

**FATEC** São Caetano Do Sul – Antônio Russo

**Curso:** Análise e Desenvolvimento de Sistemas – ADS AMS

**Disciplina:** Projeto Integrador II

**Professor:** Carlos Veríssimo

Proposta técnica - Evoluti

### **IDENTIFICAÇÃO**

Tema do projeto: Evoluti – Evolução de tratamento fisioterápico

|  |  |  |
| --- | --- | --- |
| **Número Grupo: 2**    **Gerente do Grupo: Juliana Araujo** | | |
| **R.A** | Nome | **Papéis do Componente** |
| 1681432312003 | Bruno Carvalho Martins | DBA, Testador, QA |
| 1681432312036 | Eduardo Pielich Sanchez | DEV-Back |
| 1681432312018 | Juliana Alves de Araújo | GP, PO, Scrum Master, UX |
| 1681432312004 | Renato Alexandre | DEV-Front |

### **CONTEXTUALIZAÇÃO**

1. **Problema**

* Desafios no cotidiano de clínicas de fisioterapia, incluindo a falta de integração de registros dos pacientes, a carência de análises aprofundadas sobre os tratamentos e a dificuldade de acompanhamento do progresso dos pacientes ao longo do tempo.
* A solução tem como objetivo proporcionar uma gestão mais eficiente, integrar os dados dos pacientes, gerando análises dos tratamentos e aprimorando o acompanhamento da progressão dos pacientes durante seu processo de reabilitação.

1. **Solução Técnica para o problema**

* Implementação de um sistema de prontuário eletrônico inteligente, que centralizará todos os registros dos pacientes, fornecerá análises inteligentes sobre o progresso do tratamento, automatizará a geração de documentos clínicos e facilitará o acesso colaborativo à informação.

### **DETALHAMENTO**

1. **Levantamento dos Requisitos do Sistema de Software**

Este capítulo tem como objetivo apresentar o levantamento dos requisitos do Sistema de Software e a forma de extração dos Requisitos

1. **Requisitos funcionais**

* Web Page
* Home
* Nosso propósito
* Quem somos?
* Cadastro de usuários
* Nivelamento de usuários: Administrador, Fisioterapeuta e Estagiário

*Esse requisito tem como objetivo controlar o acesso dos tipos de usuários na plataforma.*

* Cadastro de pacientes
* Cadastro completo
* Anexo de arquivos importantes
* Busca
* Sistema de filtragem

*Esse requisito tem como objetivo facilitar a busca por informações específicas na plataforma, tanto em pacientes como em usuários.*

* Atendimentos
* Avaliação
* Evolução

*Esses requisitos têm como objetivo registrar o progresso dos pacientes: avaliação marcando início do tratamento de uma lesão e evolução registrando o decorrer das sessões de fisioterapia.*

* Anexo de documentos e imagens
* Anexo de receitas, exames, atestados

*Esses requisitos têm como objetivo permitir o upload de arquivos relacionados aos atendimentos.*

* Ciclo de diagnósticos

*Esse requisito tem como objetivo acompanhar os diagnósticos ao longo do tratamento.*

* Marcação de pontos de dor

*Esse requisito tem como objetivo registrar os pontos de dor dos pacientes, de maneira gráfica.*

* Questionários

*Esse requisito tem como objetivo fornecer questionários tradicionais da fisioterapia para medir o nível da lesão nas avaliações.*

* Prontuário Eletrônico
* Histórico de atendimentos

*Esse requisito tem como objetivo fornecer a lista de atendimentos registrados.*

* Visualização dos pontos de dor

*Esse requisito tem como objetivo visualizar os pontos de dor dos pacientes, de maneira gráfica.*

* Documentação
* Termo de Consentimento para atendimentos

*Esse requisito tem como objetivo garantir que os pacientes forneçam consentimento informado para os procedimentos fisioterapêuticos realizados.*

* Termo LGPD

*Esse requisito tem como objetivo assegurar o cumprimento das regulamentações de proteção de dados, protegendo a privacidade e confidencialidade das informações dos pacientes*.

* Encaminhamentos

*Esse requisito tem como objetivo facilitar o encaminhamento de pacientes para outros profissionais de saúde ou especialistas, quando necessário.*

* Cartão de registro de atendimentos

*Esse requisito tem como objetivo fornecer uma lista com data e quantidade dos atendimentos realizados com cada paciente.*

* PDF Prontuário eletrônico

*Esse requisito tem como objetivo permitir a geração de um documento em formato PDF contendo o prontuário eletrônico completo de cada paciente, para compartilhamento e arquivamento.*

* Declaração de comparecimento

*Esse requisito tem como objetivo fornecer aos pacientes uma declaração oficial de comparecimento aos atendimentos, quando necessário para comprovação.*

* Atestados

*Esse requisito tem como objetivo permitir a emissão de atestados médicos pelos fisioterapeutas, quando apropriado.*

* 1. **Requisitos não funcionais**
* **[RNF001] - Segurança**  
  Este requisito tem como objetivo garantir a segurança dos dados do paciente através de medidas como criptografia de dados, autenticação de usuários e controle de acesso.
* **[RNF002] - Privacidade**  
  Este requisito tem como objetivo assegurar a privacidade dos dados do paciente, em conformidade com as regulamentações de privacidade de dados, como a GDPR (Regulamento Geral de Proteção de Dados).
* **[RNF003] - Desempenho**  
  Este requisito tem como objetivo fazer o site ser responsivo e eficiente, mesmo sob carga pesada, para evitar atrasos no agendamento de consultas e tratamentos.
* **[RNF004] - Usabilidade**  
  Este requisito tem como objetivo projetar uma interface de usuário intuitiva e fácil de usar, permitindo que os usuários naveguem pelo sistema sem dificuldades.
* **[RNF005] - Escalabilidade**  
  Este requisito tem como objetivo projetar o sistema de forma a ser escalável, permitindo que ele cresça conforme a demanda e o número de pacientes e usuários aumente.
* **[RNF006] - Disponibilidade:**  
  Este requisito tem como objetivo garantir alta disponibilidade do sistema, minimizando o tempo de inatividade e garantindo que os usuários possam acessar o sistema quando necessário.
* **[RNF007] - Compatibilidade:**  
  Este requisito tem como objetivo assegurar que o sistema seja compatível com diferentes dispositivos e navegadores, permitindo que os usuários acessem o software de qualquer lugar e dispositivo.
* **[RNF008] - Manutenibilidade**  
  Este requisito tem como objetivo projetar o sistema de forma modular e bem documentada, facilitando a manutenção e a adição de novas funcionalidades no futuro.
* **[RNF009] - Confiabilidade:**  
  Este requisito tem como objetivo garantir que o sistema seja confiável, minimizando erros e falhas que possam comprometer a integridade dos dados do paciente.
* **[RNF010] - Backup e Recuperação:**  
  Este requisito tem como objetivo implementar rotinas de backup regulares para garantir a disponibilidade e a integridade dos dados do paciente em caso de falha do sistema ou desastres.
  1. **Modelagem Funcional**
* Diagrama

  Descrição gerada automaticamente**Diagrama de Caso de Uso – Visão Geral**
* **Caso de Uso – Caso de Uso #1 – Faz Login**
  + **Diagrama de Caso de Uso #1**

Diagrama

Descrição gerada automaticamente

* **Detalhamento do Caso de Uso #1.1 - Acessa Tela de Login**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 1.1 - Acessa Tela de Login |
| **Atores** | Administrador de Clínica, Fisioterapia e Estagiário em Fisioterapia |
| **Trigger** | Necessidade de acessar o sistema |
| **Pré-Requisitos** | Ter acesso a página inicial |
| **Fluxo de Eventos** | Um dos tipos de usuário acessa a tela inicial; Pressiona o botão de “Login”. |

* **Detalhamento do Caso de Uso #1.2 - Insere as credenciais**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 1.2 - Insere as credenciais |
| **Atores** | Administrador de Clínica, Fisioterapia e Estagiário em Fisioterapia |
| **Trigger** | Necessidade de acessar o sistema |
| **Pré-Requisitos** | Ter acesso a página inicial; Conter as credenciais cadastradas por um Administrador da Clínica |
| **Fluxo de Eventos** | Um dos tipos de usuário clica em um campo; Preenche com as credenciais; Pressiona o botão “Entrar” ou Pressiona o “Enter” do teclado. |

* **Detalhamento do Caso de Uso #1.3 - Recuperação de Senha**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 1.3 - Recuperação de Senha |
| **Atores** | Administrador de Clínica, Fisioterapia e Estagiário em Fisioterapia |
| **Trigger** | Necessidade de recuperar a senha esquecida |
| **Pré-Requisitos** | Ter acesso a página inicial; Conter as credenciais cadastradas por um Administrador da Clínica |
| **Fluxo de Eventos** | Um dos tipos de usuário clica na opção “Esqueceu a senha?”; Usuário preenche a credencial solicitada; Um email é enviado com o link de recuperação; O usuário altera a senha. |

* **Detalhamento do Caso de Uso #1.4 - Validação das credenciais**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 1.4 - Validação das credenciais |
| **Atores** | Sistema |
| **Trigger** | Necessidade de verificar se as credenciais são válidas |
| **Pré-Requisitos** | Um dos tipos de usuário querer acessar o sistema |
| **Fluxo de Eventos** | Pega os dados inseridos nos campos; Consulta o banco de dados como operação de READ; Compara os valores encontrados junto aos inseridos; Retorna uma caixa de texto com a resposta. |

* **Detalhamento do Caso de Uso #1.5 - Nivelamento de Acesso**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 1.5 - Nivelamento de Acesso |
| **Atores** | Sistema |
| **Trigger** | Necessidade de diferenciar |
| **Pré-Requisitos** | Conter as credenciais cadastradas por um Administrador da Clínica |
| **Fluxo de Eventos** | Após a validação, é feito um READ no tipo de usuário; Após descobrir o tipo, ele é mandado para página de seu respectivo cargo. |

* **Caso de Uso - Caso de Uso #2 - Fazer novos cadastros**
  + **Diagrama de Caso de Uso #2**

Diagrama

Descrição gerada automaticamente

* **Detalhamento do Caso de Uso #2.1 - Acessar tela inicial**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.1 - Acessa tela inicial |
| **Atores** | Administrador de Clínica |
| **Trigger** | Necessidade de executar suas tarefas |
| **Pré-Requisitos** | Ter um cadastro no banco de dados e acessado por ele. |
| **Fluxo de Eventos** | Após a inserção do login, o administrador é levado para a tela inicial. |

* **Detalhamento do Caso de Uso #2.2 - Acessar tela de cadastros**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.2 - Acessa tela de cadastros |
| **Atores** | Administrador de Clínica |
| **Trigger** | Necessidade de adicionar novos usuários ou pacientes ao sistema |
| **Pré-Requisitos** | Ter o acesso a tela inicial do Administrador |
| **Fluxo de Eventos** | Acessar o barra lateral com funcionalidades; Selecionar o cadastro. |

* **Detalhamento do Caso de Uso #2.3 - Selecionar novo tipo de cadastros**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.3 - Selecionar novo tipo de cadastros |
| **Atores** | Administrador de Clínica |
| **Trigger** | Necessidade de adicionar novos registros ao banco de dados |
| **Pré-Requisitos** | Ter o acesso a tela inicial do Administrador; |
| **Fluxo de Eventos** | Acessar o barra lateral com funcionalidades; Selecionar o ícone de Cadastro. |

* **Detalhamento do Caso de Uso #2.4 - Selecionar entre Paciente e Usuário**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.4 - Selecionar entre paciente e Usuário |
| **Atores** | Administrador de Clínica |
| **Trigger** | Diferenciar os cadastros no banco de dados |
| **Pré-Requisitos** | Ter o acesso a tela inicial do Administrador; Selecionar a opção de cadastro na barra lateral. |
| **Fluxo de Eventos** | Acessar a tela de cadastro; Selecionar a opção de “Novo Paciente” ou “Novo Usuário”. |

* **Detalhamento do Caso de Uso #2.5 - Nivelamento de Usuário**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.5 - Nivelamento de Usuário |
| **Atores** | Administrador de Clínica, Sistema |
| **Trigger** | Necessidade de diferenciar os níveis de acesso entre os usuários |
| **Pré-Requisitos** | Estar logado no sistema e ter acessado a página de “Novos Usuários”. |
| **Fluxo de Eventos** | Acessar a tela de cadastro de usuários; Selecionar qual o tipo de usuário entre os três disponíveis na tela; O Sistema adapta a tela e registrará no banco de dados de acordo com o usuário escolhido. |

* **Detalhamento do Caso de Uso #2.6 – Preencher credenciais**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.6 – Preencher credenciais |
| **Atores** | Administrador de Clínica |
| **Trigger** | Necessidade de criar um registro |
| **Pré-Requisitos** | Estar logado no sistema e ter acessado a página uma das duas páginas de cadastros: “Novo Paciente” ou “Novo Usuário”. |
| **Fluxo de Eventos** | Selecionar os campos mostrados na tela; Preencher os campos com os dados pedidos; Finalizar o cadasto. |

* **Detalhamento do Caso de Uso #2.7 – Validar credenciais**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.7 – Validar credenciais |
| **Atores** | Sistema |
| **Trigger** | Inserir o novo registro corretamente ao banco de dados |
| **Pré-Requisitos** | O Administrador deve ter inserido pressionado o botão para finalizar o cadastro |
| **Fluxo de Eventos** | Pegar as informações dadas nos campos; Formar um objeto JSON; Mandar uma requisição de CREATE para o banco de dados |

* **Detalhamento do Caso e Uso #2.8 – Verificar Preenchimento de Campos Obrigatórios**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.8 – **Verificar Preenchimento de Campos Obrigatórios** |
| **Atores** | Sistema |
| **Trigger** | Necessidade de inserção de itens obrigatórios para o cadastro |
| **Pré-Requisitos** | O Administrador deve ter inserido pressionado o botão para finalizar o cadastro |
| **Fluxo de Eventos** | Verificar se todos os campos dados como obrigatórios estão preenchidos; Avisar no caso de algum campo vazio; Prosseguir com a inserção no banco de dados. |

* **Detalhamento do Caso e Uso #2.9 – Cadastro Finalizado**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 2.9 – **Cadastro Finalizado** |
| **Atores** | Sistema, SGDB |
| **Trigger** | Inserir o novo registro ao banco de dados |
| **Pré-Requisitos** | A verificação dos dados deve ter sido realizada; Banco de dados conectado. |
| **Fluxo de Eventos** | Analisar a requisição de CREATE do sistema; Aprovar a requisição; Inserir no banco de dados. |

* **Caso de Uso - Caso de Uso #3 - Fazer novos cadastros**
  + **Diagrama de Caso de Uso #3**

**Diagrama

Descrição gerada automaticamente**

* **Detalhamento do Caso e Uso #3.1 – Acessa tela inicial**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.1 - Acessa tela inicial |
| **Atores** | Fisioterapeuta, Estagiário em Fisioterapia |
| **Trigger** | Necessidade de executar suas tarefas |
| **Pré-Requisitos** | Ter um cadastro no banco de dados e acessado por ele. |
| **Fluxo de Eventos** | Após a inserção do login, o fisioterapeuta ou o estagiário são levados para a tela inicial. |

* **Detalhamento do Caso e Uso #3.2 – Acessa tela de atendimento**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.2 - Acessa tela de atendimento |
| **Atores** | Fisioterapeuta, Estagiário em Fisioterapia |
| **Trigger** | Necessidade de registra um atendimento realizado |
| **Pré-Requisitos** | Ter o acesso a tela inicial do profissional |
| **Fluxo de Eventos** | Acessar a barra lateral com funcionalidades; Selecionar o ícone de “Atendimento”. |

* **Detalhamento do Caso e Uso #3.3 – Navegar pela lista de pacientes**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.3 - Navegar pela lista de pacientes |
| **Atores** | Fisioterapeuta, Estagiário em Fisioterapia |
| **Trigger** | Encontrar o paciente atendido |
| **Pré-Requisitos** | Ter o acesso a tela de avaliação |
| **Fluxo de Eventos** | Procurar na lista o nome do paciente; Selecionar e prosseguir ao próximo estágio do atendimento. |

* **Detalhamento do Caso e Uso #3.4 – Filtrar Pacientes**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.4 – Filtrar Pacientes |
| **Atores** | Fisioterapeuta, Estagiário em Fisioterapia |
| **Trigger** | Encontrar o paciente atendido |
| **Pré-Requisitos** | Ter o acesso a tela de avaliação |
| **Fluxo de Eventos** | Escrever o nome do paciente no campo de filtragem presente lista o nome do paciente; Selecionar e prosseguir ao próximo estágio do atendimento. |

* **Detalhamento do Caso e Uso #3.5 – Fazer Avaliação**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.5 – Fazer avaliação |
| **Atores** | Fisioterapeuta, SGBD |
| **Trigger** | Registrar avaliação feita pelo fisioterapeuta. |
| **Pré-Requisitos** | Selecionar o paciente correto. |
| **Fluxo de Eventos** | Selecionar a opção de registrar uma “Avaliação”; Preencher os relatórios; Anexar algum documento se necessário; Confirmar o registro da avaliação; Armazenar no banco de dados. |

* **Detalhamento do Caso e Uso #3.6 – Fazer Evolução**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.6 – Fazer evolução |
| **Atores** | Fisioterapeuta, Estagiário em Fisioterapia, SGBD |
| **Trigger** | Registrar evolução feita pelo fisioterapeuta ou pelo estagiário. |
| **Pré-Requisitos** | Selecionar o paciente correto. |
| **Fluxo de Eventos** | Selecionar a opção de registrar uma “Evolução”; Preencher os relatórios; Anexar algum documento se necessário; Confirmar o registro da evolução; Armazenar no banco de dados. |

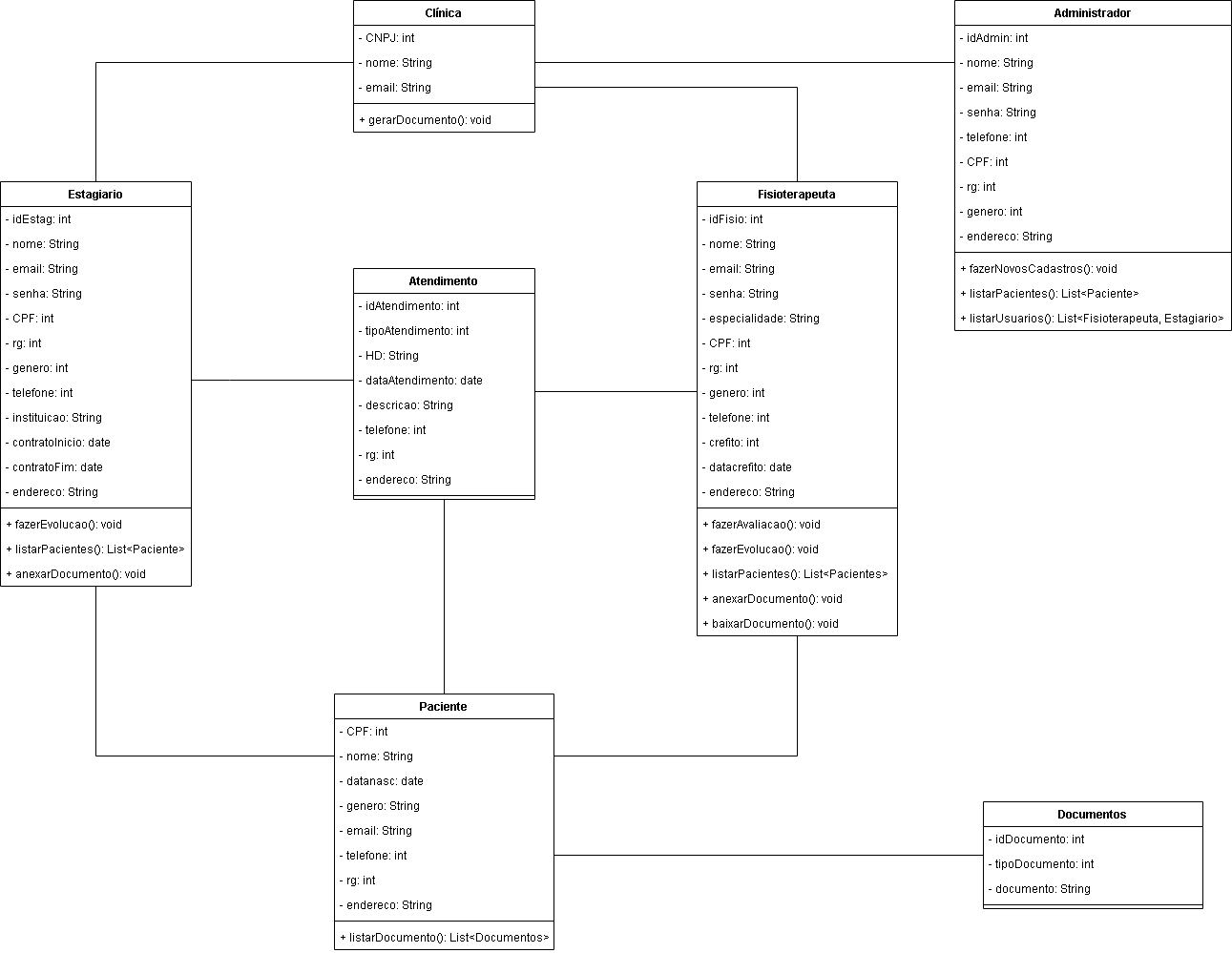
* **Detalhamento do Caso e Uso #3.7 – Visualizar documentos atrelados ao paciente**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.7 – Visualizar documentos atrelados ao paciente |
| **Atores** | Fisioterapeuta, SGBD |
| **Trigger** | Necessidade de ver algum documento atrelado ao paciente |
| **Pré-Requisitos** | Selecionar o paciente que deseja visualizar algum documento. |
| **Fluxo de Eventos** | Selecionar um dos documentos atrelados ao paciente selecionado, tudo trago pelo SGBD |

* **Detalhamento do Caso e Uso #3.8 – Baixar documento específico**

|  |  |
| --- | --- |
| **Nome do Caso de Uso** | 3.8 – Visualizar documentos atrelados ao paciente |
| **Atores** | Fisioterapeuta, SGBD |
| **Trigger** | Baixar para melhor visualização do documento. |
| **Pré-Requisitos** | Selecionar o paciente que deseja visualizar algum documento. |
| **Fluxo de Eventos** | Selecionar um dos documentos atrelados ao paciente selecionado, tudo trago pelo SGBD; Será automaticamente baixado o documento. |

* 1. **Diagrama de Classes**

****

* 1. **Arquitetura de Software**

**Diagrama

Descrição gerada automaticamente**

1. **ESCOPO**

**4 - Definição das Ferramentas Tecnológicas**

Visando melhorar a infraestrutura dos códigos com novas tecnologias do mercado, a equipe de desenvolvimento fez pesquisas e decidiu transferir todo o software, que se encontrava em tecnologias web como HTML, CSS, JavaScript e PHP, para as seguintes linguagens:  
  
 **React**  
 A biblioteca do JavaScript, o React, será usada para construir toda a interface de usuário dinâmica e interativa, ou seja, o frontend. Por levar uma abordagem por componentes, acaba por facilitar a criação das interfaces intuitivas e responsivas para o usuário. Com a implementação do React, o reuso de componentes é uma abordagem plausível para o projeto, resultando em um desenvolvimento mais eficiente, e em caso de manutenções, elas acabam sendo mais simplificadas.

**TailwindCSS**  
 Juntamente ao React, para a criação de interfaces, optamos por utilizar o TailwindCSS, uma biblioteca do CSS de baixo nível. O Tailwind possuí diversas classes utilitárias para estilizar todos os elementos produzidos no software, tendo uma abordagem funcional falicitando a estilização consistente e responsiva, sem necessidade de escrever arquivos .css personalizado para cada seção.

**Node.js com Express.js**  
 Passando para a parte lógica do software, o backend, foi decidido a utilização do Node.js e Express.js. O Node.js é uma plataforma de server-side que permite a construção de softwares escaláveis com alto desempenho. Já o Express.js é uma biblioteca web para o Node,js que simplifica todo o processo de criações de APIs RESTful. Com essa combinação, é possível oferecer um desempenho robusto com escalabilidade efiente para lidar com grandes volumes de dados.

**MongoDB**  
 Junto com todas as linguagens ditas, é necessário um banco de dados para armazenar os dados das clínicas, pacientes, avaliações e mais. Para isso foi decidido usar o MongoDB, um banco de dados NoSQL, altamente escalável e orientado a documentos. Tendo sua estrutura flexível de documentos JSON, permitindo que os dados sejam modelados de forma mais natural, além de uma integração nativa com Node.js, oferecendo operações de database rápidas e eficientes, ele será necessário para armazenar todo o volume de dados que nosso software pode gerar.

**Git e GitHub**    
 Utilizaremos o Git para gerenciar o código-fonte, permitindo o versionamento seguro e eficiente do código. Então, podemos colaborar com desenvolvedores de forma assíncrona, rastrear e revisar alterações de código e reverter para versões anteriores se necessário, garantindo a integridade e a qualidade do código-fonte, e armazenando-o remotamente no GitHub.

Ao utilizar React e TailwindCSS no frontend, Node.js com Express.js e MongoDB no backend e o Git para o versionamento do código-fonte, estamos aproveitando um conjunto de ferramentas modernas que oferecem uma combinação de eficiência, desempenho e flexibilidade para desenvolver um software que atenda às necessidades dos profissionais de saúde e de suas respectivas tarefas.

**Figma**  
 O Figma é uma ferramenta de design colaborativa que permite criar protótipos interativos e layouts de interface de usuário de forma eficiente. Utilizaremos o Figma para criar wireframes e protótipos das telas, permitindo uma visualização prévia das funcionalidades e fluxos de usuário.

**Jira**  
 Utilizaremos o Jira para criar e rastrear histórias de usuário, definir sprints, atribuir tarefas e acompanhar o progresso do desenvolvimento com base no Scrum. Com ele, podemos organizar o trabalho de forma transparente e colaborativa, priorizando as funcionalidades mais importantes e garantindo entregas incrementais e regulares.

**4.1 - Definição de Testes**

Definir e executar testes é essencial para garantir que nosso software de gestão de pacientes e documentos em fisioterapia funcione sem falhas e atenda às necessidades dos profissionais de saúde e dos pacientes.

**Testes Unitários**

São como pequenas verificações de cada parte do código para garantir que tudo funcione corretamente. É como testar cada peça de um quebra-cabeça para ter certeza de que elas se encaixam perfeitamente.

**Testes de Integração**

Verificam se todas as partes do sistema funcionam bem juntas. É como garantir que os diferentes instrumentos de uma orquestra toquem em harmonia.

**Testes End-to-End (E2E)**

Simulam a experiência real do usuário, garantindo que todas as funcionalidades do software estejam acessíveis e operacionais. É como testar um carro para ter certeza de que ele não só funciona, mas também é fácil de dirigir.

**Testes de Desempenho e Escalabilidade**

Garantem que o software possa lidar com muitos usuários ao mesmo tempo e permanecer rápido e estável. É como garantir que uma ponte possa suportar o tráfego pesado sem desmoronar.

Ao realizar esses testes, estamos investindo na qualidade e confiabilidade do nosso software, o que é fundamental para oferecer uma experiência superior tanto para os profissionais de saúde quanto para os pacientes que utilizam nossa plataforma.