
FACULDADE DE TECNOLOGIA DE AMERICANA "Ministro Ralph Biasi"
Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Wayne Andrade Silva Neto
Wellington Luz Caires

TRACKMED

Americana, SP
2023

FACULDADE DE TECNOLOGIA DE AMERICANA "Ministro Ralph Biasi"
Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Wayne Andrade Silva Neto
Wellington Luz Caires

TRACKMED

Trabalho de Conclusão de Curso desenvolvido em cumprimento à exigência curricular do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, sob a orientação do Mestre em Educação Vagner Ferreira.

Área de concentração: Tecnologia da Informação.

Americana, SP

2023

**FICHA CATALOGRÁFICA – Biblioteca Fatec Americana Ministro Ralph Biasi-
CEETEPS Dados Internacionais de Catalogação-na-fonte**

SILVA NETO, Wayne Andrade

Trackmed. / Wayne Andrade Silva Neto, Wellington Luz Caires – Americana, 2023.

92f.

Monografia (Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas) - - Faculdade de Tecnologia de Americana Ministro Ralph Biasi – Centro Estadual de Educação Tecnológica Paula Souza

Orientador: Prof. Ms. Vagner Ferreira

1. Banco de dados 2. Desenvolvimento de software. I. FERREIRA, Vagner II. Centro Estadual de Educação Tecnológica Paula Souza – Faculdade de Tecnologia de Americana Ministro Ralph Biasi

CDU: 681.3.07

681.3.05

Elaborada pelo autor por meio de sistema automático gerador de ficha catalográfica da Fatec de Americana Ministro Ralph Biasi.

Wayne Andrade Silva Neto
Wellington Luz Caires

TRACKMED

Trabalho de graduação apresentado como exigência parcial para obtenção do título de Tecnólogo em Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas pelo Centro Paula Souza – FATEC Faculdade de Tecnologia de Americana – Ralph Biasi.
Área de concentração: Tecnologia da Informação.

Americana, 15 de junho de 2023

Banca Examinadora:



Vagner Ferreira
Mestre em Educação
Fatec Americana



Ivan Menerval da Silva
Dr. Engenharia de Produção
Fatec Americana



Eduardo Antônio Vicentini
Mestre em Direito
Fatec Americana

RESUMO

Apesar do pouco investimento na saúde, o Brasil precisa lidar ainda com a falta de informatização nessa área. Embora existam diversos sistemas informatizados no Sistema Único de Saúde (SUS), os dados gerados são dispersos, não existe um compartilhamento expressivo entre eles, de maneira que possam gerar valor e auxiliar a poupar verba, evitar falsificações, auxiliar os profissionais e melhorar a qualidade de vida da população.

No projeto TrackMed, foi proposto uma Interface de Programação de Aplicação (API) baseada em um sistema de microsserviços que pode sanar esse problema, a API tem o intuito de unir os mais variados órgãos do governo, fazendo uso de dados de diversas instituições que ainda não foram unidas ao ambiente Gov.br. Isso tornará possível fiscalizações de clínicas, hospitais, médicos, farmácias e medicamentos em tempo real, trazendo inúmeros benefícios a população, profissionais da área da saúde e ao Estado brasileiro, além de auxiliar os profissionais durante seu trabalho, como a possibilidade de os médicos consultar o histórico do paciente. O objetivo do trabalho foi trazer uma maneira de interligar os dados da base do governo, visando sempre a otimização de recursos e a melhoria da qualidade de vida dos cidadãos e profissionais da área da saúde. Na metodologia, por meio de uma pesquisa exploratória, de abordagem qualitativa, verificou-se que existe a real necessidade da digitalização do SUS, e que embora o governo disponha de uma base de dados muito grande, essa base está sendo subutilizada.

Palavras-Chave: SUS, informatização, prescrição médica.

ABSTRACT

Despite the limited investment in healthcare, Brazil still needs to deal with the lack of computerization in this field, despite the existence of various computerized systems in the Unified Health System (SUS), the generated data is scattered, lacking significant sharing among them to generate value and help save resources, prevent counterfeiting, assist professionals, and improve the population's quality of life.

In the TrackMed project, an API based on a microservices system has been proposed to solve this problem. The API aims to bring together various government agencies, making use of data from different institutions that have not yet been integrated into the Gov.br environment. This integration allows for real-time monitoring of clinics, hospitals, doctors, pharmacies, and medications, bringing numerous benefits to the population, healthcare professionals, and the Brazilian government. Additionally, it assists professionals in their work, such as allowing doctors to access a patient's medical history. The aim of the work was to provide a way to connect government database information, always seeking resource optimization and improvement in the quality of life for citizens and healthcare professionals.

In the methodology, through exploratory research with a qualitative approach, it was found that there is a real need for digitizing the SUS (Unified Health System) and that despite the government having a very large database, this database is being underutilized.

Keywords: SUS, informatization, medical prescription.

SUMÁRIO

| | | |
|-------|---|----|
| 1 | INTRODUÇÃO | 10 |
| 2 | METODOLOGIA DE PESQUISA | 12 |
| 3 | PROJETO DO SISTEMA | 13 |
| 3.1 | Inclusão Social | 14 |
| 3.2 | Economia para o Governo | 15 |
| 3.3 | Implementação da API Trackmed | 16 |
| 3.4 | Vantagens da API | 18 |
| 3.5 | Arquitetura da API..... | 19 |
| 3.6 | Dificuldades da implementação | 19 |
| 3.7 | Softwares Similares | 20 |
| 3.8 | Levantamento de Requisitos | 21 |
| 3.8.1 | Requisitos Funcionais | 22 |
| 3.8.2 | Requisitos Não Funcionais..... | 23 |
| 3.9 | Recursos e Ferramentas | 24 |
| 4 | MODELAGEM | 27 |
| 4.1 | Casos De Uso | 27 |
| 4.2 | Documentação dos Casos de Uso | 31 |
| 4.2.1 | Casos de Uso Microserviço Citizen | 31 |
| 4.2.2 | Casos de Uso Microserviço Hospital..... | 34 |

| | | |
|-------|---|----|
| 4.2.3 | Casos de Uso Microserviço Pharmacy | 42 |
| 4.4 | Diagrama de Entidade e Relacionamento | 44 |
| 4.4.1 | Dicionário de Dados | 47 |
| 4.5 | Plano de testes | 62 |
| 5 | DESENVOLVIMENTO | 75 |
| 5.1 | Sprint 1..... | 75 |
| 5.2 | Sprint 2..... | 76 |
| 5.3 | Sprint 3..... | 77 |
| 5.4 | Sprint 4..... | 78 |
| 5.5 | Sprint 5..... | 79 |
| 5.6 | Sprint 6..... | 80 |
| 5.7 | Sprint 7..... | 82 |
| 5.8 | Sprint 8..... | 83 |
| 5.9 | Sprint 9..... | 84 |
| 6 | CONSIDERAÇÕES FINAIS | 85 |
| | REFERÊNCIAS: | 87 |

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 – Diagrama de caso de uso do Cidadão | 29 |
| Figura 2 - Diagrama de caso de uso da Recepcionista | 29 |
| Figura 3 - Diagrama de caso de uso do Médico | 30 |
| Figura 4 – Diagrama de caso de uso do Farmacêutico | 30 |
| Figura 5 - Diagrama de Entidade e Relacionamento Micro Serviço Cidadão..... | 45 |
| Figura 6 - Diagrama de Entidade e Relacionamento Micro Serviço Hospital | 46 |
| Figura 7 - Diagrama de Entidade e Relacionamento Micro Serviço Farmácia | 47 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 - Comparativo de funcionalidades da aplicação TrackMed em relação aos concorrentes..... | 21 |
| Tabela 2 – Requisitos funcionais do projeto..... | 23 |
| Tabela 3 – Requisitos não funcionais do projeto..... | 24 |
| Tabela 4 – Caso de uso “Entrar no Sistema (Cidadão)”..... | 31 |
| Tabela 5 - Caso de uso “Criar Cadastro Completo” | 32 |
| Tabela 6 - Caso de uso “Visualizar histórico de tratamentos” | 33 |
| Tabela 7 - Caso de uso “Atribuir compra de medicamentos para responsável” | 33 |
| Tabela 8 - Caso de uso “Alterar dados no sistema” | 34 |
| Tabela 9 - Caso de uso “Entrar no sistema (Recepcionista)” | 34 |
| Tabela 10 - Caso de uso “Agendar Consulta” | 35 |
| Tabela 11 - Caso de uso “Recepcionar Paciente Agendado” | 36 |
| Tabela 12 - Caso de uso “Liberar Paciente para Consulta”. | 37 |
| Tabela 13 - Caso de uso “Entrar no sistema (Médico)” | 38 |
| Tabela 14 - Caso de uso “Consultar Paciente”..... | 39 |
| Tabela 15 - Caso de uso “Visualizar Histórico do Paciente” | 40 |
| Tabela 16 - Caso de uso “Prescrever Receituário” | 41 |
| Tabela 17 - Caso de uso “Entrar no sistema (Farmacêutico)”..... | 42 |
| Tabela 18 - Caso de uso “Iniciar Atendimento” | 43 |
| Tabela 19 - Caso de uso “Vender Medicamento”..... | 44 |
| Tabela 20 - Dicionário de Dados Tabela <i>HOS_T_HOSPITAL_MEDIC</i> | 48 |
| Tabela 21 - Dicionário de Dados Tabela <i>HOS_T_MEDIC</i> | 48 |
| Tabela 22 - Dicionário de Dados Tabela <i>HOS_T_HOSPITAL</i> | 49 |

| | |
|--|----|
| Tabela 23 - Dicionário de Dados Tabela <i>HOS_T_APPOINTMENT</i> | 49 |
| Tabela 24 - Dicionário de Dados Tabela <i>HOS_T_RECEPCIONIST</i> | 49 |
| Tabela 25 - Dicionário de Dados Tabela <i>CIT_T_ADRESS</i> | 50 |
| Tabela 26 - Dicionário de Dados Tabela <i>CIT_T_CITIZEN</i> | 50 |
| Tabela 27 - Dicionário de Dados Tabela <i>CIT_T_TREATMENT</i> | 50 |
| Tabela 28 - Dicionário de Dados Tabela <i>CIT_T_AUTHORIZED_PERSON</i> | 51 |
| Tabela 29 - Dicionário de Dados Tabela <i>PHA_T_REGULATORY_PH_BODY</i> | 51 |
| Tabela 30 - Dicionário de Dados Tabela <i>PHA_T_PHARMACEUTICAL</i> | 51 |
| Tabela 31 - Dicionário de Dados Tabela <i>PHA_T_PHARMACY</i> | 52 |
| Tabela 32 - Dicionário de Dados Tabela <i>PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING</i> | 52 |
| Tabela 33 - Dicionário de Dados Tabela <i>PHA_T_TREATMENT_SHOPPING</i> | 52 |
| Tabela 34 - Dicionário de Dados Atributos da Tabela <i>HOS_T_REGULATORY_MEDIC_BODY</i> | 53 |
| Tabela 35 - Dicionário de Dados Atributos da Tabela <i>HOS_T_HOSPITAL_MEDIC</i> | 53 |
| Tabela 36 - Dicionário de Dados Atributos da Tabela <i>HOS_T_MEDIC</i> | 54 |
| Tabela 37 - Dicionário de Dados Atributos da Tabela <i>HOS_T_HOSPITAL</i> | 54 |
| Tabela 38 - Dicionário de Dados Atributos da Tabela <i>HOS_T_APPOINTMENT</i> | 55 |
| Tabela 39 - Dicionário de Dados Atributos da Tabela <i>HOS_T_RECEPCIONIST</i> | 56 |
| Tabela 40 - Dicionário de Dados Atributos da Tabela <i>CIT_T_ADRESS</i> | 57 |
| Tabela 41 - Dicionário de Dados Atributos da Tabela <i>CIT_T_CITIZEN</i> | 58 |
| Tabela 42 - Dicionário de Dados Atributos da Tabela <i>CIT_T_TREATMENT</i> | 59 |
| Tabela 43 - Dicionário de Dados Atributos da Tabela <i>CIT_T_AUTHORIZED_PERSON</i> | 59 |

| | |
|---|----|
| Tabela 44 - Dicionário de Dados Atributos da Tabela <i>PHA_T_REGULATORY_PH_BODY</i> | 60 |
| Tabela 45 - Dicionário de Dados Atributos da Tabela <i>PHA_T_PHARMACY</i> | 60 |
| Tabela 46 - Dicionário de Dados Atributos da Tabela <i>PHA_T_PHARMACEUTICAL</i> | 61 |
| Tabela 47 - Dicionário de Dados Atributos da Tabela <i>PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING</i> | 61 |
| Tabela 48 - Dicionário de Dados Atributos da Tabela <i>PHA_T_TREATMENT_SHOPPING</i> | 62 |
| Tabela 49 - Caso de Teste - <i>Cadastro Completo</i> | 63 |
| Tabela 50 - Caso de Teste - <i>Login Cidadão</i> | 63 |
| Tabela 51 - Caso de Teste - <i>Visualizar Histórico de Tratamentos</i> | 63 |
| Tabela 52 - Caso de Teste - <i>Atribuir Compra de Medicamentos para Responsável</i> . | 64 |
| Tabela 53 - Caso de Teste - <i>Alteração de Cadastro</i> | 65 |
| Tabela 54 - Caso de Teste - <i>Login Recepcionista</i> | 65 |
| Tabela 55 - Caso de Teste - <i>Agendar Consulta</i> | 66 |
| Tabela 56 - Caso de Teste - <i>Recepcionar Paciente Agendado</i> | 67 |
| Tabela 57 - Caso de Teste - <i>Liberar Paciente para Consulta</i> | 68 |
| Tabela 58 - Caso de Teste - <i>Login Médico</i> | 68 |
| Tabela 59 - Caso de Teste - <i>Consultar Paciente</i> | 69 |
| Tabela 60 - Caso de Teste - <i>Visualizar Histórico do Paciente</i> | 70 |
| Tabela 61 - Caso de Teste - <i>Prescrever Receituário</i> | 71 |
| Tabela 62 - Caso de Teste - <i>Login Farmacêutico</i> | 72 |
| Tabela 63 - Caso de Teste - <i>Iniciar Atendimento</i> | 73 |
| Tabela 64 - Caso de Teste - <i>Vender Medicamento</i> | 74 |
| Tabela 65 – Planejamento realizado para primeira entrega | 75 |

| | |
|---|----|
| Tabela 66 – Planejamento realizado para primeira entrega..... | 76 |
| Tabela 67 – Planejamento realizado para primeira entrega..... | 78 |
| Tabela 68 – Planejamento realizado para primeira entrega..... | 79 |
| Tabela 69 – Planejamento realizado para primeira entrega..... | 80 |
| Tabela 70 – Planejamento realizado para primeira entrega..... | 81 |
| Tabela 71 – Planejamento realizado para primeira entrega..... | 82 |
| Tabela 72 – Planejamento realizado para primeira entrega..... | 83 |
| Tabela 73 – Planejamento realizado para primeira entrega..... | 84 |

1 INTRODUÇÃO

Todos os cidadãos brasileiros ou estrangeiros quando passam por algum hospital, seja público ou privado, é necessário ter um registro no Sistema Único de Saúde (SUS), caso não tenha esse cadastro, ele é feito imediatamente. Esse procedimento visa manter uma base de registros de todos os usuários do SUS, embora o sistema possua os dados dos usuários como, registros sobre as cirurgias, registro das consultas que muitas vezes não entram em sistema informatizado, muitos desses dados não são aproveitados da melhor forma pelos usuários nem pelo próprio sistema do SUS.

A proposta do Trackmed é tornar todos esses dados que transitam por todo sistema nacional de saúde em informações que podem auxiliar na fiscalização, disponibilização mais precisa de recursos, melhor redistribuição de agendamento de consultas além da validação imediata de medicamentos que estão aprovados ou não pela Anvisa para o consumo. Também, validação dos registros médicos, farmacêuticos e hospitais, fazendo uma melhor utilização dos recursos que já existem hoje, além de possibilitar um crescimento imenso em diversas áreas de pesquisas que podem se beneficiar dos dados gerados para gerar informação precisa, seja na área de ensino, ou farmacêutico ou demais áreas que possam fazer uso desses dados. Logicamente sempre preservando os usuários, ou seja, os dados não são isolados por pacientes, mas apenas o montante do volume.

Trackmed é um sistema baseado em uma arquitetura de microsserviços que visa integrar todos os diversos sistemas da área da saúde sem que implique nos processos de rotinas de cada estabelecimento ou necessidade de trocar um sistema complexo que já está sendo utilizado por um sistema novo que demanda uma curva de aprendizado. No entanto os softwares que hoje auxiliam hospitais e farmácias precisam passar apenas por uma adequação que possa ser validado pela API do Trackmed que auxiliará para a geração das informações para a base de dados do governo.

Por outro lado, os cidadãos também podem usufruir de benefícios que o Trackmed pode oferecer.

Durante a pandemia o governo criou o ConecteSUS que hoje basicamente informa as vacinas que o cidadão tomou contra o coronavírus e mais algumas poucas

funcionalidades. Para que os cidadãos consigam visualizar seus históricos de consultas, realizar compra de medicamentos ou poder atribuir a compra de um responsável para realizar a compra, o aplicativo do ConecteSUS pode realizar essa tarefa, apenas implementando a API do Trackmed. Nesses aspectos, justifica-se a relevância desta pesquisa.

Embora muitas das informações e sistemas utilizados para a integração já estejam prontos, será necessária uma adequação na infraestrutura para conseguir atender a todas as requisições.

A pergunta que direciona o problema de pesquisa é: como a criação de uma API, o Trackmed, pode contribuir com a melhoria da integração de serviços e informações no âmbito do sistema de saúde?

Nesse contexto, o objetivo desta pesquisa foi elaborar uma proposta de criação de uma API para poder facilitar a fiscalização tanto dos profissionais que atuam na área médica e farmacêutica, estabelecimentos médicos, além de facilitar a vida do cidadão no momento da compra de seu medicamento, ou a possível atribuição dessa atividade a um terceiro. Além da possibilidade de crescimento de atribuições dentro da própria aplicação, com a utilização de gerenciamento de dados, surge a possibilidade da melhor gestão de recursos público, uma visão mais ampla da saúde no Brasil, além de diversas fontes iniciais para pesquisas voltadas a saúde de modo geral.

2 METODOLOGIA DE PESQUISA

A metodologia de pesquisa adotada foi a exploratória, pois não foram identificadas tentativas anteriores de abordar os gastos excessivos e integração do SUS (Sistema Único de Saúde). Assim, este projeto representou uma iniciativa pioneira e inovadora.

Na primeira etapa, realizou-se uma investigação sobre como desenvolver uma solução inteligente para melhorar a qualidade de vida dos pacientes e otimizar a obtenção de medicamentos. Nesse contexto, foi observado que o governo brasileiro não possui sistemas modernos ou inteligentes para identificar os cidadãos, o que é um desafio considerando o tamanho do país.

Na segunda etapa da pesquisa, foi constatado que a digitalização desses processos traria benefícios significativos para os recursos públicos. Foi explorado conjuntamente maneiras de melhorar a qualidade de vida dos usuários ao adquirirem medicamentos mediante a substituição do receituário físico, o que também proporciona maior segurança na comercialização de remédios pelas farmácias e reduz fraudes na distribuição.

Na terceira etapa da pesquisa, foi identificada a importância de abordar as falsificações de identidade, nas quais pessoas não autorizadas se passam por médicos ou farmacêuticos, colocando em risco a saúde dos pacientes. Para combater essa questão, foi proposto a validação da API por meio da consulta aos conselhos regionais responsáveis por essas atividades.

Com base nessas informações e pesquisas direcionadas, chegou-se a um ponto crucial que aborda todas essas necessidades: a centralização e integração dessas informações. Isso permite que o governo brasileiro tome decisões embasadas para investimentos na saúde, priorizando as áreas que mais necessitam.

Essa abordagem de coleta e tratamento de dados, que se transformam em informações para a central de inteligência do governo, possibilitaria uma resposta reativa e proativa em relação a surtos de doenças, proporcionando tratamentos mais eficazes em cada região do país.

Como resultado desse trabalho de pesquisa, foi proposto a criação da API chamada TRACKMED, que representa a consolidação dessas ideias e soluções para enfrentar os desafios identificados.

3 PROJETO DO SISTEMA

De acordo com o site medicinasa.com, o Brasil é o país que mais realiza coleta de dados no setor da saúde, conta com o Sistema Único de Saúde (SUS), que possui uma base de informações do DATASUS criada no ano de 1991. Mesmo com o sistema efetivo e toda a informação gerada pela base, não há como reunir e transformá-los essa informação em dados e conhecimento, isso devido à falta de digitalização das informações e de inoperabilidade, impactando diretamente em melhorias para todo o setor.

O setor de saúde no Brasil, necessita urgentemente de modernização e digitalização, que mesmo com a pandemia, ainda ocorreram de maneira muito lenta. É necessário trabalhar esses dados, transformá-los em informações para que as instituições tenham, previsibilidade de custos e uma melhor gestão que resultará em excelência de atendimento ao paciente final.

Hoje ainda existe muita dispersão de performance no elo, entre as empresas, setor público e privado, com muitos operando abaixo do seu potencial.

O DATASUS possui 600 sistemas para fornecer informações, com dados de nascimentos, óbitos, hospitalizações, dados ambulatoriais dos últimos 20 anos. Uma base de dado com extrema riqueza de detalhes publicamente abertas, inexistente em outra parte do mundo. Então resumindo, de um lado temos muitas informações do setor público, com estatísticas integradas pelo Ministério da Saúde, do outro lado, o sistema privado muito avançado, mas ainda muito fragmentado.

O ponto principal a ser considerado é este, a integração entre o sistema público e privado, que busca o ganho de eficiência. O sistema a longo médio prazo não irá sustentar, sem essa premissa, os custos serão impeditivos para execução. É necessário buscar em um único sistema consultas e exames distintos e encontrá-los precisamente, as integrações dos sistemas trarão ganhos significativos para todo o âmbito envolvido.

3.1 Inclusão Social

Em razão do nosso dia a dia, em que tudo é conectado, sempre existe um aplicativo para alguma necessidade, pode passar despercebido que mais de 33 milhões de brasileiros não tem acesso à internet de acordo com pesquisa realizada pela empresa Pwc em parceria com o Instituto Locomotiva (2022), seja por questões financeiras, localidade, idade, incapacidade física-mental entre outras.

É preciso levar em conta que esses cidadãos merecem e devem ter o mesmo tipo de atendimento e benefício que os demais, para solucionar essa questão de pacientes que não possuem ou simplesmente não querem acessar um aplicativo para compra de medicamento, visualizar seu histórico de tratamentos, ou se ainda não teve conhecimento dos novos recursos disponibilizados. O atendimento deles continuarão ocorrendo sem alteração alguma, esses cidadãos irão até sua consulta normalmente, e caso seja sua primeira consulta desde a implementação da API Trackmed, apenas será solicitado para que a recepcionista insira as informações de cpf e número do cartão do sus desse cidadão, informações que normalmente já são solicitadas em qualquer consulta, com isso existe um cadastro básico desse cidadão na base de dados Trackmed, garantindo integridade em todas as operações.

Mas pode ocorrer ainda outra situação: suponha-se que um cidadão que não utilizava o aplicativo Gov.br teve conhecimento sobre as novas funcionalidades e se interessou em utilizar o aplicativo, para isso ele baixa o aplicativo e cria sua conta, mas supondo que ele já passou algumas vezes pelo médico desde a implementação do Trackmed, sua conta já foi criada e não existe uma senha cadastrada. Nesses casos, em que já existe um cadastro básico do cidadão na base de dados, o cidadão terá de informar os dados iniciais como CPF, número do cartão do sus, e a verificação ocorrerá com a respostas a perguntas específicas com base nos registros desse cidadão na base do governo, como ocorre hoje com outros aplicativos do governo, como por exemplo, se você ainda não possui a carteira de trabalho digital e decide que quer utilizar, você informa seu CPF, e responde algumas perguntas para que sua entrada seja validada.

3.2 Economia para o Governo

Após uma análise aprofundada sobre o impacto da informatização do SUS na economia brasileira, constatou-se que a implementação dessa medida poderia gerar uma economia significativa, alcançando a marca de R\$ 22 bilhões de reais por ano. Essa estimativa se baseia em um estudo realizado em 2020 pelo Banco Mundial e requer um investimento inicial de R\$ 1,5 bilhão de reais anuais.

Nesse quesito não apenas a API Trackmed auxiliaria nesse processo, como também em uma segunda etapa seria capaz de fazer muito mais do que isso, poderia servir como base para todo um direcionamento de recursos, tanto para hospitais públicos, como para hospitais privados, que nesse caso poderia ser realizado cobrança ou não mediante a regra de negócios que não fazem parte do escopo da equipe.

Com a utilização da API, seria fácil na segunda etapa do projeto, trabalhar com os dados gerados, ou seja, análise de dados, podendo gerar informações das mais variadas possíveis auxiliando nas tomadas de decisões, tendo como foco a gestão pública, alguns exemplos podem ser citados. Suponha um hospital que atende uma cidade com 50 mil habitantes, e o orçamento mensal desse hospital seria de 2 milhões por mês (valores hipotéticos), no entanto durante a análise de dados, foi constatado que uma cidade com 30 mil habitantes utiliza-se do mesmo orçamento, obviamente existem diversos fatores que determinam o orçamento de um hospital, como se ele possui leitos de unidade intensiva de tratamento, se é hospital com alguma especialização, mas tudo isso é possível ser parametrizado, e comparar os investimentos, se ambos os hospitais possuírem as mesmas características, os custos operacionais per capita da região deveriam ser parecidos, levando em conta apenas alguns fatores econômicos da região. Por meio desses levantamentos é possível melhorar a fiscalização desses estabelecimentos, visto que as autoridades terão acesso mais facilitado à essas informações, podendo inclusive ser publicado nos portais de transparência do governo.

Além da possibilidade de se tornar uma ferramenta importante nas gestões dos recursos destinados a hospitais, a análise de dados fornecida pela API Trackmed, poderá ainda auxiliar com identificação de surtos em determinadas regiões, pesquisas que ajudem a entender melhor a saúde da população brasileira, como faixa etária dos atendimentos, aumento ou redução de determinado sintoma ou doença por região,

quantidade de consultas por hospitais por densidade demográfica, e incontáveis outros cenários que podem ser explorados conforme demanda.

3.3 Implementação da API Trackmed

Trackmed é uma API baseada na estrutura de microsserviços com a intenção de ser um agente que atuará nas interligações de dados já existentes espalhados por diversos órgãos reguladores do sistema de saúde brasileiro.

Uma API é um conjunto de rotinas, protocolos e ferramentas para criar software e aplicativos. Como tal, fornece aos desenvolvedores acesso a serviços de uma aplicação sem ter que entender como essa aplicação funciona internamente. Isso permite que as organizações exponham seus serviços para uso interno ou externo em um ambiente seguro e controlado. (JACOBSON, BRAIL, WOODS, 2011, p. 3)

O desafio é fazer com que essa informação seja utilizada de forma simples rápida e automática durante toda a cadeia de eventos em que um cidadão inicia uma consulta até a compra do seu medicamento.

Para que o sistema seja de fato efetivo, todos os eventos devem ser obrigados a passarem pela API Trackmed, que será o responsável por validar junto aos órgãos responsáveis a transação efetuada naquele momento. Por exemplo, se um médico vai prescrever um medicamento para um paciente, existem algumas validações que podem ser feitas no ato da prescrição, como validar se o médico que está prescrevendo é de fato um médico, validar se o registro dele no conselho de medicina está ativo e ainda se ele pode atender para determinada área da medicina, um cardiologista não pode ser responsável por atendimentos de um neurocirurgião. Além de outras validações como se o medicamento prescrito está liberado e aprovado pela Anvisa, todas essas validações ocorrerão automaticamente na API Trackmed, sem que o médico precise mudar sua rotina, aliás a API Trackmed não vem para sobrecarregar os envolvidos, e sim ser um aliado no combate a irregularidades e no futuro ser um auxílio ainda mais efetivo para os profissionais da área.

Obviamente fazer com que todas as prescrições médicas passem por uma única API parece ser um pouco tendencioso, inclusive pode ser visto pelo mercado como monopólio, gerando bastante descontentamento entre empresas produtoras de softwares para essa categoria. Por isso a decisão de que a API Trackmed seria uma ideia, um conceito cedido ao governo que se encarregaria de sua implementação por razões óbvias, já que a principal função da API é fazer a interligação de diversos órgãos do governo. Ceder o direito ao governo ainda não resolveria o problema do descontentamento dos produtores de softwares, ainda seria um monopólio, mas dessa vez seria por parte do governo, além disso é necessário levar em consideração os milhares de estabelecimentos médicos e farmacêuticos que já fazem uso de algum software que os ajudam a gerir toda sua rotina, seja financeira, estoques entre outros, seria simplesmente impossível criar um único software que atenda todos os requisitos de todos esses estabelecimentos.

Então para solucionar de vez todos esses desafios, a melhor alternativa encontrada é que a API Trackmed teria de passar por votações no senado e na câmara dos deputados, para que seja considerado como uma legislação, dessa forma não ferindo o que se refere a lei da declaração de direitos de liberdade econômica, "Lei nº 13.874/2019 - Lei da Liberdade Econômica", é muito burocrático o processo, mas é a melhor maneira para garantir a implementação da API Trackmed e a livre comercialização de softwares, mas o ponto mais importante de todo esse processo, é que ao se tornar uma legislação, os fabricantes de softwares terão um período para se adequarem as exigências.

Essas exigências seriam basicamente que determinadas rotinas executadas por profissionais da área da saúde tenham que passar pela API Trackmed, como o exemplo que foi dado inicialmente, em que um médico prescreve um medicamento para um paciente. Para o médico ele simplesmente poderia estar inserindo um medicamento em uma tela de sistema que ele já utiliza hoje no hospital em que trabalha, com o software que o hospital adquiriu, mas a implementação que o fornecedor do software teve de fazer por força de lei, garante que as validações sejam executadas imediatamente. Nesse exemplo é possível ver como a rotina médica de prescrever um medicamento não foi impactada de forma alguma para o profissional médico, mas algumas situações poderiam acontecer aqui, como o fato de determinado medicamento estar barrado pela Anvisa naquele momento, e o médico ter que trocar

o medicamento, ou até mesmo se for um falso médico ser descoberto imediatamente, caso ele já atue como médico naquele hospital antes da implementação da API.

3.4 Vantagens da API

Os benefícios da utilização da API Trackmed logo no início são diversos, validações dos profissionais envolvidos, como se o médico está apto ou não a exercer a função, a farmácia possui seus farmacêuticos registrados, além de verificar no ato da prescrição e da compra se o medicamento proposto está legalizado. Essa dupla validação deve acontecer pelo motivo de que na maioria das vezes o médico prescreve apenas o composto e dosagem do medicamento, enquanto na farmácia ocorre algo mais específico que é a compra de determinado medicamento produzido por determinada empresa, essa dupla validação faz com que o composto prescrito e o lote e/ou fabricante sejam todos verificados. Outra parte das validações são os controles de estabelecimentos médicos, cruzando informações do Cadastro Nacional de Estabelecimentos de Saúde (CNES), é possível permitir ou não agendamentos de consultas em determinada clínica ou hospital.

Ainda para o cidadão, a API pode fornecer diversas vantagens logo no início, como a possibilidade de visualizar o histórico de consultas e medicamentos que ele teve durante sua vida, além de poder ter a opção de ter a compra de seus medicamentos fracionada, visto que alguns medicamentos são caros, outra possibilidade seria de ao acessar o aplicativo Gov.br, ele ter a opção de permitir que outra pessoa realize a compra do medicamento para ele, ou seja, a receita não estaria mais vinculada à um simples pedaço de papel com um carimbo que pode facilmente ser alterado, a receita seria algo digital, que seria validado em banco de dados, tornando o processo de fraudes mais difícil. Além de inibir a fraude de compra de medicamentos, outra vantagem seria fiscalizar a venda de medicamentos pirateados e contrabandeados, já que toda venda de medicamento será persistida no banco de dados as informações de quem prescreveu a medicação, hospital em que o profissional atua, cidadão que realizou a compra, farmacêutico que realizou a venda, a farmácia onde foi realizada a compra, além de informações do medicamento. Todas

essas informações fornece um rastro de todos os envolvidos nesse processo. Além de ocorrerem as validações no momento da compra, como por exemplo, se a receita é válida, verificando se o médico existe e pode atuar, se a data da receita não está expirada, ou se o saldo para compra do medicamento já foi atingido, se a pessoa que está indo realizar a compra é a pessoa descrita na receita, ou se foi destinada pelo cidadão dono da prescrição médica, todo esse sistema pode ainda ser melhorado no futuro, como por exemplo a utilização de blockchain para melhorar ainda mais a eficiência da segurança das informações.

3.5 Arquitetura da API

Apesar de não ter sido estimada a quantidade de requisições por segundo que a API terá, é notório que serão milhares por segundo, já que a todo tempo validará um médico, ou o acesso de um cidadão a sua conta, a venda de um medicamento etc. Tendo em mente esse cenário, foi feita a escolha da utilização da arquitetura em microserviços, dando uma flexibilidade maior para a implementação da infraestrutura. Segundo enfatiza Martin Fowler (2014) em seu artigo "*Microservices - a definition of this new architectural term*", esse tipo de arquitetura são pequenos serviços autônomos, que são implementados de maneira independente, por isso a sua capacidade de escalabilidade. Além de ter diversas vantagens sobre uma API monolítica em se tratando de um sistema desse porte, que cobrirá um país com mais de 230 milhões de habitantes, como a escalabilidade e dimensionamento mais preciso das máquinas que servirão o sistema, adicionar novos módulos futuros conforme necessidade, sem a necessidade de grandes modificações nos demais módulos entre outros.

3.6 Dificuldades da implementação

Todo projeto é criado com a finalidade de fornecer soluções para algum tipo de problema, o projeto Trackmed é algo ambicioso que tem como intenção atender uma

das maiores populações do planeta, sendo uma solução única que até a presente data não foi possível identificar um país que utilize algo similar ao que é proposto com esse projeto. Então, os desafios são os mais diversos possíveis, iniciando pela forte dependência com informações dos órgãos governamentais, esse é o motivo principal de o Trackmed ser apenas um conceito que deve ser cedido ao governo, e não uma propriedade intelectual, com a finalidade de monetizar com ele, embora a possibilidade de monetização exista, mas através da gestão da base de dados ou a venda de dados para laboratórios farmacêuticos ou pesquisas mediante acordos, mas esse não é o foco do projeto.

Outro fator crucial para o funcionamento da API, são as melhorias necessárias nas bases de dados dos órgãos do governo que farão parte do sistema, hoje os servidores dessas bases de dados são dimensionados para atenderem a um número muito inferior de requisições do que a API Trackmed precisa para seu funcionamento, se tornando um gargalo que tornaria inviável o projeto. Além é claro, da burocracia que o projeto precisa enfrentar para se tornar uma legislação e das votações dos recursos que devem ser destinados para a realização do projeto. Mesmo enfrentando todos esses desafios que fogem da alçada dos desenvolvedores, ainda sim é viável a execução do projeto, inclusive no quesito aprovação na câmara e senado, já que como sugerido pelas pesquisas, com a digitalização do SUS, o governo economizaria muito dinheiro, isso sem levar em consideração os ganhos de receita provenientes com a redução de medicamentos contrabandeados, melhor gestão de recursos empregados, e fiscalização mais contundente.

3.7 Softwares Similares

Atualmente existem alguns aplicativos privados e governamentais que auxiliam na gestão dos dados do paciente e indicando as especialidades e médicos para tratamento, foram selecionadas às três aplicações mais populares e mais bem avaliadas pelos usuários da Play Store, são eles:

- **ConectSUS:** O aplicativo registra todo o histórico de atendimento de uma pessoa no Sistema Único de Saúde (SUS). Esse aplicativo disponibiliza informações sobre consultas médicas, internações hospitalares, medicamentos

prescritos, exames realizados, além de permitir o agendamento de consultas na rede pública de saúde. Aplicação com nota 4,6 na Google Play (GOOGLE PLAY, 2023a).

- **Unimed Cliente** aplicativo que reúne as soluções para o beneficiário, como número da carteirinha para atendimento e médicos mais próximos, para facilitar. Aplicação com nota 4,4 na Google Play (GOOGLE PLAY, 2023b).
- **Bradesco Saúde:** aplicação com o foco voltado para o agendamento de consultas e identificação do paciente, buscando por geolocalização o atendimento mais próximo, facilitando a localização de pronto socorro e farmácias. Aplicação com nota 4,3 na Google Play (GOOGLE PLAY, 2023c).

Levando estes aspectos em consideração, foi elaborada a Tabela 1 mostrando as principais diferenças do TrackMed, em relação aos aplicativos citados anteriormente:

Tabela 1 - Comparativo de funcionalidades da aplicação TrackMed em relação aos concorrentes.

| Funcionalidades | ConectSUS | Unimed Cliente | Bradesco | TrackMed |
|---|------------------|-----------------------|-----------------|-----------------|
| Gerenciar consultas no hospital | x | x | x | x |
| Gerenciar receituário médico | | | | x |
| Permissões para terceiros comprar medicamento | | | | x |
| Verificação medicamento junto com ANVISA | | | | x |
| Verificação de identificação CRM | | | | x |
| Verificação de identificação CRF | | | | x |
| Geração de dados para pesquisas | | | | x |

Fonte: Elaborado pelo autor (2023).

3.8 Levantamento de Requisitos

Na etapa de levantamento de requisitos, foram analisadas quais seriam as reais necessidades que a API deveria atender, quais as validações e restrições o sistema deveria verificar, sendo possível entender o que os usuários esperariam do sistema, analisar como o sistema poderia aumentar a produtividade dos profissionais, sem

aumentar as atribuições para os mesmos. Como é o caso da recepcionista, que ela criaria um cadastro básico para um paciente caso ele ainda não possuía, foi pensado para que o sistema em si criasse de maneira automática esse cadastro sem que a recepcionista ao menos percebesse que ela estaria realizando esse cadastro básico.

3.8.1 Requisitos Funcionais

Para os requisitos funcionais foram levantadas questões que seriam essenciais para o funcionamento do sistema, como a definição do nome de usuário como sendo a chave única de cada usuário dentro do sistema, considerando que profissionais acessarão o sistema tanto através do seu respectivo cargo, ou até mesmo em diversas ocupações que venha ter, e permitir que esse mesmo usuário também possa acessar como um cidadão com o intuito de verificar suas informações na base de dados. Os requisitos funcionais levantados foram os descritos na tabela 2.

Tabela 2 – Requisitos funcionais do projeto.

| Identificação | Requisito Funcional | Prioridade |
|---------------|---|------------|
| RF001 | O sistema deve gerar um ID único para cada usuário | Essencial |
| RF002 | O usuário médico deverá ter acesso para fazer login como médico, mas também deve ter o acesso usuário cidadão, para ser atendido. | Essencial |
| RF003 | O usuário médico terá acesso com seu nome de usuário único a “N” clínicas ou hospitais, pois poderá fazer atendimento em locais diferentes. | Essencial |
| RF004 | A partir do momento que o usuário médico logar no sistema, ele não poderá logar em outra plataforma, evitando múltipla estância. | Essencial |
| RF005 | O sistema deverá ter acesso aos órgãos reguladores como Conselho Regional de Medicina, Conselho Regional de Enfermagem, Conselho Regional de Farmácia, Anvisa etc. | Importante |
| RF006 | O sistema deverá permitir Integração ao DATASUS e toda necessidade do gov.br para dados | Essencial |
| RF007 | Todo o histórico do usuário cidadão referente a consultas e receituário médico, deverá ser armazenado. | Importante |
| RF008 | Quando houver um tratamento a longo prazo em seu receituário, este deve permitir armazenagem e disponibilização fracionada junto a farmácia para o usuário cidadão. | Importante |
| RF009 | O usuário paciente, deverá ter acesso ilimitado a consulta | Essencial |
| RF010 | Na contratação e para emissão de receitas o usuário médico deverá ser validado na base de dados do Conselho Regional de Medicina | Essencial |
| RF011 | Na contratação e para efetuar a venda o usuário farmácia deverá ser validado na base de dados do Conselho Regional de Medicina | Essencial |
| RF012 | Os remédios serão validados junto a ANVISA | Essencial |

Fonte: Elaborado pelo autor.

3.8.2 Requisitos Não Funcionais

A parte que faz com que a API ganhe sua devida importância, deve-se também pela sua capacidade de integração e disponibilidade de uso, o sistema em si tem que estar disponível 24 horas por dia e 7 dias na semana, justamente para garantir a precisão dos dados coletados, úteis para pesquisas futuras, além do fato de ser capaz de atender todas as requisições diárias, por isso a escolha da arquitetura em micro serviços, que pode ser escalável conforme demanda.

Tabela 3 – Requisitos não funcionais do projeto.

| Identificação | Requisito não funcional | Categoria | Prioridade |
|---------------|---|----------------|------------|
| RNF001 | O sistema deverá ter disponibilidade de 99,98% | Confiabilidade | Essencial |
| RNF002 | Deve atender as todas as integrações como: Conselho Regional de Medicina, Conselho Regional de Farmácia, Anvisa, Data SUS, com estabilidade | Desempenho | Essencial |
| RNF003 | Deve atender as necessidades da LGPD | Segurança | Essencial |
| RNF004 | O sistema deverá ser desenvolvido utilizando arquitetura distribuída | Padrões | Essencial |

Fonte: Elaborado pelo autor.

3.9 Recursos e Ferramentas

Esta seção contempla as ferramentas de programação e os conceitos necessários para o desenvolvimento do sistema:

- **Java:** É uma tecnologia utilizada para desenvolvimento de aplicações, que podem ser *Web*, *Desktop* ou até mesmo *mobile*, sendo orientada a objetos, compilada e interpretada. Atualmente o Java é uma das tecnologias mais utilizadas do mundo, que de acordo com a Oracle, cerca de 97% dos *Desktops* executam Java e 89% dos *desktops* nos Estados Unidos também o executa. Esses números se dão ao fato de que essa tecnologia foi projetada para permitir o desenvolvimento de aplicativos portáteis de alto desempenho e para que se possa abranger todas as plataformas possíveis (ORACLE, 2018).
- **SpringBoot:** É um *framework* de desenvolvimento de aplicativos Java que oferece uma abordagem rápida e fácil para criar aplicativos autônomos e de produção pronta. Ele utiliza o Spring Framework como base e fornece uma configuração pré-definida para muitos aspectos comuns do desenvolvimento, incluindo configuração de banco de dados, gerenciamento de dependências e configuração de segurança. Com o Spring Boot, os desenvolvedores podem criar aplicativos com rapidez e eficiência, sem ter que se preocupar com a configuração complexa e tediosa que geralmente vem com o desenvolvimento de aplicativos Java tradicionais.

- **Docker:** É uma plataforma de virtualização de aplicativos que permite que os desenvolvedores criem, implantem e executem aplicativos em contêineres virtualizados. Esses contêineres são isolados uns dos outros e do sistema operacional hospedeiro, permitindo que os aplicativos sejam executados em qualquer ambiente de forma consistente, independentemente das diferenças de infraestrutura subjacente. Isso torna o desenvolvimento, teste e implantação de aplicativos mais rápidos, flexíveis e portáteis.
- **Kubernetes:** É uma plataforma de código aberto que permite a automação, o gerenciamento e a orquestração de contêineres de aplicativos em ambientes de computação em nuvem. Ele foi projetado para simplificar o processo de implantação, dimensionamento e gerenciamento de aplicativos em contêineres, fornecendo uma estrutura para a implementação de contêineres de forma escalável, flexível e confiável, independentemente do local em que estejam sendo executados. Com Kubernetes, os desenvolvedores podem se concentrar no desenvolvimento de aplicativos, enquanto os administradores de sistemas podem se concentrar em gerenciar a infraestrutura subjacente.
- **Tomcat H2:** É uma combinação do servidor web Apache Tomcat com o banco de dados relacional H2. Ele é uma implementação de um servidor de aplicativos Java EE que utiliza o Tomcat como seu container de servlets e JSPs, e o H2 como seu banco de dados. O Tomcat H2 oferece aos desenvolvedores uma solução de banco de dados leve e rápida para suas aplicações web, permitindo que eles criem e implantem facilmente aplicativos web escaláveis e de alta performance.
- **PostgreSQL:** É um sistema de gerenciamento de banco de dados relacional de código aberto e gratuito, desenvolvido para ser altamente extensível, confiável e compatível com padrões de SQL. O PostgreSQL é amplamente considerado um dos bancos de dados mais poderosos e avançados disponíveis, com recursos avançados como suporte a transações, integridade referencial, replicação, particionamento, procedimentos armazenados e muito mais. É utilizado em aplicações de todos os tamanhos, desde pequenos projetos até grandes empresas e governos.
- **RabbitMQ:** É um software de mensagem de código aberto que atua como um intermediário de mensagens entre diferentes sistemas, permitindo que

aplicativos se comuniquem e compartilhem informações de forma confiável e escalável. Ele implementa o protocolo de fila de mensagens AMQP (Advanced Message Queuing Protocol), que fornece uma plataforma robusta e flexível para troca de dados entre aplicativos distribuídos em uma arquitetura de micro serviços ou em um ambiente de processamento assíncrono. RabbitMQ é amplamente utilizado em empresas e organizações para construir soluções de integração de sistemas e para melhorar a escalabilidade e a confiabilidade de aplicativos distribuídos.

4 MODELAGEM

Etapa onde é estruturado ou agrupado os dados, dando significado a eles, Bezerra (2014, p. 3) diz que modelo se baseia no princípio da abstração, onde apenas características necessárias para a solução de um problema são consideradas.

Na fase da modelagem foi possível observar quais as melhores abordagens para estruturar a aplicação, e com isso descobrir quais relacionamentos entre entidades são necessários, atributos e funcionalidades antes não pensados, ou remoção de dados não necessários para o perfeito funcionamento.

4.1 Casos De Uso

A etapa de criação dos diagramas de casos de uso foi a etapa onde mais foi observado a necessidade de alterações no projeto, nessa etapa o comportamento da API foi feito e refeito, sempre que uma nova abordagem surgia, mas foi uma etapa essencial para que fosse possível compreender com maior profundidade o lado do usuário final, aquele que estaria utilizando o sistema e suas implementações, para BEZERRA (2014, p. 54) “Um caso de uso representa um relato de uso de certa funcionalidade do sistema em questão, sem revelar a estrutura e o comportamento internos desse sistema”.

Os atores que interagem com o sistema são: o Cidadão, Recepcionista, Médico, Hospital, Farmacêutico, Farmácia, API do Conselho Regional de Medicina, API do Conselho Regional de Farmácia, API da Anvisa e o GOV representado todas as integrações do sistema. O sistema é um caso de uso explícito e se trata do sistema em si em que os casos de uso acontecem.

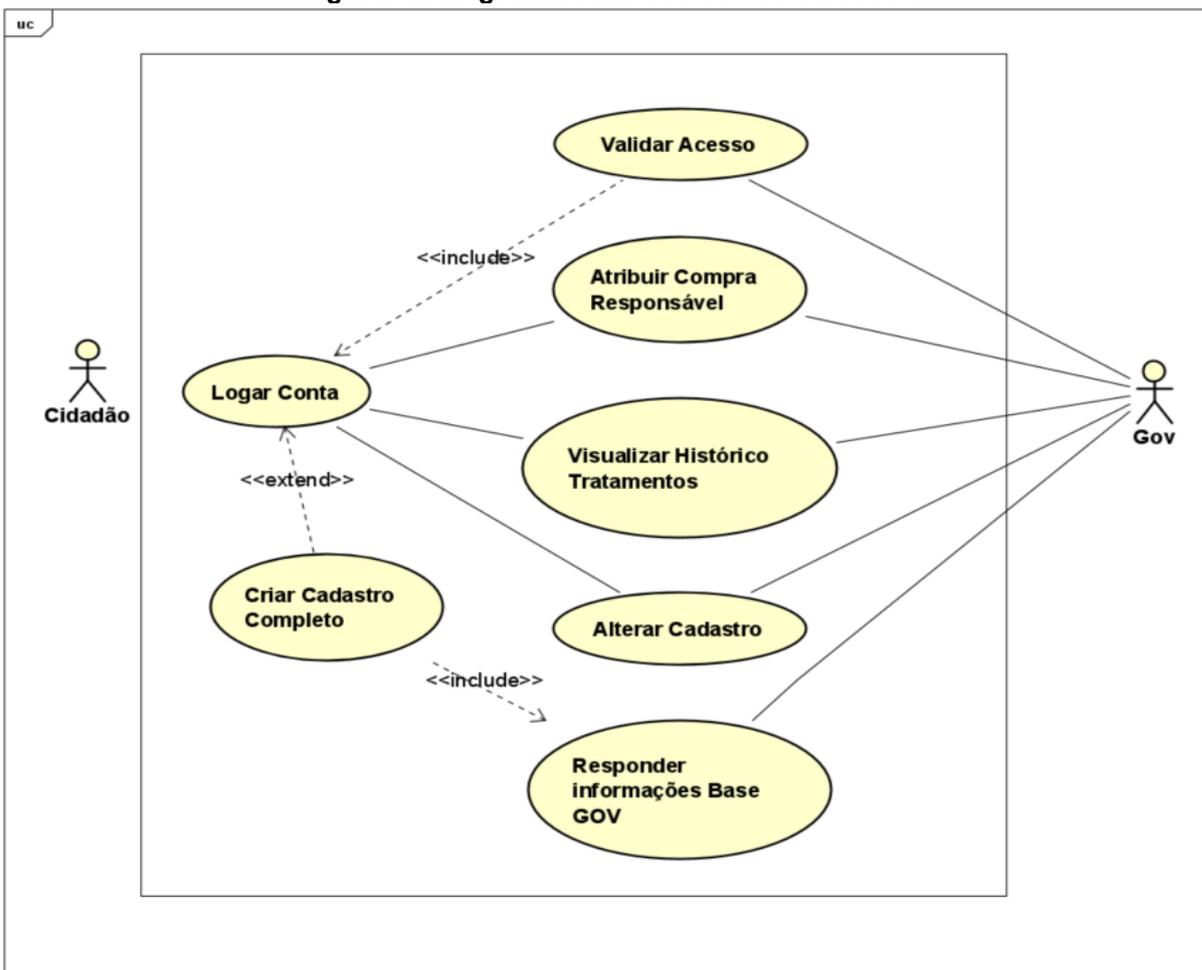
- Cidadão: é o ator que representa os pacientes ou representantes deste aplicativo. Um cidadão pode agendar uma consulta, visualizar seu histórico de tratamentos, visualizar suas consultas agendadas e atribuir a compra de seu medicamento para um terceiro.
- Recepcionista: é o ator responsável por criar o cadastro básico do cidadão caso ele ainda não tenha um cadastro na base de dados Trackmed, agendar uma

consulta “presencialmente” e libera a visualização e edição (apenas inserção) do histórico do paciente.

- Médico: é o ator que insere novos tratamentos médicos no registro do cidadão e visualizar o histórico do paciente, porém com restrições.
- CRM: é o ator que representa o Conselho Regional de Medicina, validando em cada requisição se o médico que está para inserir um novo tratamento no registro do paciente, está com seu registro ativo no órgão e que suas especialidades condizem com a necessidade do paciente.
- Hospital: é o ator responsável por sediar as consultas do paciente, onde ficam os registros daquela consulta específica.
- Anvisa: é o ator responsável por validar se um determinado medicamento ou marca está liberado para a venda.
- Gov: é o ator que gerencia, coleta e mantém toda a base de dados entre as requisições.
- Farmacêutico: é o ator que realiza a venda dos medicamentos do paciente conforme especificações, esse ator é responsável pela atualização de saldos e status nos tratamentos do cidadão.
- Farmácia: é o ator responsável por sediar a venda dos medicamentos.
- CRF: é o ator que valida se o farmacêutico está com seu registro válido.

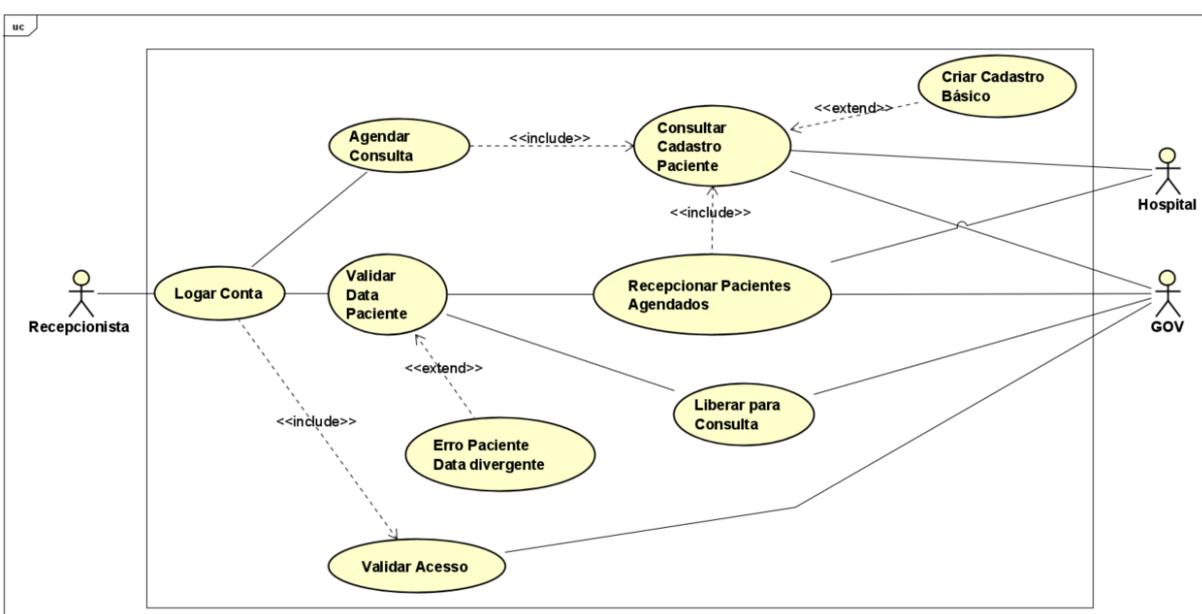
Nas figuras de 1 a 4 é possível visualizar o processamento de cada etapa que envolvem os usuários do sistema.

Figura 1 – Diagrama de caso de uso do Cidadão



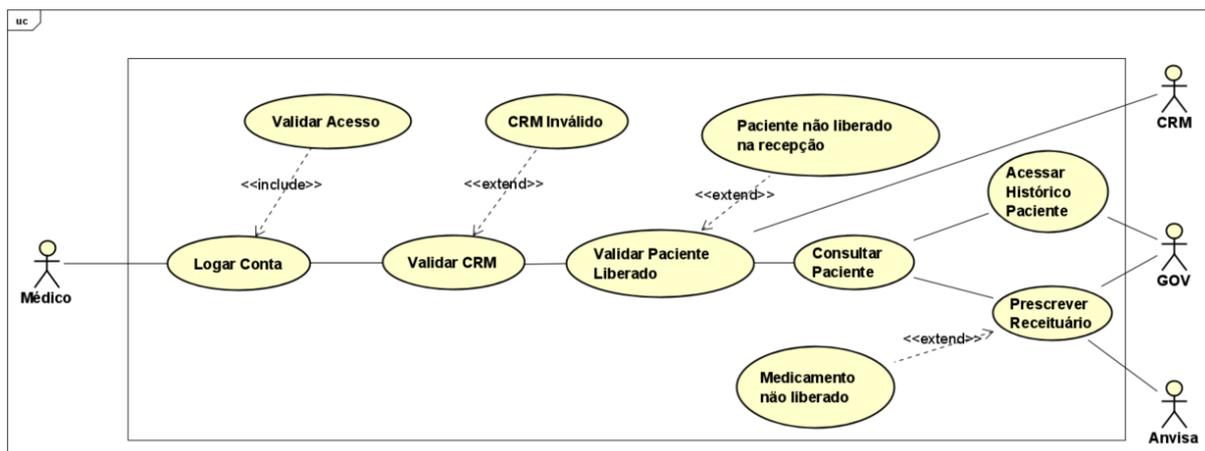
Fonte: Elaborado pelo autor.

Figura 2 - Diagrama de caso de uso da Recepcionista



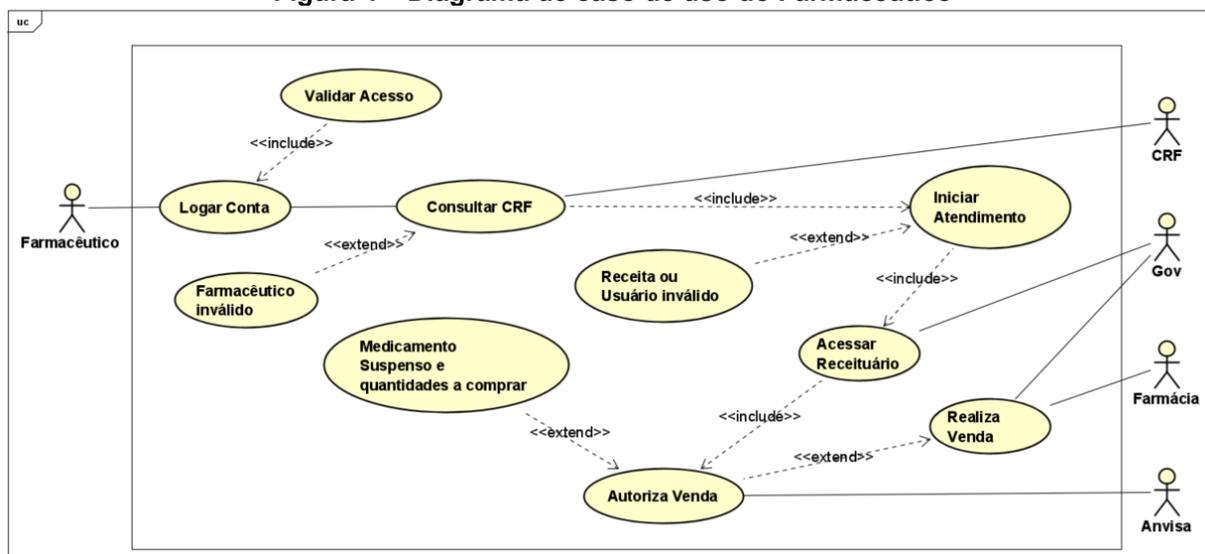
Fonte: Elaborado pelo autor.

Figura 3 - Diagrama de caso de uso do Médico



Fonte: Elaborado pelo autor.

Figura 4 – Diagrama de caso de uso do Farmacêutico



Fonte: Elaborado pelo autor.

4.2 Documentação dos Casos de Uso

4.2.1 Casos de Uso Microserviço Citizen

Tabela 4 – Caso de uso “Entrar no Sistema (Cidadão)”

| | |
|--|--|
| Nome do caso de uso | Entrar no sistema |
| Atores envolvidos | Cidadão (usuário), Gov. |
| Objetivo | Este caso de uso descreve os passos do login de um usuário no sistema |
| Ações do ator | Ações do sistema |
| 1. O usuário acessa Conta, passando nome de usuário e senha como parâmetros. | |
| | 2. O sistema verifica se o usuário possui cadastro completo no sistema, caso não possua, solicitará a criação do cadastro. |
| | 3. O sistema retorna um token JWT. |
| Validações | |
| | 1. O sistema valida o nome de usuário e senha. |

Fonte: Elaborado pelo autor.

Tabela 5 - Caso de uso “Criar Cadastro Completo”

| | |
|--|---|
| Nome do caso de uso | Cadastrar completamente no sistema |
| Atores envolvidos | Cidadão (usuário), Gov. |
| Objetivo | Este caso de uso descreve os passos do cadastro completo de um usuário no sistema |
| Ações do ator | Ações do sistema |
| 1. O usuário acessa Criar Conta, passando cpf como parâmetro | |
| | 2. O sistema busca informações sigilosas do usuário já contidas na base GOV. Com as informações, o sistema retorna uma série de perguntas a serem respondidas para validar se realmente se trata do responsável do cpf informado. |
| 3. Passando como parâmetro o cpf, o usuário envia as respostas no corpo da requisição. | |
| | 4. Após validações e confirmado, o sistema solicita um nome de usuário (username). |
| 5. O usuário informa passa o username, como parâmetro. | |
| | 6. O sistema válida os dados e faz a persistência na base do Gov. |
| Validações | |
| | 1. Validação de cpf. |
| | 2. Caso as respostas informadas não sejam corretas, o tempo para poder iniciar nova série de novas perguntas é incrementado exponencialmente. |
| | 3. Caso o usuário informe um username que já tenha sido registrado, o sistema solicitará um novo username. |

Fonte: Elaborado pelo autor.

Tabela 6 - Caso de uso “Visualizar histórico de tratamentos”

| | |
|--|--|
| Nome do caso de uso | Visualizar histórico de tratamentos |
| Atores envolvidos | Cidadão (usuário), Gov. |
| Objetivo | Este caso de uso descreve os passos para visualização do histórico de tratamentos do usuário |
| Ações do ator | Ações do sistema |
| 1. O usuário acessa Histórico, passando o username como parâmetro. | |
| | 2. O sistema retorna as informações do histórico de tratamentos do usuário. |
| Validações | |
| | 1. Para visualizar o histórico é necessário que o usuário esteja logado em sua conta no sistema. |

Fonte: Elaborado pelo autor.

Tabela 7 - Caso de uso “Atribuir compra de medicamentos para responsável”

| | |
|---|--|
| Nome do caso de uso | Atribuir compra de medicamentos para responsável |
| Atores envolvidos | Cidadão (usuário), Cidadão (Responsável), Gov. |
| Objetivo | Este caso de uso descreve os passos para atribuição de compra de medicamento para um responsável. |
| Ações do ator | Ações do sistema |
| 1. O usuário acessa Atribuir Compra, passando o usuário proprietário da conta, e o cpf da pessoa a ser atribuída, e lista dos tratamentos que serão atribuídos. | |
| | 2. O sistema atribui os tratamentos para o responsável informado e retorna a lista dos tratamentos atualizada. |
| Validações | |
| | 1. O sistema valida se os tratamentos solicitados estão dentro da validade prescrita e se ainda tem saldo para compra. |
| | 2. O sistema valida o cpf do responsável atribuído, e se o mesmo é maior de 16 anos. |

Fonte: Elaborado pelo autor.

Tabela 8 - Caso de uso “Alterar dados no sistema”

| | |
|--|--|
| Nome do caso de uso | Alterar dados no sistema |
| Atores envolvidos | Cidadão (usuário), Gov. |
| Objetivo | Este caso de uso descreve os passos para um usuário alterar suas informações no sistema |
| Ações do ator | Ações do sistema |
| 1. O usuário acessa Alterar Conta, passando como parâmetro seu username ou cpf, e no corpo da requisição um objeto DAO com as informações da classe Citizen, apenas com os atributos a serem alteradas, como Nome, Endereço, Responsável Atribuído, Email e Senha. | |
| | 2. O sistema valida as informações e persiste os dados. |
| Validações | |
| | 1. O sistema faz as validações dos dados como, e-mail, cep inválido e exigência mínima para o padrão de senha etc. |

Fonte: Elaborado pelo autor.

4.2.2 Casos de Uso Microserviço Hospital

Tabela 9 - Caso de uso “Entrar no sistema (Recepcionista)”

| | |
|----------------------------|--|
| Nome do caso de uso | Entrar no sistema |
| Atores envolvidos | Recepcionista (usuário), Gov., Hospital |
| Objetivo | Este caso de uso descreve os passos do login de uma recepcionista no sistema |
| Ações do ator | Ações do sistema |

| | |
|--|---|
| 1. O recepcionista acessa Conta, passando como parâmetros nome de usuário, senha e hospital. | |
| | 2. Após as validações e confirmações das informações inseridas pela recepcionista, é liberado o acesso. |
| | 3. O sistema retorna um token JWT. |
| Validações | |
| | 1. O sistema valida o nome de usuário, senha e hospital. |

Fonte: Elaborado pelo autor.

Tabela 10 - Caso de uso “Agendar Consulta”

| | |
|--|---|
| Nome do caso de uso | Agendar Consulta |
| Atores envolvidos | Recepcionista (usuário), Gov., Hospital |
| Objetivo | Este caso de uso descreve os passos para uma recepcionista agendar um paciente |
| Ações do ator | Ações do sistema |
| 1. A recepcionista acessa Agendar, passando como parâmetros cpf do paciente, cartão do sus do paciente e código do hospital. E no corpo da requisição as informações da classe Appointment | |
| | 2. Caso o paciente não tenha um cadastro na base de dados, é criado um cadastro básico com o nome, cpf e número cartão sus do paciente. Caso o paciente já tenha ao menos um cadastro básico, é realizado o agendamento conforme disponibilidade do hospital. |
| Validações | |
| | 1. Para acessar Agendar Consulta é necessário que a recepcionista esteja logado em sua conta no sistema. |
| | 2. Valida se o hospital possui a especialidade para o atendimento. |

Fonte: Elaborado pelo autor.

Tabela 11 - Caso de uso “Recepcionar Paciente Agendado”

| | |
|--|---|
| Nome do caso de uso | Recepcionar Paciente Agendado |
| Atores envolvidos | Recepcionista (usuário), Gov., Hospital |
| Objetivo | Este caso de uso descreve os passos para uma recepcionista recepcionar um paciente agendado. |
| Ações do ator | Ações do sistema |
| 1. A recepcionista acessa Recepcionar, passando como parâmetros cpf do paciente, cartão do sus do paciente e código do hospital. | |
| | 2. O sistema verifica no banco de dados as informações do paciente e horários disponibilizados pelo hospital, e persiste a entrada do paciente no hospital. |
| Validações | |
| | 1. Para acessar Recepcionar é necessário que a recepcionista esteja logado em sua conta no sistema. |
| | 2. O sistema valida a data, hora, hospital e especialidade da consulta do paciente. |

Fonte: Elaborado pelo autor.

Tabela 12 - Caso de uso “Liberar Paciente para Consulta”.

| | |
|--|--|
| Nome do caso de uso | Liberar Paciente para Consulta |
| Atores envolvidos | Recepcionista (usuário), Gov., Hospital |
| Objetivo | Este caso de uso descreve os passos para uma recepcionista liberar o paciente para a consulta. |
| Ações do ator | Ações do sistema |
| 1. A recepcionista acessa Liberar Paciente, passando como parâmetros cpf do paciente e código do hospital. | |
| | 2. O sistema verifica no banco de dados as informações do paciente e horários disponibilizados pelo hospital, libera a visualização do histórico de consultas e permissão para o médico adicionar novos tratamentos. |
| Validações | |
| | 1. Para acessar Liberar Paciente é necessário que a recepcionista esteja logado em sua conta no sistema. |
| | 2. O sistema valida a data, hora, hospital e especialidade da consulta do paciente. |

Fonte: Elaborado pelo autor.

Tabela 13 - Caso de uso “Entrar no sistema (Médico)”

| | |
|---|---|
| Nome do caso de uso | Entrar no sistema |
| Atores envolvidos | Médico (usuário), Gov., CRM, Hospital |
| Objetivo | Este caso de uso descreve os passos do login de um médico no sistema |
| Ações do ator | Ações do sistema |
| 1. O médico acessa Conta, passando como parâmetros nome de usuário, senha e hospital. | |
| | 2. O sistema busca as informações do médico na base de dados e válida seu registro e especialidade no CRM, e libera o acesso a sua conta. |
| | 3. O sistema retorna um token JWT. |
| Validações | |
| | 1. O sistema valida o nome de usuário, senha e hospital. |
| | 2. O sistema verifica a situação de seu registro junto ao CRM. |

Fonte: Elaborado pelo autor.

Tabela 14 - Caso de uso “Consultar Paciente”

| | |
|--|--|
| Nome do caso de uso | Consultar Paciente |
| Atores envolvidos | Médico (usuário), Gov., Hospital |
| Objetivo | Este caso de uso descreve os passos para o médico consultar um paciente |
| Ações do ator | Ações do sistema |
| 1. O médico acessa Consultar, passando como parâmetros cpf do paciente e código do hospital. | |
| | 2. O sistema valida os dados do paciente, e registra atendimento no histórico do paciente. |
| Validações | |
| | 1. Para acessar Consultar é necessário que o médico esteja logado em sua conta no sistema. |
| | 2. O sistema verifica se o paciente foi liberado pela recepção. |
| | 3. O sistema verifica se o médico é apto a exercer a especialidade da consulta. |

Fonte: Elaborado pelo autor.

Tabela 15 - Caso de uso “Visualizar Histórico do Paciente”

| | |
|---|--|
| Nome do caso de uso | Visualizar Histórico do Paciente |
| Atores envolvidos | Médico (usuário), Gov. Hospital |
| Objetivo | Este caso de uso descreve os passos para o médico consultar um paciente |
| Ações do ator | Ações do sistema |
| 1. O médico acessa Consultar Histórico Paciente, passando como parâmetros cpf do paciente e código do hospital. | |
| | 2. O sistema valida os dados do paciente, e exibe as consultas e tratamentos do paciente. |
| Validações | |
| | 1. Para acessar Consultar Histórico é necessário que o médico esteja logado em sua conta no sistema. |
| | 2. O sistema verifica se o paciente foi liberado pela recepção. |

Fonte: Elaborado pelo autor.

Tabela 16 - Caso de uso “Prescrever Receituário”

| | |
|---|---|
| Nome do caso de uso | Prescrever Receituário |
| Atores envolvidos | Médico (usuário), Gov., Anvisa, Hospital |
| Objetivo | Este caso de uso descreve os passos para o médico prescrever um tratamento para o paciente. |
| Ações do ator | Ações do sistema |
| 1. O médico acessa Prescrever, passando como parâmetros cpf do paciente, código do hospital e no corpo da requisição as informações da classe Treatment | |
| | 2. O sistema consulta se existem restrições na Anvisa para os medicamentos, e realiza a persistência dos mesmos. Caso não ocorra erros durante esse processo, o sistema encerra a visibilidade do histórico do paciente e possibilidades de adição de novos tratamentos no histórico do paciente. |
| Validações | |
| | 1. Para acessar Prescrever é necessário que o médico esteja logado em sua conta no sistema. |
| | 2. O sistema verifica se o paciente foi liberado pela recepção. |
| | 3. O sistema verifica na Anvisa se não existem restrições dos medicamentos. |
| | 4. O sistema valida os dados do tratamento. |

Fonte: Elaborado pelo autor.

4.2.3 Casos de Uso Microserviço Pharmacy

Tabela 17 - Caso de uso “Entrar no sistema (Farmacêutico)”

| | |
|---|---|
| Nome do caso de uso | Entrar no sistema |
| Atores envolvidos | Farmacêutico (usuário), Gov., CRF, Farmácia |
| Objetivo | Este caso de uso descreve os passos do login de um médico no sistema |
| Ações do ator | Ações do sistema |
| 1. O farmacêutico acessa Conta, passando como parâmetros nome de usuário, senha e farmácia. | |
| | 2. O sistema busca as informações do farmacêutico na base de dados e válida seu registro e especialidade no CRF, e libera o acesso a sua conta. |
| Validações | |
| | 1. O sistema valida o nome de usuário, senha e farmácia. |
| | 2. O sistema verifica a situação de seu registro junto ao CRF. |

Fonte: Elaborado pelo autor.

Tabela 18 - Caso de uso “Iniciar Atendimento”

| | |
|--|--|
| Nome do caso de uso | Iniciar Atendimento |
| Atores envolvidos | Farmacêutico (usuário), Gov., Farmácia |
| Objetivo | Este caso de uso descreve os passos do login de um médico no sistema |
| Ações do ator | Ações do sistema |
| 1. O farmacêutico acessa Iniciar Atendimento, passando como parâmetros cpf do cidadão ou responsável atribuído e o código do tratamento. | |
| | 2. O sistema valida os dados do cidadão ou responsável atribuído e retorna a lista dos medicamentos da prescrição. |
| Validações | |
| | 1. Para acessar Iniciar Atendimento é necessário que o farmacêutico esteja logado em sua conta no sistema. |
| | 2. O sistema verifica se o cidadão ou responsável atribuído tem permissão para a compra do medicamento. |
| | 3. Valida dados da prescrição como, data, e saldo do medicamento para compra. |

Fonte: Elaborado pelo autor.

Tabela 19 - Caso de uso “Vender Medicamento”

| | |
|--|--|
| Nome do caso de uso | Vender Medicamento |
| Atores envolvidos | Farmacêutico (usuário), Gov., Farmácia, Anvisa |
| Objetivo | Este caso de uso descreve os passos do login de um médico no sistema |
| Ações do ator | Ações do sistema |
| 1. O farmacêutico acessa Vender, passando como parâmetros, cpf do cidadão ou responsável atribuído e no corpo da requisição as informações da classe TreatmentShopping | |
| | 2. O sistema verifica os dados dos medicamentos junto a Anvisa e a quantidade no Gov., e realiza a persistência. |
| Validações | |
| | 1. Para acessar Iniciar Atendimento é necessário que o farmacêutico esteja logado em sua conta no sistema. |
| | 2. O sistema válida na Anvisa se os medicamentos estão liberados para venda. |
| | 3. O sistema valida no Gov. se a quantidade informada está de acordo com o saldo da prescrição. |

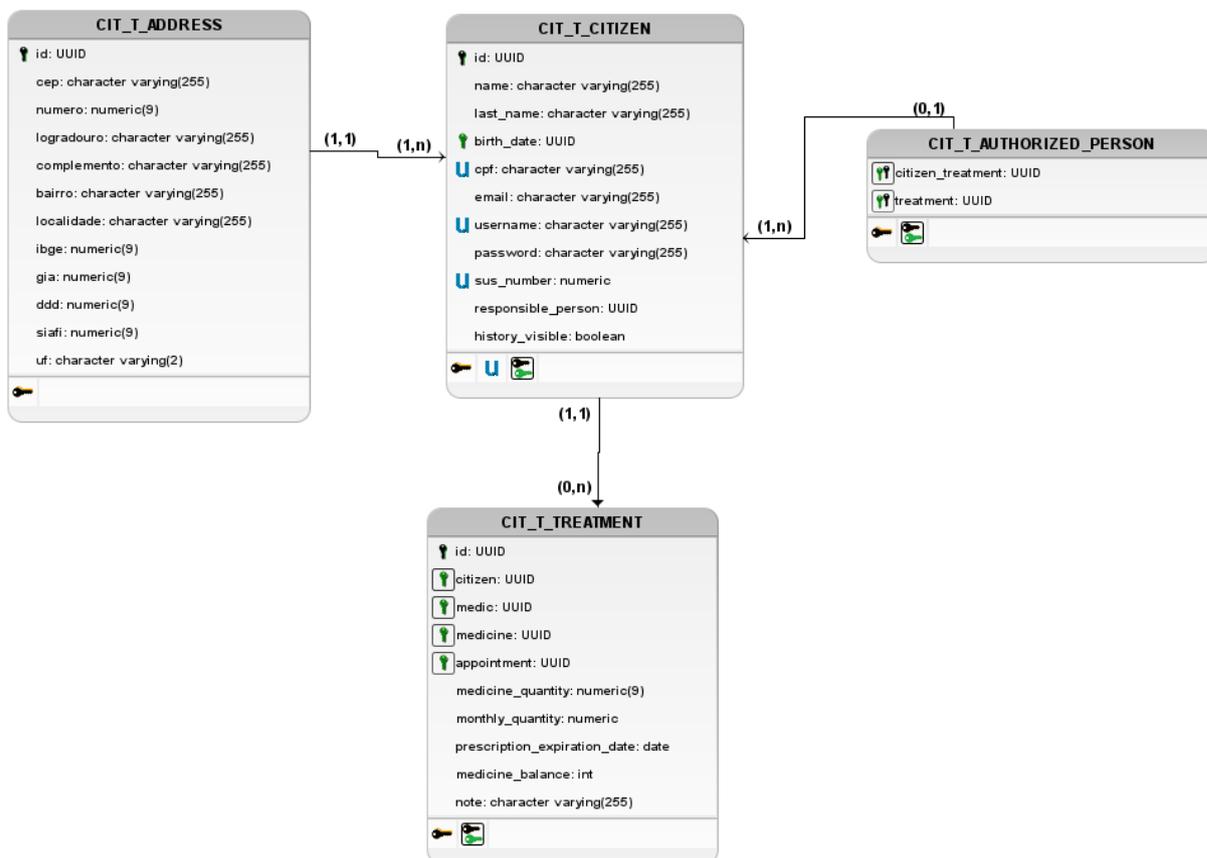
Fonte: Elaborado pelo autor.

4.4 Diagrama de Entidade e Relacionamento

A etapa de mapeamento dos campos de cada tabela é algo fundamental para a armazenagem dos dados e compreender os relacionamentos entre cada entidade, tornando possível entender quando os relacionamentos entre tabelas devem acontecer. Por se tratar de uma arquitetura de software baseado em micro serviços, essa etapa trouxe algumas novas oportunidades de conhecimento, como a atribuição do *Universally Unique Identifier* (UUID) como chave primária para as tabelas. Durante os mapeamentos das entidades foi possível observar a disposição dos dados de cada

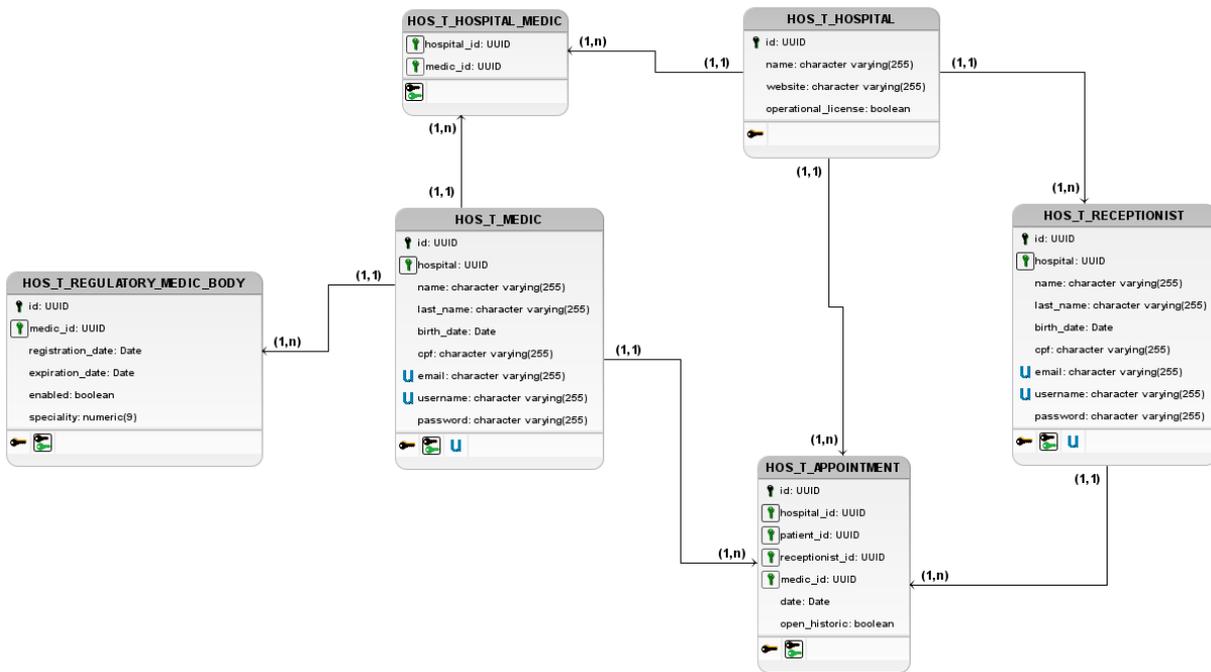
processo executado do sistema, podendo assim adequar para melhor atender um grupo de dados específico. Abaixo segue a diagramação das tabelas de cada micro serviço.

Figura 5 - Diagrama de Entidade e Relacionamento Micro Serviço Cidadão



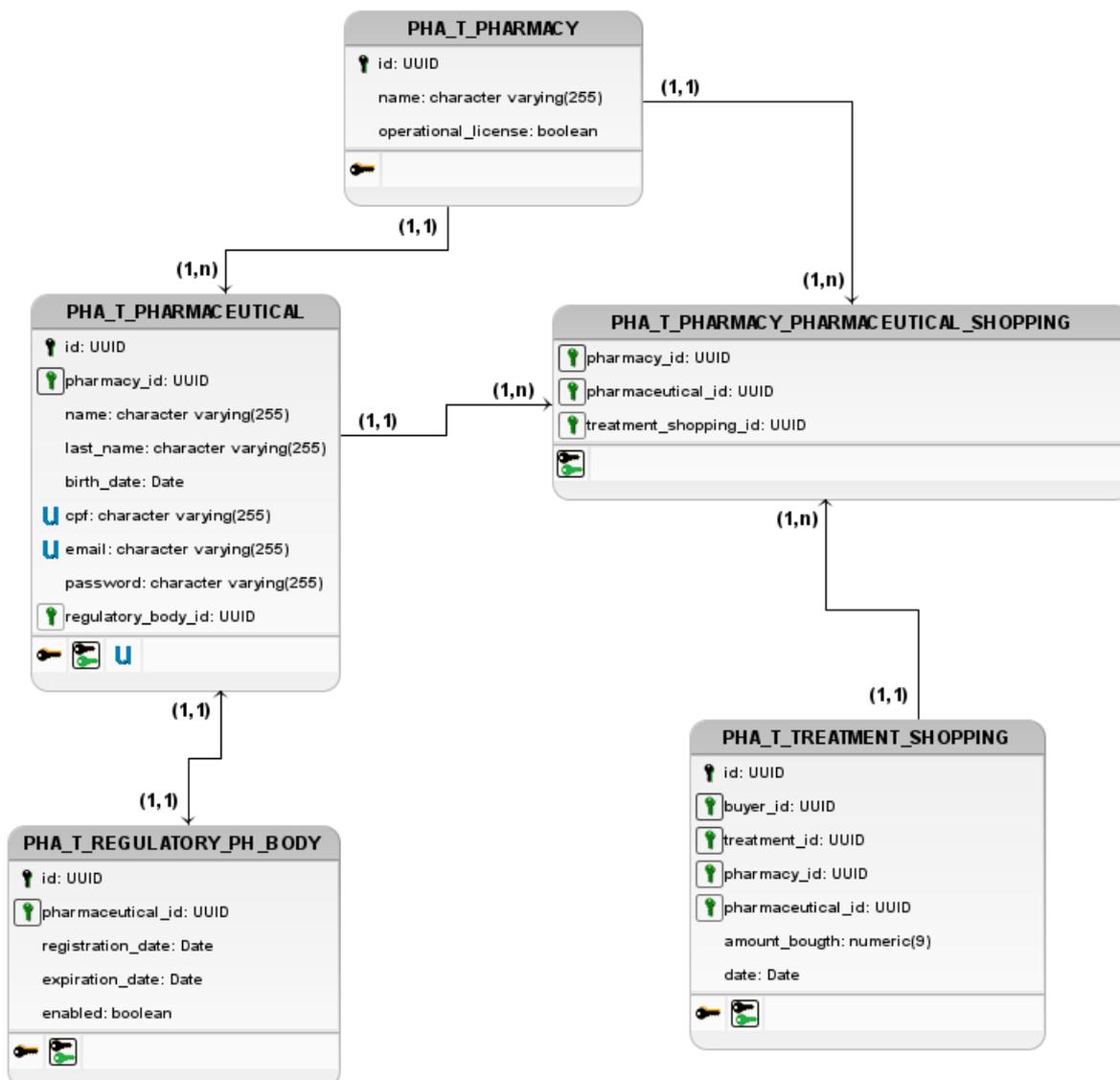
Fonte: Elaborado pelo autor

Figura 6 - Diagrama de Entidade e Relacionamento Micro Serviço Hospital



Fonte: Elaborado pelo autor

Figura 7 - Diagrama de Entidade e Relacionamento Micro Serviço Farmácia



Fonte: Elaborado pelo autor

4.4.1 Dicionário de Dados

Na criação do dicionário de dados é possível saber qual o tamanho que cada registro ocupa no banco, e através da quantidade de requisições que se espera receber por dia, levando ao conhecimento do tamanho mínimo do banco de dados para atender o sistema por um determinado período. Durante essa etapa foi levantado qual o tamanho de cada atributo para cada tabela, elencado suas descrições e suas

constraints, além de ter levantado em forma de tabela todas as relações que determinada tabela possui.

Tabela 20 - Dicionário de Dados Tabela *HOS_T_HOSPITAL_MEDIC*

| HOS_T_HOSPITAL_MEDIC | | |
|-----------------------------|-------------------------------|-------------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| HOS_T_MEDIC | Hospital tem médico | Relaciona o hospital com os médicos |
| HOS_T_HOSPITAL | Médico pertence hospital | |

Fonte: Elaborado pelo autor.

Tabela 21 - Dicionário de Dados Tabela *HOS_T_MEDIC*

| HOS_T_MEDIC | | |
|-----------------------------|-------------------------------|-------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| HOS_T_REGULATORY_MEDIC_BODY | Médico possui registro | Tabela de cadastro de médicos |
| HOS_T_HOSPITAL_MEDIC | Hospital tem médico | |
| HOS_T_APPOINTMENT | Atendimento pertence médico | |

Fonte: Elaborado pelo autor.

Tabela 22 - Dicionário de Dados Tabela *HOS_T_HOSPITAL*

| HOS_T_HOSPITAL | | |
|----------------------|----------------------------|---------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| HOS_T_HOSPITAL_MEDIC | Hospital tem médico | Tabela de cadastro de hospitais |
| HOS_T_APPOINTMENT | Hospital tem atendimento | |
| HOS_T_RECEPCIONIST | Hospital tem recepcionista | |

Fonte: Elaborado pelo autor.

Tabela 23 - Dicionário de Dados Tabela *HOS_T_APPOINTMENT*

| HOS_T_APPOINTMENT | | |
|--------------------|-----------------------------|---------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| HOS_T_MEDIC | Hospital tem atendimento | Tabela de cadastro de hospitais |
| HOS_T_HOSPITAL | Atendimento pertence médico | |
| HOS_T_RECEPCIONIST | Recepcionista atendeu | |

Fonte: Elaborado pelo autor.

Tabela 24 - Dicionário de Dados Tabela *HOS_T_RECEPCIONIST*

| HOS_T_RECEPCIONIST | | |
|--------------------|----------------------------|--------------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| HOS_T_HOSPITAL | Hospital tem recepcionista | Tabela de cadastro de recepcionistas |
| HOS_T_APPOINTMENT | Recepcionista atendeu | |

Fonte: Elaborado pelo autor.

Tabela 25 - Dicionário de Dados Tabela *CIT_T_ADRESS*

| CIT_T_ADRESS | | |
|----------------|------------------------|---------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| CIT_T_CITIZEN | Cidadão mora | Tabela de cadastro de endereços |

Fonte: Elaborado pelo autor.

Tabela 26 - Dicionário de Dados Tabela *CIT_T_CITIZEN*

| CIT_T_CITIZEN | | |
|-------------------------|----------------------------|--------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| CIT_T_ADRESS | Cidadão mora | Tabela de cadastro de cidadãos |
| CIT_T_TREATMENT | Cidadão possui tratamento | |
| CIT_T_AUTHORIZED_PERSON | Cidadão possui responsável | |

Fonte: Elaborado pelo autor.

Tabela 27 - Dicionário de Dados Tabela *CIT_T_TREATMENT*

| CIT_T_TREATMENT | | |
|-----------------|---------------------------|-----------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| CIT_T_CITIZEN | Cidadão possui tratamento | Tabela de cadastro de tratamentos |

Fonte: Elaborado pelo autor.

Tabela 28 - Dicionário de Dados Tabela *CIT_T_AUTHORIZED_PERSON*

| CIT_T_AUTHORIZED_PERSON | | |
|-------------------------|----------------------------|------------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| CIT_T_CITIZEN | Cidadão possui responsável | Tabela de cadastro de responsáveis |

Fonte: Elaborado pelo autor.

Tabela 29 - Dicionário de Dados Tabela *PHA_T_REGULATORY_PH_BODY*

| PHA_T_REGULATORY_PH_BODY | | |
|--------------------------|------------------------------|---|
| Relacionamento | Nome do Relacionamento | Descrição |
| PHA_T_PHARMACEUTICAL | Farmacêutico possui registro | Tabela de cadastro de registros do farmacêutico |

Fonte: Elaborado pelo autor.

Tabela 30 - Dicionário de Dados Tabela *PHA_T_PHARMACEUTICAL*

| PHA_T_PHARMACEUTICAL | | |
|--|------------------------------|-------------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| PHA_T_REGULATORY_PH_BODY | Farmacêutico possui registro | Tabela de cadastro de farmacêuticos |
| PHA_T_PHARMACY | Farmácia tem farmacêutico | |
| PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING | Farmacêutico realiza venda | |

Fonte: Elaborado pelo autor.

Tabela 31 - Dicionário de Dados Tabela *PHA_T_PHARMACY*

| PHA_T_PHARMACY | | |
|--|---------------------------|---------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| PHA_T_PHARMACEUTICAL | Farmácia tem farmacêutico | Tabela de cadastro de farmácias |
| PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING | Farmácia realiza venda | |

Fonte: Elaborado pelo autor.

Tabela 32 - Dicionário de Dados Tabela *PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING*

| PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING | | |
|--|----------------------------|---|
| Relacionamento | Nome do Relacionamento | Descrição |
| PHA_T_PHARMACY | Farmácia realiza venda | Tabela de cadastro de vendas da farmácia/farmacêutico |
| PHA_T_PHARMACEUTICAL | Farmacêutico realiza venda | |
| PHA_T_TREATMENT_SHOPPING | Venda possui tratamento | |

Fonte: Elaborado pelo autor.

Tabela 33 - Dicionário de Dados Tabela *PHA_T_TREATMENT_SHOPPING*

| PHA_T_TREATMENT_SHOPPING | | |
|--|-------------------------|-------------------------------|
| Relacionamento | Nome do Relacionamento | Descrição |
| PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING | Venda possui tratamento | Tabela de registros de vendas |

Fonte: Elaborado pelo autor.

Tabela 34 - Dicionário de Dados Atributos da Tabela *HOS_T_REGULATORY_MEDIC_BODY*

| HOS_T_REGULATORY_MEDIC_BODY | | | | | |
|-----------------------------|--------------|----------|--------------------|--------------|---|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO CONSELHO REGIONAL |
| MEDIC_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO MÉDICO |
| REGISTRATION_DATE | DATE | 4 BYTES | NOT NULL | N/D | DATA DE EXPEDIÇÃO DO REGISTRO |
| EXPIRATION_DATE | DATE | 4 BYTES | NOT NULL | N/D | DATA EXPIRAÇÃO DO REGISTRO |
| ENABLED | BOOLEAN | 1 BYTE | NOT NULL | N/D | PROFISSIONAL APTO A EXERCER ESPECIALIDADE |
| SPECIALITY | NUMERIC (9) | 4 BYTES | NOT NULL | N/D | ESPECIALIDADE DO REGISTRO |

Fonte: Elaborado pelo autor.

Tabela 35 - Dicionário de Dados Atributos da Tabela *HOS_T_HOSPITAL_MEDIC*

| HOS_T_HOSPITAL_MEDIC | | | | | |
|----------------------|--------------|----------|--------------|--------------|---------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| HOSPITAL_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO HOSPITAL |
| MEDIC_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO MÉDICO |

Fonte: Elaborado pelo autor.

Tabela 36 - Dicionário de Dados Atributos da Tabela *HOS_T_MEDIC*

| HOS_T_MEDIC | | | | | |
|-------------|-------------------------|-----------|--------------------|--------------|---------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO MÉDICO |
| HOSPITAL | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO HOSPITAL |
| NAME | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | NOME DO MÉDICO |
| LAST_NAME | DATE | 4 BYTES | NOT NULL | N/D | SOBRENOME DO MÉDICO |
| BIRTH_DATE | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | DATA DE NASCIMENTO |
| CPF | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | CPF DO MÉDICO |
| EMAIL | CHARACTER VARYING (255) | 256 BYTES | UNIQUE | N/D | E-MAIL DO MÉDICO |
| USERNAME | CHARACTER VARYING (255) | 256 BYTES | UNIQUE | N/D | NOME DE USUÁRIO DO MÉDICO |
| PASSWORD | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | SENHA DO MÉDICO |

Fonte: Elaborado pelo autor.

Tabela 37 - Dicionário de Dados Atributos da Tabela *HOS_T_HOSPITAL*

| HOS_T_HOSPITAL | | | | | |
|---------------------|-------------------------|-----------|--------------------|--------------|--|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO HOSPITAL |
| NAME | CHARACTER VARYING (255) | 256 BYTES | N/D | N/D | NOME DO HOSPITAL |
| WEBSITE | CHARACTER VARYING (255) | 256 BYTES | N/D | N/D | WEBSITE DO HOSPITAL |
| OPERATIONAL_LICENCE | BOOLEAN | 1 BYTE | N/D | N/D | STATUS LICENÇA DE OPERAÇÃO DO HOSPITAL |

Fonte: Elaborado pelo autor.

Tabela 38 - Dicionário de Dados Atributos da Tabela *HOS_T_APPOINTMENT*

| HOS_T_APPOINTMENT | | | | | |
|-------------------|--------------|----------|-----------------------|--------------|---|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DA CONSULTA |
| HOSPITAL_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO HOSPITAL |
| PATIENT_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO PACIENTE |
| RECEPTIONIST_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA RECEPCIONISTA |
| MEDIC_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO MÉDICO |
| DATE | DATE | 4 BYTES | NOT NULL | N/D | DATA DA CONSULTA |
| OPEN_HISTORIC | BOOLEAN | 1 BYTE | NOT NULL | FALSE | HISTÓRICO DISPONÍVEL PARA VISUALIZAÇÃO / ADIÇÃO |

Fonte: Elaborado pelo autor.

Tabela 39 - Dicionário de Dados Atributos da Tabela *HOS_T_RECEPCIONIST*

| HOS_T_RECEPCIONIST | | | | | |
|--------------------|-------------------------|-----------|--------------------|--------------|----------------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DA RECEPCIONISTA |
| HOSPITAL_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO HOSPITAL |
| NAME | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | NOME DA RECEPCIONISTA |
| LAST_NAME | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | SOBRENOME DA RECEPCIONISTA |
| BIRTH_DATE | DATE | 4 BYTES | NOT NULL | N/D | DARTA DE NASCIMENTO |
| CPF | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | CPF DA RECEPCIONISTA |
| EMAIL | CHARACTER VARYING (255) | 256 BYTES | UNIQUE | N/D | E-MAIL DA REEPCIONISTA |
| USERNAME | CHARACTER VARYING (255) | 256 BYTES | UNIQUE | N/D | NOME DE USUÁRIO DA RECEPCIONISTA |
| PASSWORD | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | SENHA DA RECEPCIONISTA |

Fonte: Elaborado pelo autor.

Tabela 40 - Dicionário de Dados Atributos da Tabela *CIT_T_ADRESS*

| CIT_T_ADRESS | | | | | |
|--------------|-------------------------|-----------|-----------------------|--------------|---------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO ENDEREÇO |
| CEP | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | CEP ENDEREÇO |
| NUMERO | NUMERIC(9) | 4 BYTES | | N/D | NÚMERO DO ENDEREÇO |
| LOGRADOURO | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | NOME DA RUA |
| COMPLEMENTO | CHARACTER VARYING (255) | 256 BYTES | | N/D | COMPLEMENTO DO ENDEREÇO |
| BAIRRO | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | BAIRRO |
| LOCALIDADE | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | CIDADE |
| IBGE | NUMERIC(9) | 4 BYTES | NOT NULL | N/D | CÓDIGO DO IBGE |
| GIA | NUMERIC(9) | 4 BYTES | | N/D | CÓDIO GIA |
| DDD | NUMERIC(9) | 4 BYTES | NOT NULL | N/D | COÓDIGO DE ÁREA |
| SIAFI | NUMERIC(9) | 4 BYTES | | N/D | CÓDIDO SIAFI |
| UF | CHARACTER VARYING(2) | 2 BYTES | NOT NULL | | ESTADO |

Fonte: Elaborado pelo autor.

Tabela 41 - Dicionário de Dados Atributos da Tabela CIT_T_CITIZEN

| CIT_T_CITIZEN | | | | | |
|--------------------|-------------------------|-----------|-----------------------|--------------|---|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO CIDADÃO |
| NAME | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | NOME DO CIDADÃO |
| LAST_NAME | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | SOBRENOME DO CIDADÃO |
| BIRTH_DATE | DATE | 4 bytes | NOT NULL | N/D | DATA DE NASCIMENTO DO CIDADÃO |
| CPF | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | CPF DO CIDADÃO |
| E-MAIL | CHARACTER VARYING (255) | 256 BYTES | UNIQUE | N/D | E-MAIL DO CIDADÃO |
| USERNAME | CHARACTER VARYING (255) | 256 BYTES | UNIQUE | N/D | NOME DE USUÁRIO |
| PASSWORD | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | SENHA |
| SUS_NUMBER | NUMERIC(9) | 4 BYTES | UNIQUE | N/D | NÚMERO DO SUS CIDADÃO |
| RESPONSIBLE_PERSON | UUID | 128 BITS | FK | N/D | PESSOA RESPONSÁVEL PELO CIDADÃO |
| HISTORY_VISIBLE | BOOLEAN | 1 BYTE | NOT NULL | FALSE | HISTÓRICO DISPONÍVEL PARA VISUALIZAÇÃO / ADIÇÃO |

Fonte: Elaborado pelo autor.

Tabela 42 - Dicionário de Dados Atributos da Tabela *CIT_T_TREATMENT*

| CIT_T_TREATMENT | | | | | |
|------------------------------|-------------------------|-----------|--------------------|--------------|---|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DA RECEITA |
| CITIZEN | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO CIDADÃO |
| MEDIC | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO MÉDICO |
| MEDICINE | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO MEDICAMENTO |
| APPOIMENT | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA CONSULTA |
| MEDICINE_QUANTITY | NUMERIC(9) | 4 BYTES | NOT_NULL | N/D | QUANTIDADE DE MEDICAMENTOS PRESCRITO |
| MONTHLY_QUANTITY | NUMERIC(9) | 4 BYTES | | N/D | QUANTIDADE DE USO MENSAL |
| PRESCRIPTION_EXPIRATION_DATE | DATE | 4 BYTES | NOT_NULL | N/D | DATA DE EXPIRAÇÃO DA RECEITA |
| MEDICINE_BALANCE | NUMERIC(9) | 4 BYTES | NOT_NULL | N/D | SALDO DE MEDICAMENTO DISPONÍVEL PARA COMPRA |
| NOTE | CHARACTER VARYING (255) | 256 BYTES | | N/D | OBSERVAÇÕES DEIXADAS PELO MÉDICO |

Fonte: Elaborado pelo autor.

Tabela 43 - Dicionário de Dados Atributos da Tabela *CIT_T_AUTHORIZED_PERSON*

| CIT_T_AUTHORIZED_PERSON | | | | | |
|-------------------------|--------------|----------|--------------|--------------|------------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| CITIZEN_TREATMENT | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO RESPONSÁVEL |
| TREATMENT | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA RECEITA |

Fonte: Elaborado pelo autor.

Tabela 44 - Dicionário de Dados Atributos da Tabela *PHA_T_REGULATORY_PH_BODY*

| PHA_T_REGULATORY_PH_BODY | | | | | |
|--------------------------|--------------|----------|-----------------------|--------------|---|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO CONSELHO REGIONAL |
| PHARMACEUTICAL_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO FARMACÊUTICO |
| REGISTRATION_DATE | DATE | 4 BYTES | NOT_NULL | N/D | DATA DE EXPEDIÇÃO DO REGISTRO |
| EXPIRATION_DATE | DATE | 4 BYTES | | N/D | CASO NECESSITE REVALIDAR O REGISTRO |
| ENABLE | BOOLEAN | 1 BYTE | | N/D | SE O PROFISSIONAL ESTÁ APTO A EXERCER ESPECIALIDADE |

Fonte: Elaborado pelo autor.

Tabela 45 - Dicionário de Dados Atributos da Tabela *PHA_T_PHARMACY*

| PHA_T_PHARMACY | | | | | |
|---------------------|-------------------------|-----------|-----------------------|--------------|---|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DA FARMÁCIA |
| NAME | CHARACTER VARYING (255) | 256 BYTES | NOT_NULL | N/D | NOME DA FARMÁCIA |
| OPERATIONAL_LICENCE | BOOLEAN | 1 BYTE | N/D | N/D | STATUS DO ALVARÁ E DEMAIS LICENSAS DE FUNCIONAMENTO |

Fonte: Elaborado pelo autor.

Tabela 46 - Dicionário de Dados Atributos da Tabela *PHA_T_PHARMACEUTICAL*

| PHA_T_PHARMACEUTICAL | | | | | |
|----------------------|-------------------------|-----------|--------------------|--------------|------------------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DO FARMACÊUTICO |
| PHARMACY_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA FARMÁCIA |
| NAME | CHARACTER VARYING (255) | 256 BYTES | | N/D | NOME DO FARMACÊUTICO |
| LASTNAME | CHARACTER VARYING (255) | 256 BYTES | | N/D | SOBRENOME DO FARMACÊUTICO |
| BIRTHDATE | DATE | 4 BYTES | | N/D | DATA DE NASCIMENTO |
| CPF | CHARACTER VARYING (255) | 256 BYTES | NOT NULL | N/D | CPF DO FARMACÊUTICO |
| EMAIL | CHARACTER VARYING (255) | 256 BYTES | NOT_NULL | N/D | EMAIL DO FARMACÊUTICO |
| PASSWORD | CHARACTER VARYING (255) | 256 BYTES | | N/D | SENHA |
| REGULATORY | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO CONSELHO REGIONAL |

Fonte: Elaborado pelo autor.

Tabela 47 - Dicionário de Dados Atributos da Tabela *PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING*

| PHA_T_PHARMACY_PHARMACEUTICAL_SHOPPING | | | | | |
|--|--------------|----------|--------------|--------------|-------------------------------|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| PHARMACY_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA FARMÁCIA |
| PHARMACEUTICAL_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO FARMACÊUTICO |
| TREATMENT_SHOPPING_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA COMPRA |

Fonte: Elaborado pelo autor.

Tabela 48 - Dicionário de Dados Atributos da Tabela *PHA_T_TREATMENT_SHOPPING*

| PHA_T_TREATMENT_SHOPPING | | | | | |
|--------------------------|--------------|----------|--------------------|--------------|--|
| Atributo | Tipo de Dado | Tam. | Restrições | Valor Padrão | Descrição |
| ID | UUID | 128 BITS | PK, AUTO_INCREMENT | N/D | IDENTIFICADOR DA COMPRA |
| BUYER_ID | UUID | 128 BITS | NOT_NULL | N/D | IDENTIFICADOR DO CIDADÃO QUE REALIZOU A COMPRA |
| TRATAMENT_ID | UUID | 128 BITS | NOT_NULL | N/D | IDENTIFICADOR DA RECEITA |
| PHARMACY_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DA FARMÁCIA |
| PHARMACEUTICAL_ID | UUID | 128 BITS | FK, NOT NULL | N/D | IDENTIFICADOR DO FARMACÊUTICO |
| AMOUT_BOUGHT | NUMERIC(9) | 4 BYTES | FK, NOT_NULL | N/D | QUANTIDADE COMPRADA |
| DATE | DATE | 4 BYTES | | SYSDATE | DATA E HORA DA COMPRA |

Fonte: Elaborado pelo autor.

4.5 Plano de testes

Algumas validações podem parecer um pouco repetitivo durante um mesmo fluxo de processo para quem estiver observando do lado de implementação, no entanto, por se tratar de uma API, devemos levar em consideração as rotas de acesso, e que elas deverão estar visíveis, por tanto a necessidade de dupla checagem durante o mesmo processo, como é o caso que ocorre do farmacêutico visualizar uma receita e realizar a venda, onde o sistema verifica por exemplo a data de expiração da prescrição médica.

Ainda para validar os testes, foi necessário levar em consideração situações como, a nulidade dos atributos obrigatórios, validação de atributos compatíveis e

dentro do formato esperado, e cruzamento de informações nas bases de dados necessárias.

Tabela 49 - Caso de Teste - Cadastro Completo

| Caso de Teste - Cadastro Completo | |
|--|--|
| Informar cpf inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Informar cpf já cadastrado | O sistema lança exceção com a mensagem "CPF já cadastrado no sistema." |
| Enviar de respostas inválidas e/ou incorretas. | O sistema lança exceção com a mensagem "Respostas incorretas." |
| Enviar de nome de usuário em branco | O sistema lança exceção com a mensagem "Obrigatório informar o nome de usuário." |
| Enviar de nome de usuário já cadastrado para outro cpf | O sistema lança exceção com a mensagem "Nome de usuário já cadastrado, informe outro nome de usuário." |

Fonte: Elaborado pelo autor.

Tabela 50 - Caso de Teste - Login Cidadão

| Caso de Teste - Login Cidadão | |
|---|--|
| Informar nome de usuário e/ou senha inválidos | O sistema lança exceção com a mensagem "Nome de usuário e/ou senha inválidos." |

Fonte: Elaborado pelo autor.

Tabela 51 - Caso de Teste - Visualizar Histórico de Tratamentos

| Caso de Teste - Visualizar Histórico de Tratamentos | |
|--|---|
| Não possuir um token válido | O sistema lança exceção com a mensagem "Necessário estar logado." |

Fonte: Elaborado pelo autor.

Tabela 52 - Caso de Teste - Atribuir Compra de Medicamentos para Responsável

| Caso de Teste - Atribuir Compra de Medicamentos para Responsável | |
|---|--|
| Não possuir um token válido | O sistema lança exceção com a mensagem "Necessário estar logado." |
| Informar um cpf inválido para responsável | O sistema lança exceção com a mensagem "Responsável não encontrado, atribua um CPF válido para o responsável pela compra." |
| Informar um cpf de uma pessoa menor de 16 anos | O sistema lança exceção com a mensagem "Responsável menor de 16 anos, atribuição não permitida." |
| Informar medicamento com a prescrição com data expirada | O sistema lança exceção com a mensagem "O medicamento informado está com a prescrição expirada." |
| Informar medicamento sem saldo de compra | O sistema lança exceção com a mensagem "O medicamento informado não possui mais saldo para compras." |

Fonte: Elaborado pelo autor.

Tabela 53 - Caso de Teste - Alteração de Cadastro

| Caso de Teste - Alteração de Cadastro | |
|--|---|
| Não possuir um token válido | O sistema lança exceção com a mensagem "Necessário estar logado." |
| Não informar nome | O sistema lança exceção com a mensagem "Necessário informar um nome." |
| Não informar sobrenome | O sistema lança exceção com a mensagem "Necessário informar um sobrenome." |
| Informar e-mail inválido | O sistema lança exceção com a mensagem "E-mail inválido." |
| Informar senha inválida | O sistema lança exceção com a mensagem "Senha inválida. A senha deve conter pelo menos 8 caracteres contendo letras maiúsculas e minúsculas, números e caracteres especiais." |
| Informar cep inválido | O sistema lança exceção com a mensagem "CEP inválido." |
| Não informar bairro | O sistema lança exceção com a mensagem "Bairro inválido." |
| Não informar logradouro | O sistema lança exceção com a mensagem "Logradouro inválido. Informe se o logradouro é aeroporto, alameda, área, avenida etc." |
| Não informar número | O sistema lança exceção com a mensagem "Número inválido. Caso o endereço não possua número, informe S/N." |

Fonte: Elaborado pelo autor.

Tabela 54 - Caso de Teste - Login Recepcionista

| Caso de Teste - Login Recepcionista | |
|---|--|
| Informar nome de usuário, senha e/ou hospital inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou hospital inválidos." |

Fonte: Elaborado pelo autor.

Tabela 55 - Caso de Teste - *Agendar Consulta*

| Caso de Teste - Agendar Consulta | |
|--|---|
| Não possuir um token válido | O sistema lança exceção com a mensagem "Necessário estar logado." |
| Não informar o cpf do paciente | O sistema lança exceção com a mensagem "CPF do paciente não informado." |
| Informar cpf inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Não informar o número do cartão do sus do paciente | O sistema lança exceção com a mensagem "Número do cartão do SUS do paciente não informado." |
| Informar número do cartão do sus inválido | O sistema lança exceção com a mensagem "Número do cartão do SUS inválido." |
| Não informar hospital | O sistema lança exceção com a mensagem "Hospital não informado." |
| Informar hospital inválido | O sistema lança exceção com a mensagem "Hospital inválido." |
| Agendamento de uma especialidade não disponibilizada pelo hospital | O sistema lança exceção com a mensagem "Hospital não atende esse tipo de especialidade." |
| Não informar recepcionista | O sistema lança exceção com a mensagem "Recepcionista não informado." |
| Informar recepcionista inválida | O sistema lança exceção com a mensagem "Recepcionista inválido." |
| Não informar médico | O sistema lança exceção com a mensagem "Médico não informado." |
| Informar médico inválida | O sistema lança exceção com a mensagem "Médico inválido." |

Fonte: Elaborado pelo autor.

Tabela 56 - Caso de Teste - Recepcionar Paciente Agendado

| Caso de Teste - Recepcionar Paciente Agendado | |
|--|---|
| Não possuir um token válido | O sistema lança exceção com a mensagem "Necessário estar logado." |
| Não informar o cpf do paciente | O sistema lança exceção com a mensagem "CPF do paciente não informado." |
| Informar cpf inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Não informar o número do cartão do sus do paciente | O sistema lança exceção com a mensagem "Número do cartão do SUS do paciente não informado." |
| Informar número do cartão do sus inválido | O sistema lança exceção com a mensagem "Número do cartão do SUS inválido." |
| Não informar hospital | O sistema lança exceção com a mensagem "Hospital não informado." |
| Informar hospital inválido | O sistema lança exceção com a mensagem "Hospital inválido." |
| Horário da consulta errado | O sistema lança exceção com a mensagem "Data da consulta errada." |

Fonte: Elaborado pelo autor.

Tabela 57 - Caso de Teste - Liberar Paciente para Consulta

| Caso de Teste - Liberar Paciente para Consulta | |
|---|---|
| Não possuir um token válido | O sistema lança exceção com a mensagem "Necessário estar logado." |
| Não informar o cpf do paciente | O sistema lança exceção com a mensagem "CPF do paciente não informado." |
| Informar cpf inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Não informar hospital | O sistema lança exceção com a mensagem "Hospital não informado." |
| Informar hospital inválido | O sistema lança exceção com a mensagem "Hospital inválido, paciente não possui agendamento nesse hospital." |
| Horário da consulta errado | O sistema lança exceção com a mensagem "Data da consulta errada." |
| Especialidade errada | O sistema lança exceção com a mensagem "Especialidade da consulta errada para o atendimento." |

Fonte: Elaborado pelo autor.

Tabela 58 - Caso de Teste - Login Médico

| Caso de Teste - Login Médico | |
|---|--|
| Informar nome de usuário, senha e/ou hospital inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou hospital inválidos." |
| Registro inválido. | O sistema lança exceção com a mensagem "Registro do médico inválido." |
| Registro expirado. | O sistema lança exceção com a mensagem "Registro do médico expirado." |

Fonte: Elaborado pelo autor.

Tabela 59 - Caso de Teste - Consultar Paciente

| Caso de Teste - Consultar Paciente | |
|--|--|
| Informar nome de usuário, senha e/ou hospital inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou hospital inválidos." |
| Registro inválido | O sistema lança exceção com a mensagem "Registro do médico inválido." |
| Registro expirado | O sistema lança exceção com a mensagem "Registro do médico expirado." |
| Não informar cpf cliente | O sistema lança exceção com a mensagem "CPF do cliente não informado." |
| Cpf cliente inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Paciente não liberado pela recepção | O sistema lança exceção com a mensagem "Paciente não liberado para atendimento na recepção." |
| Médico com especialidade diferente da agendada na consulta | O sistema lança exceção com a mensagem "Especialidade do médico divergente da agendada." |

Fonte: Elaborado pelo autor.

Tabela 60 - Caso de Teste - Visualizar Histórico do Paciente

| Caso de Teste - Visualizar Histórico do Paciente | |
|---|--|
| Informar nome de usuário, senha e/ou hospital inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou hospital inválidos." |
| Registro inválido | O sistema lança exceção com a mensagem "Registro do médico inválido." |
| Registro expirado | O sistema lança exceção com a mensagem "Registro do médico expirado." |
| Não informar cpf cliente | O sistema lança exceção com a mensagem "CPF do cliente não informado." |
| Cpf cliente inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Paciente não liberado pela recepção | O sistema lança exceção com a mensagem "Paciente não liberado para atendimento na recepção." |

Fonte: Elaborado pelo autor.

Tabela 61 - Caso de Teste - *Prescrever Receituário*

| Caso de Teste - Prescrever Receituário | |
|---|--|
| Informar nome de usuário, senha e/ou hospital inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou hospital inválidos." |
| Registro inválido | O sistema lança exceção com a mensagem "Registro do médico inválido." |
| Registro expirado | O sistema lança exceção com a mensagem "Registro do médico expirado." |
| Não informar cpf cliente | O sistema lança exceção com a mensagem "CPF do cliente não informado." |
| Cpf cliente inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Paciente não liberado pela recepção | O sistema lança exceção com a mensagem "Paciente não liberado para atendimento na recepção." |
| Não informar medicamento | O sistema lança exceção com a mensagem " Medicamento não informado." |
| Medicamento inválido | O sistema lança exceção com a mensagem "Medicamento inválido." |
| Medicamentos suspensos pela Anvisa | O sistema lança exceção com a mensagem "O medicamento informado encontra-se com as vendas temporariamente suspensa pela Anvisa." |
| Não informar a consulta | O sistema lança exceção com a mensagem "Consulta não informada." |
| Consulta inválido | O sistema lança exceção com a mensagem "Consulta inválido." |
| Não informar a quantidade | O sistema lança exceção com a mensagem "Quantidade de medicamento não informada." |
| Quantidade mensal maior que a quantidade informada | O sistema lança exceção com a mensagem "Quantidade mensal não pode exceder a quantidade do medicamento." |

| | |
|--|--|
| Não informar a data de expiração da prescrição | O sistema lança exceção com a mensagem "Data de expiração da prescrição não informada." |
| Data de expiração menor que a data atual | O sistema lança exceção com a mensagem "Data de expiração da prescrição não pode ser inferior a data atual." |

Fonte: Elaborado pelo autor.

Tabela 62 - Caso de Teste - Login Farmacêutico

| Caso de Teste - Login Farmacêutico | |
|---|--|
| Informar nome de usuário, senha e/ou farmácia inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou farmácia inválidos." |
| Registro inválido. | O sistema lança exceção com a mensagem "Registro do farmacêutico inválido." |
| Registro expirado. | O sistema lança exceção com a mensagem "Registro do farmacêutico expirado." |

Fonte: Elaborado pelo autor.

Tabela 63 - Caso de Teste - Iniciar Atendimento

| Caso de Teste - Iniciar Atendimento | |
|---|--|
| Informar nome de usuário, senha e/ou farmácia inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou farmácia inválidos." |
| Registro inválido | O sistema lança exceção com a mensagem "Registro do farmacêutico inválido." |
| Registro expirado | O sistema lança exceção com a mensagem "Registro do farmacêutico expirado." |
| Não informar cpf cliente | O sistema lança exceção com a mensagem "CPF do cliente não informado." |
| Cpf cliente inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Não informar código do tratamento | O sistema lança exceção com a mensagem "Código do Tratamento não informado." |
| Código do tratamento inválido | O sistema lança exceção com a mensagem "Código do Tratamento inválido." |
| Cpf não permitido para o código do tratamento | O sistema lança exceção com a mensagem "O cliente não tem permissão para compra dos medicamentos da prescrição." |
| Informar medicamento com a prescrição com data expirada | O sistema lança exceção com a mensagem "O medicamento informado está com a prescrição expirada." |
| Informar medicamento sem saldo de compra | O sistema lança exceção com a mensagem "O medicamento informado não possui mais saldo para compras." |

Fonte: Elaborado pelo autor.

Tabela 64 - Caso de Teste - Vender Medicamento

| Caso de Teste - Vender Medicamento | |
|---|--|
| Informar nome de usuário, senha e/ou farmácia inválidos | O sistema lança exceção com a mensagem "Nome de usuário, senha e/ou farmácia inválidos." |
| Registro inválido | O sistema lança exceção com a mensagem "Registro do farmacêutico inválido." |
| Registro expirado | O sistema lança exceção com a mensagem "Registro do farmacêutico expirado." |
| Não informar cpf cliente | O sistema lança exceção com a mensagem "CPF do cliente não informado." |
| Cpf cliente inválido | O sistema lança exceção com a mensagem "CPF inválido." |
| Não informar código do tratamento | O sistema lança exceção com a mensagem "Código do Tratamento não informado." |
| Código do tratamento inválido | O sistema lança exceção com a mensagem "Código do Tratamento inválido." |
| Cpf não permitido para o código do tratamento | O sistema lança exceção com a mensagem "O cliente não tem permissão para compra dos medicamentos da prescrição." |
| Informar medicamento com a prescrição com data expirada | O sistema lança exceção com a mensagem "O medicamento informado está com a prescrição expirada." |
| Informar medicamento sem saldo de compra | O sistema lança exceção com a mensagem "O medicamento informado não possui mais saldo para compras." |
| Medicamentos suspensos pela Anvisa | O sistema lança exceção com a mensagem "O medicamento informado encontra-se com as vendas temporariamente suspensa pela Anvisa." |

Fonte: Elaborado pelo autor.

5 DESENVOLVIMENTO

Durante o desenvolvimento do projeto, foi utilizado metodologias ágeis juntamente com o *framework* Scrum, para realizar entregas por *sprint* e gerenciar melhor as atividades e prazo de cada entrega, seguindo os melhores padrões de desenvolvimento de software.

5.1 Sprint 1

Nos dias 08 e 09 de fevereiro de 2023, foi iniciado a análise sobre o Projeto, para entender as necessidades de todos os usuários envolvidos, como cliente, recepcionista, médico e farmacêutico. Observando também as necessidades dos órgãos que serão validadores de informações, como Anvisa, Conselho Regional de Medicina e Conselho Regional de Farmácia integrando com o principal interessado o GOV. Sprint também dedicado ao início da elaboração da primeira entrega solicitada. Nos dias 15 e 16 de fevereiro foi iniciado o desenvolvimento da documentação do Projeto.

Tabela 65 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|---------------------------------|--------------|---------------|
| Análise do perfil usuário | 2 | 0,25 |
| Análise do perfil recepcionista | 2 | 0,25 |
| Análise do perfil farmácia | 1 | 0,12 |
| Análise do perfil médico | 1 | 0,12 |
| Análise do CRM | 2 | 0,25 |
| Análise CRF | 2 | 0,25 |
| Análise GOV | 2 | 0,25 |
| Análise Anvisa | 3 | 0,37 |
| Início da documentação | 2 | 0,25 |
| | | |
| | | |
| | | |
| | | |
| | | |
| Total | 17 | 2,11 |

Fonte: Elaborado pelo autor

No dia 16 de fevereiro de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Houve uma boa comunicação entre os integrantes. Dessa forma todos estavam cientes das dificuldades enfrentadas;
- O que deu errado: Não houve iniciativa de um integrante para ajudar a resolver os problemas identificados;
- Ações de melhorias: Alinhamento entre os integrantes para ter uma maior proatividade para entendimento e execução da demanda;

5.2 Sprint 2

Nos dias 22 e 23 de fevereiro foi elaborada a solicitação da tarefa, no qual era composta por entrega da capa, contracapa, resumo e softwares similares ao TrackMed. Como o software que está sendo desenvolvido é muito específico, foi comparado com softwares de convênios médico e com o próprio aplicativo ConecteSUS. Entregue toda a solicitação em documento PDF. Nos dias 01 e 02 de março continuou-se com a estruturação do material documental, iniciando os estudos dos requisitos funcionais e não funcionais do Projeto.

Tabela 66 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|---|--------------|---------------|
| Elaboração Capa | 0,5 | 0,06 |
| Elaboração contracapa | 0,5 | 0,06 |
| Elaboração Resumo | 2 | 0,25 |
| Elaboração softwares Similares | 3 | 0,37 |
| Desenvolvimento Requisitos Funcionais | 3 | 0,37 |
| Desenvolvimento Requisitos Não Funcionais | 3 | 0,37 |
| Total | 12 | 1,48 |

Fonte: Elaborado pelo autor.

No dia 02 de março de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Tive uma melhora na entrega do time, integrantes mais participativos;
- O que deu errado: Desenvolvimento de uma das atividades, demandou mais tempo que o previsto;
- Ações de melhorias: Uma reunião de detalhamento de todo o projeto para todos os integrantes novamente;

5.3 Sprint 3

Nos dias 08 e 09 de março a documentação do projeto continuou. Levantou-se item a item com cada integrante do grupo os requisitos funcionais e não funcionais de toda a estrutura do software, elencado todas as especificações necessárias, a fim de mitigar os erros da solução. Com esse novo *brainstorm* da equipe, foi identificado que ficaram muitos casos sem analisar, portanto o volume de requisitos aumentou consideravelmente, porém foi possível entregar todos os passos mapeados da API. A entrega foi realizada no final do período com a documentação toda em PDF. Nos dias 15 e 16 iniciou-se o mapeamento dos recursos e ferramentas, todas as tecnologias que iriam ser utilizadas para o desenvolvimento da API, nesta parte contou-se com uma maior expertise de um integrante da equipe, com grande conhecimento para a execução da demanda.

Tabela 67 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|---|--------------|---------------|
| Desenvolvimento Requisitos Funcionais | 1 | 0,12 |
| Desenvolvimento Requisitos não Funcionais | 1 | 0,12 |
| Elaborar documentação Entrega | 3 | 0,37 |
| Mapeamento recursos | 2 | 0,25 |
| Mapeamento ferramentas | 2 | 0,25 |
| Total | 9 | 1,11 |

Fonte: Elaborado pelo autor.

No dia 16 de março de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: O foco de dois integrantes em atividades que já atuam profissionalmente;
- O que deu errado: Integrantes com altas demandas, equipe desbalanceada com as demandas;
- Ações de melhorias: Alinhamento do time para balancear melhor as demandas;

5.4 Sprint 4

Nos dias 29 e 30 de março foi concluída a demanda de recursos e ferramentas que serão utilizadas no desenvolvimento da solução, foram escolhidas linguagens e tecnologias mais abrangentes e difundidas no mercado, para um desenvolvimento com maior suporte, foi reduzida a demanda desse Sprint devido feriado.

Tabela 68 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|-------------------------------------|--------------|---------------|
| Conclusão mapeamento de ferramentas | 4 | 0,5 |
| Conclusão mapeamento de recursos | 3 | 0,37 |
| Total | 7 | 0,87 |

Fonte: Elaborado pelo autor.

No dia 30 de março de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Entrega foi realizada com sucesso;
- O que deu errado: Sem nenhum percalço, todos se empenharam na demanda;
- Ações de melhorias: Manter a sinergia e empenho dos integrantes;

5.5 Sprint 5

Nos dias 05 e 06 de abril foi iniciado a tratativa do modelo de caso de uso, este Sprint foi complexo devido vários personagens envolvidos como: cidadão, recepcionista, médico, farmacêutico e todos os órgãos validadores e o órgão onde serão reunidas todas as informações. Primeiro foi realizado um caso de uso inteiro, porém foi identificado que não ficou correto, muitas informações ficaram em um mesmo modelo. A partir de então foi separado por cada ator, ficando com uma visão mais completa e organizada, porém não foi feito corretamente na primeira vez, foi necessário realizar cerca de 3 versões, como havia pouco tempo, focou-se um integrante, para iniciar o dicionário de dados e outro focado no dicionário de entidades, portanto toda equipe estava trabalhando na mesma demanda. No dia 12/04 um integrante focou para entregar o roteiro do Projeto (Engenharia de Software). Neste

momento era um integrante focado em cada atividade. Um em entrega de roteiro e dicionário de dados e entidades e outro ainda atuando no modelo de caso de uso. A entrega do roteiro foi realizada no dia 12/03. No dia 13/04 um integrante retornou para o modelo de caso de uso e o outro integrante ainda atuava nos dois dicionários.

Tabela 69 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|---|--------------|---------------|
| Início Modelo de casos de uso | 0,5 | 0,06 |
| Uso de caso cidadão | 3 | 0,37 |
| Uso de caso Recepcionista | 2 | 0,25 |
| Uso de casos médico | 3 | 0,37 |
| Desenvolvimento dicionário de dados | 7 | 0,87 |
| Desenvolvimento dicionário de entidades | 5 | 0,62 |
| Total | 20,5 | 2,54 |

Fonte: Elaborado pelo autor.

No dia 13 de abril de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Êxito em mitigar os erros dentro da própria sprint;
- O que deu errado: Demanda de tarefas acima do executado nos outros sprints;
- Ações de melhorias: Dimensionar melhor a quantidade de horas dos sprints;

5.6 Sprint 6

Neste Sprint dos dias 19 e 20/04 continuou-se trabalhando no Caso de uso do Projeto, executando a quinta versão deste material, pois os cenários eram complexos

e quando revisado, ainda era possível encontrar problemas, estavam todos os integrantes focados nesta demanda, nesta Sprint conseguiu-se concluir esta fase do projeto. No dia 19/04 entregou-se o Projeto parcialmente conforme solicitado, faltando cerca de 20% para a conclusão do mesmo. O integrante que estava atuando no Dicionário de entidade e banco de dados acabou concluindo no dia 27/04. Realizou-se uma reunião para dar início às necessidades da arquitetura do projeto, isso foi realizado no dia 26/04 para inserção desta informação, pois como trata-se de uma API não possui tela de interface do usuário.

Tabela 70 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|---------------------------------------|--------------|---------------|
| Revalidação caso de uso cidadão | 3 | 0,37 |
| Revalidação caso de uso recepcionista | 2 | 0,25 |
| Revalidação caso de uso farmacêutico | 4 | 0,5 |
| Revalidação caso de uso farmácia | 3 | 0,37 |
| Revalidação caso de uso médico | 3 | 0,37 |
| Entrega Projeto Parcial | 2 | 0,25 |
| Conclusão Dicionário de dados | 5 | 0,62 |
| Conclusão Dicionário de entidades | 6 | 0,75 |
| Reunião inicial arquitetura | 0,5 | 0,06 |
| Total | 28,5 | 3,54 |

Fonte: Elaborado pelo autor.

No dia 27 de abril de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Êxito em mitigar os erros dentro do próprio sprint;
- O que deu errado: Novamente alta demanda de horas neste sprint;
- Ações de melhorias: Realizar uma melhor divisão das histórias, para não acumular as horas do sprint;

5.7 Sprint 7

Neste sprint que iniciou dia 03 e 04/05 iniciou-se o desenvolvimento da arquitetura, como é API, não foi desenvolvido nenhuma interface para este projeto. Durante a elaboração da arquitetura, foi necessário parar este desenvolvimento para focar em uma visão geral do andamento do projeto, para a própria equipe, a fim de visualizar pontos que não estavam sendo levados em consideração. A apresentação foi realizada no dia 04/05 explanando o andamento do projeto, tudo que foi realizado até o momento, focando no que até então foi o mais complexo, o modelo de caso de uso. Concluído toda a arquitetura no dia 10/05, iniciou-se o Diagrama de entidade e Relacionamento, atividade focada com apenas um integrante no dia 11/05

Tabela 71 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|--|--------------|---------------|
| Continuação reunião arquitetura | 2 | 0,23 |
| Desenvolvimento da arquitetura | 3 | 0,37 |
| Conclusão arquitetura | 1 | 0,12 |
| Apresentação Projeto | 1 | 0,12 |
| Foco no Modelo de caso de uso | 3 | 0,37 |
| Início diagrama entidade de relacionamento | 4 | 0,5 |
| Total | 14 | 1,71 |

Fonte: Elaborado pelo autor.

No dia 11 de maio de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Encontrado novos pontos de melhorias, que haviam passado despercebidos pelo time;
- O que deu errado: Sprint conforme planejado;
- Ações de melhorias: Sem pontos de melhorias;

5.8 Sprint 8

Iniciando no dia 17/05, com um integrante focado no diagrama de entidade e relacionamento, conseguindo finalizar no dia 18/05, devido a todo mapeamento realizado durante o trabalho, ficando somente esta atividade para finalizar. Um integrante, realizava as correções / adições de informações ao projeto. Validando todos os dados, identificou-se a falta de dados comprobatórios de economia do governo ao digitalizar o SUS, informação foi inserida. No dia 24/05, iniciou-se o plano de testes do projeto, para assim mitigar os erros da API, realizando até o dia 25/05.

Tabela 72 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|---|--------------|---------------|
| Início diagrama entidade de relacionamento | 3 | 0,37 |
| Conclusão diagrama entidade de relacionamento | 2 | 0,25 |
| Correção Projeto | 4 | 0,5 |
| Inserção fonte comprobatórias | 2 | 0,25 |
| Total | 11 | 1,37 |

Fonte: Elaborado pelo autor.

No dia 25 de maio de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Atividade de diagrama de entidade, facilitado devido toda estruturação em sprints anteriores;
- O que deu errado: Informações faltantes do projeto, inseridos nesse sprint;
- Ações de melhorias: Revisão constante Sprint -1 projeto;

5.9 Sprint 9

Na semana final, no dia 31/05 foi realizado todas as considerações finais do projeto, com todos os integrantes envolvidos finalizando rapidamente esta etapa no mesmo dia. Foram atualizadas todas as referências utilizadas. No dia 01/06 concluiu-se todo o projeto apenas fazendo mais uma nova verificação. No dia 01/06, também teve início a apresentação do projeto, onde envolveu algumas chamadas via Teams entre os integrantes para de fato obter as melhores ideias para apresentação, conseguindo concluir essa demanda somente no dia 09/06, finalizando assim as atividades neste projeto.

Tabela 73 – Planejamento realizado para primeira entrega.

| Atividade | Tempo | Pontos |
|-------------------------|-------|--------|
| Revisão Projeto | 3 | 0,37 |
| Revisão documentação | 3 | 0,37 |
| Brainstorm apresentação | 2 | 0,25 |
| Início apresentação | 3 | 0,37 |
| Término Apresentação | 2 | 0,25 |
| Total | 13 | 1,36 |

Fonte: Elaborado pelo autor.

No dia 09 de junho de 2023 a equipe se reuniu para realizar a revisão do entregável. Nesta reunião todos os membros estavam presentes. Durante a reunião, a equipe fez uma autoavaliação procurando detectar os principais pontos de acertos e erros, e juntos, propor ações de melhorias para o desempenho da equipe nos próximos entregáveis. Ao final, a equipe fez o seguinte resumo sobre algumas questões:

- O que deu certo: Atividade de diagrama de entidade facilitado devido toda estruturação em sprints anteriores;
- O que deu errado: Informações faltantes do projeto, inseridos nesse sprint;
- Ações de melhorias: Revisão constante Sprint -1 Projeto;

6 CONSIDERAÇÕES FINAIS

No ano de 2019, no Brasil foram destinados para a área da saúde o equivalente a 9,6% do Produto Interno Bruto (PIB), sendo que desse montante apenas 3,6% (\$67,428 bilhões de dólares americanos) partiram dos cofres públicos, sendo o restante proveniente da iniciativa privada. Pesquisas apontam que esse valor não é suficiente para cuidar da saúde da população brasileira de maneira eficiente, além do fato de ter a diferença de distribuição de renda entre os hospitais, que agrava ainda mais a situação, e para piorar, existem os problemas de desvio de verbas públicas, falsos médicos, falta de medicamentos nos hospitais e farmácias populares, ou medicamentos vencidos que tiveram de ser descartados entre diversos outros casos de desrespeito com o dinheiro do contribuinte.

Os problemas são diversos nessa área, apesar de um software não ser a solução para todos esses problemas, acredita-se que a modernização de setores, sejam eles no âmbito público ou privado, é algo inevitável, cedo ou tarde irá acontecer, não apenas por conta da economia que isso possa trazer em um segundo momento, mas pela praticidade que a informatização das rotinas nos proporciona.

Nesse projeto foi proposto uma solução que une a comodidade para os cidadãos brasileiros de visualizar o histórico médico através de um aplicativo junto com a inteligência de fornecer dados para o Governo, para atuar de maneira estratégica e inteligente, com as informações geradas, sanando algumas, senão a maioria das dificuldades listadas no parágrafo anterior, da má gestão e dificuldade do estado brasileiro em fiscalizar e melhor distribuir os recursos destinados para a saúde. Para tal é necessário não apenas o investimento inicial sobre toda a infraestrutura para permitir a troca e armazenagem de dados. Entretanto, após a inicialização das atividades com TrackMed, seria indispensável também investimento em análise de dados, que será capaz de gerar informações a respeito de onde estão indo os recursos, a densidade demográfica atendida por um determinado montante de dinheiro. Obviamente o software em si não tem a capacidade de fiscalizar fraudes financeiras, mas seria uma ferramenta importante para os órgãos fiscalizadores, não apenas no quesito do dinheiro público, mas também no combate a medicamentos falsificados, clínicas e médicos ilegais, além de poder gerar informações que seriam de grande importância para estudos científicos, estudos de surtos e conhecer melhor

a saúde de cada região do país conforme a precisão solicitada, entre inúmeras possibilidades de pesquisas que podem surgir a partir dos dados levantados.

Para o futuro ainda seria possível, devido ao grande volume de dados gerados, aplicar o uso de inteligência artificial para auxiliar os profissionais da saúde a prestarem um atendimento rápido, objetivo de qualidade e precisão sobre o diagnóstico.

REFERÊNCIAS:

ANAHP, **Prontuário eletrônico pode gerar economia de R\$22 bilhões para o SUS.**

Disponível em: <<https://www.anahp.com.br/noticias/prontuario-eletronico-pode-gerar-economia-de-r-22-bi-para-o-sus/>>. Acesso em 27 abr 2023.

BANCO DO NORDESTE, **Panorama e perspectivas da cadeia de saúde do Brasil, nordeste, Ceará, Pernambuco e Bahia.** Disponível em:

<https://www.bnb.gov.br/s482-dspace/bitstream/123456789/1150/3/2022_CDS_217.pdf/>. Acesso em 11 maio 2023

BEZERRA, Eduardo. **Princípios de Análise e Projeto de Sistemas com UML.** 1ª Edição, Elsevier, 2014.

BRASIL. **Lei n. 13874**, de 20 de set. de 2019. Lei da Liberdade Econômica. Disponível em: <http://www.planalto.gov.br/ccivil_03/_ato2019-2022/2019/lei/L13874.htm>. Acesso em: 31 mai 2023

BRASIL. **Portaria n. 940**, de 28 de abr. de 2011. Regulamenta o Sistema Cartão Nacional de Saúde. Art. 4º parágrafo || Das disposições gerais.

BRASIL. **Portaria n. 2022**, de 7 de ago. de 2017. Dispõe da metodologia para cadastrar e atualizar estabelecimentos de saúde. Anexo III Classificação dos tipos de estabelecimentos de saúde.

CÂMARA DOS DEPUTADOS, **LEI Nº 13.787, DE 27 DE DEZEMBRO DE 2018, Legislação Informatizada - Dados da Norma.** Disponível em:

<<https://www2.camara.leg.br/legin/fed/lei/2018/lei-13787-27-dezembro-2018-787543-norma-pl.html>>. Acesso em: 01 out 2022

Disponível em: <<https://medicinas.com.br/brasil-digitalizacao-saude>>. Acesso em: 01 out 2022

CNESSP, **O que é CNES? Qual o prazo de sua validade? Você é obrigado a ter?**

Disponível em: <https://cnessp.com.br/o-que-e-cnes-qual-o-prazo-de-sua-validade-voce-e-obrigado-a-ter/?gclid=EAlaIQobChMlkdqWx5Ti-glVa-hcCh0BEQaLEAAYASAAEgJP-PD_BwE>. Acesso em: 15 out. 2022

EXAME, **Médicos são indiciados por falsidade ideológica.** Disponível em:

<<https://exame.com/brasil/medicos-sao-indiciados-por-falsidade-ideologica/>>. Acesso em 29 maio 2023

GZH, **Como funciona fraude que teria desviado R\$ 2,6 bilhões do programa Farmácia Popular.** Disponível em:

<<https://gauchazh.clicrbs.com.br/geral/noticia/2022/05/como-funciona-fraude-que-teria-desviado-r-26-bilhoes-do-programa-farmacia-popular-cl34tmaa70010019iayim7lgj.html>>. Acesso em: 29 maio 2023

JACOBSON, Daniel; BRAIL, Gregory; WOODS, Dan, **APIs: A Strategy Guide.** O'Reilly Media.

MARTINFOWLER.COM, **A definition of this new architectural term.** Disponível em:

<<https://martinfowler.com/articles/microservices.html>>. Acesso em 29 maio 2023

PWC, **O abismo digital no Brasil: Saiba como desigualdade de acesso à internet, a infraestrutura inadequada e a educação deficitária limitam as nossas opções para o futuro.**

Disponível em: <<https://www.pwc.com.br/pt/estudos/preocupacoes-ceos/mais-temas/2022/o-abismo-digital-no-brasil.html>>. Acesso em: 15 abr 2023

Rádio Senado, **Senado vota proposta de digitalização de prontuários de pacientes do SUS.** Disponível em:

<<https://www12.senado.leg.br/radio/1/noticia/2021/05/17/senado-vota-proposta-de-digitalizacao-de-prontuarios-de-pacientes-do-sus>>. Acesso em: 01 out 2022

MORSH, **Como está a saúde pública no Brasil e o que fazer para melhorar.**

Disponível em: <<https://telemedicinamorsch.com.br/blog/saude-publica-no-brasil/>>.

Acesso em 29 maio 2023