



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS FLORIANÓPOLIS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Maicon Francisco

**Desenvolvimento de uma Plataforma Web Multiusuário Utilizando Protocolo HL7
FHIR para Visualização, Análise e Acompanhamento de Dados de Indivíduos com
Diabetes Mellitus**

Florianópolis

2022

Maicon Francisco

Desenvolvimento de uma Plataforma Web Multiusuário Utilizando Protocolo HL7 FHIR para Visualização, Análise e Acompanhamento de Dados de Indivíduos com Diabetes Mellitus

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina para a obtenção do título de mestre em Engenharia Elétrica.

Orientador: Prof. Jefferson Luiz Brum Marques, PhD.

Florianópolis

2022

Ficha de identificação da obra

Francisco, Maicon

Desenvolvimento de uma Plataforma Web Multiusuário Utilizando Protocolo HL7 FHIR para Visualização, Análise e Acompanhamento de Dados de Indivíduos com Diabetes Mellitus / Maicon Francisco ; orientador, Jefferson Luiz Brum Marques, 2022.

106 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia Elétrica, Florianópolis, 2022.

Inclui referências.

1. Engenharia Elétrica. 2. HL7 FHIR. 3. Interoperabilidade. 4. Saúde 4.0. 5. Diabetes Mellitus. I. Marques, Jefferson Luiz Brum. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

Maicon Francisco

Desenvolvimento de uma Plataforma Web Multiusuário Utilizando Protocolo HL7 FHIR para Visualização, Análise e Acompanhamento de Dados de Indivíduos com Diabetes Mellitus

O presente trabalho em nível de mestrado foi avaliado e aprovado por banca examinadora composta pelos seguintes membros:

Profa. Andrea Teresa Riccio Barbosa, Dra.
Universidade Federal do Mato Grosso do Sul (UFMS)

Prof. José Marino Neto, Dr.
Universidade Federal de Santa Catarina (UFSC)

Certificamos que esta é a **versão original e final** do trabalho de conclusão que foi julgado adequado para obtenção do título de mestre em Engenharia Elétrica.

Prof. Telles Brunelli Lazzarin, Dr.
Coordenador do Programa de Pós-Graduação

Prof. Jefferson Luiz Brum Marques, PhD
Orientador

Florianópolis, 2022.

Este trabalho é dedicado aos meus queridos pais, irmãos, amigos e minha amada noiva.

AGRADECIMENTOS

Agradeço primeiramente aos meus pais, Walmor e Clarice, que sempre batalharam e incentivaram eu e meus irmãos a terem uma boa educação, e aqui me apoiaram em mais uma conquista na vida acadêmica para me tornar mestre.

Minha imensa gratidão aquela faz tudo tornar possível, minha noiva e amada Suzana, por me auxiliar em todos os momentos dessa conturbada caminhada, tendo compreensão, carinho e muita paciência.

Aos meus irmãos, Clailton e Larissa, que me incentivaram em seguir na vida acadêmica, e que nem mesmo a distância diminui nosso companheirismo.

Ao meu orientador, Jefferson Luiz Brum Marques, pelas discussões e orientações, não apenas deste trabalho, mas sobre minha vida acadêmica e profissional.

Aos meus amigos da Família JJ, que sempre me apoiaram e incentivaram nos meus estudos, estando sempre ao meu lado proporcionando momentos de descontração, e que mesmo em período de pandemia encontramos nosso jeito de manter nossos elos de amizade. Em especial ao Guilherme, que me deu suporte em diversos pontos técnicos deste trabalho.

Ao colega do Instituto de Engenharia Biomédica (IEB) Igor, que me auxiliou no entendimento e desenvolvimento do tema deste trabalho.

À CAPES pelo incentivo e apoio financeiro.

E por fim agradeço a Deus, por me dar forças para realizar todas as minhas conquistas, as quais só são possíveis com as pessoas incríveis que colocou na minha vida.

“Se a história de fadas é digna de ser lida, então é digna de ser escrita e lida por adultos. É claro que elas acrescentarão mais e extrairão mais do que as crianças são capazes de lidar. Então, como um ramo de arte genuína, as crianças poderão desejar ler histórias de fadas adequadas para elas e que estejam ao seu alcance, bem como poderão desejar receber introduções apropriadas à poesia, à história e às ciências [...]. Seus livros, como suas roupas, devem dar espaço para crescer e, no mínimo, devem estimular o crescimento.” (J. J. R. TOLKIEN, Sobre Histórias de Fadas, 2010)

PRODUÇÃO ACADÊMICA

FRANCISCO, M.; MARQUES, J. L. B. **Use of HAPI FHIR server and development of a multi-user Web Interface for Visualization and Analysis of data from patients with Diabetes Mellitus.** IX Congresso Latino Americano de Engenharia Biomédica e XXVII Congresso Brasileiro de Engenharia Biomédica, 2022.

FRANCISCO, M.; VIEIRA, I. O.; NETO, J. M.; OJEDA, R. G.; MARQUES, J. L. B. **Aplicação Web em Saúde 4.0: Uma Plataforma Multiusuário Baseada em um Servidor HAPI FHIR.** XIII Simpósio de Engenharia Biomédica, 2021.

VIEIRA, I. O.; FRANCISCO, M.; NETO, J. M.; OJEDA, R. G.; MARQUES, J. L. B. **Proposta de um Servidor de Interoperabilidade de Dados em Pesquisa Clínica com Padrão HL7 FHIR.** XIII Simpósio de Engenharia Biomédica, 2021.

RESUMO

O avanço tecnológico da indústria é marcado por tendências que marcam sua época. Atualmente discute-se se é vivenciado uma nova revolução industrial, a chamada Indústria 4.0, marcada pela conectividade de sistemas, o monitoramento remoto e a automação. A tecnologia em saúde acompanha estas tendências, resultando na chamada Saúde 4.0, que busca solucionar o desafio de conectar pacientes, equipamentos médicos, profissionais, empresas e todos os setores da saúde em uma rede compartilhada, integrada e segura. A interoperabilidade é uma metodologia que atende esses desafios, permitindo que sistemas independentes tenham a capacidade de trocar informações. Essa demanda resulta em padrões para interoperabilidade nos sistemas informatizados em saúde, como o *Fast Healthcare Interoperability Resources* (FHIR) desenvolvido pela *Health Level Seven* (HL7). Este padrão tem se tornado amplamente utilizado e possui diversas aplicações na saúde, apresentando versatilidade de implementação e tecnologias de padrão *web*. Estas soluções podem resultar em diversos benefícios na área da saúde, uma vez que os sistemas de saúde carecem de uma plataforma bem estruturada e que apresente capacidade de integrar e compartilhar informações e dados de indivíduos entre todos os atores envolvidos na gestão de doenças crônicas, como o *Diabetes Mellitus* (DM). O DM apresenta um aumento significativo de casos no mundo e precisa de um monitoramento e acompanhamento efetivo para o controle da doença. Este trabalho apresenta o desenvolvimento de uma plataforma *web* para contribuir na solução das dificuldades para um acompanhamento acurado de pacientes com DM. Esta solução é estruturada pensando nos desafios de interoperabilidade entre sistemas, armazenando e trafegando dados através da implantação de um servidor HAPI FHIR complementado por um servidor Elasticsearch para requisições mais complexas que um servidor FHIR não é estruturado para realizar. A análise e visualização desses dados é apresentada através de uma plataforma *web*, que é incorporada ao projeto *Screening for Diabetes Complications* (SDC-X) do Instituto de Engenharia Biomédica da Universidade Federal de Santa Catarina (IEB-UFSC), onde o conceito de Saúde 4.0 é incorporado em todo o seu desenvolvimento, entregando uma solução para coleta, armazenamento, análise e apresentação para o acompanhamento de dados e informações de indivíduos com DM. A plataforma é desenvolvida com base na arquitetura *Model View Controller* (MVC) em PHP, e apresenta grande fluidez e modularidade na sua construção. Ela apresenta acessos para cinco perfis diferentes de usuários, apresentando as informações associadas a cada perfil em um formato direcionado conforme suas permissões e autorizações implementadas pelo sistema. Automações e protocolos de segurança foram aplicados à interface para atender as diretrizes da Lei Geral de Proteção de Dados (LGPD) vigente no Brasil. O trabalho desenvolvido complementa o conceito de tecnologia em Saúde 4.0 aplicada ao projeto SDC-X, que atende todas as diretrizes ao incorporar o paradigma ubíquo desenvolvido ao projeto.

Palavras-chave: Diabetes Mellitus. Saúde 4.0. HL7 FHIR. Arquitetura MVC. Interoperabilidade. Plataforma Web. Sistema de Informação em Saúde. Engenharia Biomédica.

ABSTRACT

The technological advance of the industry is marked by trends that mark its time. Currently, it is discussed whether a new revolution is being experienced, the so-called Industry 4.0, characterised by systems connectivity, remote monitoring and automation. Health technology follows these trends, resulting in the so-called Health 4.0, which seeks to solve the challenge of connecting patients, medical equipment, professionals, companies and all health sectors in a shared, integrated and secure network. Interoperability is a methodology that addresses these challenges, allowing independent systems to have the ability to exchange information. This demand results in standards for interoperability in healthcare information systems, such as the Fast Healthcare Interoperability Resources (FHIR) developed by Health Level Seven (HL7). This standard has become widely used and has several applications in health, presenting implementation versatility and web standard technologies. These solutions can result in several benefits in the healthcare area since health systems lack a well-structured platform that can integrate and share information and data from individuals among all the sectors involved in managing chronic diseases, such as diabetes mellitus (DM). DM shows a significant global increase and needs monitoring and follow-up tools to manage the disease. This work presents the development of a web platform to contribute to solving the difficulties for an accurate follow-up of individuals with DM. This solution is structured thinking about the interoperability challenges between systems, storing and transporting data through implementing a HAPI FHIR server complemented by an Elasticsearch server for more complex requests that an FHIR server is not structured to perform. The analysis and visualisation of these data are carried out through a web platform, which implements the Screening for Diabetes Complications (SDC-X) project of the Institute of Biomedical Engineering of the Federal University of Santa Catarina (IEB-UFSC). In the SDC-X platform, the concept of Health 4.0 is incorporated throughout its development, delivering a solution for collection, storage, analysis and presentation for tracking and managing data and information from patients with DM. The platform is developed based on the Model View Controller (MVC) architecture in PHP and has excellent fluidity and modularity in its construction. It presents accesses for five different user profiles, presenting the information for each kind of user in a format directed according to their permissions and authorisations implemented by the system. Automation and security protocols were applied to the interface to meet Brazil's General Data Protection Law (LGPD) guidelines. The work developed complements the concept of technology in Health 4.0 used in the SDC-X project, which meets all the procedures by incorporating the ubiquitous paradigm developed in the project.

Keywords: Diabetes Mellitus. Health 4.0. HL7 FHIR. MVC design pattern. Interoperability. Web Platform. Health Information System. Biomedical engineering.

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 - Evolução das Revoluções Industriais. | 26 |
| Figura 2. Rede de Saúde inteligente interligada pela “ <i>Fog computing</i> ” de cada sistema de saúde. O termo “ <i>Fog computing</i> ” compõe a arquitetura de computação entre a fonte de dados e a nuvem. | 28 |
| Figura 3. Formato de acesso as informações usando FHIR. | 33 |
| Figura 4. Estruturação do formato de um Recurso em XML composto pelos metadados, narrativa, extensões e os elementos. | 35 |
| Figura 5. Camadas dos Recursos do FHIR. | 37 |
| Figura 6 – Objetivo de conexão da RNDS entre todas as entidades de saúde do Brasil, como uma plataforma de inovação, informação e serviços digitais em saúde. | 40 |
| Figura 7 – Módulos do projeto SDC-X. O Módulo de Aquisição é composto por <i>hardware</i> , composto por equipamentos para realizar medidas fisiológicas, e os outros módulos são essencialmente <i>softwares</i> | 46 |
| Figura 8 – Etapas dos módulos do projeto SDC-X. | 47 |
| Figura 9 - URL para acessar os dados dos recursos do servidor HL7 FHIR. | 50 |
| Figura 10 - URL para acessar os dados de um documento do servidor <i>ElasticSearch</i> | 51 |
| Figura 11 - Diagrama da realização de requisições RESTful para os servidores HAPI FHIR e <i>ElasticSearch</i> | 52 |
| Figura 12 - Diagrama de blocos do modelo de projeto MVC. | 54 |
| Figura 13. Estruturação de pastas do projeto desenvolvido. | 59 |
| Figura 14 - Tela inicial da plataforma de acesso público. a) Menus de navegação. b) Versão da plataforma. c) Local de apresentação das informações. d) Botão de login para acessar a plataforma. | 60 |
| Figura 15 – Formato do roteamento de endereço da plataforma. | 61 |
| Figura 16 – Endereço para acessar a lista de pacientes após ter realizado login na plataforma. | 61 |
| Figura 17 – Exemplo de acesso ao perfil do usuário de <i>id 23</i> | 61 |
| Figura 18 – Página para erro 404 e página não encontrada caso o endereço procurado não exista nenhuma classe correspondente de <i>Controller</i> | 62 |
| Figura 19 - Página de login e senha para o usuário acessar a Plataforma SDC-X. | 63 |
| Figura 20 - Diagrama de relação entre as tabelas do banco de dados referentes ao usuário. | 64 |

| | |
|---|----|
| Figura 21 – Fluxograma dos recursos utilizados para atender o usuário Pesquisador. | 65 |
| Figura 22 - Tela da listagem de pacientes para o perfil Pesquisador. | 66 |
| Figura 23 - Visualização gráfica dos resultados de exames do paciente. | 68 |
| Figura 24 - Lista de exames realizados pelo paciente. | 68 |
| Figura 25 - Lista de medicamentos utilizados pelo paciente. | 69 |
| Figura 26 - Resultado de exame de Lipídios gerado a partir do “id” do exame selecionado. ... | 69 |
| Figura 27 - Fluxograma dos recursos utilizados para atender o usuário Profissional. | 70 |
| Figura 28 - Página de visualização da lista de pacientes pelo perfil Profissional. | 71 |
| Figura 29 - Página de visualização de dados de paciente através do perfil Profissional. | 72 |
| Figura 30 - Fluxograma dos recursos utilizados para atender o usuário Gestor. | 73 |
| Figura 31 – Tela de Dados Epidemiológicos acessada pelo perfil Gestor. | 74 |
| Figura 32 – Gráficos e informações visualizados pelo perfil Gestor. | 74 |
| Figura 33 - Fluxograma dos recursos utilizados para atender o usuário Paciente. | 75 |
| Figura 34 - Fluxograma dos recursos utilizados para atender o usuário Administrador. | 76 |
| Figura 35 - Tela de aplicação de regra para bloqueio de visualização de dados de paciente para um profissional. | 77 |
| Figura 36 - Visualização da tela de exame para dados reais obtidos pelo SDC-Cardio. | 79 |
| Figura 37 - Visualização da tela de exame para dados reais obtidos pelo SDC-Esc. | 79 |
| Figura 38. Projeto SDC-X tem capacidade de integrar algumas das frentes propostas pela RNDS como Suporte a Linhas de Cuidado, Serviços de Informação e Alerta, Profissionais de Saúde, Gestores e Centros de Pesquisa e Desenvolvimento. | 85 |

LISTA DE QUADROS

| | |
|--|----|
| Quadro 1 – Respostas de status de requisições para uma arquitetura RESTful. | 51 |
|--|----|

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 – Funções e permissões de cada perfil de usuário da plataforma. | 55 |
|---|----|

LISTA DE ABREVIATURAS E SIGLAS

ABS Atenção Básica de Saúde
API *Application Programming Interface*
COVID-19 *Coronavirus Disease 2019*
DM *Diabetes Mellitus*
DG Diabete Gestacional
ECG Eletrocardiograma
FHIR *Fast Healthcare Interoperability Resources*
GDC *Global Diabetes Compact*
HL7 *Health Level Seven*
HTML *HiperText Markup Language*
HTTP *Hypertext Transfer Protocol*
IEB Instituto de Engenharia Biomédica
IoT *Internet of Things*
JSON *JavaScript Object Notation*
LGPD Lei Geral de Proteção de Dados
LOINC *Logical Observation Identifiers, Names and Codes*
MS Ministério da Saúde
MVC *Model-View-Controller*
ND Neuropatia Diabética
OSI *Open Systems Interconnection*
PHP *Hypertext Preprocessor*
PNISS Política Nacional de Informação e Informática em Saúde
RD Retinopatia Diabética
REST *Representational State Transfer*
RNDS Rede Nacional de Dados em Saúde
SDC *Screening for Diabetes Complications*
SNOMED CT *Systematized Nomenclature of Medicine Clinical Terms*
SQL *Standard Query Language*
SUS Sistema Único de Saúde
UFSC Universidade Federal de Santa Catarina
UML *Unified Modeling Language*

URL *Uniform Resource Locator*

VFC *Variabilidade da Frequência Cardíaca*

XHTML *Extensible Hypertext Markup Language*

XML *Extensible Markup Language*

WHO *World Health Organization*

SUMÁRIO

| | | |
|--------------|--|-----------|
| 1 | INTRODUÇÃO | 21 |
| 1.1 | Motivação e Justificativa | 22 |
| 1.2 | OBJETIVOS | 23 |
| 1.2.1 | Objetivo Geral..... | 23 |
| 1.2.2 | Objetivos Específicos | 24 |
| 2 | Referencial Teórico..... | 25 |
| 2.1 | Indústria 4.0 em Saúde | 25 |
| 2.2 | Área da Saúde e Saúde 4.0 | 27 |
| 2.3 | Áreas e Aplicação | 29 |
| 3 | Interoperabilidade | 29 |
| 4 | <i>Health Level Seven (HL7)</i>..... | 30 |
| 4.1 | Versão HL7 v2..... | 31 |
| 4.2 | Versão HL7 v3..... | 31 |
| 4.3 | <i>HL7 Fast Healthcare Interoperability (HL7 FHIR)</i> | 32 |
| 4.3.1 | <i>Representational State Transfer (REST)</i> | 33 |
| 4.3.2 | Recursos (“<i>Resources</i>”) FHIR | 34 |
| 4.3.3 | Ferramentas de Aplicação | 38 |
| 5 | Rede Nacional de Dados em Saúde | 39 |
| 6 | Diabetes Mellitus..... | 41 |
| 6.1 | Tipos de Diabetes | 42 |
| 6.1.1 | Diabetes Tipo 1..... | 42 |
| 6.1.2 | Diabetes Tipo 2..... | 42 |
| 6.1.3 | Diabetes Gestacional..... | 43 |
| 6.1.4 | Outros Tipos de Diabetes | 44 |
| 7 | Trabalhos Relacionados | 44 |
| 8 | Screening for Diabetes Complications | 45 |

| | | |
|-----------------|---|-----------|
| 9 | Materiais e Métodos | 49 |
| 9.1 | Servidores FHIR e <i>ElasticSearch</i> | 49 |
| 9.1.1 | Interface RESTFUL API | 50 |
| 9.2 | Inclusão de Dados aos Servidores | 52 |
| 9.2.1 | Dados Sintéticos | 53 |
| 9.3 | Interface SDC-X | 53 |
| 10 | Resultados..... | 56 |
| 10.1 | Servidores | 56 |
| 10.1.1 | Servidor FHIR | 56 |
| 10.1.2 | Servidor <i>ElasticSearch</i> | 57 |
| 10.2 | Interface SDC-X | 58 |
| 10.2.1 | Roteamento e Endereçamento | 60 |
| 10.2.2 | Controle de Acesso..... | 62 |
| 10.2.3 | Perfis da Interface SDC-X | 65 |
| <i>10.2.3.1</i> | <i> Perfil Pesquisador</i> | <i>65</i> |
| <i>10.2.3.2</i> | <i> Perfil Profissional.....</i> | <i>70</i> |
| <i>10.2.3.3</i> | <i> Perfil Gestor</i> | <i>72</i> |
| <i>10.2.3.4</i> | <i> Perfil Paciente</i> | <i>75</i> |
| <i>10.2.3.5</i> | <i> Perfil Administrador.....</i> | <i>76</i> |
| 10.2.4 | Termos de Consentimento e Política de Privacidade | 78 |
| 10.3 | Estudo de Caso | 78 |
| 10.4 | Avaliação de Tecnologia em Saúde 4.0..... | 80 |
| 11 | Discussão..... | 81 |
| 11.1 | Padrão HL7 FHIR..... | 81 |
| 11.2 | Servidores HAPI FHIR e <i>ElasticSearch</i> | 82 |
| 11.3 | Arquitetura MVC..... | 82 |
| 11.4 | Plataforma Web | 83 |
| 11.5 | Potencial de Integração do Projeto SDC-X | 84 |

| | | |
|-----------|---|------------|
| 11.6 | Validação com Dados SIMULADOS E REAIS..... | 86 |
| 11.7 | Projeto SDC-X como Tecnologia em Saúde 4.0 | 86 |
| 12 | CONCLUSÃO..... | 87 |
| 12.1 | Trabalhos Futuros | 88 |
| | REFERÊNCIAS..... | 90 |
| | APÊNDICE A – Requisições FHIR do projeto | 101 |
| | APÊNDICE B – Requisições <i>ElasticSearch</i> do projeto | 102 |
| | APÊNDICE C – Diagrama de Classes da Plataforma Web | 103 |
| | APÊNDICE D – Diagrama UML para o perfil Pesquisador | 104 |
| | APÊNDICE E – Diagrama UML para o perfil Profissional..... | 105 |
| | APÊNDICE F – Diagrama UML para o perfil Gestor..... | 106 |
| | APÊNDICE G – Diagrama UML para o perfil Administrador | 107 |
| | APÊNDICE H – Diagrama UML para o perfil Paciente | 108 |

1 INTRODUÇÃO

O sistema de saúde vem evoluindo com o avanço tecnológico, melhorando equipamentos e serviços, buscando maior satisfação dos pacientes. Essas evoluções são marcadas por eventos que acontecem na indústria, que atualmente está passando por uma transição, a chamada Indústria 4.0. Esta transição implica em uma mudança no setor de desenvolvimento da indústria, aplicando conceitos de monitoramento remoto e maior automação, conectando diversos setores, máquinas, sensores, e diversos dispositivos para coletar dados e criar um sistema mais inteligente (DRATH; HORCH, 2014).

Estes novos conceitos também são aplicados na área da saúde, resultando na chamada Saúde 4.0. A possibilidade de digitalização e implementação de plataformas de saúde no formato virtual são os principais fatores que impulsionam este novo conceito. A aplicação de Saúde 4.0 possibilita criar uma rede de integração entre todas as áreas da saúde, coletando dados de paciente, hospitais, setores de pesquisa, farmácias, e diversos outros, e armazenando essas informações na nuvem. Assim é possível criar serviços mais inteligentes, descentralizados, e de maior qualidade (LHOTSKA, 2020).

Essa integração de informações e dados, para poder funcionar de forma correta, necessita que seja interoperável. A interoperabilidade possibilita a comunicação entre dispositivos e sistemas que troquem informações, de forma que possam interpreta-la (IEEE, 1991). A interoperabilidade traz diversos benefícios na tecnologia aplicada na saúde. A implementação de padrões reduz riscos, custos e tempo de desenvolvimento de tecnologia em saúde (BENDER; SARTIPI, 2013; WALKER et al., 2005).

O crescimento da quantidade de dados e o desafio da interoperabilidade na área da saúde proporcionaram o desenvolvimento de padrões buscando modelar e representar os dados criando ambientes de interoperabilidade entre os diferentes sistemas informatizados em saúde. O padrão mais amplamente adotado pela comunidade da saúde é o padrão FHIR (*Fast Healthcare Interoperability Resources*) desenvolvido pela *Health Level Seven* (HL7). Este padrão utiliza ferramentas e tecnologias atuais presentes no mercado, e permite uma estruturação abrangente como um padrão para troca, integração, compartilhamento e recuperação de informações eletrônicas de saúde (HONG et al., 2017) (HL7, 2022a).

A utilização da padronização da informação através de sistemas interoperáveis pode beneficiar o monitoramento de doenças crônicas, tanto para um controle populacional como individual. O *diabetes mellitus* (DM) é uma doença crônica. Há uma estimativa que até o ano

de 2045, 700 milhões de pessoas vão ser diagnosticadas com DM no mundo (IDF, 2021). Esta doença atinge diretamente as atividades de vida diárias da pessoa, podendo acarretar em, cegueira, perda de membros, neuropatias e até mesmo o óbito, quando tratada de forma incorreta ou sem o devido monitoramento. Ela apresenta ser onerosa tanto para o paciente quanto para os órgãos públicos (SBD, 2022).

Serviços de saúde bem estruturados providenciam intervenções importantes e acompanhamento necessário para ajudar pessoas com DM, melhorando sua qualidade de vida. São observadas reduções nas taxas de complicações relacionadas ao DM em países que possuem dados adequados para examinar as tendências ao longo do tempo (WHO, 2016).

Com isso, este trabalho propõe o desenvolvimento de uma plataforma *web* para múltiplos usuários com o intuito de visualizar e analisar dados de pacientes com DM. É proposto também a utilização de um servidor de dados de pesquisa clínica utilizando o HL7 FHIR, um servidor *ElasticSearch* para complementar e melhorar a performance de troca de informações entre interface e servidores. Assim, o trabalho desenvolvido apresenta uma ferramenta interoperável, com dados de pesquisa com alta veracidade, através de uma plataforma intuitiva.

1.1 MOTIVAÇÃO E JUSTIFICATIVA

Existem diversos estudos relacionados sobre a DM no âmbito acadêmico, mas muitas vezes sem padrões de metodologias de estudos. Isto dificulta o compartilhamento e reutilização dos dados obtidos por estes estudos. Esta problemática é considerada pelo Instituto de Engenharia Biomédica da Universidade Federal de Santa Catarina (IEB-UFSC) que desenvolve pesquisas sobre diagnóstico e acompanhamento do DM e suas complicações.

O DM é caracterizado como uma doença metabólica que resulta na deficiência da secreção de insulina, resultando em altos níveis de glicose no sangue, causando hiperglicemia. A falta de monitoramento da doença pode resultar em diversas outras complicações, como o desenvolvimento de doenças cardiovasculares, doença arterial e cerebrovascular periférica, obesidade, tuberculose, entre outros (WHO, 2019). O número de pessoas com DM é crescente, e estudos apontam que de 463 milhões de pessoas com DM no mundo em 2019, este valor possa aumentar para 700 milhões em 2045 (IDF, 2021). No Brasil, os números da doença tendem a seguir essa mesma linha de crescimento, com consequências alarmantes, que resultaram na maior causa de morte no ano de 2017. O impacto financeiro da DM também é alarmante, visto

que o Brasil é o terceiro país do mundo com mais gastos em saúde relacionados à doença (IDF, 2021).

Tendo em vista o que foi abordado anteriormente, estratégias de *screening* para detecção precoce da doença e acompanhamento tornam-se fundamentais. Uma das características da DM, é que ela pode ser subclínica. Em alguns desses pacientes assintomáticos, podem ocorrer complicações microvasculares e macrovasculares, antes que esses recebam o diagnóstico. Nestes casos, o monitoramento e acompanhamento da DM são cruciais para o controle e para evitar maiores complicações da doença (IDF, 2021).

Buscando atender os desafios de uma plataforma unificada, interoperável, e com dados de estudos relevantes envolvendo a DM, o IEB-UFSC está desenvolvendo um sistema de saúde digital ubíquo, visando atender os requisitos de Saúde 4.0. O sistema é denominado de *Screening for Diabetes Complications* (SDC-X) e tem como objetivo monitorar, acompanhar e detectar complicações causadas pelo DM. O sistema consiste em um desenvolvimento multidisciplinar, separado por módulos, que adquirem os dados fisiológicos, processam, analisam, armazenam, distribuem e descentralizam sua visualização.

Considerando o desenvolvimento do sistema SDC-X pelo IEB-UFSC, que se apresenta como uma ferramenta com grande potencial de auxílio no acompanhamento e diagnóstico de indivíduos com DM, este trabalho tem como objetivo o desenvolvimento de uma plataforma *web* baseada em servidores de dados interoperáveis utilizando o padrão HL7 FHIR. Este trabalho irá implantar dois servidores para dados, que compõem os módulos de armazenamento do projeto SDC-X, e também irá desenvolver uma plataforma *web* como um módulo de apresentação visual dos dados e informações coletadas.

1.2 OBJETIVOS

Nas seções abaixo estão descritos o objetivo geral e os objetivos específicos deste trabalho.

1.2.1 Objetivo Geral

Desenvolver uma aplicação *web* para visualização de dados em saúde com base na comunicação através do padrão HL7 FHIR e implantar dois servidores, HAPI FHIR e

ElasticSearch, para realizar a comunicação entre dados em saúde. O trabalho visa complementar projeto SDC-X desenvolvido no IEB-UFSC.

1.2.2 Objetivos Específicos

Para cumprir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Projetar e desenvolver uma aplicação *web* utilizando o padrão de *design* MVC;
- Implantar o servidor HAPI FHIR já validado pelo projeto SDC-X;
- Implantar e validar a utilização de um servidor *ElasticSearch* para complementar e auxiliar nas requisições realizadas pela aplicação *web*.
- Avaliar e validar a aplicação *web* e os servidores utilizando dados simulados e dados reais coletados pelo projeto SDC-X.

2 REFERENCIAL TEÓRICO

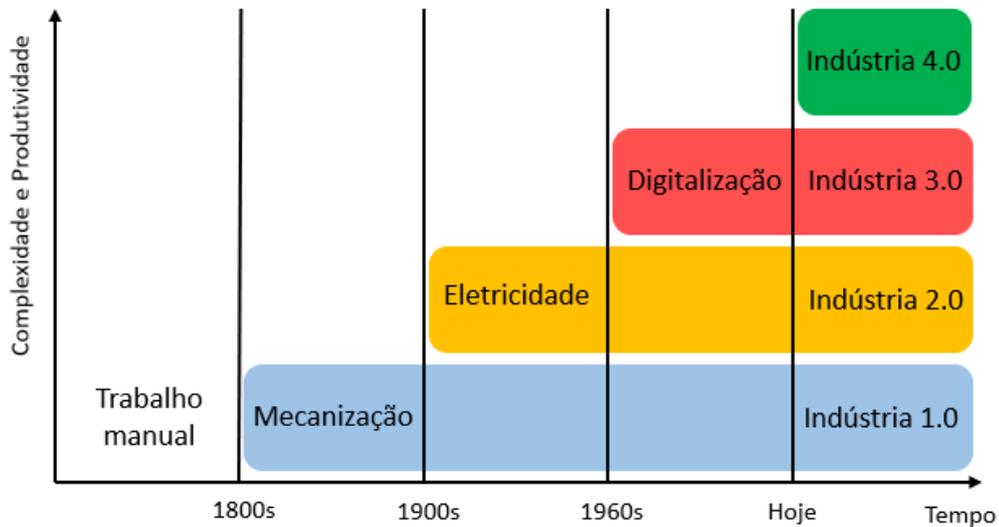
Neste capítulo é apresentada a fundamentação teórica que embasa a execução deste trabalho. Inicialmente é apresentado o conceito de Indústria 4.0 e Saúde 4.0, a importância da interoperabilidade no desenvolvimento tecnológico e seu contexto no desenvolvimento de padrões utilizados na área da saúde. Em seguida são apresentados os conceitos da DM e a importância do desenvolvimento de tecnologias para o estudo e análise da doença.

2.1 INDÚSTRIA 4.0 EM SAÚDE

O desenvolvimento tecnológico e científico tem melhorado o sistema de saúde como um todo. Este processo de desenvolvimento é possível devido a digitalização da informação, que resulta em sistemas de suportes de decisões, processamento de sinais e imagens, sistemas de gerenciamento, e o mais importante, o compartilhamento de dados. Para acompanhar o avanço tecnológico, a indústria da saúde é motivada a vencer desafios como escassez de profissionais, demanda de maior qualidade nos serviços de saúde, colaboração e suporte entre a própria indústria de saúde e o desenvolvimento de equipamentos de alta tecnologia (AL-JAROODI *et al.*, 2019; LHOTSKA, 2020).

Esse avanço tecnológico é guiado pela indústria, que segue os paradigmas tecnológicos que sofreram mudanças no decorrer dos anos, chamados de “revoluções industriais”. A Primeira Revolução Industrial ocorreu devido a mecanização de equipamentos nos anos de 1780, fazendo com que a produção deixasse o serviço manual e se concentrasse em grandes fábricas, aumentando a produtividade. A Segunda Revolução Industrial começou por volta dos anos 1900 com o uso intenso da energia elétrica e a produção em massa nas linhas de produção, atingindo seu clímax com a produção do modelo Fordismo, nos Estados Unidos. O desenvolvimento contínuo das linhas de produção, evolução da eletrônica e digitalização, levaram ao surgimento da Terceira Revolução Industrial, que tornou possível a programação de controladores que viriam a possibilitar a programação digital de sistemas autônomos (DRATH; HORCH, 2014; LASI *et al.*, 2014; CHUTE; FRENCH, 2019). Essas mudanças de paradigma, e atualmente discutida Indústria 4.0, podem ser vistas na Figura 1, onde observa-se que a complexidade e produção aumentaram em cada revolução.

Figura 1 - Evolução das Revoluções Industriais.



Fonte: Adaptado de DRATH e HORCH (2014).

Atualmente existem muitos questionamentos é vivenciada uma nova revolução industrial, a chamada Quarta Revolução Industrial, ou Indústria 4.0. As tecnologias para aquisição e troca de informações e automação vem crescendo na indústria com utilização de tecnologias como IoT (*Internet of Things*), sistemas ciber físicos e análises de dados. Essas são tecnologias que permeiam o que se chama de Indústria 4.0 (DRATH; HORCH, 2014; LASI *et al.*, 2014; LHOTSKA, 2020; CHUTE; FRENCH, 2019).

O termo Indústria 4.0 surgiu na feira de Hanover, em 2011, na Alemanha e refere-se a aplicação dos conceitos genéricos de Sistemas Ciber Físicos (SCF) para o sistema de produção industrial.

Este novo conceito introduz uma visão de produção inteligente, que não envolve apenas o ambiente da fábrica, mas todos os fatores externos e internos, seja na coleta da matéria prima até a forma de utilização do produto pelo consumidor. A regra mais importante deste novo conceito é a informação gerada pela aquisição de todos esses dados. Assim, tem-se o desenvolvimento de diversos dispositivos conectados com sensores, tecnologias cibernéticas, *big data*, robôs autônomos, computação e armazenamento em nuvem, realidade aumentada, IoT, e inteligência artificial, que suportam uma grande quantidade de informação. Todos esses processos trazem questionamentos importantes como privacidade e segurança de dados, segurança de infraestrutura, sistemas de energia e proteção de propriedade intelectual. Assim exige-se a criação de padrões e ajustes legais que fazem parte da construção dessa nova estrutura (LHOTSKA, 2020; LASI *et al.*, 2014).

Outro fator importante, é que a infraestrutura e ferramentas para coleta e armazenamento de dados deve seguir uma série de regras para que mudanças ou extensões criadas para o processo de produção não resultem em sistemas completamente novos sem a possibilidade de integração (LHOTSKA, 2020).

2.2 ÁREA DA SAÚDE E SAÚDE 4.0

A área da saúde vem mostrando constante evolução no desenvolvimento tecnológico e transformação digital. Novos conceitos em saúde e modelos baseados em inovação e utilização de tecnologias de comunicação e informação já estão sendo estudados e implementados. Estes conceitos seguem os princípios das aplicações de Indústria 4.0, adaptando-os para a indústria de saúde chamada de Indústria em Saúde 4.0, ou Saúde 4.0. O sistema de Saúde 4.0 pode ser definido como uma implementação da integração de plataformas de saúde de forma virtual, distribuída e em tempo real para pacientes e profissionais. Este conceito permite a integração, compartilhamento e otimização do uso dos serviços em saúde para melhorar seu gerenciamento e reduzir custos (AL-JAROODI *et al.*, 2019; CHUTE; FRENCH, 2019; KHELASSI *et al.*, 2019).

Este sistema de saúde faz uso de tecnologias como SCF, computação em nuvem, IoT, e sistemas de comunicação 5G. A implementação do paradigma de SCF ajudará a *softwares* se adequarem a sistemas distribuídos e *Big Data* e inteligência artificial irão virtualizar as informações, permitindo a geração de diagnósticos de forma rápida e precisa (KHELASSI *et al.*, 2019).

A utilização de tecnologia da informação na saúde começou com a introdução de *websites* sobre saúde pública e gerenciamento de doenças, e em sequência, os Registros Eletrônicos em Saúde (RES), computação em nuvem e interações de aplicativos *mobiles*. O crescimento da Saúde 4.0 traz todas as tecnologias acompanhadas de coleta de dados, crescimento da inteligência artificial, e soluções sofisticadas para facilitar interfaces virtuais interativas. O objetivo da Saúde 4.0 é conectar pacientes, aparelhos médicos, hospitais, clínicas, fornecedores médicos e farmacêuticos, organizações de pesquisa, e outros componentes relacionados a saúde (AL-JAROODI *et al.*, 2019). A Figura 2 ilustra essa rede que abrange o conceito de Saúde 4.0 e que busca melhorar a interação entre as diversas frentes de atuação na área da saúde.

A aplicação de Saúde 4.0 é baseada em seis princípios da Indústria 4.0 (CHUTE; FRENCH, 2019; KHELASSI *et al.*, 2019; THUEMMLER; BAI, 2017):

- Interoperabilidade: capacidade de trocar informações entre sistemas de saúde através de uma estrutura padronizada de comunicação;
- Virtualização: a aplicação da interoperabilidade para conexão entre diversos aparelhos de saúde permite criar a digitalização de informações, sistemas e processos;
- Descentralização: sistemas de saúde, com a aplicação de IA, irão tomar decisões e se auto controlar;
- Capacidade em Tempo Real: capacidade de adquirir dados, apresentá-los e tomar decisões em uma janela de tempo muito pequena;
- Serviços Orientados: criação de *softwares* de serviço para interagir com aparelhos e sistemas médicos;
- Modularidade: sistemas com habilidades de melhorar módulos individuais para atender novos requisitos e reutilizar módulos para criar novos sistemas de saúde.

Figura 2. Rede de Saúde inteligente interligada pela “*Fog computing*” de cada sistema de saúde. O termo “*Fog computing*” compõe a arquitetura de computação entre a fonte de dados e a nuvem.



Fonte: Adaptado de Al-Jaroodi, Mohamed e Abukhousa (2019).

Um dos desafios do uso de nuvem para o armazenamento de dados, é garantir a privacidade dos pacientes, sem que seus dados sejam violados. Por este motivo os dados não podem ser armazenados em servidores públicos, ou, em alguns casos, os dados devem estar imediatamente acessíveis, com baixa latência ou erros com a nuvem (PACE *et al.*, 2019).

2.3 ÁREAS E APLICAÇÃO

Diversas pesquisas e projetos são desenvolvidos para validar alguma aplicação ou teoria para o uso dos conceitos de Saúde 4.0, como monitoramento e armazenamento de dados, IoT, robótica, sensores, aplicações em nuvem e outros dispositivos.

O monitoramento constante de sinais fisiológicos apresenta resultados como identificação preventiva e preditiva de alguma doença, e ajuda nas tomadas de decisões. Um bom exemplo é o *Diabetes Mellitus* (DM), que em alguns casos a doença pode não apresentar sintomas, mas desenvolver complicações microvasculares e macrovasculares antes do diagnóstico. Em indivíduos já diagnosticados com DM o monitoramento é ainda mais importante, pois auxilia em tratamentos mais assertivos, retarda o aparecimento de complicações e prolonga a vida desse indivíduo (IDF, 2021).

Percebe-se que as aplicações de Saúde 4.0 se baseiam em diferentes tipos de tecnologias e serviços, com uma infraestrutura distribuída e heterogênea, e necessitam de interoperabilidade para se encaixarem nos conceitos de Saúde 4.0. Remodelar uma aplicação pronta para se adequar a esses conceitos é complexo e necessita de tempo de desenvolvimento, tornando esse processo oneroso, enquanto que aplicar esses conceitos desde o início do desenvolvimento reduziria os custos. Isto ressalta a importância de padrões em todas as estruturas de uma aplicação, pois diminui o tempo de desenvolvimento, possibilita alterações e atualizações futuras da aplicação, e facilita a interoperabilidade.

3 INTEROPERABILIDADE

Interoperabilidade é “a habilidade de dois ou mais sistemas ou componentes de trocarem informações e usá-las” (IEEE, 1991). Este conceito permite que diferentes tipos de sistemas, que podem ser equipamentos e *softwares* de diferentes empresas ou organizações, possam trocar dados através de uma arquitetura interoperável. Isto permite, na área da saúde, maior comunicação entre os diferentes estabelecimentos de saúde, como hospitais ou unidades

básicas de saúde, minimizando o tempo de atendimento ou tomada de decisão ao paciente, redução de exames repetidos, além de permitir que os profissionais da saúde foquem em providenciar melhor serviço ao paciente.

Um sistema de arquitetura interoperável deve realizar a aplicação de padrões consistentes que definem a sintaxe e semântica da informação. A implementação de padrões reduz os riscos, custos e tempo de desenvolvimento de tecnologia em saúde (BENDER; SARTIPI, 2013).

A interoperabilidade aplicada na saúde reduz o custo das aplicações. Estima-se que só nos EUA, em 2003, os sistemas informatizados em saúde poderiam economizar até USD\$77,8 bilhões se tivessem aplicado a interoperabilidade entre os sistemas (WALKER *et al.*, 2005). A aplicação deste conceito tem grande importância na área da saúde não somente em reduzir custos, mas também pelo compartilhamento e fácil acesso a dados que podem beneficiar o combate às doenças.

O crescimento da quantidade e dos tipos de dados na área da saúde impulsionou o desenvolvimento de padrões para modelar e representar essas informações para criar a interoperabilidade entre diferentes sistemas informatizados em saúde. Alguns padrões foram desenvolvidos por organizações em saúde, tais como “*Health Level Seven*” (HL7), o *Observational Health Data Sciences and Informatics* (OHDSI), o *Patient-Centered Outcomes Research Institute* (PCORI) (HONG *et al.*, 2017).

A organização HL7 desenvolveu o padrão *Fast Healthcare Interoperability Resources* (FHIR) que combina as tecnologias modernas em evolução e as necessidades do mercado, e vem sendo aceito como a próxima geração de padrão para compartilhamento de informações de saúde entre sistemas e organizações (HONG *et al.*, 2017; JANKI *et al.*, 2018).

4 HEALTH LEVEL SEVEN (HL7)

Segundo o *site* da organização:

Fundado em 1987, o Health Level Seven International (HL7) é uma organização sem fins lucrativos de desenvolvimento de padrões credenciados pela ANSI (*American National Standards Institute*) dedicado a fornecer uma estrutura abrangente e padrões relacionados para troca, integração, compartilhamento e recuperação de informações eletrônicas de saúde que apoiam a prática clínica e a gestão, prestação e avaliação de serviços de saúde (DOCUMENTAÇÃO HL7).

O HL7 é formado por uma comunidade internacional de especialistas que colaboram para desenvolver padrões para troca de informações de saúde e interoperabilidade entre

sistemas em saúde. O termo “*Level Seven*” refere-se ao sétimo nível do modelo de comunicações de sete camadas da *International Organization for Standardization (ISO)* para *Open Systems Interconnection (OSI)* (HL7, 2022a).

O HL7 produz padrões para sistemas eletrônicos de mensagens e outros padrões como estruturas de documentos eletrônicos e padrões de conteúdo para sustentar a interoperabilidade de sistemas. Os padrões de mensagem estão disponíveis nas versões HL7 v2, HL7 v3 e o novo HL7 (BENDER; SARTIPI, 2013).

4.1 VERSÃO HL7 V2

Um dos padrões mais implementados no mundo (mais de 35 países tem implementações utilizando este padrão), o HL7 v2, ou *Version 2 Messaging Standard*, foi criado em 1987 como um protocolo de aplicação para troca de dados eletrônicos em um sistema de saúde. Este padrão foi projetado para atender um sistema central de atendimento ao paciente, bem como um ambiente mais distribuído onde os dados residem em sistemas departamentais (HL7, 2022b).

Apesar da sua vasta utilização, este padrão depende de uma personalização local em seu desenvolvimento, chamados de “segmentos Z”. As interfaces dos sistemas projetos são definidas com cerca de 80% pelas especificações HL7 e 20% personalizadas pela implementação local. Este padrão também carece de uma ontologia formal para unificar os conceitos que estão sendo trocados em várias mensagens e interfaces (BENDER; SARTIPI, 2013).

4.2 VERSÃO HL7 V3

A versão HL7 v3 começou em 1995 e não foi desenvolvida para incrementar ou melhorar a versão HL7 v2, e sim como um novo modelo, desenvolvido para resolver as deficiências encontradas na versão HL7 v2. O novo modelo introduziu o processo de desenvolvimento *HL7 Development Framework (HDF)* e um modelo de informação central chamado de *Reference Information Model (RIM)* (BENDER; SARTIPI, 2013), e fornece uma única fonte que permite especificações da implementação do HL7 v3 para trabalhar com todas as mensagens, tipos de dados e terminologias necessárias para construir uma implementação completa (HL7, 2022c).

O modelo RIM define as estruturas para todos os elementos semânticos e lexicais do HL7 v3. O processo faz uso de técnicas *top-down* de *Model Driven Architecture* para desenvolver modelos de informação clínica e tenta gerar automaticamente os artefatos de mensagens (BENDER; SARTIPI, 2013). Essas informações seguem uma nova abordagem com base em uma metodologia orientada por modelo que produz mensagens e documentos eletrônicos expressos em sintaxe XML (*Extensible Markup Language*) (HL7, 2022c). O padrão HL7 v3 não é diretamente implementável e requer ferramentas para a geração de sistemas de *software* executáveis (BENDER; SARTIPI, 2013; HL7, 2022c).

O padrão HL7 v3 não foi construído para ser diretamente implementado, mas que devem ser usados modelos iniciais previamente criados e que serão restritos para uso por determinada aplicação. Isto aumenta a margem de não compatibilidade entre implementações, dificultando que fornecedores de *software* e provedores de soluções internacionais ofereçam soluções consistentes em todas as regiões e idiomas (BENDER; SARTIPI, 2013).

Apesar da abordagem de modelo *top-down* apresentar um fluxo contextual amigável em suas descrições de alto nível, na verdade se tornou inviável quando o estágio de aplicação é extremamente complexo conforme o modelo se transforma de artefatos de implementação abstratos em concretos. Além da complexidade de implementação, as versões v2 e v3 não são compatíveis, e para que ocorra troca de informações entre as versões necessita de um *software* de tradução sofisticado, que não é fornecido pelo HL7 (BENDER; SARTIPI, 2013; HL7, 2022c).

4.3 HL7 FAST HEALTHCARE INTEROPERABILITY (HL7 FHIR)

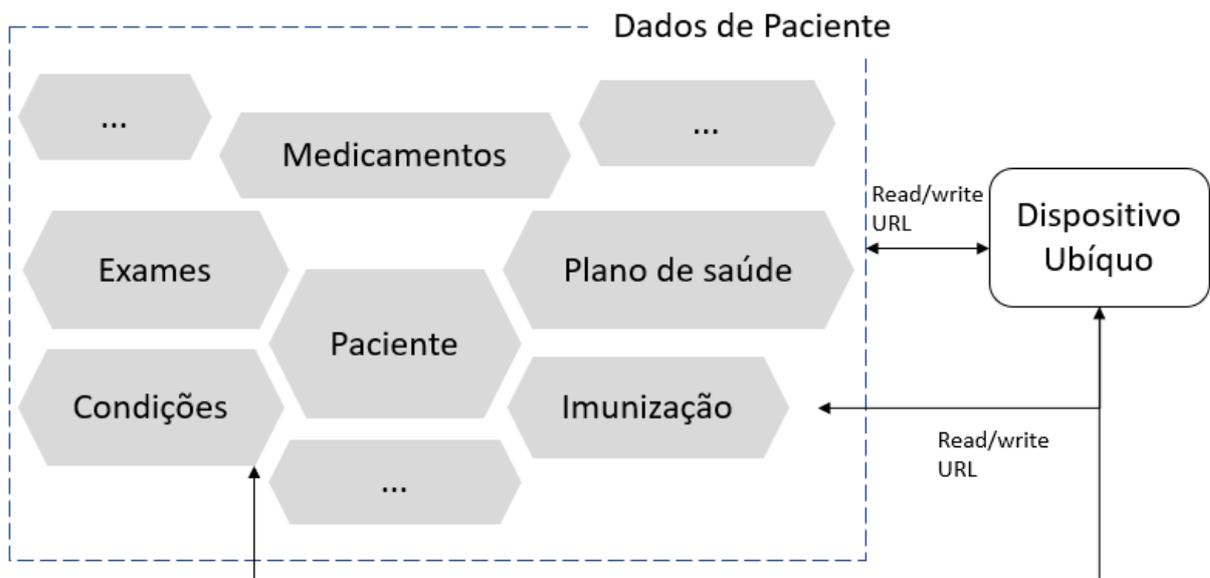
O HL7 FHIR (lê-se “*fire*”), inicialmente chamado de *Resources for Health*, foi criado em 2011 como uma força tarefa para melhorar o padrão HL7 devido à pressão da falta de implementações em HL7 v3. Este novo modelo é baseado nos princípios da tecnologia RESTful (*Representational State Transfer*) (BENDER; SARTIPI, 2013).

Apesar de não ser compatível com as versões v2 e v3, ele oferece melhorias. O HL7 FHIR apresenta uma “mistura” do que funcionou nas versões v2 e v3, resultando em especificações mais concisas e de fácil compreensão. Existe uma diversidade de bibliotecas que as implementam, e a construção é baseada em padrões *web* bem estabelecidos. Assim, o FHIR pode ser utilizado em diversas aplicações, como aplicativos móveis, comunicação em nuvem,

compartilhamento de dados, aplicações *web*, entre outros (BENSON; GRIEVE, 2016a; YAMAGUTI *et al.*, 2018).

O padrão FHIR é fundamentalmente baseado no conceito de “Recurso”, que é um módulo de dados fundamental que pode ser acessado usando um URL via REST. Por exemplo, um dado de paciente é formado agregando recursos FHIR como paciente, condição, procedimento, medicamentos, observação, imunizações, e plano de saúde. De forma geral, um recurso FHIR pode ser visto como uma peça de “Lego”, e você pode juntar essas peças para criar um *output* final. Dessa forma um registro médico de paciente pode ser acessado como um conjunto de recursos FHIR através de uma URL, como mostra a Figura 3. Quando os dados são acessados via URL, os recursos são comunicados nos formatos XML ou JSON (*JavaScript Object Notation*) (SARIPALLE, 2019).

Figura 3. Formato de acesso as informações usando FHIR.



Fonte: Adaptado de SARIPALLE (2019).

4.3.1 Representational State Transfer (REST)

O modelo de sistema REST foi descrito por Fielding no ano 2000, mais tarde este modelo foi chamado de arquitetura RESTful, e se tornou amplamente adotado como a abstração de informação dominante da *WorldWideWeb*. A maioria das plataformas da Internet migraram para os serviços RESTful e saíram de outras opções de arquitetura (BENDER; SARTIPI, 2013).

As vantagens práticas da arquitetura RESTful inclui interfaces mais leves que permitem rápida transmissão e processamento de dados, mais adequado para dispositivos *mobile* e *tablet*. Interfaces RESTful também permitem um desenvolvimento mais rápido por ter estruturas mais simples (BENDER; SARTIPI, 2013).

A manipulação de recursos segue uma API (*Application Programming Interface*) REST com as seguintes interações [DOCUMENTAÇÃO FHIR Developers]:

- Criar = POST `https://example.com/path/{resourceType}`
- Ler = GET `https://example.com/path/{resourceType}/{id}`
- Atualizar = PUT `https://example.com/path/{resourceType}/{id}`
- Deletar = DELETE `https://example.com/path/{resourceType}/{id}`
- Procurar = GET `https://example.com/path/{resourceType}?search parameters...`
- Histórico = GET `https://example.com/path/{resourceType}/{id}/_history`
- Transação = POST `https://example.com/path/ (POST a transaction bundle to the system)`
- Operação = GET `https://example.com/path/{resourceType}/{id}/${opname}`

A especificação FHIR descreve outros tipos de trocas além desta API RESTful, incluindo a troca de grupos de recursos como Documentos, Mensagens, e usando vários tipos de Serviços (HL7 FHIR, 2022).

4.3.2 Recursos (“*Resources*”) FHIR

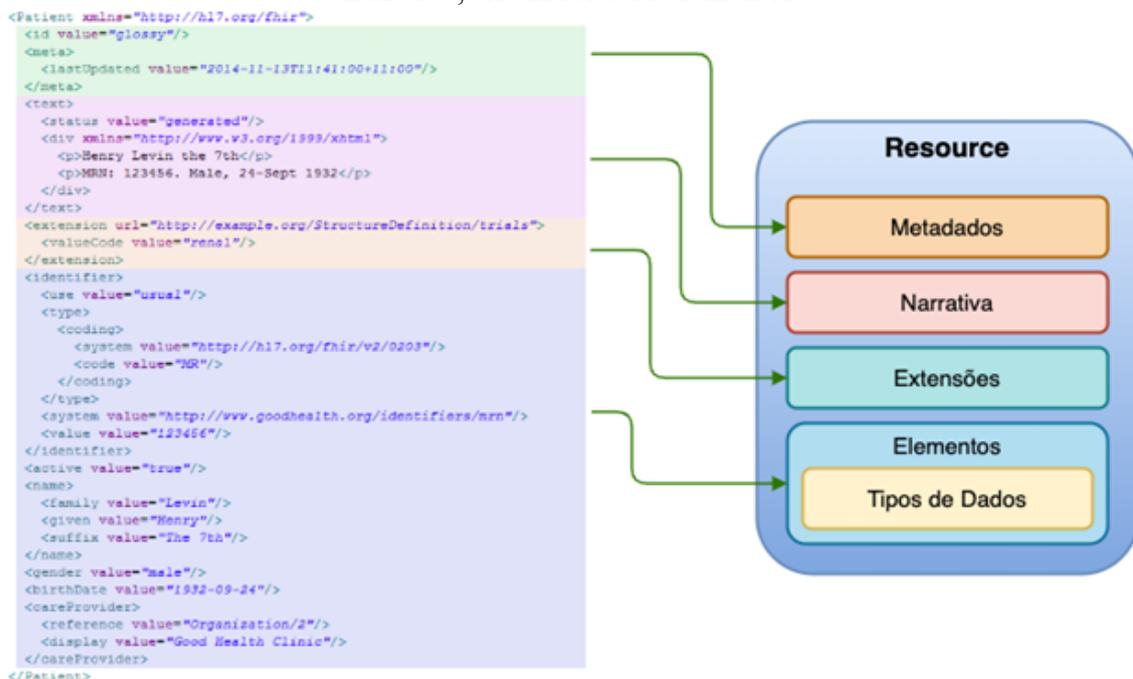
O objetivo do FHIR é definir as principais entidades envolvidas na troca de informações em saúde como Recursos. Cada recurso é uma identidade distinta (BENDER; SARTIPI, 2013). Os recursos são uma representação no nível da instância de algum tipo de entidade de saúde. Todos os recursos têm as seguintes características em comum (HL7 FHIR, 2022):

- Uma URL que identifica o recurso;
- Metadados comuns;
- Um sumário XHTML legível;
- Um conjunto de elementos de dados definidos;

- Uma extensão para dar suporte ao *framework* às variações na área da saúde.

As instâncias de recursos são representadas em XML ou JSON e atualmente existem 145 diferentes tipos de recursos definidos na especificação FHIR. Esses formatos são os mais utilizados em aplicações *web*, uma vez que são facilmente elaborados, lidos e utilizados em APIs. A estruturação dos dados nesses formatos apresenta um padrão com elementos obrigatórios e opcionais, como apresentado na Figura 4.

Figura 4. Estruturação do formato de um Recurso em XML composto pelos metadados, narrativa, extensões e os elementos.



Fonte: (VIEIRA; 2021)

Os metadados apresentam a identidade lógica de um Recurso com base no sistema onde está armazenado, informações de versões, atualizações e se o recurso é baseado em algum perfil. A narrativa é um código em XHTML (*Extensible Hypertext Markup Language*) que se torna legível ao ser interpretado por algum sistema. Nele consta um resumo do conteúdo do Recurso. As extensões descrevem as estruturas criadas em casos particulares e não presentes na documentação base. Por fim, os elementos apresentam os dados e informações de forma padronizada, as quais são previamente definidas nas especificações e são distintas para cada tipo de Recurso (HL7 FHIR, 2022; VIEIRA; 2022).

É extremamente complexo modelar a totalidade dos dados de saúde em um único modelo de informações, dessa forma o FHIR decompõe o domínio da saúde em fragmentos de informação menores e mais gerenciáveis. É importante, quando se quebra o modelo de informação em pedaços menores, ter um *framework* e um conjunto de diretrizes para garantir consistência e integridade dentro das estruturas de recursos e na forma como os recursos se referem uns aos outros (HL7 FHIR, 2022).

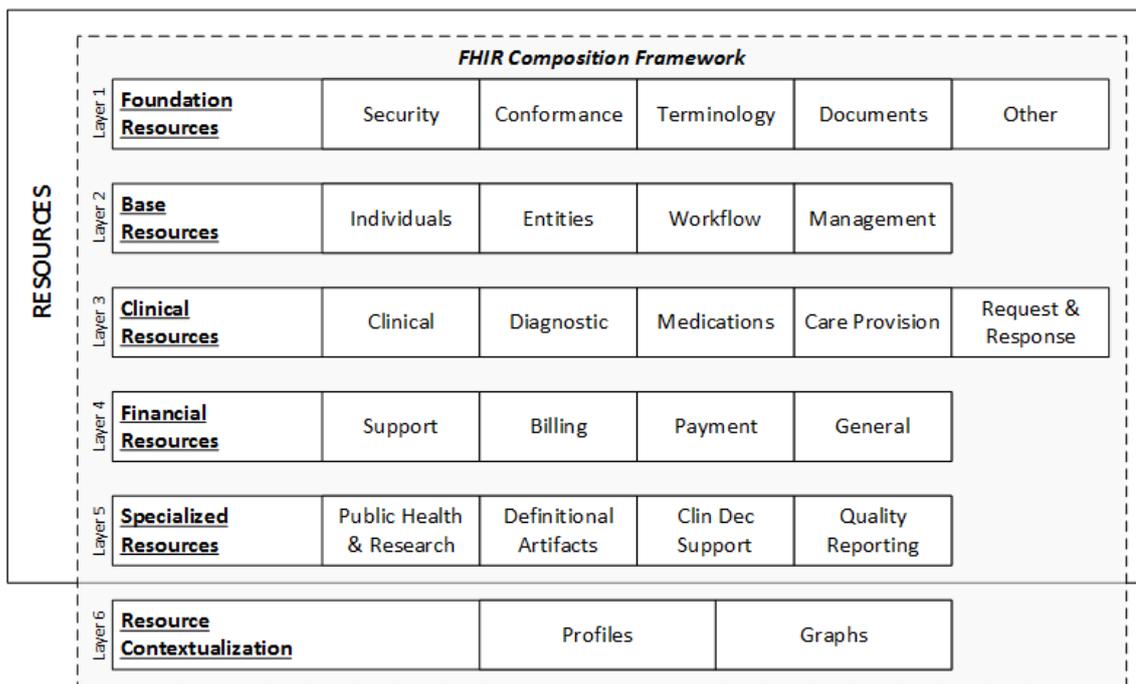
A documentação FHIR apresenta a configuração das subcategorias do modelo de informações de saúde organizadas em camadas. Essas camadas e categorias devem ser bem definidas, garantindo consistência e devem ser rigidamente controladas pois são mais comuns (mais genéricas) (HL7 FHIR, 2022).

Na Figura 5 pode-se ver a distribuição das camadas, e segundo descrito nas especificações de arquitetura do FHIR (HL7 FHIR, 2022):

- **Foundation Resource:** São os recursos básicos mais rudimentares. Eles são frequentemente usados para tarefas de infraestrutura.
- **Base Resources:** A camada dois consiste em recursos básicos. São normalmente os mais usados e exigem o mais alto grau de consistência e rigor na arquitetura. Governanças são os mais indicados para os recursos de camadas um e dois.
- **Clinical Resources:** A camada três inclui recursos de natureza clínica, mas também são usados em muitos casos de uso, isso inclui recursos para observações clínicas, tratamento clínico, prestação de cuidados e medicamentos. Esses recursos podem ser usados por si próprios, mas normalmente se baseiam nos recursos da camada dois. Por exemplo, um recurso de observação fará referência ao recurso do paciente da camada dois.
- **Financial Resources:** Nesta camada, os recursos são construídos sobre os recursos clínicos e básicos. Por exemplo, um recurso de cobrança fará referência à eventos e atividades clínicas, assim como recursos de base como um paciente.
- **Specialized Resources:** São recursos com menos casos de uso. Esses recursos quase sempre fazem referência a recursos em camadas inferiores. Como o FHIR prioriza a satisfação dos casos de uso mais comuns, existem menos recursos nesta camada.

- **Resource Contextualization:** Não existem recursos na camada seis. Entretanto, esta camada estende a estrutura de composição formada pelas cinco primeiras camadas de recursos. A camada seis inclui perfis e gráficos. Os perfis são usados para estender, restringir ou contextualizar recursos para um determinado fim. Os gráficos são composições de recursos, ou “teias” de recursos, que contém atributos próprios.

Figura 5. Camadas dos Recursos do FHIR.



Fonte: (HL7 FHIR, 2022)

Os recursos geralmente fazem referências a recursos de mesma camada ou superior, por exemplo, um recurso da camada 4 normalmente fará referência apenas aos recursos nas camadas 4, 3, 2 ou 1. Não existe uma proibição de que recursos não possam fazer referências a recursos de camadas inferiores, mas isso não é comum de acontecer. Esse fato possibilita o HL7 FHIR de identificar os recursos que provavelmente são os mais comuns de serem utilizados, exigindo maior grau de consistência e governança, e ajuda também a identificar as áreas de maior prioridade para a criação de novos recursos. Geralmente os recursos nas camadas mais altas (camadas 1, 2 e 3) tem maior prioridade na criação de novo recursos que os das camadas inferiores (camadas 4 e 5), porque os recursos das camadas superiores fornecerão maior valor devido ao maior número de usos de casos (HL7 FHIR, 2022).

A camada 6 do framework não exatamente um recurso. Perfis e gráficos são extensões dos recursos ou composições dos recursos que continuam a progressão da composição do *Framework* FHIR. Eles fornecem uma contextualização adicional quando requeridos para satisfazer certos casos de uso (HL7 FHIR, 2022).

Na documentação FHIR, existem recursos que foram validados e que estão em validação, ou seja, que não existe ainda um consenso sobre a sua formatação. Os recursos validados, ou normativos, atualmente, dos 145, apenas 11 foram validados como normativos.

O *framework* possibilita também a criação de novos recursos, ou mecanismos de extensão, caso uma aplicação não tenha o recurso que deseja utilizar, existe a liberdade de criar uma nova, ou alterar um recurso existente, para isso existe uma série de regras de como implementar.

4.3.3 Ferramentas de Aplicação

Existem diversos servidores públicos, *open sources*, que podem ser baixados e implementados na sua máquina, ou mesmo acessados *online*. Os desenvolvedores podem se conectar com esses servidores e utilizar suas APIs, e todos eles suportam acesso não autenticado. Os responsáveis por esses servidores geralmente reiniciam o sistema de tempos em tempos, pois em muitos deles é liberado que os desenvolvedores usuários possam criar ou modificar recursos para realizar testes de implementação (BENSON; GRIEVE, 2016a).

Um desses servidores muito utilizado é o HAPI (*HL7 Application Programming Interface*), uma implementação completa do HL7 FHIR em JAVA. O projeto é primeiramente suportado pela *University Health Network* (UHN), e é formado por um time de programadores com histórico e reconhecimento por produzirem *open source softwares* relacionados a projetos integrados ao HL7 (BENSON; GRIEVE, 2016a). O projeto tem vários benefícios, entre eles um servidor *open source*, excelente documentação, comunidade grande e colaborativa, e frequentemente é atualizado (arrumando *bugs*, adicionando recursos, e expandindo recursos para novos padrões e recursos FHIR) (HUSSAIN; LANGER; KOHLI, 2018). HAPI FHIR é geralmente a ferramenta mais utilizada para desenvolvedores, e tem sido usado em diversas aplicações, desde soluções *mobile* até implementações pelos maiores provedores de serviços em nuvem (BENSON; GRIEVE, 2016a).

Um exemplo de aplicação *open source* que foi construído utilizando HAPI FHIR é o “*FHIR Broker*” desenvolvido pelo programa *NIH Sync for Science* e o *RSNA Image Share*

Network, onde utiliza chamadas RESTful FHIR e os distribui no PACS existente usando o protocolo DICOM (HUSSAIN; LANGER; KOHLI, 2018).

5 REDE NACIONAL DE DADOS EM SAÚDE

No Brasil existem estratégias para implementar um sistema integrado dos dados em saúde, atendendo a Política Nacional de Informação e Informática em Saúde (PNIIS), e a Estratégia e-Saúde para o Brasil e no Plano de Ação, Monitoramento e Avaliação de Saúde Digital para o Brasil (BRASIL, 2020b).

Assim, no ano de 2020 foi estabelecida a criação da Rede Nacional de Dados em Saúde (RNDS) como uma plataforma nacional de interoperabilidade em saúde, instituída pela Portaria GM/MS nº 1.434, de 28 de maio de 2020. Ela, além de ser um projeto estruturante do Conecte SUS, é um programa do Governo Federal voltado para a transformação digital da saúde no Brasil e tem o objetivo de promover a troca de informações entre os pontos da Rede de Atenção à Saúde, permitindo a transição e continuidade do cuidado nos setores públicos e privados. A RNDS faz parte da implementação de um ambiente de interconectividade para ser criado entre o período de 2020 e 2028. Buscando atender os conceitos de interoperabilidade na área da saúde, ela define o HL7 FHIR como protocolo padrão (BRASIL, 2020a; MINISTÉRIO DA SAÚDE, 2022a).

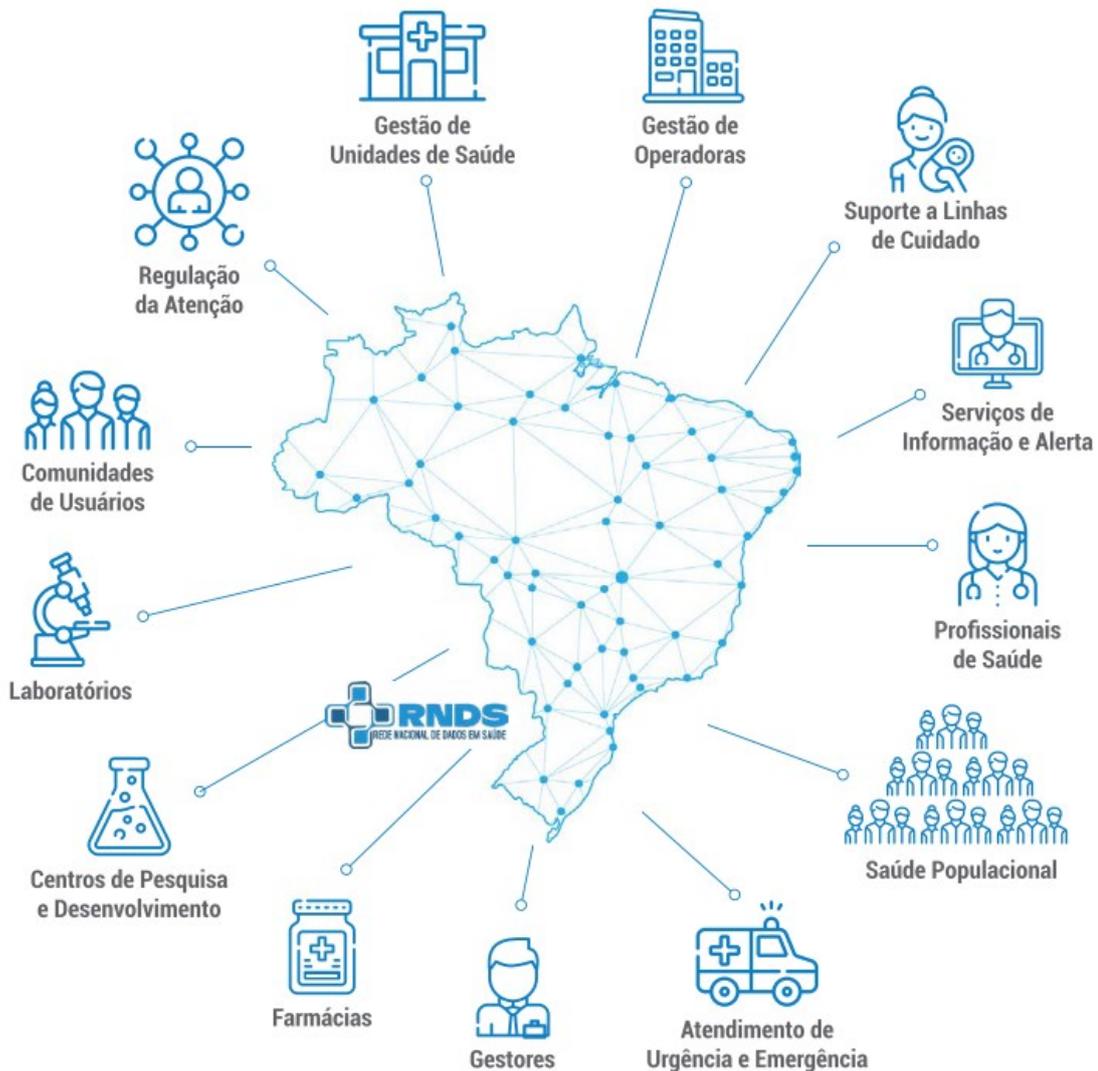
Desde a sua criação, diferentes portarias já foram criadas e implementadas para atender as necessidades atuais do Brasil, atendendo as demandas causadas pelo vírus SARS-CoV-2 (COVID-19), que em 11 de março de 2020, foi decretado como pandemia pela OMS. Essas portarias exigem a integração com a RNDS, descrevendo o compartilhamento de resultados de testes de COVID-19 (Portaria GM/MS nº 1.792 de 17 de julho de 2020), obrigatoriedade de registro de aplicação de vacinas contra a COVID-19 (Portaria GM/MS nº 69 de 14 de janeiro de 2021), e entre outras portarias publicadas pelo MS, que são atualizadas e acrescentadas conforme o projeto da RNDS evolui (MINISTÉRIO DA SAÚDE, 2022b).

Segundo os dados do Ministério da Saúde, a RNDS já coletou mais de 9 milhões de resultados de exames de COVID-19, mais de 391 milhões de registros de vacinação de COVID-19 e 26 milhões registros de autorização de internação hospitalar (MINISTÉRIO DA SAÚDE, 2022b).

O objetivo da RNDS é se tornar uma plataforma padronizada, moderna e interoperável de serviços, informações e conectividade, permitindo a transição e continuidade do cuidado nos

setores público e privado. Ela visa atingir um nível de conexão entre todos os setores de saúde, como apresentado na Figura 6 (BRASIL, 2020a).

Figura 6 – Objetivo de conexão da RNDS entre todas as entidades de saúde do Brasil, como uma plataforma de inovação, informação e serviços digitais em saúde.



Fonte: (BRASIL, 2020a).

É interessante de observar que existe uma preocupação de integração com os centros de pesquisas, o que possibilita o compartilhamento e aplicação direta das pesquisas desenvolvidas no país.

Uma plataforma integrada como a RNDS permite um melhor controle e administração dos dados em saúde, além de apresentar um panorama geral da saúde no país. Um dos resultados

dessa integração permite realizar um controle de doenças, pois auxilia as autoridades de saúde nas tomadas de decisão e conhecimento do problema, como a atual pandemia da COVID-19.

6 DIABETES MELLITUS

O monitoramento constante de doenças crônicas é de extrema importância para o seu controle, sendo elas responsáveis por mais de 70% das mortes no mundo (WORLD HEALTH ORGANIZATION, 2020). O Diabetes Mellitus (DM) é uma doença crônica e um dos problemas de saúde pública mais desafiadores do mundo devido à alta e crescente prevalência de casos e alto índice de morbidade, estando entre as 10 principais causas de morte no mundo (IDF, 2021).

O DM é uma doença caracterizada pela falta, ou baixos níveis de produção, do hormônio insulina, o que resulta em altos níveis de glicose no sangue, causando hiperglicemia (IDF, 2021). Pessoas com diabetes também correm maior risco de desenvolver outras doenças como: doenças cardiovasculares, doença arterial e cerebrovascular periférica, obesidade, catarata, tuberculose, entre outras (WHO, 2019). Estima-se que em 2019, o número de pessoas com diabetes era de 463 milhões e a projeção é que em 2045 esse número cresça para 700 milhões de pessoas. Além do número de pessoas com diabetes, outro fator alarmante é que mais de 50% das pessoas com diabetes não tem o seu quadro diagnosticado (IDF, 2021).

No Brasil o número de pessoas com DM está aumentando. Segundo os dados epidemiológicos da Sociedade Brasileira de Diabetes (2022), no período entre 2006 a 2018, o número de homens (≥ 18 anos) com diabetes cresceu 54%, e 28% entre as mulheres (≥ 18 anos). Em 2019, o número de pessoas entre 20-79 anos com DM era de 16,8 milhões de pessoas. Em 2017, DM foi a terceira maior causa de mortes (mortes por 100 mil habitantes). Segundo dados da Federação Internacional de Diabetes (2021), em 2019 o Brasil foi o terceiro país com maior gasto em saúde relacionados a diabetes (dados relacionados entre adultos de 20-79 anos).

Tendo todos esses números em vista, o impacto e custos globais relacionados a diabetes tende a crescer consideravelmente. Estudos e monitoramento da doença permitem realizar controles e aplicar medidas precisas para o tratamento. Para pessoas com DM, o gerenciamento de medicamentos essenciais, controle de glicemia e fatores de risco cardiometabólico, além de um monitoramento precoce das complicações via um sistema organizado de saúde, reduzem as complicações aguda e crônicas da doença e prolongam a vida (GREGG *et al.*, 2022; WHO, 2016).

6.1 TIPOS DE DIABETES

Existem diferentes tipos de classificação de DM, como explicado em IDF (2021), e são mais comumente conhecidas como Tipo 1, Tipo 2, Diabete Gestacional e outros tipos.

6.1.1 Diabetes Tipo 1

A Diabetes Tipo 1 é causada por uma reação autoimune, onde o sistema imune do corpo ataca as células *beta* do pâncreas, que são responsáveis pela produção de insulina. Como consequência da disfunção das células *beta*, o corpo produz pouco ou não produz insulina (IDF, 2021; WHO, 2019).

Diabetes Tipo 1 pode ser desenvolvida em qualquer idade, mas é mais comum de aparecer na criança e adolescente (IDF, 2021).

Pessoas com diabetes tipo 1 necessitam de um controle diário dos valores de insulina circulantes, para que possam fazer reposição adequada deste hormônio (via injetável), com o intuito de manter os níveis de glicose estáveis. Realizando um controle apropriado do tratamento com insulina, monitoramento dos níveis de glicose do sangue, além de receber um suporte e educação sobre a doença, essas pessoas podem ter uma vida saudável e atrasar ou prevenir maiores complicações relacionadas a diabetes. Para crianças e adolescentes, esse tratamento se torna mais dificultoso, tendo em vista que para a sua realização é necessário o manuseio de lancetas, agulhas e também é preciso fazer cálculos para aplicação da quantidade correta do hormônio (IDF, 2021).

A falta de acesso a insulina pode resultar em complicações fatais da cetoacidose diabética, hiperglicemia e complicações do sistema circulatório (IDF, 2021).

6.1.2 Diabetes Tipo 2

O Diabetes Tipo 2 causa hiperglicemia devido a resistência a insulina, ou seja, as células do corpo não respondem integralmente a insulina. Essa resistência a insulina faz com que o hormônio não tenha efetividade, resultando num aumento da produção de insulina. Com o tempo, as células *beta* do pâncreas acabam perdendo a capacidade de produzir insulina (IDF, 2021; WHO, 2019).

Mais de 90% das pessoas diagnosticadas com DM são pertencentes ao tipo 2. Ela se desenvolve com mais frequência em indivíduos adultos, e pessoas obesas. Porém, o número de crianças e adolescentes com DM2 têm aumentado ao longo dos anos, em decorrência de dieta inapropriada e falta de atividades físicas (IDF, 2021; WHO, 2019).

Os sintomas do Tipo 2 são muito similares com os do Tipo 1, mas geralmente são menos intensos, e algumas vezes a pessoa pode nem apresentar sintomas. Por isto, o tipo 2 é mais difícil de ser detectado, e a pessoa pode ficar por anos com a doença sem ser diagnosticada, e conseqüentemente não é tratada. Quando a pessoa tem a doença por muito tempo sem o devido tratamento, ela pode desenvolver complicações como retinopatia ou úlcera de membro inferior que não cicatriza. Pessoas com Diabetes Tipo 2 apresentam maiores riscos de desenvolverem problemas macrovasculares e microvasculares (IDF, 2021; WHO, 2019).

O controle para pessoas com Diabetes Tipo 2 inclui alterar seu estilo de vida, melhorando seus hábitos alimentares, atividade física regular e manter um peso corporal saudável. Em casos onde essa mudança não é suficiente para controle os níveis de glicose no sangue, o profissional realiza um tratamento com o apoio de medicamentos, e quando isso também não é suficiente, então necessita-se de injeções de insulina (IDF, 2021).

6.1.3 Diabetes Gestacional

A Diabete Gestacional (DG) envolve o diagnóstico de hiperglicemia em qualquer período da gravidez. Testes para verificar a doença são altamente recomendados entre a 24^a e 28^a semana da gravidez, mas para mulheres com maiores riscos de desenvolverem a doença deve ser realizado um acompanhamento durante toda a gravidez. Os fatores de risco incluem idade avançada, sobrepeso e obesidade, casos de hiperglicemia em outra gravidez, ganho excessivo de peso durante a gravidez, genética, entre outros (IDF, 2021).

A DG geralmente é autolimitada ao período gestacional, porém, em alguns casos pode se prolongar e se estabelecer. Além disso, em gravidezes futuras o risco de desenvolver DG novamente é mais elevado. Crianças nascidas de mães que tiveram DG também tem um maior risco de desenvolver a doença (IDF, 2021).

O DG traz riscos para a mãe e para o bebê. A mãe ao ser diagnosticada necessita de acompanhamento médico mais frequente, pela sua gestação ser considerada de risco. O bebê geralmente apresenta um peso muito superior ao normal para a idade gestacional, dificultando a sua passagem pelo canal de parto, podendo também haver danos neurológicos e fraturas. A

identificação prévia da doença na gravidez, o controle da glicose no sangue e acompanhamento médico podem reduzir os riscos citados (IDF, 2021).

6.1.4 Outros Tipos de Diabetes

Segundo a OMS existe uma lista de outros tipos específicos de diabetes, sendo entre eles o diabete monogênico. Este tipo de diabete é resultado de um único gene, deferentes da contribuição de múltiplos genes e de fatores devido ao ambiente como os Tipos 1 e 2. Existe também a Doença do pâncreas exócrino, Desordens Exócrinas, Diabetes Induzida por drogas ou química, Diabetes relacionadas a Infecções, entre outras (IDF, 2021).

7 TRABALHOS RELACIONADOS

O cuidado com o DM é multidimensional como resultado de complexas interações de ambiente, estilo de vida, genético e clínico. Cada pessoa tem seu perfil de riscos, e o acesso contínuo a tratamentos e acompanhamentos, em uma rede multidisciplinar, auxilia os profissionais a serem mais assertivos no tratamento da doença (IDF, 2021; MARTINEZ *et al.*, 2019).

Devido ao aumento das morbidades relacionadas a diabetes e a falta de fortes evidências, acompanhamentos e prevenções da doença, a OMS anunciou a *Global Diabetes Compact* (GDC), que visa estimular e priorizar o monitoramento da doença. Um dos objetivos da GDC é de realizar ações para melhorar o acesso a medicamentos, e o desenvolvimento de tecnologias para o tratamento da doença como parte dos esforços de fortalecimento da atenção primária à saúde, em particular em países de baixa e média renda (WHO, 2022).

Estudos apontam que o acompanhamento da doença reduz em 10 anos riscos de complicações microvasculares (neuropatia, retinopatia, doença renal). Em GREGG *et al.* (2021) são apresentadas diversas pesquisas e trabalhos para os diferentes tipos de marcadores que são importantes para realizar um acompanhamento, como HbA1c, pressão arterial e colesterol, mas os autores relatam que os tamanhos inadequados de amostras da população estudada podem apresentar diagnósticos imprecisos sobre a doença.

Na meta análise de SU *et al.* (2016), é apresentado o resultado da aplicação da telemedicina para o acompanhamento de pacientes com DM. Este trabalho analisa a comparação do marcador HbA1c em 4607 pacientes com o acompanhamento de telemedicina,

e 4651 pacientes com o tratamento convencional. Os resultados do estudo apresentam um controle maior da HbA1c em pacientes acompanhados com telemedicina, e que os benefícios são maiores entre pacientes com DM Tipo 2.

O uso de diferentes dispositivos e sistemas para realizar a aquisição de dados e monitoramento resulta na complexidade de comunicação entre sistemas de saúde, aumentando a carência de sistemas integrados voltados para a DM. Muitos trabalhos utilizam o HL7 FHIR para solucionar esse desafio presente nos sistemas de saúde do mundo todo. MAXHELAKU e KIKA (2019) apresentam a dificuldade do compartilhamento dos dados coletados nas diferentes unidades de saúde da Albânia, principalmente dados dos setores de radiologia. Neste trabalho descreve o uso do HL7 FHIR e como ele permite o acesso remoto dos dados e auxilia radiologistas no monitoramento dos pacientes.

O trabalho de HIDAYAT e HERMANTO (2020) avalia o uso do HL7 FHIR para atender as necessidades de interoperabilidade nos centros de Atenção Básica de Saúde (ABS) na Indonésia, onde o padrão FHIR atende 65,8% da representação dos dados, e são desenvolvidos perfis (“*Profiles*”) para atender os demais dados.

8 SCREENING FOR DIABETES COMPLICATIONS

Existem diversas diretrizes para realizar um acompanhamento, ou *screening*, para a DM no mundo. No Brasil temos o acompanhamento de exames laboratoriais, tais como Glicemia de jejum mensal, colesterol LDL anual, colesterol HDL anual, triglicérides anual, pressão arterial a cada consulta, HbA1c a cada 3 meses até alcançar o controle, após é realizada a cada 6 meses. Outras avaliações são recomendadas para o acompanhamento da doença, como orientação alimentar e verificar se o paciente está realizando atividade física (BRASIL, 2006).

Para a prevenção do avanço da DM para outras complicações, como a Neuropatia Diabética (ND) são necessárias consultas periódicas. A ND compreende um conjunto de síndromes clínicas que afetam o sistema nervoso periférico sensitivo, motor e autonômico. A consulta clínica para ND deve ser realizada anualmente, o qual o profissional irá realizar uma série de testes para avaliar os tipos de sensações (dolorosa, tátil, térmica, vibratória, motora, limiar de percepção cutânea). Outros testes procuram verificar e diagnosticar a neuropatia autonômica cardiovascular, onde os testes avaliam a pressão arterial. A avaliação para Retinopatia Diabética (RD), doença que pode causar cegueira irreversível, é diagnosticada por um profissional oftalmologista, e recomenda-se a avaliação a cada 1-2 anos (BRASIL, 2006).

Buscando atender a multidisciplinaridade e desenvolvimento tecnológico para alcançar um sistema de *screening* como ferramenta para coleta, estudo e acompanhamento da DM, o IEB-UFSC tem desenvolvido o sistema ubíquo *Screening for Diabetes Complications* (SDC-X). O “X” da sigla refere-se aos diferentes módulos de análise, e.g. SDC-Cardio, SDC-Eye, SDC-Neuro, etc.

O SDC-X possui aplicações em pesquisa e gestão na área da saúde com o princípio de realizar a detecção precoce de complicações do DM e de um mapeamento histórico e geográfico da doença. O sistema busca avaliar os pacientes de diferentes localidades aplicando diferentes técnicas e análises nos dados adquiridos, através de um armazenamento e acesso interoperável e seguro (TERRA, 2020; ÜCKER, 2020; VIEIRA, 2021).

O sistema SDC-X é composto por 4 módulos, como apresentado na Figura 7, sendo eles o Módulo de Aquisição, Módulo Concentrador, Módulo de Armazenamento e Módulo de Visualização e Interface.

Figura 7 – Módulos do projeto SDC-X. O Módulo de Aquisição é composto por *hardware*, composto por equipamentos para realizar medidas fisiológicas, e os outros módulos são essencialmente *softwares*.

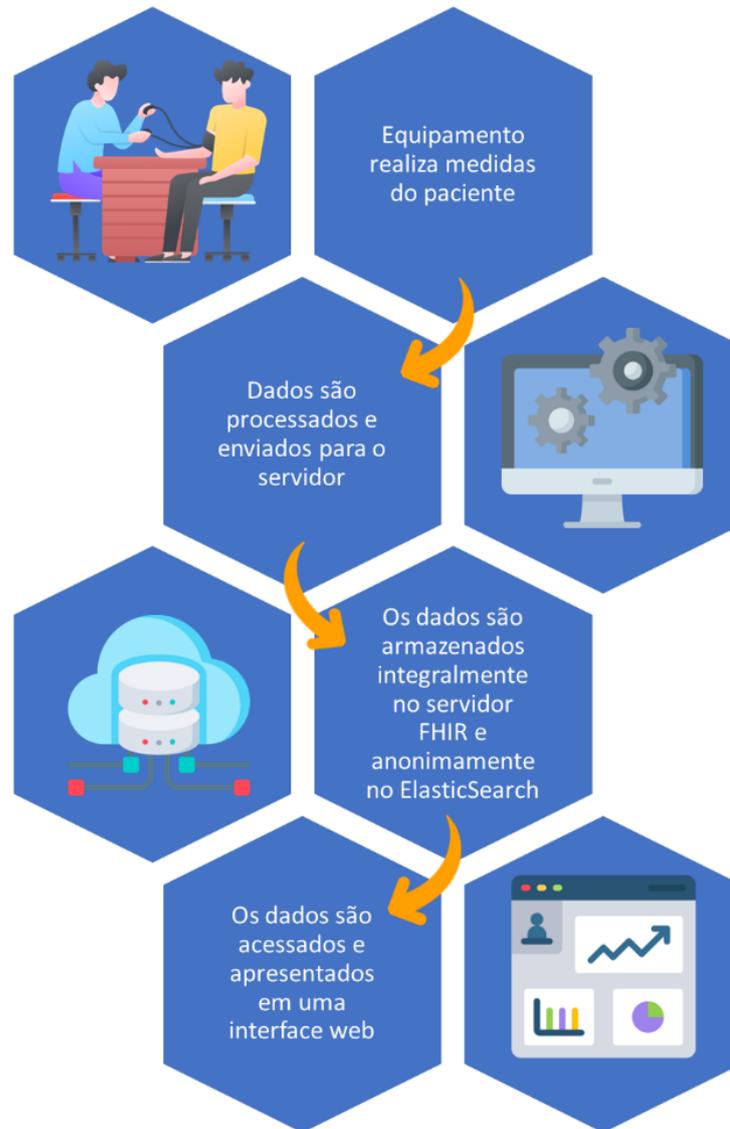


Fonte: Elaborado pelo autor, 2022

O projeto compõe então 4 fases, onde pode-se ver na Figura 8, a etapa de aquisição dos dados paciente, que será realizado por um equipamento do Módulo de Aquisição. Estes dados coletados são processados e analisados através de um computador contendo o *software* de análise do Módulo Concentrador. Em seguida, os dados coletados e gerados pelas análises são enviados para os servidores, e assim podem ser interpretados por outros usuários através de

uma plataforma *web*, representados pelo Módulo de Armazenamento e Módulo de Interface, respectivamente.

Figura 8 – Etapas dos módulos do projeto SDC-X.



Fonte: Elaborado pelo autor, 2022.

O Módulo de Aquisição é essencialmente composto por *hardware*, que são os equipamentos que irão realizar a coleta de sinais fisiológicos (e.g. batimento cardíaco, respiração, eletromiografia). Esses equipamentos são manuseados por um profissional capacitado que irá coletar dos dados do paciente. Os dados são enviados (com ou sem fio) para um computador que tenha o *software* SDC-X instalado. Este módulo é composto por diversos

equipamentos já projetados e que ainda estão sendo desenvolvidos (TERRA, 2020; ÜCKER, 2020).

O Módulo Concentrador é o *software* que irá receber os dados do Módulo de Aquisição. Este módulo é um *software* que deve ser instalado em um computador e que deve estar presente e atuante no momento da coleta de dados pelo Módulo de Aquisição. O Módulo Concentrador irá analisar e processar os dados, e por fim enviá-los ao servidor composto pelo Módulo de Armazenamento utilizando o protocolo FHIR (TERRA, 2020; VIEIRA, 2021).

O Módulo de Armazenamento é composto por um servidor FHIR implementado e validado por Vieira (2021), e como adição ao trabalho anterior, este trabalho acrescenta mais um servidor, o *ElasticSearch*. Esses servidores servem para armazenar e facilitar o compartilhamento e acesso aos dados coletados, seguindo o padrão FHIR e protocolos de segurança.

O Módulo de Visualização e Interface é desenvolvido neste trabalho e busca apresentar uma visualização dos dados coletados pelo sistema utilizando um protocolo interoperável, protocolo FHIR, que facilita o compartilhamento e estruturação da informação. Esta interface é composta um sistema de usuário, que irá apresentar uma interface diferente de acordo com o perfil cadastrado no sistema e suas permissões. Este módulo tem como objetivo facilitar a visualização dos dados de forma direta e intuitiva para profissionais da saúde que irão realizar o acompanhamento dos pacientes, para gestores da área da saúde para poderem criar políticas e ações públicas de forma assertiva e baseada em evidências, e para pesquisadores como ferramenta de análise e pesquisa com dados coerentes e precisos.

9 MATERIAIS E MÉTODOS

Neste capítulo são apresentados a utilização dos servidores FHIR (versão R4) validado por (VIEIRA, 2021) e *ElasticSearch* (versão 7.9.2) e a implementação e estruturação de uma plataforma *web* para visualização e análise dos dados contidos pelos servidores.

9.1 SERVIDORES FHIR E *ELASTICSEARCH*

A implantação e validação do servidor HAPI FHIR foi realizada no trabalho de Vieira (2021) onde foram apresentados os conceitos, utilização e funcionamento desse servidor.

Para a utilização do *ElasticSearch*, é necessário realizar uma instalação através do Docker (versão 4.3.0) para criar um *deploy* (implantação final de uma aplicação). Neste trabalho são realizados dois *deploys*, um para o servidor FHIR e um para o *ElasticSearch*, ambos utilizando imagem Docker e executados em paralelo.

O desenvolvimento deste trabalho e os testes foram realizados em uma máquina local, um notebook com processador 1,8 GHz Inter i7-8550U, memória RAM de 12 GB (2666 MHz DDR4) e sistema operacional Windows 10 Pro.

Os *deploys* foram realizados utilizando a tecnologia de *container*, que combina a aplicação, dependências relacionadas e bibliotecas de sistemas para que seja executada em um ambiente de computação. Este ambiente, ou plataforma, é conhecido como Docker, que verifica se a aplicação funciona em todos os ambientes. As aplicações podem ser desenvolvidas, distribuídas e executadas por meio de imagens utilizando o serviço Docker. Uma imagem de *container* do Docker é um pacote de *software* leve, autônomo e executável que inclui tudo que é necessário para executar uma aplicação. A execução de imagens *containers* são realizadas pelo *software Docker Engine*. A Docker (2021) descreve três benefícios no uso de *containers* executados pelo *Docker Engine*, são eles:

- Padronizado: padrão criado pela Docker no setor de *containers*, permitindo a portabilidade em qualquer lugar;
- Leve: os *containers* compartilham o *kernel* do sistema operacional da máquina e, portanto, não exigem um sistema operacional por aplicativo, aumentando a eficiência do servidor e reduzindo os custos de servidor e licenciamento;

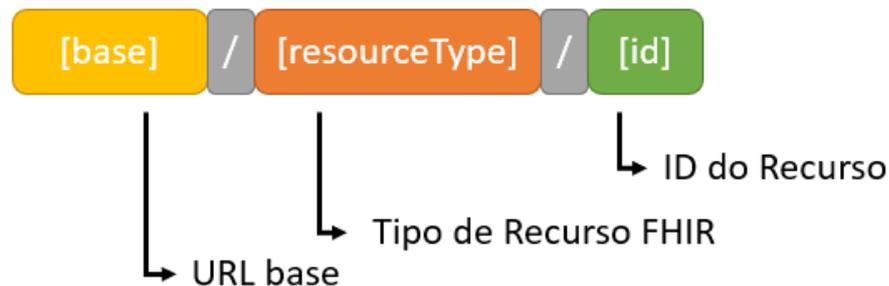
- Seguro: os aplicativos são mais seguros em *containers* e o Docker fornece os recursos de isolamento padrão mais fortes do setor.

9.1.1 Interface RESTFUL API

Ambos os servidores contam com uma RESTful API própria para suas aplicações, e são alocados em portas diferentes, respondendo às requisições conforme sua aplicação. Para montar e validar o funcionamento das requisições RESTful, será utilizado o *software* Postman.

Em relação aos recursos de um servidor HL7 FHIR, existe uma URL que identifica o recurso e especifica onde ele é acessado. Por exemplo, quando um recurso for acessado via FHIR RESTful API, tem-se a seguinte composição da URL apresentada da Figura 9 (HL7 FHIR, 2022).

Figura 9 - URL para acessar os dados dos recursos do servidor HL7 FHIR.



Fonte: Elaborado pelo autor, 2022.

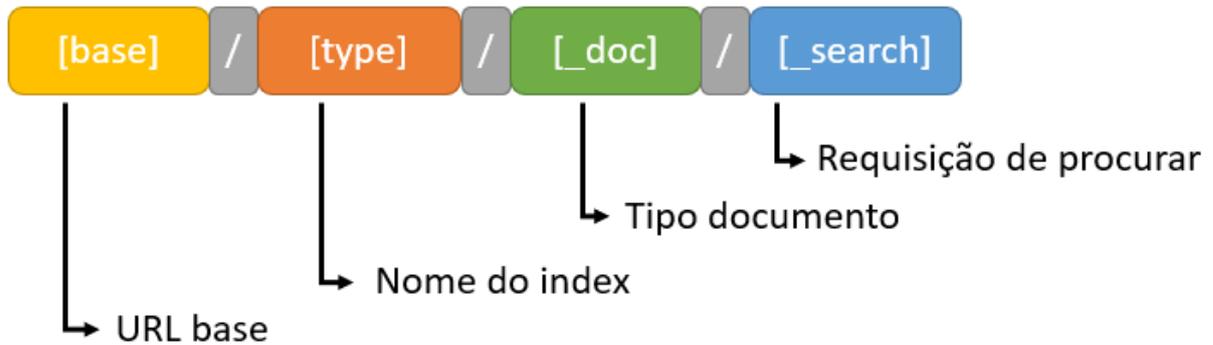
O *ElasticSearch* é um mecanismo de busca e análise de dados distribuído, gratuito e aberto para todos os tipos de dados, podendo ser textuais, numéricos, geoespaciais, estruturados e não estruturados. Ele suporta linguagens como JAVA, Python, C#, PHP e outras linguagens convencionais. Desenvolvido sobre o Apache Lucene, o *ElasticSearch* foi lançado em 2010 pela *ElasticSearch* N.V. (hoje conhecida como Elastic) (HE, 2020; ELASTIC, 2022).

A pesquisa por dados no *ElasticSearch* utiliza a ferramenta de indexação para um conjunto de dados. Essa indexação permite criar um *template* para os dados a serem inseridos, ou gerar automaticamente a partir do primeiro conjunto de dados recebido.

No servidor *ElasticSearch* implantado, buscou-se facilitar o desenvolvimento e coerência dos dados, criando uma indexação para cada “Resource” do padrão FHIR.

Para facilitar a implementação e coerência dos dados, é criada uma indexação para cada “Resource” do padrão FHIR. A URL de acesso aos dados no servidor *ElasticSearch* é apresentada na Figura 10.

Figura 10 - URL para acessar os dados de um documento do servidor *ElasticSearch*.



Fonte: Elaborado pelo autor, 2022.

Em ambos os servidores, para saber se as requisições são realizadas corretamente, a arquitetura RESTful utiliza códigos de respostas HTTP, existindo 41 tipos oficialmente, e são agrupadas em cinco classificações:

- Respostas de informação (100 - 199);
- Respostas de sucesso (200 -299);
- Redirecionamentos (300 - 499);
- Erros de cliente (400 - 499);
- Erros do servidor (500 - 599).

Algumas delas são listadas no quadro abaixo.

Quadro 1 – Respostas de status de requisições para uma arquitetura RESTful.

| Status | Significado |
|--------------------------|--|
| 200 “Ok” | Requisição executada e aceita com sucesso. |
| 201 “Created” | Requisição bem sucedida e um novo recurso foi criado. |
| 202 “Accepted” | Requisição aceita, mas nenhuma ação foi tomada sobre ela. |
| 204 “No Content” | Requisição bem sucedida, mas sem retorno de conteúdo. |
| 400 “Bad Request” | O servidor não entendeu a requisição pois existe um erro de sintaxe na solicitação. |
| 401 “Not Authorized” | Autorização necessária para realizar a requisição ao servidor. |
| 404 “Not Found” | O servidor não encontrou o recurso solicitado. |
| 405 “Method Not allowed” | A solicitação ao servidor é conhecida pelo servidor, mas foi desativado devido a atualizações. |
| 409 “Conflict” | Conflito com o estado atual do servidor. |

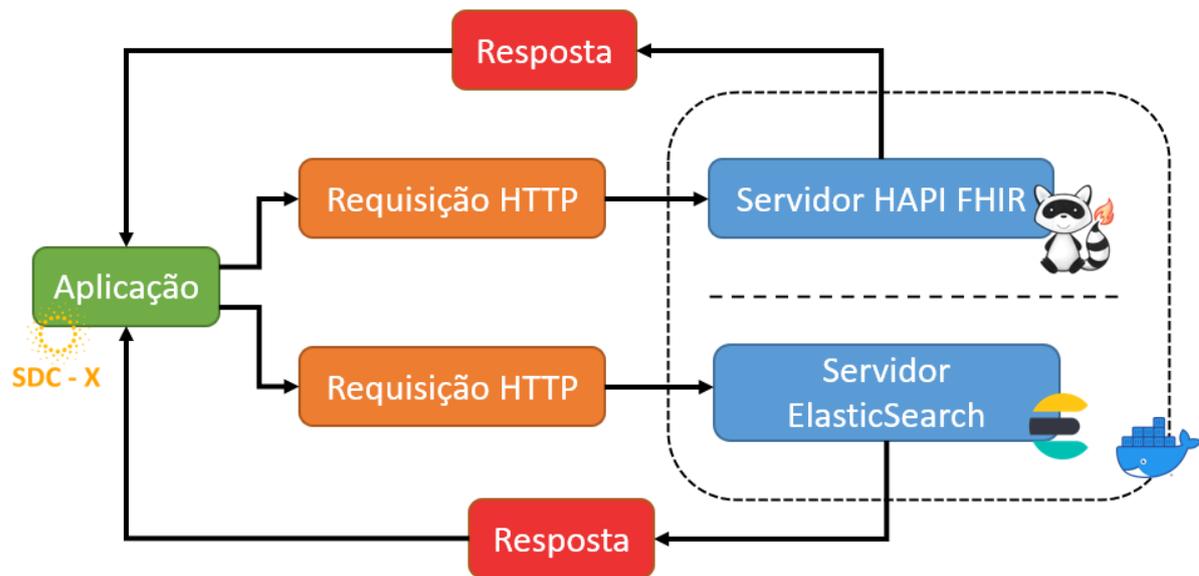
| | |
|------------------------------------|---|
| 422 “Unprocessable Entity” | Requisição realizada é válida, mas existem erros semânticos. No FHIR pode representar uma violação de um perfil. |
| 500 “Internal Server Error” | Falha interna do servidor. |
| 501 “Not Implemented” | Método da requisição não é suportado e não pode ser manipulado pelo servidor. |
| 502 “Bad Gateway” | Significa que o servidor ao trabalhar como <i>gateway</i> para obter uma resposta necessária para manipular a requisição, acabou recebendo uma resposta inválida. |
| 503 “Service Unavailable” | Servidor indisponível para manipular as requisições recebidas. |

Fonte: (RICHARDSON; RUBY, 2007; HL7 FHIR, 2022)

9.2 INCLUSÃO DE DADOS AOS SERVIDORES

Os servidores HAPI FHIR e *ElasticSearch* devem ser executados em paralelo como containers utilizando o *Docker Engine*, e os dados são incluídos em cada servidor utilizando suas RESTful API's, mantendo o padrão HL7 FHIR (Figura 11).

Figura 11 - Diagrama da realização de requisições RESTful para os servidores HAPI FHIR e *ElasticSearch*.



Fonte: Elaborado pelo autor, 2022.

Os acessos a esses dados devem ser realizados através de uma requisição de leitura, utilizando o parâmetro GET. Os servidores têm propósitos diferentes na aplicação da interface, de forma que, o servidor HAPI FHIR é utilizado para visualização dos dados de pacientes de forma individualizada, isto é, um valor de identificação deve ser utilizado na requisição HTTP. O Servidor *ElasticSearch* serve para um propósito de visualização dos dados em conjunto, o

que significa que nenhum identificador é necessário, e a resposta da requisição deve ser um conjunto de dados relacionado à pesquisa desejada.

9.2.1 Dados Sintéticos

A validação da utilização dos servidores é realizada por meio de dados fictícios gerados por uma ferramenta *open-source* de Simulador de População Sintéticos chamado *Synthea*. Este simulador permite gerar dados realísticos de pacientes em diferentes formatos, incluindo o HL7 FHIR (SYNTHEA, 2022).

Esses dados simulam condições de pacientes com informações desde o nascimento até o falecimento, condições vitais, alergias, medicamentos, vacinas, procedimentos, observações e outros tipos informações (SYNTHEA, 2022).

Neste trabalho, conseguiu-se realizar um estudo das ferramentas a serem utilizadas e desenvolvidas sem restrições legais, como aprovação de comitê de ética, restrições de privacidade e segurança.

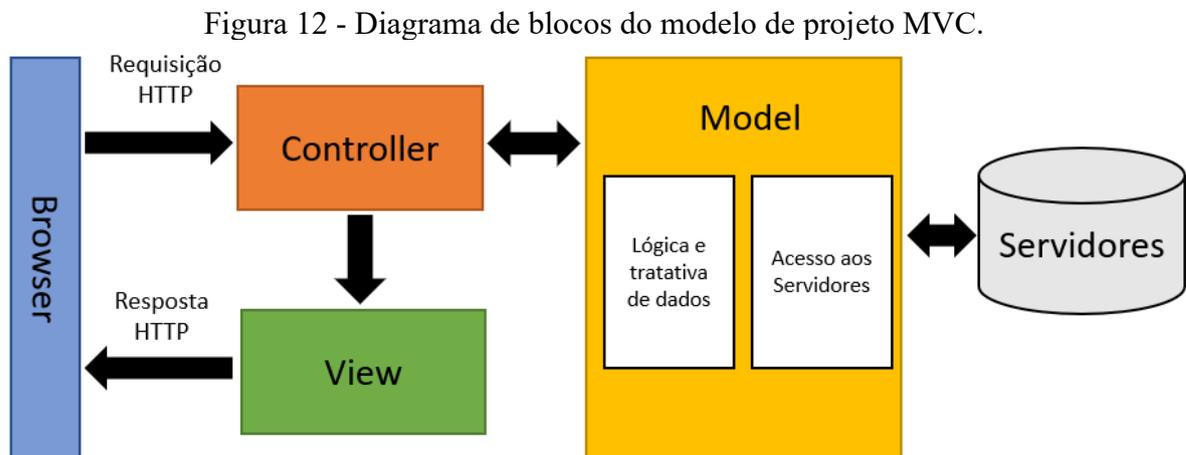
9.3 INTERFACE SDC-X

A plataforma *web* representa o Módulo de Visualização e Interface do projeto SDC-X. Ela tem como finalidade apresentar as informações dos pacientes, seus dados coletados através do Módulo de Aquisição, e mostrar, de forma intuitiva, a visualização desses dados para diferentes perfis de usuários. Essa plataforma conta também com ferramentas de segurança através de código, criptografia e funcionalidades manuais para atender a Lei Geral de Proteção de Dados (LGPD) lei nº 13.709 de 14 de agosto de 2018 (BRASIL, 2020a).

A interface foi desenvolvida na linguagem de programação PHP (*Hypertext Preprocessor*) no padrão de design de projeto MVC (*Models-Views-Controllers*) e testada localmente utilizando o servidor *web* XAMPP (versão 3.3.0), que é um servidor de código aberto desenvolvido pela Apache Friends. Este padrão de design estrutura o projeto a ser desenvolvido em 3 camadas, auxiliando na redução de acoplamento e promovendo o aumento de coesão nas classes do projeto, facilitando a reutilização do código. Este padrão permite grande flexibilidade, replicabilidade, clareza e segurança (GAMMA *et al.*, 2007).

O *framework* utilizado neste trabalho para o modelo MVC usa o *Controller* como única camada para troca de dados entre camadas. Dessa forma o *Controller* se comunica com a

View e com o *Model*, e nunca ocorrerá troca de dados direta entre *View* e *Model*, como apresentado na Figura 12.



Fonte: Elaborado pelo autor, 2022.

Cada camada representa uma função dentro do padrão MVC, sendo eles:

- **Controller:** Processa e responde a eventos, geralmente ações do usuário. Invocado através das requisições HTTP, esta camada irá realizar troca de informações com o *Model* para adquirir os dados, e invocar a camada *View* para visualização.
- **View:** Visualização da interface para que o usuário possa interagir. Nessa camada podem apresentar lógicas, como para determinar quais elementos da *view* o usuário pode ter acesso.
- **Model:** É a representação específica do domínio dos dados sobre os quais o aplicativo opera e responde as requisições do *Controller*. Nesta camada são realizadas as comunicações com bancos de dados (ADAM; ANDOLO, 2019; CUI *et al.*, 2009).

A interface tem acesso a 5 perfis de usuários: Administrador, Pacientes, Pesquisador, Profissional e Gestor. Os dados apresentados para cada perfil irão seguir uma regra de permissões que serão verificadas pelos blocos *Controller* e *Model*. A descrição de cada perfil de usuário é apresentada abaixo.

- **Administrador:** Realizar o controle de quem pode acessar a interface do projeto.
- **Paciente:** Visualizar e acompanhar seus dados.

- **Pesquisador:** Visualizar e exportar os dados, em formato anônimo, para trabalhar conforme sua necessidade e interesse.
- **Profissional:** Perfil voltado aos profissionais da saúde que poderão realizar análises e consultas aos pacientes de acordo com as informações apresentadas pelo projeto.
- **Gestor:** Terá acesso a uma série de análises do conjunto dos dados coletados. Este perfil tem o intuito de auxiliar na obtenção de informações do sistema de saúde, permitindo assim, traçar metas e soluções de forma embasada e assertiva.

Para atender as demandas de cada perfil mantendo a segurança e proteção dos dados, a Tabela 1 apresenta as funções que cada perfil pode realizar dentro da interface projetada.

Tabela 1 – Funções e permissões de cada perfil de usuário da plataforma.

| Funções | Administrador | Paciente | Pesquisador | Profissional | Gestor |
|--|----------------------|-----------------|--------------------|---------------------|---------------|
| Controle de usuários. | X | | | | |
| Controle de visualização de pacientes por profissionais. | X | | | | |
| Visualização e identificação dos dados. | | X | | X | |
| Visualização anônima dos dados. | | | X | | |
| Exportação dos dados em formato anônimo. | | | X | | |
| Visualização de informações epidemiológicas. | | | | | X |

Fonte: Elaborado pelo autor, 2022.

10 RESULTADOS

Neste capítulo são apresentados os resultados obtidos do sistema proposto, implantado e desenvolvido. Este capítulo está dividido em apresentação dos resultados da implantação e validação dos servidores, resultados da estruturação e desenvolvimento da plataforma *web*, e também validação da plataforma *web* proposta através de dados reais coletados através de outros trabalhos do SDC-X.

10.1 SERVIDORES

Os dados sintéticos utilizados foram exportados do *site* da *Synthea* (SYNTHEA, 2022b) no formato FHIR R4. Os arquivos de amostra contam com 1.000 registros de pacientes, e aplicou-se um filtro para utilizar apenas registros de pacientes com diabetes, resultando em 824 pacientes que foram utilizados no projeto.

Os arquivos exportados contam com recursos em FHIR separados por arquivo, e cada um é composto por recursos de um paciente. Esses dados foram incluídos nos servidores através de uma rotina em PHP, executada através do servidor *web* utilizado com o XAMPP, que verifica se o paciente tem diabetes através do seu código identificador. A verificação utiliza a requisição GET através de um *bundle* para os recursos “*Patient*” e “*Condition*”. Curiosamente, nesses dados não foram encontrados pacientes para os códigos referentes a DM Tipo 1 nem DM Tipo 2, apenas para neuropatia devido a DM Tipo 2.

Os dados utilizados contam com informações de pacientes, como por exemplo, identificação (e.g. nome, sobrenome, cidade, plano de saúde), dados de condições clínicas, medicamentos utilizados, diagnósticos, resultados laboratoriais, entre várias outras informações. Todos esses dados estão relacionados aos seus respectivos recursos de referência como: *Patient*, *Observation*, *Medication*, *Condition*, entre outros.

10.1.1 Servidor FHIR

O *deploy* do servidor FHIR foi executado utilizando a tecnologia *container* através da aplicação HAPI FHIR, validada por Oliveira (2021), através do *Docker Engine* para Windows 10. A imagem do servidor que foi utilizado como modelo pode ser encontrada no repositório oficial do HAPI FHIR (HAPI FHIR, 2022b).

A implantação do *container* funcionou conforme esperado, e em seguida realizou-se o armazenamento dos dados sintéticos através da rotina em PHP, comentada anteriormente. Para esta rotina foi preciso configurar o tempo de execução de um *script* pelo servidor *web*, pois o XAMPP tem como *default* esse tempo em 300 segundos. Como a quantidade de dados é grande, esse tempo é pequeno para realizar todas as requisições de armazenamento ao servidor. Para resolver este problema, alterou-se o tempo para 3600 segundos, através da função “*set_time_limit*” do PHP.

A RESTful API foi validada através do *software* Postman, onde foram testadas diferentes requisições. Para a interface, utilizou-se os seguintes recursos (HL7 FHIR, 2022):

- *Patient*: informações demográficas e outras informações administrativas de um indivíduo.
- *DiagnosticReport*: apresenta os resultados de interpretações de testes de diagnósticos realizados em pacientes, como contexto clínico, resultados de exames, interpretações textuais e relatórios de diagnósticos.
- *Observation*: medições e afirmações simples feitas sobre um paciente.
- *MedicationAdministration*: descreve o evento de um paciente consumir ou receber um medicamento.

O APÊNDICE A – Requisições FHIR do projeto apresenta todas as requisições FHIR utilizadas para o projeto e seus objetivos.

10.1.2 Servidor *ElasticSearch*

O *deploy* do servidor *ElasticSearch*, como mencionado anteriormente, também foi executado utilizando a tecnologia *container* através do *Docker Engine*. Desta forma, é possível realizar a execução dos dois servidores em paralelo, em *containers* separados, pelo Docker.

A implantação do *container* e funcionamento de ambos os servidores em paralelo funcionou conforme esperado. A mesma rotina criada para atribuir os dados ao servidor HAPI FHIR foi utilizada para adicionar os mesmos dados ao servidor *ElasticSearch*, mas optou-se por anonimizar os dados. Dessa forma, apenas os dados do HAPI FHIR contam com todas as informações íntegras dos pacientes.

A anonimização dos dados atribuídos ao servidor *ElasticSearch* realiza a substituição de informações nos recursos de *Patient*, substituindo nome, sobrenome, números de

documentos, rua e telefone por campo vazio. O campo de data de nascimento é substituído por idade. O id de identificação dos pacientes não é alterado, dessa forma, caso exista a necessidade de vincular os dados dos outros recursos, pode-se utilizar o id como informação em comum.

Os dados foram importados para o servidor através do método POST. Foi criado um *index* de identificação para recurso, desta forma os dados ficam agrupados por recursos dentro do *ElasticSearch*.

A RESTful API do *ElasticSearch* também foi validada utilizando o Postman. Diferente das requisições para o servidor HAPI FHIR, a ferramenta de pesquisa por requisições RESTful permite utilizar a linguagem SQL no corpo de uma requisição POST. Esta possibilidade torna o *ElasticSearch* em uma ferramenta muito útil para realizar qualquer tipo de pesquisa dentro do servidor. Infelizmente existem algumas limitações nas *queries* SQL que o *ElasticSearch* permite, mas que podem ser contornadas utilizando outras *queries* para a lógica, resultando em *queries* longas (ELASTIC, 2022). O APÊNDICE B – Requisições *ElasticSearch* do projeto apresenta todas as requisições utilizadas para o projeto e seus objetivos.

10.2 INTERFACE SDC-X

A plataforma *web* foi desenvolvida como uma aplicação PHP utilizando o padrão de camadas MVC, e essas camadas são divididas pelas linguagens de programação utilizadas, onde tem-se o *Controller* e *Model* utilizando PHP e a *View* com as linguagens HTML, CSS e *JavaScript*. Na *View* o *JavaScript* pode realizar ações de interações do usuário dentro da interface.

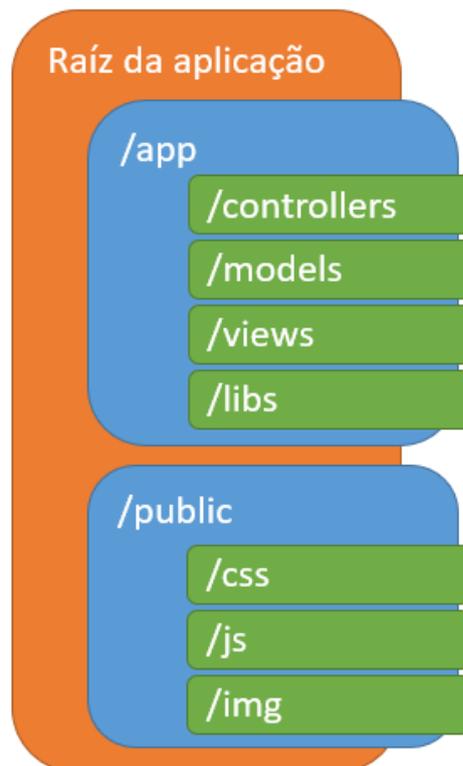
Na construção das *views* foi utilizado o *framework* Materialize, que é um *framework* de CSS moderno baseado no conceito de *Material Design* criado pela Google. Esse conceito traz o objetivo de permitir uma experiência unificada pelo usuário em qualquer plataforma (MATERIALIZE, 2022).

Para os gráficos gerados para a interface, foi utilizada a biblioteca Chart.js. Esta biblioteca, *open source* e mantida pela comunidade, auxilia na visualização de dados utilizando *JavaScript*. Ela tem suporte para visualizar dados em 8 tipos de gráficos diferentes e todos são responsivos (se adaptam para diferentes dispositivos e tamanhos de tela) (CHARTJS, 2022).

A aplicação é executada pelo servidor *web* XAMPP e tem como “index” da aplicação a execução das configurações. Essas configurações definem o correto endereçamento da aplicação, tornando assim acessível a partir de qualquer diretório *web* que a aplicação esteja

sendo executada. Ele permite verificar a certificação SSL, relógio de referência (São Paulo) e o comportamento das invocações das classes no projeto. Através da função “*spl_autoload_register*”, é possível invocar qualquer classe que esteja dentro dos endereços “*app/libs*” e “*app/models*”. A Figura 13 apresenta a organização e estruturação das pastas para os arquivos do projeto.

Figura 13. Estruturação de pastas do projeto desenvolvido.



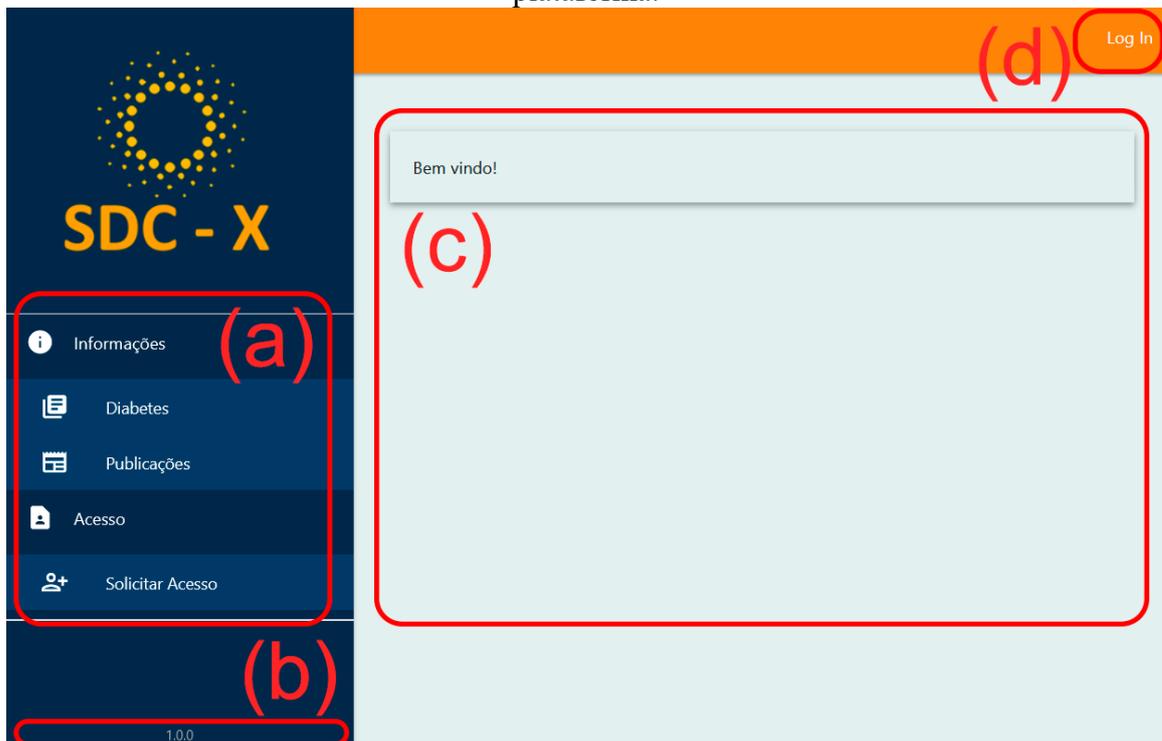
Fonte: Elaborado pelo autor (2022).

O projeto foi dividido em pastas “*app*” e “*public*”. A pasta “*app*” é dividida pelo modelo MVC, ou seja, uma pasta para cada camada mais uma pasta “*libs*”, que contém as bibliotecas desenvolvidas para a aplicação. A pasta “*public*” contém os arquivos de acesso público, divididos em “*css*”, “*img*” e “*js*”, representando os arquivos de código CSS, imagens e códigos *JavaScript*.

Uma característica adotada para todas as classes utilizadas no modelo MVC, é a utilização do padrão “*CamelCase*” para o nome das classes, e é criado um arquivo *script* para cada classe individualmente que segue o mesmo padrão de nomenclatura. Este padrão adiciona

a primeira letra maiúscula para a composição de um nome, e. g. a classe para determinar os acessos ao servidor HL7 FHIR é um arquivo *Model*, assim nomeamos como “*ModelHL7*”.

Figura 14 - Tela inicial da plataforma de acesso público. a) Menus de navegação. b) Versão da plataforma. c) Local de apresentação das informações. d) Botão de login para acessar a plataforma.



Fonte: Elaborado pelo autor, 2022.

A interface permite alguns acessos públicos (Figura 14), com páginas de apresentação do projeto SDC-X, informações sobre a DM, publicações referentes ao projeto e um ambiente para solicitação de acesso às funcionalidades da interface de acordo com cada perfil. A interface apresenta um menu lateral esquerdo (Figura 14a) para acessar as páginas mencionadas, e no canto superior direito (Figura 14d) um botão de acesso ao painel de *login*. Um versionamento da interface é apresentado no canto inferior esquerdo (Figura 14b).

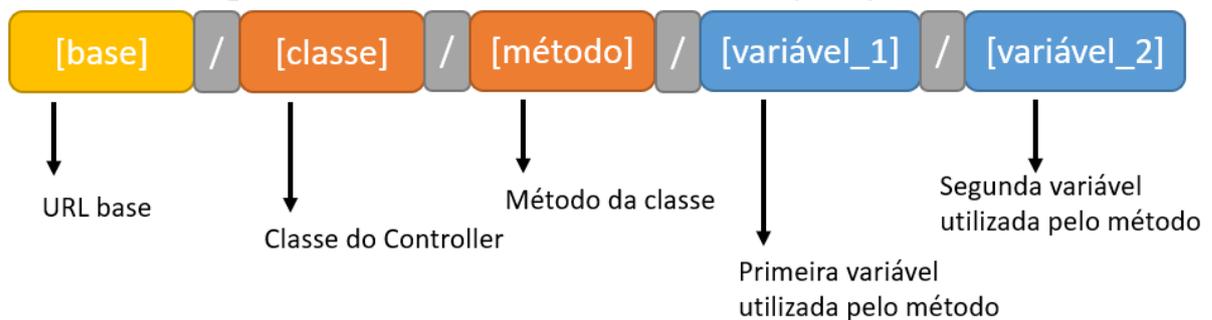
10.2.1 Roteamento e Endereçamento

No arquivo de configurações é criado o formato de roteamento da aplicação, ou seja, como o *browser* deve procurar e executar os arquivos para apresentar ao usuário a tela de visualização. O roteamento da aplicação deve seguir primeiro pelo endereço local, classe de um

Controller, e se necessário, a continuação são valores utilizados pela classe *Controller* invocada, como apresentado nas Figura 15, Figura 16 e Figura 17.

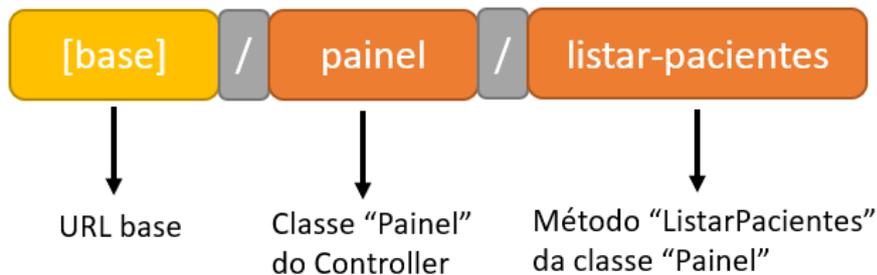
Seguindo o padrão de endereço de outras aplicações *web* (e. g. *Wordpress*), o nome das classes segue o padrão “*CamelCase*”, mas o endereço deve separar as palavras compostas por hífen. Um exemplo de endereço utilizando esta lógica é para a lista de pacientes, método da classe *Controller* “*Painel*”. O método dentro da classe é escrito como “*listaDePacientes*”, no endereço deve-se digitar “*lista-de-pacientes*”.

Figura 15 – Formato do roteamento de endereço da plataforma.



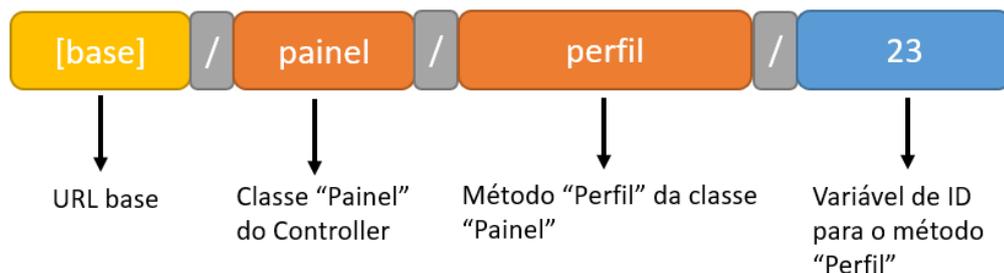
Fonte: Elaborado pelo autor, 2022.

Figura 16 – Endereço para acessar a lista de pacientes após ter realizado *login* na plataforma.



Fonte: Elaborado pelo autor, 2022.

Figura 17 – Exemplo de acesso ao perfil do usuário de *id* 23.



Fonte: Elaborado pelo autor, 2022.

Caso a página endereçada não exista, ou seja, uma classe inexistente no *Controller*, a visualização de uma página para 404 é apresentada (Figura 18).

Figura 18 – Página para erro 404 e página não encontrada caso o endereço procurado não exista nenhuma classe correspondente de *Controller*.



Fonte: Elaborado pelo autor, 2022.

10.2.2 Controle de Acesso

Os acessos a interface devem ser requisitados através dos formulários no painel “Solicitar Acesso”. Nesses formulários deve-se preencher os dados e selecionar qual tipo de acesso deseja (Pesquisador, Profissional ou Gestor). O perfil Administrador terá acesso a lista de requisições e as informações de quem solicitou, para avaliar e tomar a decisão de permitir o acesso ou não a interface SDC-X.

Clicando no botão de *login* apresentado na Figura 14, o usuário é redirecionado para uma página com os campos de *login* e senha, como mostra a Figura 19.

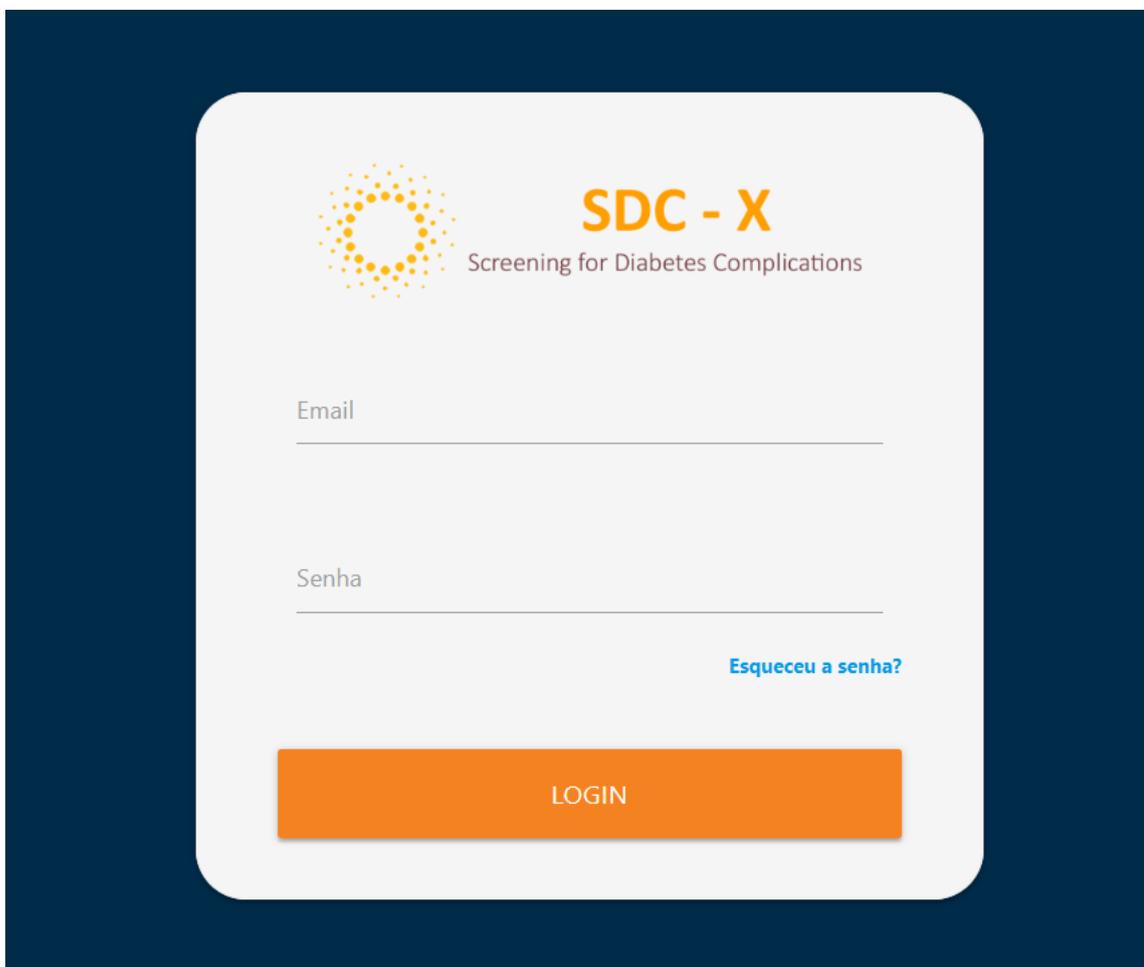
São criadas 3 tabelas em um banco de dados MySQL para realizar a lógica de acesso à interface SDC-X. A estruturação do banco de dados das tabelas comentadas acima é apresentada na Figura 20.

A tabela “*sdc_infousuario*” armazena as informações preenchidas pelo formulário do usuário que fez o requerimento de acesso, e a coluna “*status*” desta tabela representa se o usuário tem acesso ou não a interface através de três valores, “*ativo*”, “*bloqueado*” ou “*pendente*”.

A categorização dos tipos de perfis é realizada na tabela “*sdc_tipousuario*”, onde a coluna “*perfil*” descreve o tipo de perfil com seus respectivos indicadores através da coluna “*id*”.

A verificação de *login* e senha é realizada consultando a tabela “*sdc_usuario*”, onde constam o *email* de acesso, senha e o perfil de usuário através da coluna “*tipo_usuario_id*”.

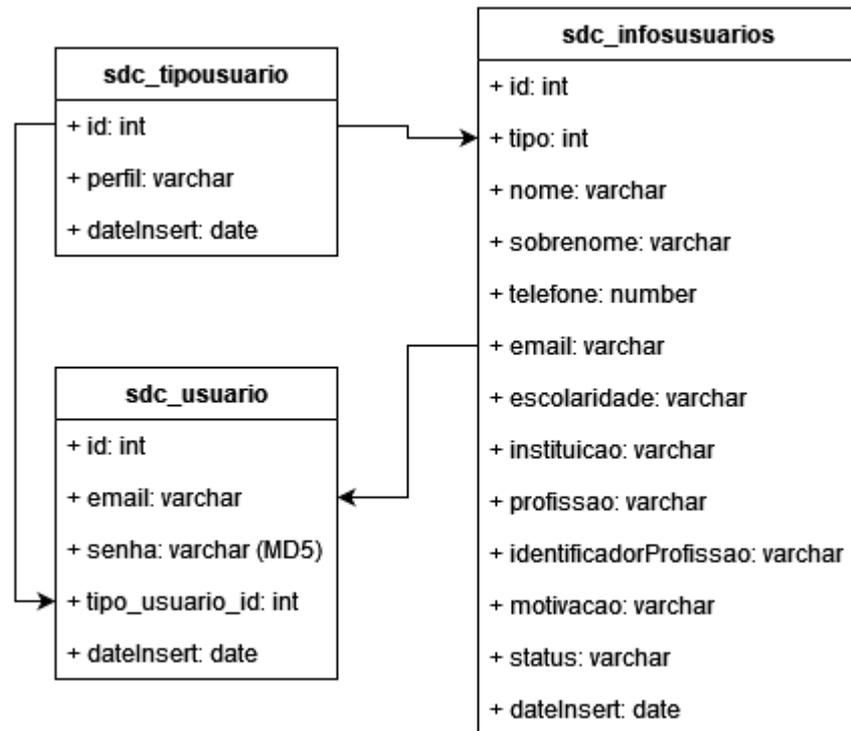
Figura 19 - Página de *login* e senha para o usuário acessar a Plataforma SDC-X.



A imagem mostra a interface de login da Plataforma SDC-X. No topo, há o logotipo da plataforma, que consiste em um círculo de pontos amarelos, e o texto "SDC - X" em laranja, seguido por "Screening for Diabetes Complications" em cinza. Abaixo, há dois campos de entrada: "Email" e "Senha", ambos com linhas de texto cinzas. À direita do campo "Senha", há um link azul "Esqueceu a senha?". No final do formulário, há um botão laranja com o texto "LOGIN" em branco.

Fonte: Elaborado pelo autor, 2022.

Figura 20 - Diagrama de relação entre as tabelas do banco de dados referentes ao usuário.



Fonte: Elaborado pelo autor, 2022.

A lógica para o controle de acesso às páginas acessadas pelos usuários e determinadas *views*, é configurada a partir de uma sessão no PHP. As sessões em PHP consistem em preservar dados do usuário através de acessos subsequentes, através de acessos a múltiplas páginas. O acesso a uma página *web* ganha um identificador único, chamado de *id* de sessão, que é salvo em um *cookie* do lado do usuário ou propagado via URL. Esse suporte a sessão permite armazenar dados no *array* super global `$_SESSION`, e assim, quando um usuário realiza um *login* na plataforma, são armazenados os dados de *login*, id de usuário, e permissão (PHP, 2022; W3SCHOOLS, 2022).

Dessa forma é realizada uma verificação das permissões do usuário sempre que uma página é acessada pelo *Controller*, através das informações contidas no `$_SESSION`. Se a variável de sessão não tiver nenhuma informação, o *Controller* envia o usuário para a página de *login*.

10.2.3 Perfis da Interface SDC-X

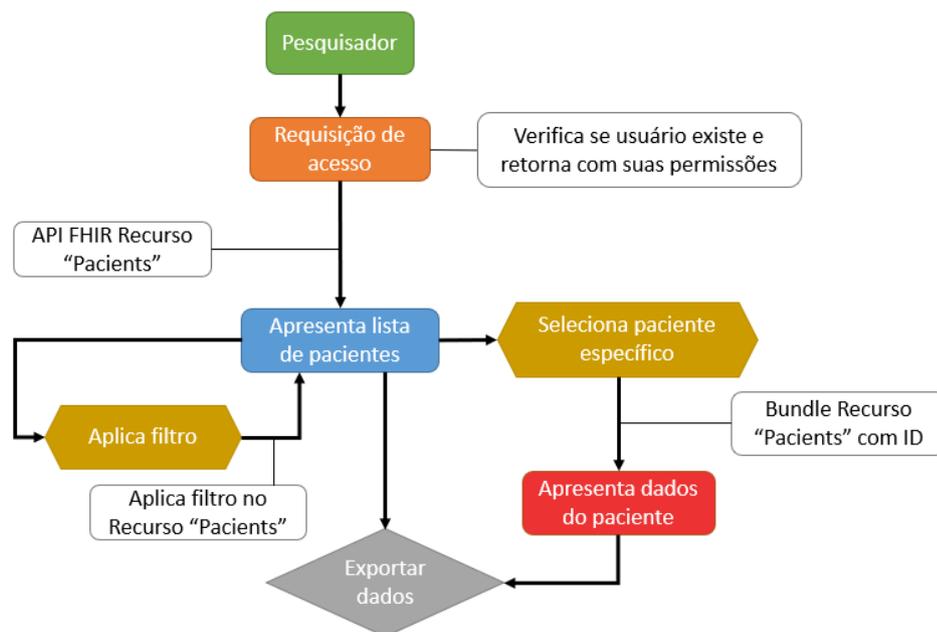
Abaixo serão descritos o funcionamento, estruturação de cada perfil criado para a interface SDC-X, no APÊNDICE C – Diagrama de Classes da Plataforma Web é possível ver o diagrama de classes da plataforma *web*. Por padrão de projeto, todas as respostas em JSON do servidor para as requisições dos *models* são transformadas em *arrays*, e então retornadas para o *controller* que realizou a solicitação. Para complementar são apresentados os diagramas UML das interações dos usuários em cada perfil.

10.2.3.1 Perfil Pesquisador

O perfil Pesquisador foi criado com a finalidade de exportar dados para que o usuário possa analisá-los como desejar. Desta maneira, a interface conta com a funcionalidade de listar pacientes, e a página pode ser acessada no menu esquerdo.

O fluxograma do usuário Pesquisador é apresentado na Figura 21, onde é possível observar as ações realizadas pelo usuário e as requisições solicitadas aos servidores. No APÊNDICE D – Diagrama UML para o perfil Pesquisador é possível ver o diagrama UML para o perfil Pesquisador, onde são apresentados os sistemas acionados e respostas geradas por este perfil.

Figura 21 – Fluxograma dos recursos utilizados para atender o usuário Pesquisador.



Fonte: Elaborado pelo autor, 2022.

A Figura 22 apresenta a tela da lista de pacientes, onde é possível realizar um filtro para selecionar os pacientes que deseja observar. Como este perfil não tem autorização para identificar os pacientes, todos os campos relevantes foram anonimizados, e a identificação é referenciada pelo “*id*” no servidor HAPI FHIR.

O acesso a página é realizado através do endereço “*painel/lista-de-pacientes*” que executa o método “*listaDePacientes*” da classe *controller* “*Painel*”. Neste método é realizada uma comunicação para o *model* “*ModelHL7*” pedindo para retornar os pacientes. Este executa então uma requisição GET para o recurso “*Patient*” ao servidor HAPI FHIR.

Figura 22 - Tela da listagem de pacientes para o perfil Pesquisador.

The screenshot shows a web application interface for a researcher profile. On the left is a dark blue sidebar with the logo 'SDC - X' and navigation links: 'Sistema SDC-X', 'Lista de Pacientes', 'Informações', 'Diabetes', and 'Publicações'. The main content area has an orange header with 'Usuário' and a right arrow. Below the header is a 'Filtros' section with three dropdown menus labeled 'Região', 'Gênero', and 'Faixa Etária', each with the placeholder text 'Escolha uma opção'. A 'GERAR PACIENTES' button is positioned below the filters. The main area contains a table with two tabs: 'LISTA' (active) and 'OUTROS'. The table has three columns: 'Paciente (ID)', 'Gênero', and 'Nascimento'. The data rows are as follows:

| Paciente (ID) | Gênero | Nascimento |
|---------------|--------|------------|
| 13535 | female | 1966-08-13 |
| 24171 | female | 1945-06-13 |
| 30116 | male | 1917-08-06 |
| 32997 | male | 1924-06-30 |

Fonte: Elaborado pelo autor, 2022.

Por padrão todos os pacientes são listados quando esta página é acessada, e o servidor HAPI FHIR limita o número de resultados em 20 pacientes. Assim adicionou-se um botão de “Próximo” ao final da página para avançar para os próximos 20 pacientes. Esta função de próximo é possível pois a resposta do servidor retorna um *link* de “*next*” e “*previous*” para avançar ou voltar nos dados visualizados. Abaixo desta função é adicionado um botão de exportar dados, onde são selecionados todos os dados dos pacientes listados e criados arquivos JSON.

Nesta tela existe um bloco de filtros, para pesquisar uma lista de pacientes com alguma característica em comum, como região, gênero e idade. Este filtro utiliza a função de POST em *JavaScript* para acessar o *Controller* “*Api*” da aplicação *web*. Este *Controller* serve para realizar comunicação para acessos via *JavaScript*, pois optou-se por não recarregar a página em que o usuário está, e apenas atualizar os dados conforme a busca realizada pelo filtro. O *controller* realiza então uma chamada para o “*ModelHL7*”, e este realiza um *bundle* para o recurso “*Patient*”, conforme os dados de filtro.

A lista de pacientes apresentada é clicável, podendo selecionar um dos pacientes para visualizar seus dados. Esta ação pega o “*id*” do usuário e atribui no endereço para outro método do *Controller* “*Painel*” resultando em “*painel/paciente/id*”. Esta ação do *controller* realiza uma chamada para o “*ModelHL7*”, que faz a requisição GET ao servidor HAPI FHIR utilizando o recurso “*Patient*” mais o “*id*”.

Através do “*id*” do paciente são realizadas requisições para adquirir informações de dados de exames, listar exames realizados e medicamentos cadastrados como uso pelo paciente. Esses dados são entregues pelo “*ModelHL7*” que realiza as requisições de “*Observation*”, “*DiagnosticReport*” e “*MedicationAdministration*”. Através de um botão é possível exportar os dados do paciente selecionado.

Através do recurso “*Observation*” é possível apresentar um histórico de valores de exames específicos do paciente. Neste trabalho criou-se o histórico para exames de glicose, peso, pressão arterial, colesterol, HDL e LDL. Essas pesquisas de “*Observation*” seguem o padrão LOINC (*Logical Observation Identifiers, Names and Codes*) dos tipos de exames adquiridos pelos dados sintéticos. A requisição segue um *bundle* contendo o recurso, identificador do paciente, código LOINC, categoria do recurso, e para adicionar mais dados a resposta do servidor é adicionado uma extensão para alterar a quantidade de dados padrão do servidor HAPI FHIR para 200 respostas. Assim, para receber os valores de exames de glicose é criado o *bundle* “*Observation?patient=123&code=2093-3&category=laboratory&_count=200*”.

As Figura 23, Figura 24 e Figura 25 apresentam os resultados dos gráficos dos valores de exames, lista de exames e lista de medicamentos do paciente.

Figura 23 - Visualização gráfica dos resultados de exames do paciente.



Fonte: Elaborado pelo autor, 2022.

Figura 24 - Lista de exames realizados pelo paciente.

The screenshot shows the patient profile for ID 13535. The patient's details are:

- Gênero:** female
- Idade:** 56
- Endereço:** Marblehead, Massachusetts

Below the profile, the 'EXAMES' tab is active, displaying a list of tests performed:

| Exame | Data |
|--|------------|
| Lipid Panel | 31/08/2019 |
| Complete blood count (hemogram) panel - Blood by Automated count | 18/08/2019 |
| Urinalysis macro (dipstick) panel - Urine | 17/07/2019 |
| Complete blood count (hemogram) panel - Blood by Automated count | 17/06/2019 |
| Urinalysis macro (dipstick) panel - Urine | 20/05/2019 |

Fonte: Elaborado pelo autor, 2022.

Figura 25 - Lista de medicamentos utilizados pelo paciente.

The screenshot displays the SDC-X system interface. On the left is a dark blue sidebar with the SDC-X logo and navigation menu items: Sistema SDC-X, Lista de Pacientes, Informações, Diabetes, and Publicações. The main content area has an orange header with 'Usuário' and a right arrow. Below the header, patient information is shown: ID 13535, Gender: female, Age: 56, and Address: Marblehead, Massachusetts. A tabbed interface below shows 'MEDICAMENTOS' selected, displaying a table of medications.

| Medicamento | Data |
|-----------------------------|------------|
| Cisplatin 50 MG Injection | 18/08/2019 |
| PACLitaxel 100 MG Injection | 17/07/2019 |
| Cisplatin 50 MG Injection | 17/06/2019 |
| PACLitaxel 100 MG Injection | 20/05/2019 |
| PACLitaxel 100 MG Injection | 19/04/2019 |

Fonte: Elaborado pelo autor, 2022.

Na aba de exames, é possível clicar sobre um exame da lista e uma página com os resultados é apresentada (Figura 26). Isso é possível através do método “*exame*” do *Controller* “*Painel*”, que utiliza o “*id*” do exame selecionado e realiza uma solicitação para o “*ModelHL7*”.

Figura 26 - Resultado de exame de Lipídios gerado a partir do “*id*” do exame selecionado.

The screenshot shows the SDC-X system interface with the 'Lipid Panel' exam results. The sidebar and header are the same as in Figure 25. The main content area displays the exam details: Status: FINAL, Issued: 31 August 2019 20:13:10. Below this is a table of results.

| Name | Value | Interpretation | Reference Range | Status |
|--------------------------------------|--------------------------|----------------|-----------------|--------|
| Total Cholesterol | 251.72035026288614 mg/dL | | | FINAL |
| Triglycerides | 317.6510197133579 mg/dL | | | FINAL |
| Low Density Lipoprotein Cholesterol | 154.00828801317496 mg/dL | | | FINAL |
| High Density Lipoprotein Cholesterol | 34.181858307039576 mg/dL | | | FINAL |

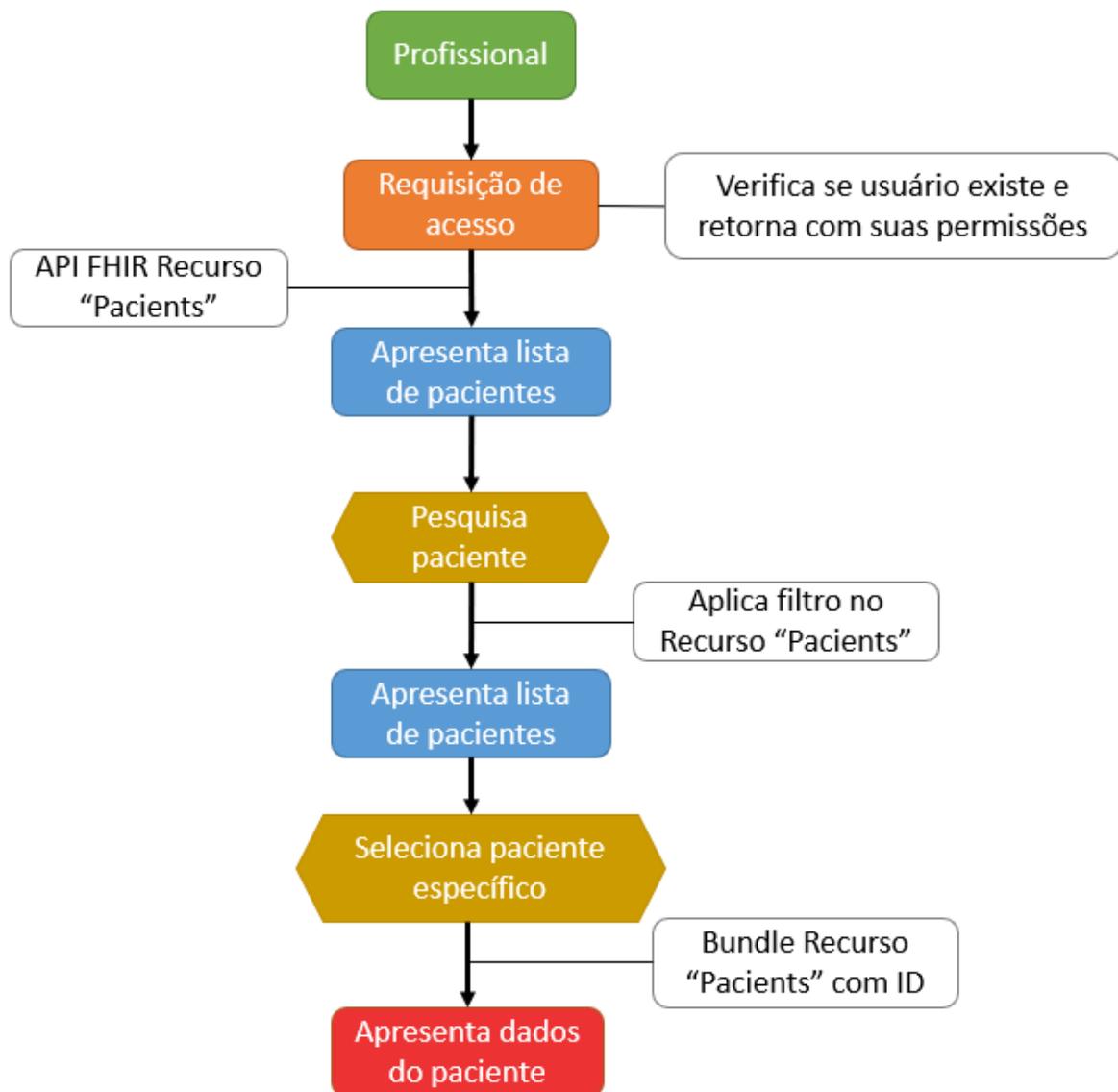
Fonte: Elaborado pelo autor, 2022.

10.2.3.2 Perfil Profissional

Este perfil foi pensado principalmente para o profissional que irá avaliar um paciente. Nele é possível procurar um paciente específico e ver todos os seus dados.

O fluxograma do usuário Profissional é apresentado na Figura 27, onde é possível observar as ações realizadas pelo usuário e as requisições solicitadas aos servidores. No APÊNDICE E – Diagrama UML para o perfil Profissional é possível ver o diagrama UML para o perfil Profissional, onde são apresentados os sistemas acionados e respostas geradas por este perfil.

Figura 27 - Fluxograma dos recursos utilizados para atender o usuário Profissional.



Fonte: Elaborado pelo autor, 2022.

A Figura 28 apresenta a tela da lista de pacientes e o filtro para procurar um paciente específico. Diferente do perfil Pesquisador, o Profissional tem acesso a todos os dados, e assim é possível ver a identificação dos pacientes, aqui listados pelo CPF.

Figura 28 - Página de visualização da lista de pacientes pelo perfil Profissional.

| Paciente (CPF) | Gênero | Nascimento |
|--------------------------------------|--------|------------|
| 2c405456-ee6d-475c-b0fe-e8675589845c | female | 1966-08-13 |
| 801082c3-4f07-46d3-9ab6-dc648099b376 | female | 1945-06-13 |
| b3e1def9-d4cf-4c54-a15a-38e936f06235 | male | 1917-08-06 |
| 4b8ca9cd-6ba9-4175-ab43-a5a07f105afd | male | 1924-06-30 |

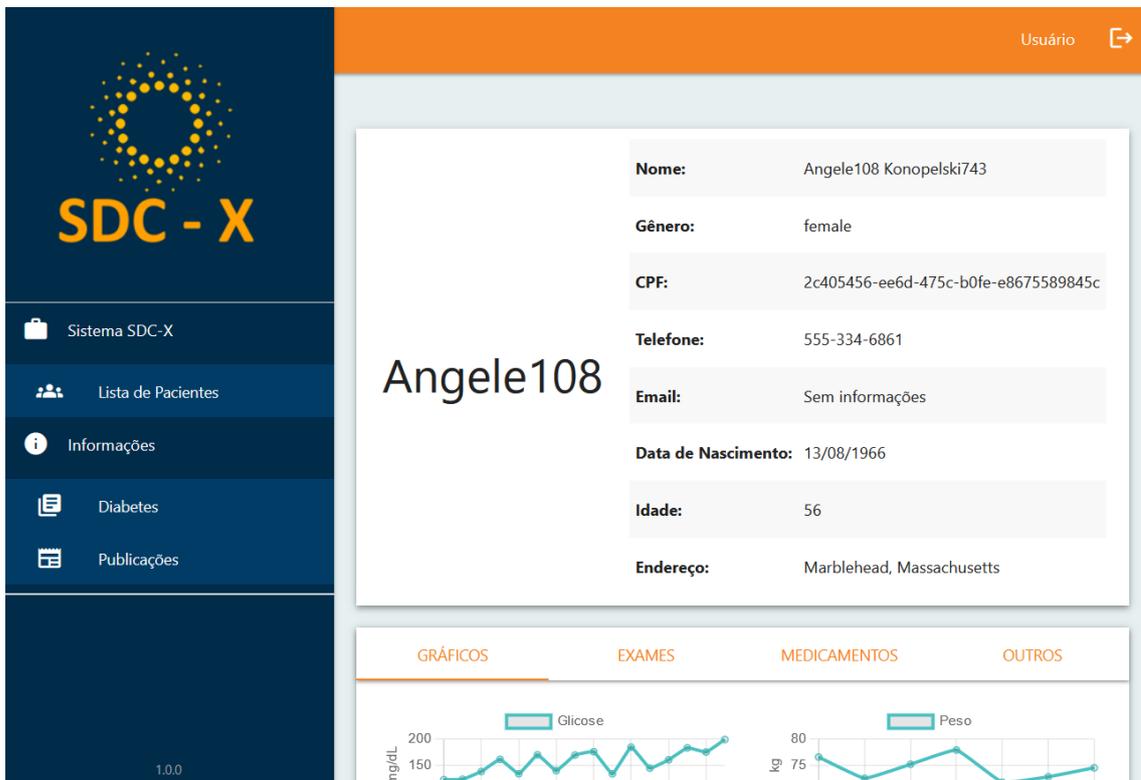
Fonte: Elaborado pelo autor, 2022.

A geração desta *View* é realizada da mesma maneira que o perfil Pesquisador, com a diferença que na *View*, é identificado o perfil do usuário através da *session* do PHP, neste caso um Profissional, e selecionado o código HTML a ser interpretado pelo *browser*.

O filtro apresentado nesta página é realizado através do CPF do paciente desejado. Como não deseja-se recarregar a página, a pesquisa é acionada por uma ação POST via *JavaScript* para o *Controller* “*Api*”, que por sua vez realiza a requisição ao “*ModelHHL7*”, que monta o *bundle* para o recurso “*Patient*” contendo o CPF informado. A resposta deste filtro é interpretada pelo *JavaScript*, que irá ocultar a lista de pacientes e apresentar apenas o paciente retornado pelo servidor HAPI FHIR, caso não seja possível uma mensagem de “Paciente não encontrado” é apresentada.

É possível selecionar um paciente clicando sobre ele na lista de pacientes. Todas as interações de *Controller* e *Model* funcionam da mesma maneira explicada no perfil Pesquisador, e novamente a *View* verifica o perfil do usuário. Na página de Paciente é necessário realizar esta verificação pois o profissional irá visualizar todos os dados sensíveis do paciente como nome, sobrenome, identificador (e.g. CPF), telefone, e-mail, data de nascimento e endereço completo, como mostra a Figura 29.

Figura 29 - Página de visualização de dados de paciente através do perfil Profissional.



Fonte: Elaborado pelo autor, 2022.

O comportamento das classes para gerar a visualização das informações de gráficos dos exames, lista de exames, medicamentos e resultados de exames é a mesma descrita no item Perfil Pesquisador.

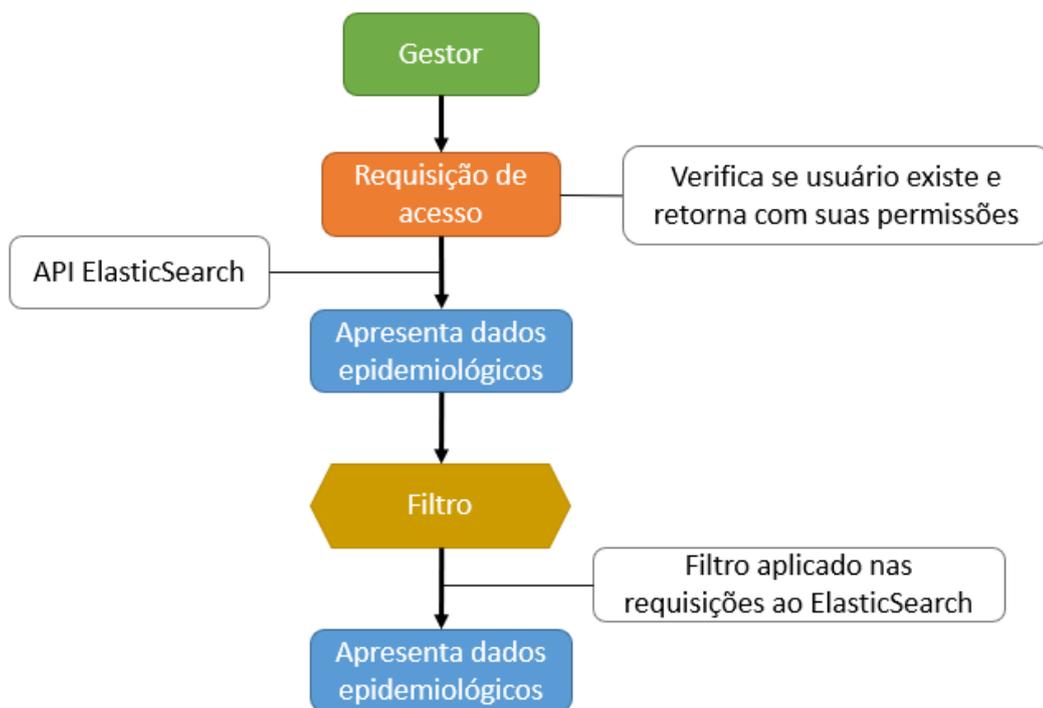
10.2.3.3 Perfil Gestor

No perfil Gestor são apresentadas informações sobre os dados adquiridos pelo projeto SDC-X em geral. As funcionalidades criadas para este perfil foram desenvolvidas para entregar

uma visão epidemiológica sobre a DM, auxiliando na tomada de decisões por gestores, como por exemplo, entidades públicas.

O fluxograma do usuário Profissional é apresentado na Figura 30, onde é possível observar as ações realizadas pelo usuário e as requisições solicitadas aos servidores. No APÊNDICE F – Diagrama UML para o perfil Gestor é possível ver o diagrama UML para o perfil Profissional, onde são apresentados os sistemas acionados e respostas geradas por este perfil.

Figura 30 - Fluxograma dos recursos utilizados para atender o usuário Gestor.



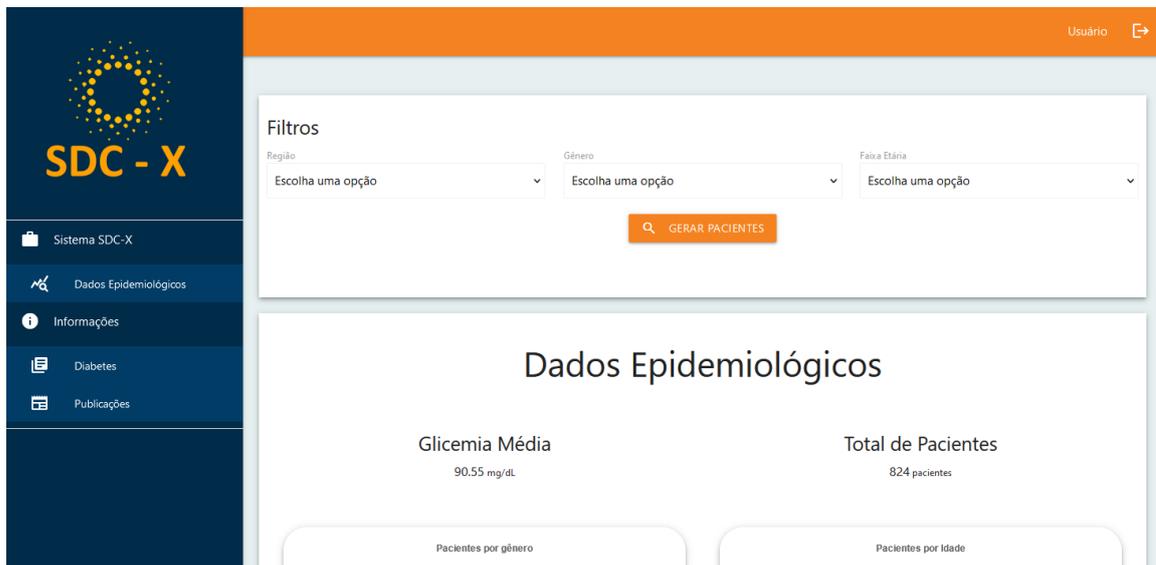
Fonte: Elaborado pelo autor, 2022.

Este perfil não tem permissão de visualização de nenhuma das páginas sobre dados de pacientes. Nele contém apenas o menu de “Dados Epidemiológicos”, onde são apresentados gráficos sobre diversas informações dos dados adquiridos pelo sistema SDC-X.

Este perfil é o principal motivo para a implantação do servidor *ElasticSearch*. Todas as requisições necessárias para atender este perfil são geradas a partir do *ElasticSearch*. A versatilidade de pesquisar por *querys* SQL permite gerar dados e informações rápidas e práticas, aumentando o desempenho de processamento do servidor e desenvolvimento de código.

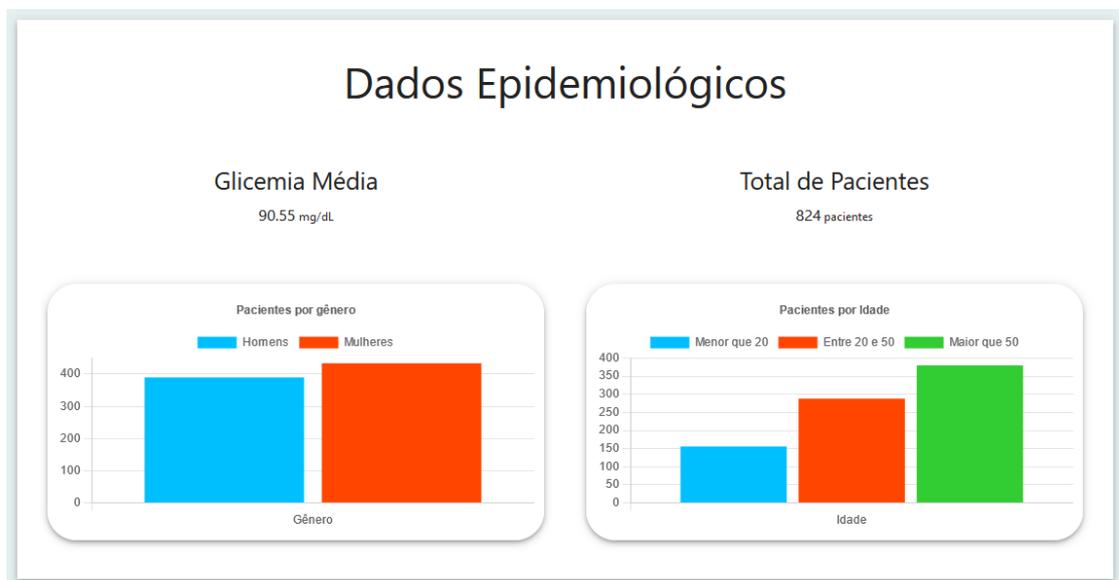
Foram realizados 4 exemplos de *queries* SQL para a visualização da página de Dados Epidemiológicos, a informação de glicemia média de todos os pacientes, a contagem total de pacientes, a divisão por gênero e por idade. A divisão por gênero foi realizada para as categorias dos dados sintéticos como homem e mulher (“male” e “female”), e categorizada a idade entre “Menor que 20 anos”, “Entre 20 e 50 anos” e “Maior que 50 anos”. As Figura 31 e Figura 32 apresentam a página de Dados Epidemiológicos do perfil Gestor.

Figura 31 – Tela de Dados Epidemiológicos acessada pelo perfil Gestor.



Fonte: Elaborado pelo autor, 2022.

Figura 32 – Gráficos e informações visualizados pelo perfil Gestor.



Fonte: Elaborado pelo autor, 2022.

Como é observado na Figura 31, é possível realizar filtros sobre os dados a serem apresentados, o que resulta em consultas e *queries* diferentes ao *ElasticSearch*.

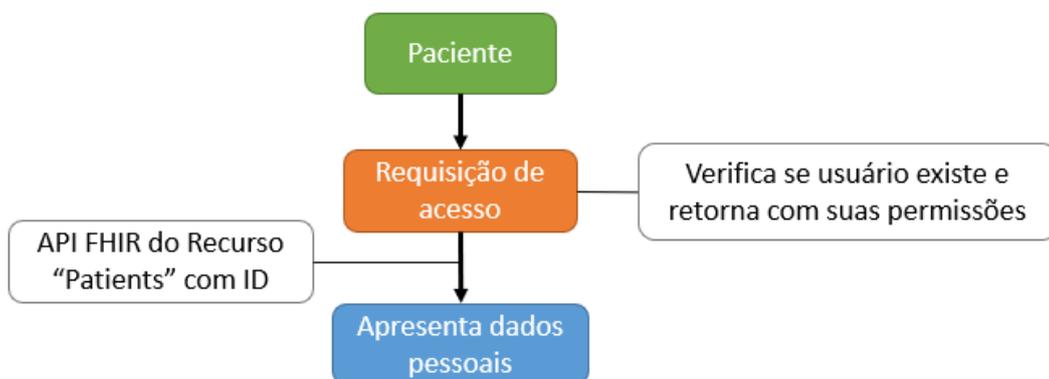
Esta página acessada pelo Gestor é realizada através do método “*dadosEpidemiologicos()*” do *Controller* “*Painel*”, resultando no endereço “*painel/dados-epidemiologicos*”. Por padrão o *Controller* gera a *View* com todos os dados presentes no servidor *ElasticSearch*, que realiza as solicitações dos dados para o “*ModelElastic*”. O *Model* solicitado monta as *queries* SQL (como apresentado no APÊNDICE B – Requisições *ElasticSearch* do projeto) e realiza as requisições para o servidor. A execução do filtro é semelhante aos aplicados pelos perfis Pesquisador e Profissional, com a diferença que o *Model* solicitado para adquirir os dados é o “*ModelElastic*”.

10.2.3.4 Perfil Paciente

O perfil Paciente foi criado especificamente para atender os artigos 6, 9 e 18 da LGPD, que especificam que os dados pessoais devem apresentar transparência, fornecer livre acesso aos titulares dos dados, bem como apresentar a finalidade do uso dos seus dados (BRASIL, 2020a). Através do menu “Informações”, este perfil pode consultar todas as informações das pesquisas realizadas, assim como endereço eletrônico e telefone para dúvidas.

A Figura 33 apresenta o fluxograma de comportamento do usuário e seu UML no APÊNDICE H – Diagrama UML para o perfil Paciente.

Figura 33 - Fluxograma dos recursos utilizados para atender o usuário Paciente.



Fonte: Elaborado pelo autor, 2022.

Este perfil é restrito apenas a visualização das páginas dos seus próprios dados. Esta tela é a mesma apresentada na Figura 28 do perfil Profissional.

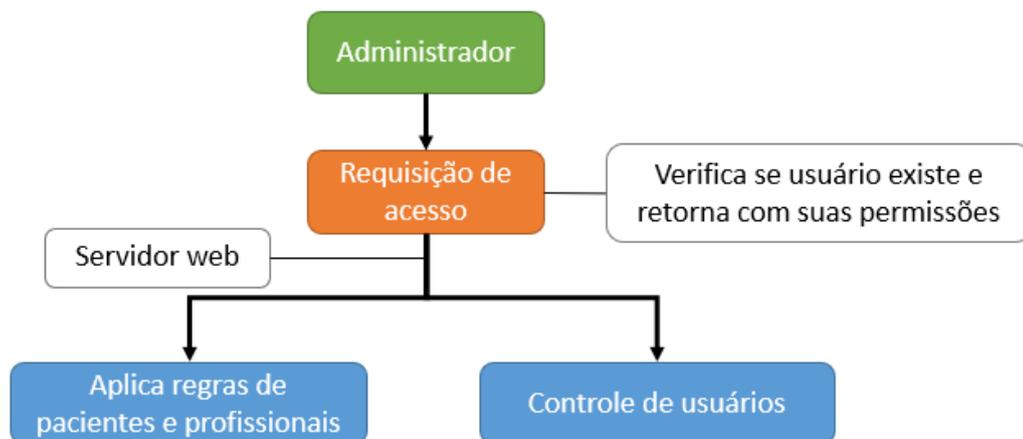
A diferença desta página entre o perfil Paciente e Profissional é na geração. Como não existe uma lista com os “*id’s*” de cada paciente, o *Controller* solicita o CPF do usuário para o “*ModelPainel*” através das informações de acesso. Este *Model* busca a informação requerida na tabela “*sdc_infosuarios*”. Com a informação de CPF, o *Controller* utiliza o “*id*” para o “*ModelHL7*”, que realiza um *bundle* para o servidor HAPI FHIR. Através deste “*id*”, o *Controller* redireciona o usuário para a página “*paciente/id*”, gerando então a tela da Figura 29.

10.2.3.5 Perfil Administrador

O perfil desenvolvido como Administrador apresenta duas telas que podem ser acessadas no menu esquerdo. Essas telas apresentam duas funcionalidades: gerência de usuários e regra para visualização de pacientes.

A Figura 34 apresenta o fluxograma de comportamento do usuário e seu UML no APÊNDICE G – Diagrama UML para o perfil Administrador.

Figura 34 - Fluxograma dos recursos utilizados para atender o usuário Administrador.



Fonte: Elaborado pelo autor, 2022.

A gerência de usuários, acessada em “*painel/listar-usuarios*”, realiza a aceitação de novos usuários a interface, que tenham devidamente preenchido o formulário. Esses dados são salvos na tabela “*sdc_infosuarios*” e são apresentados ao perfil Administrador pelo *Controller* “*Painel*”, que realiza a requisição de informações ao “*ModelPainel*”. Nesta tela são

apresentados os usuários divididos por perfil e por classificação de *status*, que pode ser ativo, bloqueado e pendente. O *status* ativo permite o usuário acesso a interface, bloqueado retira a permissão de acesso, e o pendente significa que um usuário pediu permissão de acesso e falta o Administrador avaliar seu perfil.

Uma tela de regra para visualização dos dados de determinados pacientes, acessada em “*painel/regras-pacientes*”, foi criada para atender o parágrafo 5º do artigo 7º da LGPD, que diz que o controlador dos dados deve ter consentimento do dono dos dados em caso de compartilhamento de informações (BRASIL, 2020a). Como o projeto SDC-X irá trabalhar em parceria com diferentes profissionais, e para evitar que um Termo de Consentimento precise ser aceito ou alterado para o paciente, este pode pedir para que um profissional específico não tenha acesso aos seus dados. Desta forma a ferramenta permite o Administrador selecionar o paciente, selecionar o profissional incluso na tabela “*sdc_infosuarios*” e salvar a relação de bloqueio. Esta informação é salva na tabela “*sdc_regraspacientes*”, e armazena o “*id*” (referente ao armazenado pelo servidor HAPI FHIR) e nome do paciente, “*id*” (referente ao armazenado na tabela “*sdc_infosuarios*”) e nome do profissional.

A Figura 35 apresenta a tela da regra de visualização de pacientes, onde existe o campo para pesquisar o paciente e o profissional para aplicar a regra, na parte inferior as regras já armazenadas em “*sdc_infosuarios*”. Vale ressaltar que se o paciente pedir para que o profissional selecionado possa ver seus dados, basta selecionar a regra e excluir.

Figura 35 - Tela de aplicação de regra para bloqueio de visualização de dados de paciente para um profissional.



Fonte: Elaborado pelo autor, 2022.

10.2.4 Termos de Consentimento e Política de Privacidade

Para atender os princípios da LGPD, neste trabalho é criada uma Política de Privacidade para a plataforma *web*, a qual os usuários devem estar em conformidade para utilizá-la. A Política de Privacidade faz parte da estrutura de documentos para a proteção de dados e preza pela privacidade do usuário. Este documento é endereçado aos usuários que realizam acesso a plataforma e é um documento público (MARTINY, DENKER, 2018; SINGH, BHUSRY, 2017)

Este documento é necessário pois para acessar a plataforma desenvolvida os usuários devem realizar um cadastro, e a aplicação faz uso de *cookies* para sua usabilidade. Assim, na Política de Privacidade desenvolvida é esclarecido sobre a organização responsável pelo tratamento dos dados, finalidade dos dados coletados, prazo de retenção dos dados pessoais, informações de contato. É explicado também sobre o compartilhamento de dados com terceiros e qual a finalidade, sobre o envio de *e-mail marketing*, sobre decisões automatizadas, sobre a proteção dos dados sensíveis, e entre outras descrições que podem ser analisadas na Política de Privacidade desenvolvida.

10.3 ESTUDO DE CASO

Foram desenvolvidas duas telas para representação e visualização de dados gerados pelos Módulos de Aquisição SDC-Cardio e SDC-Hand, e processados pelo Módulo Concentrador. Os dados não estão no padrão HL7 FHIR nem foram incluídos nos servidores HAPI FHIR e *ElasticSearch*. A validação será realizada utilizando os dados localmente acessados por arquivos JSON.

Esses dados são classificados como exames, logo a sua visualização é realizada pelos perfis Pesquisador e Profissional. O acesso é realizado a partir do *Controller* “*Painel*” e selecionando através da lista de exames. Como os dados estão em arquivo local, o *Model* realiza a chamada desses dados JSON, converte em *array* e retorna para o *Controller*.

O Módulo SDC-Cardio realiza uma análise da Variabilidade da Frequência Cardíaca (VFC) para a avaliação do estágio da DM. A partir da coleta de batimentos cardíacos, são gerados dados no domínio da frequência resultando em um gráfico de densidade espectral da potência, sendo a região representada pela cor vermelha a densidade espectral da potência dos sinais de muito baixas frequências, região verde baixas frequências, e região amarela altas

frequências (TERRA, 2020). Assim, foi criado o gráfico da Figura 36, assim como as tabelas com os resultados dos parâmetros de cada região de frequência.

Figura 36 - Visualização da tela de exame para dados reais obtidos pelo SDC-Cardio.



Fonte: Elaborado pelo autor, 2022.

O Módulo SDC-Esc realiza um teste de impedância entre as mãos do paciente, e através de métricas realiza diagnóstico se o paciente é pré-diabético ou não, ou qual o grau da DM. A partir dos dados coletados, a tela da Figura 37 foi desenvolvida (KANAMURA, 2022).

Figura 37 - Visualização da tela de exame para dados reais obtidos pelo SDC-Esc.



Fonte: Elaborado pelo autor, 2022.

10.4 AVALIAÇÃO DE TECNOLOGIA EM SAÚDE 4.0

Como já descrito no item 2.2, uma aplicação de Saúde 4.0 é baseada em 6 princípios, que serão avaliados para verificar se o projeto SDC-X se enquadra como uma aplicação em Saúde 4.0.

O primeiro princípio aborda a interoperabilidade, capacidade de trocar informações entre sistemas de saúde, e ele é cumprido ao projeto utilizar o padrão HL7 FHIR para comunicação dos dados entre suas plataformas. O uso do HL7 FHIR permite a integração com outros sistemas de saúde, como a RNDS.

Através dos Módulos de Aquisição tem-se a capacidade de virtualização da informação. Dados analógicos que são convertidos em dados digitais, permitindo o seu processamento, análise e visualização através do Módulo de Visualização e Interface.

A descentralização da informação é possível através do Módulo de Visualização e Interface, que permite que diferentes usuários tenham acesso aos dados.

O quarto princípio aborda a capacidade em tempo real, que é possível no estágio da coleta de dados do paciente pelos Módulos de Aquisição e Concentrador, e é no módulo concentrador que diferentes métricas são realizadas e os dados são amostrados inicialmente.

O quinto princípio fala sobre a criação de *softwares* de serviço para interagir com aparelhos e sistemas médicos. Este item é complexo pois a interação entre sistema deve ser permitida por ambos os sistemas. O projeto SDC-X trabalha com os servidores HAPI FHIR e *ElasticSearch*, que através das suas RESTful API's é possível acessar as informações por outros sistemas. Assim como o Módulo de Análise de Interface é uma aplicação *web*, permite que tenha a comunicação com outros sistemas *web*.

O último princípio define a habilidade de modularidade. O projeto SDC-X apresenta a modularidade no próprio nome, onde o "X" representa os diferentes Módulos de Aquisição. O projeto permite a inclusão de novos Módulos de Aquisição e os servidores apresentam modularidade para realizar a padronização desses dados. O Módulo de Visualização e Interface é desenvolvido com o padrão de *design* MVC, que tem como características a manutenção para inclusão de novas funcionalidades e reutilização de código.

11 DISCUSSÃO

Este capítulo apresenta uma discussão sobre o padrão HL7 FHIR como protocolo de interoperabilidade de dados em saúde, a implantação do servidor HAPI FHIR, implantação e validação de um servidor *ElasticSearch* como auxiliar ao servidor HAPI FHIR, o desenvolvimento da plataforma *web* e suas funcionalidades para o contexto do projeto SDC-X utilizando o padrão de *design* MVC, avaliação com dados sintéticos e dados reais e uma análise do projeto SDC-X como uma tecnologia em Saúde 4.0.

11.1 PADRÃO HL7 FHIR

A organização HL7 apresenta um contexto no desenvolvimento de padrões para a interoperabilidade de dados em saúde muito estruturado, com versões anteriores como o HL7 V2 e V3, e atualmente com o desenvolvimento do FHIR. O padrão FHIR apresenta tecnologias atuais utilizadas pelo mercado, e como apresentado por Benson e Grieve (2016) e Lehne *et al.* (2019) vem ganhando visibilidade de sua utilização como forte solução para o desafio da interoperabilidade entre sistemas informatizados em saúde.

O uso do padrão FHIR nas aplicações em saúde se mostra muito versátil, podendo ser aplicado em diferentes contextos tecnológicos. Em El-Sappagh *et al.* (2019) descrevem a aplicação de um sistema *mobile* para monitoramento de pacientes utilizando o padrão FHIR para comunicação de dados entre os sistemas envolvidos. Aplicações *web* conseguem realizar alterações para incluir o padrão FHIR como o OpenMRS (plataforma *open source* para registros médicos) como desenvolvido no trabalho de Kasthurirathne *et al.* (2015). Outros contextos envolvem registros médicos eletrônicos, aplicações *mobile*, *wearable devices*, medicina de precisão, suporte em decisões clínicas e análise em *big data* (LEHNE *et al.*, 2019).

Sua aceitação pela comunidade se deve a facilidade de compreensão e implementação. O HL7 disponibiliza gratuitamente uma vasta documentação sobre o padrão FHIR, apresentando guias de implementação e exemplos. A comunidade FHIR é bastante ativa, desenvolvendo aplicações *open source* e realizando validações do uso FHIR em diferentes contextos tecnológicos em saúde.

Sua relevância é observada por grandes corporações de tecnologia, que estão incluindo o padrão FHIR como parte dos seus catálogos de produtos, como é o caso da Google, Amazon e Microsoft. Elas oferecem plataformas de hospedagem de dados em nuvem utilizando

servidores que comportam o FHIR, ferramentas de anonimização de dados FHIR e diversas outras aplicações (VIEIRA, 2021).

11.2 SERVIDORES HAPI FHIR E *ELASTICSEARCH*

O servidor HAPI FHIR já validado por Vieira (2021), apresentou uma performance adequada ao que foi proposto. O servidor HAPI FHIR, sendo uma aplicação *open source*, oferece uma boa resposta para implementação rápida de um servidor FHIR, contendo todos os recursos do padrão. Através do seu endereço eletrônico, a HAPI-FHIR permite realizar diferentes testes da sua aplicação, executando seu servidor *online* ou exportando localmente (HAPIFHIR, 2022a), como é o caso deste trabalho.

O trabalho utilizou também um servidor *ElasticSearch*. Através da divisão de recursos FHIR em indexações no *ElasticSearch*, o servidor permitiu reduzir o número de requisições realizadas ao servidor HAPI FHIR, evitando um *overload* do servidor e melhorando o desempenho do tempo de resposta para a visualização da plataforma *web* para o usuário.

As implantações das aplicações dos servidores foram realizadas através de *deploys* com a tecnologia *container* no *Docker Engine*, e ambos os servidores responderam as todas as requisições apresentadas neste trabalho conforme esperado.

11.3 ARQUITETURA MVC

Padrão utilizado para a criação de *web sites*, a arquitetura MVC é utilizada para evitar as dificuldades da implementação devido a separação entre as camadas de interface e camadas de aplicação lógica. Esta arquitetura é utilizada em diversos *frameworks* de linguagens como JAVA, PHP e .NET devido as suas vantagens, como reutilização de código e facilidade de manutenção.

A comunicação entre as camadas da arquitetura MVC podem variar conforme estratégia adotada pelos projetistas, desde que obedeça ao princípio de cada camada. Optou-se por desenvolver a plataforma *web* realizando a comunicação entre *Controller-Model* e *Controller-View*, ou seja, não existe comunicação entre *Model-View*, assim o fluxograma de implementação é melhor desenvolvido ao ser iniciado sempre pelo *Controller*. Observou-se que esta abordagem permite realizar um desenvolvimento mais fluído, e identificação de *bugs* no código sempre partindo do *Controller*.

O padrão de codificação foi a criação de um arquivo PHP para classe, e cada Controller e Model são formados por classes, com seus devidos objetos e métodos.

A interações de usuário com a interface que demandam a utilização de dados dos servidores, como os casos de aplicação de filtros, utilizou-se comandos POST via *JavaScript* para um único *Controller* para realizar esta tarefa. Assim o *Controller* realiza o comando para o *Model*, e o *Controller* devolve a resposta para a *View*, mantendo o padrão de comunicação das camadas do projeto.

11.4 PLATAFORMA *WEB*

A estruturação MVC apresentou uma arquitetura sólida para o desenvolvimento da plataforma *web*, tendo o roteamento da aplicação apontado para as classes do Controller.

A implementação da sequência de endereçamento permitiu desenvolver páginas que necessitavam de dados específicos, como perfil de paciente e exames, a partir do “id” do recurso FHIR a ser buscado no servidor.

A plataforma *web* desenvolvida complementa o Módulo de Visualização e Interface do projeto SDC-X, e apresenta de forma clara e direta os dados e informações contidos nos servidores. Esses dados que são coletados a partir do Módulo de Aquisição e processados pelo Módulo Concentrador. A interface é uma plataforma multiusuário e atende os requisitos de comunicação com as RESTful API's dos servidores, e entrega assim um local descentralizado dos dados coletados pelo projeto SDC-X.

A manipulação de dados sensíveis e permissão de visualização dos dados a partir de usuários, tornou o desenvolvimento criterioso da interface para atender a LGPD. Assim, para que cada usuário tenha uma conta de acesso, é preciso aceitar o Termo de Consentimento e Política de Privacidade desenvolvidos para o projeto. Também foram desenvolvidas lógicas para impedir o acesso a dados caso o usuário não tenha permissão.

A interface foi desenvolvida buscando atender 5 perfis de usuários, onde dois deles foram acrescentados ao projeto para atender exigências da LGPD.

O perfil Administrador visa aplicar regras a visualização de dados dos pacientes para o perfil Profissional, pois a LGPD apresenta margem de interpretação para que o portador dos dados possa escolher deixar de apresentar seus dados para determinados indivíduos. Um exemplo deste caso é um paciente não querer compartilhar seus dados com um profissional

médico que seja seu familiar. Outra funcionalidade deste perfil é o controle de quem pode acessar a plataforma *web*.

Criado apenas para atender aos requisitos da LGPD, o perfil Paciente permite que o usuário possa ver suas informações coletada pelo projeto SDC-X, sendo dados de identificação e dados fisiológicos. A interface também apresenta um local de acesso de contato ao IEB-UFSC para dúvidas.

Pensado para a área de pesquisa, o perfil Pesquisador tem acesso aos dados coletados pelo projeto SDC-X. Esses dados são anonimizados e é permitido a exportação dos mesmos pelo usuário. Assim a interface desenvolvida entrega ao Módulo de Análise e Interface a propriedade de compartilhamento de dados em saúde na área de pesquisa, atendendo níveis de padronização e veracidade dos dados, que, como descrito por Leroux, Metke-Jimenez e Lawley (2017), os dados em saúde precisam ser compartilhados e comparados entre si, apoiando o conhecimento baseado em evidência e reduzindo o custo da realização de estudos.

O perfil Profissional é desenvolvido principalmente para o Médico profissional que irá avaliar um paciente. A interface apresenta gráficos de dados históricos de exames, entregando de maneira rápida e prática as informações do paciente. Os exames também podem ser consultados e avaliados individualmente. De acordo com a resolução nº 1.638/2002 do Conselho Federal de Medicina (CFM-Brasil, 2002) a plataforma *web* não pode ser considerada um prontuário por não atender ao requisito de permitir o acompanhamento de um tratamento de forma direta entre diferentes profissionais. Esta funcionalidade pode ser atendida acrescentando um campo para um registro do profissional dentro da interface e acrescentando um *Profile* no servidor HAPI FHIR utilizado pelo projeto.

Buscando atender entidades públicas para auxiliar nas tomadas de decisões baseadas em evidências, e devido a tendência de crescimentos de indivíduos com DM, o perfil Gestor foi desenvolvido para entregar uma visualização do conjunto de dados do projeto SDC-X. A partir deste perfil, diversas análises de dados epidemiológicos podem ser desenvolvidas para entregar informações como projeção do DM, categorizar seu impacto por cidade, gênero, idade e diversas outras aplicações que podem ser adicionadas utilizando o método MVC.

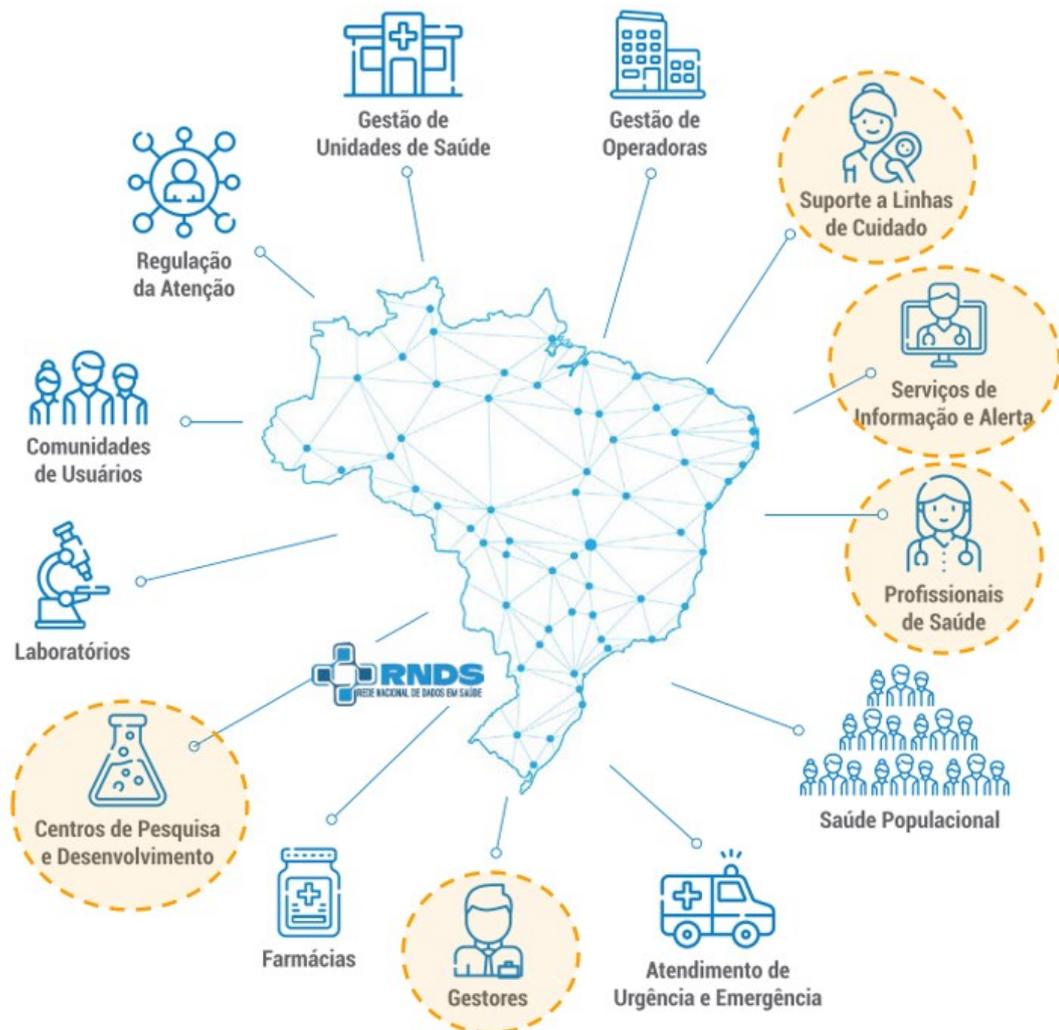
11.5 POTENCIAL DE INTEGRAÇÃO DO PROJETO SDC-X

O projeto SDC-X, através das ferramentas desenvolvidas e utilização do padrão FHIR para interoperabilidade, permite que tenha possibilidade de integração do sistema público de

saúde brasileiro. Este promete o surgimento de serviços de pesquisa para os benefícios da população, e da participação das universidades nos eixos de fomento à inovação (DEPARTAMENTO DE INFORMÁTICA DO SUS, 2020).

Uma vez que o a RNDS também utiliza o padrão FHIR, os dados do servidor do projeto SDC-X podem ser compartilhados, e a plataforma *web* pode realizar integrações com os dados da RNDS para estender a usabilidade e visualização dos dados entre os sistemas. O projeto tem capacidade de atingir diversas áreas propostas pela RNDS para conexão e integração como mostra a Figura 38.

Figura 38. Projeto SDC-X tem capacidade de integrar algumas das frentes propostas pela RNDS como Suporte a Linhas de Cuidado, Serviços de Informação e Alerta, Profissionais de Saúde, Gestores e Centros de Pesquisa e Desenvolvimento.



Fonte: Adaptado de BRASIL (2022a).

11.6 VALIDAÇÃO COM DADOS SIMULADOS E REAIS

A utilização de dados sintéticos apresentou uma ótima ferramenta para validar a comunicação entre a plataforma *web* e os servidores, uma vez que esses dados geraram registros de saúde em larga escala e sem riscos de quebra de sigilo, reduzindo barreiras de desenvolvimento nos estágios iniciais de projeto de pesquisa.

Para validar a modularidade da plataforma *web* desenvolvida com a arquitetura MVC, foram utilizados dados coletados de outros trabalhos, referentes aos Módulos de Aquisição. A flexibilidade do MVC permitiu desenvolver novas páginas *web* para representar os dados em questão, mesmo os que não seguiram o padrão FHIR.

11.7 PROJETO SDC-X COMO TECNOLOGIA EM SAÚDE 4.0

O projeto SDC-X é composto por diversas tecnologias aplicadas em saúde, que vão desde a coleta de dados, processamento, armazenamento e apresentação gráfica. Ele apresenta uma estrutura que conecta pacientes, aparelhos médicos, organizações de pesquisa, hospitais e clínicas. Tecnologias essas, conforme descrito no item 2.2 e discutido anteriormente, compõem os objetivos da Saúde 4.0.

Através dos 6 princípios da Saúde 4.0 apresentados 2.1 Indústria 4.0 em Saúde, é observado o cumprimento de todos os itens pelo projeto SDC-X, e que cada módulo do projeto é responsável pelo cumprimento desses princípios. Assim, os módulos observados individualmente, podem representar aplicações e até soluções na área da saúde, mas não se caracterizam como aplicação em Saúde 4.0 individualmente.

A análise dos princípios da Saúde 4.0 permitiu verificar a característica de modularidade que o projeto SDC-X apresenta. Ele permite integrar novos Módulos de Aquisição, que seguindo o padrão FHIR a partir do Módulo Concentrador, podendo armazenar esses dados nos servidores presentes no Módulo de Armazenamento, e com a arquitetura MVC da plataforma *web* do Módulo de Visualização e Interface é possível acrescentar a visualização dos novos dados.

12 CONCLUSÃO

Este trabalho apresentou a implantação e usabilidade de dois servidores, HAPI FHIR e *ElasticSearch*, buscando atender as conformidades do padrão HL7 FHIR de interoperabilidade de dados relacionados a saúde. Foi apresentado também o desenvolvimento de uma aplicação *web* utilizando os recursos FHIR para apresentação visual das informações dos servidores. A proposta complementa trabalhos anteriores realizados no IEB-UFSC com o intuito de possibilitar o compartilhamento de informações e dados clínicos em pesquisa em um formato descentralizado.

O padrão FHIR se apresenta como uma solução robusta para os desafios de interoperabilidade entre sistemas da área da saúde. Sua grande aceitação pela comunidade de desenvolvedores da área da saúde proporciona o desenvolvimento e compartilhamento de tecnologias *open source*, como o servidor HAPI FHIR utilizado no projeto SDC-X. Este servidor atendeu a todos requisitos do trabalho de maneira bastante satisfatória, que através da sua RESTful API, permite facilmente a integração de sistemas *web*.

Apesar das facilidades do servidor HAPI FHIR, a interface desenvolvida apresenta demandas que podem gerar uma sobrecarga no servidor, ou um tempo de resposta de lógica resultando em um baixo desempenho em termos de velocidade de carregamento de páginas *web*. Este desafio é solucionado utilizando um servidor *ElasticSearch* que serve como auxiliar ao servidor HAPI FHIR. Este servidor, que também apresenta uma RESTful API, permitiu realizar requisições elaboradas utilizando pesquisa em SQL, reduzindo assim tempo de resposta do servidor e tempo execução da lógica de programação.

A plataforma *web* desenvolvida representa o Módulo de Visualização e Interface do projeto SDC-X. Utilizando a arquitetura MVC, seu desenvolvimento apresentou fluidez na implementação dos códigos estruturando-os em camadas, além da facilidade em reutilizar e dar manutenção em código. Essa arquitetura apresenta a característica importante de modularidade do trabalho desenvolvido, pois o projeto SDC-X irá incluir novos módulos e novas análises de diferentes tipos de dados, e ela permite facilmente realizar alterações e incluir novas funcionalidades.

Essa plataforma *web* desenvolvida apresenta 5 tipos de perfis, cada um com suas modalidades de visualização dos dados, entregando de forma direta as informações ao usuário. Os perfis apresentam características e usabilidades diferentes dentro da interface, apresentando características que servem para obtenção de dados para pesquisas e também para o

compartilhamento de dados clínicos em diferentes níveis dentro e fora da comunidade científica.

Uma validação da interface é realizada para certificá-la como um módulo integrante ao projeto SDC-X utilizando dados sintéticos estruturado em FHIR, e dados reais adquiridos por módulos do projeto, apresentando sua modularidade e integração de novas funcionalidades.

Por fim, uma análise da integração da interface ao projeto SDC-X é realizada para categorizar o projeto como uma solução de tecnologia em Saúde 4.0, apresentando características presentes na indústria que marcam novas tendências tecnológicas. Além disso, apresenta soluções importantes para o contexto do DM, que necessita de tecnologias para acompanhamento e monitoramento, tanto para o indivíduo com DM quanto para entidades públicas em suas tomadas de decisão e planejamento. Neste contexto o projeto apresenta grande possibilidade de integração à RNDS, contribuindo para a agregação e compartilhamento de dados em saúde no Brasil.

12.1 TRABALHOS FUTUROS

O desenvolvimento deste trabalho permitiu observar o projeto SDC-X por completo, e algumas atividades são necessárias para que este esteja operável. Algumas considerações para o projeto e para a plataforma *web* são levadas em consideração, são elas:

- Adicionar novos Módulos de Aquisição e análise de sinais e informações para o SDC-X, tornando mais consistentes as pesquisas e desenvolvimentos relacionados ao DM para o diagnóstico, avaliação e controle, detecção e predição de complicações;
- Implementar e testar os servidores em nuvem;
- Desenvolver e aplicar soluções de segurança em todos os módulos do projeto visando atender os requisitos da LGPD;
- Integrar o Módulo Concentrador ao Módulo de Armazenamento;
- Criar *Profiles* para o servidor HAPI FHIR de todos os dados do projeto SDC-X;
- Adicionar novas funcionalidades para atender a resolução nº 1.638/2002 do Conselho Federal de Medicina para tornar a plataforma *web* um Prontuário Eletrônico;
- Adicionar novas informações ao perfil Gestor como predições e análises sobre o impacto da DM e suas complicações na população;

- Avaliar e realizar um estudo piloto visando a possibilidade de integração do projeto SDC-X ao sistema da RNDS implantada no Brasil.

REFERÊNCIAS

ADAM, Stenly Ibrahim; ANDOLO, Stevani. A New PHP Web Application Development Framework Based on MVC Architectural Pattern and Ajax Technology. *In: 2019 1ST INTERNATIONAL CONFERENCE ON CYBERNETICS AND INTELLIGENT SYSTEM, ICORIS 2019 2019, Anais [...].* : Institute of Electrical and Electronics Engineers Inc., 2019. p. 45–50. DOI: 10.1109/ICORIS.2019.8874912.

AL-JAROODI, Jameela; MOHAMED, Nader; ABUKHOUSA, Eman. Healthcare 4.0. **IEEE Engineering Management Review**, [S. l.], v. 47, n. 3, p. 24–28, 2019. DOI: 10.1109/EMR.2019.2930702. Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8782570>. Acesso em: 3 fev. 2021.

BENDER, Duane; SARTIPI, Kamran. HL7 FHIR: An agile and RESTful approach to healthcare information exchange. *In: PROCEEDINGS OF CBMS 2013 - 26TH IEEE INTERNATIONAL SYMPOSIUM ON COMPUTER-BASED MEDICAL SYSTEMS 2013, Anais [...].* [s.l: s.n.] p. 326–331. DOI: 10.1109/CBMS.2013.6627810.

BENSON, Tim; GRIEVE, Grahame. Implementing FHIR. **Principles of Health Interoperability**, [S. l.], p. 397–416, 2016. a. DOI: 10.1007/978-3-319-30370-3_22.

BRASIL. Ministério da Saúde. Cadernos de Atenção Básica, n. 16. Departamento de Atenção Básica – Brasília: Ministério da Saúde, 2006.

BRASIL. Ministério da Saúde. Departamento de Informática do SUS. Estratégia de Saúde Digital para o Brasil 2020-2028. Brasília, DF: MS, 2020. Disponível em: <http://bvsmms.saude.gov.br/bvs/publicacoes/estrategia_saude_digital_Brasil.pdf>. Acesso em: 16 de Julho de 2020a.

BRASIL. Ministério da Saúde. Departamento de Informática do SUS. Plano de Ação, Monitoramento e Avaliação da Estratégia de Saúde Digital para o Brasil 2019-2023. Brasília, DF: MS, 2020. Disponível em: <<https://saudedigital.saude.gov.br/wp-content/uploads/2020/04/PAMA-Saude-digital.pdf>>. Acesso em: 06 de Julho de 2020b.

CHARTJS. Chart.js. Disponível em: < <https://www.chartjs.org/>>. Acessado em: 10 de Julho de 2022.

CHUTE, Chaloner; FRENCH, Tara. Introducing Care 4.0: An Integrated Care Paradigm Built on Industry 4.0 Capabilities. **International Journal of Environmental Research and Public Health**, [S. l.], v. 16, n. 12, p. 2247, 2019. DOI: 10.3390/ijerph16122247. Disponível em: <https://www.mdpi.com/1660-4601/16/12/2247>. Acesso em: 8 mar. 2021.

CONSELHO FEDERAL DE MEDICINA (CFM - Brasil). Resolução nº 1.638/2002. Brasília, 2002.

CUI, Wei; HUANG, Lin; LIANG, Li Jing; LI, Jing. The research of PHP development framework based on MVC pattern. *In*: ICCIT 2009 - 4TH INTERNATIONAL CONFERENCE ON COMPUTER SCIENCES AND CONVERGENCE INFORMATION TECHNOLOGY 2009, **Anais [...]**. [s.l: s.n.] p. 947–949. DOI: 10.1109/ICCIT.2009.130.

DRATH, Rainer; HORCH, Alexander. **Industrie 4.0: Hit or hype? [Industry Forum]IEEE Industrial Electronics Magazine**Institute of Electrical and Electronics Engineers Inc., , 2014. DOI: 10.1109/MIE.2014.2312079.

ELASTIC. O coração do Elastic Stack gratuito e aberto. Disponível em: <<https://www.elastic.co/pt/ElasticSearch/>>. Acessado em: 10 de Julho 2022a.

ELASTIC. SQL Limitations. Disponível em: <<https://www.elastic.co/guide/en/ElasticSearch/reference/current/sql-limitations.html>>. Acessado em: 12 de abril de 2022b.

EL-SAPPAGH, Shaker; ALI, Farman; HENDAWI, Abdeltawab; JANG, Jun Hyeog; KWAK, Kyung Sup. A mobile health monitoring-and-treatment system based on integration of the SSN sensor ontology and the HL7 FHIR standard. **BMC Medical Informatics and Decision Making**, [S. l.], v. 19, n. 1, p. 1–36, 2019. DOI: 10.1186/s12911-019-0806-z.

FITCH, J. Patrick. Engineering a Global Response to Infectious Diseases. **Proceedings of the IEEE**, [S. l.], v. 103, n. 2, p. 263–272, 2015. DOI: 10.1109/JPROC.2015.2389146.

FHIR. RESTful API. Disponível em: <<http://hl7.org/fhir/http.html>>. Acessado em: 14 de Julho de 2022.

FREITAS, C. Como Elaborar uma política de privacidade aderente à LGPD? Disponível em: <<https://www.serpro.gov.br/lgpd/noticias/2019/elabora-politica-privacidade-aderente-lgpd-dados-pessoais>>. Acessado em: 17 de Julho de 2022.

GAMMA, Erich; HELM, Richard; JOHNSON, Ralph; VLISSIDES, John. **Padrões de Projetos - Soluções Reutilizáveis de Software Orientado a Objetos**. [s.l.: s.n.].

GREGG, Edward et al. Improving Health Outcomes of People with Diabetes Mellitus: Target Setting to Reduce the Global Burden of Diabetes Mellitus by 2030 Diabetic Association of. **Glennis Andall-Brereton**, [S. l.], v. 8, [s.d.].

HAPIFHIR. Some Ways You Can Use HAPI FHIR. Disponível em: <<https://hapifhir.io>>. Acessado em: 30 de junho de 2022a.

HAPIFHIR. HAPI-FHIR Starter Project. Disponível em: <<https://github.com/hapifhir/hapi-fhir-jpaserver-starter>>. Acessado em: 28 de junho de 2022b.

HE, Jiayu. Research on Personalized Search Based on Elasticsearch. *In*: PROCEEDINGS OF 2020 IEEE 2ND INTERNATIONAL CONFERENCE ON CIVIL AVIATION SAFETY AND INFORMATION TECHNOLOGY, ICCASIT 2020 2020, **Anais** [...]. : Institute of Electrical and Electronics Engineers Inc., 2020. p. 572–575. DOI: 10.1109/ICCASIT50869.2020.9368774.

HIDAYAT, Inten Fauziah; HERMANTO, Beni Rio. A Preliminary Implementation of HL7 FHIR to Achieve Interoperability in Indonesia’s Local EHR. *In*: 2020, **Anais** [...]. :

Institute of Electrical and Electronics Engineers (IEEE), 2020. p. 1–5. DOI: 10.1109/ict49546.2020.9239534.

HONG, Na; WANG, Kui; YAO, Lixia; JIANG, Guoqian. Visual FHIR: An Interactive Browser to Navigate HL7 FHIR Specification. *In: PROCEEDINGS - 2017 IEEE INTERNATIONAL CONFERENCE ON HEALTHCARE INFORMATICS, ICHI 2017 2017, Anais [...].* : Institute of Electrical and Electronics Engineers Inc., 2017. p. 26–30. DOI: 10.1109/ICHI.2017.54.

HUSSAIN, Mohannad A.; LANGER, Steve G.; KOHLI, Marc. Learning HL7 FHIR Using the HAPI FHIR Server and Its Use in Medical Imaging with the SIIM Dataset. **Journal of Digital Imaging**, [S. l.], v. 31, n. 3, p. 334–340, 2018. DOI: 10.1007/s10278-018-0090-y.

IDF. ATLAS. **International Diabetes Federation**. [s.l: s.n.]. v. 266 DOI: 10.1016/S0140-6736(55)92135-8.

IEEE. **IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. IEEE Std 610**. 1991. Disponível em: <https://ieeexplore.ieee.org/document/182763>. Acesso em: 29 mar. 2021.

JANKI, Zoltan Richard; SZABO, Zoltan; BILICKI, Vilmos; FIDRICH, Marta. Authorization solution for full stack FHIR HAPI access. *In: IEEE 30TH JUBILEE NEUMANN COLLOQUIUM, NC 2017 2018, Anais [...].* : Institute of Electrical and Electronics Engineers Inc., 2018. p. 121–124. DOI: 10.1109/NC.2017.8263266.

KASTHURIRATHNE, Suranga N.; MAMLIN, Burke; KUMARA, Harsha; GRIEVE, Grahame; BIONDICH, Paul. Enabling Better Interoperability for HealthCare: Lessons in Developing a Standards Based Application Programming Interface for Electronic Medical Record Systems. **Journal of Medical Systems**, [S. l.], v. 39, n. 11, 2015. DOI: 10.1007/s10916-015-0356-6.

KHELASSI, Abdeldjalil; ESTRELA, Vania Vieira; MONTEIRO, Ana Carolina Borges; FRANÇA, Reinaldo Padilha; IANO, Yuzo; RAZMJOOY, Navid. Health 4.0:

Applications, Management, Technologies and Review. **Medical Technologies Journal**, [S. l.], v. 2, n. 4, p. 262–276, 2019. DOI: 10.26415/2572-004x-vol2iss4p262-276.

LASI, Heiner; FETTKE, Peter; KEMPER, Hans Georg; FELD, Thomas; HOFFMANN, Michael. Industry 4.0. **Business and Information Systems Engineering**, [S. l.], v. 6, n. 4, p. 239–242, 2014. DOI: 10.1007/s12599-014-0334-4.

LEHNE, Moritz; LUIJTEN, Sandra; VOM FELDE GENANNT IMBUSCH, Paulina; THUN, Sylvia. The use of FHIR in digital health – A review of the scientific literature. **Studies in Health Technology and Informatics**, [S. l.], v. 267, n. September, p. 52–58, 2019. DOI: 10.3233/SHTI190805.

LEROUX, Hugo; METKE-JIMENEZ, Alejandro; LAWLEY, Michael J. Towards achieving semantic interoperability of clinical study data with FHIR. **Journal of Biomedical Semantics**, [S. l.], v. 8, n. 1, p. 41, 2017. DOI: 10.1186/s13326-017-0148-7. Disponível em: <http://jbiomedsem.biomedcentral.com/articles/10.1186/s13326-017-0148-7>. Acesso em: 30 jun. 2022.

LHOTSKA, Lenka. Application of industry 4.0 concept to health care. *In*: STUDIES IN HEALTH TECHNOLOGY AND INFORMATICS 2020, **Anais [...]**. : IOS Press BV, 2020. p. 23–37. DOI: 10.3233/SHTI200613. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/33087590/>. Acesso em: 22 fev. 2021.

MARTINEZ, Lisa C.; SHERLING, Dawn; HOLLEY, Allison. **The Screening and Prevention of Diabetes Mellitus Primary Care - Clinics in Office Practice** W.B. Saunders, , 2019. DOI: 10.1016/j.pop.2018.10.006.

MARTINY, K., ELENIUS, D., DENKER, G. **Protecting Privacy with a Declarative Policy Framework**. 2018 IEEE 12th International Conference on Semantic Computing (ICSC), 2018, pp. 227-234, doi: 10.1109/ICSC.2018.00039.

MATERIALIZE. About. Disponível em: < <https://materializecss.com/about.html>>. Acessado em: 10 de Julho de 2022.

MAXHELAKU, Suela; KIKA, Alda. Improving Interoperability in Healthcare Using H17 Fhir. *[S. l.]*, n. June, p. 35–42, 2019. DOI: 10.20472/iac.2019.047.012.

MINISTÉRIO DA SAÚDE. Rede Nacional de Dados em Saúde. Disponível em: <. Acessado em: 17 de Julho de 2022a.

MINISTÉRIO DA SAÚDE. Rede Nacional de Dados em Saúde - Legislação. Disponível em: <<https://www.gov.br/saude/pt-br/assuntos/rnds/legislacao/legislacao>>. Acessado em: 17 de Julho de 2022b.

NAKAMURA, Karla Kaori. **Sistema para Avaliação da Função Sudomotora Aplicado na Detecção Precoce do Diabetes Mellitus tipo 2**. 2022. Universidade Federal de Santa Catarina, 2022.

PACE, Pasquale; ALOI, Gianluca; GRAVINA, Raffaele; CALICIURI, Giuseppe; FORTINO, Giancarlo; LIOTTA, Antonio. An Edge-Based Architecture to Support Efficient Applications for Healthcare Industry 4.0. **IEEE Transactions on Industrial Informatics**, *[S. l.]*, v. 15, n. 1, p. 481–489, 2019. DOI: 10.1109/TII.2018.2843169.

PHP. Introdução. Disponível em: <https://www.php.net/manual/pt_BR/intro.session.php>. Acessado em: 13 de Julho de 2022.

RICHARDSON, Leonard; RUBY, Sam. **RESTful Web Services**. United States Of America: O'Reilly Media, 2007.

SINGH, H., BHUSRY, M. **Privacy policy inference of multiple user-uploaded images on social context websites (Automated generation of privacy policy)**. 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), 2017, pp. 1-5, doi: 10.1109/CICT.2017.7977304.

SARIPALLE, Rishi Kanth. Fast health interoperability resources (FHIR): Current status in the healthcare system. **International Journal of E-Health and Medical**

Communications, [S. l.], v. 10, n. 1, p. 76–93, 2019. DOI: 10.4018/IJEHMC.2019010105.

SBD. Sociedade Brasileira de Diabetes. **Dados Epidemiológicos do diabetes mellitus no Brasil**. Disponível em: < <https://diabetes.org.br/dados-epidemiologicos/>>. Acessado em: 20 de Julho de 2022.

SU, Dejun; ZHOU, Junmin; KELLEY, Megan S.; MICHAUD, Tzeyu L.; SIAHPUSH, Mohammad; KIM, Jungyoon; WILSON, Fernando; STIMPSON, Jim P.; PAGÁN, José A. **Does telemedicine improve treatment outcomes for diabetes? A meta-analysis of results from 55 randomized controlled trials** *Diabetes Research and Clinical Practice*. Elsevier Ireland Ltd, , 2016. DOI: 10.1016/j.diabres.2016.04.019.

TERRA, Thiago Gomes. **Software de Gerenciamento, Análise e Detecção Precoce da Disfunção Autonômica em Indivíduos com Diabetes Mellitus**. 2020. Universidade Federal de Santa Catarina, [S. l.], 2020.

THUEMMLER, Christoph; BAI, Chunxue. **Health 4.0: How virtualization and big data are revolutionizing healthcare**. Cham: Springer International Publishing, 2017. DOI: 10.1007/978-3-319-47617-9. Disponível em: <http://link.springer.com/10.1007/978-3-319-47617-9>. Acesso em: 8 mar. 2021.

ÜCKER, Mariana. **Desenvolvimento de sistema point-of-care de pupilometria dinâmica aplicado no screening da Neuropatia Autonômica Diabética**. 2020. Universidade Federal de Santa Catarina, [S. l.], 2020.

VIEIRA, Igor Oliveira. **Estruturação e implantação de um servidor de sistema de informação em saúde baseado no padrão HL7 FHIR de interoperabilidade para aplicações em engenharia biomédica**, 2021. DOI: 10.1109/IEEESTD.1991.106963.

WALKER, Jan; PAN, Eric; JOHNSTON, Douglas; ADLER-MILSTEIN, Julia; BATES, David W.; MIDDLETON, Blackford. The value of health care information exchange and interoperability. **Health affairs (Project Hope)**, [S. l.], v. Suppl Web, 2005. DOI: 10.1377/hlthaff.w5.10. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/15659453/>. Acesso

em: 22 mar. 2021.

WHO. **Classification of diabetes mellitus.** [s.l: s.n.]. v. 21 DOI: 10.5005/jp/books/12855_84.

WHO. Global Report on Diabetes. 2016. Disponível em: <https://apps.who.int/iris/bitstream/handle/10665/204871/9789241565257_eng.pdf?sequence=1>. Acessado em: 12 de Julho de 2022.

WHO. WHO Technical Advisory Group of Experts on Diabetes: Objective. Disponível em: <[https://www.who.int/groups/who-technical-advisory-group-of-experts-on-diabetes-\(tag-d\)](https://www.who.int/groups/who-technical-advisory-group-of-experts-on-diabetes-(tag-d))>. Acessado em: 19 de outubro de 2022.

WORLD HEALTH ORGANIZATION. **Noncommunicable Diseases Progress Monitor 2020.** Geneva. Disponível em: <https://www.who.int/publications/i/item/ncd-progress-monitor-2020>. Acesso em: 13 maio. 2021.

W3SCHOOLS. PHP Sessions. Disponível em: <https://www.w3schools.com/php/php_sessions.asp>. Acessado em: 13 de Julho de 2022.

YAMAGUTI, Verena Hokino; MIYOSHI, Newton Shydeo Brandão; DELFINI, Milena Gomes; RIJO, Rui Pedro Charters Lopes; ALVES, Domingos. Estudo e Projeto de um Servidor de Terminologia HL7 FHIR. **Journal of Health Informatics**, [S. l.], v. 10, n. 1, p. 3–8, 2018. Disponível em: <http://www.jhi-sbis.saude.ws/ojs-jhi/index.php/jhi-sbis/article/view/523>.

APÊNDICE A – Requisições FHIR do projeto

Requisições criadas para o projeto da plataforma web desenvolvido.

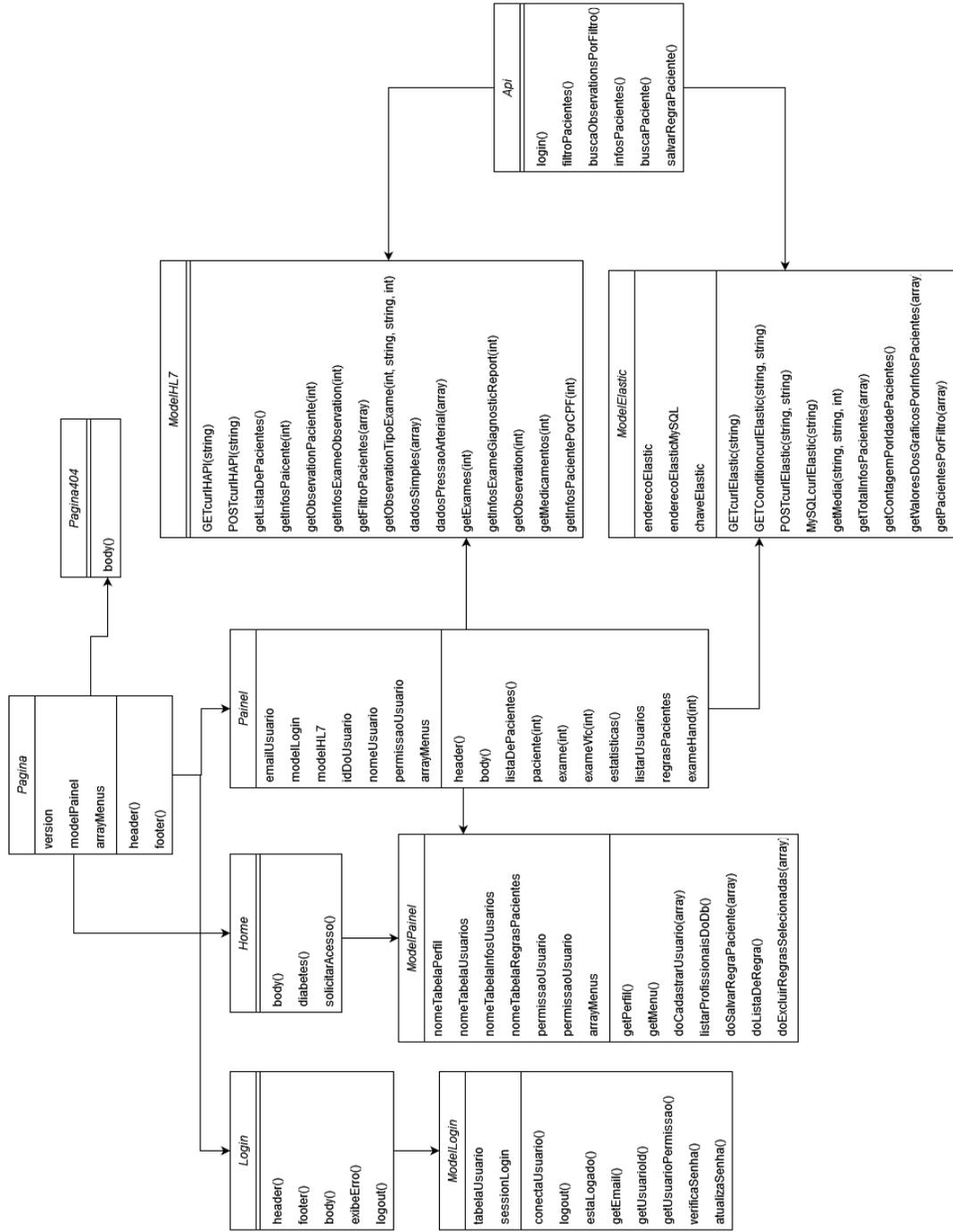
| Requisição | Função |
|---|---|
| GET { local }/fhir/Patient | Busca todos os pacientes. Por padrão o servidor retorna no máximo 20 respostas, caso tenham mais pacientes um link para acessar os próximos 20 é fornecido. |
| GET { local }/fhir/Patient?identifier={ numeroDeIdentificacao } | Pesquisa um paciente com o campo "identifier" igual ao número fornecido. Este campo pode conter o CPF do paciente para realizar esta consulta. |
| GET { local }/fhir/Patient?hasCondition:subject:code=36851000119106&_count=200 | Pesquisa pacientes com a condição de neuropatia devido a DM tipo 2. Neste caso é utilizado um código de acordo com o LOINC. |
| GET { local }/fhir/DiagnosticReport?patient={ idPaciente }&_count=1000 | Pesquisa todos os dados no recurso "DiagnosticReport" de acordo com o id do paciente e retorna 1000 resultados, caso tenham mais de 1000 resultados, o servidor adiciona um campo com um link para acessar os próximos 1000 resultados. |
| GET { local }/fhir/DiagnosticReport/{ idDiagnosticReport } | Pesquisa por id de um "DiagnosticReport" |
| GET { local }/fhir/Observation?patient={ idPaciente }&_count=5000 | Pesquisa todos os dados no recurso "Observation" de acordo com o id do paciente e retorna 500 resultados, caso tenham mais de 500 resultados, o servidor adiciona um campo com um link para acessar os próximos 500 resultados. |
| GET { local }/fhir/Observation/{ idObservation } | Pesquisa por id de um "Observation" |
| GET { local }/fhir/Observation?patient={ idPaciente }&category=laboratory&code={ codigoPadrao }&_count=200 | Pesquisa no recurso "Observation" com a categoria "laboratory" utilizando um código padrão LOINC. Esta requisição é utilizada para buscar dados de exames específicos (e.g. pressão arterial, peso, altura, glicose). |
| GET { local }/fhir/MedicationAdministration?patient={ idPaciente }&_count=500 | Busca todos os medicamentos cadastrados para um paciente. |
| GET { local }/fhir/Patient?gender={ selecao }&birthdate=lt{ dataMaior }&birthdate=gt{ dataMenor } | Requisição para buscar pacientes de um gênero e de uma determinada faixa etária. Esta requisição é utilizada para filtrar os pacientes conforme seleção do usuário. |

APÊNDICE B – Requisições *ElasticSearch* do projeto

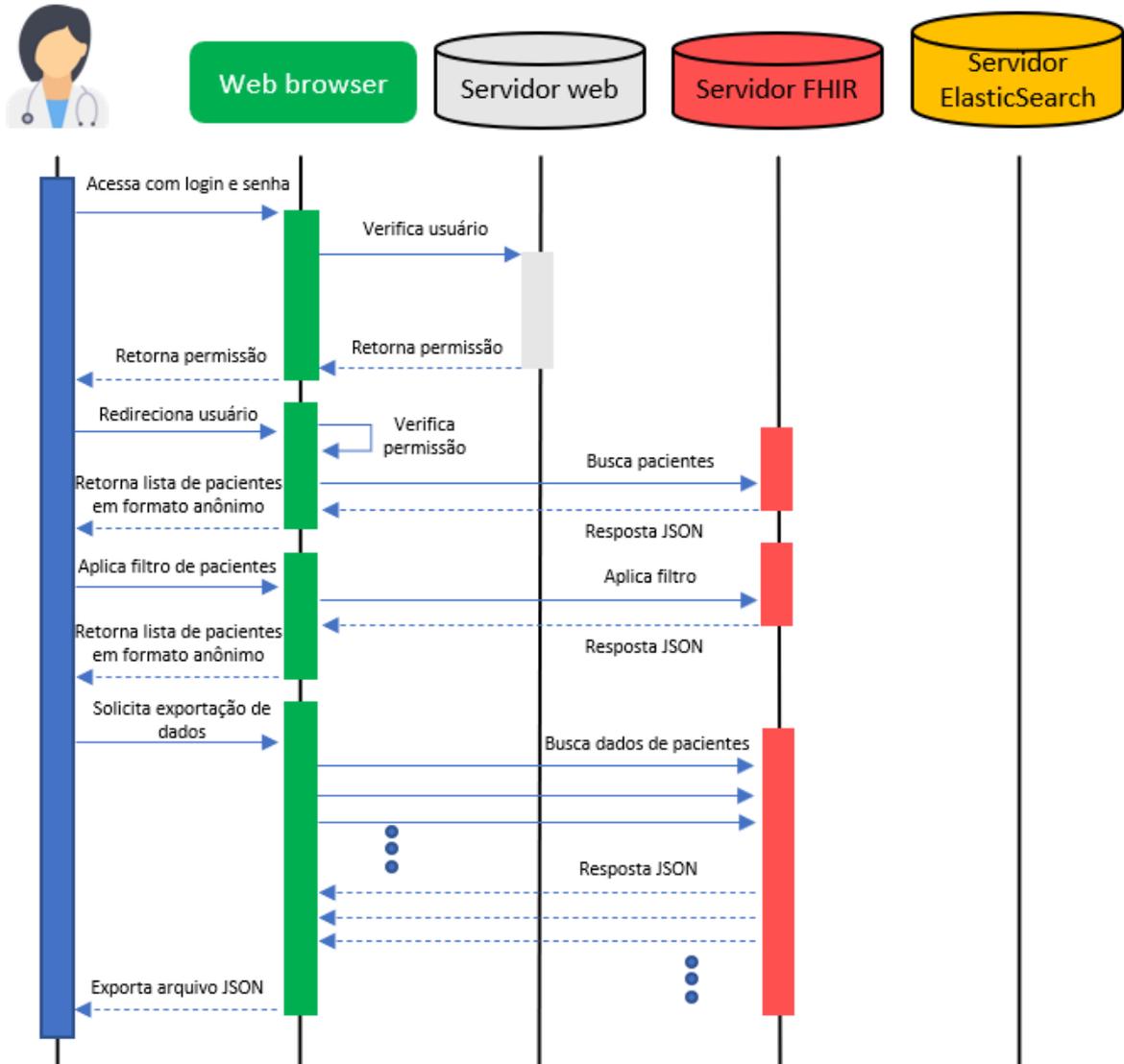
Requisições criadas para o projeto da plataforma web desenvolvido.

| Requisição | Código | Função |
|---|--|--|
| GET { local } }/patient/_doc/_search | {"track_total_hits": true} | Retorna o número total de pacientes no servidor. |
| GET { local } }/patient/_doc/_search | {"track_total_hits": true, "query": {"match": {"resource.gender": " { genero } "}}} | Retorna o número total de pacientes de acordo com o gênero. |
| POST { local } }/_xpack/sql?format=json | {"query": "SELECT ((YEAR(CURRENT_TIMESTAMP))- (YEAR(resource.birthDate)) < 20) as menor20, ((YEAR(CURRENT_TIMESTAMP))-(YEAR(resource.birthDate)) BETWEEN 20 AND 50) as maior20menor50, ((YEAR(CURRENT_TIMESTAMP))-(YEAR(resource.birthDate)) > 50) as maior50 FROM patient"} | Retorna o número total de pacientes categorizado por faixa etária |
| POST { local } }/_xpack/sql?format=json | {"query": "SELECT fullUrl, resource.gender FROM patient WHERE resource.gender IN ('male', 'female')"}} | Seleção de filtro para ambos os gêneros |
| POST { local } }/_xpack/sql?format=json | {"query": "SELECT fullUrl, resource.gender FROM patient WHERE resource.gender IN ({ variavelFiltroGenero })"}} | Seleção de filtro para um gênero selecionado |
| POST { local } }/_xpack/sql?format=json | {"query": "SELECT fullUrl, resource.gender FROM patient WHERE resource.gender IN ('male', 'female') AND resource.birthDate > ' { dataVariavelPrimeiro } ' AND resource.birthDate < ' { dataVariavelPrimeiro } '"}} | Filtro para ambos os gêneros com seleção de faixa etária |
| POST { local } }/_xpack/sql?format=json | {"query": "SELECT fullUrl, resource.gender FROM patient WHERE resource.gender IN ({ variavelFiltroGenero }) AND resource.birthDate > ' { dataVariavelPrimeiro } ' AND resource.birthDate < ' { dataVariavelPrimeiro } '"}} | Filtro com seleção para gênero e para faixa etária |
| POST { local } }/_xpack/sql?format=json | {"query": "SELECT AVG(resource.valueQuantity.value) FROM observation WHERE resource.code.coding.code = { codigoLOINC }"}} | Retorna a média de todos os valores de um exame no recurso FHIR "Observation" de acordo com seu código LOINC |

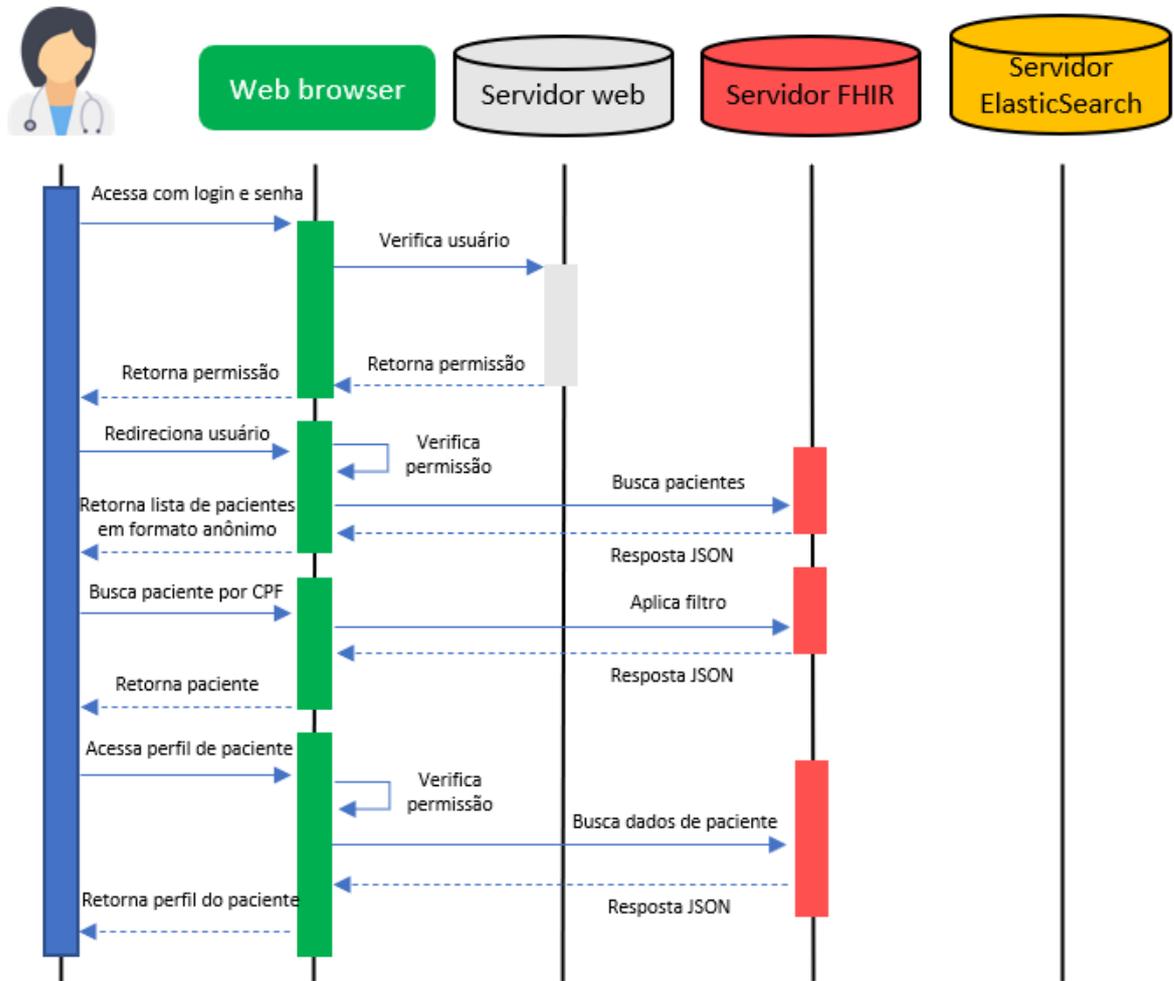
APÊNDICE C – Diagrama de Classes da Plataforma Web

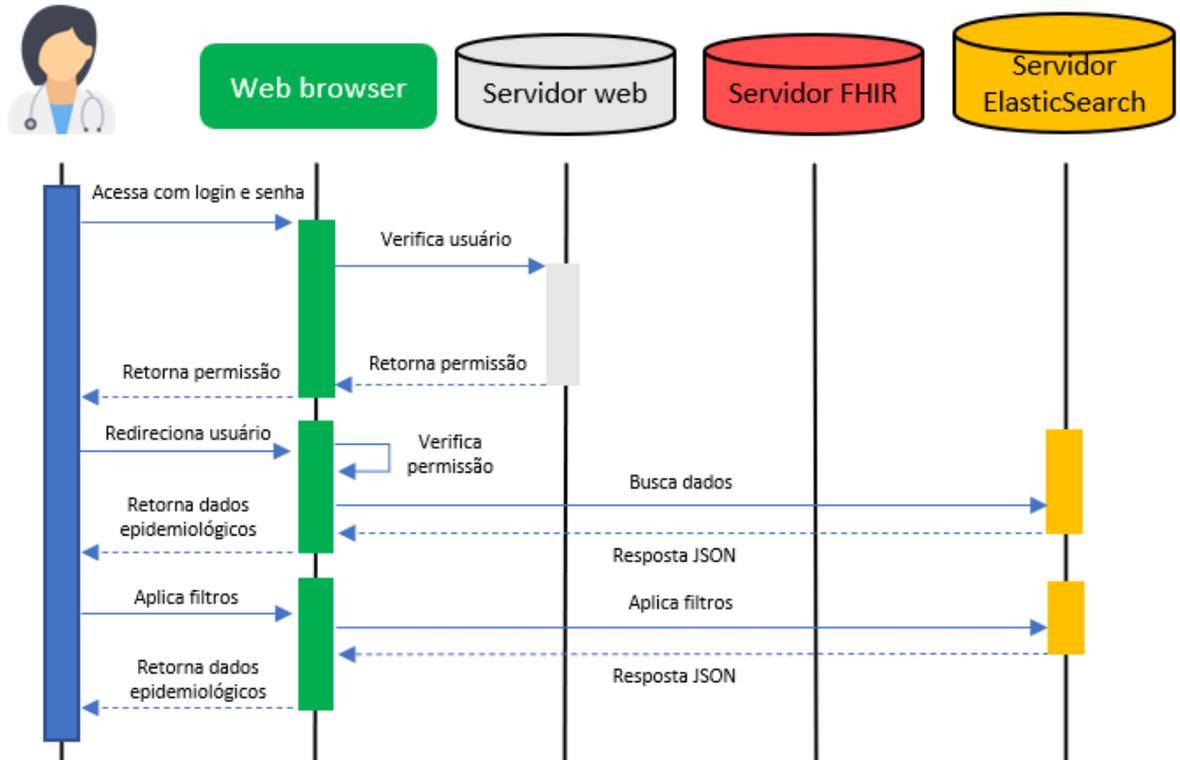


APÊNDICE D – Diagrama UML para o perfil Pesquisador

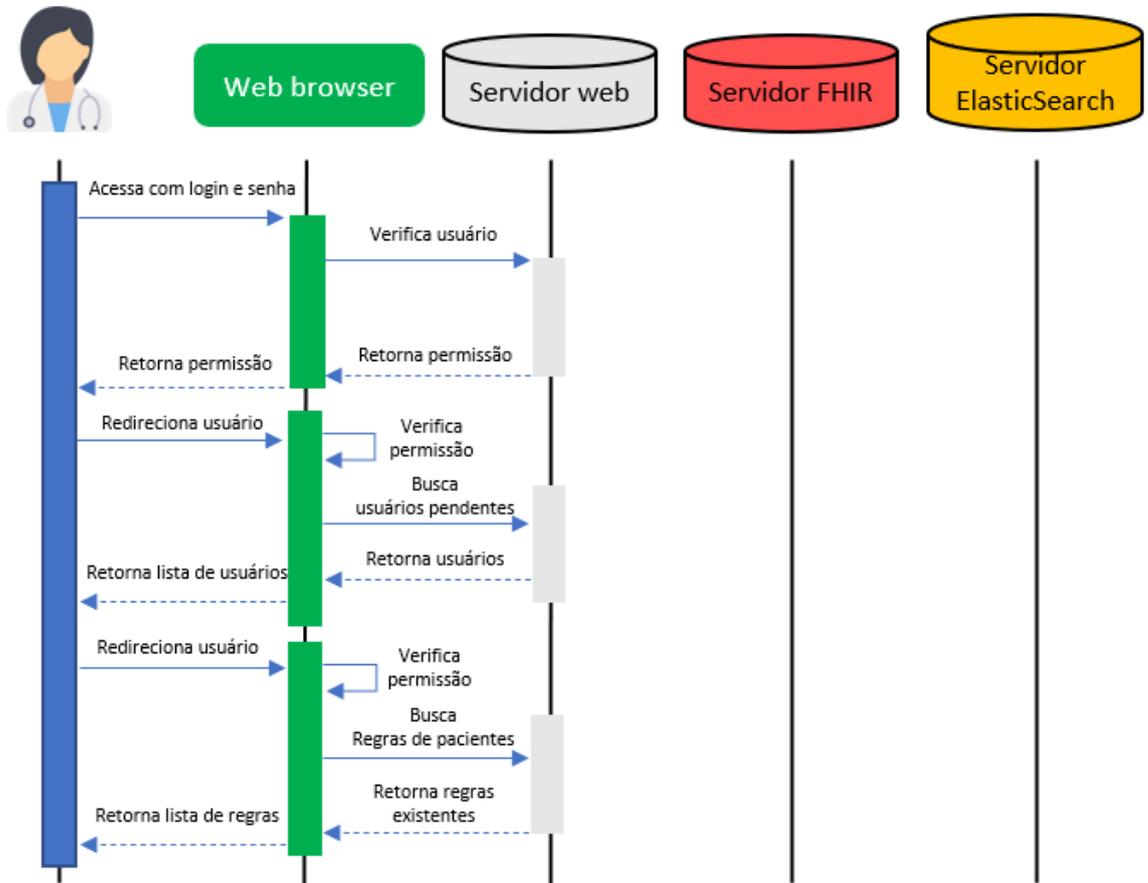


APÊNDICE E – Diagrama UML para o perfil Profissional



APÊNDICE F – Diagrama UML para o perfil Gestor

APÊNDICE G – Diagrama UML para o perfil Administrador



APÊNDICE H – Diagrama UML para o perfil Paciente

