CENTRO UNIVERSITÁRIO UNINTER

TECNÓLOGO EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

PROJETO: DESENVOLVIMENTO BACK-END

AMANDA MAJCZAK THEODORO DA SILVA SPADER

RU.: 2481377

PAP ABERLADO LUZ

Prof. Winston Sen Lun Fung, Me.

2025.

Sumário

[1. INTRODUÇÃO 3](#_Toc203428890)

[1.1. Contexto do Estudo de Caso 3](#_Toc203428891)

[1.2. Objetivos do Projeto 4](#_Toc203428892)

[1.3. Principais Usuários 4](#_Toc203428893)

[2. ANÁLISE E REQUISITOS 5](#_Toc203428894)

[3. MODELAGEM E ARQUITETURA 8](#_Toc203428895)

[3.1. Arquitetura do Sistema 8](#_Toc203428896)

[3.2. Diagrama de Classes 9](#_Toc203428897)

[3.3. Diagrama Entidade-Relacionamento (DER) 10](#_Toc203428898)

[3.4. Endpoints da API 10](#_Toc203428899)

[3.5. Tecnologias de Persistência 12](#_Toc203428900)

[4. IMPLEMENTAÇÃO 12](#_Toc203428901)

[4.1. Tecnologias Utilizadas 12](#_Toc203428902)

[4.2. Estrutura do Sistema 13](#_Toc203428903)

[4.3. Telas da Prototipação 16](#_Toc203428904)

[4.4. Trechos de Códigos 17](#_Toc203428905)

[4.5. Exemplo de Fluxo 20](#_Toc203428906)

[5. PLANO DE TESTES 21](#_Toc203428907)

[5.1. POST /administrador 22](#_Toc203428908)

[5.2. GET /administrador 22](#_Toc203428909)

[5.3. GET /paciente/:idAdm 23](#_Toc203428910)

[6. CONCLUSÃO 23](#_Toc203428911)

[REFERÊNCIAS 24](#_Toc203428912)

[ANEXO A – CASO DE USO 25](#_Toc203428913)

[ANEXO B – FICHA DE ANAMNESE 26](#_Toc203428914)

[ANEXO C – PRONTUÁRIO DE TRIAGEM 27](#_Toc203428915)

[ANEXO D – MODELO DE CARTEIRA DE ENFERMEIRO 28](#_Toc203428916)

[ANEXO E – MODELO CARTEIRA DE FARMACÊUTICO 29](#_Toc203428917)

[ANEXO F – MODELO CARTEIRA DE MÉDICO 30](#_Toc203428918)

## INTRODUÇÃO

Nos últimos anos, o crescimento da indústria de Saúde e Bem-estar tem aumentado cada vez mais, contendo grande influência das mídias sociais. Com isso, as empresas procuram sempre melhorar seus serviços para conseguir se sobressair a suas concorrentes no mercado (Agência O Globo, 2023).

Além disso, o tema de gestão hospitalar tem sido analisado e trabalhado cada vez mais por conta dos avanços tecnológicos, conforme (Lima *et al.*, 2024):

Com o surgimento de novas tecnologias, modelos de gestão inovadores e uma maior conscientização sobre a importância da qualidade no atendimento, os gestores hospitalares estão buscando constantemente maneiras de melhorar a eficiência e eficácia dos serviços de saúde no país.

A gestão Hospitalar é um tema cada vez mais falado por conta da necessidade que vem sendo trabalhado e levado mais a sério cada vez mais, principalmente após a pandemia. Essa área trabalha no contexto da união entre saúde, administração e tecnologia, onde “processos e estratégias voltadas para administrar e organizar os recursos de uma instituição de saúde, como hospitais e clínicas.” (Riviere, 2024).

Para que a Gestão Hospitalar esteja dentro do seu propósito, o ideal é algo que traga a integração entre todas as áreas envolvidas e realize um fluxo eficiente, com todos os envolvidos com seus papeis desenvolvidos. A implementação desse sistema é de extrema relevância para garantir a eficiência operacional, a segurança no atendimento e a conformidade com normas sanitárias. A informatização dos processos contribui para a redução de erros, otimização de recursos e tomada de decisões mais assertivas por parte da gestão hospitalar.

### Contexto do Estudo de Caso

O estudo de caso para esse projeto foi para a criação de um Sistema de Gestão Hospitalar e de Serviços de Saúde (SGHSS), onde a instituição fictícia VidaPlus necessita de um sistema que atenda aos requisitos de sua administração, que envolve hospitais, clínicas de bairro, laboratórios e equipes de home care. Isso trás a necessidade crescente de sistemas informatizados no ambiente hospitalar, visando a melhoria da gestão de pessoal, pacientes e suprimentos.

O Sistema de Gestão tem como proposta de:

* Gerenciar os seus pacientes, controlando suas consultas, exames e prontuários.
* Gerenciar seus profissionais de saúde, como médicos, enfermeiros e farmacêuticos.
* Permitir que os médicos realizem consultas e monitorem suas agendas, assim como enfermeiros controlem suas atividades.
* Gerenciamento da Administração Hospitalar como Leitos, Suprimentos e realização de relatórios.
* Possibilitação de realização de Telemedicinas.

Para que esse projeto fosse melhor executado, realizou-se pesquisas para saber como funciona o sistema hospitalar, bem como as funções de cada pessoa ali dentro e quais são os dados fundamentais para poder realizar as manipulações de dados de forma mais próxima a realidade.

### Objetivos do Projeto

Os objetivos desse projeto, conforme o Estudo de Caso, é o desenvolvimento de um sistema que realize esse gerenciamento hospitalar, permitindo registros e monitoramentos de seus pacientes e seus profissionais de saúde, além da parte estrutural como Leito e Internações, e suprimentos em estoque. Com isso, haverá um único local em que se mantenha todo o gerenciamento em único local.

### Principais Usuários

Os principais usuários foram definidos conforme o estudo de caso:

* Administrador: responsável pelos cuidados estruturais e gerenciamento de pessoal, como Médicos, Enfermeiros e Farmacêuticos cadastrados, além dos leitos e suprimentos em estoque, monitorando todas as áreas necessárias para um bom gerenciamento.
* Profissionais de Saúde: os profissionais de saúde estão envolvidos diretamente nesse gerenciamento, trabalhando conforme sua área:
  + Médicos: gerenciando sua agenda médica, realizando consultas e prescrições médicas, realizando internações e organizando seus pacientes da melhor forma possível.
  + Enfermeiros: Tendo um maior controle sobre o que é necessário realizar em cada paciente e visualizar as atividades para aquele dia, seguindo as orientações médicas passadas.
  + Farmacêuticos: gerenciando o estoque de suprimentos conforme as demandas que são recebidas, sem desperdícios desnecessários.
* Pacientes: que poderão agendar consultas e acompanhar seu prontuário médico, de forma facilitada e em um único lugar.

## ANÁLISE E REQUISITOS

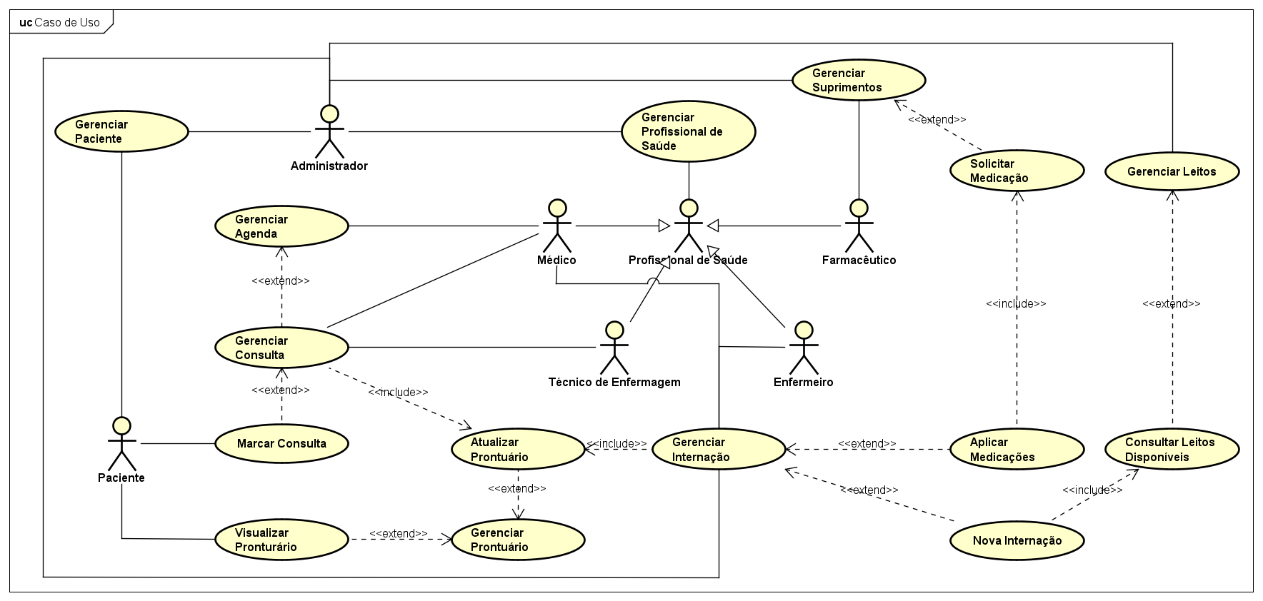
A lista de requisitos foi criada com base na análise da documentação de estudo de caso do sistema proposto. Para tal definição desses requisitos, baseou-se na diferença entre Requisitos Funcionais e Requisitos Não Funcionais.

Os Requisitos Funcionais são aqueles que “[...] visam atingir a solução dos problemas do usuário. [...] visa atender as necessidades dos usuários, materializando estes conceitos em forma de funções no novo sistema” (DESENVOLVEDOR, 2023).

Já os Requisitos Não Funcionais são “[...] premissas essenciais para a execuções das funções definidas pelos requisitos funcionais.” (DESENVOLVEDOR, 2023).

Para uma melhor compreensão do sistema e quem irá interagir com ele, foi criado um Diagrama de Caso de Uso, conforme os usuários principais listados e o que o sistema necessita fazer.

**Figura 1 – Diagrama de Caso de Uso dos Requisitos do Sistema**

****

Fonte: Próprio Autor

Tabela 1 – Lista de Requisitos

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Descrição** | **Tipo** | **Prioridade** |
| RF001 | Administração: Cadastrar Pacientes | Funcional | Alta |
| RF002 | Administração: Cadastrar Profissionais de Saúde (Médico, Enfermeiros, Farmacêuticos) | Funcional | Alta |
| RF003 | Administração: Cadastrar Leitos | Funcional | Alta |
| RF004 | Administração: Cadastrar Suprimentos | Funcional | Alta |
| RF005 | Administração: Visualizar Pacientes | Funcional | Alta |
| RF006 | Administração: Visualizar Profissionais de Saúde cadastrados | Funcional | Alta |
| RF007 | Administração: Atender solicitações de internação e escolher leito | Funcional | Alta |
| RF008 | Médico: Agenda - Cadastrar datas e horas para consultas | Funcional | Alta |
| RF009 | Médico: Agenda – Remover horários de consultas | Funcional | Alta |
| RF010 | Médico: Agenda – Visualizar Agenda Completa | Funcional | Alta |
| RF011 | Médico: Prontuário – Atualizar Prontuário do Paciente | Funcional | Alta |
| RF012 | Médico: Consultas – Realizar consultas médicas | Funcional | Alta |
| RF013 | Médico: Consultas – Emitir Receitas Digitais | Funcional | Média |
| RF014 | Médico: Anamnese – Realizar Anamnese do Paciente | Funcional | Média |
| RF015 | Médico: Anamnese – Alterar Anamnese do Paciente a cada consulta | Funcional | Média |
| RF016 | Médico: Internação – Solicitar Internação do Paciente | Funcional | Alta |
| RF017 | Médico: Internação – Criar Prescrições durante a internação do Paciente | Funcional | Alta |
| RF018 | Médico: Internação – Realizar Alta | Funcional | Alta |
| RF019 | Enfermeiro: Internação – Solicitar Suprimentos para realizar as Prescrições | Funcional | Alta |
| RF020 | Enfermeiro: Internação – Realizar o procedimento solicitado | Funcional | Alta |
| RF021 | Enfermeiro: Internação – Atualizar o Prontuário do Paciente após a realização do procedimento | Funcional | Alta |
| RF022 | Farmacêutico: Suprimentos – Receber solicitação dos suprimentos | Funcional | Alta |
| RF023 | Farmacêutico: Suprimento – Registrar entrega de Suprimentos ao Enfermeiro | Funcional | Alta |
| RF024 | Farmacêutico: Suprimento – Dar baixa no estoque de Suprimentos | Funcional | Alta |
| RF025 | Paciente: Marcar Consultas com um Médico específico | Funcional | Alta |
| RF026 | Paciente: Visualizar próximas consultas marcadas | Funcional | Alta |
| RF027 | Paciente: Receber Notificação de Consultas Próximas | Funcional | Alta |
| RF028 | Paciente: Visualizar Prontuário Médico | Funcional | Alta |
| RF029 | Teleconsulta: Realizar Videochamadas seguras | Funcional | Alta |
| RF030 | Teleconsulta: Atualizar Prontuário do Paciente | Funcional | Alta |
| RF031 | Teleconsulta: Realizar prescrições online | Funcional | Alta |
| RNF001 | Controle de Acesso de Perfis | Não Funcional | Alta |
| RNF002 | Criptografia de Dados Sensíveis | Não Funcional | Alta |
| RNF003 | Registros de logs para auditoria | Não Funcional | Alta |
| RNF004 | Suportar várias unidades hospitalares | Não Funcional | Alta |
| RNF005 | Tempo de resposta rápido | Não Funcional | Médio |
| RNF006 | Interface amigável e responsiva | Não Funcional | Médio |

Fonte: Elaborado pelo autor com base no estudo de caso e diagrama de caso de uso

## MODELAGEM E ARQUITETURA

A modelagem e arquitetura escolhida para esse projeto foi desenvolver um sistema em Java com Maven e com JDBC (Java Database Conectivity), utilizando juntamente o SparkleJava, que é um microframework web para construção de API Rest ou páginas web leves e o banco de dados MySQL, trazendo em sua junção um software de forma a separar responsabilidades, facilitar manutenção, reutilização de partes do código e escalabilidade.

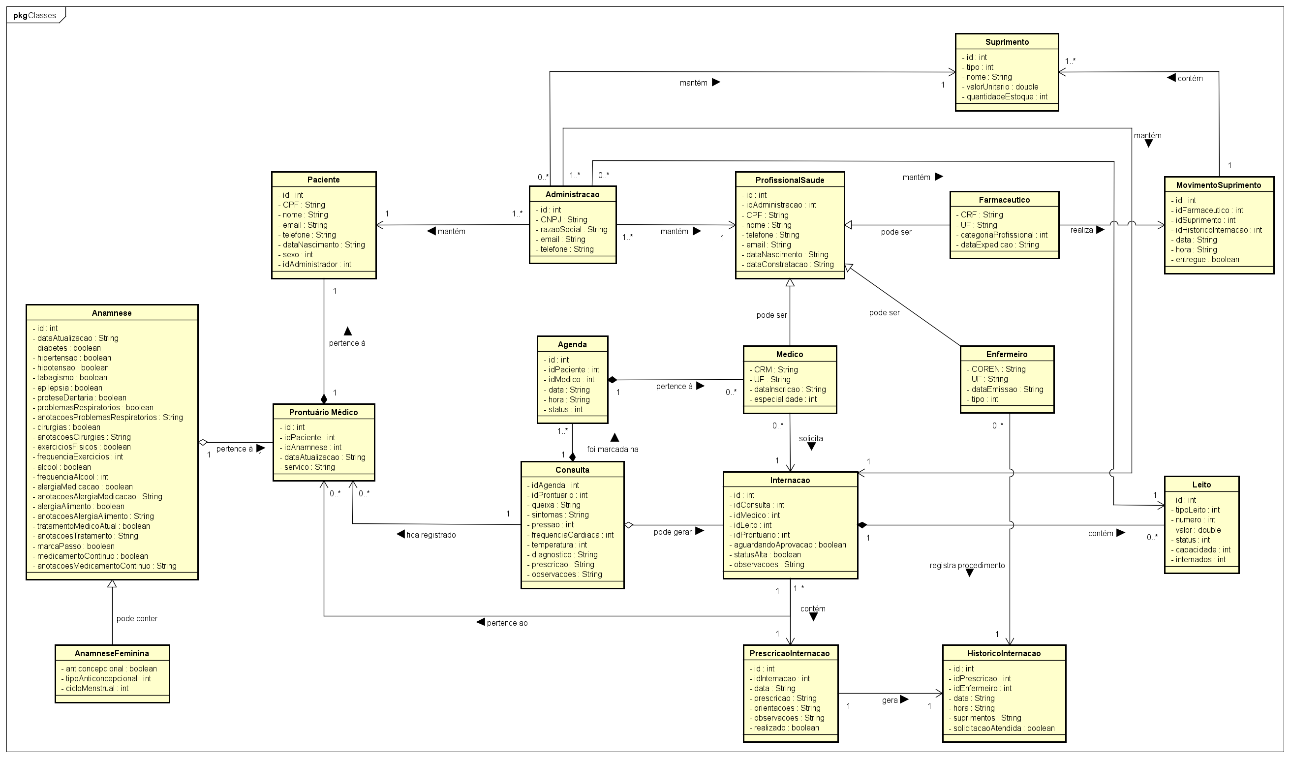
### Arquitetura do Sistema

O sistema foi desenvolvido em 5 camadas, que são compostas por:

* **Camada de Apresentação**: titulada no sistema como interfaces, onde ficam as interfaces gráficas responsáveis por interagir com os usuários, feitos com o JFrame Swing.
* **Camada de Controle**: titulada no sistema como controles, eles controlam o fluxo de interação do sistema feito pelas interfaces com o banco de dados, recebendo as informações digitadas pelo usuário e devolve uma resposta.
* **Camada de Classes**: titulada no sistema como classes, ali contém toda a base de informações que serão recebidas e enviadas tanto pelas views como pelo banco de dados.
* **Camada de Enums**:titulada no sistema como enums, ela contém classes Enums para facilitar o processo de algumas informações que são mais específicas no sistema e não mudam toda hora.
* **Camada de Persistência**:titulada no sistema como dao, ela contém as classes que utiliza o JDBC para conectar com o banco de dados. Contém métodos de cada classe para manipular dados como inserir, atualizar, remover, listar, buscas em geral.

### Diagrama de Classes

**Figura 2 – Diagrama de Classes de acordo com pesquisas e estudo de caso**



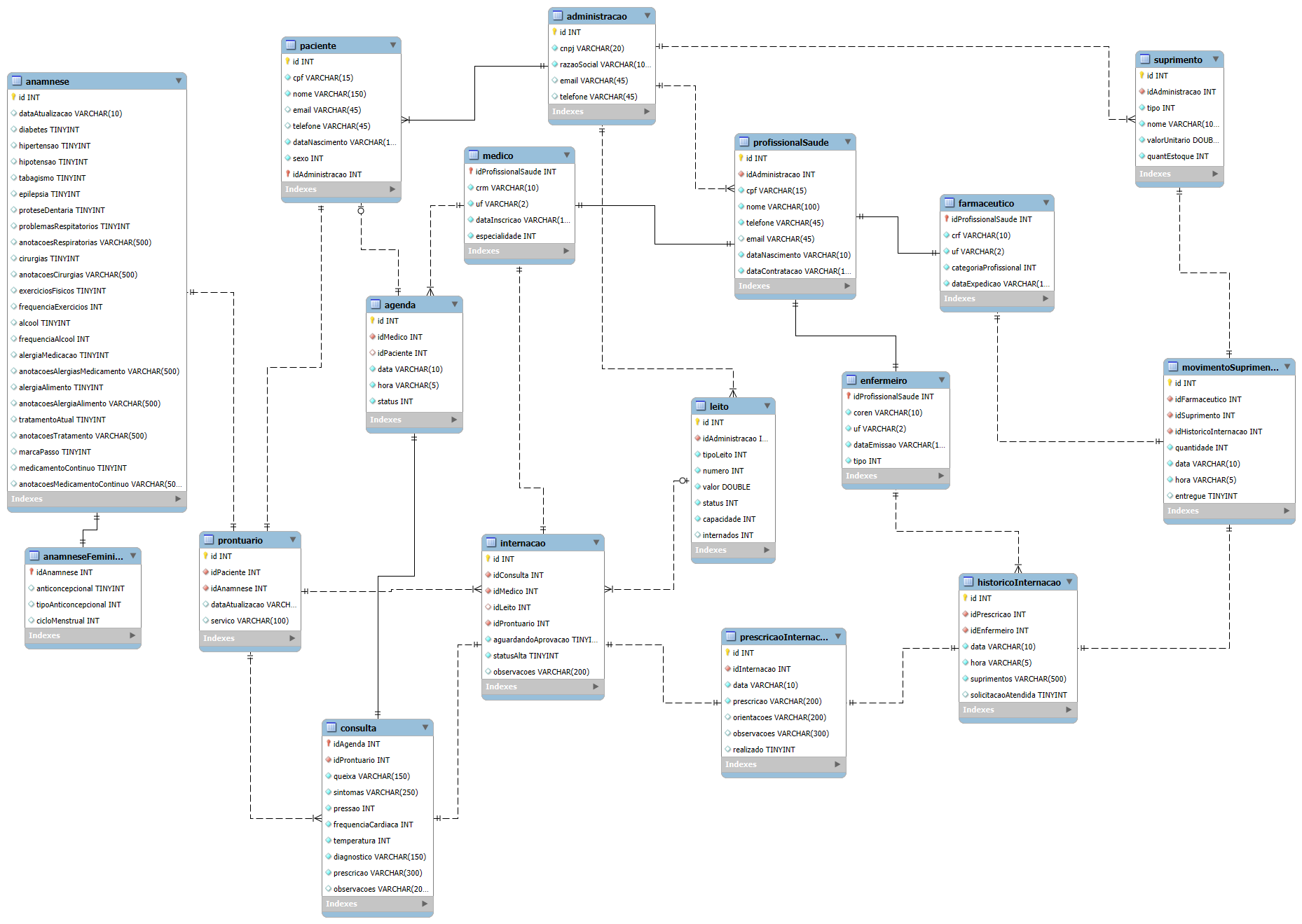
Fonte: Próprio Autor

O Diagrama de Classes acima foi desenvolvido de acordo com pesquisas sobre o temas e ligados aos requisitos dos sistemas que foram retirados do estudo de caso. Esse tipo de diagrama é muito útil para estágio de design do sistema, e ajudar a entender melhor os requisitos e componentes para identificar problemas. No âmbito de desenvolvimento de sistemas, as classes se tornam classes e objetos reais durante os estágios iniciais do projeto, podendo ser modificado conforme o passar do tempo, dependendo da lógica, interfaces e testes (IBM, 2021).

Aqui é apresentado as interações das classes entre elas, como o Administrador gerenciando os Paciente, Leitos, Suprimentos e Profissionais de Saúde; o Médico cuidando das internações dos Pacientes e realizando consultas através da sua própria agenda e as enfermeiras e farmacêuticas podendo monitorar e cuidar dos procedimentos e estoques necessários.

### Diagrama Entidade-Relacionamento (DER)

**Figura 3 – Diagrama DER do modelo relacional do Banco de Dados**



Fonte: Próprio Autor

O Diagrama Entidade-Relacionamento trabalha como uma ferramenta de modelagem que é utilizada para planejar a estrutura de um Banco de Dados antes de sua implementação. Ele mostra quais são as informações que serão armazenadas em cada tabela e como elas interagem entre si, facilitando a visualização e o planejamento antes da implementação, ajudando a evitar erros antes da criação do banco de dados.

Nesse DER, vemos que algumas tabelas influenciam as outras através das chaves, como o Profissional de Saúde que passa sua chave primária como chave para Médicos, Farmacêuticos e Enfermeiros. Aqui, é possível ver que o Administrador tem gerenciamento de todo os processos e a possibilidade de que, quando houver novas filiais, poder separar quem está ligado a cada administração.

### Endpoints da API

A API contém os seguintes endpoints principais:

Tabela 2 – Endpoints da API

|  |  |  |
| --- | --- | --- |
| **Método** | **Rota** | **Descrição** |
| POST | /administrador | Cadastra Administrador |
| GET | /administrador | Retorna Administrador |
| POST | /paciente/novo | Cadastrar Paciente |
| GET | /paciente/:idAdm | Retorna Pacientes da Unidade |
| GET | /paciente/busca/:cpf | Retorna Paciente com o CPF específico |
| POST | /profissional | Cadastrar Profissional de Saúde |
| GET | /profissional/:idAdm | Busca Profissionais de Saúde da Unidade |
| POST | /medico | Cadastra Médico |
| GET | /medico | Retorna todos os Médicos |
| GET | /medico/busca/:crm | Retorna o Médico do CRM específico |
| POST | /enfermeiro | Cadastra Enfermeiro |
| GET | /enfermeiro | Retorna todos os Enfermeiros |
| GET | /enfermeiro/busca/:coren | Retorna o Enfermeiro do COREN específico |
| POST | /farmaceutico | Cadastra Farmacêutico |
| GET | /farmaceutico | Retorna todos os Farmacêuticos |
| GET | /farmaceutico/busca/:crf | Busca Farmacêutico pelo CRF |
| POST | /leito | Cadastra Leito na Unidade |
| GET | /leito/:idAdm | Retorna Leitos da Unidade |
| GET | /leito/busca/:num | Retorna o Leito de acordo com o número |
| GET | /leito/disponíveis/:idAdm | Retorna Leitos Disponíveis |
| POST | /suprimento | Cadastra Suprimento na Unidade |
| GET | /suprimento | Recupera Suprimentos da Unidade |
| GET | /suprimento/busca/:nome | Recupera o Suprimento de acordo com o Nome |

Fonte: Próprio Autor

### Tecnologias de Persistência

Para a persistência dos dados, foi utilizado o **MySQL** que é um banco de dados relacional gratuito e open source, tem alta performance, suporta o padrão de SQL, integra facilmente com outras plataformas, além da compatibilidade com a linguagem escolhida para esse projeto que foi Java.

Tabela 3 – Tecnologias de Persistência

|  |  |
| --- | --- |
| **Camada** | **Tecnologia** |
| Backend/API REST | Java + Spark Java + Gson + JDBC |
| Banco de Dados | MySQL |
| Frontend Desktop | Java Swing (JFrame) |

Fonte: Próprio Autor

## IMPLEMENTAÇÃO

Aqui será apresentado a implementação do protótipo funcional parcial do sistema. O objetivo é demonstrar como o backend foi desenvolvido e estruturado e como as requisições foram tratadas. Por questão de tempo, foi optado por desenvolver uma versão mínima viável com um frontend simples para melhor entendimento e exemplificação.

### Tecnologias Utilizadas

As tecnologias utilizadas para o desenvolvimento desse protótipo foram:

* Linguagem: Java
* Framework web: Spark Java
* Banco de Dados: MySQL
* Biblioteca JSON: Gson
* Gerenciador de Dependências: Maven
* Ferramenta de Testes: Postman
* Frontend: Java Swing (JFrame)

### Estrutura do Sistema

A estrutura de pastas e do sistema foi feita da seguinte forma:

*/src*

*+-- classes/*

*¦ +-- Administracao.java*

*¦ +-- Agenda.java*

*¦ +-- Anamnese.java*

*¦ +-- AnamneseFeminina.java*

*¦ +-- Consulta.java*

*¦ +-- Enfermeiro.java*

*¦ +-- Farmaceutico.java*

*¦ +-- HistoricoInternacao.java*

*¦ +-- Internacao.java*

*¦ +-- Leito.java*

*¦ +-- Medico.java*

*¦ +-- MovimentoSuprimento.java*

*¦ +-- Paciente.java*

*¦ +-- PrescricaoInternacao.java*

*¦ +-- ProfissisonalSaude.java*

*¦ +-- ProntuarioMedico.java*

*¦ +-- Suprimento.java*

*+--com/*

*¦ +-- Main.java*

*+--controles/*

*¦ +-- ControleAdministracao.java*

*¦ +-- ControleAgenda.java*

*¦ +-- ControleConsulta.java*

*¦ +-- ControleEnfermeiro.java*

*¦ +-- ControleFarmaceutico.java*

*¦ +-- ControleHistoricoInternacao.java*

*¦ +-- ControleInternacao.java*

*¦ +-- ControleLeito.java*

*¦ +-- ControleMedico.java*

*¦ +-- ControleMovimentoSuprimento.java*

*¦ +-- ControlePaciente.java*

*¦ +-- ControlePrescricao.java*

*¦ +-- ControleProfissional.java*

*¦ +-- ControleProntuario.java*

*¦ +-- ControleSuprimento.java*

*+--dao/*

*¦ +-- AdministracaoDAO.java*

*¦ +-- AgendaDAO.java*

*¦ +-- AnamneseDAO.java*

*¦ +-- ConsultaDAO.java*

*¦ +-- DB.java*

*¦ +-- EnfermeiroDAO.java*

*¦ +-- FarmaceuticoDAO.java*

*¦ +-- HistoricoInternacaoDAO.java*

*¦ +-- InternacaoDAO.java*

*¦ +-- LeitoDAO.java*

*¦ +-- MedicoDAO.java*

*¦ +-- MovimentoSuprimentoDAO.java*

*¦ +-- PacienteDAO.java*

*¦ +-- PrescricaoInternacaoDAO.java*

*¦ +-- ProfissionalSaudeDAO.java*

*¦ +-- ProntuarioDAO.java*

*¦ +-- SuprimentoDAO.java*

*+--enums/*

*¦ +-- CategoriaProfissional.java*

*¦ +-- CicloMenstrual.java*

*¦ +-- DisponibilidadeLeito.java*

*¦ +-- Especialidades.java*

*¦ +-- IntervaloConsultas.java*

*¦ +-- ServicoConsulta.java*

*¦ +-- ServicoProntuario.java*

*¦ +-- Sexo.java*

*¦ +-- TipoAnticoncepcional.java*

*¦ +-- TipoEnfermeiro.java*

*¦ +-- TipoLeito.java*

*¦ +-- TipoSuprimento.java*

*+--interfaces/*

*¦ +-- TelaAgendaCompleta.java*

*¦ +-- TelaAnamnesePaciente.java*

*¦ +-- TelaCNPJ.java*

*¦ +-- TelaCPFPaciente.java*

*¦ +-- TelaCPFProfissional.java*

*¦ +-- TelaCadastrarEnfermeiro.java*

*¦ +-- TelaCadastrarFarmaceutico.java*

*¦ +-- TelaCadastrarMedico.java*

*¦ +-- TelaCadastrarPaciente.java*

*¦ +-- TelaCadastrarDatasConsultas.java*

*¦ +-- TelaCadastrarLeito.java*

*¦ +-- TelaCadastrarSuprimento.java*

*¦ +-- TelaConsulta.java*

*¦ +-- TelaEnfermeirosCadastrados.java*

*¦ +-- TelaFarmaceuticosCadastrados.java*

*¦ +-- TelaInicial.java*

*¦ +-- TelaInicialAdministrador.java*

*¦ +-- TelaInicialEnfermeiro.java*

*¦ +-- TelaInicialFarmaceutico.java*

*¦ +-- TelaInicialMedico.java*

*¦ +-- TelaInicialPaciente.java*

*¦ +-- TelaInternacoesAtuais.java*

*¦ +-- TelaLeitosCadastrados.java*

*¦ +-- TelaMarcarConsulta.java*

*¦ +-- TelaMedicosCadastrados.java*

*¦ +-- TelaPacientesCadastrados.java*

*¦ +-- TelaPrescricaoInternacao.java*

*¦ +-- TelaPrimeiroAcessoADM.java*

*¦ +-- TelaSolicitacoesInternacoes.java*

*¦ +-- TelaSolicitarSuprimentos.java*

*¦ +-- TelaSuprimentosCadastrados.java*

*¦ +-- TelaVerificarInternacoes.java*

Essa estrutura foi definida e separada dessa forma para que cada parte fique em sua pasta e seja mais fácil de localizar o que desejar sem tanto trabalho.

### Telas da Prototipação

O projeto foi criado juntamente com a parte visual simples para facilitar o entendimento das conexões que são realizadas durante o processo. Então, separei algumas temas que achei interessante pra demonstrar um pouco do processo.

Para a compilação do projeto, é necessário digitar no prompt de comando na pasta dentro do projeto: mvn clean compile exec:java

**Figura 4 – Tela Inicial do Sistema**



Fonte: Próprio Autor

Essa será a Primeira tela apresentada caso já haja algum Administrador cadastrado no sistema. Caso o Banco de Dados esteja limpo, irá ser uma tela para realizar esse cadastro.

O protótipo que está sendo apresentado está considerando como somente um único Administrador, mas com planejamento no banco de dados e no sistema a possibilidade de trabalhar com filiais.

Tanto o Cadastro de Leitos e Suprimentos seguem o mesmo padrão de tela, que é apresentado somente para o Administrador do sistema, pois é de sua responsabilidade os cuidados desse gerenciamento:

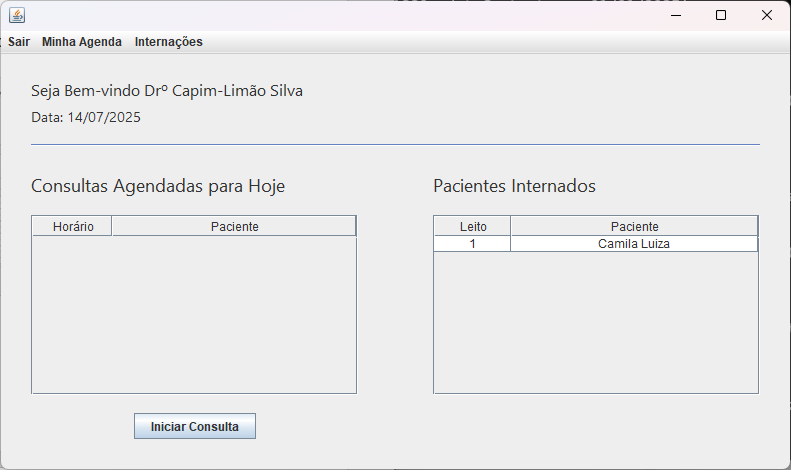
**Figura 5 – Telas de Cadastro de Leitos e Suprimentos**

|  |  |
| --- | --- |
|  |  |

Fonte: Próprio Autor

Os médicos cadastrados podem visualizar diretamente da sua tela Inicial suas próximas consultas além dos pacientes internados.

**Figura 6 – Tela Inicial do Médico**

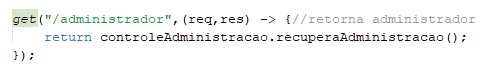


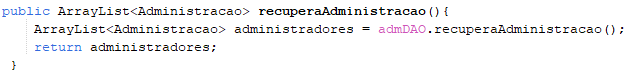
Fonte: Próprio Autor

AAAA

### Trechos de Códigos

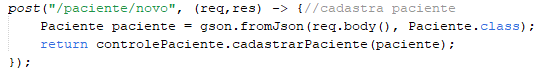
#### Recuperação dos Administradores Cadastrados

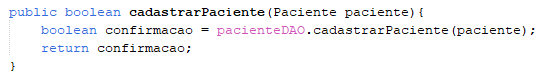






#### Cadastro de Novos Pacientes

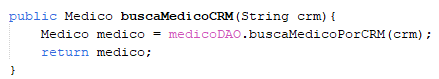


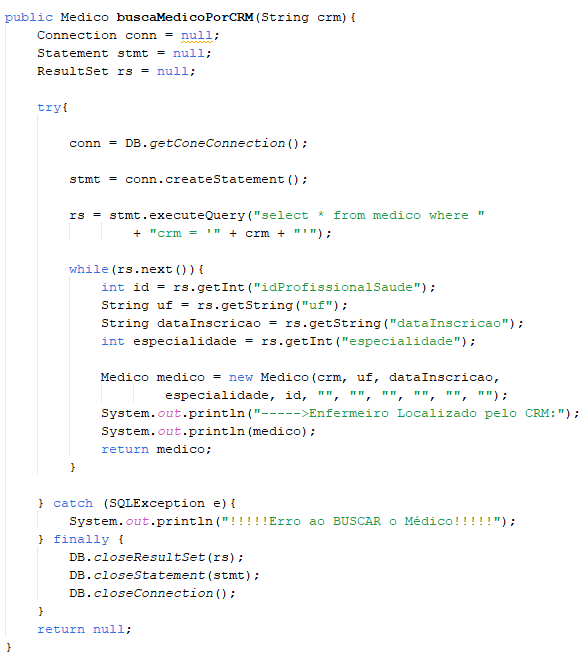




#### Busca Médico via CRM

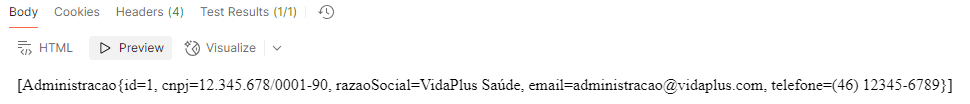




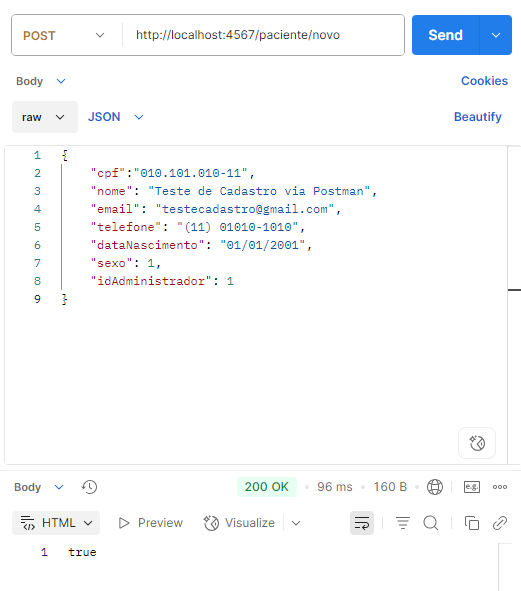


### Exemplo de Fluxo

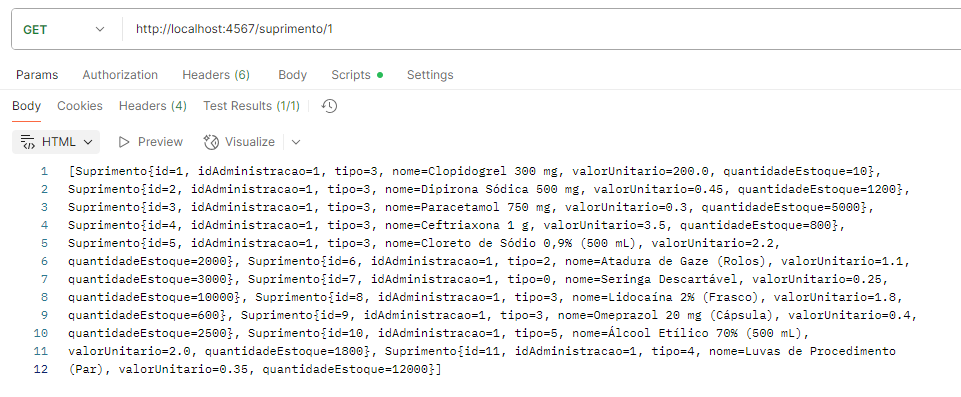
O endpoint GET /administrador retorna todos os administradores cadastrados no sistema. Resposta no Postman:



Já o endpoint POST /paciente/novo cadastra um novo paciente e retorna a confirmação como forma de boolean.



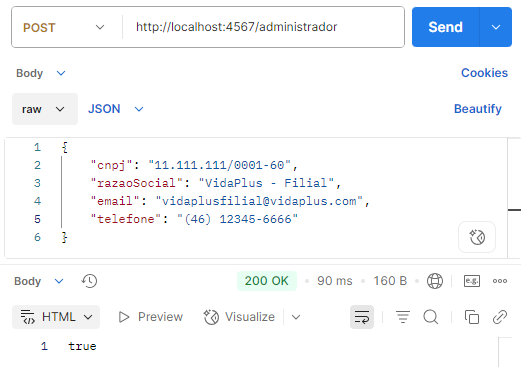
O endpoint GET /suprimento/:idAdm mostra todos os suprimentos cadastrados em uma unidade:



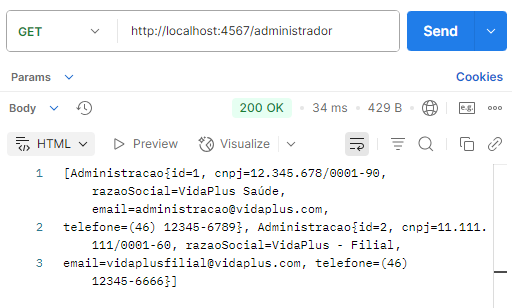
## PLANO DE TESTES

Esse plano de teste teve como objetivo mostrar algumas funcionalidades e validar as principais delas via API REST do sistema desenvolvido em com Spark Framework e banco de dados MySQL. Neles foram testados métodos como GET e POST, de forma manual diretamente com o Postman.

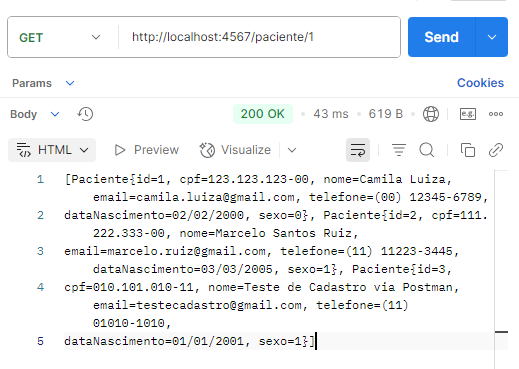
### POST /administrador



### GET /administrador



### GET /paciente/:idAdm



## CONCLUSÃO

Foi um projeto bem complexo pra mim, principalmente por nunca ter utilizado ferramentas como Postman ou framewors como o Spark, mas fico feliz por ter conseguido atender ao mínimo do que foi proposto, com os conhecimentos que eu já tinha juntamente com novos desafios.

O prazo do projeto foi um pouco apertado pra mim pra que eu conseguisse fazer etapas como de verificações de outros itens, mas ao mesmo tempo, é gratificante ver que o trabalho aconteceu realmente.

Deixo aqui meus agradecimentos e o link do GitHub com o projeto: [Amandickz/VidaPlus: Projeto final do curso de TADS da UNINTER.](https://github.com/Amandickz/VidaPlus)

## REFERÊNCIAS

DESENVOLVIMENTO, C. do. **Requisitos funcionais e não funcionais: o que são e como identificar?**. Casa do Desenvolvedor. Disponível em: https://blog.casadodesenvolvedor.com.br/requisitos-funcionais-e-nao-funcionais/. Acesso em:15 jun. 2025.

Lima, L. A. O. *et al*. **Gestão Hospitalar E Os Desafios Na Saúde Pública No Brasil**. Disponível em: https://www.iosrjournals.org/iosr-jbm/papers/Vol26-issue3/Ser-6/G2603063338.pdf. Acesso em: 13 jul. 2025.

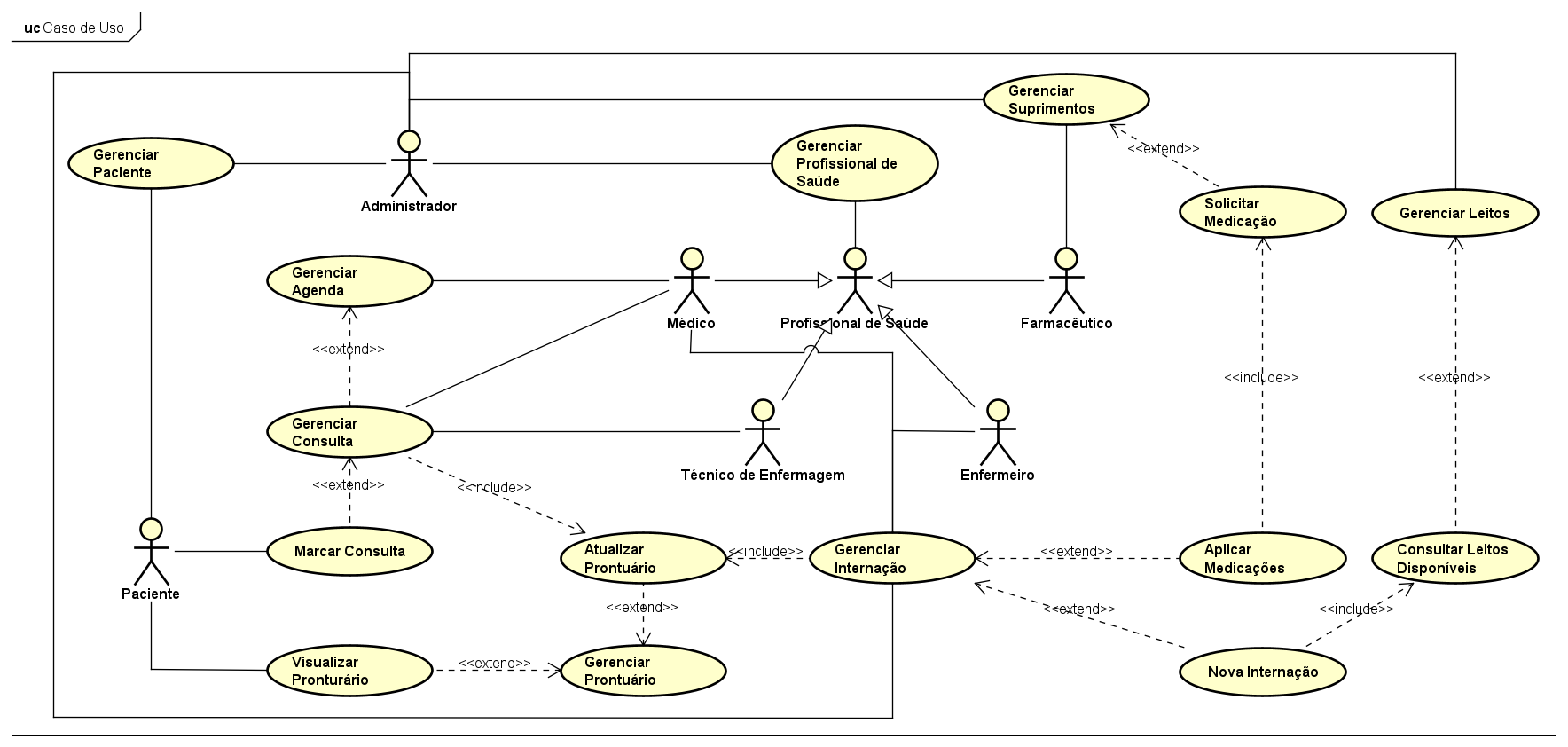
Riviere, R. **Como funciona a gestão hospitalar e qual a sua importância?**. Disponível em: https://versatilis.com.br/como-funciona-a-gestao-hospitalar/. Acesso em: 13 jul. 2025.

**DIAGRAMAS de Classes. [S. l.],** 05 mar 2021. Disponível em: https://www.ibm.com/docs/pt-br/rsas/7.5.0?topic=structure-class-diagrams. Acesso em: 14 jul. 2025.

MEMED**, Hierarquia na área da saúde: como funciona na prática?**. Disponível em: https://blog.memed.com.br/hierarquia-na-area-da-saude/. Acesso em 18/04/2025

## ANEXO A – CASO DE USO

**MODELO DE CASO DE USO COMPLETO DO SISTEMA DE GESTÃO HOSPITALAR E DE SERVIÇOS DE SAÚDE (SGHSS)**

****

## ANEXO B – FICHA DE ANAMNESE

**EXEMPLO DE FICHA DE ANAMNESE PARA APLICAÇÃO ANTES DA CONSULTA**



## ANEXO C – PRONTUÁRIO DE TRIAGEM

**MODELO DE PRONTUÁRIO DE TRIAGEM**



## ANEXO D – MODELO DE CARTEIRA DE ENFERMEIRO

**MODELO DE CARTEIRA DE ENFERMEIRO EMITIDA PELO CONSELHO REGIONAL DE ENFERMAGEM DO PARANÁ (COREN/PR)**



## ANEXO E – MODELO CARTEIRA DE FARMACÊUTICO

**MODELO DA CÉDULA DE IDENTIFICAÇÃO DE FARMACÊUTICO EMITIDA PELO CONSELHO FEDERAL DE FARMÁCIA (CRF)**



## ANEXO F – MODELO CARTEIRA DE MÉDICO

**MODELO DA CÉDULA DE IDENTIDADE MÉDICA EMITIDA PELO CONSELHO FEDERAL DE MEDICINA (CFM)**

