

RECRUTA-IA: APLICAÇÃO PARA ANÁLISE DE CURRÍCULOS PARA VAGAS UTILIZANDO RACIOCÍNIO BASEADO EM CASOS

Aluno(a): Carlos Alberto Curbani

Orientador: Francisco Adell Péricas

Roteiro

- Motivação
- Objetivos
- Fundamentação teórica
- Trabalhos correlatos
- Requisitos
- Especificação
- Implementação
- Resultados
- Conclusão

Motivação

- Usar casos anteriores para solucionar problemas atuais.
- Utilizar Inteligência Artificial para resolução de problemas.
- Facilitar o cadastro dos candidatos e vagas
- Auxiliar o setor de RH na contratação de novos funcionários.

Objetivos

Objetivo geral:

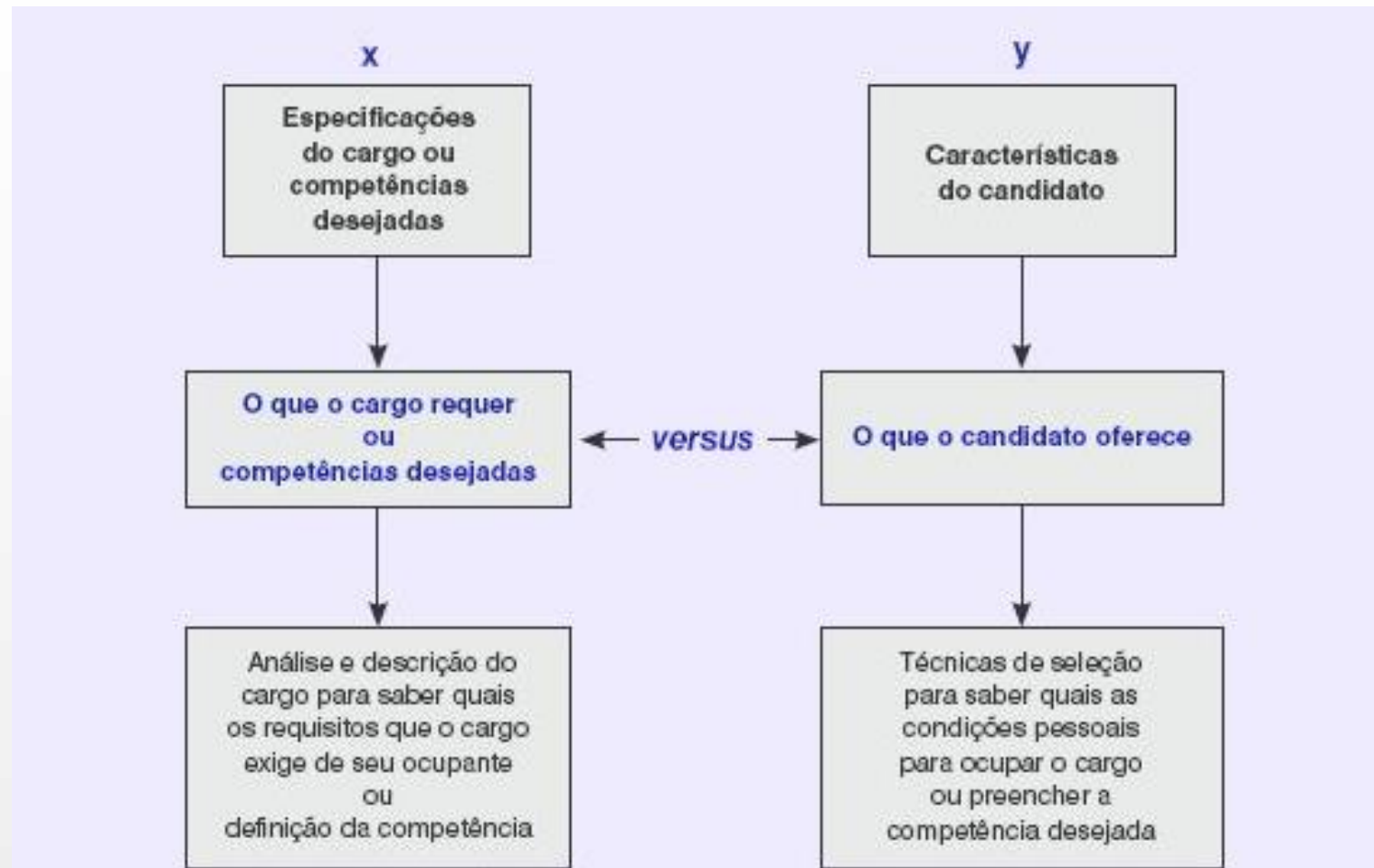
Desenvolver uma aplicação para auxiliar na análise dos currículos para preenchimento de vagas utilizando técnicas de Raciocínio Baseado em Casos(RBC)

Objetivos específicos:

- disponibilizar uma interface Web para cadastro das vagas e de currículos recebidos
- desenvolver uma integração com o LinkedIn
- possibilitar definir o valor do peso dos atributos

Fundamentação Teórica

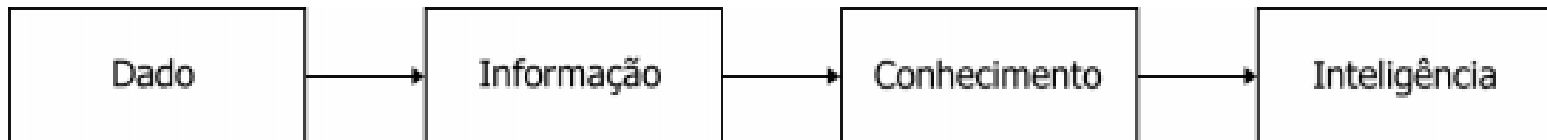
Recursos Humanos



Fundamentação Teórica

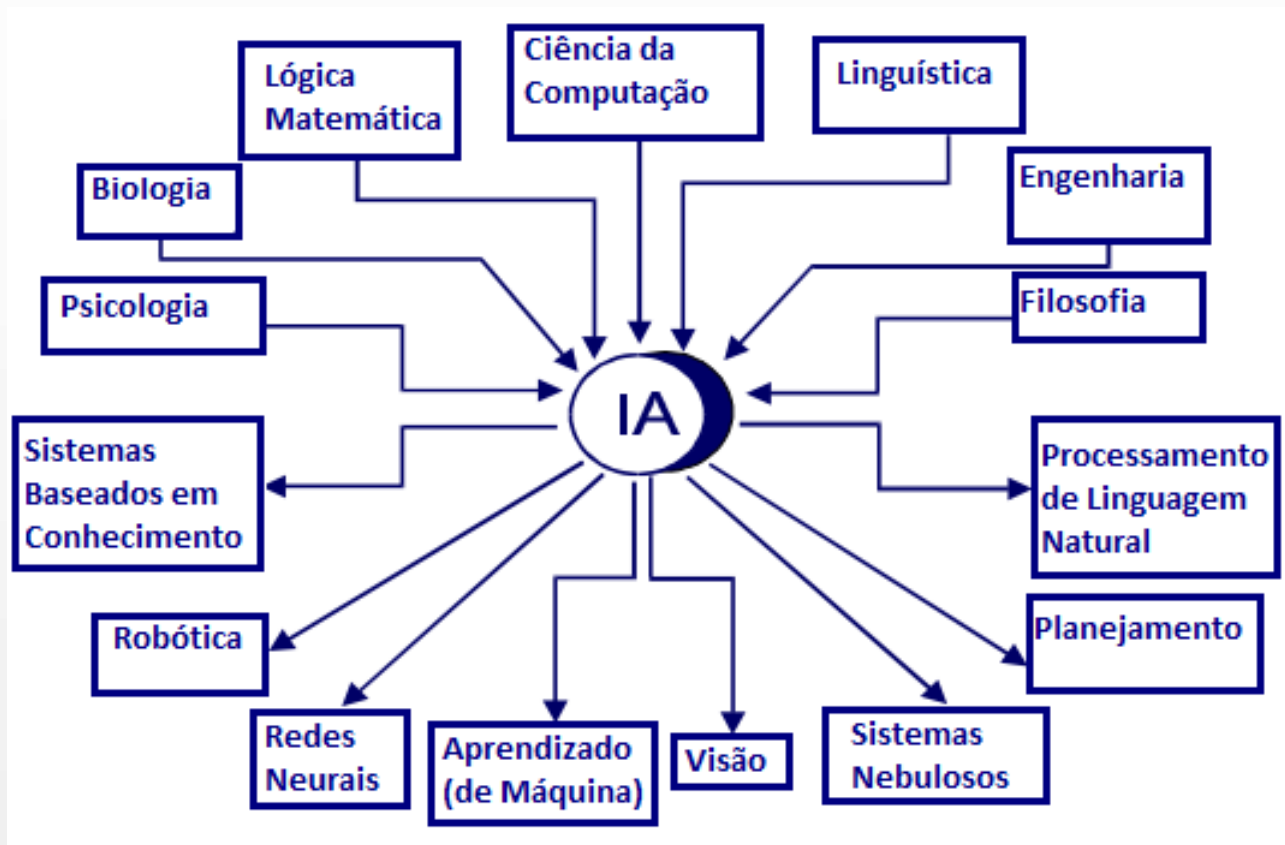
Inteligência Artificial

- Fazer o computador pensar ou se comportar de forma inteligente.



Fundamentação Teórica

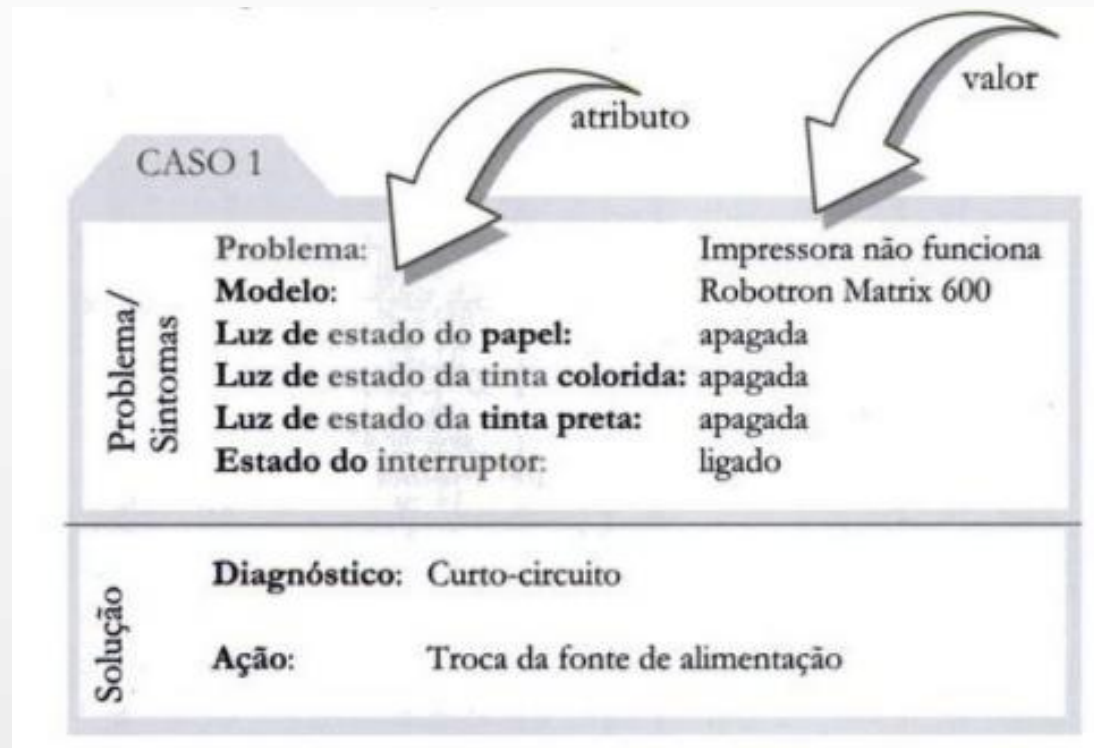
Inteligência Artificial



Fundamentação Teórica

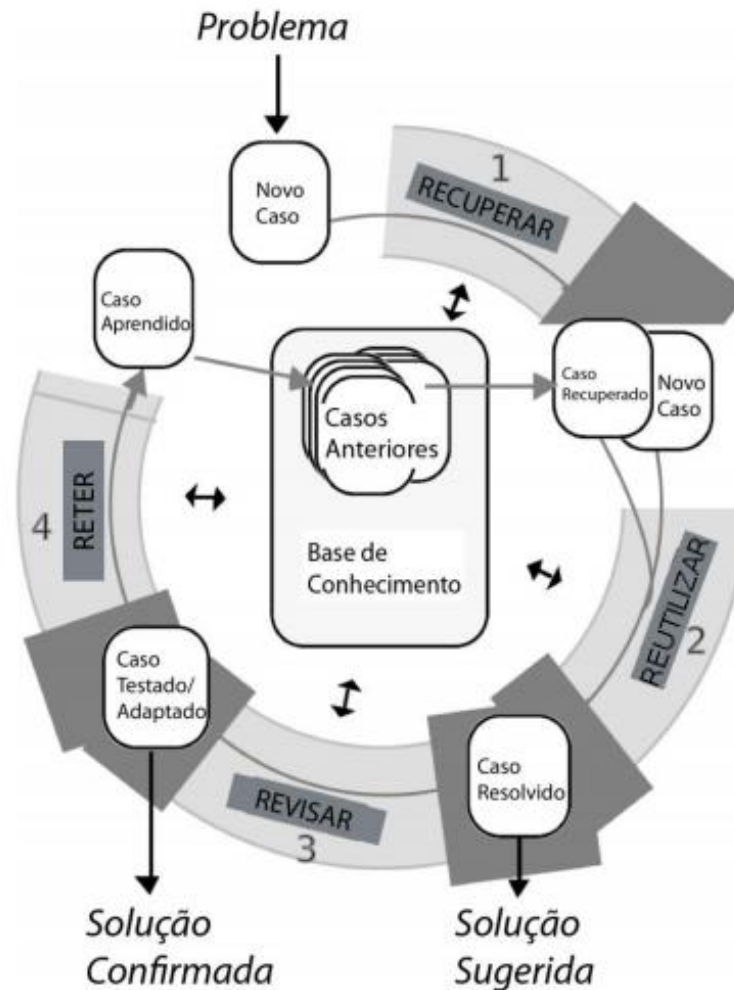
Raciocínio Baseado em Casos (RBC)

- Utilizar os casos anteriores para solução do problema atual.



Fundamentação Teórica

Raciocínio Baseado em Casos (RBC)



Trabalhos correlatos (1/3)

Título: Sistema RBC para Recrutamento e Seleção de Profissionais
Silva (2004)

Cadastrados

Dados Pessoais | Idiomas | Formação | Contatos | Cursos | Cargo Pretendido | Emprego Anterior

dados pessoais

Dados do Candidato

Código: []
Nome: DANIELE FERNANDA DE LIMA
Sexo: Femi. | Nascido: 18/10/1980 | Est. Civil: Solteiro
RG: 45454545 | Emissão: 15/02/2004 | Orgão: SSPSC
CPF: 155.155.155-1 | Nacionalidade: Brasileiro
Possui Veículo: Sim Não | Possui Habilitação: Sim Não | Mod. Carteira

Endereço

Logradouro: RUA FREI JUSTINO
Número: 35 | Compl.: CASA 2º ANDAR
Bairro: Centro | CEP: 88.888-888
Cidade: Lages | UF: SC

Outros Dados Pessoais

Deficiência: 3 Não possui
Conjuge: []

Filhos

Nome	Nascimento

Comentários

teste

Gravar | Cancela | Fechar

Característica/ Trabalho Relacionados	Silva (2004)
Técnica de IA utilizada	RBC
Usuários com acesso à aplicação	Profissionais do RH da empresa específica
Possui versão Web	Não
Linguagem de programação	Delphi
Foco do recrutamento	Candidatos interessados nas vagas de uma empresa específica
Integração com Aplicação externa	Não

Trabalhos correlatos (2/3)

Título: Desenvolvimento de Sistema Online para Recrutamento de Pessoas
Preis (2013)

Lista de Vagas Disponíveis

Panfletagem - Vaga para Estágio
Descrição: Realizar Panfletagem. Vestibular Em toda a região Oeste de Santa Catarina e Noroeste do Rio Grande do Sul
Benefícios: Vale Alimentação. Vale Transporte
Salário: R\$ 900,00
Informações de Contato: Falar com: Giovane Preis
Telefone: (49) 8423-8859
E-mail: giupreis@gmail.com
Disponível: De 05/10/2013 até 31/10/2013
[Concorrer à Vaga](#)

Programação Web
Descrição: Desenvolvimento de sistemas online
Benefícios: Este cargo é um teste do sistema
Salário: 1000,00
Informações de Contato: Empresa: Empresa TI Profissional
Falar com: Giovane
Telefone: (99) 9999-9999
E-mail: empresa@tiprofissional.com
Disponível: De 27/10/2013 até 31/10/2013
[Não Concorrer à Vaga](#)

Característica/ Trabalho Relacionados	Preis (2013)
Técnica de IA utilizada	Nenhuma
Usuários com acesso à aplicação	Profissionais do RH e candidatos a vagas
Possui versão Web	Sim
Linguagem	C#
Foco do recrutamento	Candidatos para vagas cadastradas no site por diversas empresas
Integração com Aplicação externa	Não

Trabalhos correlatos (3/3)

Título: SIIEE – Sistema de Raciocínio Baseado em Casos Para Análise de Perfil de Candidatos a Vagas de Estágio

Panceri (2012)

SIIEE - Buscar Currículos

Selecionar Oportunidade

Código da Oportunidade: 1 Importância

Horário: 1.0

Residência

Cidade: 1.0

Conhecimentos Gerais

Informática: 1.0

Espanhol: 1.0

Inglês: 1.0

Conhecimentos Específicos

1.0

1.0

1.0

1.0

Dados Acadêmicos

Curso: 1.0

Ano/Semestre: 1.0

Nível Escolaridade: 1.0

Cancelar Executar Busca

Característica/ Trabalho Relacionados	Panceri (2012)
Técnica de IA utilizada	RBC
Usuários com acesso à aplicação	Profissionais do RH e candidatos a vagas de estágio
Possui versão Web	Não
Linguagem	Java
Foco do recrutamento	Alunos para estágios em empresas
Integração com Aplicação externa	Não

Requisitos Funcionais

- Permitir realizar login no sistema.
- Permitir cadastrar currículos recebidos pela empresa.
- Permitir cadastrar vagas disponíveis na empresa.
- Permitir configurar o peso dos atributos usados no cálculo da similaridade.
- Aplicar a técnica de RBC para comparar a similaridade entre os candidatos e as vagas disponíveis.
- Exibir um relatório com o resultado da similaridade da vaga processada.
- Permitir acesso ao candidato para executar o próprio cadastro.
- Possibilitar ao candidato executar a busca de informações no LinkedIn, para completar seu cadastro.

Requisitos Não Funcionais

- Implementar a aplicação usando Java, JavaScript e HTML.
- Implementar as páginas da aplicação utilizando o *framework* Primefaces.

Especificação

Diagrama de casos de uso

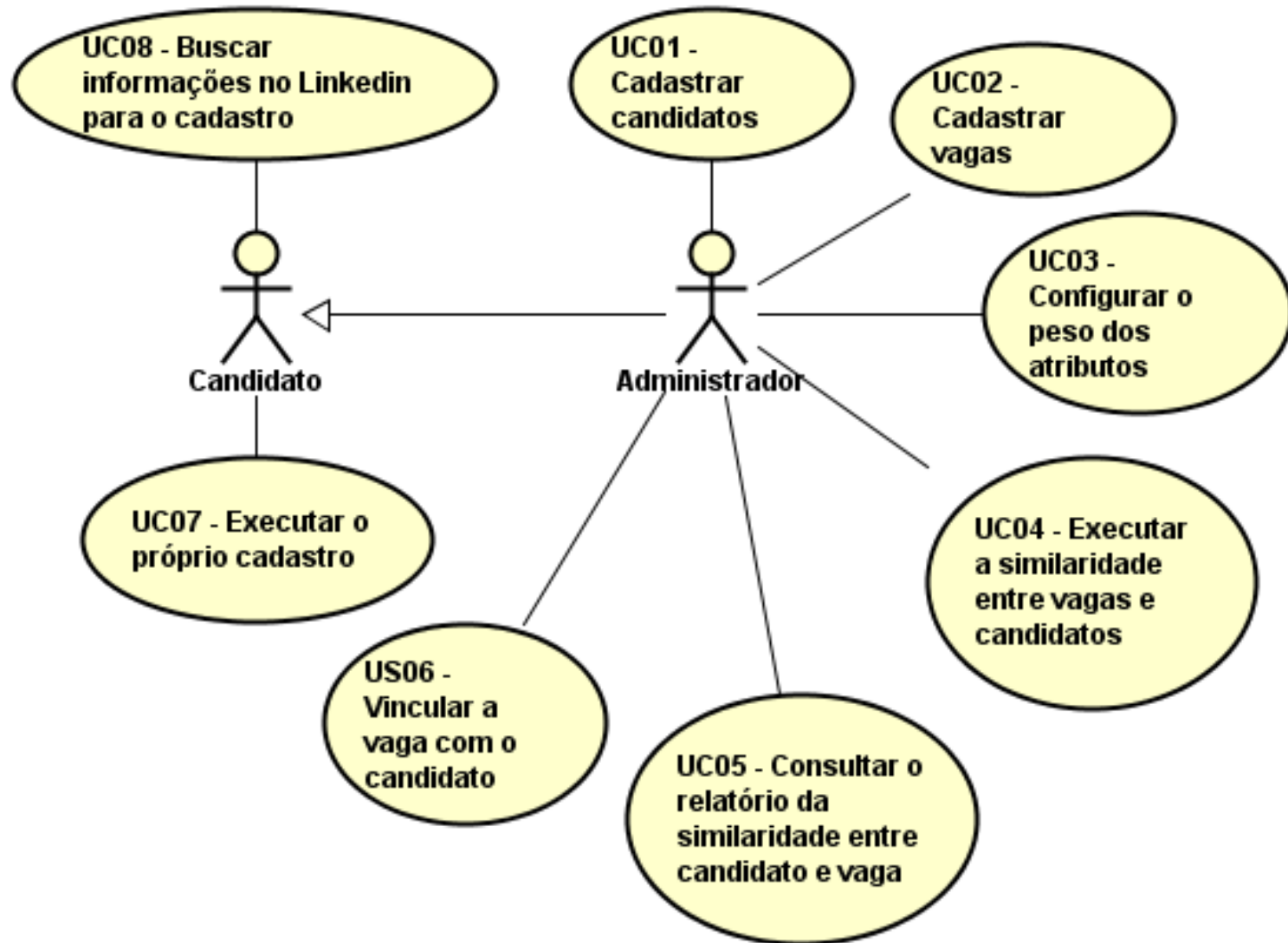


Diagrama de classe: Vaga

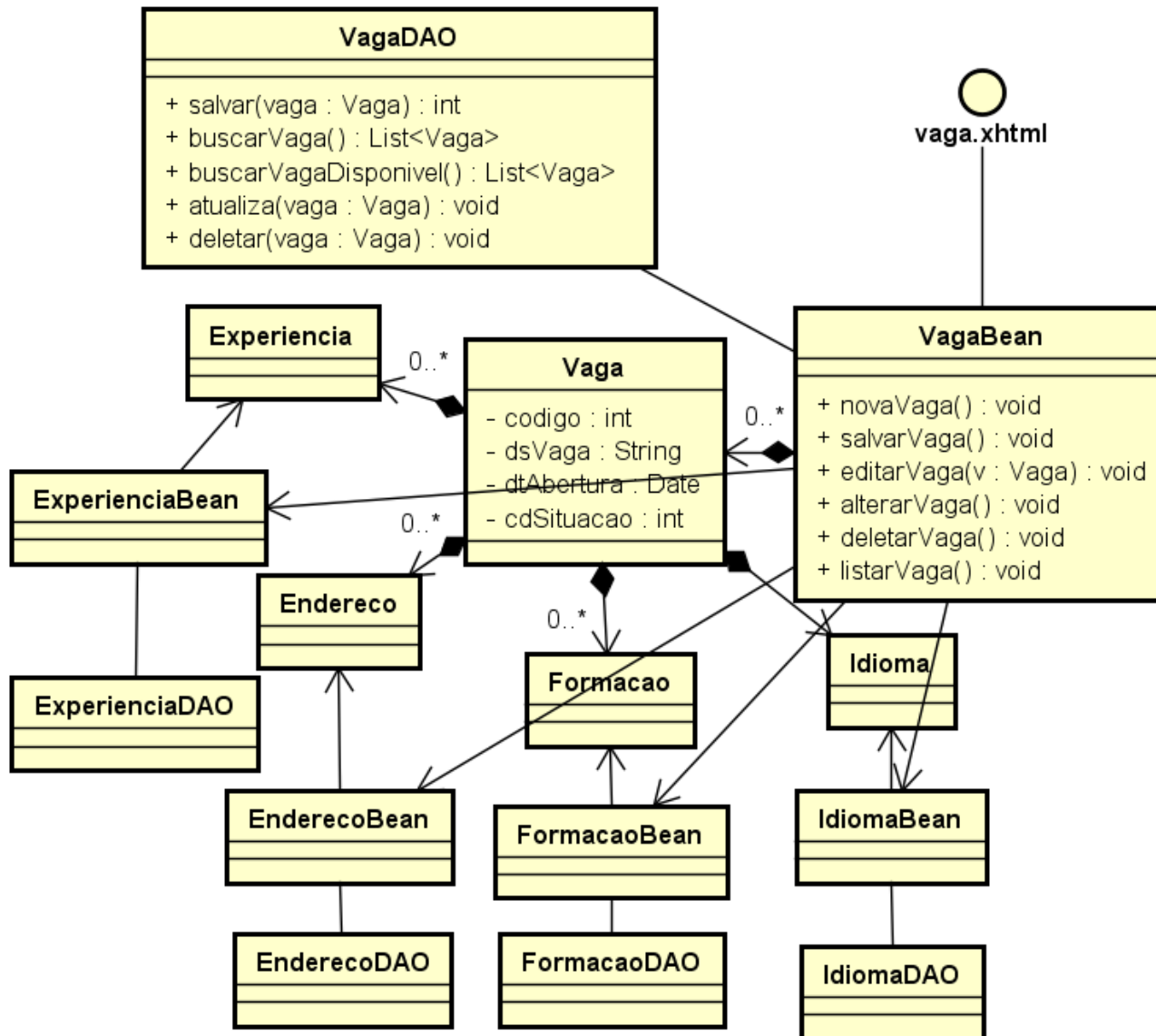


Diagrama de classes: Similaridade

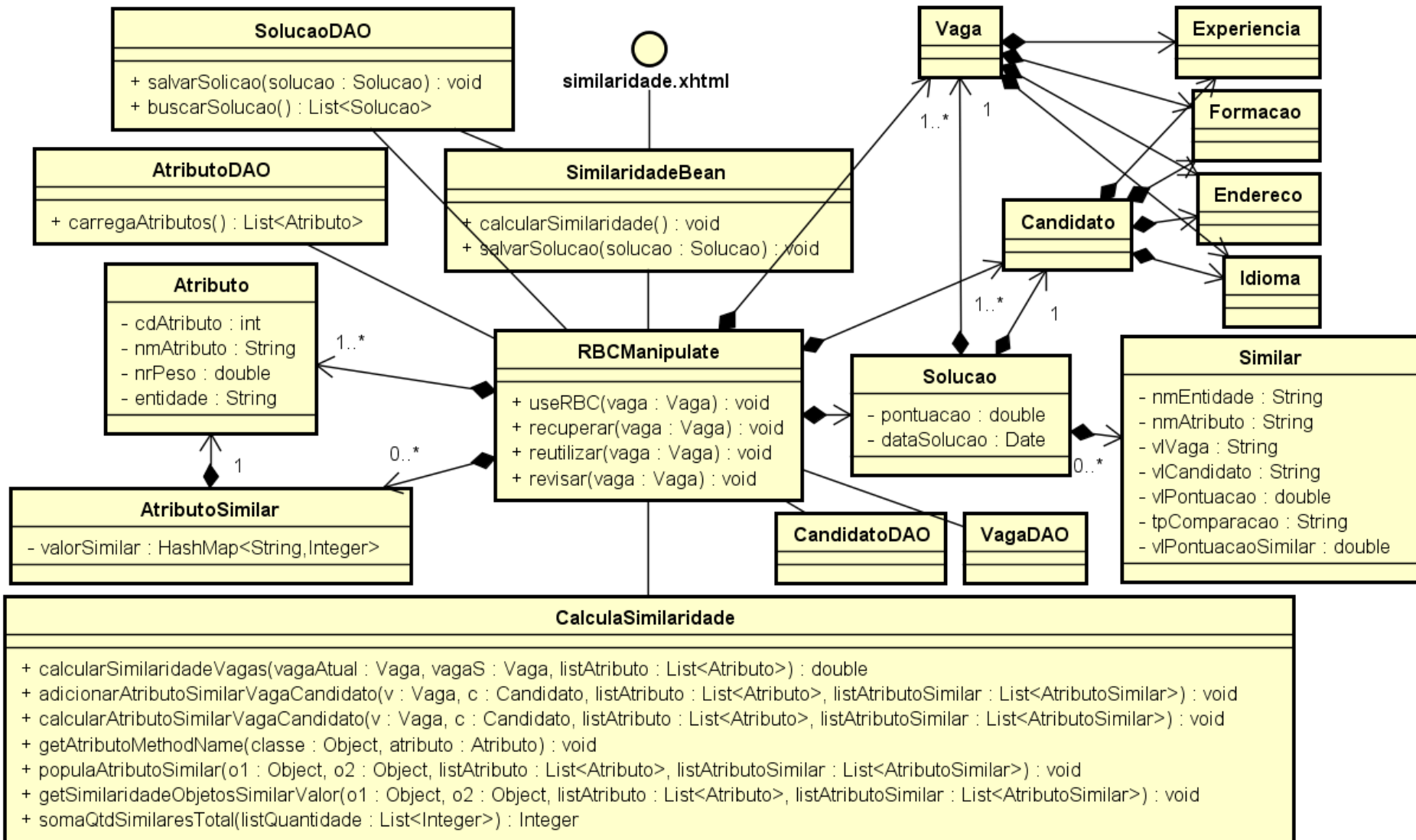
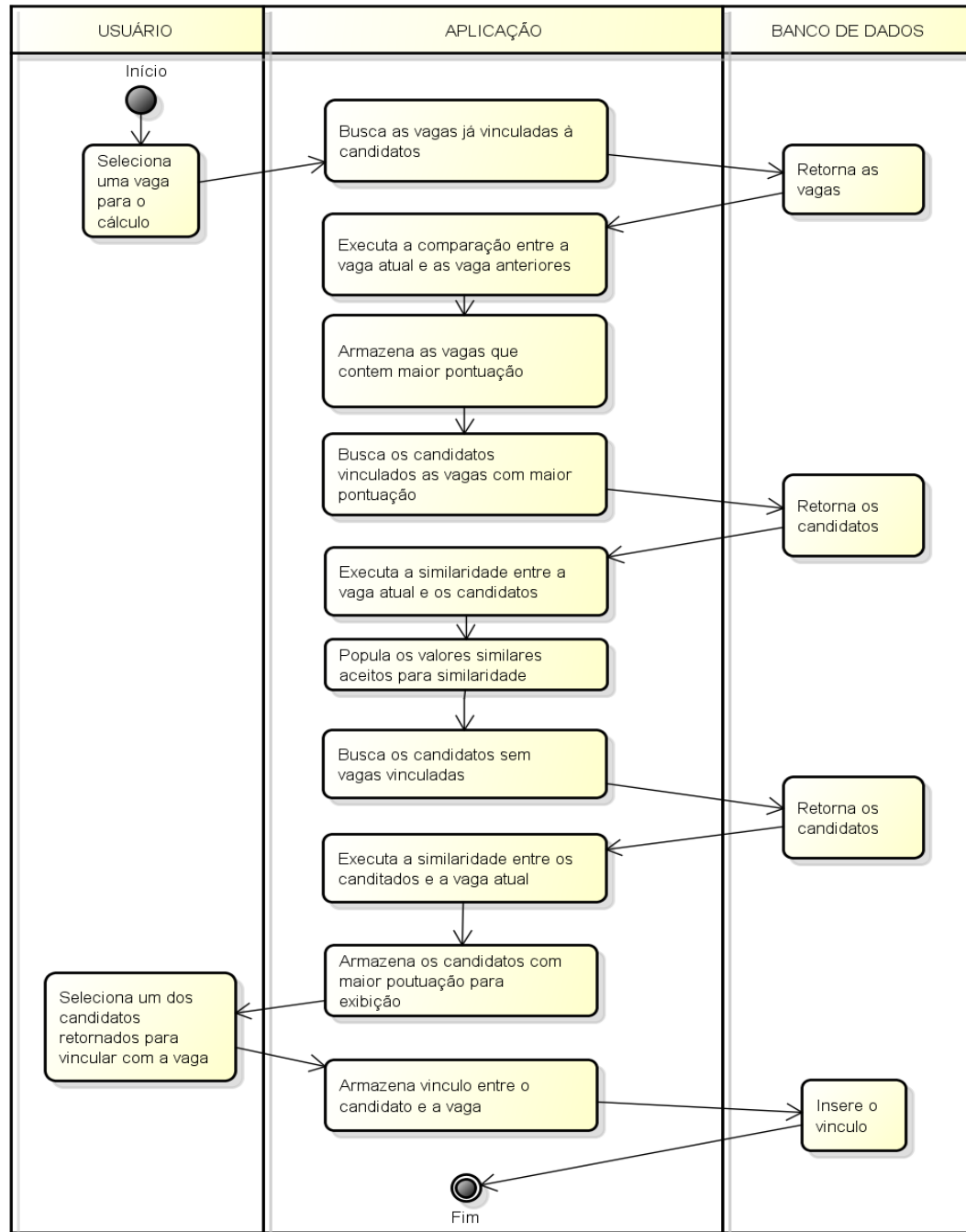


Diagrama de atividade



Implementação

Ciclo do RBC

```
public static void recuperar(Vaga vagaAtual) {
    int nrQtdVagas = Integer.parseInt(
ParametroDAO.buscarValorParametro("recruta.qtd.vagas.similares"));
    List<Solucao> listSolucaoConsultada = new ArrayList<Solucao>();
    List<Solucao> listSolucaoPontuacaoCalculada = new ArrayList<Solucao>();
    listSolucaoConsultada = SolucaoDAO.buscarVagaComSolucao();
    listSolucoes = new ArrayList<Solucao>();
    for (Solucao solucaoUnica : listSolucaoConsultada) {

        solucaoUnica.setVaga(carregaVaga(solucaoUnica.getVaga().getCodigo()));
        solucaoUnica.setCandidato(carregaCandidato(
solucaoUnica.getCandidato().getCodigo()));
        solucaoUnica.setNrPontuacao(

        CalculaSimilaridade.calcularSimilaridadeVagas(vagaAtual,
solucaoUnica.getVaga(), listAtributos));
        listSolucaoPontuacaoCalculada.add(solucaoUnica);
    }
    SolucaoController.ordenarNrPontuacaoDesc(listSolucaoPontuacaoCalculada);
    for (Solucao solucaoUnica : listSolucaoPontuacaoCalculada) {
        listSolucoes.add(solucaoUnica);
        if (listSolucoes.size() >= nrQtdVagas) {
            break;
        }
    }
    reutilizar(vagaAtual);
}
```

Similaridade: Atributos

```
@AnotacaoEntidade(nome="endereco")
public class Endereco {
    private Integer cdEndereco;
    private String cdCEP;
    private String nmCidade;
    private String nmEstado;
    private String nmBairro;
    private String nmRua;
    private int nrEndereco;

    @AnotacaoAtributo(atributo="cidade", comparacao=TipoComparacaoEnum.IGUAL)
    public String getNmCidade() {
        return nmCidade;
    }
    public void setNmCidade(String nmCidade) {
        this.nmCidade = nmCidade;
    }

    @AnotacaoAtributo(atributo="estado", comparacao= TipoComparacaoEnum.IGUAL)
    public String getNmEstado() {
        return nmEstado;
    }
    public void setNmEstado(String nmEstado) {
        this.nmEstado = nmEstado;
    }
}
```

Similaridade: Cálculo da pontuação

```
public static BigDecimal getSimilaridadeObjetosSimilarValor(Object objeto1, Object
objeto2,
List<Atributo> listAtributos, List<AtributoSimilar> listAtributosSimilares,
List<Similar> listSimilares) throws IllegalAccessException, IllegalArgumentException,
InvocationTargetException {
    BigDecimal somaSimilaridade = new BigDecimal(0);
    ...
    if (tipoValor == TipoComparacaoEnum.IGUAL) {
        if (valor.equalsIgnoreCase(valo2r)) {
            somaSimilaridade = somaSimilaridade.add(atributo.getNrPeso());
            populaSimilarUnico(similarUnico, atributo, valor, valo2r, "Igualdade",
atributo.getNrPeso());
        } else {
            ...
            if (unicoAtrSimilar.getValorSimilar().containsKey(valo2r)) {
                ...
                BigDecimal pontuacao = new BigDecimal(unicoAtrSimilar.getValorSimilar().get(valo2r))
                .divide(new BigDecimal(qtdValoresSimilares), 3, RoundingMode.HALF_UP)
                .multiply(atributo.getNrPeso().multiply(vlPorcentagemSimilar)
                .divide(new BigDecimal(100), 3, RoundingMode.HALF_UP));
                somaSimilaridade = somaSimilaridade.add(pontuacao);
                populaSimilarUnico(similarUnico, atributo, valor, valo2r, "Similar", pontuacao);
            }
            ...
        }
        return somaSimilaridade;
    }
}
```

Linkedin

```
function OnLinkedInAuth() {
    IN.API.Profile("me").fields(["firstName","lastName", "emailAddress","positions"])
    .result(function(result) {
var dataFim;
var dataInicio;
var nm_candidato = result.values[0].firstName + ' ' + result.values[0].lastName;
$("#id$='nome_completo']").val(nm_candidato);
$("#id$='email_contato']").val(result.values[0].emailAddress);
if(result.values[0].positions.values[0].startDate != null){
    dataInicio = '01';
    dataInicio += '/' + result.values[0].positions.values[0].startDate.month ;
    dataInicio += '/' + result.values[0].positions.values[0].startDate.year ;
    $("#id$='exp_dt_entrada_input']").val(dataInicio);
}
if(result.values[0].positions.values[0].endDate != null){
    dataFim = '01';
    dataFim += '/' + result.values[0].positions.values[0].endDate.month ;
    dataFim += '/' + result.values[0].positions.values[0].endDate.year ;
    $("#id$='exp_dt_saida_input']").val(dataFim);
}
$("#id$='exp_nm_empresa']").val(result.values[0].positions.values[0].company.name);
$("#id$='exp_nm_cargo']").val(result.values[0].positions.values[0].title);
closeSession();
    });
}
```


Operacionalidade da Implementação

Página de similaridade



Recruta-IA

Home Candidatos Vagas Similaridade Soluções Atributos Parâmetros

Similaridade

Listar

Lista de Vagas disponíveis

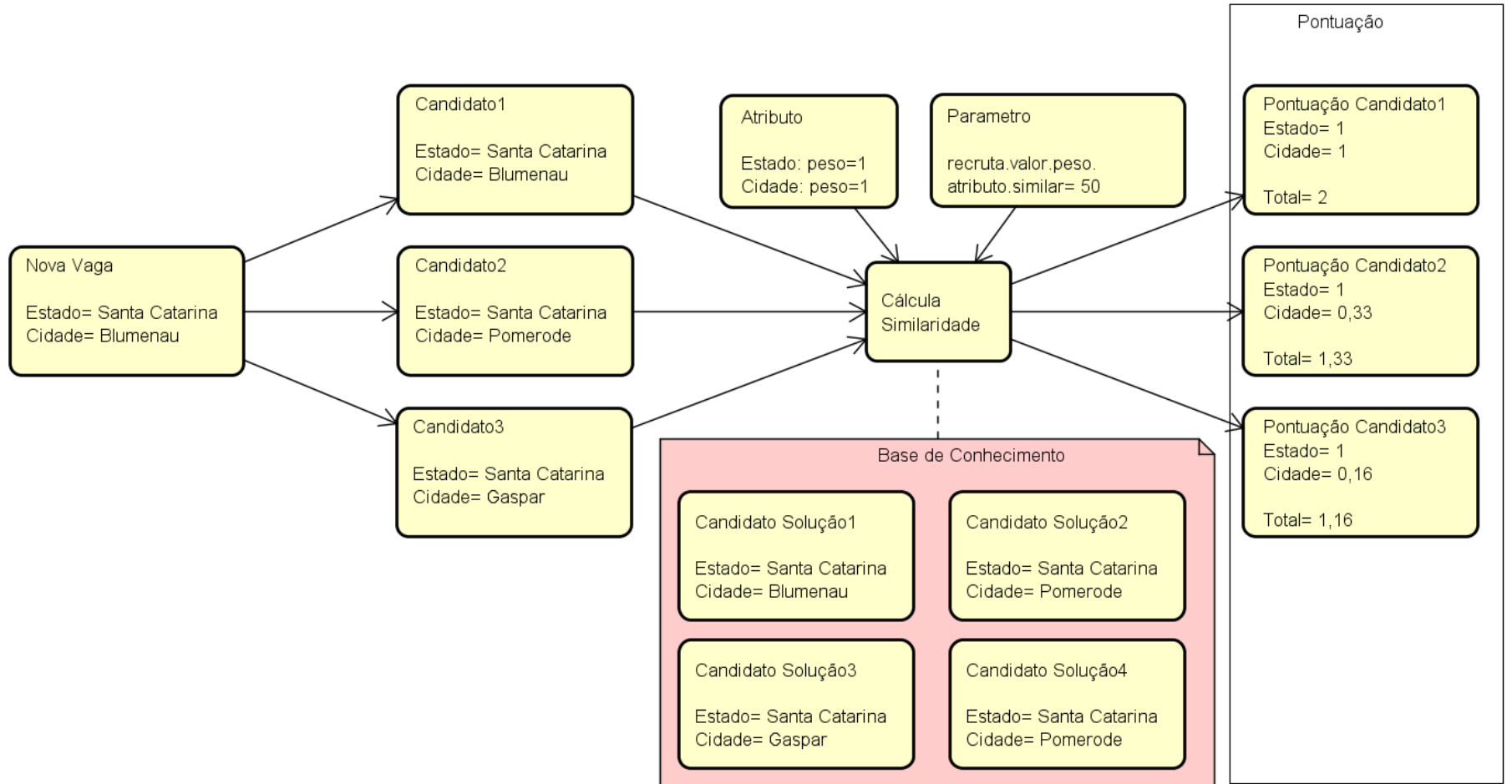
	Descrição	Data Abertura	Situação
	Programador delphi	2017-09-10	1
 Calc. Similar	Programador delphi	2017-08-04	1

Lista de Candidatos possíveis

	Pontuação	Nome Candidato	CPF
 	16.00	Eduardo	987.654.321-00
 	14.75	Gabriel	888.888.888-88
 	11.00	Ana	222.222.222-22

Resultados e Discussões

Funcionamento da similaridade



Comparação com os correlatos

Característica/Tra- balho Relacionados	Sistema RBC para Recrutamento e Seleção de Profissionais (SILVA, 2004)	Desenvolvimento de sistema online para recrutamento de pessoas (PREIS, 2013)	SIIEE - Sistema de Raciocínio baseado em casos para análise de perfil de candidatos a vagas de estágio (PANCERI, 2012)	Recruta-IA: trabalho proposto
Técnica de inteligência artificial utilizada	RBC	Nenhuma	RBC	RBC
Usuários com acesso à aplicação	Profissionais do RH da empresa especifica	Profissionais do RH e candidatos a vagas	Profissionais do RH e candidatos a vagas de estágio	Profissionais do RH e candidatos à busca de emprego
Possui versão Web	Não	Sim	Não	Sim
Linguagem de programação	Delphi	C#	Java	Java
Foco do recrutamento	Candidatos interessados nas vagas de uma empresa especifica	Candidatos para vagas cadastradas no site por diversas empresas	Alunos para estágios em empresas	Foco específico na análise de candidatos que se encaixem nas vagas cadastradas
Integração com Aplicação externa	Não	Não	Não	Linkedin

Conclusões

- O objetivo principal foi concluído.
- Aplicação Web com uso de IA para auxiliar na tomada de decisão.

Sugestões

- Aprimorar os cadastros.
- Desenvolver um Webservice para integração de informações vindas de outras aplicações.
- Utilizar a permissão completa do LinkedIn.

Demonstração...