PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

PUC Minas Virtual

Pós-graduação *Lato Sensu* em Engenharia de *Software*

Projeto Integrado

Relatório Técnico

Azumy Tools Software de Avaliação Física

Lucas Henrique Teixeira Fonseca

Belo Horizonte

08/2022

# Projeto Integrado

Sumário

[Projeto Integrado 3](#_Toc122039129)

[1. Cronograma de Trabalho 4](#_Toc122039130)

[2. Introdução 5](#_Toc122039131)

[3. Definição Conceitual da Solução 6](#_Toc122039132)

[3.1 Diagrama de Casos de Uso 6](#_Toc122039133)

[3.2 Requisitos Funcionais 7](#_Toc122039134)

[3.3 Requisitos Não-funcionais 8](#_Toc122039135)

[4. Protótipo Navegável do Sistema 9](#_Toc122039136)

[5. Diagrama de Classes de Domínio 10](#_Toc122039137)

[6. Arquitetura da Solução 11](#_Toc122039138)

[6.1 Padrão Arquitetural 11](#_Toc122039139)

[6.2 C4 model - Diagrama de Contexto 12](#_Toc122039140)

[7. Frameworks de Trabalho 13](#_Toc122039141)

[8. Estrutura Base do Front End 14](#_Toc122039142)

[9. Modelo Relacional ou Projeto de Banco de Dados NoSQL 15](#_Toc122039143)

[10. Plano de Testes 16](#_Toc122039144)

[11. Relatório de Execução de Testes de Software 21](#_Toc122039145)

[12. Apropriação de Horas no Projeto 24](#_Toc122039146)

[13. Avaliação Retrospectiva 25](#_Toc122039147)

[14. Código da Aplicação 26](#_Toc122039148)

## Cronograma de Trabalho

|  |  |  |  |
| --- | --- | --- | --- |
| **Datas** | | **Atividade / Tarefa** | **Produto / Resultado** |
| **De** | **Até** |
| 05/08/2022 | 05/08/2022 | 1. Leitura da documentação do roteiro e escopo do tcc. | Entendimento do Projeto e proposto |
| 05/08/2022 | 06/08/2022 | 1. Identificação dos atores, requisitos mais relevantes do sistema. | Lista de atores |
| 06/08/2022 | 10/08/2022 | 1. Identificação de ferramentas para desenvolvimento dos casos de uso,   Elaboração dos requisitos funcionais e não funcionais | Diagrama de caso de uso |
| 11/08/2022 | 12/08/2022 | 1. Identificação de ferramentas para elaboração do protótipo navegável | Protótipo navegável |
| 13/08/2022 | 13/08/2022 | 1. Identificação de ferramentas para elaboração do diagrama de classes | Diagrama de Classes |
| 16/08/2022 | 25/08/2022 | 1. Escolha da arquitetura para o desenvolvimento da aplicação. | Arquitetura e ferramental |
| 26/08/2022 | 30/08/2022 | 1. Elaboração do Diagrama de contexto | Diagrama de contexto |
| 01/09/2022 | 01/09/2022 | 1. Elaborar brevemente resumo sobre elaboração da aplicação estrutura utilizada assim como frameworks | Frameworks utilizados |
| 05/09/2022 | 10/09/2022 | 1. Modelo relacional do banco de dados nosql | Modelo relacional |
| 11/09/2022 | 25/10/2022 | 1. Desenvolvimento da aplicação back-end | Backend do Sistema |
| 11/09/2022 | 25/10/2022 | 1. Desenvolvimento da aplicação front-end (HTML5, CSS3, javaScript) | Front-end do Sistema |
| 26/10/2022 | 10/11/2022 | 1. Elaboração do plano de testes, objetivos e resultados | Plano de Testes |
| 26/10/2022 | 10/11/2022 | 1. Elaborar Relatório de execução de Testes | Relatório de execução de testes |
| 11/11/2022 | 05/12/2022 | 1. Deploy da Aplicação AWS | Deploy da Aplicação |
| 15/12/2022 | 15/12/2022 | 15. Disponibilizar código fonte hospedado no git para aprovação do projeto pelos instrutores. | Visão geral do código fonte e sistema |
| 15/12/2022 | 15/12/2022 | 16. Envio do trabalho |  |

## Introdução

Com o crescente interesse e preocupação das pessoas e profissionais de diversas áreas com a saúde e desempenho físico de atletas, profissionais e amadores, o mercado carece de ferramentas para controle, análise e verificação dos dados físicos de um indivíduo. Equipes esportivas e profissionais da área da saúde vêm investindo em equipamentos e sistemas de análise para garantir melhores resultados e que tais resultados possam ser alcançados de formas mais eficazes e eficientes possíveis.

Atualmente a empresa Psyco Sport Center uma academia que fica localizada na cidade Ribeirão Preto vem sofrendo muitos problemas devido à falta de gestão prática para gerir seu negócio, sem suporte de qualquer tipo de ferramenta o cliente ainda utiliza o tradicional método de armazenar suas informações e avaliações em Excel.

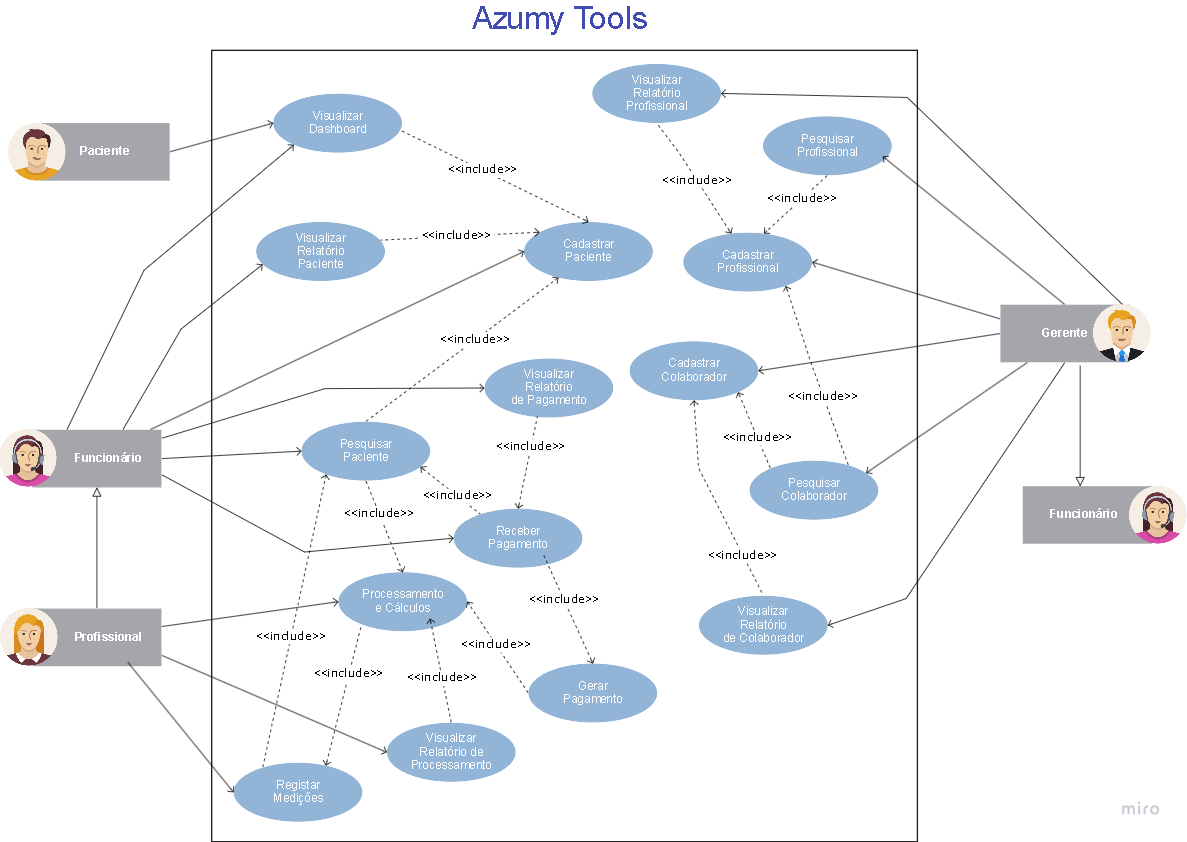
Esse projeto tem como objetivo o desenvolvimento e implementação de um software de avaliação física totalmente web para a academia Psyco Sport Center. O sistema será capaz de realizar a análise e o armazenamento de informações sobre avaliações físicas e de desempenho de pacientes e atletas de diferentes níveis que frequentam a academia e passagem por avaliações semestralmente. O software poderá ser utilizado por diversos tipos de profissionais da Saúde e do Esporte assim sendo ele médico, nutricionistas, profissionais de educação física entre outros.

A partir dos dados obtidos nos relatórios gerados, o profissional pode tomar decisões em como agir sobre o tratamento ou treinamento de seus pacientes/atletas.

A motivação para realização deste projeto é a elaboração do projeto proposto para conclusão da pós-graduação em Engenharia de Software pela Puc Minas, sendo assim estritamente acadêmico.

## Definição Conceitual da Solução

## Diagrama de Casos de Uso



## Requisitos Funcionais

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Descrição Resumida** | **Dificuldade (B/M/A)\*** | **Prioridade**  **(B/M/A)\*** |
| RF01 | Cadastrar Paciente | B | A |
| RF02 | Pesquisar Paciente | B | B |
| RF03 | Visualizar Relatório de Paciente | B | B |
| RF04 | Cadastrar Funcionário | B | A |
| RF05 | Pesquisar Funcionário | B | B |
| RF06 | Visualizar Relatório de Funcionários | B | B |
| RF07 | Visualizar relatório de processamento | B | B |
| RF08 | Baixar Pagamento | M | A |
| RF09 | Visualizar Relatório de Pagamentos | B | A |
| RF010 | Efetuar baixa do pagamento | M | A |
| RF011 | Preencher informações do Paciente | A | A |
| RF012 | Avaliação de percentual de Gorduras | A | A |
| RF013 | Cálculo do Índice de Massa Corpórea (IMC) | A | A |
| RF014 | Cadastrar Profissional | B | A |
| RF015 | Pesquisar Profissional | B | A |
| RF016 | Visualizar Relatório de Profissional | M | M |

\* B = Baixa, M = Média, A = Alta.

## Requisitos Não-funcionais

|  |  |  |
| --- | --- | --- |
| **ID** | **Descrição** | **Prioridade**  **B/M/A** |
| RNF01 | Ao Buscar uma informação única no sistema o Tempo de resposta deve ser menos que < 3s | A |
| RNF02 | Durante o processamento, cálculos e exibição de conteúdo, deve ser mostrada ao usuário alguma animação Tempo de resposta deve ser < 5 segundos. | A |
| RNF04 | O sistema deverá ser fácil de usar, auto intuitivo | A |
| RNF05 | A senha do paciente será armazenada em nossa base de dados sobre criptografia | A |
| RNF06 | O sistema será hospedado em uma plataforma segura e rápida (AWS) | A |
| RNF07 | O sistema deverá utilizar arquitetura de container (DOCKER) | A |
| RNF08 | O sistema será executado em qualquer navegador | B |
| RNF09 | O sistema deverá ser desenvolvido utilizando a linguagem Java | B |
| RNF010 | O sistema deverá se comunicar com banco de dados Postgres | A |
| RNF011 | O sistema deverá atender às normas de programação assim como padrões de projeto. | A |
|  |  |  |

**Observação: acrescente quantas linhas forem necessárias.**

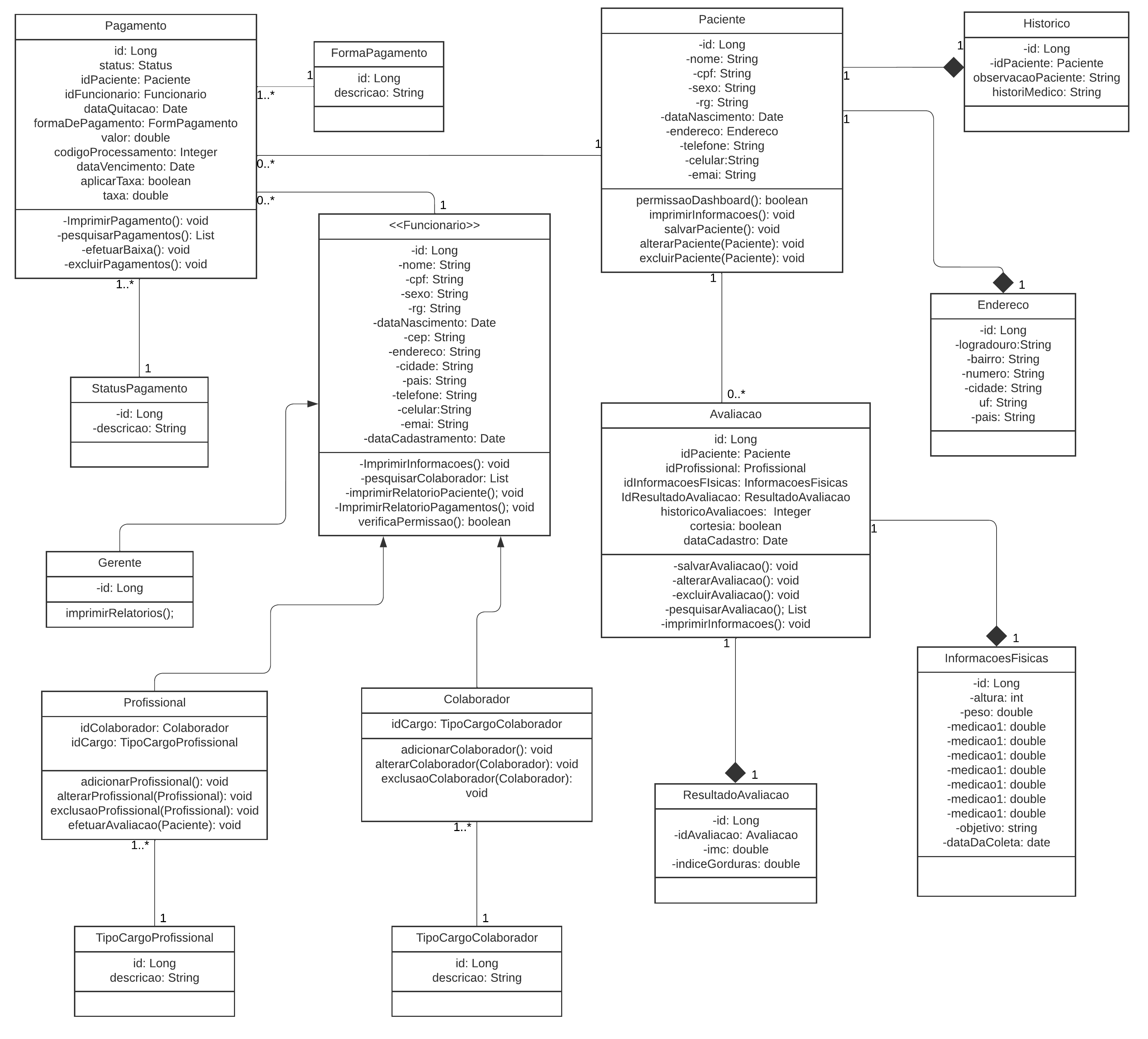
## Protótipo Navegável do Sistema

<https://youtu.be/c6F44KFfIx4> -- Vídeo protótipo navegável

<https://github.com/lucasb4rn/AzumyEngenhariaDeSoftware>

## Diagrama de Classes de Domínio

Azumy Tools



## Arquitetura da Solução

O padrão utilizado para o projeto foi MVC com SpringBoot juntamente da implementação do design pattern de services e repositories, no Frontend foi utilizado angular com do Material Design, a especificação de design para interfaces interativas do Google para desenvolvimento dos componentes.

## Padrão Arquitetural

Tecnologias Utilizadas:

* Backend
* JAVA
* POSTGRESQL
* H2 (AMBIENTE DE TESTES)
* SPRING SECURITY
* JWT
* Frontend
* TYPESCRIPT
* ANGULAR
* MATERIAL
* Infra

AWS / EC2 / DOCKER - backend

AWS / DOCKER - frotnend

## C4 model - Diagrama de Contexto

Interface gráfica do usuário, Aplicativo, Teams

Descrição gerada automaticamente

## Frameworks de Trabalho

**Backend**

SpringBoot, framework para aplicativos independes baseado em spring, super rápido para construção de aplicações restfull

**ORM**

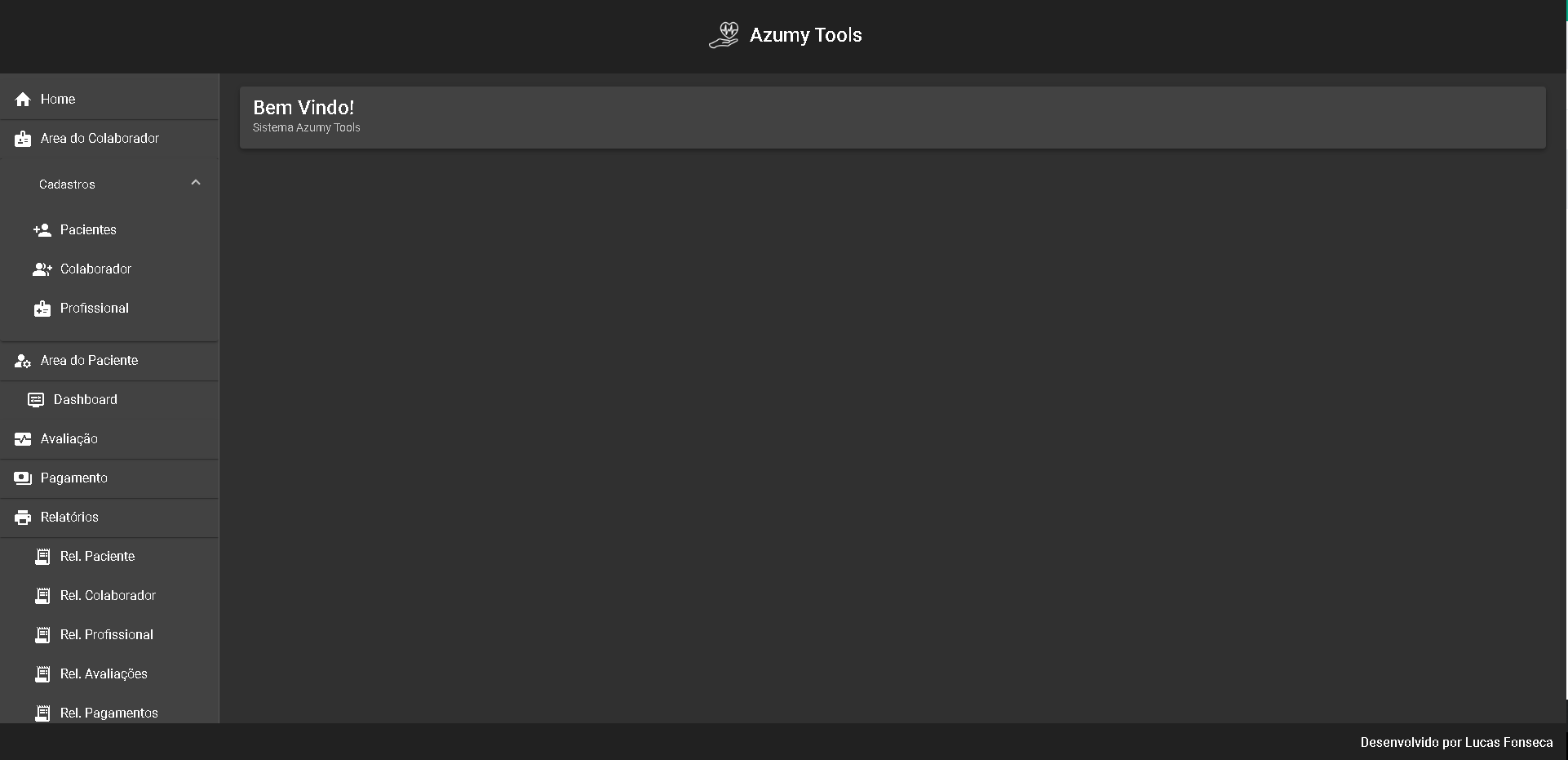
Hibernate, solução ORM java que segue as especificações JPA (Java Persistente API) para tratar mapeamento e gerenciar as persistências de dados.

**Frontend**

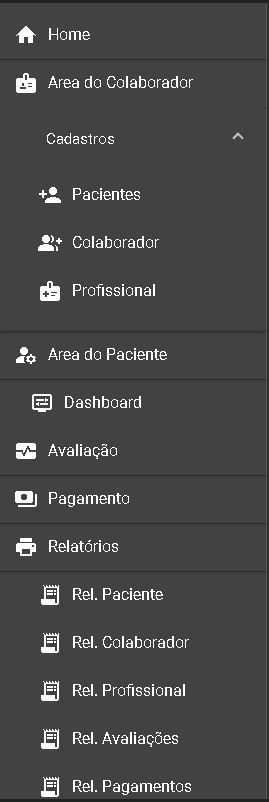
Angular que é baseado em typescript para construção dos componentes e logica de negócio / consumo da API, com Material Angular para desenvolvimento das interfaces visuais do programa

## Estrutura Base do Front End

Leiaute de exibição geral do sistema.



Menus de navegação do Sistema.



## Modelo Relacional ou Projeto de Banco de Dados NoSQL

Linha do tempo

Descrição gerada automaticamente

## Plano de Testes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Número** | **Caso de uso** | **Objetivo do caso de teste** | **Entradas** | **Resultados esperados** |
|  | Cadastrar Usuário | Validar preenchimentos dos campos | -Abrir Cadastro de usuário  -Inserir informações em todos os campos  -Clicar em salvar | “Usuário Cadastrado com Sucesso” deve ser exibido e a informação gravada no banco de dados |
|  | Validar Cadastro de Usuário | Validar falta de preenchimento de campos | -Abrir Cadastro de usuário  -Inserir informações em quase todos os campos | “A mensagem de Campo requirido deve ser exibida e o botão Salvar não deve ser liberado” |
|  | Update de Usuário cadastrado no sistema | Validar se a mudança será efetuada | -Abrir tela de usuário  -Clicar em Editar Usuário  -Alterar Informação  -Clicar em Atualizar | “A mensagem de Usuário Atualizado com sucesso será exibida e as informações devem estar alteradas e o usuário deve ser redirecionado para tela de listar usuário” |
|  | Validar falta de preenchimento de campos no update de usuário | Validar Falta de Preenchimento de Campo | -Abrir tela de usuário  -Clicar em Editar Usuário  -Alterar Informação deixando sem preenchimento  -Clicar em Atualizar | “Uma mensagem de preenchimento requerido deve ser exibida para o campo que não foi preenchido e o botão de atualizar não deve ser habilitado” |
|  | Lista Usuário | Validar a exibição dos usuários na tela | -Abrir tela de usuário | As informações de todos os usuários cadastrados devem ser exibidas em uma grade com seus respectivos campos todos preenchidos |
|  | Deletar Usuário | Deletar um usuário o removendo do banco de dados | -Abrir tela de usuário  -Clicar em Deletar Usuário  -Na tela de deleção clicar em deletar | A mensagem de Usuário deletado com sucesso deve ser exibido e o registro removido do banco de dados. |
|  | Cancelar deleção de Usuário | Cancelar a deleção do usuário | -Abrir tela de usuário  -Clicar em Deletar Usuário  -Na tela de deleção clicar em cancelar | Será redirecionado a tela de lista de usuário e nada deve acontecer. |
|  | Efetuar Cadastro de Paciente | Salvar paciente no banco de dados | Abrir tela de cadastro de Paciente  -Preencher todas as informações  -Clicar em Salvar | Registro será salvo no banco de dados e será exibida uma mensagem de cadastro efetuado com sucesso junto com redirecionamento da página para tela de listar Pacientes |
|  | Validar cadastro de Paciente de Paciente | Validar Campo CPF | -Abrir tela de cadastro de Paciente  -Preencher todas as informações e colocar um CPF inválido  -Clicar em Salvar | Será exibida uma mensagem de erro informando que o formato do CPF está incorreto e o cadastro não será efetuado |
|  | Update de Paciente | Efetuar alteração de paciente no banco de dados | -Abrir a tela de listar pacientes  -Clicar em Editar paciente  -Alterar informações  -Salvar Registro | Será exibida uma informação de cadastro efetuado com sucesso, registro será salvo no banco de dados e a página será redirecionado para listar paciente, juntamente com a sua exibição na grade com as informações alteradas. |
|  | Validar Preenchimento dos Campos ao alterar Paciente | Validar os campos da tela de update /edit paciente | -Abrir a tela de listar pacientes  -Clicar em Editar paciente  -Alterar informações  -Deixar campos sem preencher  -Salvar Registro | Enquanto houver campos sem preencher o botão atualizar não deve ser liberado. |
|  | Listar Pacientes | Visualizar todos os pacientes em uma lista ao entrar na tela | -No menu  -Clicar em cadastro de Pacientes | Deverá ser exibido em uma lista todos os pacientes |
|  | Deletar paciente | Remover paciente do banco de dados e informar Usuário sucesso da deleção | -Entra na tela de paciente  -Selecionar 1 paciente e clicar em deletar  -Será redirecionado para tela de deleção  -Usuário clica em deletar | Paciente é removido do banco de dados e uma mensagem é exibida para o usuário informando que o registro foi deletado com sucesso. |
|  | Cadastrar um profissional | Efetua um cadastro de profissional | -Entra na tela de cadastro de profissional  -Preenche as informações do profissional  -Clica em salvar | Registro é salvo no banco de dados e uma mensagem é exibida para o usuário que a operação foi um sucesso. |
|  | Valida Campos do Cadastro de profissional | Não deixar incluir profissionais com qualquer campo vazio | -Entra na tela de cadastro de profissional  -Preenche as informações e deixa apenas 1 campo sem preencher  -Tenta salvar | O campo que não está preenchido deverá ficar com uma mensagem de erro e o botão salvar deve ficar todo o tempo desabilitado. |
|  | Atualizar Profissional | Muda as informações do profissional com sucesso e informa o usuário | -Entra na tela de Profissional  -Seleciona um profissional e clica em atualizar/edit  -Altera uma informação  -Clica em salvar | O registro deve ser alterado com sucesso informando o usuário com uma mensagem e retornando a tela de listar profissional |
|  | Valida Campos na tela de atualizar | Dar erro ao deixar qualquer campo da tela de update sem preencher e impedir de atualizar o registro | -Entra na tela de Profissional  -Seleciona um profissional e clica em atualizar/edit  -Deixar um campo vazio  -Clica em salvar | O campo ficará com uma mensagem de erro e o botão de atualizar bloqueado todo o tempo. |
|  | Cadastrar uma Avaliação | Efetuar cadastramento de uma avaliação | -Entrar na tela de cadastro de avaliação  -Preencher as informações  -Clicar em salvar | O registro deverá ser incluído no banco de dados e uma mensagem de sucesso deve ser exibida. |
|  | Validar Campos do cadastro de Avaliação | Verificar se ao não preencher os campos será exibida uma mensagem de erro e bloquear o botão de Salvar | -Entrar na tela de cadastro de avaliação  -Tentar incluir Registro com alguns campos vazios | Deve ser exibido no campo uma mensagem informação que o mesmo é requerido e o botão de salvar deve ficar todo o tempo bloqueado. |
|  |  |  |  |  |

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

## Relatório de Execução de Testes de Software

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Número** | **Caso de teste** | **Saída esperada** | **Resultados encontrados** | **Aprovado?** |
|  | Cadastrar Usuário | Usuário inserido no banco de dados e mensagem para o usuário “Cadastrado com sucesso!” | Usuário inserido no banco de dados e mensagem para o usuário “Cadastrado com sucesso!” | Sim |
|  | Validar Cadastro de Usuário | Usuário não inserido no sistema e mensagem de erro de campos não preenchido para o usuário. | Usuário não foi inserido e a mensagem foi exibida corretamente. | Sim |
|  | Update de Usuário cadastrado no sistema | Usuário deve ser alterado no banco de dados e mensagem de registro atualizado com sucesso será exibida para o Usuário. | Usuário foi alterado e a mensagem foi exibida corretamente. | Sim |
|  | Validar falta de preenchimento de campos no update de usuário | Usuário não deve ser alterado e mensagem de erro de campo requerido deve ser exibido para todos os campos da tela. | Mensagem foi exibida e o usuário não foi alterado. | Sim |
|  | Deletar Usuário | Usuário deve ser removido do sistema e mensagem de “Usuário removido com sucesso” | Mensagem foi exibida e o usuário foi removido do banco de dados. | Sim |
|  | Cancelar deleção de Usuário | Usuário será redirecionado para tela de listar Usuário | Usuário foi redirecionado e nada aconteceu. | Sim |
|  | Lista Usuário | Exibir todos os usuários do sistema em uma lista ao clicar em Cadastro de Usuário | Todos os usuários foram exibidos com suas respectivas informações | Sim |
|  | Efetuar cadastro de Paciente | Paciente deve ser salvo no banco de dados e uma mensagem de sucesso deve ser exibida, juntamente com o redirecionamento da tela | Paciente foi salvo no banco de dados, mensagem de sucesso mais redirecionamento foram exibidos | Sim |
|  | Validar campos do cadastro de Paciente | Mensagem de campo requirido deve ser exibida e o botão salvar deve ficar desabilitado durante todo o processo em que ainda faltar campos a serem preenchidos | Mensagem apareceu no campo e o botão salvar ficou desabilitado durante todo o processo | Sim |
|  | Update de Paciente | Paciente deve ser alterado com sucesso e mensagem de sucesso deve ser exibida | Paciente foi alterado com sucesso e a mensagem foi exibida corretamente | Sim |
|  | Validar Preenchimento dos Campos ao alterar Paciente | Mensagem de erro nos campos que não estão preenchidos e botão de Atualizar deve ficar desabilitado | Mensagem apareceu nos campos e o botão de atualizar ficou desabilitado durante todo o processo | Sim |
|  | Listar Pacientes | Exibir a lista com todos os pacientes | Todos os pacientes foram exibidos | Sim |
|  | Deletar Paciente | Apagar com sucesso um paciente e informar ao usuário que o mesmo foi removido com sucesso | Paciente foi removido e a mensagem foi exibida. | Sim |
|  | Cadastrar um Profissional | Salvar profissional no banco de dados e informar que a operação foi um sucesso. | Profissional foi cadastrado com sucesso e uma mensagem foi exibida. | Sim |
|  | Valida Campos do Cadastro de profissional | Verifica se os campos podem ficar vazios | Os campos ficaram com uma mensagem de erro informando que precisam ser preenchidos e o botão de Salvar desabilitado | Sim |
|  | Atualizar Profissional | Atualiza as informações do profissional selecionado | As informações foram atualizadas com sucesso | Sim |
|  | Valida Campos na tela de atualizar | Verifica se os campos podem ficar vazios | Os campos vazios exibiram uma mensagem de erro e o botão atualizar todo o tempo desabilitado. |  |
|  | Cadastrar uma Avaliação | Efetuar cadastro de Avaliação | O cadastro foi incluído na base de dados. | Sim |
|  | Validar Campos do cadastro de Avaliação | Verificar se está dando erro ao não preencher campos | Erro foi exibido em todos os casos e o botão Adicionar se manteve todo o tempo bloqueado | Sim |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Apropriação de Horas no Projeto

|  |  |  |
| --- | --- | --- |
| **Histórico de apropriação de horas** | | |
| **Data do registro** | **Atividade** | **Quantidade de horas** |
| 05/08/2022 | Leitura e entendimento da documentação e roteiro | 2h |
| 05/05/2022 | Levantamento de informações e requisitos | 4h |
| 06/08/2022 | Diagrama de caso de uso | 7h |
| 16/08/2022 | Protótipo Navegável | 15h |
| 13/08/2022 | Elaboração do Diagrama de Classes | 5h |
| 16/08/2022 | Escolha da arquitetura | 8h |
| 26/08/2022 | Elaboração do Diagrama de Contexto | 2h |
| 05/09/2022 | Modelo relacional do banco de dados | 10h |
| 11/09/2022 | Desenvolvimento do backend | 50h |
| 11/09/2022 | Desenvolvimento do frontend | 60h |
| 26/10/2022 | Plano de Testes | 6h |
| 26/10/2022 | Relatório de execução de testes | 4h |
| 11/11/2022 | Deploy da Aplicação | 4h |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Avaliação Retrospectiva

Iniciamos o projeto com o objeto de desenvolver uma aplicação viável para a academia Psyco Sport Center com intuito de auxiliar os educadores na administração de alguns processos.

Iniciamos desenvolvendo a parte teórica juntamente com os diagramas para auxiliar no entendimento do projeto.

Fizemos o levantamento dos requisitos funcionais e não funcionais do sistema, que ajudou no entendimento da complexidade.

Criamos o protótipo que nos possibilitou vislumbrar já na etapa inicial como seria o rosto da nossa aplicação e o que realmente era necessário ter no sistema.

A partir desse momento criamos o diagrama de classe para obter o entendimento mais técnico dos relacionamentos e processos do sistema.

Escolhemos as tecnologias que seriam empregadas no desenvolvimento do backend e do front end.

Definimos o banco de dados como postgres e iniciamos o desenvolvimento da aplicação. Com alguns módulos da aplicação já concluídos, elaboramos o plano e o relatório de testes em sincronia.

Dados todos os objetos, conseguimos concluir com êxito sua totalidade.

## Código da Aplicação

<https://github.com/lucasb4rn/AzumyEngenhariaDeSoftware> -- Projeto técnico do Curso

<https://github.com/lucasb4rn/azummy> -- Código fonte do backend

<https://github.com/lucasb4rn/azummy-front> -- Código fonte do frontend

<http://3.18.176.164:4200> – PÁGINA DO FRONT END -> LOGIN maria senha: maria123