professores do curso de Sistemas de Informação, em especial ao meu orientador Jonas Bulegon Gassen, por me orientar e aconselhar nesse período, e ao professor Felipe Becker Nunes, por ensinar quase tudo que sei sobre programação e por ter acompanhado este projeto durante todo esse tempo.

Aos meus colegas e amigos, em especial ao Henrique Rosa, Suelen Costa, Francisco Bitencourt e Dienefer dos Santos.

Por fim, à Antonio Meneghetti Faculdade e a todos que contribuíram e fizeram parte desta importante jornada.

A todos, meu muito obrigado.

“É muito melhor arriscar coisas grandiosas, alcançar triunfos e glórias, mesmo expondo-se à derrota, do que formar fila com os pobres de espírito que nem gozam muito nem sofrem muito, porque vivem nessa penumbra cinzenta que não conhece vitória nem derrota”.

Theodore Roosevelt

RESUMO

Os profissionais de saúde do Serviço de Atenção Domiciliar do Hospital Universitário de Santa Maria necessitam estar constantemente em locomoção para as residências de seus pacientes, a fim de atendê-los. Devido a não estarem no hospital durante o atendimento, os profissionais não conseguem usufruir do sistema de informação disponibilizado, tendo assim, que recorrer a utilização de papel para anotação e coleta de dados recentes que virão a ser passados para o sistema do hospital. Diante disso, o presente trabalho tem como objetivo o desenvolvimento de um protótipo de aplicativo móvel, projetado para o SADHUSM. Foram utilizadas as linguagens de programação Node.js e TypeScript, o framework React Native e o padrão REST. Para tanto, foram elaborados diagramas UML, fluxograma de processos e protótipos de telas de média e alta fidelidade. O resultado do trabalho constitui-se em um protótipo de aplicativo móvel.

Palavras-chave: mHealth, Node.js, React Native, TypeScript, PostgreSQL.

ABSTRACT

The healthcare professionals from the Serviço de Atenção Domiciliar from the Hospital Universitário de Santa Maria (SADHUSM) are constantly visiting their patients at home. Since they are not within the hospital for their tasks, they cannot take advantage of the information system provided. Therefore, they have to use paper for taking notes and data collection, which will later on be added to the hospital system. This work aims to develop a prototype of mobile app, designed to help the SADHUSM personnel. For the development, Noje.js and TypeScript were used as programming languages. React Native framework and the REST standard were also used. UML diagrams, flowcharts and screen prototypes were created. The result is a prototype of a mobile app.

Keywords: mHealth, Node.js, React Native, TypeScript, PostgreSQL

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de Classe.....	25
Figura 2 - Diagrama de Caso de Uso.....	26
Figura 3 - Fluxograma de processo.....	27
Figura 4 - Protótipo de média fidelidade da tela de login.....	28
Figura 5 - Protótipo de média fidelidade da lista de pacientes.....	28
Figura 6 - Protótipo de média fidelidade da tela do histórico do paciente.....	29
Figura 7 - Protótipo de média fidelidade dos dados do paciente.....	30
Figura 8 - Protótipos de alta fidelidade das telas (Respectivamente da esquerda para direita: Tela de login, lista de pacientes e prontuário dos pacientes).....	31
Figura 9 - Tela de login.....	32
Figura 10 - Tela da lista de pacientes.....	33
Figura 11 - Tela de perfil do profissional.....	34
Figura 12 - Tela de perfil do paciente.....	35
Figura 13 - Tela da evolução.....	36
Figura 14 - Tela de cadastro do profissional parte 1.....	37
Figura 15 - Tela de cadastro do profissional parte 2.....	38
Figura 16 - Tela de cadastro do paciente parte 1.....	39
Figura 17 - Tela de cadastro do paciente parte 2.....	40
Figura 18 - Tela de atualização do profissional parte 1.....	41
Figura 19 - Tela de atualização do profissional parte 2.....	42
Figura 20 - Tela de atualização do paciente parte 1.....	43
Figura 21 - Tela de atualização do paciente parte 2.....	44
Gráfico 1 - Ranking das linguagens mais utilizadas no GitHub.....	17
Gráfico 2 - Ranking das <i>engines</i> de bancos de dados mais utilizadas no mundo.....	19

LISTA DE ABREVIATURAS

API - *Application Programming Interface*

Apps - Aplicativos móveis

CID - Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde

HUSM - Hospital Universitário de Santa Maria

HTTP - *Hypertext Transfer Protocol*

iOS - *internet Operating System*

JS - JavaScript

mHealth – *Mobile Health*

POO - Programação Orientada a Objetos

REST - *Representation State Transfer*

SADHUSM – Serviço de Atenção Domiciliar do Hospital Universitário de Santa Maria

SAME - Serviço de Arquivamento Médico e Estatístico

SQL - *Structured Query Language*

UML - *Unified Modeling Language*

VSCode - *Visual Studio Code*

SUMÁRIO

1	INTRODUÇÃO	11
1.1	OBJETIVOS	12
1.1.1	Objetivo principal	12
1.1.2	Objetivos específicos	12
1.2	JUSTIFICATIVA	12
2	ABORDAGEM TEÓRICA	14
2.1	O USO DE APLICATIVOS NA ÁREA DA SAÚDE	14
2.2	DESENVOLVIMENTO DO SOFTWARE PARA UM APLICATIVO	15
2.3	TECNOLOGIAS EMPREGADAS NO DESENVOLVIMENTO DO PROJETO	16
2.3.1	Programação orientada a objetos (POO)	16
2.3.2	JavaScript (JS)	16
2.3.3	TypeScript	17
2.3.4	NodeJS	18
2.3.5	React native	18
2.3.6	PostgreSQL	19
2.4	FERRAMENTAS UTILIZADAS	20
2.4.1	Visual studio code	20
2.4.2	PgAdmin 4	20
2.4.3	Insomnia REST	21
3	METODOLOGIA	22
3.1	MODELAGEM DO PROJETO	22
3.2	O BACK-END E O BANCO DE DADOS	22
3.3	O FRONT-END	22
4	O PROJETO DO SISTEMA MHEALTH	23
4.1	REQUISITOS FUNCIONAIS, NÃO FUNCIONAIS E REGRAS DE NEGÓCIO	23
4.2	ELABORAÇÃO DE DIAGRAMAS ESTRUTURAIS E COMPORTAMENTAIS	
	UML	24
4.3	FLUXOGRAMA DE PROCESSO	26
4.4	PROTÓTIPOS DAS TELAS	26

5	RESULTADOS	32
5.1	LOGIN	32
5.2	LISTA DE PACIENTES	33
5.3	PERFIL DO PROFISSIONAL	34
5.4	PERFIL DO PACIENTE	35
5.5	EVOLUÇÃO	36
5.6	CADASTRO DE PROFISSIONAL	37
5.7	CADASTRO DE PACIENTE	39
5.8	ATUALIZAÇÃO DO PROFISSIONAL	41
5.9	ATUALIZAÇÃO DO PACIENTE	43
6	CONSIDERAÇÕES FINAIS	45
7	REFERÊNCIAS	46

1 INTRODUÇÃO

Com o crescimento da internet e mídias sociais, a utilização de dispositivos móveis, *smartphones* e *tablets* tem proporcionado uma enorme renovação nos últimos tempos, nas mais diversas áreas (MACHADO, 2018). Mobile health (mHealth) é a área que trata da utilização de dispositivos móveis voltados para a saúde, assim proporcionando uma facilidade em consultas médicas, diagnósticos e acompanhamento dos pacientes (MACHADO, 2018).

Segundo a Pesquisa Digital 2017 Global Overview, cerca de $\frac{2}{3}$ da população mundial possui um smartphone, 50% da população mundial utiliza a internet sendo que destes 46% acessam a rede através do seu smartphone. No Brasil, 68% da população faz uso da internet, sendo que destes 93% acessam a rede através do seu smartphone (KEMP, 2017).

No Brasil, há uma dificuldade no uso da tecnologia na assistência à saúde. Existe uma parcela significativa de esforço em pesquisas de novas tecnologias. Contudo, há uma urgência de aprofundamento nos conhecimentos em saúde, como também da implementação de iniciativas que buscam um modelo de desenvolvimento socialmente inclusivo e economicamente competitivo.

Na área da saúde, o emprego dos aplicativos para *tablets* e *smartphones* (apps) permite a execução de atendimentos e de cuidados com avaliações e respostas imediatas, contribuindo dessa forma para uma assistência mais efetiva. Estes aplicativos têm como característica principal a simplicidade de utilização e mobilidade. Seu uso vem crescendo aceleradamente na área da saúde, graças ao fato de melhorar a precisão e agilidade para a tomada de decisão dos profissionais de saúde, além de que facilitam a pesquisa científica no local de trabalho. Os aplicativos estão disponíveis em “lojas virtuais” específicas dos dispositivos móveis, sendo que as duas principais são a “Play Store” (para dispositivos com Sistema Operacional Android - Google), e “Apple Store” (para dispositivos com Sistema Operacional iOS - Apple).

Há estudos que comprovam que os benefícios do uso dessas tecnologias, nas intervenções de saúde, têm melhorado a tomada de decisão clínica e também a educação dos pacientes, além disso oferecem suporte para a telemedicina e programas de educação continuada (DENNISON, 2013).

Porém, nem todos os profissionais da saúde têm acesso a ferramentas deste tipo. Trazendo para nossa realidade, os Profissionais do Serviço de Atenção Domiciliar do Hospital

Universitário de Santa Maria (SADHUSM) enfrentam uma dificuldade de comunicação e troca de dados entre suas equipes quando vão atender os pacientes em domicílio. Isso ocorre devido a não poderem utilizar o sistema do hospital fora do mesmo, já que o sistema foi feito apenas para desktop. Os profissionais podem acessar os dados antes de saírem do hospital, porém se uma equipe adicionou algum dado novo, e outra equipe já está em locomoção para a residência do paciente, por exemplo, esses dados novos não poderão ser acessados.

Para resolver esse problema, decidiu-se fazer um mHealth para que os profissionais possam trocar informações em tempo real durante suas visitas domiciliares, tornando assim os atendimentos mais precisos e eficientes.

1.1 OBJETIVOS

A fim de resolver o problema exposto anteriormente, este trabalho apresenta os seguintes objetivos:

1.1.1 Objetivo principal

Este trabalho tem como objetivo desenvolver um protótipo de aplicativo mHealth, para otimização dos atendimentos domiciliares da equipe de profissionais do SADHUSM.

1.1.2 Objetivos específicos

- Iniciar uma parceria com os profissionais de saúde do SADHUSM;
- Analisar quais pontos podem ser melhorados com o mHealth;
- Desenvolver um protótipo de sistema mobile voltado para a área da saúde, um mHealth, e disponibilizá-lo para os profissionais de saúde do SADHUSM.

1.2 JUSTIFICATIVA

A motivação para o desenvolvimento desta solução surgiu através da mãe do autor deste trabalho, a qual trabalha no SADHUSM a cerca de 5 anos, foi através dela que o autor veio a conhecer o problema. Com isso, o mesmo pode ir ao hospital conversar com uma das profissionais de saúde que integram a equipe deste setor no Hospital Universitário de Santa Maria

(HUSM). Esta profissional descreveu a dificuldade de troca de informações enfrentada por todos os profissionais de saúde do SADHUSM que fazem visitas domiciliares, onde atualmente é utilizado um caderno para registrar os dados coletados nos atendimentos.

Após tomar conhecimento do problema, surgiu a ideia de desenvolver um aplicativo móvel que supriria essa dificuldade, agilizando o processo de acessar os dados em qualquer lugar, tornando os atendimentos mais eficientes e precisos.

A utilização da internet tornou-se um potencial a ser explorado como forma de elevar a qualidade da saúde da população, uma vez que o número de usuários de dispositivos móveis aumenta diariamente. O smartphone, por exemplo, está ganhando um papel cada dia mais importante no auxílio e nas intervenções em saúde (DENNISON, *et al.*, 2013; SILVA & SANTOS, 2014; BOBIAN *et al.*, 2016).

Diante disso, o autor resolveu iniciar um novo projeto, utilizando tecnologias multiplataformas e melhores práticas de programação, facilitando assim o desenvolvimento do código. Em seguida, foi dado início ao processo de criação deste aplicativo que vai ser disponibilizado gratuitamente para os profissionais de saúde do SADHUSM assim que seu desenvolvimento tiver sido concluído.

2 ABORDAGEM TEÓRICA

Nesta seção, a base teórica é apresentada, o que dá maior perspectiva a esse projeto. Está exposto um maior detalhamento sobre mHealth, tecnologias empregadas no desenvolvimento da aplicação, bem como seus conceitos, e as ferramentas utilizadas.

2.1 O USO DE APLICATIVOS NA ÁREA DA SAÚDE

Com o crescimento e expansão da *internet* na década de 1990, uma série de conceitos eletrônicos como *e-mail*, por exemplo, começaram a aparecer e multiplicar-se, possibilitando o compartilhamento de experiências mais rapidamente. Assim como o *eHealth*, representando uma promessa na tecnologia da informação e comunicação para melhorar o sistema de saúde (OH *et al.*, 2005).

O termo *eHealth* é traduzido em português como *e-saúde*. O termo representa um campo em desenvolvimento na junção de informática médica, saúde pública e negócios, com o intuito de oferecer serviços de saúde, a partir de informações fornecidas e aprimoradas através da internet e tecnologias envolvidas (EYSENBACH, 2001).

Segundo a definição da Organização Mundial da Saúde (OMS), o *mHealth* representa na prática médica com suporte para dispositivos móveis, como por exemplo, para telefones celulares, dispositivos para monitoramento de pacientes, *tablets* e outros dispositivos sem fio, que suportam funções de celulares como áudio, SMS e aplicações mais complexas com serviços de 3G, 4G, GPS (*Global Positioning System*), *Bluetooth* e GRPS (*General Packet Radio Service*) (WORLD HEALTH ORGANIZATION, 2011; KUMAR *et al.*, 2013).

Os *smartphone* e *tablets* são os dispositivos mais utilizados para o acesso ao *mHealth*. O Sistema *Android*, Google e o IOS da Apple – das lojas associadas Google Play e App Store – são as principais plataformas para suporte da tecnologia e oferecem uma grande quantidade de aplicativos *mHealth* (MELLO & SGANZERLA, 2013; DEHLING, *ET AL.*, 2015).

Apesar dos avanços tecnológicos no campo da saúde digital, que incluem aplicativos médicos entre outros relacionados à saúde, pouco se sabe sobre a forma como as pessoas utilizam e se beneficiam do uso de tais tecnologias. Mas ainda, pesquisadores não conseguem dimensionar se os aplicativos desenvolvidos cumprem a sua função. (LUPTON; JUTEL, 2015).

Outra preocupação diz respeito à segurança e à privacidade das informações pessoais dos usuários (sigilo de informações). Estudos mostraram que a maioria dos aplicativos *mHealth*

solicitam acesso a informações pessoais confidenciais, fato que pode gerar danos potenciais por meio da manipulação de informações e violações de privacidade (DEHLING, *et al.*, 2015).

A atual geração de consumidores e pacientes vive uma vida agitada, onde as pessoas buscam por respostas rápidas, de preferência online. Além disso, muitos indivíduos se preocupam em cuidar da saúde. A utilização de tecnologias atuais nessa linha é crescente, existem muitos aplicativos e recursos à disposição para ajudar, desde a alimentação até exercícios físicos (ICLINIC, 2016).

De acordo com dados das lojas de aplicativos Google Play e App Store, existem mais de 97 mil aplicativos voltados ao setor de saúde e fitness, que juntos geram 4 milhões de downloads gratuitos e outros 300 mil downloads pagos em apenas um único dia (MORSCH, 2016).

Segundo a empresa Great Call, o perfil de usuários dos aplicativos de saúde tem em média 35 anos, 46% são mulheres, 54% são homens, 85% usam mídias sociais para pesquisar sobre saúde e 76% deles tem uma prescrição médica com medicamentos para seguir (MORSCH, 2016).

2.2 DESENVOLVIMENTO DO SOFTWARE PARA UM APLICATIVO

O processo de desenvolvimento de um *software* é um conjunto sequencial de ações gerenciais e técnicas colaborativas com o objetivo de gerar, especificar, projetar, implementar e testar o software. Diferentes ferramentas são utilizadas para que um *software* seja desenvolvido, sendo que as mesmas suportam a edição de diferentes tipos de documentos e coordenam as informações geradas pelo projeto inicial (SCHNEIDER; SHIPP, 2010).

Com a evolução da tecnologia, os dispositivos móveis passaram a adquirir diversas funções. A funcionalidade do uso de ligações e mensagens via *SMS* diminuiu drasticamente em decorrência do desenvolvimento de *softwares* mais avançados que permitem o acesso à *internet*, com mais recursos e funções (SILVA & SANTOS, 2014). Os dispositivos móveis passaram a ser considerados computadores de bolso devido aos seus recursos avançados, tornando assim os aplicativos de saúde variáveis e inovadores, acessíveis tanto para pacientes quanto para profissionais (BAIG; GHOLAMHOSSEINI; CONNOLLY, 2015).

Frente a essa mudança de paradigma, o mercado de aplicativos móveis apresenta um crescimento rápido, sendo disputado por diferentes plataformas tecnológicas. Mesmo assim, ainda existem pontos problemáticos no que concerne às limitações para a distribuição nas

plataformas, custo, tempo para o desenvolvimento e complexidade da tecnologia envolvida na criação e manutenção (SILVA & SANTOS, 2014).

2.3 TECNOLOGIAS EMPREGADAS NO DESENVOLVIMENTO DO PROJETO

2.3.1 Programação orientada a objetos (POO)

A Programação Orientada a Objetos é um paradigma de programação que foi criado por Alan Kay e tem como objetivo aproximar o mundo real ao mundo virtual, buscando criar termos similares entre ambos, onde o principal termo encontrado foi “Objeto”, visto que todas as coisas que vemos pode ser definido como um objeto. Em sistemas que são desenvolvidos utilizando POO, os objetos (classes) que neles foram definidos e criados podem interagir entre si, onde cada um terá especificado o tipo de informação que pode enviar e receber (propriedades) e também as ações que ele mesmo pode realizar (métodos) (MONQUEIRO, 2007).

De acordo com Gasparotto (2014), há quatro pilares principais na Programação Orientada a Objetos, utilizados como requisitos para que uma linguagem possa ser considerada nesse paradigma, que são:

- **Abstração:** cada classe deverá ter uma identidade (ser única), possuir suas propriedades e métodos;
- **Encapsulamento:** define um nível de segurança de acesso às propriedades de uma classe, sendo elas privadas, protegidas ou públicas;
- **Herança:** consiste na possibilidade de uma classe herdar propriedades e métodos de outra classe e poder utilizá-la da mesma forma, caso elas não sejam privadas;
- **Polimorfismo:** consiste na possibilidade de uma classe poder alterar o comportamento ou funcionamento de um método herdado de outra classe.

Dessa forma, o desenvolvimento de um software se torna menos complexo, mais produtivo e compreensível, onde cada código criado possui uma responsabilidade e é feito de forma lógica, de acordo com a realidade.

2.3.2 JavaScript (JS)

O JavaScript é uma linguagem leve, interpretada e baseada em objetos com funções de primeira classe, mais conhecida como a linguagem de script para páginas Web, mas usada também em vários outros ambientes sem browser, como node.js, Apache CouchDB e Adobe

Acrobat. É uma linguagem baseada em protótipos, multi-paradigma e dinâmica, suportando estilos de orientação a objetos, imperativos e declarativos (MDN, 2020).

O JavaScript acabou se tornando a linguagem mais utilizada pelos usuários do GitHub, que é o maior sistema de gerenciamento de projetos e versões de códigos do mundo, com uma diferença de mais que o dobro de utilizadores da segunda linguagem mais utilizada, o Python, como mostrado no Gráfico 1.

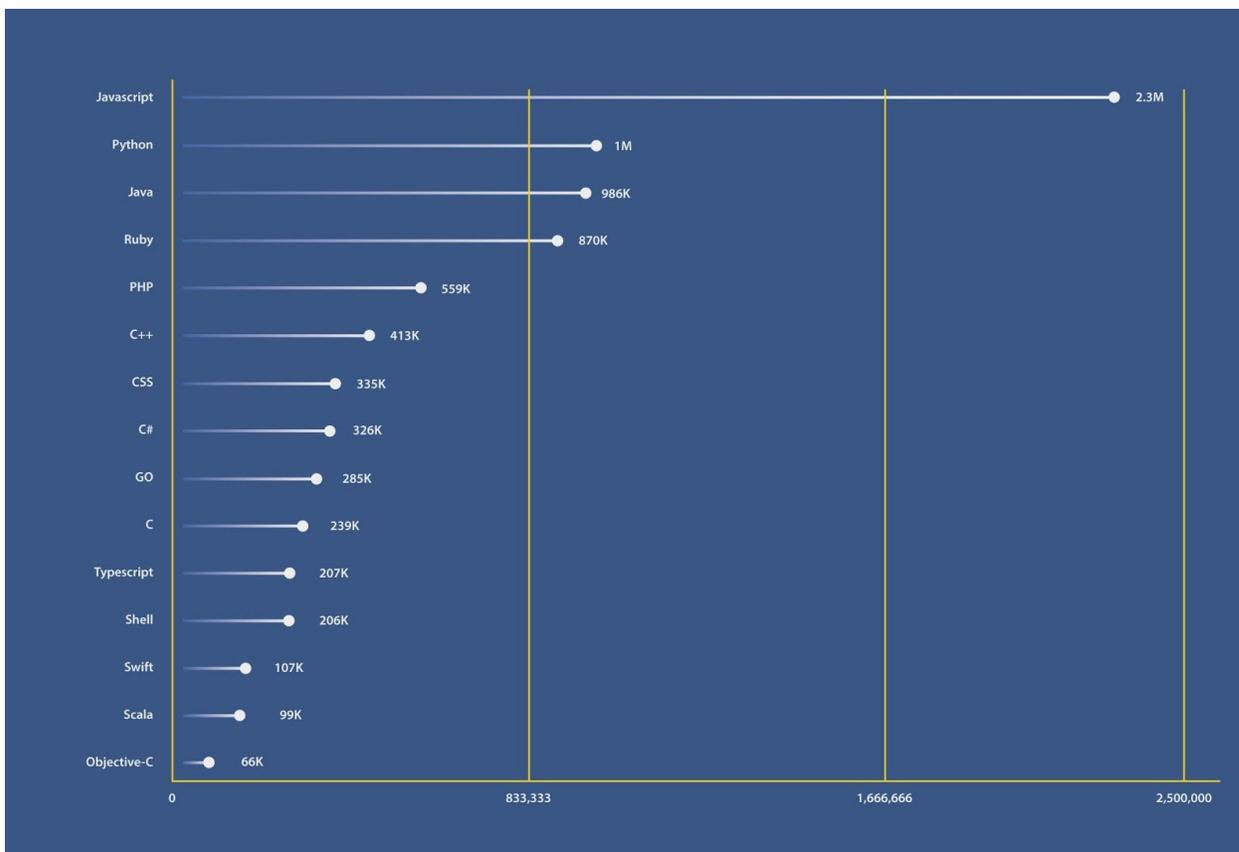


Gráfico 1. Ranking das linguagens mais populares de acordo com o GitHub.

Fonte: (LETSCODE, 2017)

2.3.3 TypeScript

O TypeScript é uma ferramenta que adiciona tipagem estática ao JavaScript, linguagem que, por padrão, possui tipagem dinâmica, ou seja, as variáveis e funções podem assumir tipos distintos durante o tempo de execução (FERNANDES, 2018).

O código TypeScript é utilizado somente em ambiente de desenvolvimento e é totalmente convertido para JavaScript no processo de *build* de produção, ou seja, o navegador ou o Node lerão somente código JS (FERNANDES, 2018).

Além de ajudar no ambiente de desenvolvimento, o TypeScript ainda permite que utilizemos funcionalidades da linguagem que ainda não estão disponíveis de forma nativa, por exemplo, no Node.js podemos utilizar os *ES Modules (import/export)* (FERNANDES, 2018).

Neste projeto, o TypeScript foi utilizado em conjunto com o React Native, na interface que o usuário utilizará (*frontend*), com o intuito de criar uma maior semelhança com o código Node.js, através das funcionalidades da Programação Orientada a Objetos presente em ambas linguagens, e também para facilitar futuras manutenções no código do cliente.

2.3.4 Node.JS

Node (ou formalmente Node.js) é um ambiente em tempo de execução *open-source* (código aberto) e multiplataforma que permite aos desenvolvedores criarem todo tipo de aplicativos e ferramentas do lado servidor (*backend*) em JavaScript (JS). Node é usado fora do contexto de um navegador, ou seja, executado diretamente no computador ou no servidor. Como tal, o ambiente omite APIs (*Application Programming Interface*) JavaScript específicas do navegador e adiciona suporte para APIs de sistemas operacionais mais tradicionais, incluindo bibliotecas de sistemas HTTP (*Hypertext Transfer Protocol*) e arquivos (MDN, 2020).

Neste projeto, o Node.js foi utilizado para desenvolver a parte de *backend* que processa e persiste os dados no banco de dados.

2.3.5 React native

O React Native é um *framework* baseado no já aclamado React, desenvolvido pela equipe do Facebook, que possibilita o desenvolvimento de aplicações *mobile*, tanto para Android, como para iOS (*internet Operating System*), utilizando apenas JavaScript (BECKER, 2020).

Antes do surgimento do React Native, desenvolver para Android e iOS era algo relativamente complexo, pois além de ter que aprender as linguagens Objective-C (iOS) e Java (Android), o desenvolvedor não aproveitava praticamente nada do código de uma plataforma para outra, fazendo assim com que as empresas contratassem um time de desenvolvimento para cada sistema operacional, tornando o projeto muito lento e caro. Porém, com o React Native, o código

pode ser reaproveitado em até 100% entre as plataformas, podendo fazer com que o custo e a duração do projeto caiam pela metade (BECKER, 2020).

2.3.6 PostgreSQL

O PostgreSQL é uma ferramenta que atua como sistema de gerenciamento de bancos de dados relacional. Seu foco é permitir implementação da linguagem SQL em estruturas, garantindo um trabalho com padrões de ordenação dos dados (SOUZA, 2020).

O PostgreSQL tem o papel de gerenciar os dados desses bancos de maneira organizada e eficaz, rodando e gravando todas as informações que ficam registradas nesses compartimentos. Por meio desse sistema, usuários podem executar consultas de maneira simples, sem precisar acessar diretamente o banco de dados (SOUZA, 2020).

Neste projeto, o PostgreSQL foi utilizado para armazenar os dados que eram utilizados como teste no decorrer do desenvolvimento, também será utilizado para armazenar os dados registrados quando o sistema for implementado antes de serem inseridos no sistema real do hospital.

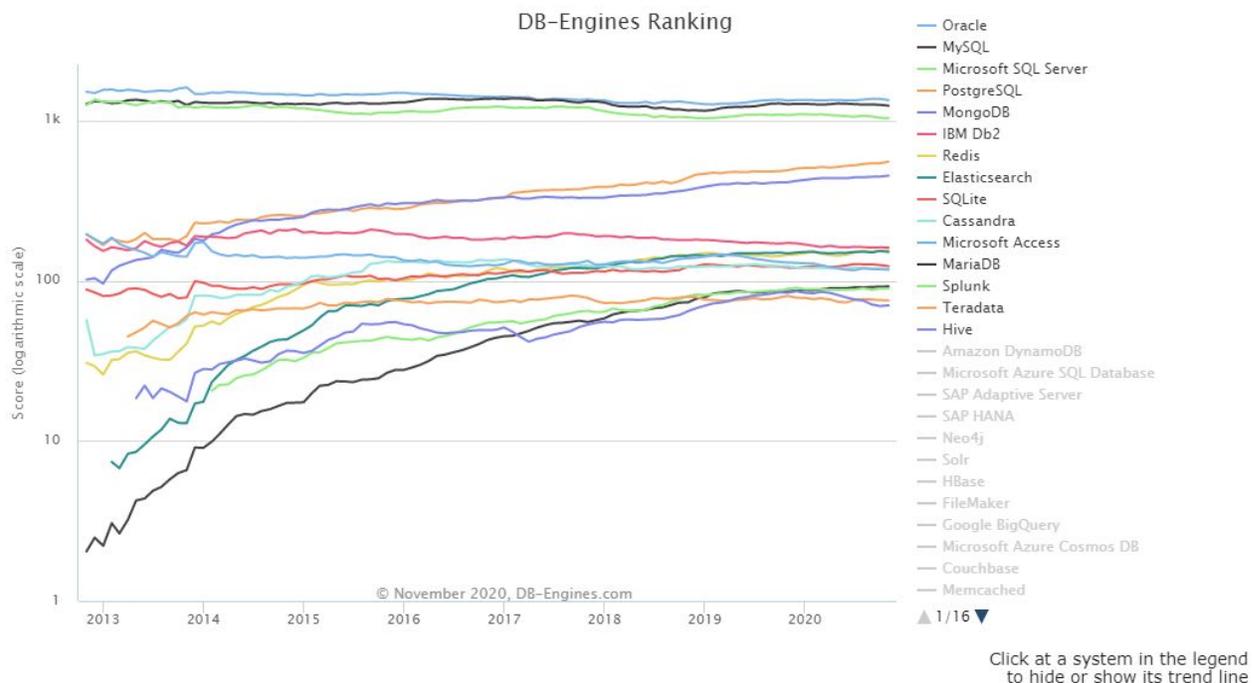


Gráfico 2. Ranking das *engines* de bancos de dados mais utilizados no mundo..

Fonte: (DB-ENGINES, 2020)

2.4 FERRAMENTAS UTILIZADAS

2.4.1 Visual studio code

Em 2015 foi lançado pela Microsoft um editor de código destinado ao desenvolvimento de aplicações web chamado de Visual Studio Code, ou simplesmente VSCode. É uma ferramenta leve e multiplataforma que está disponível tanto para Windows, quanto para Mac OS e Linux e atende a uma gama enorme de projetos, não apenas ASP.NET, como também Node.js (DIONISIO, 2016).

Além de ser totalmente gratuito, o editor é *open source*, tendo código disponibilizado no GitHub, o que permite à comunidade técnica contribuir com seu desenvolvimento e facilitando a criação de extensões e novas funcionalidades (DIONISIO, 2016).

O VSCode foi utilizado durante todo o desenvolvimento do projeto, visto que todas as linguagens de programação usadas no projeto são suportadas, possui um terminal integrado, o que facilitou a execução dos projetos do cliente e servidor de maneira simultânea, permite a personalização por parte do usuário, além de ter integração nativa com repositórios Git, é um sistema de controle de versão de arquivos. Através deles podemos desenvolver projetos na qual diversas pessoas podem contribuir simultaneamente no mesmo, editando e criando novos arquivos e permitindo que os mesmos possam existir sem o risco de suas alterações serem sobrescritas (SCHMITZ, 2015).

2.4.2 pgAdmin 4

O pgAdmin é uma plataforma *open source* de administração e desenvolvimento para PostgreSQL e seus sistemas de gerenciamento de banco de dados relacional. Desenvolvido em Python e jQuery, consegue suportar todos os recursos encontrados no PostgreSQL. Com o pgAdmin pode-se utilizar diversas funções, desde consultas SQL básicas a monitorar seus bancos de dados e configurar arquiteturas de banco de dados avançadas (DRAKE, 2018).

O pgAdmin foi usado neste projeto para administrar e desenvolver as tabelas do banco de dados PostgreSQL.

2.4.3 Insomnia REST

O Insomnia é uma ferramenta para projetar, depurar e testar APIs como um ser humano, não um robô, nele podemos criar solicitações, ver respostas e organizar espaços de trabalho, pastas, ambientes entre outras funções (INSOMNIA, 2020).

Neste projeto, o Insomnia foi utilizado para realizar os testes de inserção, atualização e visualização dos dados na linguagem Node.js durante o seu desenvolvimento.

3 METODOLOGIA

O presente trabalho é caracterizado por um estudo e pelo desenvolvimento de um protótipo de software mHealth para os profissionais da saúde do SADHUSM acessarem as informações dos pacientes atendidos, em tempo real, durante as visitas dos profissionais nas residências dos mesmos.

A fim de alcançar os objetivos aqui propostos e solucionar o problema de pesquisa, um protótipo de software foi desenvolvido para os profissionais do SADHUSM, esse software manterá os dados dos pacientes armazenados para que os profissionais possam ter um acesso mais rápido aos dados durante as consultas.

3.1 MODELAGEM DO PROJETO

O desenvolvimento do projeto começou com os levantamentos de requisitos funcionais, não funcionais e regras de negócio do sistema, após conversar com os profissionais da saúde do setor domiciliar do HUSM, o autor começou a entender o que precisaria ser feito para que fosse construído a base do projeto.

Posteriormente, foram construídos os diagramas de classe e de caso de uso, junto de um fluxograma de processo. Por fim, foi construído os protótipos de média e alta fidelidade das telas, para que se tivesse uma ideia de como construir as telas do aplicativo futuramente.

3.2 O BACK-END E O BANCO DE DADOS

Após a modelagem do projeto, começou o seu desenvolvimento. Foi escolhida a linguagem de programação JavaScript para o desenvolvimento do projeto, utilizando o software *open-source* NodeJS para construção lógica do projeto, junto do banco de dados PostgreSQL. Todavia, para administrar o banco de dados PostgreSQL foi utilizada uma plataforma *open-source* chamada pgAdmin 4 e para fazer os testes de inserção e checar se o Back-End está funcionando corretamente com o banco de dados foi utilizado o software Insomnia REST.

3.3 O FRONT-END

Logo que o desenvolvimento do Back-End foi concluído, começou a construção das telas do aplicativo, para isso foi utilizado a linguagem de programação TypeScript junto com o

framework React Native. Esta etapa do projeto acabou sendo desenvolvida mais rápida que o Back-End, pois não requer tantos testes e uma lógica muito complexa.

4 O PROJETO DO SISTEMA MHEALTH

A fim de alcançar os objetivos propostos neste trabalho e solucionar o problema de pesquisa, as seguintes etapas foram seguidas:

4.1 REQUISITOS FUNCIONAIS, NÃO FUNCIONAIS E REGRAS DE NEGÓCIO

Inicialmente foi realizado o levantamento e a definição dos requisitos funcionais e não funcionais que necessitam estar presentes no desenvolvimento do projeto. Os requisitos foram coletados em uma entrevista com a profissional da saúde do SADHUSM que falou sobre a dificuldade que todos do setor enfrentam.

Dentre os requisitos funcionais que o sistema deve possuir, destacam-se:

- Cadastros de profissionais de saúde, pacientes e prontuário;
- Alteração nos dados de profissionais e pacientes;
- Ativação e inativação de acessos dos profissionais de saúde;
- Listagens e consultas de dados cadastrados no sistema.

Já dentre os requisitos não funcionais que o sistema deve conter, destacam-se:

- Necessário acesso a internet para o sistema funcionar;
- O primeiro protótipo funcionará somente para Android, em versões futuras também em iOS;
- Autenticação de usuários;
- Design responsivo;

Como regras de negócio temos as seguintes:

- Somente um profissional poderá cadastrar outro profissional;
- Não poderá ser excluído ou alterado nenhum dado da evolução dos pacientes.

4.2 ELABORAÇÃO DE DIAGRAMAS ESTRUTURAIS E COMPORTAMENTAIS UML

Finalizada a definição dos requisitos funcionais e não funcionais, foi dado início a estruturação de diagramas da UML (*Unified Modeling Language*), uma linguagem que define uma série de artefatos que nos ajuda na tarefa de modelar e documentar os sistemas orientados a objetos que são desenvolvidos (RIBEIRO, 2012). Os diagramas definidos foram o de classes e o de casos de uso.

Começamos com o Diagrama de Classe desenvolvido para o software, seguindo a mesma regra de um prontuário médico, onde os dados não podem ser excluídos e ou atualizados. Também podemos por meio deste diagrama ver a estrutura do banco de dados, tendo 6 tabelas: Endereço, Telefone, Pessoa, Profissional, Paciente e Histórico do Paciente.

A tabela Endereço contém os dados de onde o paciente ou o profissional residem. No caso do paciente, torna mais fácil o acesso caso o profissional esteja em locomoção e precise ter acesso ao endereço do mesmo. Na tabela Telefone, encontram-se os telefones do paciente: número pessoal, residencial e de um familiar para contato.

Ambas as tabelas acima estão ligadas a tabela Pessoa, a qual contém informações de identificação do paciente ou profissional. Seguindo, temos as tabelas Profissional e Paciente que estão ligadas a tabela Pessoa. A tabela Profissional se diferencia da tabela do paciente apenas em alguns dados: Login e senha (Para terem acesso ao software, assim como podemos ver na tabela de autenticação no canto superior esquerdo da figura 1) e a área de atuação, que informa se o profissional é um médico(a), enfermeiro(a), nutricionista, etc. Na tabela do paciente, temos dois dados de extrema importância, o SAME (Serviço de Arquivamento Médico e Estatístico) e o CID (Classificação Estatística Internacional de Doenças e Problemas Relacionados com a Saúde). O SAME é o número que identifica o prontuário do paciente no sistema utilizado pelo hospital, e também nas pastas arquivadas fisicamente. O CID é o que informa aos médicos e demais profissionais quais doenças estão associadas ao paciente.

Por fim, temos a tabela Histórico do Paciente, onde estarão salvos todos os dados acrescentados pelos profissionais durante as visitas domiciliares, podendo ser visualizados futuramente por todos os profissionais.

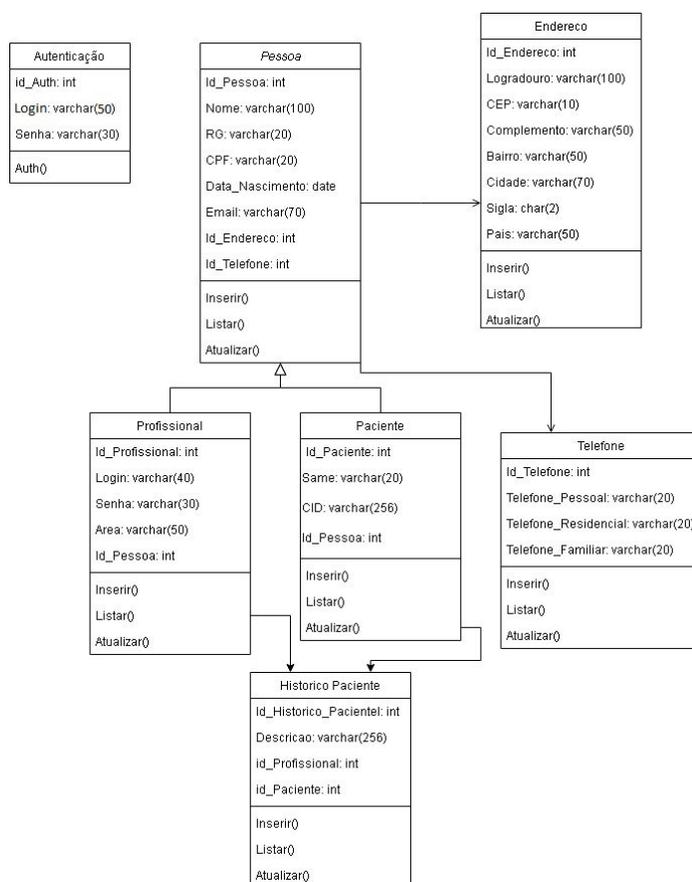


Figura 1. Diagrama de Classe.

Fonte: do Autor (2020)

Na figura 2 vemos um diagrama de caso de uso, aqui podemos ver o que os usuários do software poderão fazer de forma mais simplificada. Ambos os usuários, profissional e administrador, efetuam o login no software. O profissional pode cadastrar outro profissional ou um paciente, seguindo o profissional tem acesso a lista de pacientes, onde pode acessar os dados dos pacientes e visualizar os registros inseridos por outros profissionais nas últimas visitas domiciliares e ver se o paciente tem alguma pendência que precisa ser realizada e por fim também pode adicionar um novo registro no histórico do paciente.

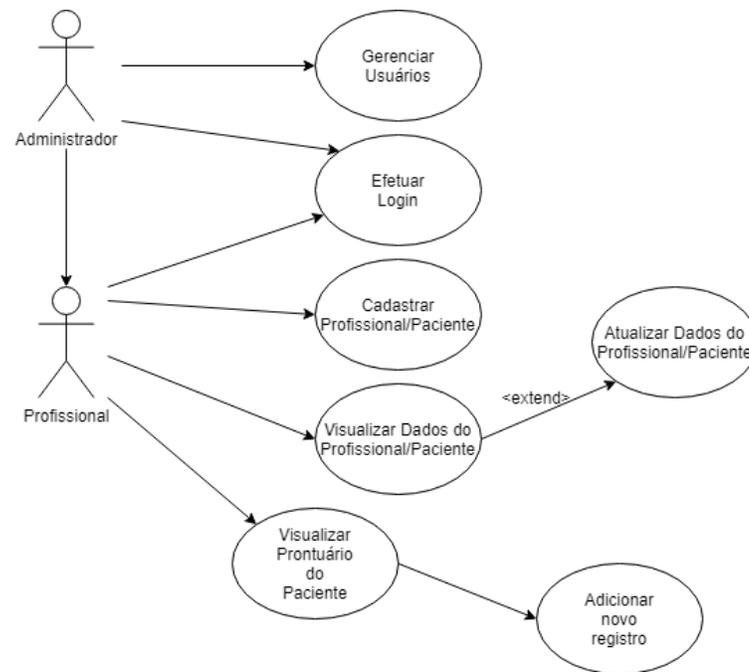


Figura 2. Diagrama de Caso de Uso.

Fonte: do Autor (2020)

O administrador, apesar de ter acesso a todos os recursos que o profissional tem, só será requisitado em caso de algum problema no software e para cadastrar algum profissional, se necessário. Não é recomendado ao administrador cadastrar pacientes ou adicionar registros no prontuário do paciente, apesar de que o sistema o permita fazê-los.

4.3 FLUXOGRAMA DE PROCESSO

Na Figura 3 vemos um fluxograma de processo de software, esse fluxograma é uma representação gráfica que descreve os passos e etapas sequenciais de um determinado processo. Muitos estudiosos emplacam o fluxograma na lista das ferramentas da qualidade (ALONÇO, 2017).

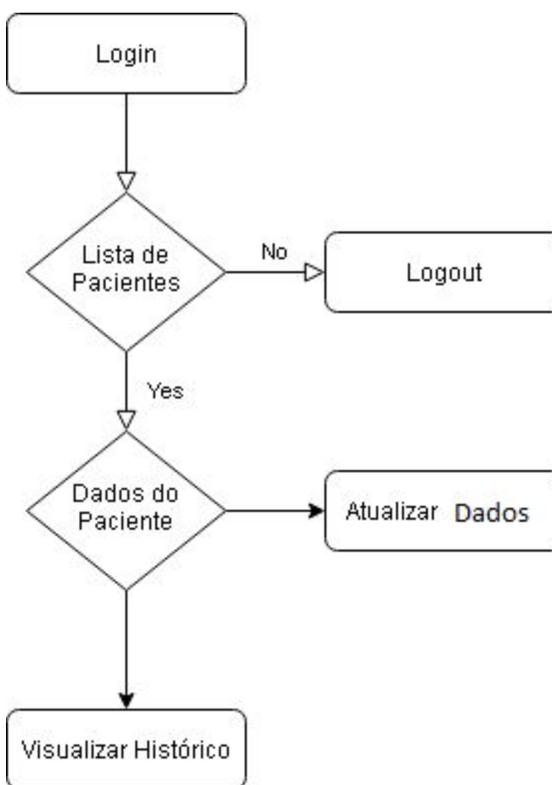


Figura 3. Fluxograma de Processo.

Fonte: do Autor (2020)

O profissional faz a autenticação no sistema usando suas credenciais de login e senha, seguido ele terá acesso a lista de pacientes que estão sendo atendidos atualmente pelo SADHUSM, ao escolher um paciente, pode-se acessar seus dados. A partir disso, o profissional pode ver também o histórico do paciente, onde os demais profissionais adicionaram informações em suas visitas, também pode atualizar os dados do paciente, tais como: RG, CPF, Nome, Endereço, Telefones e etc. E por fim, adicionar um novo registro no histórico de um paciente.

4.4 PROTÓTIPOS DAS TELAS

Na Figura 4 podemos ver o protótipo de média fidelidade da tela de login, onde o usuário realiza sua autenticação para ter acesso aos recursos do sistema.

SADHUSM

Login:
Senha:

Entrar

The image shows a login screen prototype. It features a large rectangular frame. At the top center, the text 'SADHUSM' is displayed. Below it, the labels 'Login:' and 'Senha:' are stacked vertically. At the bottom center, there is a rectangular button labeled 'Entrar'.

Figura 4. Protótipo de média fidelidade da tela de login.

Fonte: do Autor (2020)

Na Figura 5 temos o protótipo de média fidelidade da lista dos pacientes, onde o profissional já autenticado consegue ver os pacientes que estão sendo tratados atualmente pelo setor..

SADHUSM

NOME:	SAME:
1.	
2.	
3.	
4.	
5.	
6.	
7.	
8.	
9.	
10.	
11.	
12.	
13.	
14.	
15.	

The image shows a patient list screen prototype. It features a large rectangular frame. At the top center, the text 'SADHUSM' is displayed. Below it, there are two column headers: 'NOME:' and 'SAME:'. Underneath these headers is a list of 15 numbered items, from 1 to 15, arranged vertically.

Figura 5. Protótipo de média fidelidade da lista de pacientes.

Fonte: do Autor (2020)

Na Figura 7 podemos ver o protótipo de média fidelidade da tela onde mostra os dados pessoais do paciente, onde o profissional pode ver os dados mais relevantes do paciente, como SAME, CID, Nome, Endereço e demais informações.

SADHUSM	
NOME:	SAME:
RG:	CPF:
DATA DE NASCIMENTO:	
IDADE:	TELEFONE:
LOGRADOURO:	BAIRRO:
CIDADE:	UF:

Figura 7. Protótipo de média fidelidade dos dados do paciente.

Fonte: do Autor (2020)

Na Figura 8 temos os protótipos de alta fidelidade das telas, onde seguindo a mesma lógica dos protótipos de média fidelidade, as telas foram melhor definidas para que o processo de desenvolvimento das mesmas fossem baseados nesses modelos.

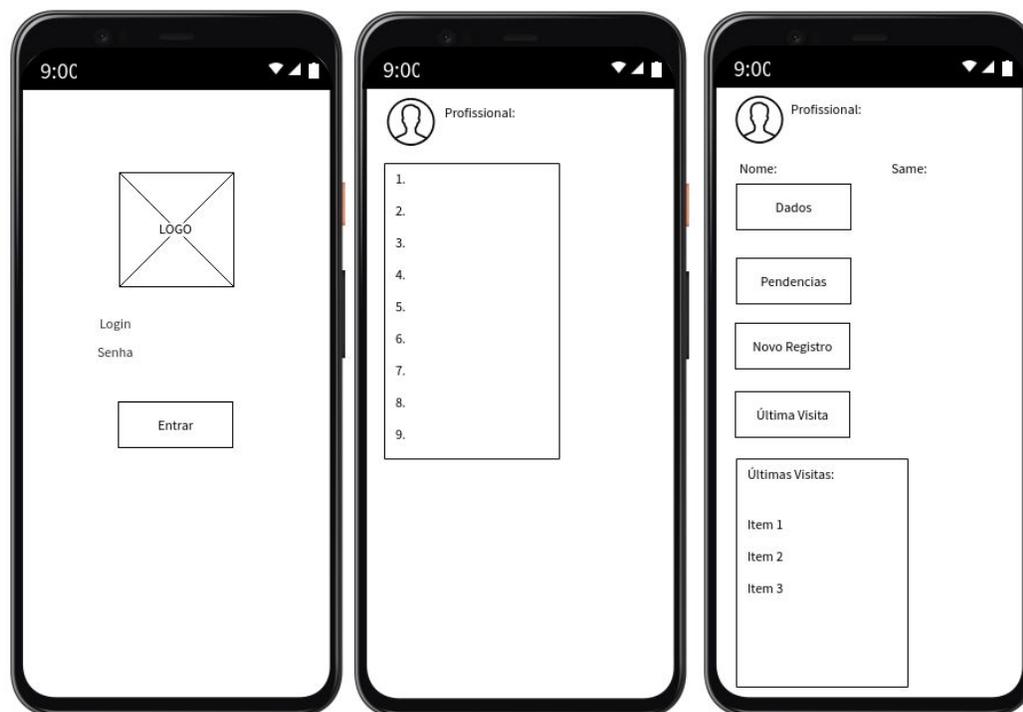


Figura 8. Protótipos de alta fidelidade das telas (Respectivamente da esquerda para direita: Tela de login, lista de pacientes e prontuário dos pacientes).

Fonte: do Autor (2020)

5 RESULTADOS

Conforme apresentado no presente trabalho, obteve-se um protótipo de aplicativo mHealth que atingiu os objetivos propostos, com funcionalidades de autenticação, uma lista atualizada dos pacientes que estão sendo tratados pelo setor, visualização do perfil do profissional, também dados dos pacientes, possibilidade de adicionar novos registros no prontuário bem como visualizar os últimos cadastrados.

Ao longo desta seção, são apresentadas as principais telas do software acompanhadas de uma breve descrição, onde pode-se observar o desenvolvimento de um layout simples e intuitivo, utilizando as cores do logotipo do SADHUSM (verde e branco).

5.1 LOGIN

A tela de login do usuário, mostrada na Figura 9, é onde o usuário que já foi cadastrado realiza o acesso e se autentica no sistema. Seu conteúdo possui o logotipo do SADHUSM e um formulário com dois campos, sendo o primeiro o login e o segundo a senha, também possui dois botões, o primeiro com a funcionalidade de submeter o formulário, e o segundo para redirecionar o usuário à tela onde poderá recuperar sua senha, caso a tenha esquecido.



Figura 9. Tela de Login.

Fonte: do Autor (2020)

5.2 LISTA DE PACIENTES

A lista de pacientes é o centro de toda a aplicação, nela podemos realizar o *logout*, cadastrar novos profissionais e pacientes, atualizar os dados do profissional ou dos pacientes e acessar o prontuário dos pacientes clicando em seus nomes.



Figura 10. Tela da lista de pacientes.

Fonte: do Autor (2020)

5.3 PERFIL DO PROFISSIONAL

O perfil do profissional informa todos os dados do profissional, desde seus dados pessoais, como Nome, CPF, RG, login no sistema, à área que o profissional atua, dados residenciais e seus números de telefone para contato.



Figura 11. Tela de perfil do profissional.

Fonte: do Autor (2020)

5.4 PERFIL DO PACIENTE

O perfil do paciente, assim como o do profissional, mostra os dados pessoais do paciente, mas diferente do profissional, aqui são exibidos o SAME e o CID. Por fim, há um botão “Evolução”, nome utilizado pelos profissionais para se referir ao prontuário, o qual apresenta a evolução do estado de saúde do paciente.



Figura 12. Tela de perfil do paciente.

Fonte: do Autor (2020)

5.5 EVOLUÇÃO

A evolução é a tela onde o profissional tem acesso aos dados inseridos pelos outros profissionais, seguido de um espaço para inserir texto. Após submetido, através do botão “Enviar”, esse novo registro será adicionado ao prontuário do paciente.

← Voltar

Evolução

Paciente refere dor no peito, falta de ar, pouco apetite e cansaço. PA: 137/96 Pulso: 72

Novo Registro

Enviar

Figura 13. Tela da evolução.

Fonte: do Autor (2020)

5.6 CADASTRO DE PROFISSIONAL

A figura 14 apresenta a tela de cadastro de profissionais. Outro profissional, após ter se autenticado no sistema, tem acesso para poder cadastrar um novo profissional, informando todos os dados pessoais solicitados, juntamente da área de atuação, login, senha e um campo extra para confirmação da senha.

A imagem mostra a interface de usuário de um aplicativo para o cadastro de profissionais. No topo, há uma barra de status com ícones de notificação, bateria, Wi-Fi, sinal de rede e o horário 20:32. Abaixo, um botão de navegação com uma seta para a esquerda e o texto "Voltar". O título centralizado da tela é "Cadastro do Profissional". Abaixo do título, há sete campos de entrada de texto, cada um com um ícone de casa à esquerda e o seguinte rótulo: "Número", "Logradouro", "CEP", "Complemento", "Bairro", "Cidade", "Estado" e "País". Na base da tela, há uma barra de navegação padrão do Android com ícones de voltar, home e avançar.

Figura 14. Tela de cadastro do profissional parte 1.

Fonte: do Autor (2020)

Na Figura 15 temos a continuação da figura 14, mostrando os dados que faltaram na figura anterior e a parte de adicionar o login, senha e área do profissional.



The image shows a mobile application interface for professional registration. It features a vertical stack of ten green input fields, each with a person icon on the left and a label on the right. The labels are: CPF, RG, Email, Data de Nascimento, Gênero, Login, Senha, Confirmar Senha, and Área. Below these fields is a green button with the text 'Cadastrar'. The top of the screen shows a status bar with the time 20:32 and various icons. At the bottom, there is a black navigation bar with three white icons: a back arrow, a circle, and a square.

Figura 15. Tela de cadastro do profissional parte 2.

Fonte: do Autor (2020)

5.7 CADASTRO DE PACIENTE

A figura 16 mostra a tela de cadastro de pacientes, na qual um profissional estando autenticado no sistema pode cadastrar um novo paciente, informando todos os dados pessoais solicitados, também o SAME e o CID do paciente.



The image shows a mobile application interface for patient registration. At the top, there is a green header bar with a back arrow and the text "Voltar". Below the header, the title "Cadastro do Paciente" is centered. The main content consists of eight green rounded rectangular input fields, each with a house icon on the left and a label on the right. The labels are: "Número", "Logradouro", "CEP", "Complemento", "Bairro", "Cidade", "Estado", and "Pais". At the bottom of the screen, there is a black navigation bar with three white icons: a back arrow, a circle, and a square.

Figura 16. Tela de cadastro do paciente parte 1.

Fonte: do Autor (2020)

Na Figura 17 temos a continuação do cadastro de paciente, onde mostra os dados que faltaram da figura anterior junto com o SAME e CID do paciente.



The image shows a mobile application interface for patient registration. The screen is dark green with white text. At the top, there is a status bar with icons for signal, Wi-Fi, battery, and the time 20:31. Below the status bar is a header bar with a back arrow and the text 'Paciente - Cadastro'. The main content area consists of eight input fields, each with a person icon on the left and a label on the right: 'Nome', 'CPF', 'RG', 'Email', 'Data de Nascimento', 'Gênero', 'SAME', and 'CID'. At the bottom of the form is a large green button with the text 'Cadastrar'. The bottom of the screen shows the Android navigation bar with back, home, and recent apps icons.

Figura 17. Tela de cadastro do paciente parte 2.

Fonte: do Autor (2020)

5.8 ATUALIZAÇÃO DO PROFISSIONAL

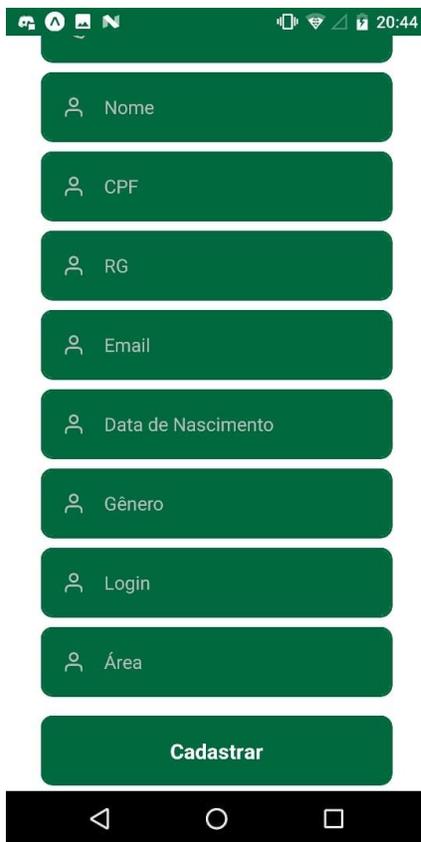
A figura 18 demonstra a tela de atualização dos dados do profissional, que está autenticado, preenche um ou mais dados que deseja atualizar.



Figura 18. Tela de atualização do profissional parte 1.

Fonte: do Autor (2020)

Na Figura 19 vemos a continuação da tela de atualização do profissional, onde podemos alterar qualquer dado pessoal do profissional, seu login ou sua área de atuação, somente a senha não pode ser alterada nesta tela em comparação com a figura 15.



The image shows a mobile application interface for updating professional data. It features a vertical stack of eight green input fields, each with a person icon on the left. The fields are labeled: Nome, CPF, RG, Email, Data de Nascimento, Gênero, Login, and Área. Below these fields is a green button labeled 'Cadastrar'. The top of the screen shows a status bar with icons for signal, Wi-Fi, battery, and the time 20:44. The bottom of the screen shows a black navigation bar with three white icons: a back arrow, a circle, and a square.

Figura 19. Tela de atualização do profissional parte 2.

Fonte: do Autor (2020)

5.9 ATUALIZAÇÃO DO PACIENTE

A figura 20 exibe a tela de atualização dos dados do paciente, onde o profissional autenticado no sistema pode alterar um ou mais dados do paciente.



Figura 20. Tela de atualização do paciente parte 1.

Fonte: do Autor (2020)

Na Figura 21 vemos a continuação da tela de atualização do paciente, onde o profissional pode somente alterar os dados pessoais do paciente caso tenha informado algo errado, ou atualizar o CID do paciente.



The image shows a mobile application interface for updating patient data. At the top, there is a status bar with icons for signal, Wi-Fi, and battery, and the time 20:43. Below the status bar is a dark green header with a back arrow and a title. The main content area contains a list of input fields, each with a person icon on the left and a label on the right: Nome, CPF, RG, Email, Data de Nascimento, Gênero, SAME, and CID. Below these fields is a large green button with the text 'Atualizar'. At the bottom, there is a black navigation bar with three icons: a back arrow, a circle, and a square.

Figura 21. Tela de atualização do paciente parte 2.

Fonte: do Autor (2020)

6 CONSIDERAÇÕES FINAIS

Este trabalho foi desenvolvido com o intuito de criar um sistema mHealth para o Serviço de Atenção Domiciliar do Hospital Universitário de Santa Maria, possibilitando que esse setor otimize a troca de informação entre seus profissionais, tornando seus atendimentos mais precisos e eficazes. Conforme apresentado, atualmente o setor utiliza papel para registrar as informações coletadas nos atendimentos, considerando o distanciamento geográfico dos mesmos, torna a troca de informações entre os profissionais um desafio.

Buscando solucionar esse problema, foi desenvolvido um protótipo de um aplicativo móvel que possibilita a troca de informações em tempo real, visualizando, inserindo e atualizando os dados pessoais dos pacientes, bem como acesso às evoluções adicionadas por cada profissional do setor.

Após a finalização da fase de testes do protótipo, será disponibilizada uma versão beta para os profissionais do SADHUSM, para que os mesmos possam se familiarizar com o sistema e também apontar possíveis erros e melhorias.

Como trabalhos futuros, será feita uma versão completa e entregue gratuitamente para os profissionais do SADHUSM. Essa versão será constantemente atualizada até ficar com um layout mais intuitivo e melhor que os apresentados no protótipo desse trabalho e também com novas funções que possam facilitar ainda mais os atendimentos dos profissionais de saúde do SADHUSM.

7 REFERÊNCIAS

- ALONÇO, Guilherme. O que é Fluxograma de Processos, Templum, 2017. Disponível em: <<https://certificacaoiso.com.br/o-que-e-fluxograma-de-processos/>>. Acesso em: 29 nov. 2020.
- BAIG, M. M.; GHOLAMHOSSEINI, H.; CONNOLLY, M. J. Mobile healthcare applications: system design review, critical issues and challenges. *Australasian Physical & Engineering Sciences in Medicine*, v. 38, n. 1, p. 23–38, 2015.
- BECKER, Lauro. O que é React Native?. *Orgânica Digital*, 2020. Disponível em: <<https://www.organicadigital.com/blog/o-que-e-react-native/>>. Acesso em: 28 nov. 2020.
- DB-ENGINES. Db-Engines. DB-Engines Ranking - Trend Popularity, 2019. Disponível em: <https://db-engines.com/en/ranking_trend>. Acesso em: 29 nov. 2020.
- DEHLING, T. et al. Exploring the Far Side of Mobile Health: Information Security and Privacy of Mobile Health Apps on iOS and Android. *JMIR mHealth and uHealth*, v. 3, n. 1, p. e8, 2015.
- DENNISON, L. et al. Opportunities and challenges for smartphone applications in supporting health behavior change: qualitative study. *Journal of medical Internet research*, v. 15, n. 4, p. 1–12, 2013.
- DIONISIO, Edson. Introdução ao Visual Studio Code. *DevMedia*, 2016. Disponível em: <<https://www.devmedia.com.br/introducao-ao-visual-studio-code/34418>>. Acesso em: 29 nov. 2020.
- DRAKE, Mark. Como instalar e configurar o pgAdmin 4 no modo servidor. *DigitalOcean*, 2018. Disponível em: <<https://www.digitalocean.com/community/tutorials/como-instalar-e-configurar-o-pgadmin-4-no-modo-servidor-pt#:~:text=O%20pgAdmin%20%C3%A9%20uma%20plataforma,os%20recurso%20encontrados%20no%20PostgreSQL.>>. Acesso em: 29 nov. 2020.
- DRESCHER, Rhuan Matheus. Desenvolvimento de sistema de ouvidoria para a Antonio Meneghetti Faculdade. Disponível em: <http://repositorio.faculdadeam.edu.br/xmlui/bitstream/handle/123456789/646/TCC_SI_RHUA_N_DRESCHER_AMF_2019.pdf?sequence=1&isAllowed=y>. Acesso em: 27 nov. 2020.
- EYSENBACH, G. What is e-health? *Journal of Medical Internet Research*, v. 3, n. 2, p. 1–5, 2001a.
- EYSENBACH, G. What is e-health ? v. 3, p. 2–3, 2001b.
- FERNANDES, Diego. TypeScript: vantagens, mitos, dicas e conceitos fundamentais. *Rocketseat*, 2018. Disponível em: <<https://blog.rocketseat.com.br/typescript-vantagens-mitos-conceitos/>>. Acesso em: 29 nov. 2020.

GUIMARÃES, C. M. S; IMAMURA, M. E; RICHTER, S; MONTEIRO, J. C; Amamentação e tecnologias mHealth: análise dos aplicativos móveis para tablets e smartphones. Rev. Eletr. Enf. [Internet]. 2018. Disponível em: <<https://doi.org/10.5216/ree.v20.48578>>. Acesso em: 18 nov. 2020.

ICLINIC. Mobile health: a tecnologia que está revolucionando a saúde. Disponível em <<https://blog.iclinic.com.br/mobile-health-conheca-a-tecnologia-que-esta-revolucionando-a-saude/>>. Acesso em: 21 dez. 2020.

INSOMNIA. Insomnia, 2020. Disponível em: <<https://insomnia.rest/>>. Acesso em: 29 nov. 2020.

KEMP, Simon. Digital in 2017: Global Overview. Disponível em: <<https://wearesocial.com/special-reports/digital-in-2017-global-overview>>. Acesso em: 29 nov. 2020.

KUMAR, S. et al. Mobile Health : p. 28–35, 2013.

LETSCODE. Ranking das linguagens de programação mais populares de acordo com o GitHub. Disponível em: <<https://letscode.com.br/blog/ranking-das-linguagens-de-programacao-mais-populares-de-acordo-com-github>>. Acesso em: 28 nov. 2020.

MACHADO, Ana Paula. Desenvolvimento de aplicativo para pacientes com Transtorno do Espectro do Autismo História Social: Indo ao Dentista. Disponível em: <<https://repositorio.ufsc.br/bitstream/handle/123456789/187350/TCC%20Ana%20P%20Machado.pdf?sequence=1&isAllowed=y>>. Acessado em: 12 jun. 2020.

MDN. JavaScript. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>>. Acesso em: 28 nov. 2020.

MDN. Introdução Express/Node. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Learn/Server-side/Express_Nodejs/Introdu%C3%A7%C3%A3o>. Acesso em: 29 nov. 2020.

MELLO, C. M. C.; SGANZERLA, M. A. R. Aplicativo android para auxiliar no que?. p. 231–239, 2013.

MONQUEIRO, J. C. B. Programação Orientada a Objetos: uma introdução. Hardware, 2007. Disponível em: <<https://www.hardware.com.br/artigos/programacao-orientada-objetos/>>. Acesso em: 27 nov. 2020.

MORSCH, José Aldair. Mobile Health: a invasão do mobile na saúde, 2016. Disponível em: <<https://telemedicinamorsch.com.br/blog/mhealth-a-invasao-do-mobile-na-saude>>. Acesso em: 21 dez. 2020.

OH, H. et al. What Is eHealth (3): A Systematic Review of Published Definitions Corresponding Author : v. 7, n. 3, p. 1–12, 2005.

RIBEIRO, Leandro. O que é UML e Diagramas de Caso de Uso: Introdução prática à UML, 2012. Disponível em: <<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/>>. Acesso em: 29 nov. 2020.

SCHMITZ, Daniel. Tudo que você queria saber sobre Git e Github, mas tinha vergonha de perguntar, 2015. Disponível em: <<https://tableless.com.br/tudo-que-voce-queria-saber-sobre-git-e-github-mas-tinha-vergonha-de-perguntar/>>. Acesso em: 29 nov. 2020.

SCHNEIDER, G. P.; SHIPP, L. Ninth Edition. [s.l: s.n.].

SILVA, M. M. DA; SANTOS, M. T. P. Os Paradigmas de Desenvolvimento de Aplicativos para Aparelhos Celulares. T.I.S - Tecnologias, Infraestrutura e Software - UFSCar, v. 3, n. 2, p. 162–170, 2014.

SOUZA, Ivan. PostgreSQL: saiba o que é, para que serve e como instalar. Rockcontent, 2020. Disponível em: <<https://rockcontent.com/br/blog/postgresql/>>. Acesso em: 29 nov. 2020.

WORLD HEALTH ORGANIZATION. mHealth: New horizons for health through mobile technologies. Observatory, v. 3, n. June, p. 66–71, 2011.