Siloam

Lenin Cristi Fernandes, RA 021030114

7 de Agosto de 2019

1. Introdução

O objetivo do projeto é criar um sistema de guarda de registros médicos que permita utilizar os dados dos exames em algoritmos de aprendizado de máquina sem ferir a privacidade do paciente. O projeto quando disparado inicia uma API REST no endereço http://localhost:8080/ com uma série de métodos de consulta e inserção, os métodos de consulta são pessoas, medicos, atendimentos, exames e upsidedown. Os métodos criaratendimento, fazertriagem e fazerconsulta são de inserção.

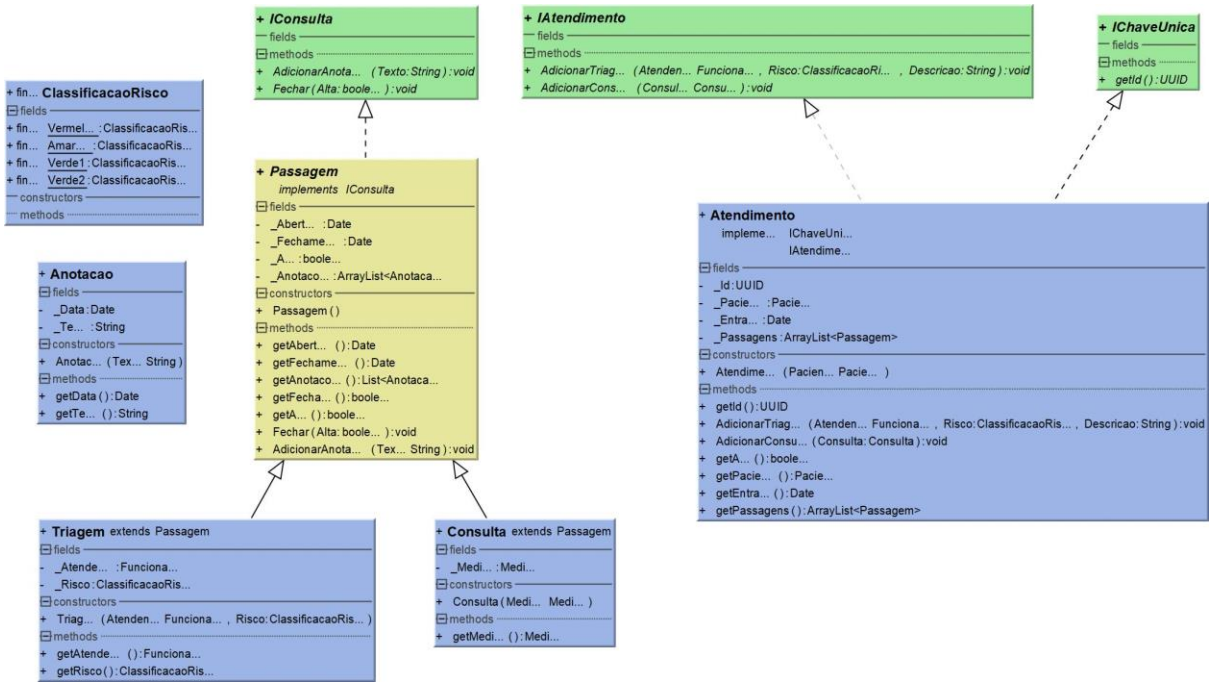
2. Descrição das classes

As classes estão dentro do grupo “healthcare” e são divididas nas seções

Consulta

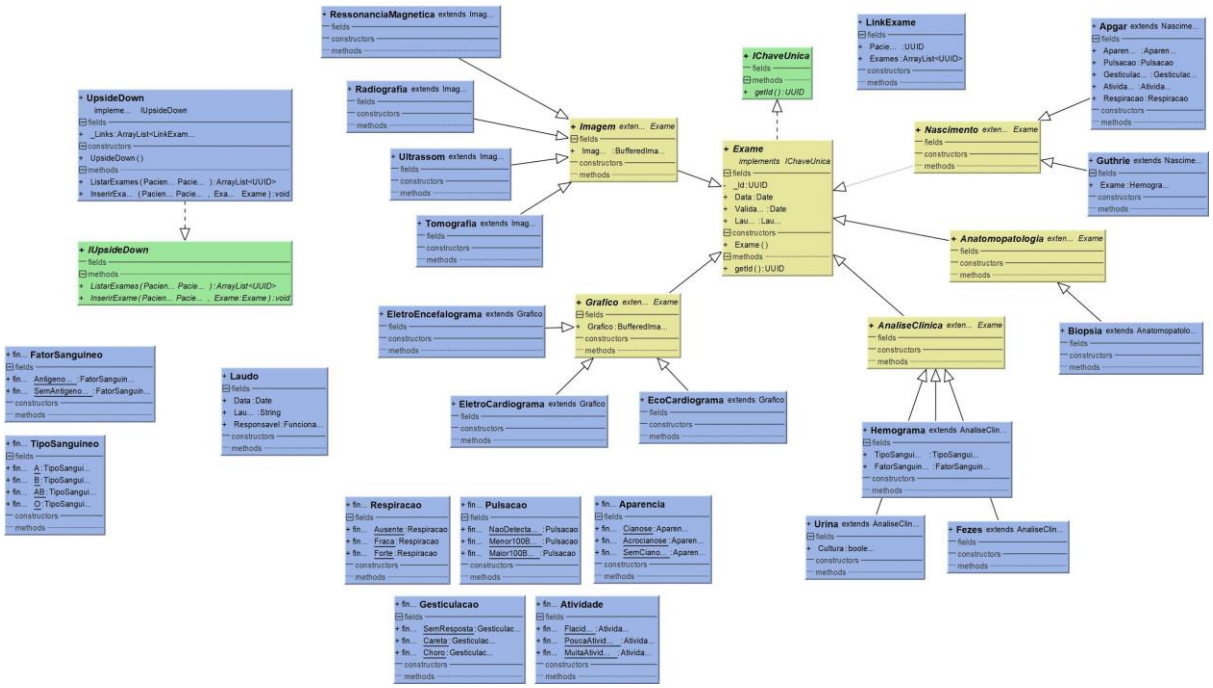
Consulta e Triagem herdam de Passagem e representam os passos possíveis dentro do Atendimento, que representa a jornada completa de um atendimento a um Paciente. Atendimento implementa IChaveUnica para ter um ID único. IConsulta e IAtendimento contém definições de métodos, Anotacoes é uma classe de apoio representa uma entrada de anotação para Triagem ou Consulta e ClassificacaoRisco é um enum com o risco atribuído na Triagem.

As classes são seletivas, por exemplo a classe Triagem pode ter um Funcionário (Medico ou Enfermeiro) como atendente, a classe Consulta, somente um Medico, sendo que as duas herdam de Pessoa.



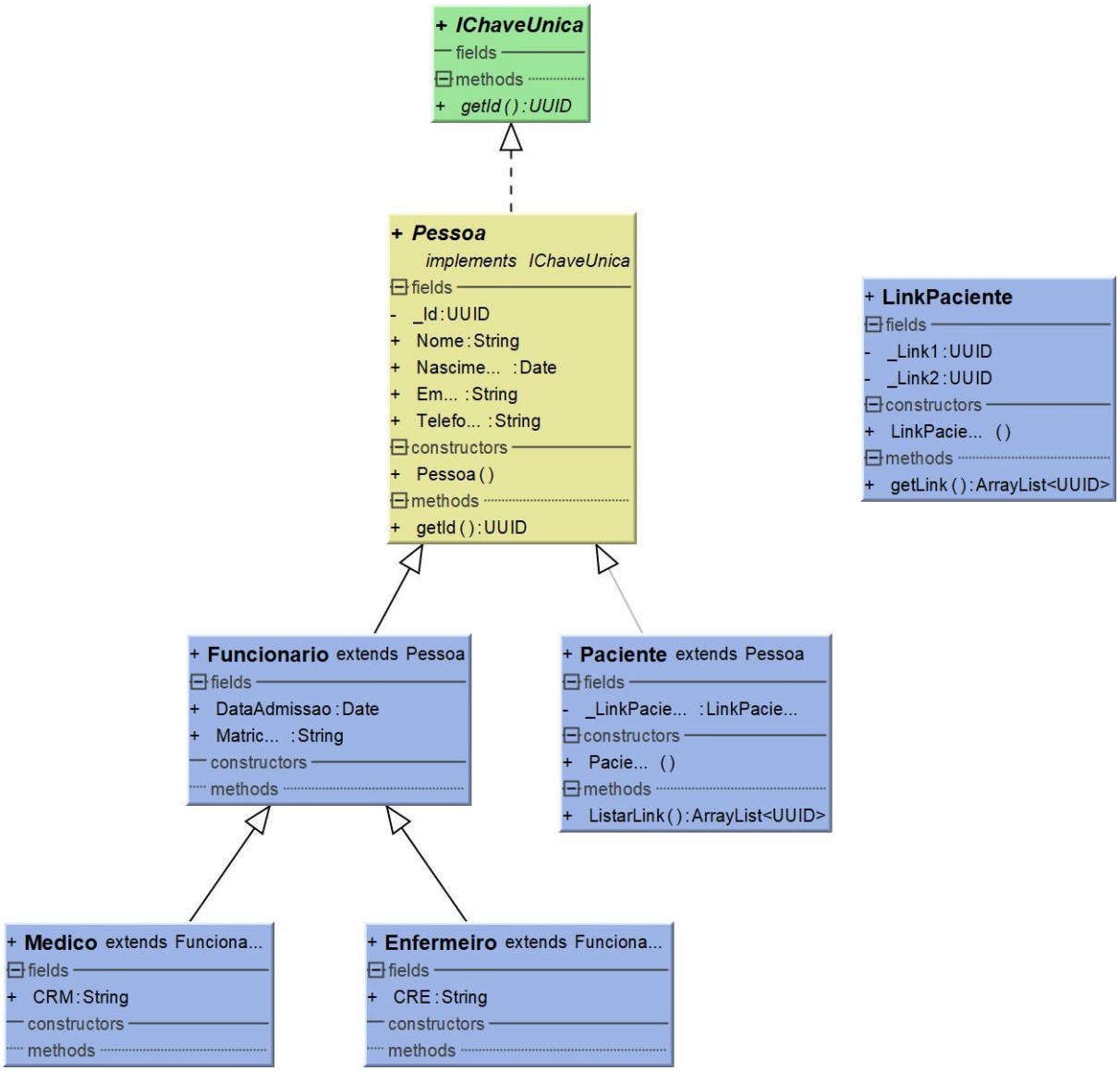
Exames

As classes de exames Herdam todas direta ou indiretamente de Exame, que implementa a interface IChaveUnica. Ao todo são 13 classes de exames, divididos em 5 categorias representadas por super-classes com 7 enums de apoio. A classe Laudo que é usada em Exame representando o laudo médico daquele exame.



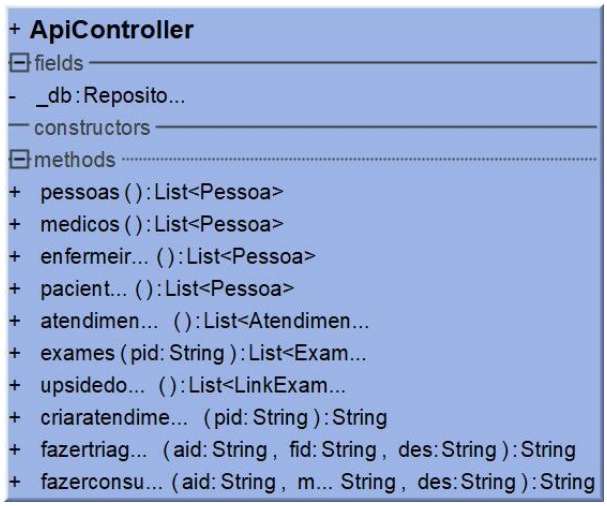
Pessoas

As classes Enfermeiro e Medico herdam de Funcionario que por sua vez herda de Pessoa. A classe Paciente herda diretamente de Pessoa. A classe Pessoa implementa IChaveUnica.



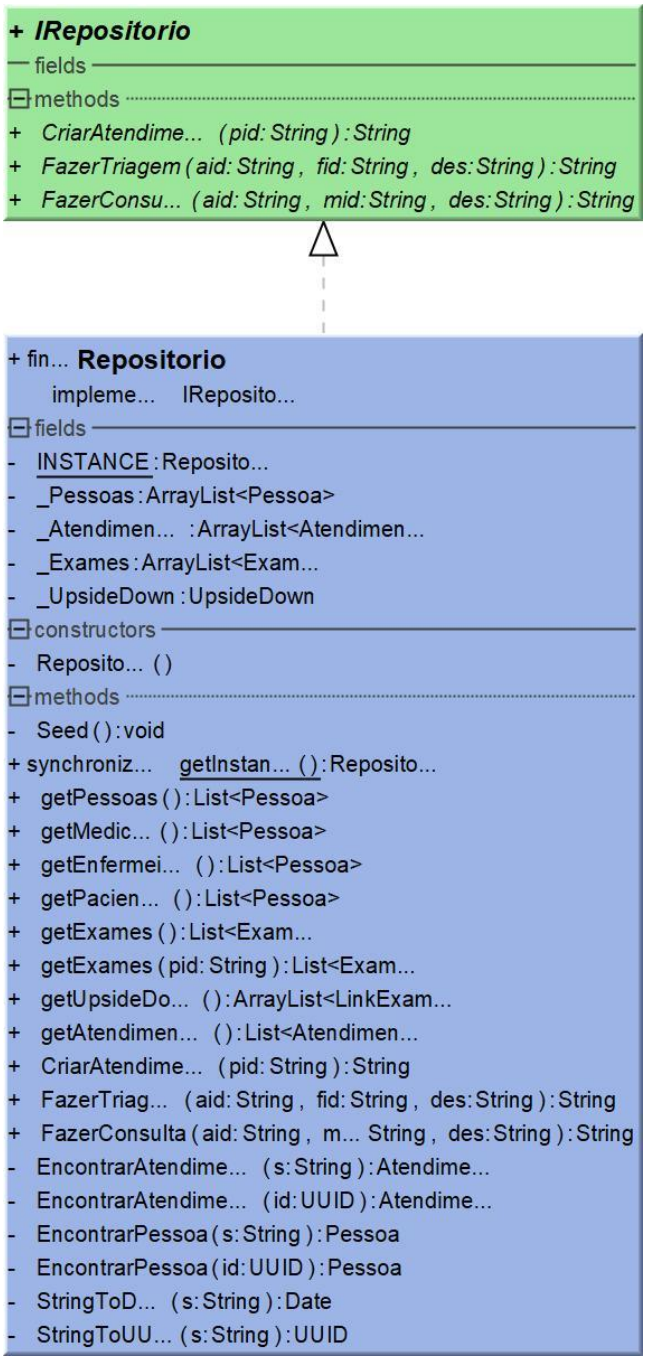
Controller

Esta seção contém a classe que gera a Api REST, é um conjunto de métodos de consulta ao Repositorio



Repositorio

O repositório contém basicamente duas classes: Repositorio e IRepositorio. A classe IRepositorio contem os metodos gerais do Repositorio que por sua vez é uma classe singleton, essa classe serve como um “banco de dados” na memória somente e uma vez instanciada, dispara um método que popula dois arrays internos, um de pessoas e um de exames. Esses arrays são usados nos métodos de abertura de atendimentos, triagem, consultas e pesquisa de exames.



Classe Especiais

As classes LinkPaciente, LinkExame e UpsideDown tem o papel de encapsular a lógica que permite atingir o objetivo do projeto: Ter um conjunto de Exames ligados ao Paciente mas ter um rastreio forte de seu acesso e não permitir que em posse do Exame se identifique o Paciente, preservando sua anonimidade para algoritmos rodando nos mesmos.

Ela faz esse processo com um conjunto de duas chaves GUID (LinkPaciente), que são rotacionadas (pela classe Paciente) cada vez que se lista os exames de um paciente específico pelo método de consulta numa classe separada que guarda essa estrutura que cruza Paciente e Exame (UpsideDown), os métodos são fortemente tipados e sem visibilidade.
Uma nota importante é que a propriedade _Links de UpsideDown não é pública na ideia da regra de negócio (adotei por convenção usar "_" nas propriedades privadas), ela está pública aqui somente para ser possível ver no browser a rotação de chaves a cada acesso a lista de Exames de um paciente específico.

3. Conceitos de orientação a objetos aplicados

O projeto tem seus pontos de POO mais importantes:

- No próprio mecanismo de recuperação do par de chaves, pois os campos internos são encapsulados e o próprio método de acesso rotaciona a chave, tornando impossível acessar os registros de um determinado paciente sem rotacionar a chave, ou seja, sem rastro a cada acesso. Isso dá um registro forte de privacidade e uso do dado médico.
- Nas listas usadas no Repositorio, todas são de super-classes que abrigam especiações, ou seja todas são polimorficas, quando um método de retorno de Medicos por exemplo é disparado, ele filtra a lista de Pessoas fazendo um cast de tipo para Medicos. A lista de Atendimentos abriga tanto Triagens quanto Consultas.
- Algumas classes são seletivas na sua construção, com,o citado o exemplo da Triagem aceitar Funcionarios e Consulta somente Medicos, Atendimento somente Pacientes. Algumas classes contam com métodos de fechamento que atualizam propriedades internas como a Consulta por exemplo.
- A chave unica implementada nas classes que se comportam como tabelas raiz é uma interface.
- O Repositorio como citado é uma singleton e mantem o estado em toda a execução do aplicativo.