

Implementação de um sistema integrado de gestão de candidaturas espontâneas para uma clínica médica

Linguagens Orientadas por Objetos

Docente:

Dora Melo

2º Ano da Licenciatura em Informática de Gestão

Janeiro de 2019

Autores:

Joana Figueiredo nº.16530

Thayná Pereira nº.16537

ÍNDICE

ÍNDICE DE ILUSTRAÇÕES	3
Introdução	4
Diagrama de Casos de Uso	5
Diagrama de Classes	7
Classe Administrador:.....	8
Classe Candidato:	9
Classe Candidatura:.....	10
Classe Especialidade:	11
Classe Clínica Médica:	12
Conclusão	13

ÍNDICE DE ILUSTRAÇÕES

Ilustração 1 - Exemplo de diagrama de casos de uso	5
Ilustração 2 - Exemplo de diagrama de classes	7

Introdução

Este relatório tem como finalidade descrever de forma detalhada o desenvolvimento de uma aplicação que implemente um serviço de gestão de submissões de candidaturas espontâneas para a contratação de médicos, numa clínica médica, de tipo persistente e parametrizável, utilizando a linguagem Java que foi instruída durante o decorrente semestre na cadeira Linguagem Orientada por Objetos.

A linguagem Java é uma poderosa linguagem de programação orientada por objetos disponível para todos os entusiastas e público em geral que a pretende programar.

O nosso relatório tem como objetivo esclarecer de forma mais clara o desenvolvimento do nosso código e a sua respetiva interpretação. Na realização do código estamos conscientes da sua semântica e das suas consequências.

Temos como objetivo explicitar e apresentar as diversas soluções que detetámos para

A construção do sistema de integração de candidatos através as respetivas candidaturas numa clínica médica. Vamos ainda definir e explicar os diversos fundamentos que achamos que são indispensáveis para as possibilidades de solução evidenciada: realização do **diagrama de casos de uso** que irá apresentar os atores e os casos de uso que achámos necessários serem implementados, realização do **diagrama de classes** que se trata em estruturar o sistema proposto e apresentar as suas classes, atributos, operações e as relações entre os objetos.

Diagrama de Casos de Uso

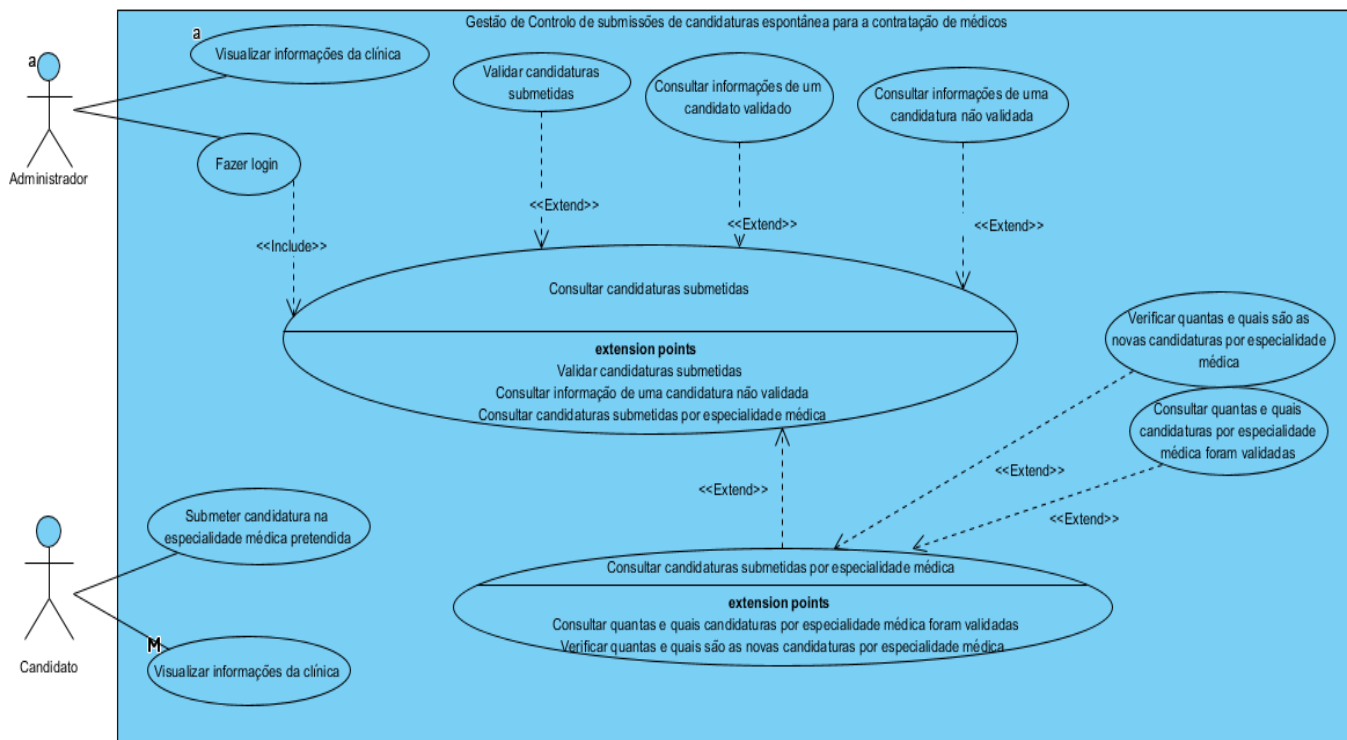


Ilustração 1 - Exemplo de diagrama de casos de uso

O diagrama de casos de uso representa graficamente os atores, os casos de uso e a relação entre estes elementos. Tem como objetivo contribuir na descrição do cenário que tem como finalidade apresentar os elementos de um sistema e as suas funcionalidades para possíveis iterações com o utilizador.

Para a possível realização do diagrama de casos de usos apresentado, utilizamos a ferramenta *Visual Paradigm*. O nosso diagrama está caracterizado pelos atores e os seus respetivos casos de uso. Os atores são denominados de Administrador e Candidato, pelo o que são esses os elementos que interagem com o sistema posto em causa.

O ator Administrador é o único dos atores que executa todas as funcionalidades do sistema, o Candidato apenas consegue interagir com o sistema para a consulta de informações da clínica e para o envio da sua candidatura.

O Administrador tem uma principal funcionalidade a qual é **fazer login** no sistema, esta é a que permite ao Administrador executar todas as outras funcionalidades.

A relação existente entre o caso de uso **fazer login** e **consultar candidaturas submetidas** é uma relação de existência obrigatória. Encontramos uma relação de *<<include>>* entre esses casos de uso porque o

Administrador tem que obrigatoriamente fazer login no sistema para que o mesmo possa consultar as candidaturas submetidas.

O caso de uso **consultar candidaturas submetidas** tem uma relação de existência opcional entres os casos de uso **validar candidaturas submetidas**, **consultar informações de um candidato validado**, **consultar informações de uma candidatura não validada** e **consultar candidaturas submetidas por especialidade médica**. É uma relação de <<extend>> porque são funcionalidades que o Administrador só irá executá-las caso seja necessário. O mesmo acontece com a relação entre o caso de uso **consultar candidaturas submetidas por especialidade médica** e os casos de uso **verificar quantas e quais são as novas candidaturas por especialidade médica** e **consultar quantas e quais candidaturas por especialidade médica foram validadas**.

Diagrama de Classes

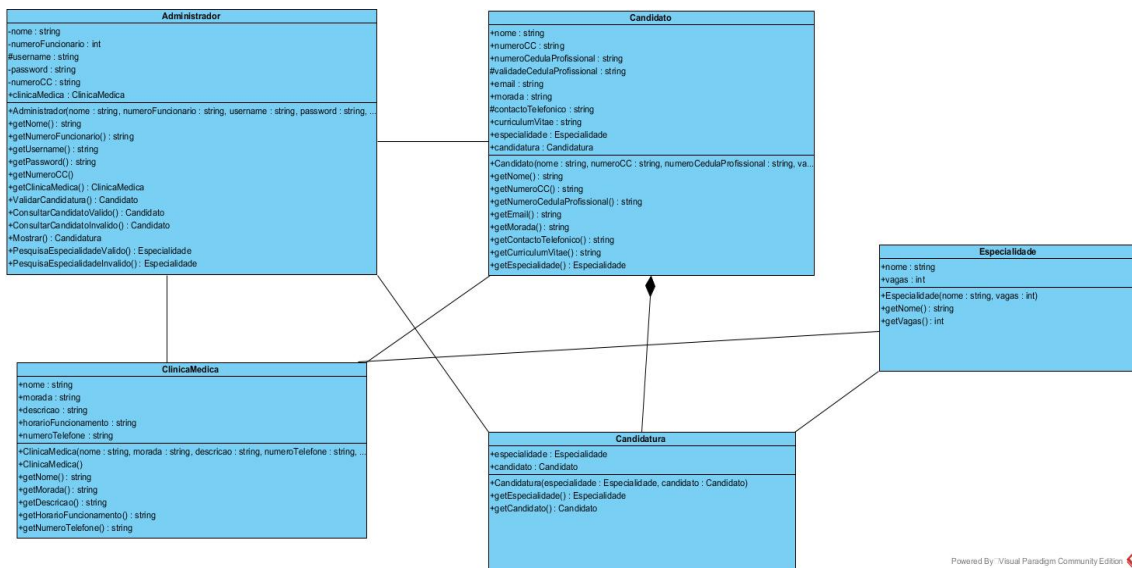


Ilustração 2 - Exemplo de diagrama de classes

O diagrama de classes representa a estrutura do sistema e as suas relações com o sistema de software, ou seja, tem como objetivo mostrar a arquitetura do sistema. Este representa o diagrama de uma forma simples e abstrata e o software que irá ser implementado no sistema informático de gestão de submissões de candidaturas espontâneas para a contratação de médicos, numa clínica médica. O diagrama de classes permite representar com detalhe cada uma das classes e como elas se correlacionam entre si.

A composição do diagrama de classes é constituída por classes, métodos, atributos e os respetivos relacionamentos entre as classes.

O nosso diagrama contém cinco classes que passaremos a explicar mais detalhadamente nos pontos abaixo:

Classe Administrador:

A classe Administrador tem como atributos o nome do administrador, o número do funcionário, o username, a password e o número do Cartão de Cidadão, que permite identificar a clínica médica.

Tem um método construtor que permite criar uma instância que vai começar por abrir a ação do administrador.

Existem 6 métodos seletores (getters) para o nome, o número de funcionário, o username, a password, o número do Cartão de Cidadão e por fim a clínica médica.

Tem um método que permite validar as candidaturas submetidas com os requisitos obrigatórios preenchidos, um que possibilita a consulta de informação de uma dada candidatura submetida não validada, um que permite consultar a informação de um dado candidato validado, um que possibilita a consulta por especialidade médica as candidaturas submetidas, um que permite saber por especialidade médica quantas e quais as novas candidaturas isto é que ainda não foram validadas e por fim outro que permite Saber por especialidade médica quantas e quais as candidaturas validadas.

Classe Candidato:

A classe Candidato recebe como atributos o nome, o número do Cartão de Cidadão, o número da Cédula Profissional, a validade da Cédula Profissional, o email, a morada, o contacto telefónico, o curriculum vitae e a especialidade.

Recebe um método construtor que permite criar uma instância que vai começar por abrir a ação do candidato.

Existem oito métodos seletores (getters): o nome, o número do Cartão de Cidadão, o número da Cédula Profissional, a validade da Cédula Profissional, o email, a morada, o contacto telefónico, o curriculum vitae e a especialidade.

Classe Candidatura:

A classe Candidatura tem como propósito a construção de uma candidatura.

A classe Candidatura recebe como atributos: a especialidade e o candidato.

Esta classe irá receber um método construtor que permite criar uma instância que vai começar por abrir a ação Candidatura.

Existem dois métodos seletores (getters): a especialidade e o candidato.

Classe Especialidade:

A classe Especialidade tem como propósito a construção de uma especialidade.

A classe Especialidade recebe como atributos: o nome e as vagas que existem em cada especialidade naquela clínica médica.

Esta classe irá receber um método construtor que permite criar uma instância que vai começar por abrir a ação Candidatura.

Existem dois métodos seletores (getters): o nome, e as vagas existentes.

Classe Clínica Médica:

A classe Clínica Médica tem como propósito a construção de um clínico médico.

Recebe como atributos a morada, que diz respeito ao local onde a clínica médica se localiza, o nome da clínica médica, a descrição, o numeroTelefone que nos dá o contacto telefónico da mesma, o horário de funcionamento que informa os clientes em que horas e que está aberto e as horas em que encerra. Esta classe irá receber um método construtor que permite criar uma instância que vai começar por abrir a ação Clínica Médica.

Existem cinco métodos seletores: a morada, o nome da clínica, a descrição, o contacto telefónico e o horário de funcionamento.

Ao analisar a relação existente entre as diferentes classes podemos observar:

- A **relação de associação** permite especificar que objetos de uma dada classe se relacionam com objetos de outra classe, sendo importante saber para cada objeto quais os objetos que lhe estão associados.

Existe este tipo de relação entre a classe administrador e clínica médica, entre a classe administrador e a classe candidato, entre a clínica médica e o candidato, entre a classe administrador e a classe candidatura, entre a classe especialidade e a classe clínica médica e por fim entre a classe candidatura e a classe especialidade. Estas relações permitem especificar

- Existe **relação de composição** entre a classe candidatura e a classe candidato. Esta relação existe devido à seguinte justificação: a instância de Candidato contém a instância Candidatura, se a instância de Candidato for excluída do sistema a correspondente instância de Candidatura também o será.

Conclusão

Este trabalho trouxe-nos alguns desafios, alguns deles foi estabelecer as ligações entre as classes e implementar algumas funções.

O trabalho em si não foi algo complexo, apesar de não funcionar de todo este trabalho foi bastante enriquecedor pois conseguimos realizar o trabalho com conhecimentos que obtivemos durante as aulas.

É de salientar que por mais que saibamos que a aplicação proposta poderia estar melhor, dentro das nossas limitações, tentamos dar o nosso melhor para chegar ao resultado final.

Contudo, ao longo da realização do trabalho, tentamos ao máximo recorrer a métodos simples e mais eficazes para que pudéssemos realizar um bom trabalho.