

UNIVERSIDADE FEDERAL DA PARAÍBA
DEPARTAMENTO DE INFORMÁTICA
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MATHEUS ARAUJO DE MORAIS

**OTIMIZANDO O TEMPO DO ATENDIMENTO HOSPITALAR: UMA SOLUÇÃO
PARA FILAS EM HOSPITAIS**

JOÃO PESSOA
2023

MATHEUS ARAUJO DE MORAIS

**OTIMIZANDO O TEMPO DO ATENDIMENTO HOSPITALAR: UMA SOLUÇÃO
PARA FILAS EM HOSPITAIS**

Trabalho de Conclusão de Curso apresentado à Universidade Federal da Paraíba, como requisito para a obtenção do título de bacharelado em Ciência da Computação.

Orientador: Prof. Dr. Gustavo Henrique Henrique Matos Bezerra

JOÃO PESSOA - PB

2023

Catálogo na publicação
Seção de Catalogação e Classificação

M827o Morais, Matheus Araujo de.

Otimizando o tempo do atendimento hospitalar: um
solução para filas em hospitais / Matheus Araujo de
Morais. - João Pessoa, 2023.

85 f.

Orientação: Gustavo Henrique Matos Bezerra.
TCC (Graduação) - UFPB/INFORMATICA.

1. Engenharia de software. 2. Filas em hospitais. 3.
Gestão de filas de espera. I. Bezerra, Gustavo Henrique
Matos. II. Título.

UFPB/CI

CDU 004.05

AGRADECIMENTOS

Em primeiro lugar, à Deus, pois sem ele nada seria possível e a quem entrego todas as minhas conquistas.

Meus pais, Alda e Benedito, por todo o cuidado e amor que me deram durante toda a minha vida, obrigado por serem sempre atenciosos e pacientes e por toda educação que me deram.

Meus irmãos, Jerônimo e Jeffeson, obrigado por todos os conselhos e irmandade que tivemos ao longo da minha vida, e por cuidarem de mim.

Meu cachorro Barney, nosso fiel companheiro e protetor da casa, sempre nos procura nos momentos de dificuldade e faz do dia algo melhor.

Meus sogros, Mércia e Guido, a quem considero como meus segundos pais, obrigado por todo incentivo na minha trajetória no mundo da TI.

Meu orientador, Prof. Gustavo, por todos os conselhos e ensinamentos transmitidos durante o desenvolvimento da monografia.

Minha noiva, Ana Raquel, com quem logo me casarei, a quem escolhi para ser minha eterna companheira. Obrigado por todos os momentos ao meu lado, nos momentos de alegria e tristezas, de lazer e de estudos. Obrigado por ser quem você é, sem você nada disso seria possível.

RESUMO

A questão das filas de espera em hospitais é um problema constante para os brasileiros. Em muito se deve a esse problema, como ausência de profissionais de saúde, ausência de medicamentos nos estabelecimentos e falta de leitos para receber pacientes, o que torna-se um grande desafio para as autoridades do país para administrarem. Enquanto que os graves problemas na saúde pública não são solucionados, a engenharia de software surge como um viés para tratar a questão das filas de espera nos hospitais e clínicas de saúde pública. A engenharia de software surge como uma solução para lidar com os desafios complexos relacionados ao desenvolvimento de software, buscando garantir a entrega de produtos de alta qualidade. Ela se baseia na aplicação de princípios, métodos e práticas que abrangem todo o ciclo de vida do software, desde a concepção até a manutenção. Através da engenharia de software, é possível gerenciar de forma eficiente os requisitos do sistema, modelar e projetar a arquitetura adequada, implementar boas práticas de codificação, realizar testes para assegurar a funcionalidade e qualidade do software, além de cuidar da sua manutenção ao longo do tempo. Diante disso, o presente trabalho tem por objetivo a idealização de uma aplicação de software, a níveis de requisitos e arquitetura, capaz de gerenciar filas de espera em hospitais e/ou clínicas, utilizando dos conceitos e técnicas que a engenharia de software disponibiliza para gestão de projetos.

Palavras-chave: engenharia de software, filas em hospitais, gestão de filas de espera.

ABSTRACT

The issue of waiting lines in hospitals is a constant problem for Brazilians. Much is due to this problem, such as the lack of health professionals, the absence of medication in establishments and the lack of beds to receive patients, which becomes a major challenge for the authorized authorities in the country to administer. While the serious problems in public health are not solved, software engineering appears as a bias to deal with the issue of queues in hospitals and public health clinics. Software engineering emerges as a solution to deal with the complex challenges related to software development, seeking to ensure the delivery of high quality products. It is based on the application of principles, methods and practices that cover the entire software life cycle, from conception to maintenance. Through software engineering, it is possible to efficiently manage system requirements, model and design the appropriate architecture, implement good coding practices, perform tests to ensure the functionality and quality of the software, in addition to taking care of its maintenance throughout the lifetime. time. In view of this, the present work aims to develop a software application, at architectural levels, that is capable of efficiently managing queues in hospitals and/or clinics, using the concepts and techniques that software engineering has available for project management.

Keywords: software engineering, queues in hospitals, queue management.

LISTA DE ILUSTRAÇÕES

FIGURAS

Figura 1 - A estrutura de um documento de requisitos	12
Figura 2 - Diagrama de Casos de Uso	16
Figura 3 - Diagrama de Sequência.....	17
Figura 4 - Arquitetura de um sistema de controle robotizado de empacotamento	19
Figura 5 - Uma arquitetura cliente-servidor para uma biblioteca de filmes	20
Figura 6 - Processo de Prototipagem	21
Figura 7 - Diagrama do Gestor.....	34
Figura 8 - Diagrama do Paciente.....	35
Figura 9 - Diagrama de Classe	52
Figura 10 - Arquitetura Geral.....	54
Figura 11 - Arquitetura Cliente-Servidor	55
Figura 12 - Diagrama de Sequência de Autenticação	57
Figura 13 - Diagrama de Sequência de Entidades	59
Figura 14 - Diagrama de Sequência de Usuários.....	61
Figura 15 - Diagrama Sequência de Eventos	65
Figura 16 - Diagrama Sequência de Lista de Espera	69
Figura 17 - Realizar Novo Cadastro	73
Figura 18 - Realizar Login	73
Figura 19 - Listar Estabelecimentos	74
Figura 20 - Listar Eventos	75
Figura 21 - Entrar em Filas de Espera	75
Figura 22 - Atendimento do Usuário.....	76
Figura 23 - Sair de Filas de Espera	77
Figura 24 - Editar Dados Pessoais.....	77
Figura 25 - Realizar Login	78
Figura 26 - Editar Estabelecimentos	78
Figura 27 - Cadastrar/Editar Funcionário	79
Figura 28 - Remover Funcionário.....	80
Figura 29 - Cadastrar Evento	81
Figura 30 - Editar Evento	81
Figura 31 - Remover Evento	82

Figura 32 - Remover Paciente da Lista de Espera	82
---	----

TABELAS

Tabela 1 - Subcapítulos do Desenvolvimento	22
Tabela 2 - Glossário das Entidades	23
Tabela 3 - Requisitos Funcionais - Essencial	27
Tabela 4 - Requisitos Funcionais - Condicional.....	29
Tabela 5 - Requisitos Funcionais - Opcional	30
Tabela 6 - Requisitos Não Funcionais - Essencial	31
Tabela 7 - Requisitos Não Funcionais - Condicional	32
Tabela 8 - Requisitos Não Funcionais - Opcional.....	33
Tabela 9 - Casos de Uso	36
Tabela 10 - Descrição dos Casos de Uso - Realizar Novo Cadastro	38
Tabela 11 - Descrição dos Casos de Uso - Realizar Login	39
Tabela 12 - Descrição dos Casos de Uso - Listar Estabelecimentos.....	39
Tabela 13 - Descrição dos Casos de Uso - Listar Eventos	41
Tabela 14 - Descrição dos Casos de Uso - Entrar em Filas de Espera.....	42
Tabela 15 - Descrição dos Casos de Uso - Editar Dados Pessoais	43
Tabela 16 - Descrição dos Casos de Uso - Gerenciar Funcionários.....	44
Tabela 17 - Descrição dos Casos de Uso - Gerenciar Eventos.....	47
Tabela 18 - Descrição dos Casos de Uso - Gerenciar Estabelecimentos	49
Tabela 19 - Descrição dos Casos de Uso - Gerenciar Filas de Espera.....	50
Tabela 20 - Lista de Microserviços	53

LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS

IA	Inteligências Artificiais
TI	Tecnologia da Informação
R	Requisitos
F	Funcionais
NF	Não Funcional
FA	Fluxo Alternativo
E	Essencial
O	Opcional
C	Condicional
CRUD	Create, Read, Update, Delete

SUMÁRIO

1. Introdução	9
1.1. Problematização	10
1.2. Solução proposta	10
1.3. Descrição de cenário	11
2. Fundamentação Teórica	12
2.1. Documento de Requisitos	12
2.1.1. Especificação de Requisitos	13
2.1.1.1. Requisitos Funcionais.....	15
2.1.1.2. Requisitos não Funcionais.....	15
2.1.2. Casos de uso.....	16
2.1.3. Diagrama de Sequência	17
2.2. Arquitetura de Software	18
2.2.1. Arquitetura Cliente-Servidor.....	19
2.3. Prototipação.....	21
3. Desenvolvimento.....	22
3.1. Especificações de Requisitos.....	23
3.1.1. Glossário	23
3.1.2. Análise de Requisitos	26
3.1.2.1. Requisitos Funcionais.....	26
3.1.2.2. Requisitos não funcionais.....	30
3.1.3. Casos de Uso	33
3.1.3.1. Diagrama de casos de uso	34
3.1.3.2. Lista de Casos de Uso.....	35
3.1.3.3. Descrição de Casos de Uso.....	37
3.1.4. Análise de caso de uso.....	51
3.1.4.1. Diagrama de classes de análise	52
3.2. Arquitetura de Software	52
3.2.1. Microserviços do sistema.....	53
3.2.2. Diagrama Geral da Arquitetura do Sistema.....	54
3.2.3. Diagrama Detalhado da Arquitetura do sistema.....	55
3.2.4. Diagrama de Sequência	56

3.3. Prototipação.....	73
3.3.1. Paciente.....	73
3.3.2. Gestor.....	78
4. Conclusão e Considerações Futuras.....	83
4.1 Conclusão.....	83
4.2. Considerações Futuras	83
Referências Bibliográficas.....	85

1. Introdução

Praticamente todos os países, hoje em dia, dependem de sistemas complexos baseados em computadores. Infra-estruturas e serviços nacionais constam com sistemas baseados em computadores, e a maioria dos produtos elétricos inclui um computador e um software de controle. A manufatura e a distribuição industriais estão completamente automatizadas, assim como os sistemas financeiros. Portanto, produzir e manter o software dentro de custos adequados é essencial para o funcionamento da economia nacional e internacional (SOMMERVILLE, 2007, p.3).

Enquanto o surgimento da máquina de impressão aconteceu por volta de 1439 (século XV), com o alemão Johannes Gutenberg (1398-1468), as primeiras máquinas capazes de ler informações, processá-las e gerar uma resposta surgiram em meados do século XX, com Alan Turing. Ou seja, a humanidade esperou cerca de milhares de anos para desenvolver uma máquina capaz de reproduzir textos em larga escala, enquanto após 500(quinhetos) anos ela foi capaz de criar uma máquina capaz de ler e entender o que estava escrito. Desde então, a relação homem e máquina passou a ser cada vez mais íntima, chegando ao ponto da criação de tecnologia capazes de “pensar” e agir como seres humanos, as chamadas IA.

Embora os primeiros computadores eletrônicos apareceram somente no século XX, os primeiros programas de computador surgiram em 1843, através de um trabalho público por Ada Lovelace, uma sugestão de calcular os números de Bernoulli por meio da máquina de analítica de Charles Babbage. A partir daí, com o desenvolvimento dos computadores e o aperfeiçoamento do Hardware foi possível que nos tempos atuais os programas de computador, ou simplesmente, software se tornassem cada vez mais complexos e capazes de resolver problemas insolucionáveis por mentes humanas.

A presença de software no cotidiano é extremamente perceptível, a exemplo disso: computadores, smartphones, carros, aviões, televisores e outros inúmeros equipamentos que funcionam a partir de softwares. Segundo Morais Izabelly, associar o termo software a somente os programas de computador é uma visão restritiva. “Software não é apenas o programa, mas também todos os dados de documentação e configuração associados, necessários para que o programa opere corretamente” (MORAIS, 2017, p.3). A engenharia de software busca, permanentemente, estudar

melhores formas de desempenhar sua missão de contribuir para o sucesso das organizações através do desenvolvimento de sistemas de informação, e muitas são as abordagens metodológicas desenvolvidas neste intuito, as quais podem ser enquadradas em dois tipos de modelos: rigorosos e ágeis. Ao analisarmos esses modelos, percebemos que todos têm em comum o mesmo conjunto de premissas em sua elaboração, colocadas de forma mais ou menos explícita, dependendo do autor escolhido (KRUCHTEN, 2001; SUTHERLAND, 2001; GLASS, 2001; PAULK, 2001; SOMMERVILLE. 2003):

1.1. Problematização

As filas em hospitais públicos no Brasil são um problema crônico que afeta milhões de pessoas. Pacientes aguardam meses, ou até mesmo anos, por consultas, exames e procedimentos médicos, muitas vezes sem qualquer previsão de quando serão atendidos. Essa situação é resultado de uma série de fatores, como a falta de investimentos na saúde pública, a má gestão dos recursos disponíveis e a falta de profissionais qualificados para atender a demanda.

Segundo FILHO, o problema da fila de espera é pouco comentado dentro da comunidade médica, muito provavelmente isso se deve ao fato de parecer tratar-se de uma discussão fora das academias científicas, passando assim a responsabilidade às instituições públicas. Ainda FILHO, “é preciso salientar que o acesso equitativo, justo e universal aos serviços de saúde deve ser uma preocupação constante não só do governo como de todos os profissionais envolvidos no atendimento à rede pública”. Dessa maneira, é necessário que a questão das filas seja tratada com seriedade e responsabilidade.

1.2. Solução proposta

Diante do problema mencionado anteriormente, o presente trabalho idealiza um sistema que seja capaz de gerenciar as filas de espera com eficiência e finalidade de otimizar os atendimentos médicos em hospitais e/ou clínicas de saúde. O sistema deve permitir que seus usuários realizem “Check-in” a partir de qualquer dispositivo conectado à Internet e em qualquer lugar, apenas indicando o momento exato para

que o paciente possa sair do lugar que esteja e se dirija ao estabelecimento. Da mesma maneira, funcionários dos estabelecimentos poderão efetuar a gestão dos eventos disponíveis pelas entidades, como exames e consultas, dentre elas, cadastrar e/ou remover funcionários, além de efetuar a gestão das filas de espera. Além disso, é previsto que o sistema funcione em uma arquitetura de microsserviços.

1.3. Descrição de cenário

Jorge é um jovem adulto de 25 anos e que possui uma vida financeira independente, onde trabalha em um importante banco de sua cidade como analista financeiro. Além disso, mora sozinho em um apartamento em um bairro com ótima localização. Ele possui uma vida ativa, sempre praticando esportes e vai à academia com frequência, cerca de 5 dias por semana. Jorge também é uma pessoa preocupada com a saúde, ele procura ir ao médico constantemente, além de fazer exames em hospitais e clínicas médicas.

Embora more perto de clínicas médicas, Jorge prefere se dirigir a lugares mais distantes de sua residência, por conta disso, perde muito tempo no trânsito além das próprias filas de espera dos estabelecimentos médicos. Certo dia, ele estava dirigindo seu carro a caminho do trabalho e resolveu ligar a rádio para escutar o jornal matinal, este que estava transmitindo uma matéria que mencionava o tempo médio perdido pelos brasileiros no trânsito. No dia seguinte ele tinha uma consulta marcada com um oftalmologista, que seria no período da tarde e quando foi pela manhã, a secretária ligou para o smartphone dele e pediu que fizesse o download do aplicativo de fila de espera que a clínica utiliza e entrasse na fila de espera. Seguindo a orientação, Jorge baixou o aplicativo através da Play Store e iniciou o fluxo de cadastro, este que ao ser finalizado avisou-o para validar o mesmo na caixa de entrada do endereço eletrônico cadastrado, em seguida ele realizou o login na aplicação.

Uma vez logado, procurou, no aplicativo, a clínica em que ia se consultar, depois pela consulta com o médico dele e por fim selecionou a opção de entrar na fila de espera, e então na tela mostrou que sua posição era a 15º - décima quinta. Percebendo que o tempo para ser atendido ia demorar, voltou ao que estava fazendo no trabalho, algum tempo depois o celular vibrou e ao olhar a tela, percebeu que uma notificação havia sido enviada sinalizando que ele deveria se dirigir à clínica, todavia,

para confirmar a sua posição na fila de espera, abriu a tela do aplicativo e nela mostrava o trajeto mais rápido, e em seguida Jorge se dirigiu para o destino. Ao chegar lá, ele percebeu que só existia uma pessoa à sua frente e em alguns minutos o mesmo seria atendido.

2. Fundamentação Teórica

Esta seção tem como objetivo registrar o conhecimento teórico adquirido ao longo do desenvolvimento do projeto atual. A seção está organizada nas subseções: Especificação de Requisitos, Arquitetura de Software e Prototipação.

2.1. Documento de Requisitos

A especificação de requisitos é um processo crucial no desenvolvimento de software, responsável por identificar, coletar e documentar as necessidades, expectativas e restrições de um sistema ou produto. Seu objetivo principal é compreender e definir claramente o que o sistema deve fazer e quais requisitos devem ser atendidos.

Segundo Moraes e Zanin (2020), o documento de software é particular para cada sistema desenvolvido, ou seja, o mesmo apenas é feito para aquela aplicação a ser criada, não servindo para outros softwares. "...cada sistema tem suas características próprias, assim como metodologias utilizadas, ferramentas, dentre outros detalhes. Porém, o que pode ser comum em vários projetos diferentes é a estrutura do documento. Porém, o que pode ser comum em vários projetos diferentes é a estrutura do documento. Uma das possíveis organizações do documento de requisitos pode ser baseada em uma norma IEEE para documentos de requisitos (IEEE, 2009). Conforme Sommerville (2011, p. 64), essa é uma norma genérica que pode ser adaptada para usos específicos." (MORAIS e ZANIN, 2020, p.57). O quadro abaixo ilustra o padrão de documento a ser seguido, segundo Sommerville (2011, p.64).

Figura 1 - A estrutura de um documento de requisitos

Capítulo	Descrição
Prefácio	Deve definir os possíveis leitores do documento e descrever seu histórico de versões, incluindo uma justificativa para a criação de uma nova versão e um resumo das mudanças feitas em cada versão.
Introdução	Deve descrever a necessidade para o sistema. Deve descrever brevemente as funções do sistema e explicar como ele vai funcionar com outros sistemas. Também deve descrever como o sistema atende aos objetivos globais de negócio ou estratégicos da organização que encomendou o software.
Glossário	Deve definir os termos técnicos usados no documento. Você não deve fazer suposições sobre a experiência ou o conhecimento do leitor.
Definição de requisitos de usuário	Deve descrever os serviços fornecidos ao usuário. Os requisitos não funcionais de sistema também devem ser descritos nessa seção. Essa descrição pode usar a linguagem natural, diagramas ou outras notações compreensíveis para os clientes. Normas de produto e processos que devem ser seguidos devem ser especificados.
Arquitetura do sistema	Deve apresentar uma visão geral em alto nível da arquitetura do sistema previsto, mostrando a distribuição de funções entre os módulos do sistema. Componentes de arquitetura que são reusados devem ser destacados.
Especificação de requisitos do sistema	Deve descrever em detalhes os requisitos funcionais e não funcionais. Se necessário, também podem ser adicionados mais detalhes aos requisitos não funcionais. Interfaces com outros sistemas podem ser definidas.
Modelos do sistema	Pode incluir modelos gráficos do sistema que mostram os relacionamentos entre os componentes do sistema, o sistema e seu ambiente. Exemplos de possíveis modelos são modelos de objetos, modelos de fluxo de dados ou modelos semânticos de dados.
Evolução do sistema	Deve descrever os pressupostos fundamentais em que o sistema se baseia, bem como quaisquer mudanças previstas, em decorrência da evolução de hardware, de mudanças nas necessidades do usuário etc. Essa seção é útil para projetistas de sistema, pois pode ajudá-los a evitar decisões capazes de restringir possíveis mudanças futuras no sistema.
Apêndices	Deve fornecer informações detalhadas e específicas relacionadas à aplicação em desenvolvimento, além de descrições de hardware e banco de dados, por exemplo. Os requisitos de hardware definem as configurações mínimas ideais para o sistema. Requisitos de banco de dados definem a organização lógica dos dados usados pelo sistema e os relacionamentos entre esses dados.
Índice	Vários índices podem ser incluídos no documento. Pode haver, além de um índice alfabético normal, um índice de diagramas, de funções, entre outros pertinentes.

Fonte: Sommerville (2011, p.64)

Embora essa imagem ilustra um padrão que pode ser seguido para a maioria dos sistemas de software, ele pode sofrer alterações na sua estrutura. Segundo Sommerville, a informação a ser incluída em um documento de requisitos vai depender do sistema e da abordagem que será adotada durante o desenvolvimento.

Se uma abordagem evolutiva é adotada para um produto de software (por exemplo), o documento de requisitos deixará de fora muitos dos capítulos detalhados sugeridos. O foco será sobre a definição de requisitos de usuário e os requisitos não funcionais de alto nível de sistema. Nesse caso, os projetistas e programadores usam seu julgamento para decidir como atender aos requisitos gerais de usuário para o sistema. Sommerville (2011, p.64).

2.1.1. Especificação de Requisitos

A especificação de requisitos é a etapa em que os requisitos de usuário e de sistema são registrados em um documento específico. É fundamental que esses requisitos sejam redigidos de forma objetiva, evitando ambiguidades, de modo que possam ser compreendidos claramente por todos os envolvidos. Além disso, é importante que os requisitos abrangem tanto aspectos funcionais quanto não funcionais, de modo a permitir que usuários sem conhecimentos técnicos sobre a tecnologia possam compreendê-los. A clareza e a precisão na especificação dos requisitos contribuem para um melhor entendimento e alinhamento entre as partes interessadas no desenvolvimento do software.

Segundo Sommerville, os requisitos de sistema são uma extensão dos requisitos de usuário, sendo utilizados pelos engenheiros de software como base para o projeto do sistema. Eles acrescentam detalhes e fornecem explicações sobre como os requisitos de usuário devem ser atendidos pelo sistema. Os requisitos de sistema podem ser utilizados como parte do contrato para a implementação do sistema e devem conter uma especificação completa e detalhada de todas as funcionalidades do sistema. É importante ressaltar que os requisitos de sistema devem descrever apenas o comportamento externo do sistema e suas restrições operacionais, sem entrar em detalhes sobre o seu design ou implementação.

Segundo Moraes e Zanin, os requisitos podem ser classificados da seguinte maneira:

- **Essencial:** são indispensáveis para o funcionamento do software e sua implementação ocorre desde as primeiras versões do sistema. Eles são fundamentais e devem ser atendidos para que o software cumpra sua finalidade principal.
- **Importante:** são aqueles sem os quais o sistema não pode funcionar adequadamente, embora possam resultar em um funcionamento insatisfatório. Eles são cruciais para garantir o mínimo de funcionalidade do sistema.
- **Desejável:** não comprometem o funcionamento essencial do sistema e podem ser adiados ou até mesmo não desenvolvidos inicialmente. Eles representam funcionalidades ou melhorias adicionais que podem ser consideradas no futuro, mas não são cruciais para o produto final.

Todavia, a nomenclatura dessa classificação pode mudar de documento para documento.

2.1.1.1. Requisitos Funcionais

Requisitos funcionais são aqueles que descrevem as funcionalidades e as ações específicas que o sistema deve ser capaz de executar. Esses requisitos podem variar de acordo com o tipo de software, o perfil dos usuários e a abordagem adotada pela organização para escrevê-los. Quando expressos como requisitos de usuários, eles são frequentemente descritos de forma abstrata, de modo que possam ser facilmente compreendidos pelos próprios usuários, sem a necessidade de conhecimentos técnicos avançados. Essa abordagem visa garantir que os requisitos sejam claros e compreensíveis para todas as partes interessadas.

“Ao escrever requisitos funcionais, é importante considerar a perspectiva do usuário final e descrever as funcionalidades do sistema de uma maneira que seja facilmente compreendida e identificável pelos usuários. Isso contribui para a definição de expectativas claras e ajuda a garantir que o sistema atenda às necessidades dos usuários de forma intuitiva. Em princípio, a especificação dos requisitos funcionais de um sistema deve ser completa e consistente. Completude significa que todos os serviços requeridos pelo usuário devem ser definidos. Consistência significa que os requisitos não devem ter definições contraditórias. Na prática, para sistemas grandes e complexos, é praticamente impossível alcançar completude e consistência dos requisitos”, (Sommerville, 2011, p.60).

2.1.1.2. Requisitos não Funcionais

Os requisitos não funcionais são aqueles que não estão diretamente ligados aos serviços específicos oferecidos pelo sistema aos usuários. Em vez disso, eles estão relacionados a propriedades gerais do sistema, como confiabilidade, tempo de resposta e ocupação de recursos. Outra possibilidade é que os requisitos não funcionais estabeleçam restrições sobre a implementação do sistema, como capacidades dos dispositivos de entrada e saída ou formatos de dados utilizados nas interfaces com outros sistemas.

Os requisitos não funcionais abrangem características abrangentes do sistema como um todo, como desempenho, segurança e disponibilidade. Geralmente, esses requisitos são considerados mais críticos do que requisitos funcionais individuais. Os

usuários do sistema podem encontrar maneiras alternativas de contornar uma função específica que não atenda às suas necessidades, mas a não conformidade com um requisito não funcional pode tornar todo o sistema inutilizável. Por exemplo, se um sistema de aeronaves não atender aos requisitos de confiabilidade, ele não será certificado como seguro para operação. Da mesma forma, se um sistema de controle embutido não atender aos requisitos de desempenho, as funções de controle não funcionarão corretamente.

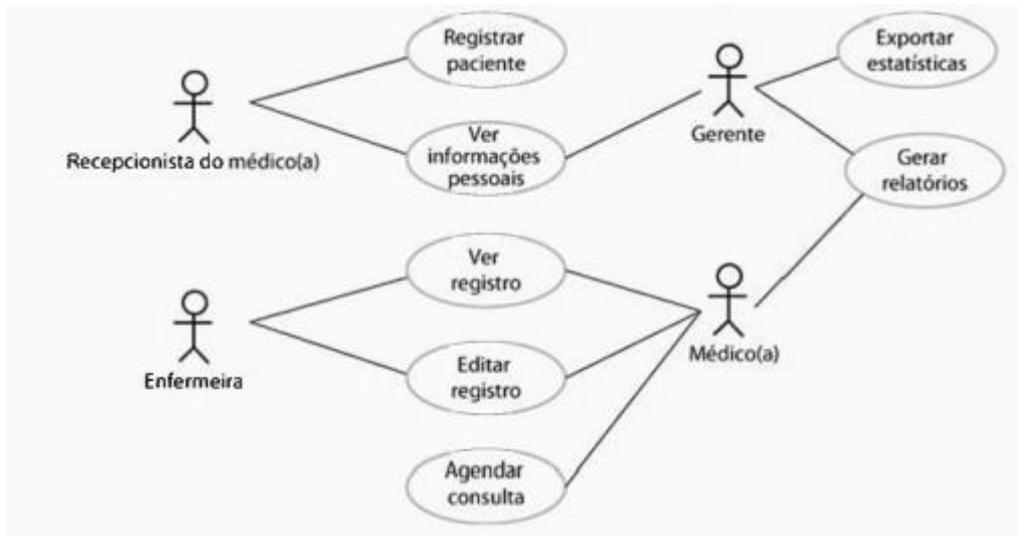
2.1.2. Casos de uso

Casos de uso são descrições em formas de narrativas textuais que são amplamente utilizadas para identificar requisitos. Segundo Morais e Zanin, o uso de casos de usos pode influenciar a análise, projeto, implementação, gestão de projeto e fase de teste do sistema.

“Os casos de uso são documentados por um diagrama de casos de uso de alto nível. O conjunto de casos de uso representa todas as possíveis interações que serão descritas nos requisitos de sistema. Atores, que podem ser pessoas ou outros sistemas, são representados como figuras 'palito'. Cada classe de interação é representada por uma elipse. Linhas fazem a ligação entre os atores e a interação. Opcionalmente, pontas de flechas podem ser adicionadas às linhas para mostrar como a interação se inicia.” (Sommerville, 2011, p.60).

Um caso de uso, em sua forma mais básica, tem a finalidade de identificar os atores envolvidos em uma interação e definir o tipo de interação em questão. Além disso, ele pode ser complementado por informações adicionais que descrevem de forma mais detalhada como ocorre a interação com o sistema. Essas informações adicionais podem ser apresentadas de diversas maneiras, como por meio de uma descrição textual ou por meio de modelos gráficos, como o diagrama de sequência ou o diagrama de estados da UML. Esses modelos visuais ajudam a representar de forma mais clara e visual a sequência de eventos e as possíveis transições de estados durante a interação com o sistema.

Figura 2 - Diagrama de Casos de Uso



Fonte: Sommerville (2011, p.75)

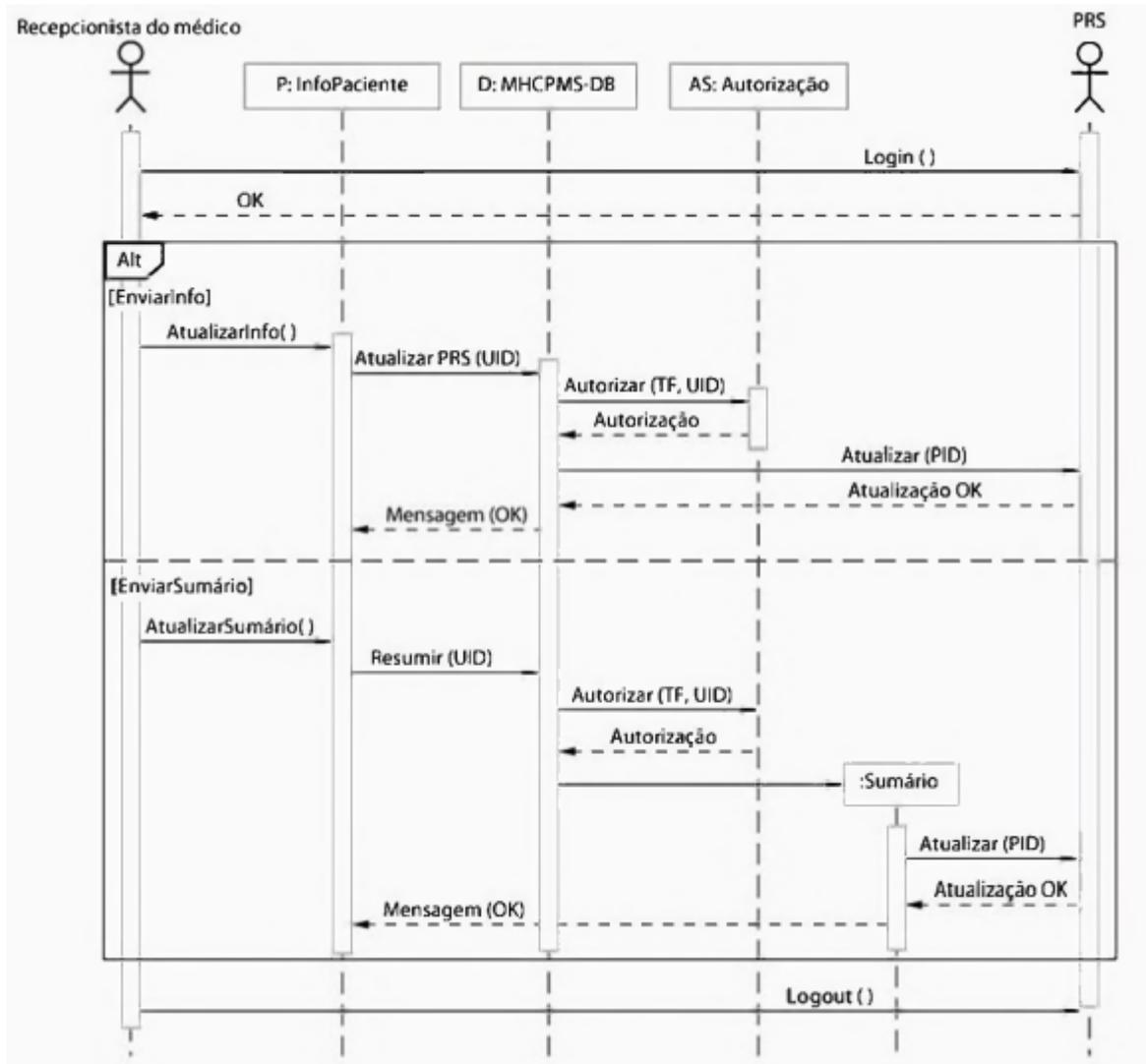
A figura 2, ilustra os casos de uso de um sistema de informações de pacientes.

2.1.3. Diagrama de Sequência

Os diagramas de sequência em UML são usados, principalmente, para modelar as interações entre os atores e os objetos em um sistema e as interações entre os próprios objetos. A UML tem uma sintaxe rica para diagramas de sequência, que permite a modelagem de vários tipos de interação (Sommerville, 2011, p.87).

Um diagrama de sequência é um tipo de diagrama de interação que mostra a interação entre objetos ou partes de um sistema, representando a ordem e o fluxo das mensagens trocadas entre eles ao longo do tempo. Ele descreve como os objetos colaboram entre si para realizar uma determinada funcionalidade ou cenário. Logo é bastante útil para visualizar o comportamento dinâmico de um sistema, destacando as interações entre os objetos envolvidos em um processo específico. Ele permite identificar a ordem em que as mensagens são enviadas e recebidas entre os objetos, bem como os eventos que desencadeiam essas interações.

Figura 3 - Diagrama de Sequência



Fonte: Sommerville (2011, p.88)

A Figura 3 de acordo com Sommerville:

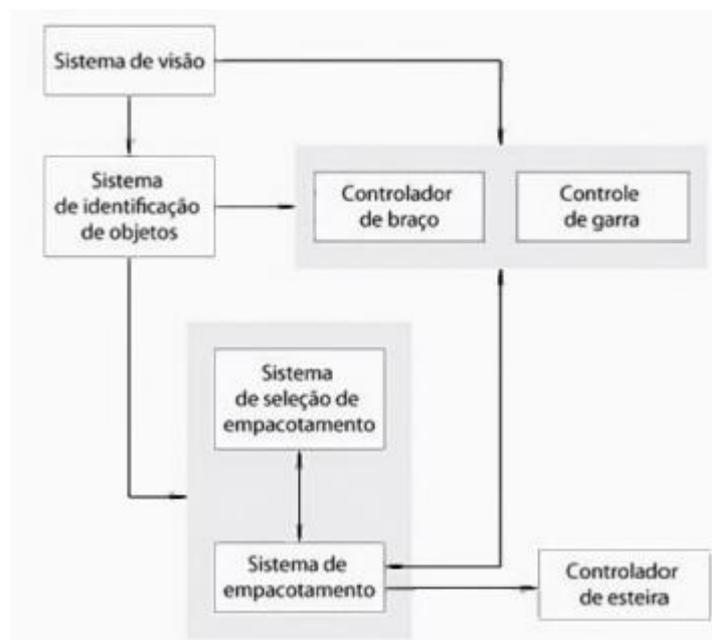
... é um segundo exemplo de diagrama de seqüência do mesmo sistema, que ilustra duas características adicionais. Estas são a comunicação direta entre os atores do sistema e a criação de objetos como parte de uma seqüência de operações. Nesse exemplo, um objeto do tipo Sumário é criado para armazenar os dados de resumo que serão enviados para o PRS (Sommerville, 2011, p.88).

2.2. Arquitetura de Software

O projeto de arquitetura de um software tem como objetivo principal a preocupação com a organização e a estrutura do sistema. Isso envolve identificar os

componentes principais que compõem o sistema e definir como esses componentes se relacionam entre si. O resultado desse processo de projeto é a descrição da organização do sistema em termos de um conjunto de componentes e suas interações. Essa descrição abrange a especificação dos componentes individuais, suas responsabilidades, interfaces e dependências. Além disso, também são estabelecidas as formas de comunicação e troca de informações entre os componentes.

Figura 4 - Arquitetura de um sistema de controle robotizado de empacotamento



Fonte: Sommerville (2011, p.104)

A Figura 4, ilustra de um sistema de controle robotizado de empacotamento.

Ela mostra um modelo abstrato da arquitetura de um sistema de controle robotizado de empacotamento, que mostra os componentes que precisam ser desenvolvidos. Esse sistema robotizado pode embalar diferentes tipos de objeto. Ele usa um componente de visão para selecionar objetos em uma esteira, identificar o tipo de objeto e selecionar o tipo correto de empacotamento. Em seguida, o sistema retira os objetos da esteira de entrega para serem empacotados. Então, o sistema coloca os objetos empacotados em outra esteira. O modelo de arquitetura mostra esses componentes e os relacionamentos entre eles (Sommerville, 2011, p.103).

2.2.1. Arquitetura Cliente-Servidor

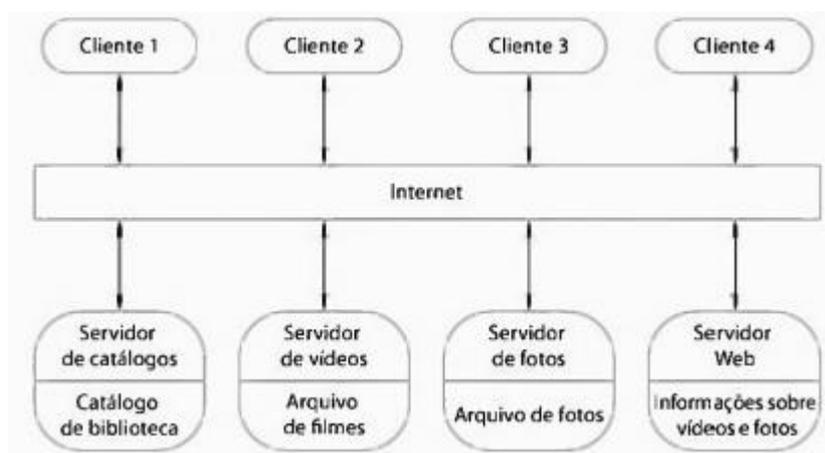
A arquitetura Cliente-Servidor é um modelo organizacional utilizado em sistemas de software, no qual um conjunto de serviços é disponibilizado por servidores que são acessados pelos clientes. De acordo com Sommerville, essa arquitetura é composta pelos seguintes componentes:

1. Conjunto de servidores: São os componentes responsáveis por oferecer os serviços aos clientes. Esses servidores são projetados para executar funções específicas e fornecer recursos ou processamento aos clientes que os solicitam.
2. Conjunto de clientes: São os componentes que utilizam os serviços oferecidos pelos servidores. Os clientes podem ser aplicativos, dispositivos ou usuários finais que interagem com o sistema para obter os benefícios e funcionalidades disponibilizadas pelos servidores.
3. Rede: É a infraestrutura que possibilita a comunicação entre os servidores e os clientes. A rede permite que os clientes enviem solicitações aos servidores e recebam as respostas correspondentes. Essa comunicação pode ocorrer em uma rede local (LAN) ou em uma rede mais ampla, como a internet.

Sommerville (2011) ressalta ainda que:

Arquiteturas cliente-servidor são normalmente consideradas arquiteturas de sistemas distribuídos, mas o modelo lógico de serviços independentes rodando em servidores separados pode ser implementado em um único computador. Um benefício importante é a separação e a independência. Serviços e servidores podem ser modificados sem afetar outras partes do sistema (Sommerville, 2011, p.103).

Figura 5 - Uma arquitetura cliente-servidor para uma biblioteca de filmes



Fonte: Sommerville(2011, p.114)

A Figura 5 segundo Sommerville (2011):

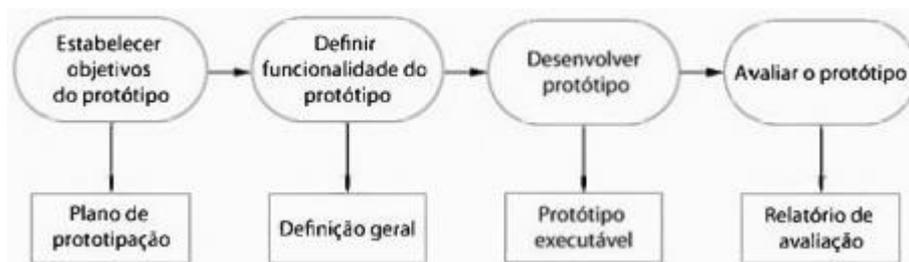
...é um sistema multiusuário baseado na Internet para fornecimento de uma biblioteca de filmes e fotos. Nesse sistema, vários servidores gerenciam e apresentam os diferentes tipos de mídia. Os quadros de vídeo precisam ser transmitidos de forma rápida e em sincronia, mas em resolução relativamente baixa. Eles podem ser comprimidos em um arquivo, de modo que o servidor de vídeo possa lidar com a compressão e descompressão de vídeo em diferentes formatos. As fotos, porém, devem permanecer em uma resolução alta; por isso, é conveniente mantê-las em um servidor separado”, (Sommerville, 2011, p.115).

2.3. Prototipação

A prototipação é o processo de criação de um protótipo, que é uma versão inicial ou modelo preliminar de um produto, sistema ou design. O objetivo da prototipação é desenvolver um artefato tangível que representa visualmente e funcionalmente as características principais do produto final. Segundo Sommerville, protótipos do sistema são úteis para que usuários possam ver como o sistema dá suporte ao trabalho. Ainda Sommerville, a utilização dele pode surgir novos requisitos e encontrar falhas ou melhorias no software.

Os objetivos da prototipação devem ser explicitados desde o início do processo. Estes podem ser o desenvolvimento de um sistema para prototipar a interface de usuário, o desenvolvimento de um sistema para validação dos requisitos funcionais do sistema ou o desenvolvimento de um sistema para demonstrar aos gerentes a viabilidade da aplicação. O mesmo protótipo não pode cumprir todos os objetivos. Se os objetivos não são declarados, a gerência ou os usuários finais podem não entender a função do protótipo. Consequentemente, eles podem não obter os benefícios que esperavam do desenvolvimento do protótipo (Sommerville, 2011, p.30)

Figura 6 - Processo de Prototipagem



Fonte: Sommerville (2011, p.30)

A Figura 6 ilustra o processo de criação de um protótipo.

O próximo estágio do processo é decidir o que colocar e, talvez mais importante ainda, o que deixar de fora do sistema de protótipo. Para reduzir os custos de prototipação e acelerar o cronograma de entrega, pode-se deixar alguma funcionalidade fora do protótipo. Você pode optar por relaxar os requisitos não funcionais, como tempo Figura 2.9 O processo de desenvolvimento de protótipo estabelecer objetivos do protótipo Plano de prototipação definir funcionalidade do protótipo Definição geral »^Avaliar o protótipo^ Relatório de avaliação de resposta e utilização de memória. Gerenciamento e tratamento de erros podem ser ignorados, a menos que o objetivo do protótipo seja estabelecer uma interface de usuário. Padrões de confiabilidade e qualidade de programa podem ser reduzidos. O estágio final do processo é a avaliação do protótipo. Durante esse estágio, provisões devem ser feitas para o treinamento do usuário, e os objetivos do protótipo devem ser usados para derivar um plano de avaliação. Os usuários necessitam de um tempo para se sentir confortáveis com um sistema novo e para se situarem em um padrão normal de uso. Uma vez que estejam usando o sistema normalmente, eles descobrem erros e omissões de requisitos (Sommerville, 2011, p.31)

3. Desenvolvimento

Este capítulo descreve as características e definições acerca do modelo que compõe o sistema idealizado como solução para a problemática apontada na introdução do presente documento. Este que é dividido em:

Tabela 1 - Subcapítulos do Desenvolvimento

Nome	Descrição
Especificação de Requisitos	Esta seção é responsável por descrever de forma detalhada as funcionalidades, características e restrições necessárias para atender a solução do problema.
Arquitetura de Software	Esta seção é responsável por descrever de maneira detalhada a estrutura global do sistema. Isso abrange a identificação e a explanação dos principais componentes do sistema, bem como as interações entre eles. .
Prototipagem	Esta seção é responsável por descrever o processo de prototipação do sistema, em outras palavras, uma versão simplificada do software que permite aos desenvolvedores e usuários experimentarem e interagirem com as funcionalidades

	básicas do sistema.
--	---------------------

Fonte: Elaborado pelo autor

3.1. Especificações de Requisitos

Esta seção descreve de maneira abrangente as funcionalidades, características e restrições essenciais para atender à solução do problema. Nela, são fornecidos detalhes minuciosos e específicos que definem o escopo e as necessidades do sistema ou software a ser desenvolvido. Essas informações são cruciais para garantir que o produto final atenda aos requisitos e expectativas dos usuários e stakeholders envolvidos no projeto. Este que é subdividido em: Glossário, Análise de Requisitos, Casos de Uso e Análise de Casos de Uso.

3.1.1. Glossário

Esta seção do documento tem como objetivo fornecer uma lista e descrição das entidades abordadas durante o desenvolvimento do projeto atual. Ela abrange os termos e atributos que compõem essas entidades, incluindo termos técnicos específicos do domínio do projeto, acrônimos, jargões e qualquer outro vocabulário especializado que seja relevante para a elucidação de requisitos.

Tabela 2 - Glossário das Entidades

Entidade Usuário: representa as informações dos usuários do app.	
Atributos	Detalhes
Nome	65535 posições - AN
E-mail	65535 posições - AN
CPF/CNPJ	65535 posições - N
RG	65535 posições - N

Data de Nascimento	50 posições - AN
Telefone	65535 posições - N
Profissão	65535 posições - AN
Número Conselho	65535 posições - N
Data Cadastro	50 posições - AN
Data Alteração	50 posições - AN

Entidade Estabelecimento: representa as informações dos estabelecimentos do app.

Atributos	Detalhes
Nome	65535 posições - AN
Nome Fantasia	65535 posições - AN
CNPJ	50 posições - N
Telefone	50 posições - N
E-mail	65535 posições - AN
Data de Abertura	50 posições - AN
Data Cadastro	50 posições - AN
Data Alteração	50 posições - AN

Entidade Endereço: representa as informações de endereços das entidades usuário e estabelecimento do app.

Atributos	Detalhes
-----------	----------

Rua	65535 posições - AN
Número	30 posições - N
Bairro	65535 posições - AN
Complemento	65535 posições - AN
Cidade	65535 posições - AN
Estado	65535 posições - AN
CEP	50 posições - AN

Entidade Login: representa as informações de login dos usuários do app.

Atributos	Detalhes
Username	65535 posições - AN
Senha	65535 posições - AN

Entidade Eventos: representa as informações de eventos dos estabelecimentos do app.

Atributos	Detalhes
Nome	65535 posições - AN
Quantidade de Vagas	50 posições - N
Profissional	65535 posições - AN
Horário	65535 posições - AN
Sala	65535 posições - AN

Entidade Tipo Usuário: representa os tipos de usuários do app.	
Atributos	Detalhes
Nome	65535 posições - AN
Função	65535 posições - AN
Entidade Tipo Estabelecimento: representa os tipos de estabelecimentos do app.	
Atributos	Detalhes
Nome	65535 posições - AN

Fonte: Elaborado pelo autor.

3.1.2. Análise de Requisitos

Esta seção do documento tem como objetivo fornecer um detalhamento dos requisitos que foram considerados para o sistema. Nesta, é possível encontrar a listagem de requisitos funcionais e não funcionais, nos quais ambos são separados por ordem de prioridade: Essencial, Condicional e Opcional, e as respectivas descrições de funcionalidade.

3.1.2.1. Requisitos Funcionais

Nesta subseção, é possível encontrar a listagem de requisitos funcionais, os quais são separados por ordem de prioridade: essenciais, condicionais e opcionais, juntamente com suas respectivas descrições de funcionalidade. O objetivo principal é fornecer uma visão clara e completa dos requisitos que devem ser atendidos pelo sistema.

a) Essencial

Tabela 3 - Requisitos Funcionais - Essencial

R-F-E-001: O usuário deve se cadastrar
Informações: Através de uma tela de cadastro, o usuário deverá fornecer alguns dados pessoais para que o processo de cadastro seja concluído.
R-F-E-002: O usuário deve se autenticar
Informações: Através de uma tela de Login, o usuário deverá fornecer seus dados de acesso para poder acessar o software.
R-F-E-003: Listar os Estabelecimentos
Informações: Ao entrar na tela de Estabelecimentos, o software deverá listar os Estabelecimentos disponíveis na tela.
R-F-E-004: Buscar por Estabelecimento
Informações: O usuário deverá ter a opção de pesquisar por um Estabelecimento a partir de informações do próprio Estabelecimento.
R-F-E-005: Listar Eventos
Informações: Ao selecionar um Estabelecimento, o sistema deverá listar os Eventos disponíveis.
R-F-E-006: Buscar por Eventos
Informações: O usuário deverá ter a opção de pesquisar por um Evento a partir de informações do próprio Evento.

R-F-E-007: Entrar na fila de um Evento

Informações: O usuário deverá ter a opção de entrar na fila de um determinado Evento.

R-F-E-008: Alterar dados pessoais

Informações: A partir da tela Dados Pessoais, o usuário deverá ter a opção de alterar os dados pessoais próprios.

R-F-E-009: Gerenciar Eventos

Informações: Os gestores dos Estabelecimentos poderão gerenciar, cadastrar novos e/ou editar, os Eventos cadastrados.

R-F-E-010: Gerenciar funcionários de Estabelecimentos

Informações: Os gestores dos Estabelecimentos poderão gerenciar, cadastrar novos e/ou editar, os funcionários cadastrados.

R-F-E-011: Gerencia dados dos Estabelecimentos

Informações: Os gestores dos Estabelecimentos poderão gerenciar, cadastrar novos e/ou editar, os Estabelecimentos cadastrados.

R-F-E-011: Gerencia as Filas

Informações: Os gestores dos Estabelecimentos poderão gerenciar, adicionar e/ou remover, as os usuários das Filas dos Eventos.

R-F-E-012: Visualizar posição na fila

Informações: Os gestores dos Estabelecimentos poderão gerenciar, adicionar e/ou remover, as os usuários das Filas dos Eventos.

Fonte: Elaborado pelo autor.

b) Condicional

Tabela 4 - Requisitos Funcionais - Condicional

R-F-C-001: Login por redes sociais

Informações: O sistema deverá ter a opção de Login através das redes sociais, como, Facebook, Google, Instagram.

R-F-C-003: Espaço de discussão

Informações: O sistema deverá ter um espaço para discussão, dúvidas ou informações.

R-F-C-004: Chat para Comunicação

Informações: O sistema deverá ter um espaço para a comunicação entre pacientes e funcionários.

R-F-C-005: Notificar o Usuário

Informações: Ao se aproximar da vez do usuário, o sistema deverá avisá-lo para se dirigir ao local do Evento.

Fonte: Elaborado pelo autor.

c) Opcional

Tabela 5 - Requisitos Funcionais - Opcional

R-F-O-001: Notificação periódica para retornar ao sistema
Informações: O sistema poderá notificar o usuário após um determinado período de tempo.
R-F-O-002: Rank de Usuários
Informações: O sistema poderá gerar uma lista dos usuários que mais acessam a plataforma.
R-F-O-003: Gerar relatórios de Eventos
Informações: O sistema poderá gerar relatórios acerca dos eventos de uma Estabelecimento.
R-F-O-004: Feedback do atendimento
Informações: O sistema deve permitir que os pacientes forneçam opiniões sobre o atendimento

Fonte: Elaborado pelo autor.

3.1.2.2. Requisitos não funcionais

Nesta subseção, é possível encontrar a listagem de requisitos não funcionais, os quais são separados por ordem de prioridade: essenciais, condicionais e opcionais, juntamente com suas respectivas descrições de funcionalidade. O objetivo principal é

fornecer uma visão clara e completa dos requisitos que devem ser atendidos pelo sistema.

a) Essencial

Tabela 6 - Requisitos Não Funcionais - Essencial

R-NF-E-001: A aplicação só funcionará On-line
Informações: A aplicação deverá funcionar apenas com acesso a internet.
R-NF-E-002: Atualização em tempo real
Informações: A aplicação deverá atualizar a fila de espera em tempo real.
R-NF-E-004: Back-end em Java
Informações: A tecnologia utilizada para desenvolver o sistema back-end deverá ser em Spring Boot Java.
R-NF-E-005: Front-end em VueJs
Informações: A tecnologia utilizada para desenvolver o sistema front-end deverá ser em VueJs Javascript.
R-NF-E-006: Padrão POO
Informações: O paradigma de linguagem de programação utilizado no projeto deverá ser o Orientado à Objetos.
R-NF-E-006: Padrão de Design Bootstrap

Informações: O padrão de design utilizado nas telas da aplicação deverá ser o Bootstrap v.5.0. ou melhor

Fonte: Elaborado pelo autor.

b) Condicional

Tabela 7 - Requisitos Não Funcionais - Condicional

R-NF-C-001: Carregamento rápido

Informações: A aplicação deverá ser fluída, com carregamento rápido das telas, maximizando o tempo de transição entre componentes.

R-NF-C-002: Tempo de aprendizagem

Informações: A aplicação deverá ter um curto tempo de aprendizagem.

R-NF-C-003: Uso intuitivo

Informações: A aplicação deve ser intuitiva, com foco em telas objetivas.

R-NF-C-004: Portabilidade

Informações: O software deve ser desenvolvido pensando na portabilidade dele futuramente para o ambiente mobile Android e/ou IOS.

R-NF-C-005: Transição fluída entre abas

Informações: A mudança de abas deverá ser visualmente fluída.

R-NF-C-006: Testes automatizados
Informações: Utilizando JUnit.

Fonte: Elaborado pelo autor.

c) Opcional

Tabela 8 - Requisitos Não Funcionais - Opcional

R-NF-O-001: Disponibilização da aplicação em lojas
Informações: Incluir a aplicação plataformas como a Apple Store ou Play Store.

Fonte: Elaborado pelo autor.

3.1.3. Casos de Uso

Esta seção do documento tem como objetivo fornecer uma descrição das interações funcionais entre um sistema e seus usuários, oferecendo uma visão detalhada de como o sistema deve ser utilizado para atender às necessidades dos usuários.

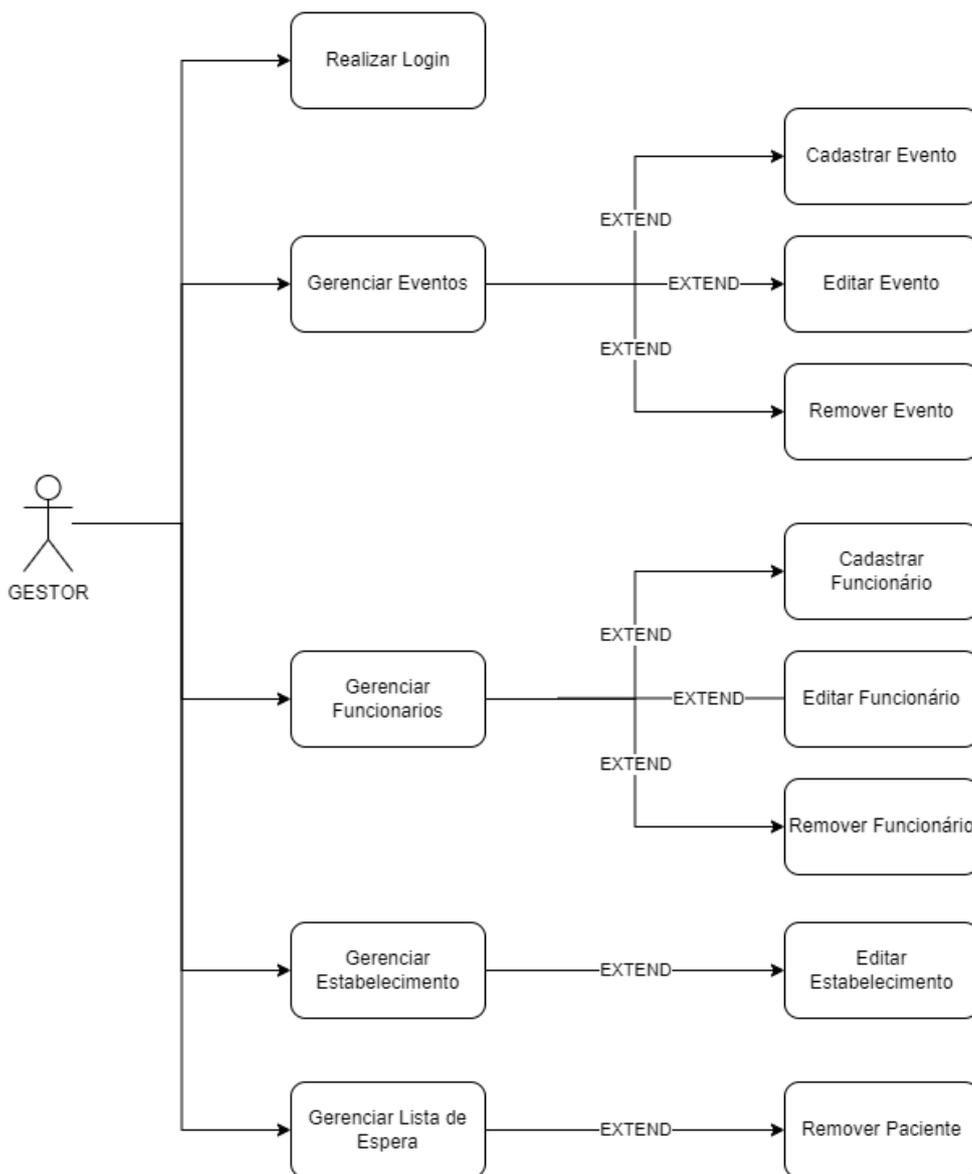
Ela é composta por três elementos principais:

1. **Diagrama de Casos de Uso:** Uma representação gráfica que descreve os diferentes casos de uso e as interações entre os atores (Paciente e Gestor). O diagrama de casos de uso proporciona uma visão geral das funcionalidades e das relações entre os atores e o sistema.
2. **Lista de Casos de Uso:** Uma subseção que tem como objetivo listar e descrever brevemente os casos de uso que estão ilustrados nos diagramas de caso de uso da seção 3.1.3.1. Essa lista fornece uma visão geral das funcionalidades disponíveis no sistema.
3. **Descrição dos casos de uso:** Essa subseção tem como objetivo descrever detalhadamente os casos de uso das seções 3.1.3.1 e 3.1.3.2. Ela fornece informações detalhadas sobre cada caso de uso, incluindo fluxos principais e alternativos, assim como os atores envolvidos na funcionalidade específica.

3.1.3.1. Diagrama de casos de uso

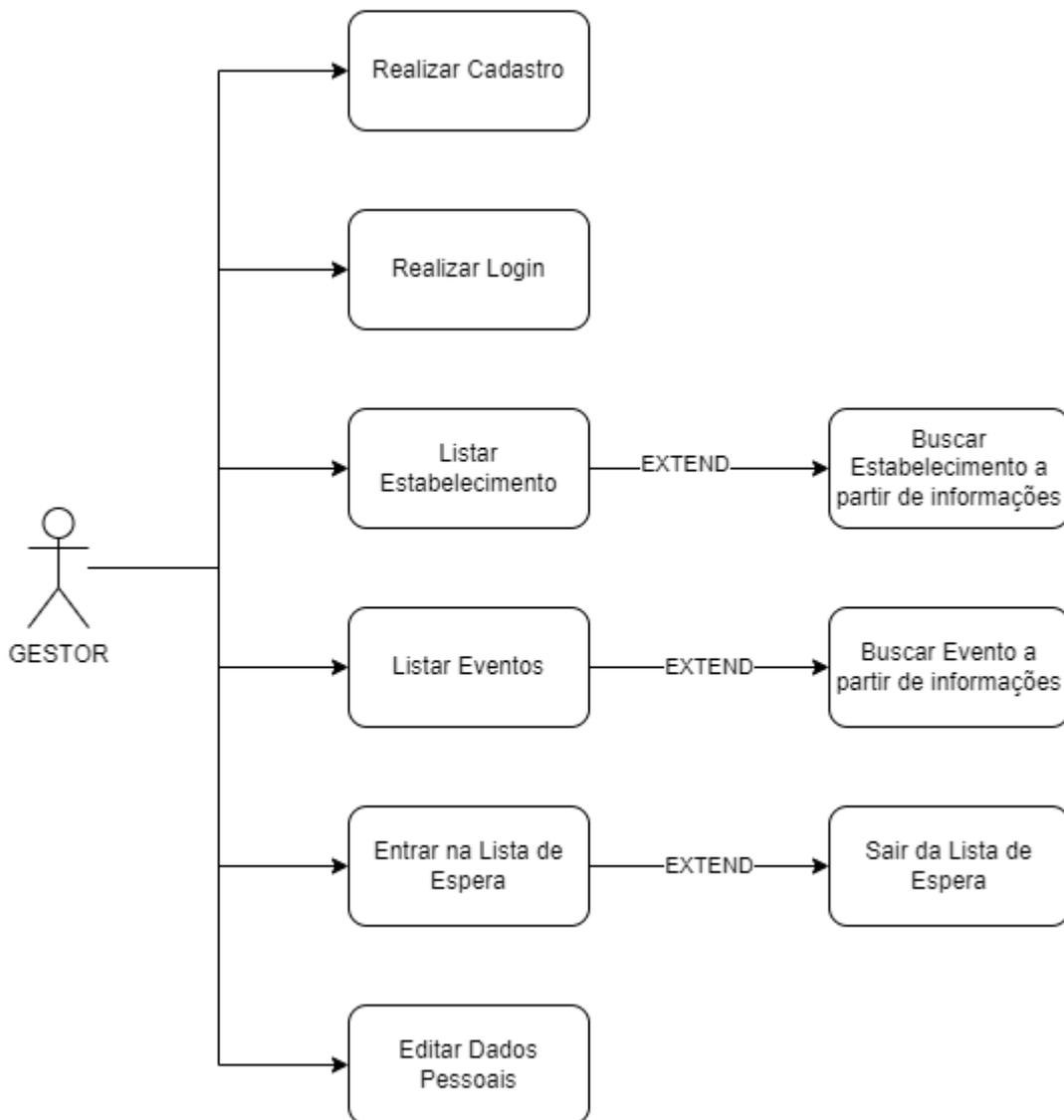
Esta subseção, apresenta os diagramas de casos de uso, que fornecem uma representação gráfica das interações entre os atores definidos pelo sistema e os diferentes casos de uso. Esses diagramas ilustram de forma visual as funcionalidades e as relações entre os atores e o sistema, permitindo uma compreensão clara das interações.

Figura 7 - Diagrama do Gestor



Fonte: Elaborado pelo autor

Figura 8 - Diagrama do Paciente



Fonte: Elaborado pelo autor

3.1.3.2. Lista de Casos de Uso

Esta seção tem como objetivo listar e fornecer uma descrição geral de cada Caso de Uso apresentado na seção 3.1.3.1. A descrição é organizada em uma tabela que inclui as seguintes colunas:

- **ID:** Identificador único atribuído à funcionalidade.
- **Nome:** Nome dado à funcionalidade.

- **Descrição:** Resumo descritivo da funcionalidade.
- **Prioridade:** Ordem de prioridade da funcionalidade.

Através dessa tabela, é possível ter uma visão geral e sistemática das funcionalidades, facilitando a compreensão das diferentes características e importâncias atribuídas a cada caso de uso.

Tabela 9 - Casos de Uso

ID	Nome	Descrição	Prioridade
UC-E-001	Realizar Novo Cadastro	O sistema deve permitir que o usuário realize o cadastro. Em caso de Funcionários, o gestor deverá efetuar o cadastro.	Essencial
UC-E-002	Realizar Login	O sistema deve permitir que o usuário realize o login.	Essencial
UC-E-003	Listar Estabeleciment o	O sistema deve permitir que os Pacientes possam listar os Estabelecimentos cadastrados.	Essencial
UC-E-004	Listar Eventos	O sistema deve permitir que os Pacientes possam listar os Eventos de cada Estabelecimento cadastrada.	Essencial
UC-E-005	Entrar em Filas de Espera	O sistema deve permitir que os Pacientes possam entrar em Listas de Espera dos Eventos disponíveis por cada Estabelecimento.	Essencial
UC-E-006	Editar Dados Pessoais	O sistema deve permitir que os Pacientes possam revisar e alterar seus dados pessoais cadastrados na aplicação.	Essencial

UC-E-007	Gerenciar Funcionários	O sistema deve permitir que os Gestores possam realizar a manutenção dos Funcionários, tais como, adicionar e remover funcionários.	Essencial
UC-E-008	Gerenciar Eventos	O sistema deve permitir que os Gestores possam realizar a manutenção dos Eventos.	Essencial
UC-E-009	Gerenciar Estabelecimentos	O sistema deve permitir que os Gestores possam realizar a manutenção de Estabelecimentos.	Essencial
UC-E-0010	Gerenciar Filas de Espera	O sistema deve permitir que os Gestores possam realizar a manutenção de Filas de Espera, como, retirar Pacientes, passar a vez e/ou encerrar a Fila.	Essencial

Fonte: Elaborado pelo autor

3.1.3.3. Descrição de Casos de Uso.

Esta seção tem como objetivo listar e fornecer uma descrição geral de cada Caso de Uso apresentado na seção 3.1.3.1. A descrição é organizada em uma tabela que inclui as seguintes colunas:

- **Importância:** Indica a prioridade da funcionalidade.
- **Atores:** Define o perfil dos usuários permitidos para acessar o serviço.
- **Pré-condições:** Estabelece os requisitos mínimos necessários para utilizar o serviço.
- **Pós-condições:** Descreve o estado final após a conclusão do serviço.
- **Fluxo principal:** Apresenta o passo a passo para executar o serviço.

- **Fluxo alternativo:** Mostra um caminho alternativo para executar o serviço, caso necessário.
- **Fluxo de exceção:** Detalha os passos para sair do fluxo principal e/ou alternativo em casos de exceção.
- **Regras de negócio:** Compreende um conjunto de diretrizes, restrições ou condições que definem o serviço e suas operações.

Tabela 10 - Descrição dos Casos de Uso - Realizar Novo Cadastro.

UC-E-001	Realizar Novo Cadastro			
Importância:	Essencial			
Atores:	Pacientes			
Pré-condições:	1. Acesso à internet			
Pós-condições	Seguindo o fluxo normal a aplicação realizará o cadastro do Usuário.			
Fluxo principal:	<ol style="list-style-type: none"> 1. Abrir a Tela de Login. 2. Clicar na opção “Cadastrar-se”. 3. Preencher os dados requisitados pela aplicação. 4. Clicar em “Realizar Cadastro”. 5. Confirmar o cadastro. 			
Fluxo alternativo:	Não possui			
Fluxo de exceção:	<table border="1"> <tr> <td>UC-E-001.FE-001</td> <td> <ol style="list-style-type: none"> 1. Clicar em “Cancelar”. 2. Confirmar a opção de retorno. </td> </tr> </table>	UC-E-001.FE-001	<ol style="list-style-type: none"> 1. Clicar em “Cancelar”. 2. Confirmar a opção de retorno. 	
UC-E-001.FE-001	<ol style="list-style-type: none"> 1. Clicar em “Cancelar”. 2. Confirmar a opção de retorno. 			
Regras de negócios:	<ol style="list-style-type: none"> 1. Possuir acesso a internet. 2. Fornecer os dados requisitados. 3. Confirmar a operação. 			

	4. Receber confirmação de cadastro.
--	-------------------------------------

Fonte: Elaborado pelo autor

Tabela 11 - Descrição dos Casos de Uso - Realizar Login.

UC-E-002	Realizar Login	
Importância:	Essencial	
Atores:	Pacientes e Gestores	
Pré-condições:	1. Possuir um cadastro ativo.	
Pós-condições:	O usuário poderá realizar o login na aplicação e realizar as atividades permitidas.	
Fluxo principal:	<ol style="list-style-type: none"> 1. Preencher o formulário de Login, fornecendo os dados necessários. 2. Clicar em “Entrar” 	
Fluxo alternativo:		
	UC-E-002.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-001. 2. Seguir o fluxo principal.
Fluxo de exceção:	Não possui	
Regras de negócios:	<ol style="list-style-type: none"> 1. Possuir cadastro ativo na aplicação. 2. Fornecer os dados necessários. 3. Caso não possua cadastro, seguir o fluxo de UC-E-001. 	

Fonte: Elaborado pelo autor

Tabela 12 - Descrição dos Casos de Uso - Listar Estabelecimentos

UC-E-003	Listar Estabelecimentos
-----------------	--------------------------------

Importância:	Essencial	
Atores:	Pacientes	
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir um cadastro ativo. 2. Estar logado 	
Pós-condições:	Possibilidade de visualizar os Estabelecimentos cadastrados na aplicação.	
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar no botão “Entidades”. 	
Fluxo alternativo:	UC-E-003.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Entidades”. 3. Digitar no campo de busca o nome do Estabelecimento. 4. Clicar em “Buscar”.
	UC-E-003.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Dados Pessoais”.
Fluxo de exceção:	UC-E-003.FE-002	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar em “Sair”. 3. Confirmar a operação de saída.
Regras de negócios:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Escolher a opção para visualizar os Estabelecimentos. 3. Buscar Estabelecimento a partir do nome. 	

Fonte: Elaborado pelo autor

Tabela 13 - Descrição dos Casos de Uso - Listar Eventos.

UC-E-004	Listar Eventos	
Importância:	Essencial	
Atores:	Pacientes e Gestores	
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir um cadastro ativo. 2. Estar logado. 	
Pós-condições:	Possibilidade de visualizar os Eventos cadastros pelos Estabelecimentos.	
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-003. 2. Clicar na opção “Visualizar Entidades”. 3. Clicar no nome de um Estabelecimento dentre as opções disponíveis. 	
Fluxo alternativo:	UC-E-004.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo de UC-E-003. 2. Clicar no nome de um Estabelecimento dentre as opções disponíveis. 3. Digitar o nome de um Estabelecimento no campo Busca. 4. Clicar no botão “Buscar”.
	UC-E-004.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo de UC-E-004. 2. Clicar no botão “Voltar”.
Fluxo de exceção:	UC-E-004.FE-002	<ol style="list-style-type: none"> 1. Seguir o fluxo de UC-E-004.FA-001. 2. Clicar no botão “Voltar”.

Regras de negócios:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-003. 2. Escolher a opção para visualizar os Estabelecimentos. 3. Buscar Estabelecimentos a partir do nome. 4. Buscar Eventos a partir do nome.
---------------------	---

Fonte: Elaborado pelo autor

Tabela 14 - Descrição dos Casos de Uso - Entrar em Filas de Espera.

UC-E-005	Entrar em Filas de Espera	
Importância:	Essencial	
Atores:	Paciente	
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir acesso a internet. 2. Realizar o Login. 3. Escolher um dos Estabelecimentos disponíveis. 4. Escolher um dos Eventos disponíveis. 	
Pós-condições:	Possibilidade de entrar em uma Fila de Espera para um dos Eventos disponíveis do Estabelecimento escolhido.	
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-003. 2. Escolher um dos Eventos disponíveis. 3. Clicar em “Entrar na Fila de Espera”. 	
Fluxo alternativo:	UC-E-005.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-004.FA-001. 2. Selecionar um dos Eventos encontrados pela opção de busca. 3. Clicar em “Entrar na Fila de Espera”.
	UC-E-005.FE-	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-005. 2. Clicar na opção “Voltar”.
Fluxo de exceção:	UC-E-005.FE-	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-005. 2. Clicar na opção “Voltar”.

	001	
	UC-E-005.FE-002	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-005.FA-001. 2. Clicar na opção "Voltar".
Regras de negócios:		<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-003. 2. Escolher um Evento disponível. 3. Selecionar a opção para entrar na Lista de Espera

Fonte: Elaborado pelo autor

Tabela 15 - Descrição dos Casos de Uso - Editar Dados Pessoais.

UC-E-006	Editar Dados Pessoais	
Importância:	Essencial	
Atores:	Pacientes e Gestores	
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir um cadastro ativo. 2. Estar logado. 	
Pós-condições:	Possibilidade de editar as informações pessoais do usuário.	
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar em "Editar Dados". 3. Modificar os campos referentes aos dados pessoais dos usuários. 4. Clicar em "Salvar". 	
Fluxo alternativo:	NÃO	

	POSSUI	
Fluxo de exceção:	UC-E-006.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar em “Editar Dados”. 3. Clicar em “Voltar”. 4. Clicar em “Confirmar Operação”.
Regras de negócios:	<ol style="list-style-type: none"> 1. Realizar Login. 2. Selecionar opção para editar os dados pessoais. 3. Modificar os dados pessoais do usuário. 	

Fonte: Elaborado pelo autor

Tabela 16 - Descrição dos Casos de Uso - Gerenciar Funcionários.

UC-E-007	Gerenciar Funcionários
Importância:	Essencial
Atores:	Gestores
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir cadastro ativo como Gestor. 2. Estar vinculado a algum Estabelecimento. 3. Estar logado.
Pós-condições:	<p>Possibilidade de gerenciar os Funcionários cadastrados e com vínculos ativos com um Estabelecimento. Dentre as funções:</p> <ol style="list-style-type: none"> 1. Adicionar novo Funcionário. 2. Remover Funcionário. 3. Visualizar informações do Funcionário.
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Funcionários”.
Fluxo	

alternativo:	<table border="1"> <tr> <td>UC-E-007.FA-001</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Remover Funcionário”. 5. Clicar em “Confirmar Operação”. </td> </tr> </table>	UC-E-007.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Remover Funcionário”. 5. Clicar em “Confirmar Operação”.
	UC-E-007.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Remover Funcionário”. 5. Clicar em “Confirmar Operação”. 	
	<table border="1"> <tr> <td>UC-E-007.FA-002</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Escolher a opção “Adicionar novo Funcionário”. 4. Inserir as informações requisitadas pelo sistema. 5. Clicar em “Salvar”. </td> </tr> </table>	UC-E-007.FA-002	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Escolher a opção “Adicionar novo Funcionário”. 4. Inserir as informações requisitadas pelo sistema. 5. Clicar em “Salvar”.
UC-E-007.FA-002	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Escolher a opção “Adicionar novo Funcionário”. 4. Inserir as informações requisitadas pelo sistema. 5. Clicar em “Salvar”. 		
<table border="1"> <tr> <td>UC-E-007.FA-003</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Visualizar Informações”. </td> </tr> </table>	UC-E-007.FA-003	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Visualizar Informações”. 	
UC-E-007.FA-003	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Visualizar Informações”. 		

| Fluxo de exceção: | | | | |------------------------|--| | UC-E-007.FE-001 | <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Funcionários”. 3. Clicar em “Voltar”. | |------------------------|--| |
| | | | |----------------------------|---| | UC-E-007.FA-001.FE- | <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clica na opção “Funcionários”. 3. Selecionar um Funcionário da lista de | |----------------------------|---| |

	<p>001</p> <p>Funcionários.</p> <ol style="list-style-type: none"> 4. Clicar no botão “Remover Funcionário”. 5. Clicar no botão “Cancelar Operação”. 6. Seguir o fluxo UC-E-007.FE-001.
	<p>UC-E-007.FA-002.FE-001</p> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Funcionários”. 3. Escolher a opção “Adicionar novo Funcionário”. 4. Inserir as informações requisitadas pelo sistema. 5. Clicar em “Voltar”. 6. Seguir o fluxo UC-E-007.FE-001.
	<p>UC-E-007.FA-003</p> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Funcionários”. 3. Selecionar um Funcionário da lista de Funcionários. 4. Clicar no botão “Informações”. 5. Clicar em “Voltar”. 6. Seguir o fluxo UC-E-007.FE-001.
<p>Regras de negócios:</p>	<ol style="list-style-type: none"> 1. Realizar Login 2. Selecionar opção de Gerenciar Funcionários. 3. Escolher entre as opções: <ol style="list-style-type: none"> 1. Adicionar Funcionário. 2. Remover Funcionário. 3. Visualizar informatização do Funcionário.

Fonte: Elaborado pelo autor

Tabela 17 - Descrição dos Casos de Uso - Gerenciar Eventos.

UC-E-008	Gerenciar Eventos	
Importância:	Essencial	
Atores:	Gestores	
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir cadastro ativo como Gestor. 2. Estar vinculado a algum Estabelecimento. 3. Estar logado. 	
Pós-condições:	<p>Possibilidade de gerenciar os Eventos cadastrados e ativos de um Estabelecimento. Dentre as funções:</p> <ol style="list-style-type: none"> 1. Adicionar novos Eventos. 2. Remover Eventos. 3. Visualizar informações dos eventos. 	
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Gerenciar Entidades”. 3. Clicar na opção “Gerenciar Eventos”. 	
Fluxo alternativo:	UC-E-008.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Eventos”. 3. Selecionar um Evento. 4. Clicar no botão “Remover Evento”. 5. Clicar em “Confirmar Operação”.
	UC-E-008.FA-002	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Eventos”. 3. Escolher a opção “Novo Evento”. 4. Inserir as informações requisitadas pelo sistema. 5. Clicar em “Salvar”.

	UC-E-008.FA-003	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Evento”. 3. Selecionar um Evento da lista de Funcionários. 4. Clicar no botão “Visualizar Informações”.
Fluxo de exceção:	UC-E-008.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Eventos”. 3. Clicar em “Voltar”.
	UC-E-008.FA-001.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clica na opção “Eventos”. 3. Selecionar um Evento da lista de Eventos. 4. Clicar no botão “Remover Evento”. 5. Clicar no botão “Cancelar Operação”. 6. Seguir o fluxo UC-E-008.FE-001.
	UC-E-008.FA-002.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Evento”. 3. Escolher a opção “Novo Evento”. 4. Inserir as informações requisitadas pelo sistema. 5. Clicar em “Voltar”. 6. Seguir o fluxo UC-E-008.FE-001.
	UC-E-008.FA-	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Evento”.
	001	

	003	<ol style="list-style-type: none"> 3. Selecionar um Evento da lista de Eventos. 4. Clicar no botão “Visualizar Informações”. 5. Clicar em “Voltar”. 6. Seguir o fluxo UC-E-008.FE-001.
Regras de negócios:	<ol style="list-style-type: none"> 1. Realizar Login 2. Selecionar opção de Gerenciar Funcionários. 	

Fonte: Elaborado pelo autor

Tabela 18 - Descrição dos Casos de Uso - Gerenciar Estabelecimentos.

UC-E-009	Gerenciar Estabelecimentos	
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir cadastro ativo como Gestor. 2. Estar vinculado a algum Estabelecimento. 3. Estar logado. 	
Pós-condições:	<ol style="list-style-type: none"> 1. Possibilidade de gerenciar o Estabelecimento cadastrado e ativo no Sistema. Dentre as funções: <ol style="list-style-type: none"> a. Editar informações do Estabelecimento. b. Visualizar informações do Estabelecimento. 	
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Entidade”. 3. Clicar na opção “Editar Informações”. 4. Alterar as informações que aparecerão na tela. 5. Clicar em “Salvar” 	
Fluxo alternativo:	Não possui.	
Fluxo de exceção:	UC-E-008.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Entidade”. 3. Clicar na opção “Editar Informações”.

	<table border="1"> <tr> <td></td> <td>4. Clicar em “Voltar”.</td> </tr> </table>		4. Clicar em “Voltar”.
	4. Clicar em “Voltar”.		
	<table border="1"> <tr> <td>UC-E-008.FA-002.FE-001</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Entidade”. 3. Clicar na opção “Editar Informações”. 4. Clicar em “Cancelar”. 5. Inserir as informações requisitadas pelo sistema. 6. Clicar em “Voltar”. 7. Seguir o fluxo UC-E-008.FE-001. </td> </tr> </table>	UC-E-008.FA-002.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Entidade”. 3. Clicar na opção “Editar Informações”. 4. Clicar em “Cancelar”. 5. Inserir as informações requisitadas pelo sistema. 6. Clicar em “Voltar”. 7. Seguir o fluxo UC-E-008.FE-001.
UC-E-008.FA-002.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Entidade”. 3. Clicar na opção “Editar Informações”. 4. Clicar em “Cancelar”. 5. Inserir as informações requisitadas pelo sistema. 6. Clicar em “Voltar”. 7. Seguir o fluxo UC-E-008.FE-001. 		
Regras de negócios:	<ol style="list-style-type: none"> 1. Realizar Login 2. Selecionar opção de Gerenciar Estabelecimento. 		

Fonte: Elaborado pelo autor

Tabela 19 - Descrição dos Casos de Uso - Gerenciar Filas de Espera.

UC-E-010	Gerenciar Filas de Espera
Importância:	Essencial
Atores:	Gestor
Pré-condições:	<ol style="list-style-type: none"> 1. Possuir cadastro ativo como Gestor. 2. Estar vinculado a algum Estabelecimento. 3. Estar logado.
Pós-condições:	<ol style="list-style-type: none"> 1. Possibilidade de gerenciar a Lista de Espera do Evento cadastrado do Estabelecimento. Dentre as funções: <ol style="list-style-type: none"> a. Editar informações da Lista de Espera. b. Visualizar informações da Lista de Espera. c. Remover Paciente da Lista de Espera.
Fluxo principal:	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002.

	<ol style="list-style-type: none"> 2. Clicar na opção “Lista de Espera”. 3. Clicar na opção “Editar Informações”. 4. Alterar os dados. 5. Clicar em “Confirmar” 				
Fluxo alternativo:	<table border="1"> <tr> <td>UC-E-010.FA-001</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera 3. Clicar na opção “Remover Paciente”. 4. Clicar em “Confirmar Operação”. </td> </tr> </table>	UC-E-010.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera 3. Clicar na opção “Remover Paciente”. 4. Clicar em “Confirmar Operação”. 		
UC-E-010.FA-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera 3. Clicar na opção “Remover Paciente”. 4. Clicar em “Confirmar Operação”. 				
Fluxo de exceção:	<table border="1"> <tr> <td>UC-E-010.FE-001</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera” 3. Clicar na opção “Editar Informações”. 4. Clicar em “Cancelar”. </td> </tr> <tr> <td>UC-E-010.FA-001.FE-001</td> <td> <ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera 3. Clicar na opção “Remover Paciente”. 4. Clicar em “Cancelar”. </td> </tr> </table>	UC-E-010.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera” 3. Clicar na opção “Editar Informações”. 4. Clicar em “Cancelar”. 	UC-E-010.FA-001.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera 3. Clicar na opção “Remover Paciente”. 4. Clicar em “Cancelar”.
UC-E-010.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera” 3. Clicar na opção “Editar Informações”. 4. Clicar em “Cancelar”. 				
UC-E-010.FA-001.FE-001	<ol style="list-style-type: none"> 1. Seguir o fluxo UC-E-002. 2. Clicar na opção “Lista de Espera 3. Clicar na opção “Remover Paciente”. 4. Clicar em “Cancelar”. 				
Regras de negócios:	<ol style="list-style-type: none"> 1. Realizar Login 2. Selecionar opção de Evento. 3. Selecionar a opção Editar Evento. 				

Fonte: Elaborado pelo autor

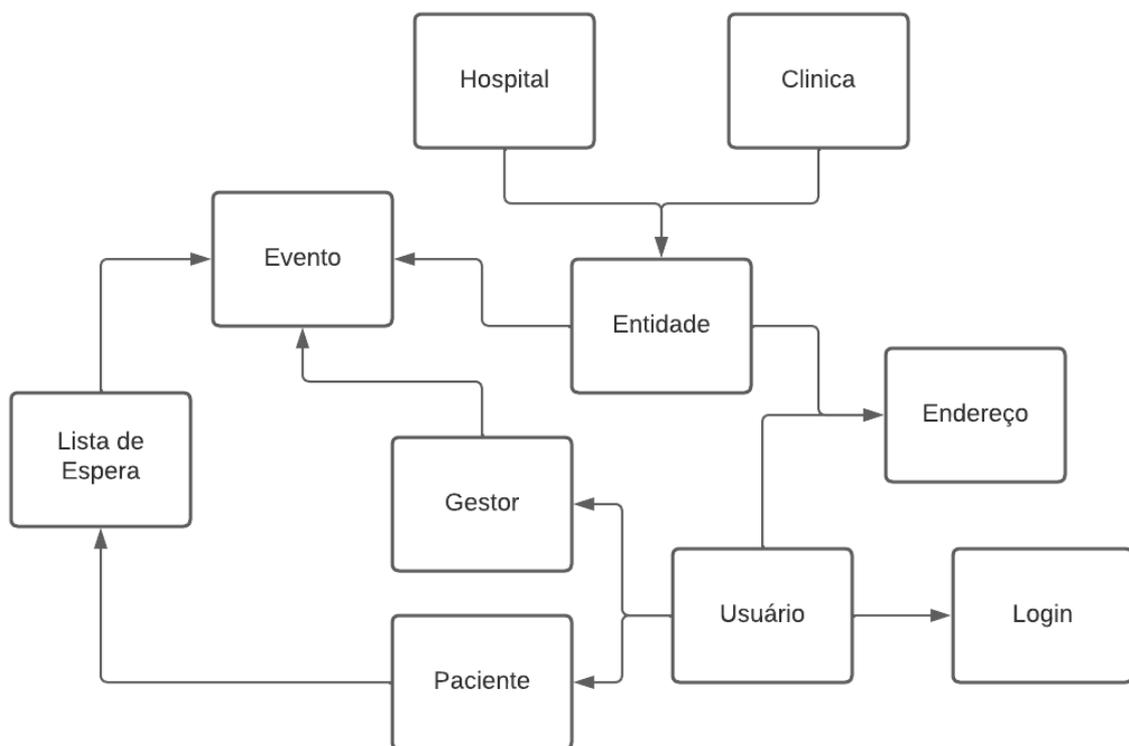
3.1.4. Análise de caso de uso

Esta seção descreve através de representação gráfica o Diagrama de Classes de Análise, que descreve a estrutura e as relações entre as classes em um sistema durante a fase de análise de um projeto de engenharia de software.

3.1.4.1. Diagrama de classes de análise

Esta seção descreve a estrutura e as relações entre as classes em um sistema durante a fase de análise de um projeto de engenharia de software. A descrição é realizada por meio de representação gráfica de classes definidas na seção 3.1.1.

Figura 9 - Diagrama de Classe



Fonte: Elaborado pelo autor

3.2. Arquitetura de Software

Esta seção tem como objetivo apresentar uma visão geral e detalhada da Arquitetura de Software definida para o projeto atual, além de descrever os microsserviços que foram estabelecidos para atender aos requisitos mencionados na seção 3.1. Ela é subdividida em:

1. **Microserviços do Sistema:** Subseção responsável por listar os microserviços definidos para atender aos requisitos do sistema e suas respectivas funções descritas.
2. **Diagrama Geral da Arquitetura do sistema:** Subseção responsável por representar visualmente a estrutura e os componentes principais do sistema de forma geral.
3. **Diagrama Detalhado da Arquitetura do sistema:** Subseção responsável por representar visualmente a estrutura e os componentes principais do sistema de forma detalhada e descritiva.

Uma arquitetura de software é a estrutura fundamental de um sistema de software, que define seus componentes principais, suas interações e como eles são organizados para atender aos requisitos do sistema. A arquitetura de software pode ser vista como um plano de alto nível que orienta o processo de desenvolvimento de software. Arquitetura baseada em microserviço envolve a divisão de um sistema em vários microserviços independentes, cada um responsável por uma função específica. Cada microserviço é executado em seu próprio processo e pode ser implantado e atualizado independentemente dos outros microserviços. Diante disso, a arquitetura escolhida no presente projeto é a arquitetura baseada em microserviço.

3.2.1. Microserviços do sistema

A seguir, uma lista dos microserviços selecionados para atender aos requisitos do sistema e suas respectivas funções. Cada micro serviço é responsável por executar uma única função específica, garantindo a modularidade e escalabilidade do sistema.

Tabela 20 - Lista de Microserviços

Nome	Descrição
Serviço de Autenticação	Este serviço é responsável por gerenciar a autenticação de usuários no sistema. Ele pode fornecer recursos como registro de usuário, login e logout da aplicação.

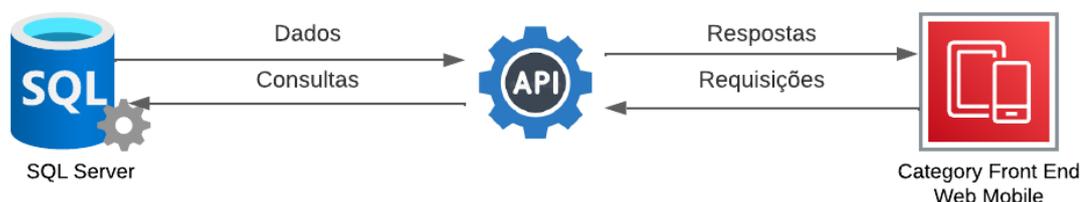
Serviço de Entidades	Este serviço é responsável pelo gerenciamento de Estabelecimentos do sistema. Ele fornece recursos como registro de Estabelecimentos, gerenciamento de dados da Estabelecimento e gerenciamento de status do sistema para cada Estabelecimento registrado.
Serviço de Usuários	Este serviço é responsável pelo gerenciamento de usuários do sistema. Ele fornece recursos como registro de usuário, gerenciamento de dados do usuário e gerenciamento de status do sistema para cada usuário registrado.
Serviço de Eventos	Este serviço é responsável pelo gerenciamento dos eventos de cada entidade do sistema. Ele fornece recursos como registro de eventos, gerenciamento de dados dos eventos e gerenciamento do status do sistema para cada evento registrado.
Serviço de Lista de Espera	Este serviço é responsável pelo gerenciamento das listas de espera dos eventos cadastrados. Ele fornece recursos como gerenciar as solicitações dos usuários e manter uma fila organizada de atendimento.

Fonte: Elaborado pelo autor

3.2.2. Diagrama Geral da Arquitetura do Sistema

A seguir, o diagrama geral da arquitetura do sistema proposto pelo trabalho.

Figura 10 - Arquitetura Geral



Fonte: Elaborado pelo autor

Nessa figura, é apresentado um esquema geral da arquitetura do sistema. Ele é dividido em duas partes principais: o Front-End e o Back-End. O Front-End é uma aplicação com interface gráfica que interage com a outra parte do sistema e realiza a troca de dados utilizando o protocolo HTTP, por meio de chamadas aos endpoints do sistema.

O Front-End é a interface do sistema que o usuário final utiliza para interagir com as funcionalidades do sistema. Ele é responsável por apresentar informações ao usuário e receber entradas do usuário para encaminhar solicitações ao Back-End. As solicitações são feitas em formato de requisições HTTP, que podem conter informações como parâmetros e dados do usuário.

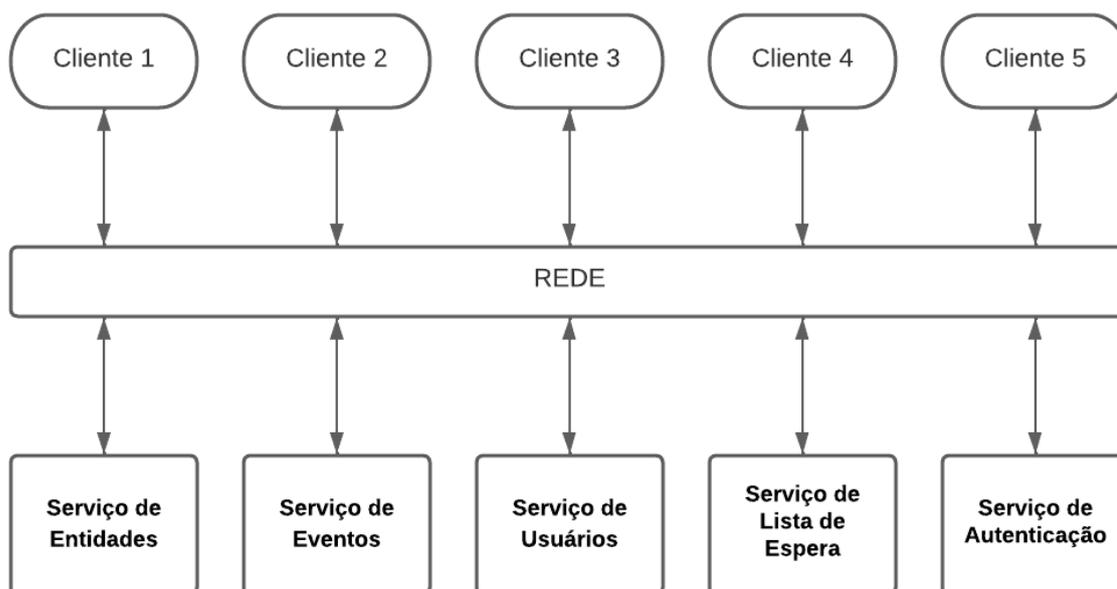
Já o Back-End é a parte do sistema responsável pelo processamento das requisições recebidas do Front-End. Ele é composto por diversos componentes, como servidores, bancos de dados, APIs e serviços de terceiros, que trabalham em conjunto para fornecer as funcionalidades do sistema. O Back-End pode receber dados de diversas fontes, como bancos de dados, serviços de terceiros, arquivos, entre outros, e processá-los para gerar as respostas às solicitações recebidas pelo Front-End.

Dessa forma, a arquitetura cliente-servidor com uma divisão entre Front-End e Back-End permite que as funcionalidades do sistema sejam distribuídas e escalonáveis, garantindo que as solicitações do usuário sejam atendidas de maneira rápida e eficiente.

3.2.3. Diagrama Detalhado da Arquitetura do sistema

A seguir, o diagrama detalhado da arquitetura do sistema, destacando os microsserviços que o compõem.

Figura 11 - Arquitetura Cliente-Servidor



Fonte: Elaborado pelo autor

O diagrama apresenta a arquitetura de microsserviços, onde os "Clientes" representam os objetos que fornecem a interface gráfica do sistema. Esses clientes podem ser acessados por meio de diversos tipos de dispositivos de hardware, como smartphones, tablets e computadores.

A comunicação entre os clientes e os microsserviços é realizada por meio dos endpoints expostos pelo sistema. Esses endpoints são responsáveis por disponibilizar as funcionalidades do sistema por meio de APIs RESTful, que permitem a troca de informações entre o cliente e o microsserviço.

Os endpoints são projetados para fornecer acesso seguro e controlado às funcionalidades do sistema, garantindo que apenas usuários autorizados possam acessar as informações e funcionalidades necessárias. Além disso, a arquitetura de microsserviços permite que os endpoints sejam facilmente escalados ou atualizados sem afetar outros componentes do sistema.

Com essa abordagem, o sistema é altamente flexível e adaptável, permitindo que diferentes tipos de clientes possam interagir com os microsserviços por meio de uma interface comum. Isso torna o sistema mais acessível e amigável para os usuários, aumentando a satisfação do cliente e a eficácia das operações do sistema.

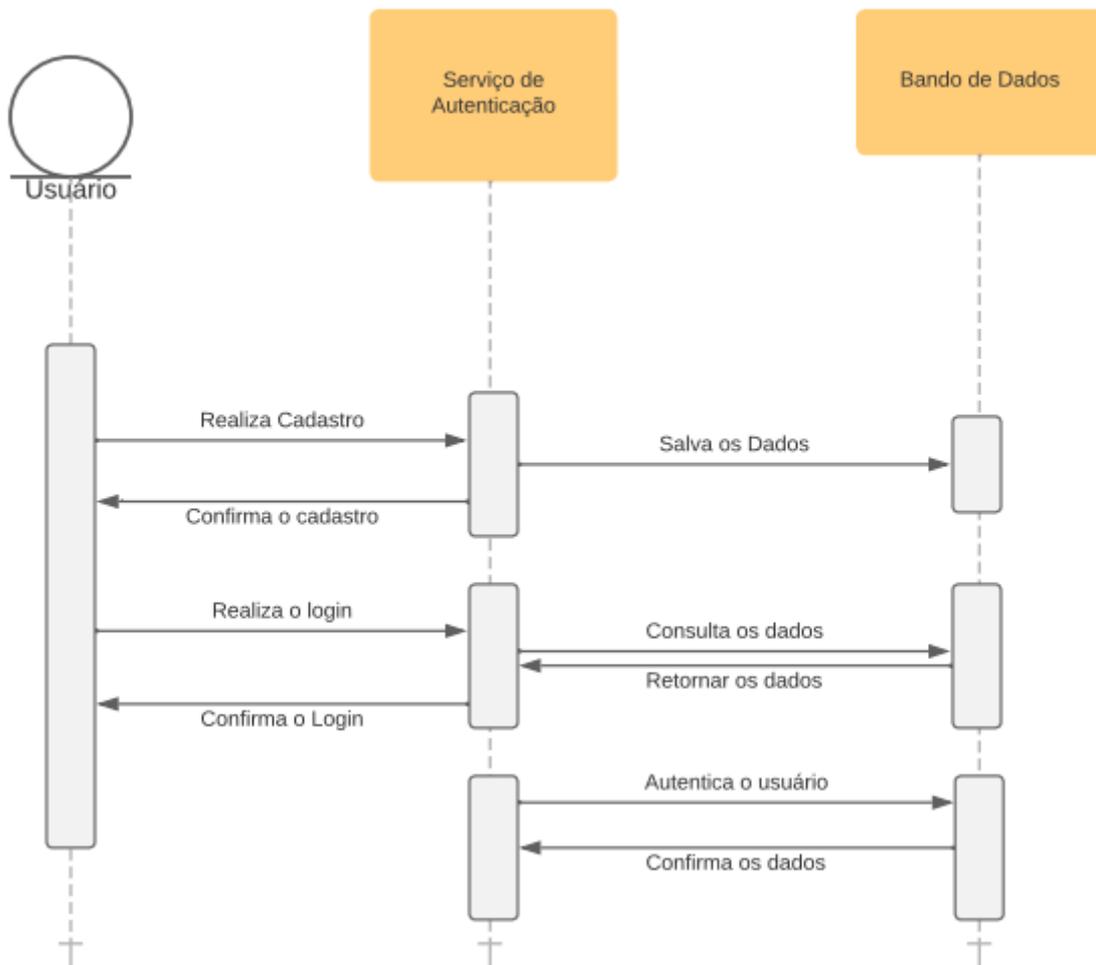
3.2.4. Diagrama de Sequência

Esta seção tem como objetivo ilustrar e descrever os diagramas de sequência que representam os micros serviços do sistema, previamente documentados nas seções anteriores.

a) Serviço de Autenticação

A seguir, o diagrama de sequência do micros serviço de autenticação.

Figura 12 - Diagrama de Sequência de Autenticação



Fonte: Elaborado pelo autor.

O Diagrama apresentado na Figura 12 ilustra de forma clara o funcionamento do micros serviço de autenticação, evidenciando a interação entre o cliente e o serviço

por meio de trocas de mensagens HTTP. Nesta figura, é possível notar dois tipos de funcionalidades: Cadastro e Login.

i) Cadastro

No serviço de Cadastro, é necessário que o cliente envie ao servidor os dados a serem armazenados na base de dados do sistema. O processo é concluído quando o servidor envia uma resposta de confirmação ao cliente, confirmando o cadastro realizado. Nesse processo, o cliente coleta os dados necessários para o cadastro e os envia ao servidor por meio de uma requisição apropriada. O servidor, por sua vez, processa esses dados e os armazena na base de dados do sistema. Após a conclusão bem-sucedida do cadastro, o servidor envia uma resposta ao cliente, confirmando que o cadastro foi realizado com sucesso.

ii) Login

No serviço de Login, o cliente precisa enviar ao servidor os dados solicitados pelo sistema. Nesse processo, o cliente coleta as informações necessárias e as envia para o servidor. Em seguida, o servidor processa esses dados e valida se eles estão corretamente armazenados no banco de dados. Uma vez validados, o servidor retorna a confirmação do login ao cliente.

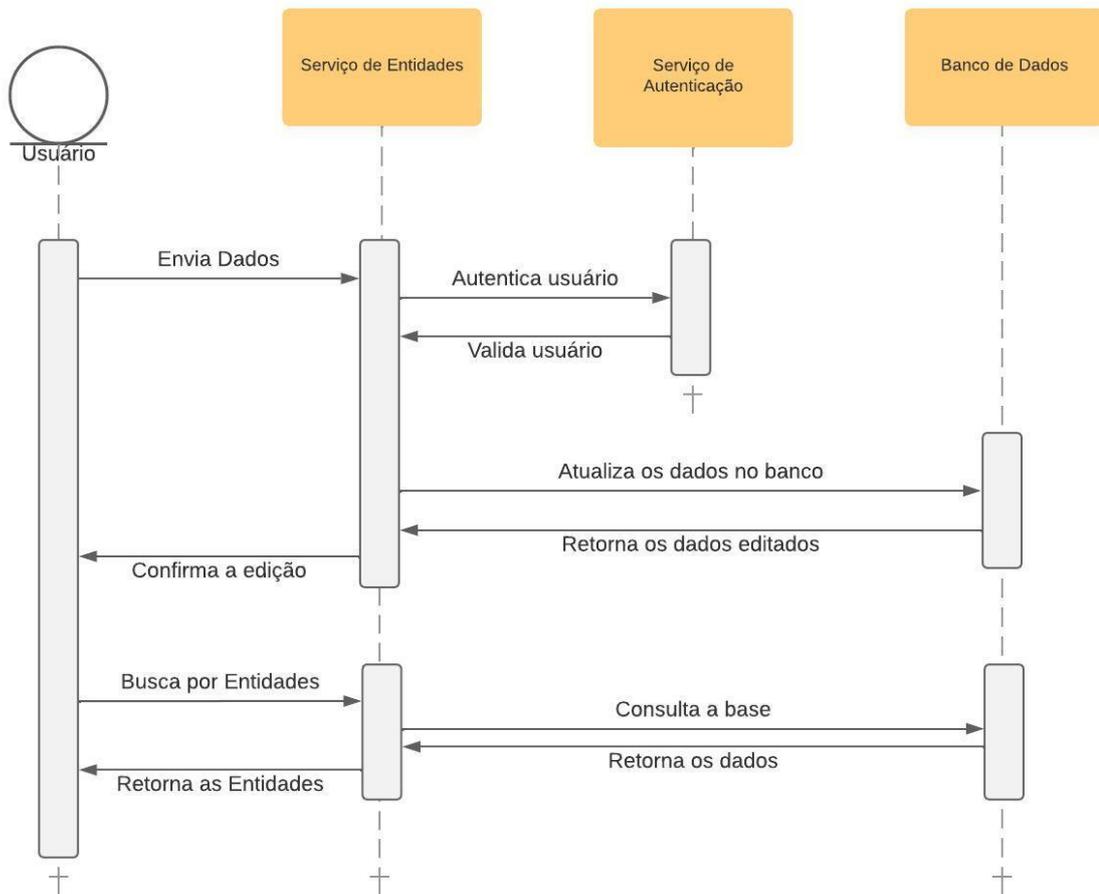
Durante a interação, o cliente fornece as credenciais de acesso, como nome de usuário e senha, ao servidor por meio de uma requisição adequada. O servidor, por sua vez, verifica se as informações fornecidas correspondem aos registros existentes na base de dados. Caso a validação seja bem-sucedida, o servidor envia uma resposta ao cliente, confirmando que o login foi realizado com sucesso.

Além disso, existe outra funcionalidade crucial executada exclusivamente pelo servidor, chamada de Autenticação. Em outros microsserviços do presente projeto, é essencial verificar se o usuário possui permissões adequadas para acessar e utilizar determinados serviços, os quais serão detalhados posteriormente. Portanto, essa funcionalidade é responsável por validar os dados, junto ao banco de dados, do usuário autenticado e suas permissões correspondentes.

b) Serviço de Entidades

A seguir, o diagrama de sequência do microserviço de entidades.

Figura 13 - Diagrama de Sequência de Entidades



Fonte: Elaborado pelo autor

O Diagrama apresentado na Figura 13 ilustra de forma clara o funcionamento do microserviço de entidades, evidenciando a interação entre o cliente e o serviço por meio de trocas de mensagens HTTP. Nesta figura, é possível notar dois tipos de funcionalidades: Edição de Estabelecimento e Busca por Estabelecimentos.

i) Edição de Estabelecimento

No serviço de Edição de Estabelecimento, o cliente tem a responsabilidade de enviar os dados do Estabelecimento que precisam ser modificados no banco de dados. Em seguida, o microsserviço entra em ação, validando as permissões do usuário por meio do serviço de autenticação. Uma vez que as permissões são verificadas e autorizadas, o microsserviço realiza as alterações nos dados do Estabelecimento no banco de dados. Em seguida, os dados atualizados são retornados ao cliente, e o microsserviço confirma a conclusão da edição.

Durante o processo, o cliente envia as informações atualizadas do Estabelecimento ao microsserviço. Este, em colaboração com o serviço de Autenticação, assegura que o usuário possua as permissões necessárias para efetuar a modificação. Uma vez validadas as permissões, o microsserviço executa as alterações no banco de dados, refletindo as mudanças solicitadas pelo cliente. Os novos dados são então recuperados e retornados ao microsserviço. Por fim, o microsserviço de Edição de Estabelecimento envia uma resposta ao cliente, confirmando que a operação de edição foi concluída com êxito.

É crucial mencionar que todo o processo é desenvolvido com ênfase na segurança. A validação das permissões do usuário e a autenticação são implementadas para garantir a integridade dos dados e a consistência das operações de edição, protegendo o sistema contra acesso não autorizado ou modificações indevidas.

ii) Busca por Estabelecimentos

No serviço de Busca por Estabelecimentos, o cliente faz uma solicitação ao servidor para obter uma lista dos Estabelecimentos disponíveis. Após receber a solicitação, o servidor processa a requisição consultando os dados no banco de dados e, em seguida, retorna ao cliente uma lista contendo as entidades disponíveis no sistema.

Durante esse processo, o cliente envia a solicitação ao servidor, indicando a intenção de buscar os Estabelecimentos. O servidor, por sua vez, recebe essa solicitação e realiza a consulta no banco de dados, buscando os registros correspondentes aos Estabelecimentos desejados. Após realizar a consulta, o

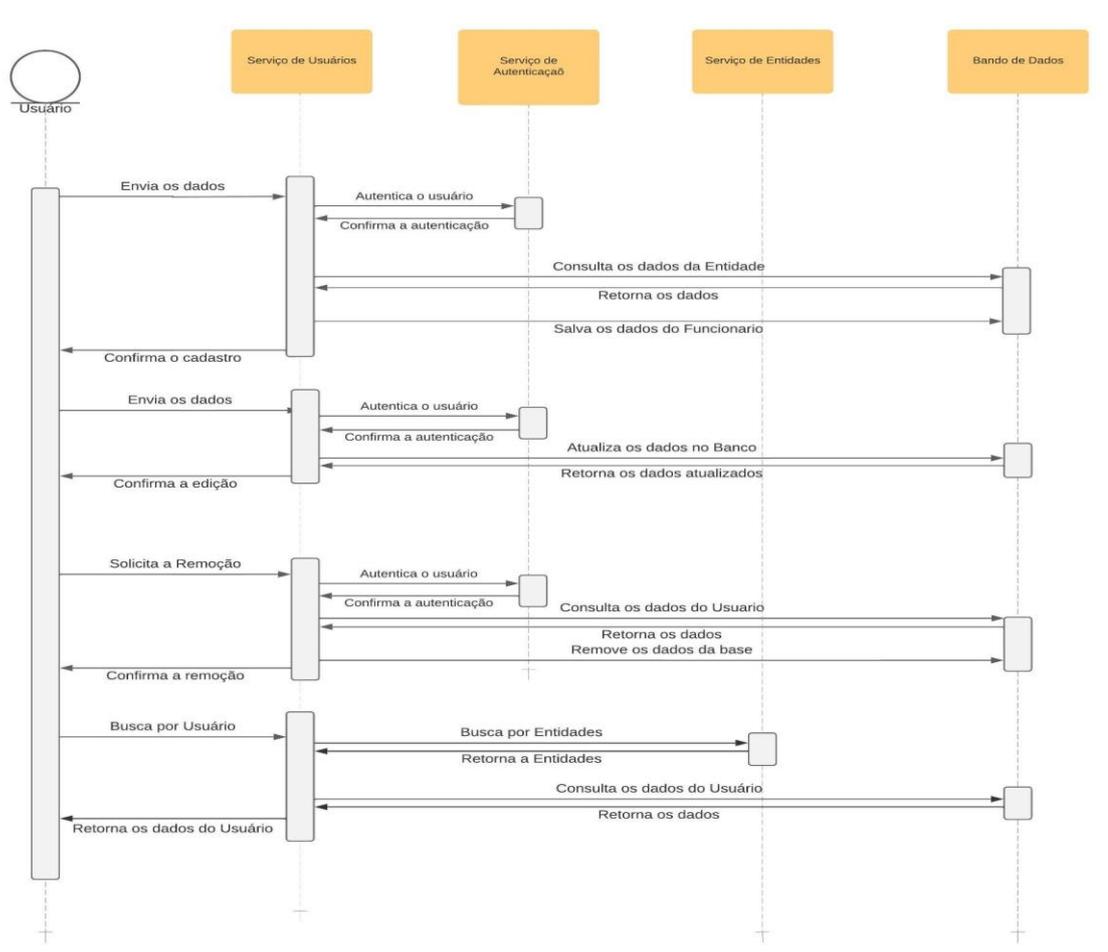
servidor compila os resultados em uma lista e a retorna ao cliente. Essa lista contém os Estabelecimentos disponíveis no sistema, conforme a solicitação feita pelo cliente.

É importante ressaltar que o serviço de Busca por Estabelecimento pode permitir parâmetros de busca adicionais, como filtros ou critérios específicos, para refinar os resultados e atender às necessidades do cliente. Em suma, o serviço de Busca por Estabelecimento proporciona ao cliente a capacidade de obter uma lista dos Estabelecimento disponíveis no sistema, através de uma comunicação com o servidor que realiza a consulta no banco de dados e retorna os resultados correspondentes.

c) Serviço de Usuários

A seguir, o diagrama de sequência do microserviço de usuários.

Figura 14 - Diagrama de Sequência de Usuários



Fonte: Elaborado pelo autor

O Diagrama apresentado na Figura 14 ilustra de forma clara o funcionamento do microsserviço de usuários, evidenciando a interação entre o cliente e o serviço por meio de trocas de mensagens HTTP. Nesta figura, é possível notar os tipos de funcionalidades: Cadastro de Usuários, Edição de Usuários, Remoção de Usuários e Busca por Funcionário.

i) Cadastro de Usuários

No contexto do serviço de Cadastro de Usuários, é importante mencionar que ele é projetado para lidar especificamente com o cadastro de usuários do tipo Funcionário.

Quando um cliente deseja cadastrar um novo funcionário no sistema, ele envia os dados relevantes para o servidor, que possui a responsabilidade de armazená-los na base de dados.

Nesse processo, o microsserviço de Usuários é acionado para tratar esses dados recebidos. Esse microsserviço, em primeiro lugar, utiliza outro microsserviço chamado Autenticação, que tem como objetivo principal verificar a identidade do usuário que está enviando as informações. Por meio do microsserviço de Autenticação, o sistema realiza uma série de verificações para garantir que o usuário seja autenticado de forma segura e confiável.

Uma vez que a autenticação é bem-sucedida, o serviço de Cadastro de Usuários realiza consultas na base de dados para validar as informações específicas do Estabelecimento. Essas consultas podem incluir a verificação de duplicatas de registros ou a consistência dos dados fornecidos. A validação adequada é fundamental para garantir a integridade dos dados armazenados no sistema.

Com os dados validados, o serviço de Cadastro de Usuários inicia o processo de processamento dessas informações. Isso pode incluir etapas como a formatação correta dos dados, a execução de cálculos adicionais ou a execução de lógica de negócios específica para a criação de um novo registro de funcionário.

Uma vez concluído o processamento, o serviço de Cadastro de Usuários realiza a inserção dos dados do funcionário na base de dados do sistema, garantindo que todas as informações relevantes sejam corretamente persistidas. Após a conclusão

desse processo, uma confirmação de cadastro é enviada de volta ao cliente, informando-o sobre o sucesso da operação.

ii) Edição de Usuários

No contexto do serviço de Edição de Usuários, é importante mencionar que ele é projetado para lidar com qualquer tipo de usuário.

Quando um cliente deseja editar um usuário no sistema, ele envia os dados relevantes para o servidor, que possui a responsabilidade de armazená-los na base de dados. Nesse processo, o microserviço de Usuários é acionado para tratar esses dados recebidos. Esse microserviço, em primeiro lugar, utiliza outro microserviço chamado Autenticação, que tem como objetivo principal verificar a identidade do usuário que está enviando as informações.

Por meio do microserviço de Autenticação, o sistema realiza uma série de verificações para garantir que o usuário seja autenticado de forma segura e confiável. Com os dados validados, o serviço de Edição de Usuários inicia o processo de processamento dessas informações. Isso pode incluir etapas como a formatação correta dos dados, a execução de cálculos adicionais ou a execução de lógica de negócios específica para a criação de um novo registro de funcionário.

Uma vez concluído o processamento, o serviço de Cadastro de Usuários realiza a inserção dos dados do funcionário na base de dados do sistema, garantindo que todas as informações relevantes sejam corretamente persistidas. Após a conclusão desse processo, uma confirmação de edição é enviada de volta ao cliente, informando-o sobre o sucesso da operação.

iii) Remoção de Usuários

No contexto do serviço de Remoção de Usuários, o foco é na exclusão de usuários do tipo Funcionário do sistema.

Quando um cliente deseja remover um funcionário, ele envia uma solicitação de remoção para o servidor, que é responsável por lidar com essa solicitação. Nesse processo, o microserviço de Usuários é acionado para processar a solicitação recebida. Para garantir a segurança e a autenticidade do usuário que está solicitando

a remoção, o microserviço de Autenticação é utilizado. Através desse microserviço, o sistema realiza verificações para autenticar o usuário de maneira confiável.

Após a autenticação bem-sucedida, o serviço de Remoção de Usuários executa consultas na base de dados para validar as informações específicas do usuário a ser removido. Essas consultas podem incluir verificação de duplicatas de registros ou a consistência dos dados fornecidos. A validação adequada é essencial para preservar a integridade dos dados armazenados no sistema.

Uma vez concluída a validação, o serviço de Remoção de Usuários realiza a exclusão dos dados do funcionário na base de dados do sistema, assegurando que todas as informações relevantes sejam removidas corretamente. Após a conclusão desse processo, uma confirmação de remoção é enviada de volta ao cliente, informando-o sobre o sucesso da operação.

iv) Busca por Usuários

O serviço de Busca de Usuário, integrante do microserviço de Usuários, permite ao cliente solicitar a busca dos usuários de um determinado Estabelecimento no sistema.

O processo tem início quando o cliente envia uma solicitação para recuperar os usuários associados a um Estabelecimento específico. Ao receber a solicitação, o microserviço realiza uma validação do Estabelecimento fornecido. Essa etapa é essencial para assegurar que o Estabelecimento seja válido e esteja em conformidade com as regras estabelecidas. A validação garante que a busca seja executada corretamente, evitando erros durante o processo.

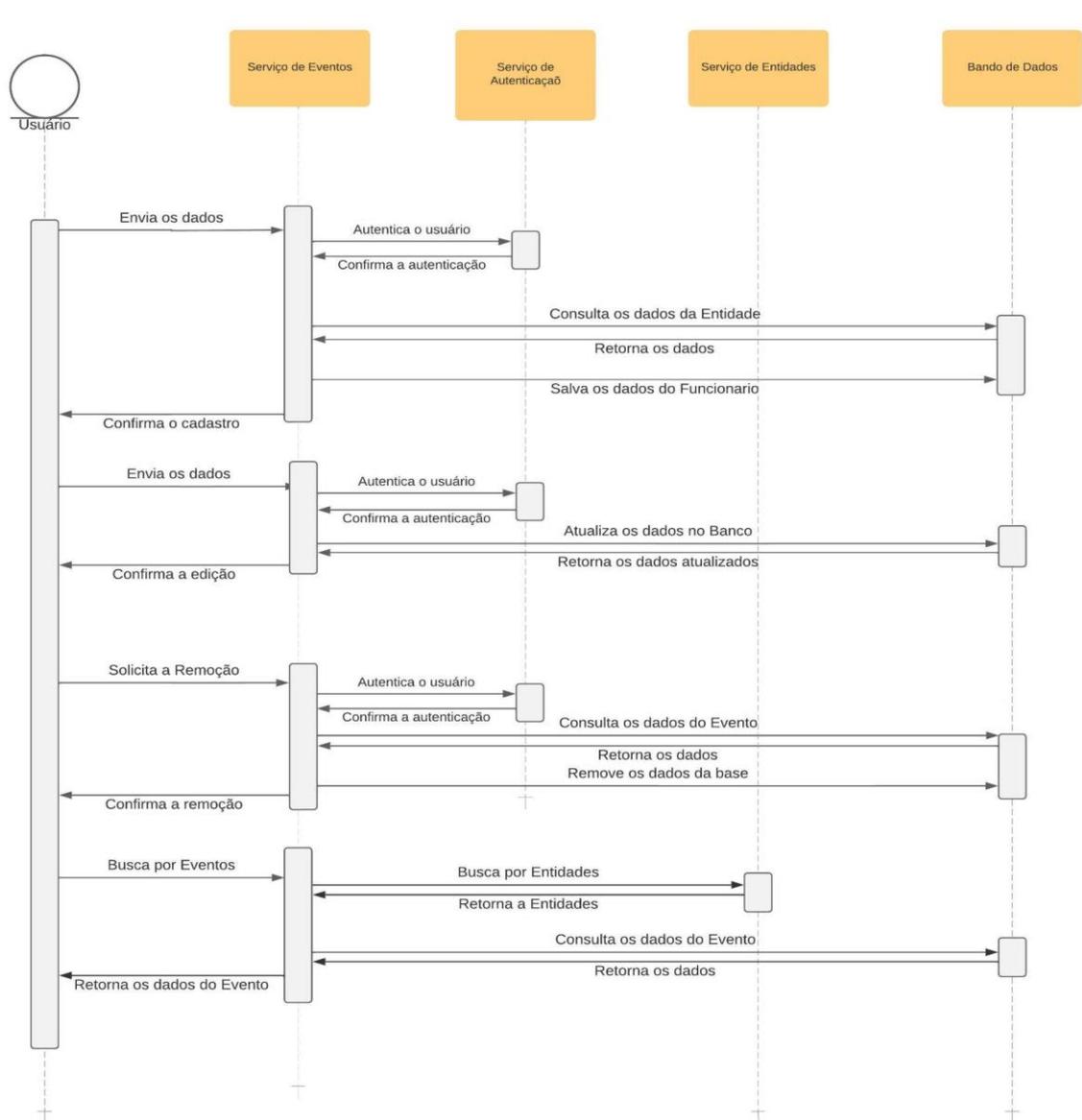
Após a validação do Estabelecimento, o microserviço inicia uma consulta à base de dados do sistema. Essa consulta é projetada para buscar os usuários relacionados ao Estabelecimento mencionado na solicitação. Além disso, é possível aplicar filtros à consulta, como critérios de pesquisa adicionais, para refinar os resultados e obter os usuários desejados. Assim que a consulta é concluída, o microserviço recebe os dados dos usuários provenientes da base de dados. Essas informações podem incluir detalhes como nome, identificação, cargo e informações de contato. O microserviço pode executar operações adicionais nos dados, como formatação ou aplicação de lógica de negócios específica, conforme necessário.

Por fim, o microserviço retorna os dados dos usuários para o cliente que fez a solicitação. Isso permite que o cliente obtenha as informações necessárias sobre os usuários do Estabelecimento buscado. Esses dados podem ser utilizados para exibir os usuários em uma interface de usuário, realizar operações adicionais ou atender a qualquer outra finalidade definida pelo cliente.

d) Serviço de Eventos

A seguir, o diagrama de sequência do microserviço de eventos.

Figura 15 - Diagrama Sequência de Eventos



Fonte: Elaborado pelo autor

O Diagrama apresentado na Figura 15 ilustra de forma clara o funcionamento do microsserviço de eventos, evidenciando a interação entre o cliente e o serviço por meio de trocas de mensagens HTTP. Nesta figura, é possível notar os tipos de funcionalidades: Cadastro de Eventos, Edição de Eventos, Remoção de Eventos e Busca por eventos.

i) Cadastro de Evento

No contexto do serviço de Cadastro de Eventos, é importante mencionar que ele é projetado para usuários do tipo Funcionario. Quando um cliente deseja cadastrar um novo evento no sistema, ele envia os dados relevantes para o servidor, que possui a responsabilidade de armazená-los na base de dados.

Nesse processo, o microsserviço de Usuários é acionado para tratar esses dados recebidos. Esse microsserviço, em primeiro lugar, utiliza outro microsserviço chamado Autenticação, que tem como objetivo principal verificar a identidade do usuário que está enviando as informações. Por meio do microsserviço de Autenticação, o sistema realiza uma série de verificações para garantir que o usuário seja autenticado de forma segura e confiável.

Uma vez que a autenticação é bem-sucedida, o serviço de Cadastro de Eventos realiza consultas na base de dados para validar as informações específicas do Estabelecimento. Essas consultas podem incluir a verificação de duplicatas de registros ou a consistência dos dados fornecidos. A validação adequada é fundamental para garantir a integridade dos dados armazenados no sistema.

Com os dados validados, o serviço de Cadastro de Eventos inicia o processo de processamento dessas informações. Isso pode incluir etapas como a formatação correta dos dados, a execução de cálculos adicionais ou a execução de lógica de negócios específica para a criação de um novo registro de funcionário.

Uma vez concluído o processamento, o serviço de Cadastro de Eventos realiza a inserção dos dados do evento na base de dados do sistema, garantindo que todas as informações relevantes sejam corretamente persistidas. Após a conclusão desse processo, uma confirmação de cadastro é enviada de volta ao cliente, informando-o sobre o sucesso da operação.

ii) Edição de Evento

No contexto do serviço de Edição de Evento, é importante mencionar que ele é projetado para lidar especificamente com usuários do tipo Funcionário.

Quando um cliente deseja editar um usuário no sistema, ele envia os dados relevantes para o servidor, que possui a responsabilidade de armazená-los na base de dados. Nesse processo, o microsserviço de Eventos é acionado para tratar esses dados recebidos. Esse microsserviço, em primeiro lugar, utiliza outro microsserviço chamado Autenticação, que tem como objetivo principal verificar a identidade do usuário que está enviando as informações.

Por meio do microsserviço de Autenticação, o sistema realiza uma série de verificações para garantir que o usuário seja autenticado de forma segura e confiável. Com os dados validados, o serviço de Edição de Evento inicia o processo de processamento dessas informações. Isso pode incluir etapas como a formatação correta dos dados, a execução de cálculos adicionais ou a execução de lógica de negócios específica para a criação de um novo registro de evento.

Uma vez concluído o processamento, o serviço de Edição de Evento realiza a inserção dos dados do evento na base de dados do sistema, garantindo que todas as informações relevantes sejam corretamente persistidas. Após a conclusão desse processo, uma confirmação de edição é enviada de volta ao cliente, informando-o sobre o sucesso da operação.

iii) Remoção de Evento

No contexto do serviço de Remoção de Evento, a funcionalidade é restrita exclusivamente a usuários do tipo Funcionário, garantindo um controle adequado de acesso e segurança.

Quando um cliente deseja remover um evento, ele envia uma solicitação de remoção para o servidor, que é responsável por lidar com essa solicitação. Nesse processo, o microsserviço de Evento é acionado para processar a solicitação recebida. Para garantir a segurança e a autenticidade do usuário que está solicitando a remoção, o microsserviço de Autenticação é utilizado. Através desse microsserviço, o sistema realiza verificações para autenticar o usuário de maneira confiável.

Após a autenticação bem-sucedida, o serviço de Remoção de Evento executa consultas na base de dados para validar as informações específicas do evento a ser removido. Essas consultas podem incluir verificação de duplicatas de registros ou a consistência dos dados fornecidos. A validação adequada é essencial para preservar a integridade dos dados armazenados no sistema.

Uma vez concluída a validação, o serviço de Remoção de Evento realiza a exclusão dos dados do evento na base de dados do sistema, assegurando que todas as informações relevantes sejam removidas corretamente. Após a conclusão desse processo, uma confirmação de remoção é enviada de volta ao cliente, informando-o sobre o sucesso da operação.

iv) Busca por Evento

No contexto do serviço de Busca de Evento, a funcionalidade é restrita exclusivamente a usuários do tipo Funcionário, garantindo um controle adequado de acesso e segurança.

O processo tem início quando o cliente envia uma solicitação para recuperar os eventos associados a um Estabelecimento específico. Ao receber a solicitação, o microsserviço realiza uma validação do Estabelecimento fornecido. Essa etapa é essencial para assegurar que o Estabelecimento seja válido e esteja em conformidade com as regras estabelecidas. A validação garante que a busca seja executada corretamente, evitando erros durante o processo.

Após a validação do Estabelecimento, o microsserviço inicia uma consulta à base de dados do sistema. Essa consulta é projetada para buscar os eventos relacionados ao Estabelecimento mencionado na solicitação. Além disso, é possível aplicar filtros à consulta, como critérios de pesquisa adicionais, para refinar os resultados e obter os eventos desejados. Assim que a consulta é concluída, o microsserviço recebe os dados dos usuários provenientes da base de dados. O microsserviço pode executar operações adicionais nos dados, como formatação ou aplicação de lógica de negócios específica, conforme necessário.

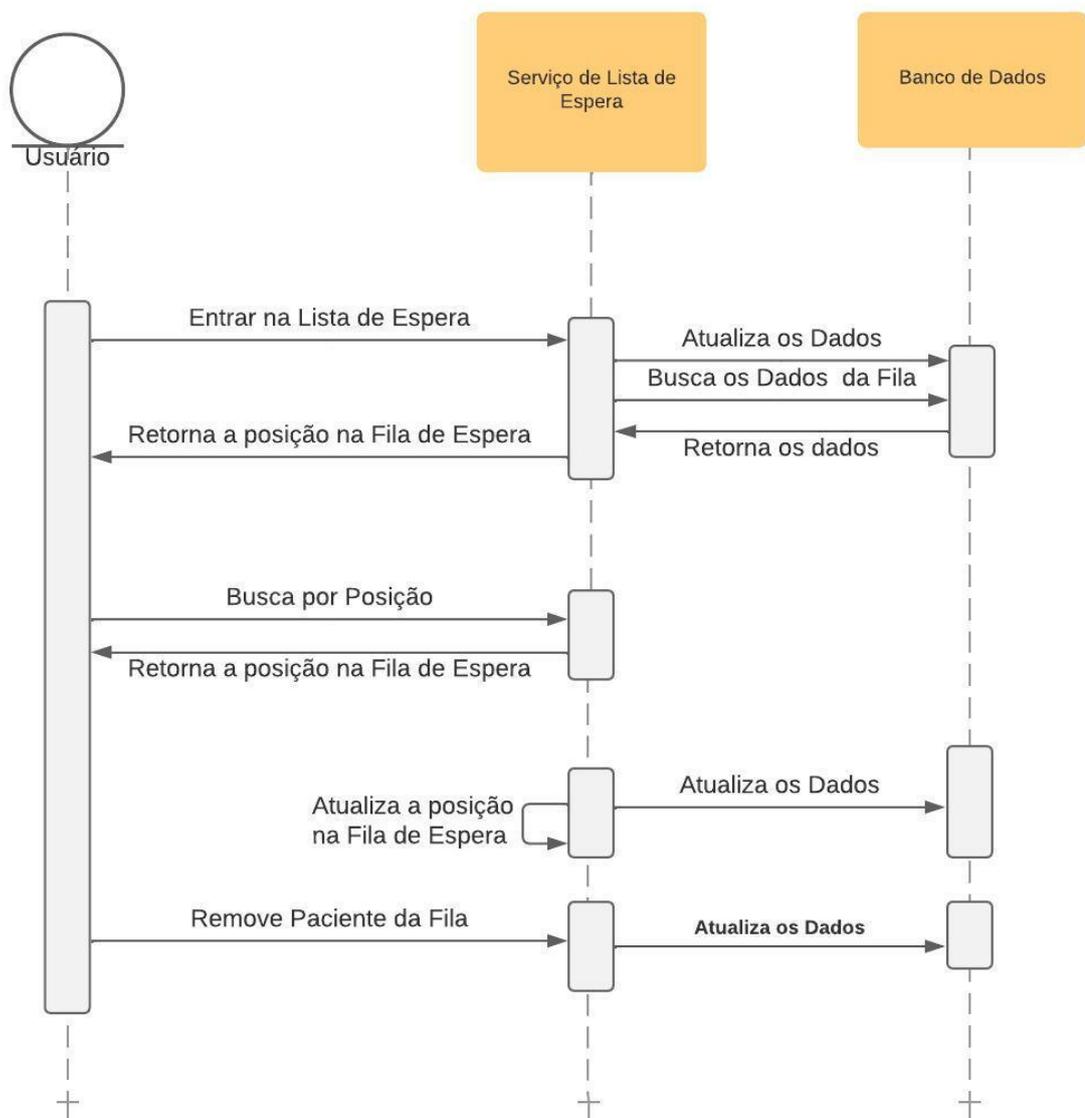
Por fim, o microsserviço retorna os dados dos eventos para o cliente que fez a solicitação. Isso permite que o cliente obtenha as informações necessárias sobre os usuários do Estabelecimento procurado. Esses dados podem ser utilizados para exibir

os eventos em uma interface de eventos, realizar operações adicionais ou atender a qualquer outra finalidade definida pelo cliente.

e) Serviço de Lista de Espera

A seguir, o diagrama de sequência do microserviço de Lista de Espera.

Figura 16 - Diagrama Sequência de Lista de Espera



Fonte: Elaborado pelo autor.

O Diagrama apresentado na Figura 16 ilustra de forma clara o funcionamento do microsserviço de Lista de Eventos, evidenciando a interação entre o cliente e o serviço por meio de trocas de mensagens HTTP. Nesta figura, é possível notar os tipos de funcionalidades: Entrar na Lista de Espera, Busca por Posição, Remoção de Paciente da lista de Espera, Atualização da Lista de Espera.

i) Entrar na Lista de Espera

No contexto do serviço de Entrar na Lista de Espera, a funcionalidade é restrita exclusivamente a usuários do tipo Paciente, garantindo um controle adequado de acesso e segurança.

O processo tem início quando o cliente envia uma solicitação ao servidor para registrar um usuário na lista de espera de um evento. Essa solicitação é processada pelo microsserviço de Lista de Espera, que atualiza os dados na base de dados do sistema. Em seguida, uma consulta aos dados da lista de espera é realizada na base e o serviço processa as informações, retornando ao cliente a posição atual do usuário na fila.

Nesse fluxo, o microsserviço de Lista de Espera desempenha um papel fundamental. Ele recebe as solicitações dos clientes e as processa de forma apropriada, garantindo a atualização precisa dos dados na base de dados do sistema. Com isso, os usuários podem obter informações em tempo real sobre sua posição na fila de espera.

A integração eficiente entre o cliente, o servidor e o microsserviço de Lista de Espera são essenciais para fornecer uma experiência fluida e transparente aos usuários. A arquitetura do sistema permite que as informações sejam processadas de forma rápida e precisa, garantindo uma resposta ágil ao cliente.

iii) Busca por Posição

No âmbito do serviço Entrar na Lista de Espera, a funcionalidade é restrita exclusivamente aos usuários designados como Pacientes, garantindo um controle adequado de acesso e segurança.

Nesse contexto, o serviço é projetado para operar de forma assíncrona, ou seja, o usuário não tem controle direto sobre o processo. Após enviar uma solicitação para entrar na lista de espera, o cliente aguarda enquanto o microsserviço de Lista de Espera processa as solicitações recebidas e consulta a base de dados utilizada, obtendo as informações pertinentes à lista de espera.

O microsserviço, por sua vez, é responsável por retornar ao cliente as informações necessárias, como a posição atual na fila. Essa abordagem assíncrona permite que os usuários acompanhem o progresso da lista de espera sem a necessidade de intervenções constantes, proporcionando uma experiência mais eficiente e conveniente.

Destaca-se que essa arquitetura assíncrona promove a segurança e o controle adequado do serviço, garantindo que apenas os usuários autorizados, no caso, os pacientes, possam acessar e interagir com a lista de espera.

iii) Atualização da Lista de Espera.

No âmbito do serviço de Atualização da Lista de Espera, a funcionalidade é restrita a qualquer tipo de usuário, garantindo assim nenhuma interferência externa aja sobre a funcionalidade. Portanto, trata-se de um serviço dentro do contexto do servidor.

O microsserviço de Lista de Espera, sua responsabilidade é consultar periodicamente as informações da base de dados em relação à fila de espera. Ao receber essas informações, o microsserviço deve processá-las e atualizar as informações pertinentes. Durante o processamento da lista, é necessário realizar várias validações, tais como:

1. Verificar se o paciente já está em atendimento: Antes de atualizar as informações da fila, o microsserviço deve verificar se o paciente já foi chamado para atendimento. Isso evita a duplicação de entradas na lista e garante a ordem correta de atendimento.
2. Verificar se o tempo de espera do paciente expirou: É importante considerar o tempo de espera máximo definido para cada paciente na fila. Se o tempo de

espera excedeu esse limite, o microserviço deve atualizar as informações adequadamente, removendo o paciente da lista ou tomando ação apropriada conforme as regras estabelecidas.

3. Verificar se o usuário foi removido da base: É fundamental garantir a integridade dos dados. Assim, é necessário validar se o usuário ainda está registrado na base de dados antes de processar as informações. Caso o usuário tenha sido removido ou não esteja mais presente na base, é importante tomar ações adequadas para manter a consistência da lista.

Essas validações garantem que a lista de espera seja corretamente atualizada, considerando os diversos cenários e condições definidas. Através desse processamento, o microserviço assegura a confiabilidade e a consistência das informações apresentadas ao cliente, oferecendo um serviço mais eficiente.

iv) Remoção de Paciente da lista de Espera

No contexto do serviço Entrar na Lista de Espera, a funcionalidade é restrita exclusivamente aos usuários do tipo Funcionário, garantindo um controle adequado de acesso e segurança.

O processo tem início quando o cliente, identificado como funcionário, envia uma solicitação de remoção de um usuário ao servidor. Nesse momento, o servidor assume a responsabilidade de verificar a existência do usuário na lista de espera e efetuar a remoção dos dados correspondentes na base de dados.

Ao receber a solicitação de remoção, o servidor realiza a validação da existência do usuário na lista de espera. Isso é feito consultando a base de dados para verificar se o usuário em questão está registrado como um paciente na lista de espera do evento específico.

Após a confirmação da existência do usuário na lista de espera, o servidor procede com a remoção dos dados relacionados ao usuário na base de dados. Isso garante que o usuário seja devidamente retirado da lista, liberando seu lugar para outros potenciais participantes.

Essa abordagem assegura o controle adequado de acesso ao serviço de Entrar na Lista de Espera, permitindo que apenas os funcionários autorizados possam

solicitar a remoção de usuários da lista. Além disso, o processamento realizado pelo servidor garante a integridade dos dados e mantém a lista de espera atualizada.

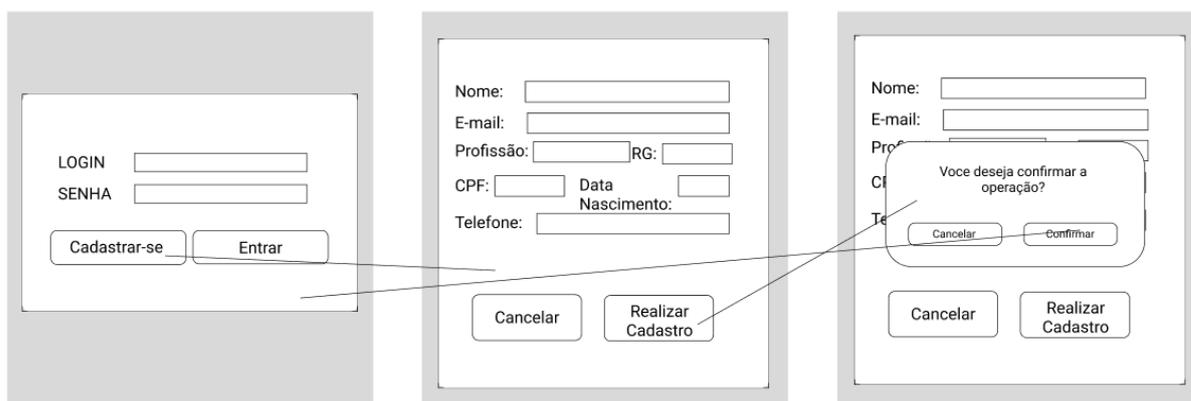
3.3. Prototipação

Esta seção tem como objetivo apresentar e descrever a prototipagem do módulo que faz parte do design do presente projeto. Serão fornecidos exemplos e descrições detalhadas dos protótipos desenvolvidos, demonstrando as soluções e funcionalidades implementadas. Esta seção é dividida na visão do Paciente e Gestor.

3.3.1. Paciente

Esta seção tem como objetivo ilustrar e descrever o processo de prototipagem das telas disponíveis para os usuários do tipo paciente no presente projeto.

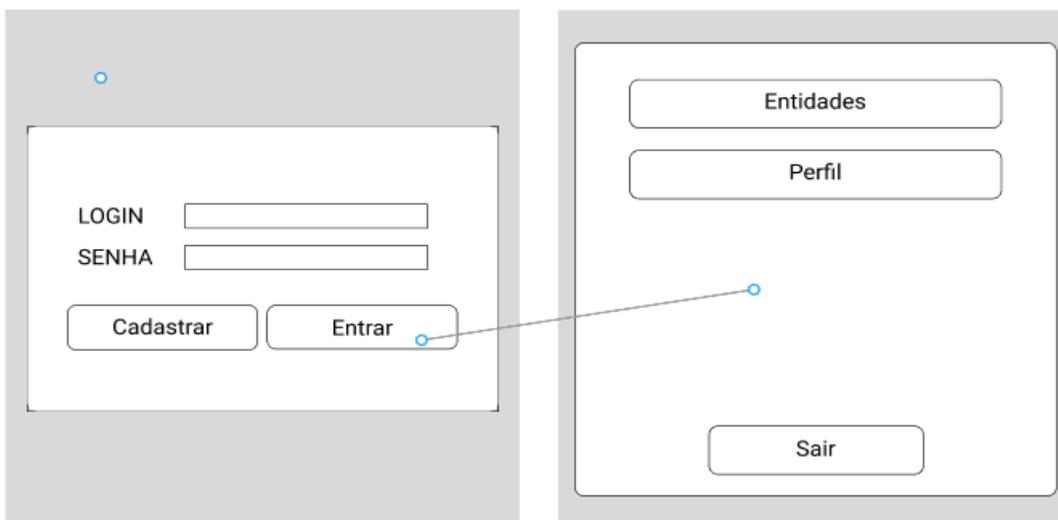
Figura 17 - Realizar Novo Cadastro



Fonte: Elaborado pelo autor via Figma

A Figura 17 ilustra o comportamento da aplicação durante o processo de Realizar Novo Cadastro, conforme descrito no requisito. O processo se inicia ao clicar no botão "Cadastrar-se". Em seguida, o fluxo leva à tela de Cadastro de Usuário, onde os dados solicitados são preenchidos. Ao clicar em "Realizar Cadastro", o usuário é redirecionado de volta à tela de Login.

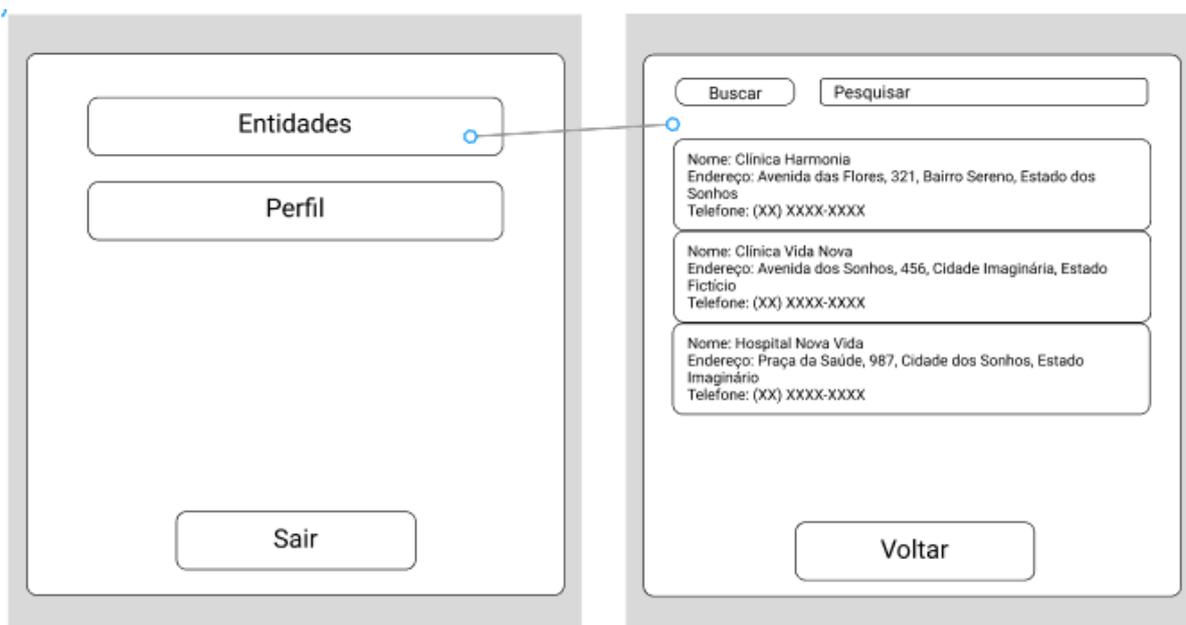
Figura 18 - Realizar Login



Fonte: Elaborado pelo autor via Figma

A Figura 18 ilustra o comportamento da aplicação durante o processo de Realizar Login, conforme descrito no requisito. O processo se inicia ao fornecer os dados solicitados: Login e Senha. Ao clicar em “Entrar” a tela é redirecionada para a tela inicial da aplicação.

Figura 19 - Listar Estabelecimentos



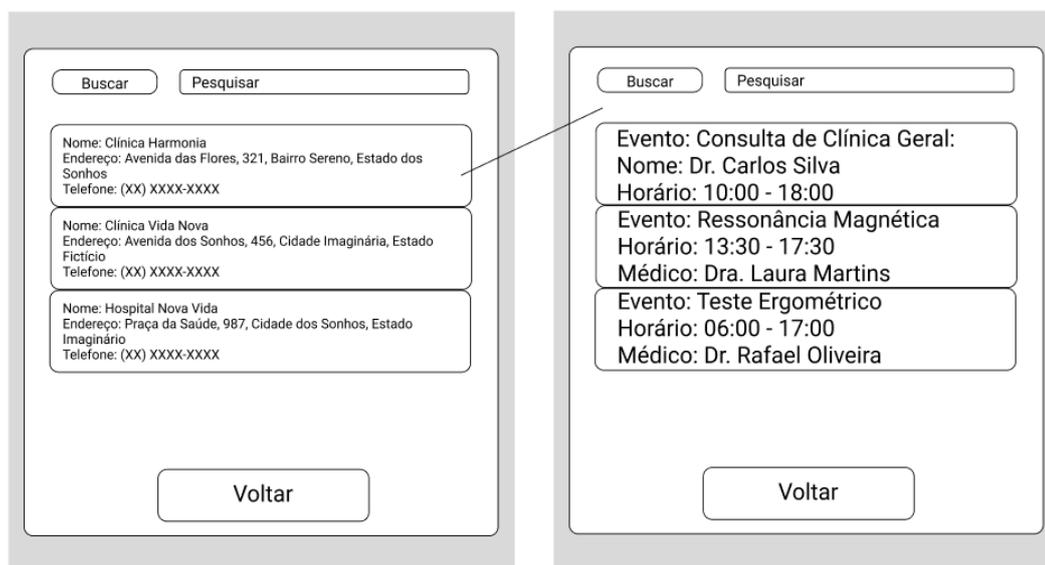
Fonte: Elaborado pelo autor via Figma

A Figura 19 ilustra o comportamento da aplicação durante o processo de Listar Estabelecimentos, conforme descrito no requisito. O processo se inicia após o usuário

realizar o login. Ao ser redirecionado para a tela inicial da aplicação, a opção "Entidades" é exibida. Ao selecioná-la, a tela muda para a tela de listagem de Estabelecimentos.

Uma alternativa para essa listagem é fornecer algum dado relevante e clicar no botão "Buscar". Nesse caso, a tela é recarregada e a lista de Estabelecimentos é atualizada com base nos critérios de busca fornecidos.

Figura 20 - Listar Eventos

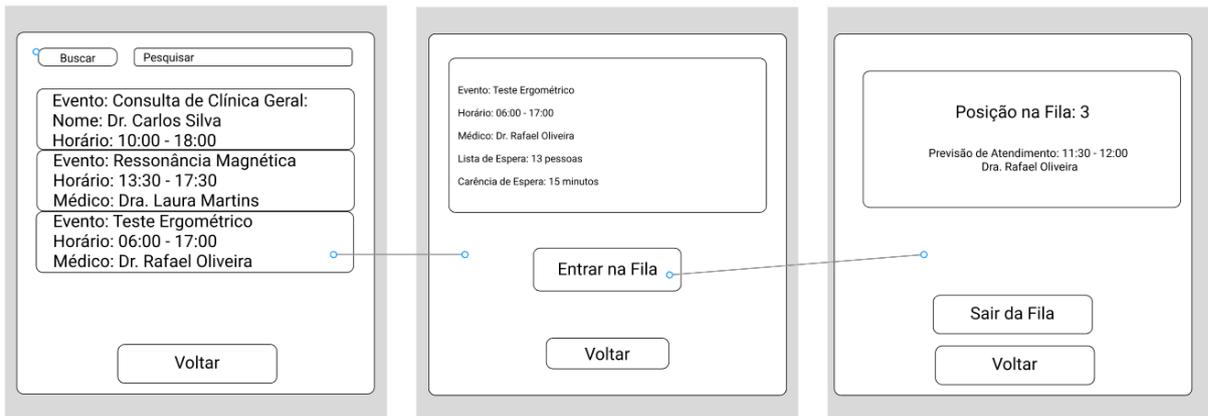


Fonte: Elaborado pelo autor via Figma

A Figura 20 ilustra o comportamento da aplicação durante o processo de Listar Eventos, conforme descrito no requisito. O processo se inicia após o usuário realizar o login e selecionar a opção "Entidades". Ao ser redirecionado para a tela de listagem de Estabelecimentos, uma lista de hospitais e/ou clínicas aparece, então o usuário escolhe uma das opções e a tela muda para a tela de listagem de Eventos.

Uma alternativa para essa listagem é fornecer algum dado relevante e clicar no botão "Buscar". Nesse caso, a tela é recarregada e a lista de eventos é atualizada com base nos critérios de busca fornecidos.

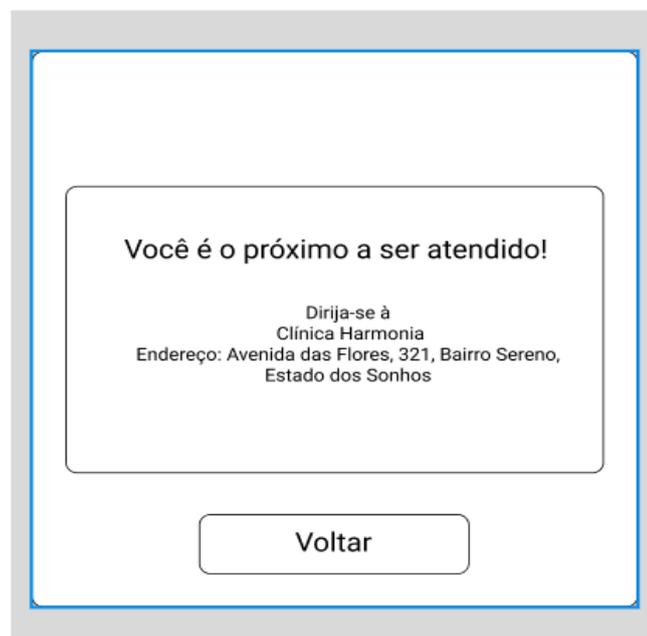
Figura 21 - Entrar em Filas de Espera



Fonte: Elaborado pelo autor via Figma

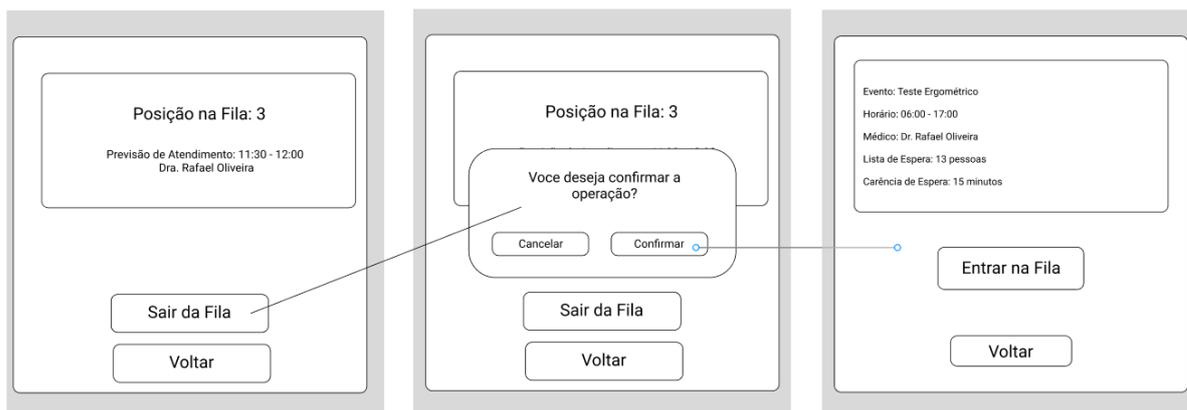
A Figura 21 ilustra o comportamento da aplicação durante o processo de Entrar em Filas de Espera, conforme descrito no requisito. O processo se inicia quando o usuário seleciona um dos eventos disponíveis. A tela é então redirecionada para a visualização do evento e, ao clicar em "Entrar na Fila", a tela muda para a tela de fila de espera. Nessa tela, algumas informações são exibidas, como a posição na fila e a estimativa de tempo de atendimento. Quando a vez do usuário chegar, a tela é atualizada e as informações da imagem Figura 22 irão aparecer.

Figura 22 - Atendimento do Usuário



Fonte: Elaborado pelo autor via Figma

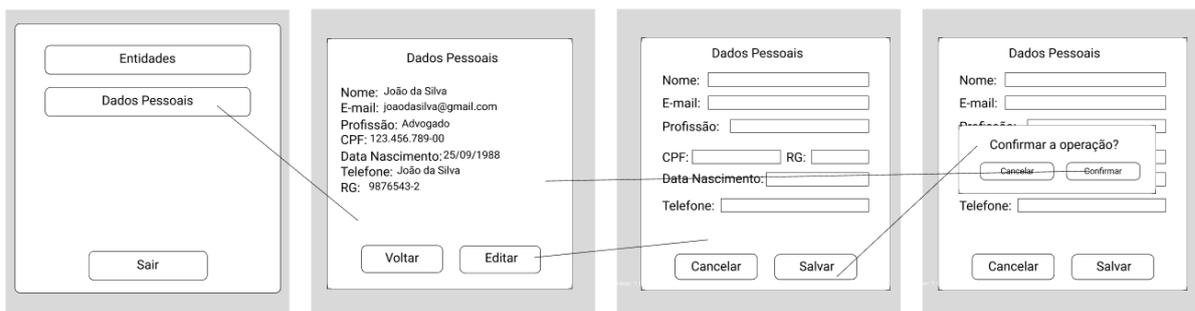
Figura 23 - Sair de Filas de Espera



Fonte: Elaborado pelo autor via Figma

A Figura 23 ilustra o comportamento da aplicação durante o processo de Sair de Filas de Espera, conforme descrito no requisito. O processo se inicia quando o usuário pressiona o botão "Sair da Fila". Em seguida, um pop-up é exibido solicitando a confirmação do usuário. Ao pressionar o botão "Confirmar", a tela é redirecionada para a tela de visualização da fila de espera.

Figura 24 - Editar Dados Pessoais



Fonte: Elaborado pelo autor via Figma

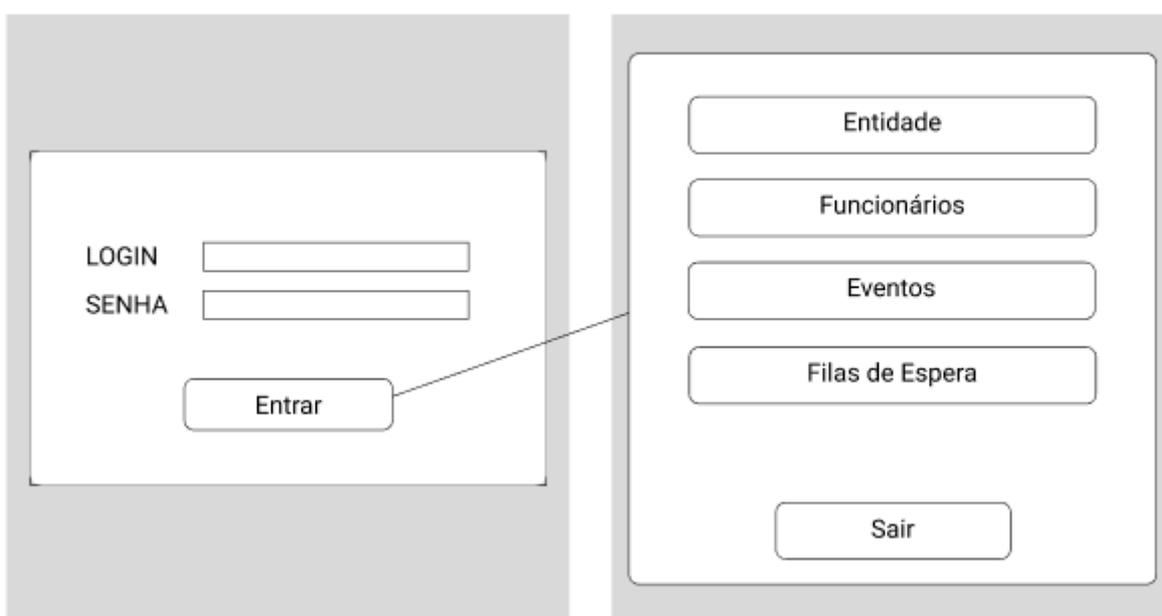
A Figura 24 ilustra o comportamento da aplicação durante o processo de Edição de Dados Pessoais, conforme descrito no requisito. O processo se inicia quando o usuário pressiona o botão "Dados Pessoais". Em seguida, é carregada a tela de Dados Pessoais. Ao pressionar o botão "Editar", a tela de Edição de Dados Pessoais é carregada, juntamente com os dados do usuário. Na tela de edição, o usuário pode fazer alterações nos dados e, ao clicar no botão "Salvar", um pop-up de confirmação

é exibido. Ao clicar em "Confirmar", os dados são alterados e a tela de Dados Pessoais é recarregada.

3.3.2. Gestor

Esta seção tem como objetivo ilustrar e descrever o processo de prototipagem das telas disponíveis para os usuários do tipo gestor no presente projeto.

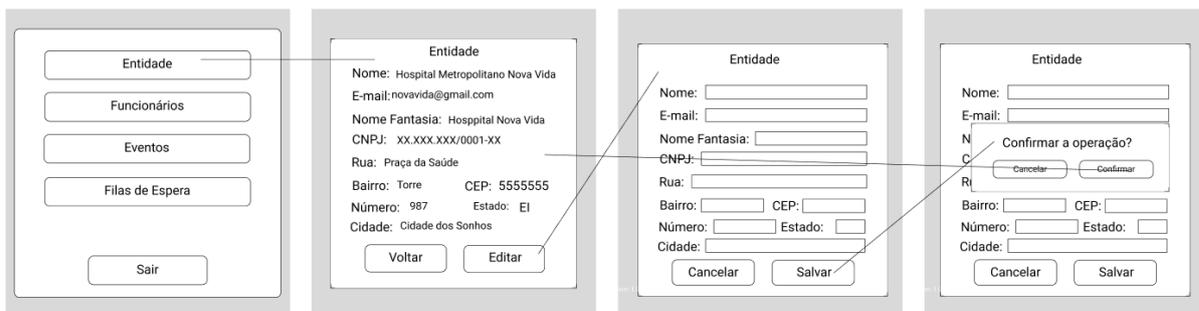
Figura 25 - Realizar Login



Fonte: Elaborado pelo autor via Figma

A Figura 25 ilustra o comportamento da aplicação durante o processo de Realizar o Login, conforme descrito no requisito. O processo é idêntico ao descrito na subseção i). A diferença está na tela inicial da aplicação, que exibe mais opções, como Funcionários, Eventos e Filas de Espera.

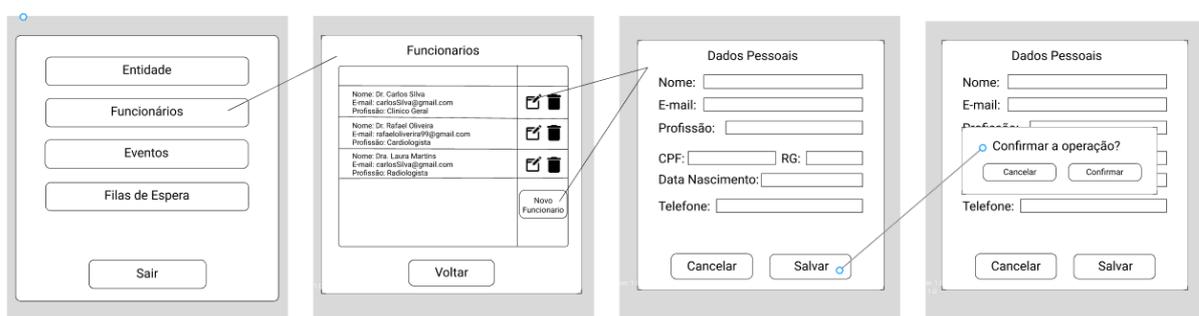
Figura 26 - Editar Estabelecimentos



Fonte: Elaborado pelo autor via Figma

A Figura 26 ilustra o comportamento da aplicação durante o processo de Editar Estabelecimento, conforme descrito no requisito. O processo se inicia na tela inicial da aplicação, quando o usuário pressiona o botão 'Entidade'. Em seguida, a tela carrega a visualização das informações do Estabelecimento. Ao pressionar o botão 'Editar', a tela é redirecionada e carrega a tela de Edição de Estabelecimento, com os dados do Estabelecimento previamente salvos na base de dados. Após realizar as alterações desejadas e pressionar o botão 'Salvar', um pop-up é exibido, solicitando a confirmação da edição. Ao clicar em 'Confirmar', as alterações são realizadas e a tela é atualizada para a visualização dos dados do Estabelecimento.

Figura 27 - Cadastrar/Editar Funcionário

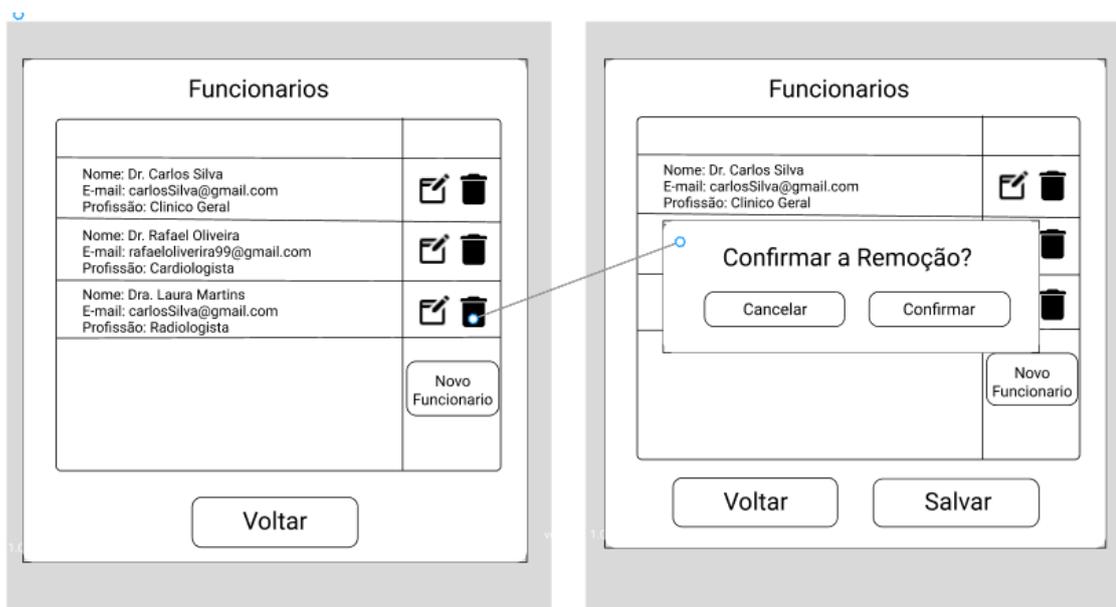


Fonte: Elaborado pelo autor via Figma

A Figura 27 ilustra o comportamento da aplicação durante o processo de Editar Funcionário e Cadastrar Funcionário, conforme descrito no requisito. O processo se inicia na tela inicial da aplicação, ao clicar no botão “Funcionários” e a tela redireciona para a listagem de Funcionários. A partir daí o processo diferencia-se em: Clicar no ícone de “Editar Funcionário” ou clicar no botão de “Novo Funcionário”. Em ambos os casos a tela é redirecionada para a de “Dados Pessoais”, no caso de “Editar

Funcionário”, os dados são carregados e preenchem os campos na página, caso contrário, esses campos permanecem vazios. Ao preencher os campos com os dados necessários e clicar no botão “Salvar” um pop-up é carregado solicitando a confirmação do usuário que ao clicar em “Confirmar” é redirecionado para a tela de listagem de Funcionários.

Figura 28 - Remover Funcionário



Fonte: Elaborado pelo autor via Figma

A Figura 28 ilustra o comportamento da aplicação durante o processo de Remover Funcionário, conforme descrito no requisito. O processo ocorre na tela de listagem dos Funcionários, então o usuário clica no ícone de “Remover Funcionário” e um pop-up é exibido, solicitando a confirmação do usuário que ao clicar em “Confirmar” é redirecionado para a tela de listagem de Funcionários.

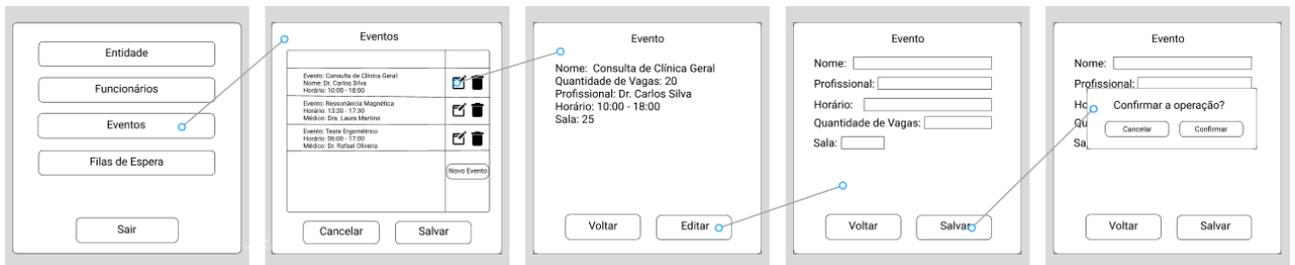
Figura 29 - Cadastrar Evento



Fonte: Elaborado pelo autor via Figma

A Figura 29 ilustra o comportamento da aplicação durante o processo de Cadastrar Evento, conforme descrito no requisito. O processo se inicia na tela inicial da aplicação, ao clicar no botão “Eventos”, e a tela é redirecionada para a listagem de Eventos. Em seguida, o usuário tem a opção de clicar no botão 'Novo Evento'. Ao selecionar essa opção, a tela de Cadastro de Evento é carregada, e a aplicação solicita que o usuário preencha os campos necessários para o cadastro. Após preencher os campos, o usuário clica no botão “Salvar”. Em seguida, um pop-up é exibido, solicitando a confirmação do usuário. Ao clicar em 'Confirmar', o usuário é redirecionado de volta para a tela de listagem de Eventos.

Figura 30 - Editar Evento

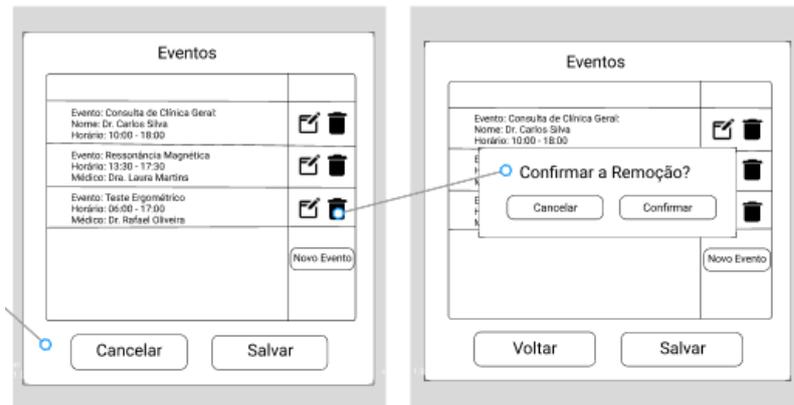


Fonte: Elaborado pelo autor via Figma

A Figura 30 ilustra o comportamento da aplicação durante o processo de Edição de Evento, conforme descrito no requisito. O processo se inicia na tela inicial da aplicação, ao clicar no botão “Eventos”, e a tela é redirecionada para a listagem de Eventos. Em seguida, o usuário tem a opção de clicar no botão “Editar Evento”. Ao selecionar essa opção, a tela carrega os dados do Evento e, ao pressionar o botão “Editar”, a aplicação carrega a tela de “Edição de Evento” e preenche os campos com

os dados recuperados da base de dados. O usuário pode realizar alterações nos campos conforme necessário. Ao concluir a edição e selecionar a opção “Salvar”, um pop-up é exibido, solicitando a confirmação do usuário. Ao clicar em “Confirmar”, o usuário é redirecionado de volta para a tela de listagem de Eventos.

Figura 31 - Remover Evento



Fonte: Elaborado pelo autor via Figma

A Figura 31 ilustra o comportamento da aplicação durante o processo de Remover Evento, conforme descrito no requisito. O processo ocorre na tela de listagem dos Eventos, então o usuário clica no ícone de “Remover Evento” e um pop-up é exibido, solicitando a confirmação do usuário que ao clicar em “Confirmar” é redirecionado para a tela de listagem de Evento.

Figura 32 - Remover Paciente da Lista de Espera



Fonte: Elaborado pelo autor via Figma

A Figura 32 ilustra o comportamento da aplicação durante o processo de Remover Paciente da Lista de Espera, conforme descrito no requisito. O processo se

inicia na tela inicial da aplicação, ao clicar no botão 'Filas de Espera'. Em seguida, a tela é redirecionada para a listagem de Listas de Espera. Quando o usuário seleciona o ícone "Editar Lista de Espera", a tela é novamente redirecionada para a tela de listagem de Pacientes na Lista de Espera. O usuário então pressiona o ícone "Remover Paciente", e um pop-up é exibido, solicitando a confirmação do usuário. Ao clicar em 'Confirmar', o paciente é removido da lista de espera, e o usuário é redirecionado de volta para a tela de listagem de Pacientes na Lista de Espera.

4. Conclusão e Considerações Futuras

Nesta seção serão apresentadas as considerações finais desse trabalho, cujo objetivo foi a idealização de uma aplicação de software, capaz de gerenciar filas de espera em ambientes médicos.

4.1 Conclusão

O presente projeto alcançou seu objetivo de idealizar um sistema de gerenciamento de filas de espera em hospitais e clínicas de saúde, utilizando técnicas de elicitação de requisitos e arquitetura de software. O desenvolvimento do sistema demonstrou o cumprimento satisfatório dos objetivos estabelecidos pelo autor. A implementação dos microsserviços permite atender às necessidades específicas de cada tipo de usuário, proporcionando uma experiência personalizada e eficiente. Cada microsserviço possui suas próprias responsabilidades e funcionalidades específicas, operando de forma isolada e sem interferências diretas dos demais microsserviços. Apesar da autonomia dos microsserviços, é importante destacar que eles podem interagir entre si quando necessário para garantir uma melhor autenticação e integridade dos dados. Dessa forma, o sistema desenvolvido neste projeto cumpriu os objetivos estabelecidos, oferecendo uma solução para o gerenciamento de filas de espera em hospitais e clínicas de saúde. O resultado final contribui para organização e gestão dos atendimentos, beneficiando tanto os pacientes quanto os gestores dessas instituições.

4.2. Considerações Futuras

Como trabalho futuro, o sistema poderia ser aprimorado para que seja possível a integração do mesmo com os sistemas já existentes no ambiente da saúde. Essa integração pode proporcionar uma maior sinergia entre as diferentes soluções tecnológicas e melhoraria a troca de informações entre elas. Também, poderia ser implementado o esquema de banco de dados do sistema capaz de indicar o relacionamento mais detalhado entre as entidades abordadas durante o desenvolvimento do trabalho. Isso poderia proporcionar uma estrutura sólida para armazenar e organizar as informações.

Por fim, é essencial que o trabalho seja efetivamente implementado e integrado a um sistema de software já existente no ambiente hospitalar ou clínico, a fim de comprovar sua funcionalidade e eficiência. Essa integração, sendo ela bem-sucedida, pode permitir a aplicação prática e real do trabalho, proporcionando benefícios concretos tanto para os profissionais de saúde quanto para os pacientes.

Referências Bibliográficas

PÁDUA, PFW. **Engenharia de Software Produtos**. Vol. 1. 4ª edição. Rio de Janeiro-RJ: Grupo GEN, v. 9, 2019. ISBN 9788521636724 Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788521636724/>. Acesso em: 29 mar. 2023.

JUNIOR, K. M. A. S.; TOMITA, Shiro; KOS, AOda. O problema da fila de espera para cirurgias otorrinolaringológicas em serviços públicos. **Rev Bras Otorrinolaringol**, v. 71, n. 3, p. 256-62, 2005. Disponível em: <https://doi.org/10.1590/S0034-72992005000300001>

KRUCHTEN, Philippe. Agility with the RUP. **Cutter IT journal**, v. 14, n. 12, p. 27-33, 2001.

MORAIS, I. S. **Engenharia de software**. 1º edição. Editora pearson, 2017.

MORAIS, Izabelly Soares D.; ZANIN, Aline. **Engenharia de software**. Grupo A, 2020. ISBN 9788595022539. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595022539/>. Acesso em: 01 jun. 2023.

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de software**. Grupo A, 2021. ISBN 9786558040118. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9786558040118/>. Acesso em: 28 mai. 2023.

REIS, Adalberto Faria dos; COSTA, Ivanir da. **Proposta de integração da engenharia de software nas estratégias empresariais**. Production, v. 15, p. 448-455, 2005. DOI: <https://doi.org/10.1590/S0103-65132005000300013>

SOMMERVILLE, Ian. et al. **Engenharia de software**. 8º edição. Editora pearson, 2007.