

NTC HEALTH

Ana Júlia Almeida Silva Neves - RM98974

Cauã Couto Basques – RM97755

Kaique Agostinho de Oliveira - RM550815

Leonardo Buonanno Del Matto – RM98315

Nicolý Oliveira Santos - RM552410

SUMÁRIO

Descrição do Projeto:

O projeto NTC Health consiste na criação de um sistema com uma agenda de acompanhamento para pacientes, onde o médico responsável poderá cadastrar as necessidades do mesmo, de acordo com os medicamentos necessários. A agenda também determinará o intervalo de horas entre os remédios e estipulará a data de início e fim de suas medicações. Assim que o paciente realizar o cadastro de sua agenda, o médico responsável será capaz de iniciar seu acompanhamento com base em seu ID e armazenar todos os dados em um banco de dados.

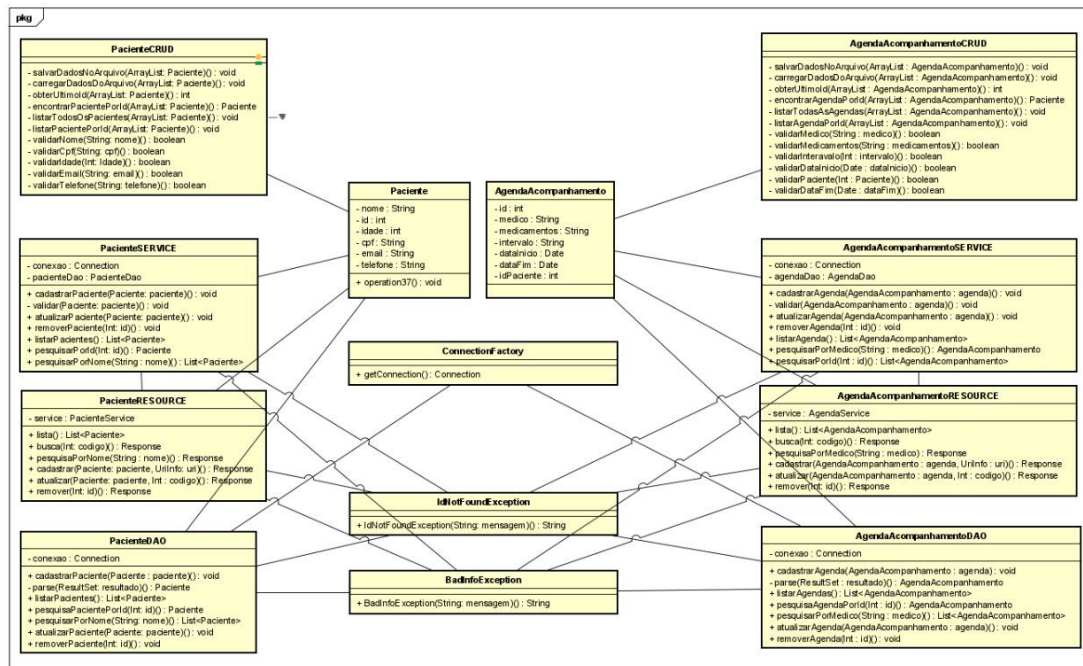
Justificativa do Projeto:

O projeto visa facilitar o acompanhamento dos pacientes pelos seus médicos, além da acessibilidade para os dados da agenda, sendo possível ser acessada tanto pelo médico quanto pelo paciente, dessa forma, melhoraremos a eficiência do processo de medicações pensando em sua organização e acessibilidade.

Objetivos do Projeto:

O objetivo do projeto é melhorar a organização diante da relação entre paciente e médico, garantindo um acompanhamento sólido por meio da agenda digital de medicação. Visando aumentar a acessibilidade a dados para o paciente e o médico.

Diagrama de classes do projeto:



Pasta Model:

AgendaAcompanhamento.java, Paciente.java: Estas classes são responsáveis por representar informações básicas sobre a agenda de acompanhamento e o paciente. Elas armazenam os dados essenciais relacionados a essas entidades e servem como estruturas de dados fundamentais para o sistema.

Pasta Crud:

AgendaAcompanhamentoCrud.java, PacienteCrud.java: Estas classes fornecem as operações básicas de criação, leitura,

atualização e exclusão de agendas e pacientes. Sendo utilizada para gerar os dados em um arquivo.txt que será enviado ao banco de dados para armazenar as informações inseridas.

Pasta Dao:

AgendaDao.java, Paciente.java: Estas classes são responsáveis pelos métodos a serem utilizados para realizar a inserção de dados no banco de dados.

Pasta Service:

AgendaService.java, PacienteService.java: Estas classes são responsáveis pelas validações que serão lançadas em cima dos dados inseridos para API, utilizando dos métodos das suas respectivas classes DAO e sendo requisitado nas classes Resource.

Pasta Resource:

AgendaResource.java, PacienteResource.java: Estas classes são responsáveis por definirem os recursos disponíveis na API, como suas operações.

Pasta Factory:

ConnectionFactory.java: Esta classe é responsável por buscar o banco de dados a ser utilizado e realizar a conexão com o mesmo.

Pasta Exception:

IdNotFoundException.java, BadInfoException.java: Estas classes representam exceções a serem lançadas e tratadas de acordo com as necessidades do sistema.

Descrição dos Endpoints da API Restful:

Endpoint	Verbo	HTTP Descrição	Códigos de Status
/api/agenda/pesquisa	GET	Pesquisar agendas por nome	200 OK, 500 Internal Server Error
/api/agenda	GET	Listar todas as agendas	200 OK
/api/agenda/{id}	GET	Obter uma agenda pelo ID	200 OK, 404 Not Found
/api/agenda/pesquisaPorMedico/{nomeMedico}	GET	Pesquisar agendas por nome de médico	200 OK, 500 Internal Server Error
/api/agenda	POST	Cadastrar uma nova agenda	201 Created, 400 Bad Request
/api/agenda/{id}	PUT	Atualizar uma agenda pelo ID	200 OK, 400 Bad Request, 404 Not Found
/api/agenda/{id}	DELETE	Excluir uma agenda pelo ID	204 No Content, 404 Not Found
/api/paciente	GET	Listar todos os pacientes	200 OK
/api/paciente/{id}	GET	Obter um paciente pelo ID	200 OK, 404 Not Found
/api/paciente/pesquisaPorNome	GET	Pesquisar pacientes por nome	200 OK, 500 Internal Server Error
/api/paciente	POST	Cadastrar um novo paciente	201 Created, 400 Bad Request
/api/paciente/{id}	PUT	Atualizar um paciente pelo ID	200 OK, 400 Bad Request, 404 Not Found
/api/paciente/{id}	DELETE	Excluir um paciente pelo ID	204 No Content, 404 Not Found

Diagrama Entidade-Relacionamento do Projeto:

Modelo Lógico:

