

Documentação de visão
Versão 1.0 (*Assessment*)

Histórico de Revisões

Data	Versão	Descrição	Autor
	1.0	Versão inicial do documento	Tauan Santana

Índice

NOME DO SISTEMA	4
1 Erro! Indicador não definido.	
Erro! Indicador não definido.	
Erro! Indicador não definido.5	
Erro! Indicador não definido.5	
2 Erro! Indicador não definido.	
2.1 Erro! Indicador não definido.	
3 Erro! Indicador não definido.6	
3.1 Erro! Indicador não definido.6	
4 Erro! Indicador não definido.7	
4.1 Erro! Indicador não definido.7	
4.2 Erro! Indicador não definido.7	
4.2.1 78	
Erro! Indicador não definido.	
9 – MICROSERVIÇOS E ENDPOINTS	15

VenturaHR

1 Introdução

O RH 2.0 ganhou protagonismo e está sendo promovido a área estratégica, essencial para a tomada de decisão e crescimento do negócio.

Problemas antigos como a dificuldade em medir o retorno sobre o investimento (ROI) das ações, ineficiência do recrutamento e baixo desempenho em treinamentos já têm suas soluções tecnológicas sob medida.

As HRTechs são startups que desenvolvem soluções tecnológicas para a área de Recursos Humanos (Human Resources), agregando inteligência aos processos. A missão dessas empresas é levar a inovação ao RH, mostrando que a tecnologia pode reduzir custos, aumentar a eficiência e agilizar o crescimento do setor.

- Cenário Atual

VenturaSoft é uma HRTech que atua no segmento de recolocação de profissionais de TI. Devido às peculiaridades desse mercado, os requisitos para contratação têm um dinamismo vertiginoso, pois as tecnologias e as “stacks” adotadas pelas empresas estão em constante evolução.

A empresa necessita que seja construída uma solução de software, chamado de VenturaHR, que tenha abrangência de todos os fluxos operacionais da sua atividade fim.

A VenturaSoft tem como clientes empresas que precisam fazer processos seletivos para vagas em aberto.

- Solução Desejada

O software VenturaHR precisa ter como diferencial o fato de não manter um banco de currículos e sim um banco de Vagas X Critérios X Candidatos.

Propósito do documento de requisitos

Este documento destina-se aos clientes, engenheiros e gerentes envolvidos no desenvolvimento do sistema, referido apenas como VenturaHR. O propósito deste documento é apresentar a descrição dos serviços e funções que o sistema a ser desenvolvido deve prover, bem como as suas restrições de operação e propriedades gerais, a fim de ilustrar uma descrição detalhada do sistema para um auxílio durante as etapas de análise, projeto e testes. O documento especifica todos os requisitos funcionais do sistema e foi preparado levando-se em conta as funcionalidades levantadas durante a fase de concepção do sistema.

Escopo do produto

O projeto consiste na construção de um sistema (aplicação web) cujo o objetivo é disponibilizar ajudar empresas a disponibilizar vagas de trabalho e ajudar pessoas que estão em busca de emprego a encontrarem empresas com vagas em aberto.

2 Descrição geral

2.1 Usuários do sistema

Empresa: Realizam o cadastro de vagas assim como a administração do perfil desejado de candidatos e descrições das vagas.

Candidato: Usuário que se candidata às vagas disponíveis que foram cadastradas previamente pelas empresas. Também é possível preencher o perfil com informações pessoais e profissionais.

Administradores: Responsáveis pelo gerenciamento da plataforma, podendo visualizar o total de vagas disponíveis e indicadores.

3 Requisitos do Software

3.1 Requisitos Funcionais

<RF01><Login>

O sistema deve permitir que o usuário possa logar com suas credenciais (CPF, matrícula ou CNPJ, e senha).

<RF02><Gerenciar vagas>

O sistema deve provê meios para que empresas sejam capazes de incluir e editar de vagas de emprego no sistema, assim como permitir o contato entre empresas e candidatos por meio da plataforma.

<RF03><Gerenciar candidaturas>

O candidato deve ser capaz de se candidatar a vagas de emprego disponíveis na plataforma, assim como visualizar histórico de candidaturas, mensagens de empresas e cancelar candidaturas.

4 Casos de uso

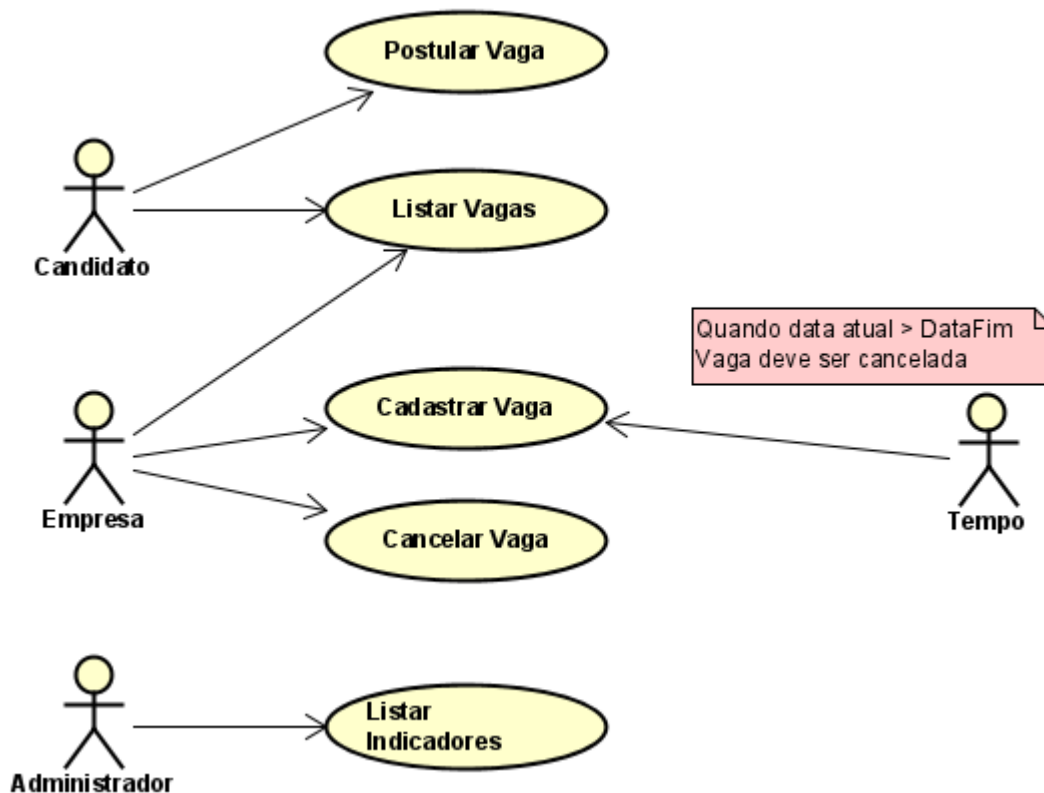
4.1 Diagrama de casos de uso

O diagrama de casos de uso, expresso em UML (*Unified Modeling Language*), expressa os requisitos funcionais do sistema na forma de casos de uso. Segundo o RUP (*Rational Unified Process*), para cada requisito funcional tem-se um caso de uso. A descrição textual detalhada dos requisitos funcionais, seus fluxos de atividades e requisitos não funcionais associados pode ser encontrada na próxima seção. Na figura abaixo mostramos a representação gráfica em UML dos casos de uso do sistema.

4.2 Descrição dos casos de uso

Dentre os casos de uso do sistema mostrados no diagrama de casos de uso, foram escolhidos dois para serem detalhados e trabalhados nas fases de análise e projeto do sistema.

4.2.1 Diagrama de caso de uso

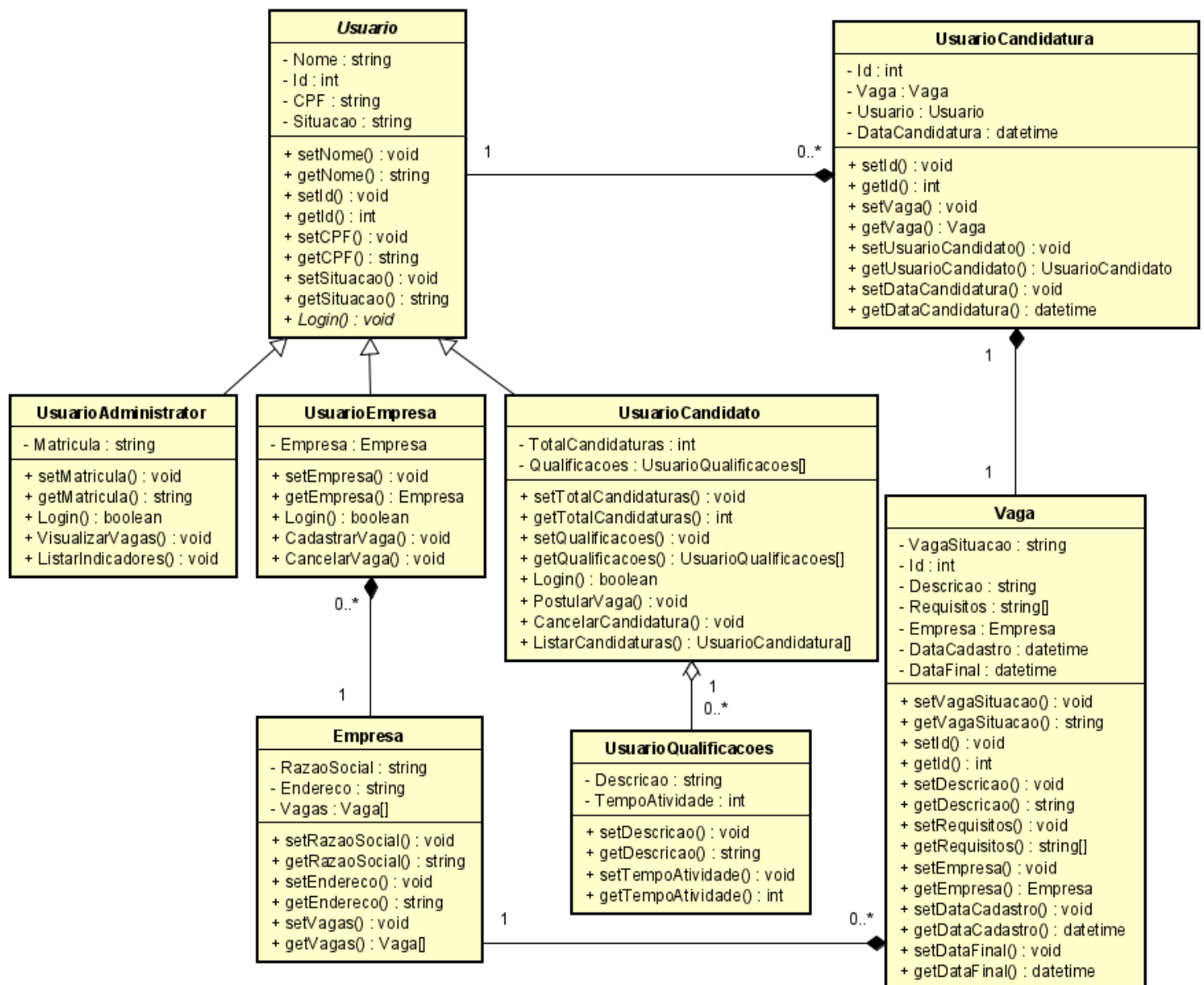


4.2.2 Casos de uso

[CDU-01]		
Nome:	Listar Vagas	
Atores:	Candidato, Empresa e Administrador	
Prioridade:	Essencial	
Entradas e pré-condições:		<ul style="list-style-type: none">• O usuário deve estar logado no sistema com perfil de “candidato”, “empresa” ou “administrador”
Fluxos de eventos		
Fluxo principal:	<div>1. Sistema lista na base de dados as vagas com situação “Publicada”<div>1.1. Se o perfil for “empresa”, sistema deverá exibir somente vagas publicadas pela empresa do usuário logado</div><div>1.2. Se o perfil for “candidato” ou “administrador” sistema exibe todas as vagas com situação “Publicada”, como citado no fluxo 1.</div></div>	
Fluxo alternativo	<div>2. Se não houver vagas cadastradas o sistema deverá exibir “Não há vagas disponíveis”, se for o perfil “administrador” ou “candidato”.<div>2.1 Se não houver vagas cadastradas o sistema deverá exibir “Não há vagas cadastradas”, se for o perfil “empresa”.</div></div>	

[CDU-02]		
Nome:	Postular Vaga	
Atores:	Candidato	
Prioridade:	Essencial	
Entradas e pré-condições:		<ul style="list-style-type: none">• O usuário deve estar logado no sistema com perfil de “candidato”
Fluxos de eventos		
Fluxo principal:	<ol style="list-style-type: none">1. Usuário executa o CDU-012. Usuário clica em “Ver detalhes” na vaga que ele desejar na tela do CDU-013. Sistema deverá redirecionar o usuário para uma tela com os detalhes da vaga.4. Usuário clica em “Postular”.5. Sistema exibe mensagem “Candidatura confirmada”.6. Sistema redireciona para a tela do CDU-01	
Fluxo alternativo	<ol style="list-style-type: none">7. Se o sistema apresentar erro, exibir a mensagem “Erro durante a confirmação da sua candidatura, tente novamente”.	

5 Diagrama de Classes



Descrição das classes

- <**Usuário**>: Classe abstrata que implementa as propriedades básicas de um usuário assim como o método abstrato "Login".

- <**UsuarioAdministrador**>: Classe que representa o tipo de usuário "Administrador" no sistema. Ela herda de Usuário, logo, torna-se acessível as propriedades e o método abstrato "Login".

Com essa especialização, é possível executar funções no sistema que só o tipo "Administrador" pode executar, como por exemplo, "ListarIndicadores".

- **Métodos:**

- **Login:** Método que implementa Login da classe "Usuário" e que preenche as informações de get/set do usuário "Administrador".

- **VisualizarVagas:** Lista todas as vagas com um totalizador.

- **ListarIndicadores:** Listar o total de vagas, candidatos e candidaturas do sistema.

- <**UsuarioEmpresa**>: Classe que representa o tipo de usuário "Empresa" no sistema. Ela herda de Usuário, logo, torna-se acessível as propriedades e o método abstrato "Login".

Com essa especialização, é possível executar funções no sistema que só o tipo "Empresa" pode executar, como por exemplo, "CadastrarVagas".

Essa classe possui um relacionamento de composição com a classe "Empresa", já que usuário do tipo "Empresa" só existe se for associado à uma empresa.

- **Métodos:**

- **Login:** Método que implementa Login da classe "Usuario" e que preenche as informações de get/set do usuário "Empresa".

- **CadastrarVaga:** Realiza o cadastro de uma vaga.

- **CancelarVaga:** Cancela o cadastro de uma vaga.

- <**UsuarioCandidato**>: Classe que representa o tipo de usuário "Candidato" no sistema. Ela herda de Usuário, logo, torna-se acessível as propriedades e o método abstrato "Login".

Com essa especialização, é possível executar funções no sistema que só o tipo "Candidato" pode executar, como por exemplo, "PostularVaga".

Essa classe possui um relacionamento de agregação com a classe

"CandidatoQualificacoes", já que não é necessário existir "qualificações" para que o usuário exista.

- **Métodos:**

- **Login:** Método que implementa Login da classe "Usuario" e que preenche as informações de get/set do usuário "Candidato".

- **PostularVaga:** Realiza a candidatura do associado à uma vaga.

- **CancelarCandidatura:** Cancela uma candidatura realizada pelo usuário.

- **ListarCandidaturas:** Lista todas as candidaturas realizadas pelo usuário.

<**Empresa**>: Classe que representa a empresa dentro do sistema.

Ela é responsável pelos dados básicos da empresa e contém a lista de vagas já cadastradas por ela.

Essa classe possui um relacionamento de composição com a classe

"UsuarioEmpresa" e "Vaga", já que usuário do tipo "empresa" e a classe "Vaga" só podem existir se forem associados a uma "Empresa".

<**CandidatoQualificacoes**>: Classe que representa a empresa dentro do sistema.

Ela é responsável pelos dados básicos da empresa e contém a lista de vagas já

cadastradas por ela.

Essa classe possui um relacionamento de agregação com a classe

"**UsuarioCandidato**", já que "UsuarioCandidato" pode existir sem "Qualificações".

<**Vaga**>: Classe que representa uma vaga disponibilizada por uma empresa.

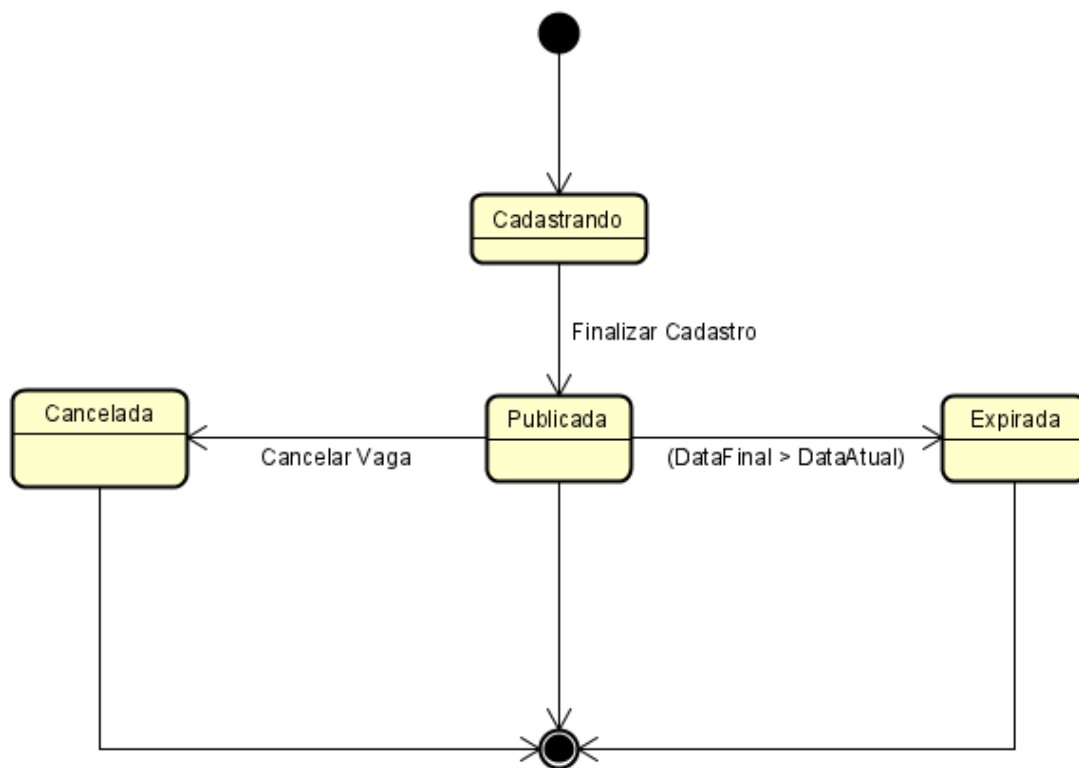
Essa classe possui um relacionamento de composição com as classes "Empresa" e "UsuarioCandidatura", já que ela só existe se houver uma empresa e um candidato na sua composição.

<**UsuarioCandidatura**>: Classe que representa a candidatura à uma vaga no sistema.

Essa classe possui um relacionamento de composição com as classes "Vaga" e "Usuario".

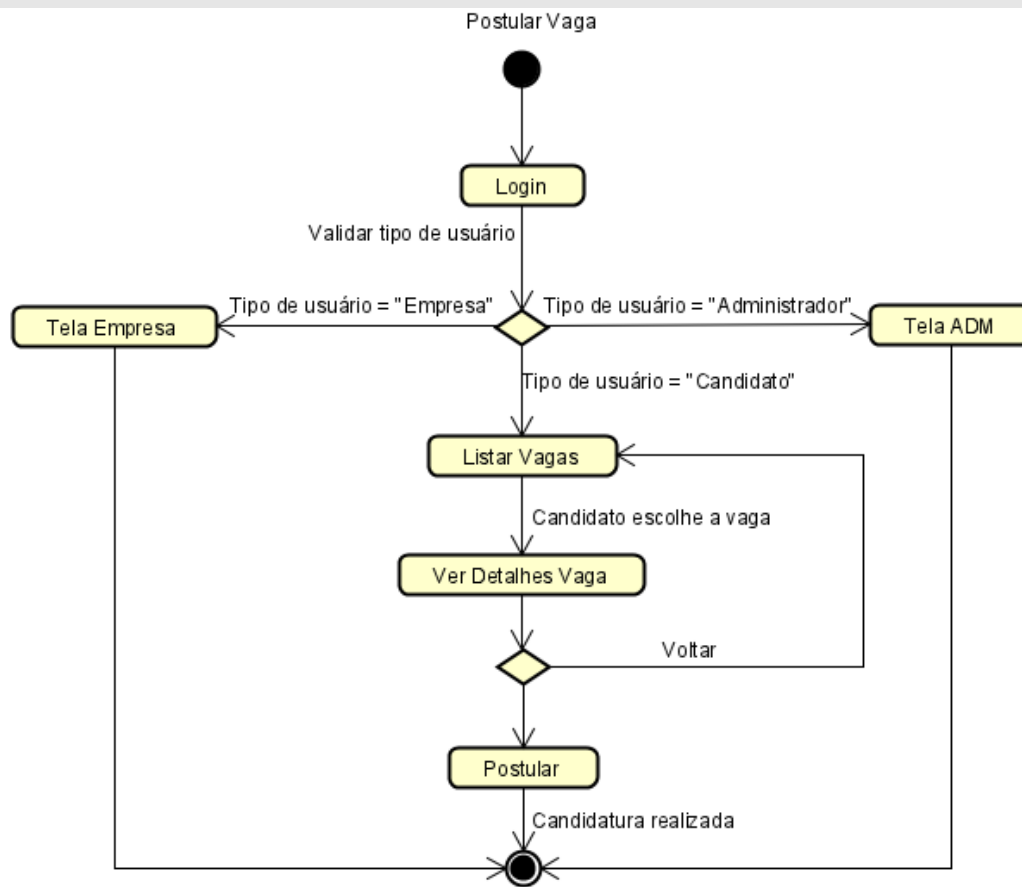
6 Diagrama de estado

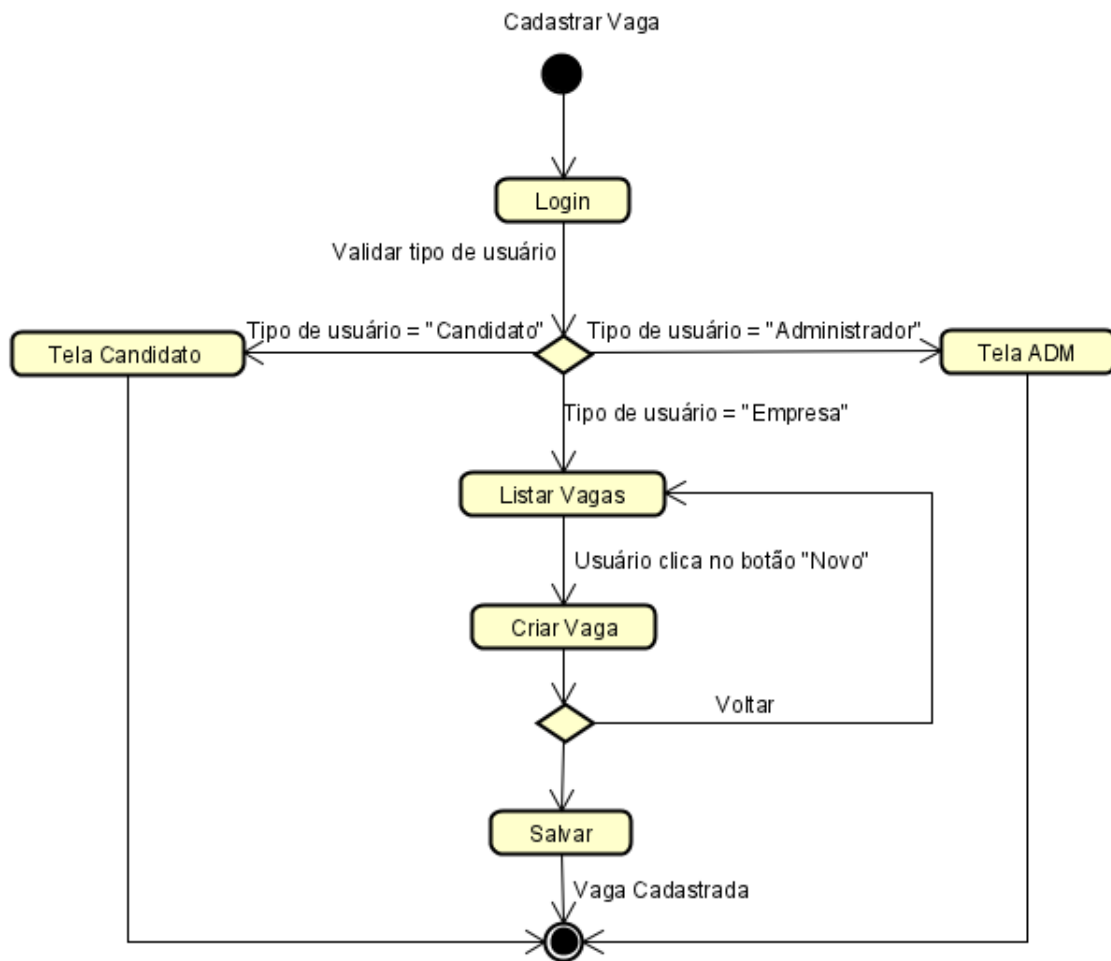
6.1 Diagrama de estado de Vaga



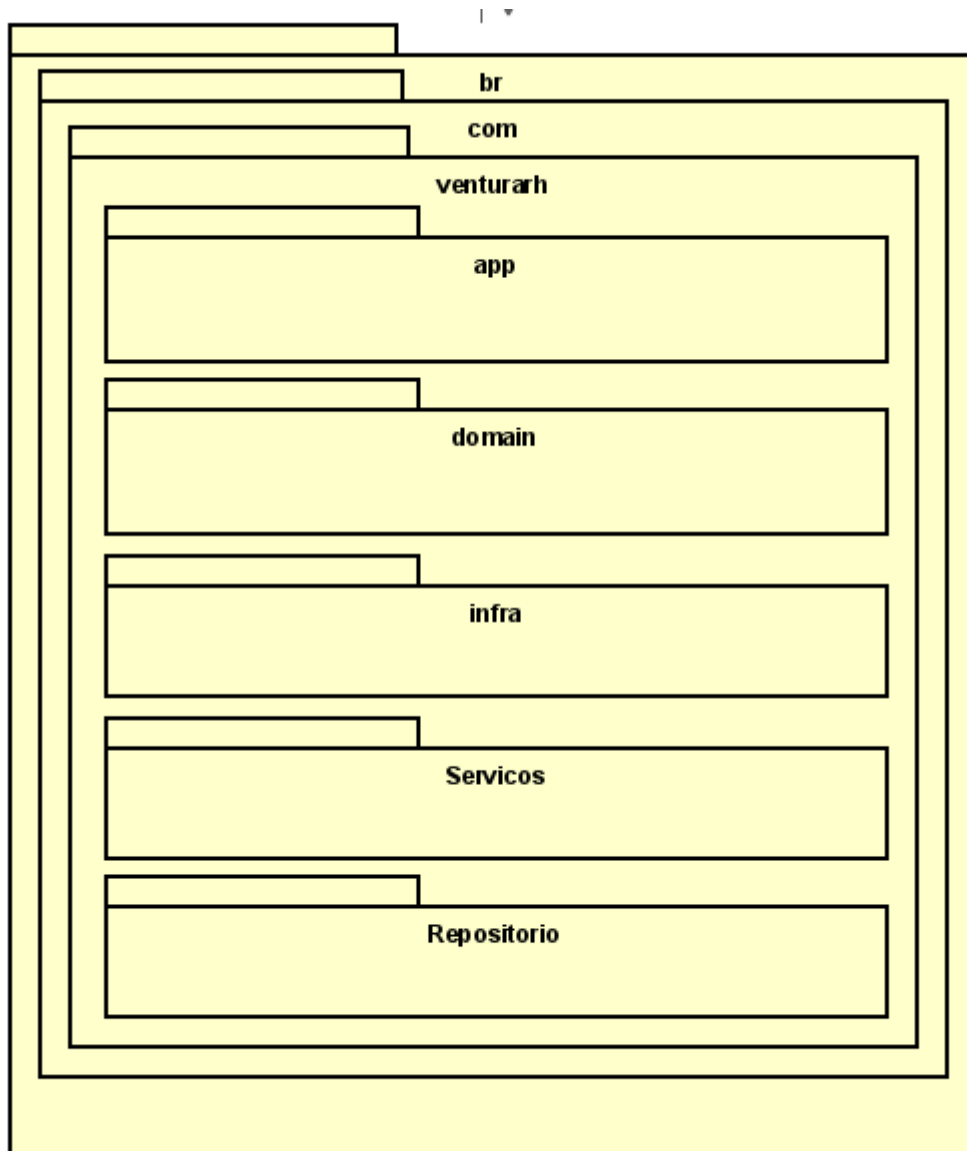
7 Diagrama de Atividade

7.1 Postular à vaga





8 Diagrama de Pacotes



Microserviços e endpoints

Empresa: Microserviço responsável pelo cadastro e listagem de empresa.

Usuario: Microserviço responsável pelo cadastro, login e listagem de usuário.

Vaga: Microserviço responsável pelo cadastro de novas vagas e postulações a vagas.

Endpoints

Empresa:

Endereço	Descrição	Parâmetros
GET /empresa/{id}	Localiza uma empresa por Id	Id é do tipo inteiro
POST /empresa/	Cadastra uma nova empresa	Json completo da classe Empresa

Vaga:

Endereço	Descrição	Parâmetros
GET /vaga/{id}	Localiza uma vaga por Id	Id é do tipo inteiro
POST /vaga	Cadastra uma nova vaga	Json completo da classe Vaga
GET /vaga	Lista todas as vagas	
PUT /vaga	Altera os dados de uma vaga	Json completo da classe Vaga
DELETE /vaga/{id}	Deleta uma vaga por Id	Id referente a vaga

Usuário:

Endereço	Descrição	Parâmetros
GET /usuario/{id}	Localiza um usuário por Id	Id é do tipo inteiro
POST /usuario/	Cadastra um novo usuário	Json completo da classe Usuario
POST /usuario/autenticar	Efetua o login do usuário validando e-mail e senha	Email: string Senha: String

Especificação

Os parâmetros que devem ser enviados nos métodos POST, devem ser enviados no Body da requisição e devem ser do tipo JSON.

Cada microserviço utiliza uma base de dados individual e hospedada no Amazon RDS.

Cada microserviço tem seu conjunto de domínio necessário para suas operações.

Toda e qualquer dependência de dados deve ser enviada para o microserviço para que evite dependência com outros serviços.

Serviços criados no AWS

Amazon RDS

Dashboard

Databases

Performance Insights

Snapshots

Automated backups

Reserved instances

Subnet groups

Parameter groups

Option groups

Events

Event subscriptions

Recommendations

Certificate update

RDS > Databases > venturahr-db

venturahr-db

Modificar

At

Resumo

Identificador de banco de dados
venturahr-db

CPU
4.91%

Status
Disponível

Classe
db.t2.micro

Função

Atividade atual
0 Conexões

Mecanismo
MySQL Community

Região e AZ
sa-east-1c

Segurança e conexão

Monitoramento

Logs e eventos

Configuração

Manutenção e backups

Tags

Segurança e conexão

Endpoint e porta

Endpoint
venturahr-db.cegh8wcbjam.sa-east-1.rds.amazonaws.com

Porta
3306

Redes

Zona de disponibilidade
sa-east-1c

VPC
vpc-05597b91672704a26

Grupo de sub-redes

Segurança

Grupo de segurança da VPC
default (sg-0e845d1a3450ab50d)
(ativo)

Publicamente acessível
Sim

Elastic Beanstalk

Ambientes

Aplicativos

Histórico de alterações

Elastic Beanstalk > Ambientes

Todos os ambientes

Filtrar resultados correspondentes aos valores de exibição

1

Nome do ambiente

Integridade

Nome do aplicativo

Data de criação

Última modificação

URL

Versões em execução

Plataforma

Estado da plataforma

Nome do nível

Venturahrempresa-env

0%

Venturahr-Empresa

06-10-2021 19:01:22 UTC-0300

06-10-2021 19:06:22 UTC-0300

Venturahrempresa-env.eba-6dqwep7.sa-east-1.elasticbeanstalk.com

Sample Application

Correto 11 running on 64bit Amazon Linux 2

Supported

WebServer

Venturahrusuario-env

0%

Venturahr-Usuario

06-10-2021 19:01:57 UTC-0300

06-10-2021 19:07:30 UTC-0300

Venturahrusuario-env.eba-cec2inew.sa-east-1.elasticbeanstalk.com

Sample Application

Correto 11 running on 64bit Amazon Linux 2

Supported

WebServer

Venturahrvagas-env

0%

Venturahr-Vagas

06-10-2021 19:02:24 UTC-0300

06-10-2021 19:07:11 UTC-0300

Venturahrvagas-env.eba-cblr28vcs.sa-east-1.elasticbeanstalk.com

Sample Application

Correto 11 running on 64bit Amazon Linux 2

Supported

WebServer

Venturahweb-env

0%

venturahr-web

06-10-2021 19:00:28 UTC-0300

06-10-2021 19:06:25 UTC-0300

Venturahweb-env.eba-mwksmm3p.sa-east-1.elasticbeanstalk.com

Sample Application

Correto 11 running on 64bit Amazon Linux 2

Supported

WebServer

venturahr-db.cegh8wcbjam.sa-east-1.rds.amazonaws.com - venturahr-db.cegh8wcbjam.sa-east-1.rds.amazonaws.com

Databases

dblnfnet

empresaDB

sys

vagaDb