

Instituto INFNET

ESTI – Escola Superior da Tecnologia da Informação



Projeto de Bloco: Desenvolvimento Java

Jennifer Omena

Professor: Armênio Torres Santiago Cardoso

Rio de Janeiro – RJ

Junho de 2022

<VenturaHR>

Documento de Requisitos
Versão 1.0

Histórico de Revisões

Data	Versão	Descrição	Autor
22/06/2022	1.0	Versão inicial do documento	Jennifer Omena

Índice

Sumário

1	Introdução	5
1.1	Propósito do documento de requisitos	5
1.2	Escopo do produto.....	5
1.3	Identificação dos Requisitos	5
1.4	Prioridade dos Requisitos	6
1.5	Visão Geral.....	6
2.	Descrição Geral	6
2.1	Usuários do Sistema	6
2.2	Solução Desejada.....	7
4.	Requisitos do Software	9
4.1	Requisitos Funcionais.....	9
4.2	Requisitos Não-Funcionais	10
4.2.1	Requisitos de Segurança.....	10
4.2.2	Requisitos Operacionais.....	11
4.2.3	Requisitos de Confiabilidade.....	11
5.	Casos de Uso	11
5.1	Diagrama de Casos de Uso	11
5.1.1	Usuário Candidato.....	12
5.1.2	Usuário Empresa	12
5.1.3	Administrador	13
5.2	Descrição dos Casos de Uso.....	13
5.2.1	Publicar Vaga	14
5.2.2	Responder Vaga	15
6.	Modelo de Domínio	18
7.	Diagramas de Classe.....	19
8.	Diagrama de Transição de Estados.....	20
9.	Diagramas de Entidade-Relacionamento	21
9.1	Diagrama Modelo Conceitual.....	21
9.2	Diagrama Modelo Lógico	21
9.3	Diagrama Modelo Físico.....	22
9.3.1	Código Gerado para o Modelo Físico	22

1 Introdução

1.1 Propósito do documento de requisitos

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

1.2 Escopo do produto

A fim de solucionar problemas antigos como a dificuldade de medir o retorno sobre o investimento (ROI) das ações, ineficiência dos recrutamentos e baixo desempenho em treinamentos, as HRTechs desenvolveram soluções tecnológicas para a área de Recursos Humanos, unindo inteligência e processos.

O projeto consiste na construção de um sistema cujo o objetivo é a recolocação dos profissionais de TI. Devido as peculiaridades desse ramo, os requisitos para contratação possuem um dinamismo vertiginoso, já que as tecnologias usadas pelas empresas estão constantemente evoluindo.

O sistema foi pensado para auxiliar empresas e profissionais de TI que precisam fazer e participar de processos seletivos, desde a criação da vaga pelo usuário Empresa até o contato com o candidato (usuário Candidato) interessado nela.

1.3 Identificação dos Requisitos

Cada requisito será unicamente identificado no formato [tipoRequisito.numero]. Para requisitos funcionais, o código do tipo de requisito será RF, e para requisitos não funcionais, RNF. Um número será assinalado a cada requisito de forma incremental, na ordem que forem mencionados neste documento.

1.4 Prioridade dos Requisitos

Foram adotadas as seguintes denominações para estabelecer a prioridade dos requisitos: essencial, importante e desejável.

- **Essencial:** é o requisito sem o qual o sistema não entra em funcionamento, ou seja, são requisitos imprescindíveis tendo que ser implementados impreterivelmente.
- **Importante:** é o requisito sem o qual o sistema entra em funcionamento, mas de maneira insatisfatória, ou seja, devem ser implementados, mas se não forem, o sistema poderá ser implantado e usado mesmo assim.
- **Desejável:** é o requisito que não compromete as funcionalidades básicas do sistema, podendo funcionar de forma satisfatória sem ele, ou seja, são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

1.5 Visão Geral

Este documento está organizado da seguinte forma:

- A seção 1 apresentou uma introdução ao documento de requisitos e ao sistema sendo especificado;
- A seção 2 apresenta uma descrição geral do sistema;
- A seção 3 apresenta a descrição textual do domínio, bem como sua camada de domínio e camada de apresentação;
- A seção 4 apresenta as definições dos requisitos funcionais e não-funcionais do sistema;
- A seção 5 apresenta o diagrama de casos de uso do sistema, bem como as descrições dos casos de uso definidos;
- A seção 6 apresenta o modelo de domínio;
- A seção 7 apresenta os diagramas de classe;
- A seção 8 apresenta o diagrama de transição de estados;
- A seção 9 apresenta os diagramas de entidade-relacionamento.

2. Descrição Geral

2.1 Usuários do Sistema

Usuários Candidato: Realizam as tarefas comuns, tal como: logar, buscar

vagas, compartilhar vagas e se aplicar a uma vaga.

Usuários Empresa: Realizam tarefas como criação de vaga, bem como editá-la, finaliza-la e renová-la e visualizar candidaturas.

Administradores: Responsáveis pelo gerenciamento de usuários Candidato e usuários Empresa, como bloqueio e desbloqueio de usuário.

2.2 Solução Desejada

O sistema VenturaHR precisa ter como diferencial o fato de não manter um banco de currículos e sim um banco de vagas x candidatos x critérios.

3. Descrição Textual do Domínio

3.1 Camada de Domínio

A Camada de Domínio é composta de:

- **UsuarioCandidato:** cadastro de nome completo, CPF, data de nascimento, e-mail, telefone e endereço completo.
- **UsuárioEmpresa:** cadastro de razão social, CNPJ, inscrição estadual, e-mail, telefone, endereço completo, nome da pessoa responsável em acessar o sistema, CPF da pessoa responsável em acessar o sistema, e-mail da pessoa responsável em acessar o sistema, telefone de contato da pessoa responsável em acessar o sistema.
- **Administrador:** cadastro de nome completo, CPF, data de nascimento, e-mail, telefone.
- **Vaga:** cadastro de nome da vaga, nome da empresa, local de trabalho (cidade ou remoto), tipo de vaga (estágio, junior, pleno, sênior, salário), quantidade de vagas, data de expiração e critérios de seleção (descrição, perfil mínimo desejado, peso dos critérios).

3.2 Camada de Apresentação

A Camada de Apresentação é composta de:

- **Home:** tela inicial com opção de cadastrar e logar, além de uma apresentação sobre o sistema, o que ele oferece, exposição de algumas vagas, um botão para ver todas as vagas, exposição de algumas empresas recrutadoras parceiras, um footer com a parte Institucional,

FAQ, termos e políticas e contatos.

- **Cadastro:** tela de login haverá opção de escolher se o usuário é candidato ou empresa. Escolhendo candidato ou empresa, o usuário deverá preencher o formulário de login que vai conter e-mail, senha, confirmação de senha. Também haverá um link com os termos de uso.
- **Login:** tela de login haverá opção de escolher se o usuário cadastrado é candidato ou empresa. Escolhendo candidato ou empresa, o usuário deverá preencher o formulário de login que vai conter e-mail, senha e um botão de “esqueci minha senha”.
- **AreaCandidato:** tela de AreaCandidato abrirá após o login do usuário candidato. Esta tela haverá um menu com Dashboard (AreaCandidato), Editar Perfil, Vagas e Candidaturas, e um botão de sair.
- **AreaEmpresa:** tela de AreaEmpresa abrirá após o login do usuário empresa. Esta tela haverá um menu com Dashboard (AreaEmpresa), Editar Perfil, Publicar Vaga, Vagas Publicadas, e um botão de sair.
- **EditarPerfil:** tela de edição de dados do usuário candidato ou usuário empresa. Será possível editar foto, nome completo ou razão social/nome fantasia, CPF/CNPJ, inscrição estadual (para usuário empresa), telefone de contato, endereço completo, data de nascimento (para usuário candidato), e-mail e um botão de salvar.
- **Vagas:** tela com lista de todas as vagas abertas, com nome da vaga, nome da empresa, local de trabalho (cidade ou remoto), tipo de vaga (estágio, junior, pleno, sênior, salário), quantidade de vagas, data de expiração, breve descrição da vaga e um botão para ver mais.
- **Candidaturas:** tela com lista de vagas aplicadas pelo UsuarioCandidato, podendo clicar sobre ela e saber os detalhes da vaga, e com um botão de candidatar.
- **PublicarVaga:** tela para cadastrar nova vaga. Haverá um formulário para o UsuarioEmpresa preencher com as informações da vaga. E um botão de publicar.
- **VagasPublicadas:** tela com lista de vagas publicadas pelo UsuarioEmpresa, podendo clicar sobre ela e saber mais detalhes da vaga e um botão para editar.
- **DetalheVaga:** tela que mostrará os detalhes da vaga seleciona na tela Vagas, Candidaturas e VagasPublicadas e um botão de voltar. O UsuarioEmpresa poderá ver a lista de candidatos inscritos para vaga clicando no botão Ver Candidatos. O UsuarioCandidato poderá clicar no botão para se candidatar.
- **EditarVaga:** tela de edição de vaga publicada pelo UsuarioEmpresa, onde ele poderá editar a data de expiração, renovar ou finalizar a vaga, e terá um botão de salvar edição.
- **Candidatos:** tela que mostrar uma lista de candidatos inscritos na vaga selecionada pelo UsuarioEmpresa. Essa lista vai conter a nota obtida nos critérios, bem como os dados do UsuarioCandidato.

4. Requisitos do Software

4.1 Requisitos Funcionais

<RF01>< Cadastrar no Sistema - Essencial>

O sistema deve permitir que o usuário possa criar uma conta no sistema. Esta conta poderá ser de Candidato ou Empresa. As contas de Administradores só poderão ser criadas por Administradores, e o sistema virá com uma conta de Administrador padrão. As contas de usuários não podem ser excluídas.

<RF02>< Editar Conta>

O sistema deve permitir que todo usuário possa editar seus dados na conta, bem como excluir sua conta.

<RF03>< Pesquisar Vagas - Essencial>

O sistema deve permitir que todos os usuários possam pesquisar vagas, porém somente Candidatos podem responder uma vaga.

<RF04>< Visualizar Estatísticas - Desejado>

O sistema deve permitir que Administradores possam visualizar estatísticas operacionais, como número de vagas criadas no dia corrente, número de vagas respondidas no dia corrente, número de usuários que criaram conta por tipo no dia corrente.

<RF05>< Bloquear Usuários - Desejado>

O sistema deve permitir que somente Administradores possam pesquisar usuários para bloquear ou desbloquear suas contas. No bloqueio, o Administrador deve informar a razão do bloqueio.

<RF06>< Notificar sobre Vagas a serem expiradas - Desejado>

Todos os dias, o sistema irá buscar vagas com expiração no dia corrente e irá notificar os usuários, para que estes tomem uma ação.

<RF07>< Deletar Vagas Expiradas - Desejado>

O sistema deverá fazer a exclusão de todas as vagas que têm data de expiração há dois dias, as finalizando automaticamente.

<RF08>< Publicação de Vaga - Essencial>

Os usuários com conta de Empresa podem publicar vagas com um conjunto de critérios de seleção, como descrição, perfil mínimo desejado e peso. O peso ordena os critérios entre si. A data de expiração não pode ser maior que 30 dias.

<RF09>< Listar Vagas - Essencial>

Os usuários com conta de Empresa podem listar suas vagas publicadas, mostrando dados da vaga e seu estado, como publicada, expirada ou

finalizada. Selecionando a vaga é possível ver a lista de critérios.

<RF10>< Consultar Usuários Inscritos para a Vaga - Essencial>

Os usuários com conta de Empresa, a partir de uma vaga publicada, pode consultar o ranking de candidatos que se aplicaram, ordenando pela nota obtida. Selecionando um candidato no ranking é possível ver seus dados de contato.

<RF11>< Renovar Vaga - Desejado>

Os usuários com conta de Empresa podem renovar uma vaga Publicada ou Expirada na lista de vagas publicadas por um período de até 30 dias.

<RF12>< Finalizar uma Vaga - Desejado>

Os usuários com conta de Empresa podem finalizar uma vaga na lista de vagas publicadas.

<RF13>< Editar Vaga>

Os usuários com conta de Empresa podem editar uma vaga desde que não haja candidatos.

<RF14>< Aplicar a uma Vaga - Essencial>

Os usuários com conta de Candidato podem se aplicar a uma vaga.

<RF15>< Listar Vagas Aplicadas>

Os usuários com conta de Candidato podem listar suas vagas respondidas. Essa lista mostra os dados da vaga e seu estado: Publicada, Expirada ou Finalizada.

<RF16>< Indicar Vaga>

Os usuários com conta de Candidato podem indicar vagas que tenham pesquisado, usando um e-mail para compartilhar a vaga.

4.2 Requisitos Não-Funcionais

Os requisitos que descrevem os aspectos não-funcionais do sistema são apresentados a seguir:

4.2.1 Requisitos de Segurança

<RNF/SEG-01>< Dar Manutenção>

O usuário autorizado deverá estar logado no sistema para poder realizar manutenção de cadastros de usuários e documentos.

3.2.2 Requisitos de Interface

<RNF/INT-01>< Interface Amigável - Importante>

O sistema deverá ter uma interface amigável, de fácil utilização, ou seja, auto

explicável.

4.2.2 Requisitos Operacionais

<RNF/OPE-01>< Linguagem - Essencial>

O sistema deverá ser desenvolvido na Linguagem Java.

<RNF/OPE-01>< Arquitetura - Essencial>

O sistema deverá ser desenvolvido na Arquitetura em Camadas.

<RNF/OPE-01>< Compatibilidade - Importante>

As camadas de aplicação para web do sistema deverão ser compatíveis com browsers do mercado, como Internet Explorer, Mozilla Firefox e Google Chrome.

4.2.3 Requisitos de Confiabilidade

<RNF/CONF-01>< Disponibilidade>

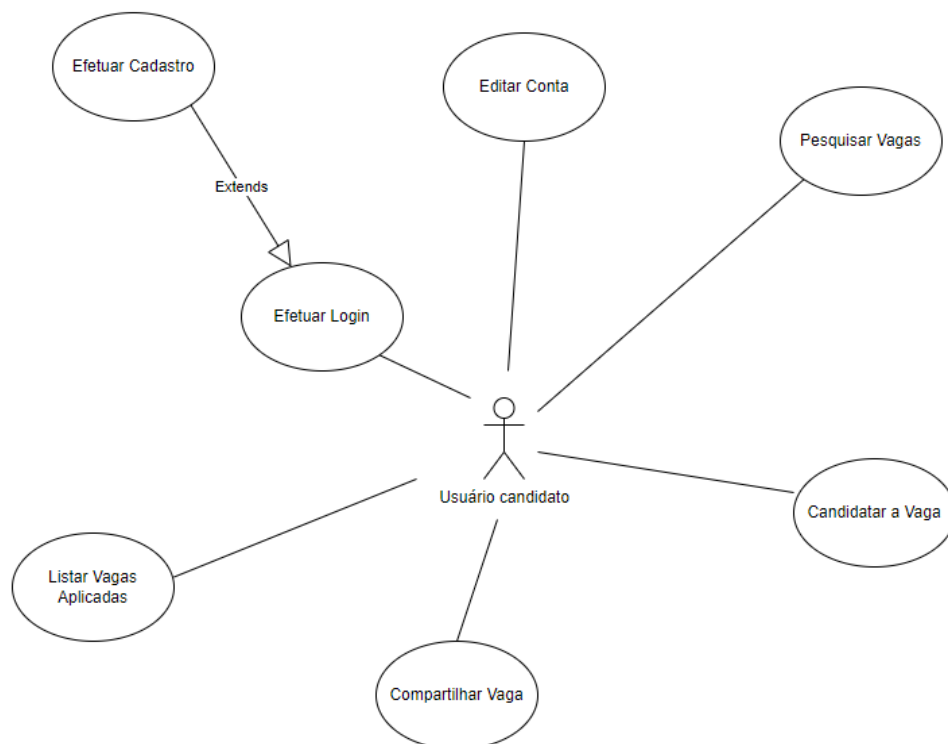
O sistema deverá estar disponível 24 horas por dia durante os 7 dias da semana. Por não se tratar de um sistema crítico, poderá ficar fora do ar até que seja corrigida alguma falha que possa ocorrer, desde que informe ao usuário o estado de manutenção e a previsão de retorno da normalidade.

5. Casos de Uso

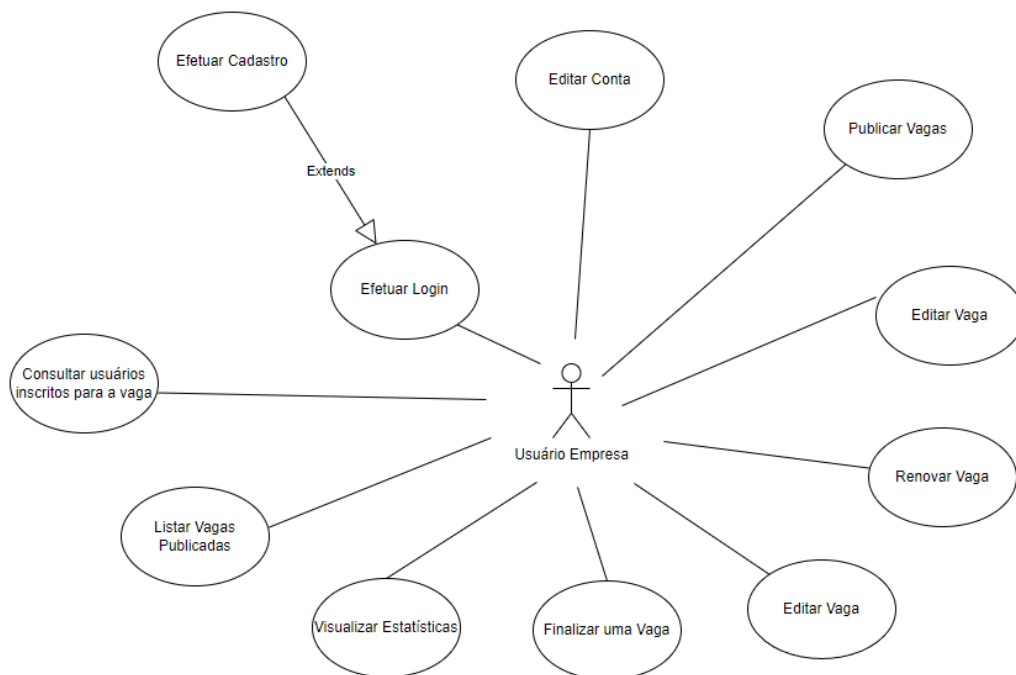
5.1 Diagrama de Casos de Uso

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.

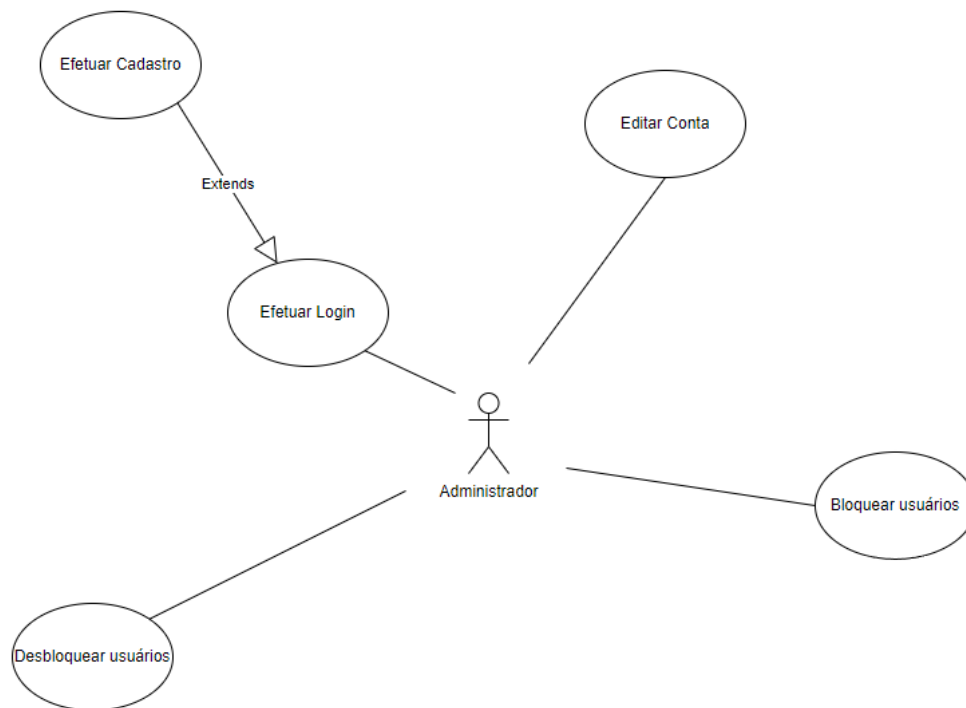
5.1.1 Usuário Candidato



5.1.2 Usuário Empresa



5.1.3 Administrador



5.2 Descrição dos Casos de Uso

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

Os usuários com conta de Empresa podem publicar vagas com um conjunto de critérios de seleção, como descrição, perfil mínimo desejado e peso. O peso ordena os critérios entre si. A data de expiração não pode ser maior que 30 dias.

5.2.1 Publicar Vaga

UC01 Publicar Vaga	
Atores	Usuário Empresa
Tipo	Primário
Pré-condições	O usuário empresa precisa ter uma conta ativa.
Pós-condições	A vaga é publicada.
Fluxo Básico	
<ol style="list-style-type: none">1. Usuário Empresa seleciona a opção “Publicar Vaga” na página inicial.2. Sistema apresenta formulário para inserção dos dados da vaga para publicação.3. Usuário Empresa insere os critérios de seleção, como descrição, perfil mínimo desejado e peso critério e clica em “Publicar”.4. Sistema informa que a vaga foi publicada com sucesso e redireciona para a página inicial.	
Fluxo Alternativo	
<ol style="list-style-type: none">3. Usuário Empresa não insere todos os campos obrigatórios e seleciona a opção para publicar<ol style="list-style-type: none">1. Sistema informa que há informações faltantes2. Usuário completa as informações faltantes e seleciona a opção para publicar3. Volta ao passo 4	

Os usuários com conta de Candidato podem responder vaga se candidatando a elas. Na tela com a lista de vagas, o usuário candidato poderá selecionar uma vaga em específico, ler mais sobre ela e, caso se interesse, poderá se candidatar.

5.2.2 Responder Vaga

UC02 Responder Vaga	
Atores	Usuário Candidato
Tipo	Primário
Pré-condições	O usuário candidato precisa ter uma conta ativa.
Pós-condições	O usuário candidato é aplicado à vaga.
Fluxo Básico	
<ol style="list-style-type: none">1. Usuário Candidato seleciona a opção “Vagas” na página inicial.2. Sistema mostra uma lista com todas as vagas abertas.3. Usuário Candidato seleciona uma vaga.4. Sistema mostra detalhes sobre a vaga selecionada.5. Usuário Candidato, após ler os requisitos da vaga selecionada, clica no botão “Cadidatar a esta vaga”.6. Sistema informa que a vaga foi aplicada com sucesso e redireciona a página de listagem de vagas.	
Fluxo Alternativo	
<ol style="list-style-type: none">5. Usuário Candidato não se qualifica nos requisitos para a vaga.<ol style="list-style-type: none">1. Usuário Candidato clica em voltar2. Volta ao passo 2	

Os usuários com conta de Empresa poderão editar as vagas publicadas, alterando a data de expiração, finalizando a vaga ou renovando a vaga.

5.2.3 Editar Vaga

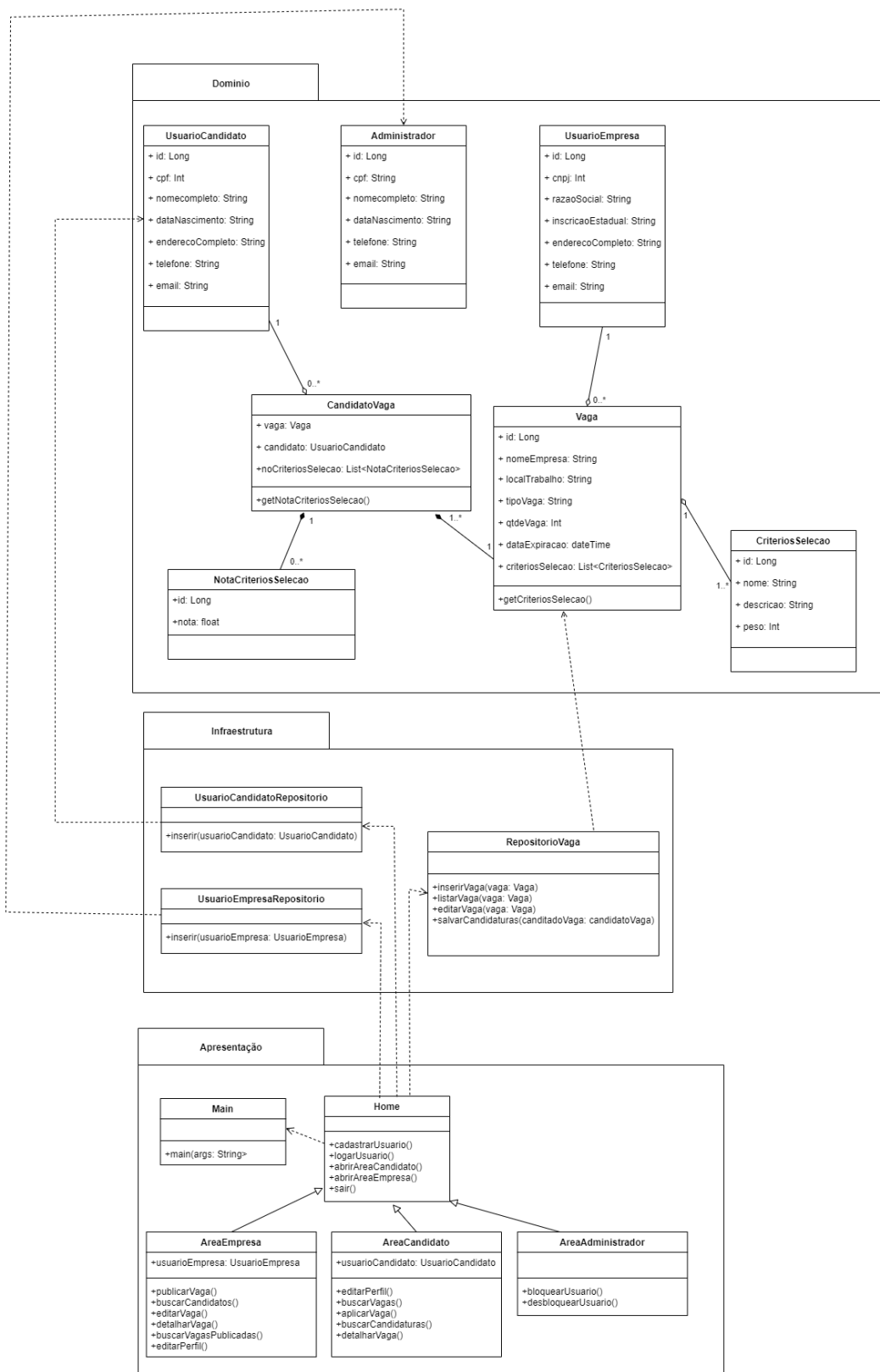
UC03 Editar Vaga	
Atores	Usuário Empresa
Tipo	Secundário
Pré-condições	O usuário empresa precisa ter uma conta ativa e ter uma vaga já publicada.
Pós-condições	A vaga terá sido atualizada.
Fluxo Básico	
<ol style="list-style-type: none">1. Usuário Empresa clica no botão “Editar” no card de uma vaga na lista de vagas publicadas.2. Sistema abre tela de Edição de Vaga com dados da vaga selecionada.3. Usuário Empresa altera a data de expiração da vaga e clica em “Salvar Alterações”.4. Sistema informa que a vaga foi atualizada e redireciona a página de listagem de vagas publicadas.	
Fluxo Alternativo	
<ol style="list-style-type: none">3. Usuário Empresa altera a data de expiração para mais de 30 dias.<ol style="list-style-type: none">1. Sistema informa que a data de expiração não pode ser superior a 30 dias.2. Usuário Empresa altera a data de expiração para menos de 30 dias e clica no botão de “Salvar alterações.”3. Volta ao passo 4.3. Usuário Empresa clica em finalizar vaga.<ol style="list-style-type: none">1. Sistema informa que a vaga foi finalizada.2. Volta ao passo 4.	

Os usuários com conta de Empresa podem ver os candidatos as vagas. Na tela com a lista de vagas publicadas, o usuário Empresa poderá selecionar uma vaga em específico, abrindo a tela de detalhes da vaga selecionada. Nessa tela, de detalhes da vaga, terá um botão de “Ver Candidatos”, ao clicar neste botão abrirá uma tela com a lista de candidatos inscritos na vaga selecionada, mostrando pontuação obtida nos critérios com os dados do candidato.

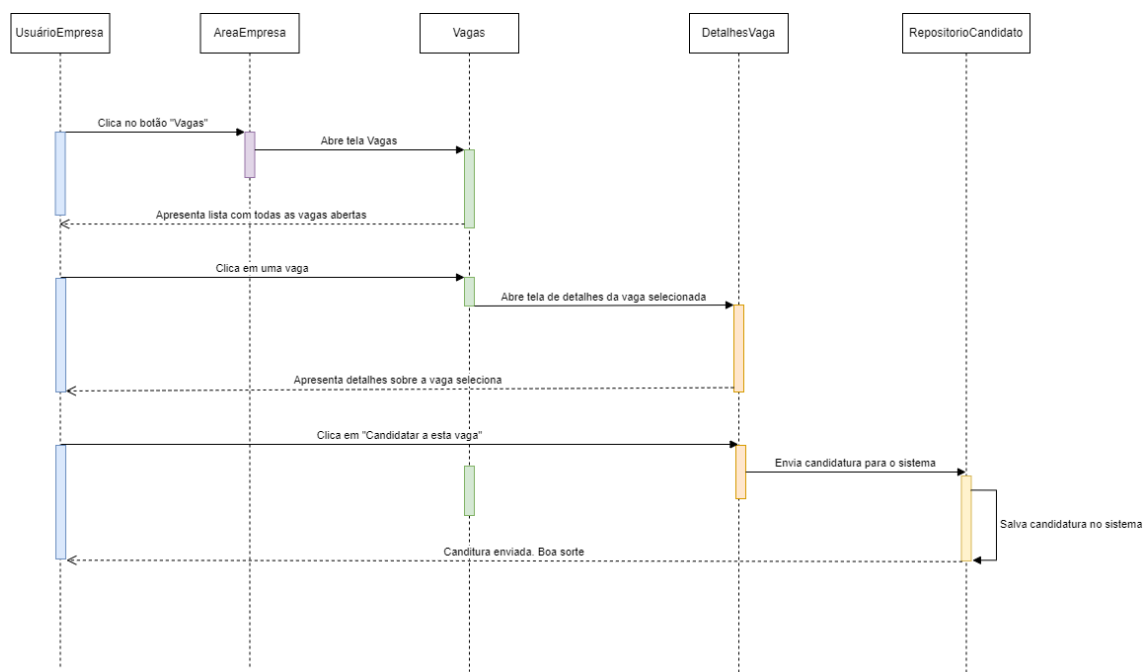
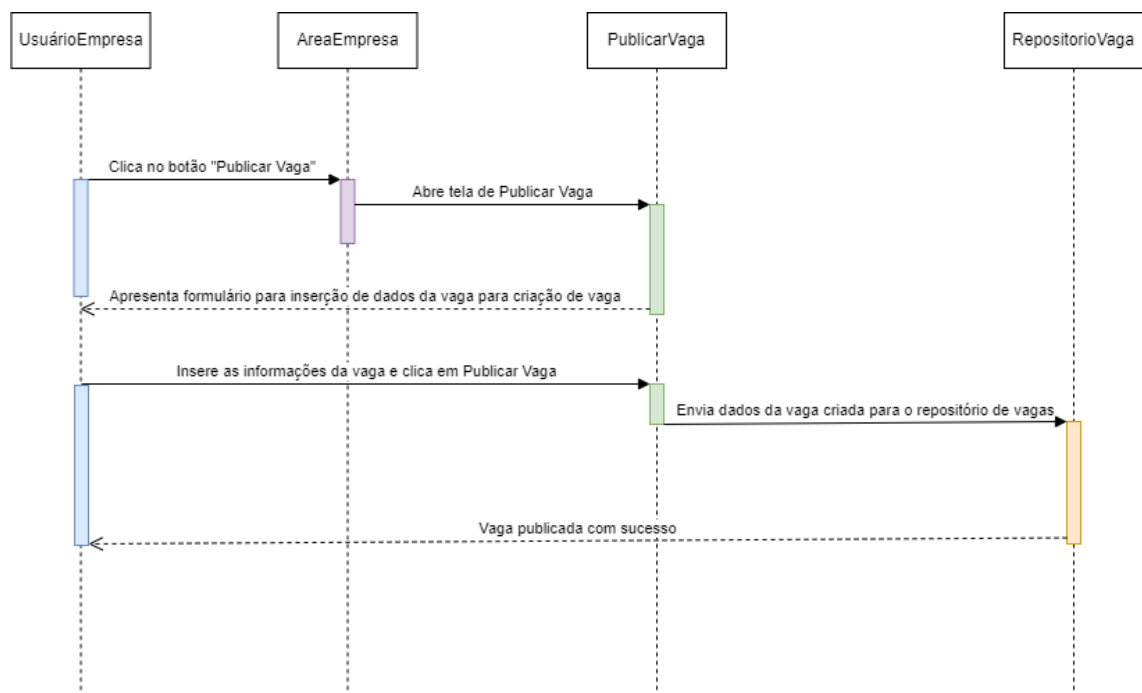
5.2.4 Listar Candidatos à Vaga

UC04 Listar Candidatos à Vaga	
Atores	Usuário Empresa
Tipo	Secundário
Pré-condições	O usuário empresa precisa ter uma conta ativa e uma vaga publicada.
Pós-condições	O usuário empresa terá acesso a lista de candidatos aplicados a vaga.
Fluxo Básico	
<ol style="list-style-type: none"> 1. Usuário Empresa seleciona uma vaga para ver detalhes dela. 2. Sistema mostra uma lista com todas as vagas publicadas por ele. 3. Usuário Empresa seleciona a opção “Ver Candidatos” na página de detalhes da vaga selecionada. 4. Sistema mostra uma lista com todas os candidatos que se aplicaram para a vaga selecionada. 	
Fluxo Alternativo	
<ol style="list-style-type: none"> 4. Sistema mostra uma lista vazia por não conter candidatos inscritos para a vaga. 1. Usuário Empresa clica em voltar, para retorna para a tela com a listagem de todas as vagas publicadas por ele 2. Volta ao passo 2 	

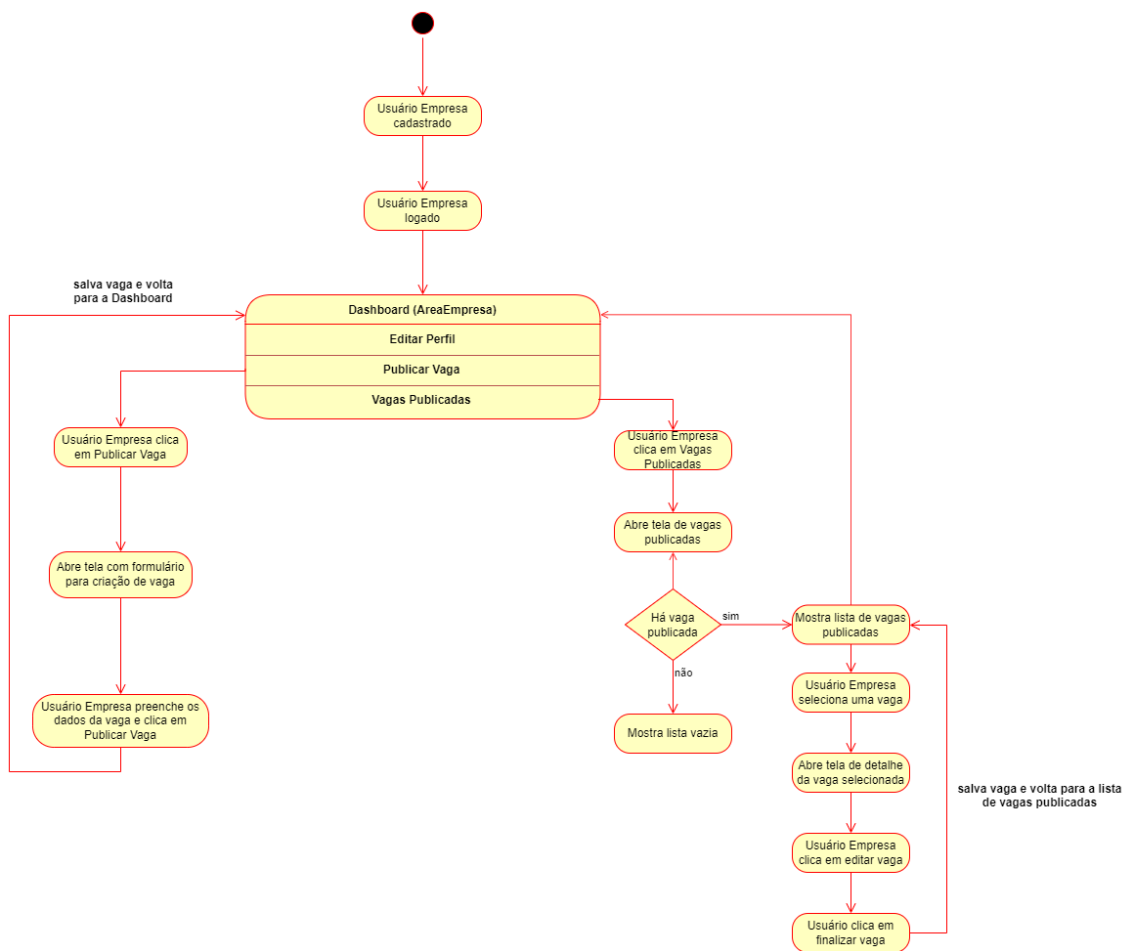
6. Modelo de Domínio



7. Diagramas de Classe

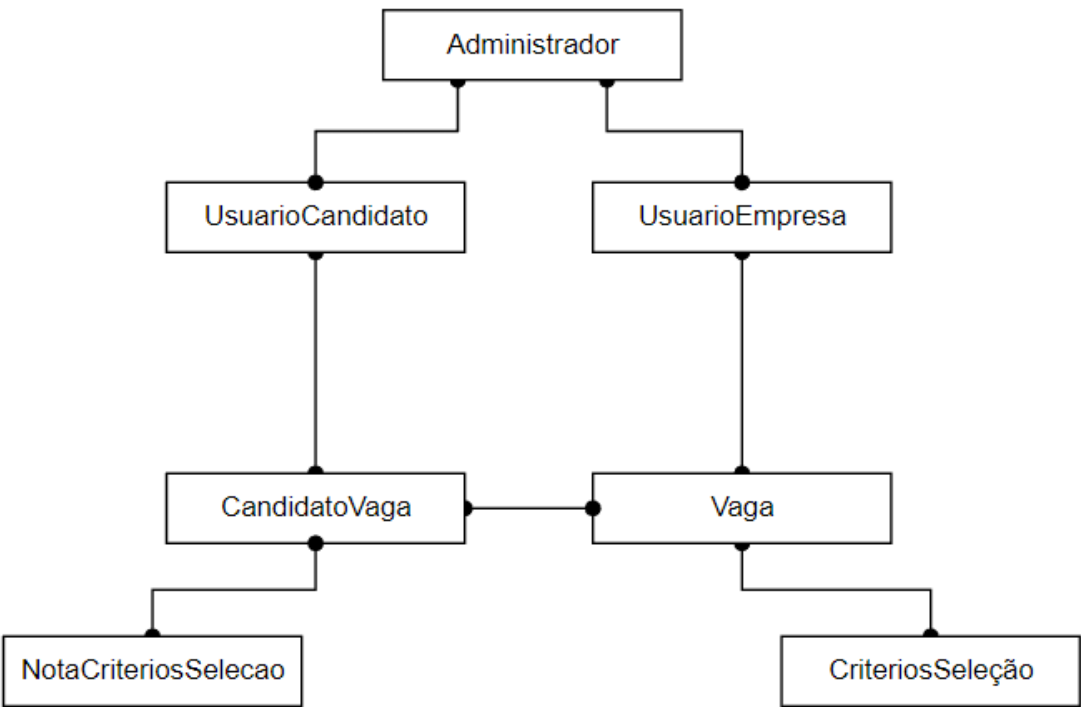


8. Diagrama de Transição de Estados

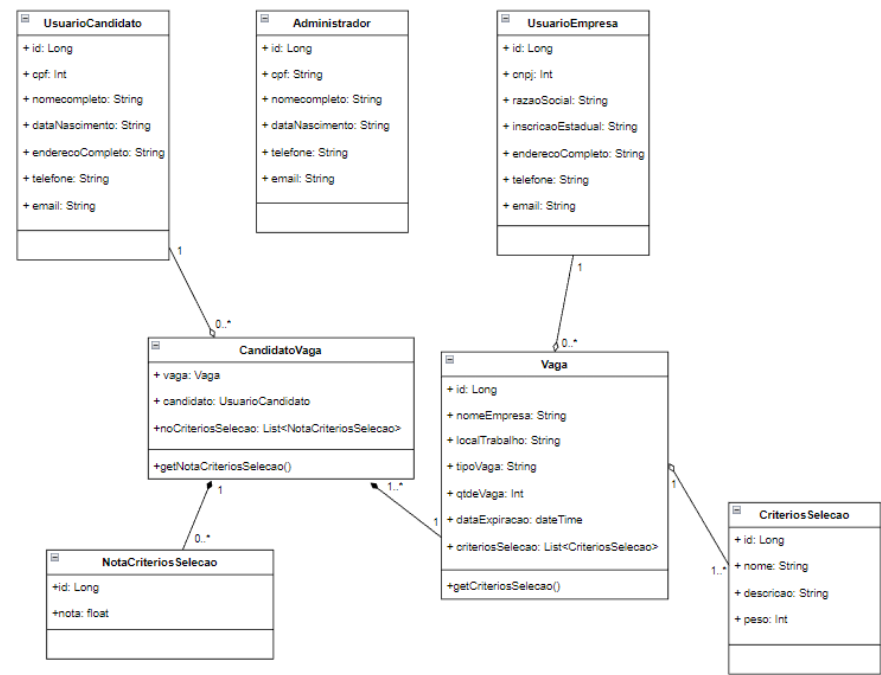


9. Diagramas de Entidade-Relacionamento

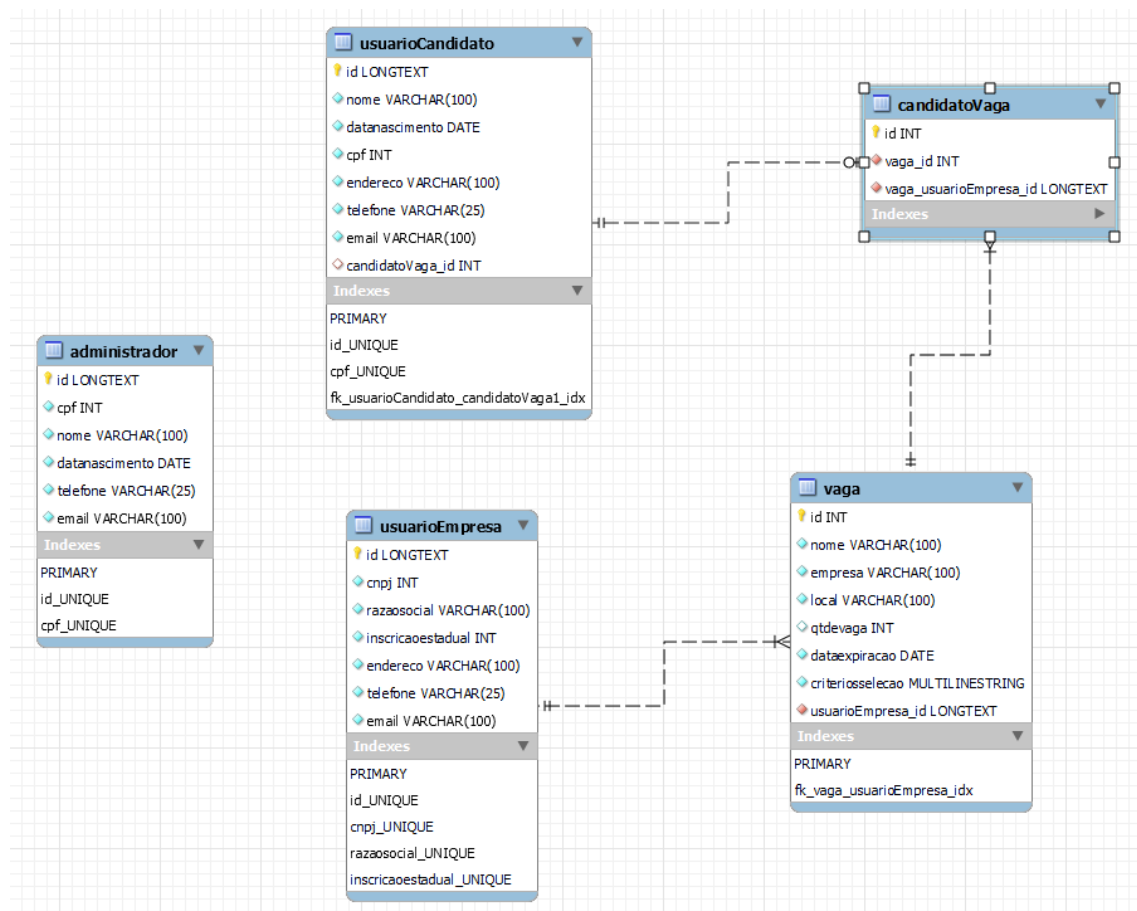
9.1 Diagrama Modelo Conceitual



9.2 Diagrama Modelo Lógico



9.3 Diagrama Modelo Físico



9.3.1 Código Gerado para o Modelo Físico

-- MySQL Workbench Forward Engineering

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE
,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

```
-- Schema mydb
```

```
-- Schema mydb
```

```
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 ;
USE `mydb` ;
```

```
-- Table `mydb`.`usuarioEmpresa`
```

```

-----
CREATE TABLE IF NOT EXISTS `mydb`.`usuarioEmpresa` (
  `id` LONGTEXT NOT NULL,
  `cnpj` INT NOT NULL,
  `razaosocial` VARCHAR(100) NOT NULL,
  `inscricaoestadual` INT NOT NULL,
  `endereco` VARCHAR(100) NOT NULL,
  `telefone` VARCHAR(25) NOT NULL,
  `email` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `id_UNIQUE` (`id` ASC) VISIBLE,
  UNIQUE INDEX `cnpj_UNIQUE` (`cnpj` ASC) VISIBLE,
  UNIQUE INDEX `razaosocial_UNIQUE` (`razaosocial` ASC) VISIBLE,
  UNIQUE INDEX `inscricaoestadual_UNIQUE` (`inscricaoestadual` ASC) VISIBLE)
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`vaga`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`vaga` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `nome` VARCHAR(100) NOT NULL,
  `empresa` VARCHAR(100) NOT NULL,
  `local` VARCHAR(100) NOT NULL,
  `qtdevaga` INT NULL,
  `dataexpiracao` DATE NOT NULL,
  `criteriosselecao` MULTILINESTRING NOT NULL,
  `usuarioEmpresa_id` LONGTEXT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_vaga_usuarioEmpresa_idx` (`usuarioEmpresa_id` ASC) VISIBLE,
  CONSTRAINT `fk_vaga_usuarioEmpresa`
    FOREIGN KEY (`usuarioEmpresa_id`)
      REFERENCES `mydb`.`usuarioEmpresa` (`id`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```

-----
-- Table `mydb`.`candidatoVaga`
-----

```

```

CREATE TABLE IF NOT EXISTS `mydb`.`candidatoVaga` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `vaga_id` INT NOT NULL,
  `vaga_usuarioEmpresa_id` LONGTEXT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_candidatoVaga_vaga1_idx` (`vaga_id` ASC, `vaga_usuarioEmpresa_id`
ASC) VISIBLE,
  CONSTRAINT `fk_candidatoVaga_vaga1`
    FOREIGN KEY (`vaga_id` , `vaga_usuarioEmpresa_id`)
      REFERENCES `mydb`.`vaga` (`id` , `usuarioEmpresa_id`)
      ON DELETE NO ACTION
      ON UPDATE NO ACTION)
ENGINE = InnoDB;

```

```
-- Table `mydb`.`usuarioCandidato`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`usuarioCandidato` (
  `id` LONGTEXT NOT NULL,
  `nome` VARCHAR(100) NOT NULL,
  `datanascimento` DATE NOT NULL,
  `cpf` INT NOT NULL,
  `endereco` VARCHAR(100) NOT NULL,
  `telefone` VARCHAR(25) NOT NULL,
  `email` VARCHAR(100) NOT NULL,
  `candidatoVaga_id` INT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `id_UNIQUE` (`id` ASC) VISIBLE,
  UNIQUE INDEX `cpf_UNIQUE` (`cpf` ASC) VISIBLE,
  INDEX `fk_usuarioCandidato_candidatoVaga1_idx` (`candidatoVaga_id` ASC)
  VISIBLE,
  CONSTRAINT `fk_usuarioCandidato_candidatoVaga1`
    FOREIGN KEY (`candidatoVaga_id`)
    REFERENCES `mydb`.`candidatoVaga` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;
```

```
-- Table `mydb`.`administrador`
-----
CREATE TABLE IF NOT EXISTS `mydb`.`administrador` (
  `id` LONGTEXT NOT NULL,
  `cpf` INT NOT NULL,
  `nome` VARCHAR(100) NOT NULL,
  `datanascimento` DATE NOT NULL,
  `telefone` VARCHAR(25) NOT NULL,
  `email` VARCHAR(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE INDEX `id_UNIQUE` (`id` ASC) VISIBLE,
  UNIQUE INDEX `cpf_UNIQUE` (`cpf` ASC) VISIBLE)
ENGINE = InnoDB;
```

```
SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```