

FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA

DEBORA RIBEIRO SILVA - RM 78297

CAIO LAURENTI BIANCHINI – RM 77850

JONATHAN VICTOR DE MELO SALU – RM 79563

MATEUS DE OLIVEIRA IGREJA – RM 77807

MAURÍCIO RODRIGUES OLIVEIRA – RM 79261

ENTREGÁVEL DE BANCO DE DADOS

São Paulo

Setembro/2018

DEBORA RIBEIRO SILVA - RM 78297
CAIO LAURENTI BIANCHINI – RM 77850
JONATHAN VICTOR DE MELO SALU – RM 79563
MATEUS DE OLIVEIRA IGREJA – RM 77807
MAURÍCIO RODRIGUES OLIVEIRA – RM 79261

ENTREGÁVEL DE BANCO DE DADOS

Trabalho apresentado como exigência parcial
para a composição da nota final da Avaliação
Multidisciplinar-AM da disciplina Design e
Desenvolvimento de Bancos de Dados sob a
orientação do Prof. MSc. Alexandre Barcelos

São Paulo
Setembro/2018

LISTA DE TABELAS

Tabela 1 - Legenda da tabela	14
Tabela 2 - Entidade: Candidato	15
Tabela 3 - Entidade: Arquivo	16
Tabela 4 - Entidade: Vaga	16
Tabela 5 - Entidade: Tag	17
Tabela 6 - Entidade: Tag_Vaga	17
Tabela 7 - Entidade: Tag_Candidato	18
Tabela 8 - Entidade: Departamento	18
Tabela 9 - Entidade: Inscricao	18
Tabela 10 - Entidade: Entrevista	19
Tabela 11 - Entidade: Prova	19
Tabela 12 - Entidade: Questão.....	19
Tabela 13 - Entidade: Vaga_Prova.....	20
Tabela 14 - Entidade: Tentativa.....	20
Tabela 15 - Entidade: Resposta_Questao	21
Tabela 16 - Entidade: Funcionario	21
Tabela 17 - Entidade: Arquivo	21
Tabela 18 - Entidade: Departamento	22
Tabela 19 - Entidade: Funcionario	22
Tabela 20 – Entidade: Inscricao	22
Tabela 21 - Entidade: Entrevista	22
Tabela 22 - Entidade: Tag	22
Tabela 23 - Entidade: Tag_Vaga	23
Tabela 24 - Entidade: Tag_Candidato	23
Tabela 25 - Entidade: Vaga_Prova.....	23
Tabela 26 - Entidade: Candidato	23
Tabela 27 - Entidade: Vaga	24
Tabela 28 - Entidade: Tentativa.....	24
Tabela 29 - Entidade: Prova	25
Tabela 30 - Entidade: Resposta_questao	25
Tabela 31 - Entidade: Questao.....	25

LISTA DE FIGURAS

Figura 1 - Chatbot do Facebook	6
Figura 2 - Gráfico de comparação de uso entre o Oracle e outros SGBD.....	12
Figura 3 – DER.....	13
Figura 4 - MER.....	26

SUMÁRIO

1 INTRODUÇÃO	6
1.1 Descrição do Problema.....	6
1.2 Objetivo	7
1.3 Escolha de SGBD	11
2 MODELO CONCEITUAL	13
2.1 Dicionário de dados	14
2.2 Teste de conceito	21
3 MODELO LÓGICO	26
4 MODELO FÍSICO	27
5 CONCLUSÃO	35

1 INTRODUÇÃO

Atualmente a Atento enfrenta um grande impasse em gerar e administrar vagas em seus portais de recrutamento, devido à enorme demanda de candidaturas para suas respectivas vagas. Foi apresentado um cenário no qual eles recebem 17.000 candidatos e apenas 17% são contratados.

1.1 Descrição do Problema

No atual sistema, o usuário pode se candidatar às vagas da empresa por meio de sites como: Facebook (Que disponibiliza um Chatbot, onde o candidato responde a perguntas seletivas e chega a uma determinada vaga por filtros como: cidade em que reside, escolaridade, etc (imagem 1).), Linkedin, Vagas.com.br, e seu próprio site.

Figura 1 - Chatbot do Facebook

The image shows a chatbot interface for Atento. It consists of three main steps, each with a question, a list of options, and a confirmation button.

- Step 1:** Question: "Você possui alguma deficiência?". Options: "Sim", "Não". Confirmation button: "Não".
- Step 2:** Question: "Já concluiu o ensino médio?". Options: "Sim", "Não". Confirmation button: "Sim".
- Step 3:** Question: "Em qual localidade você quer agendar o processo?". Options: "São Paulo", "Rio de Janeiro", "Belo Horizonte". Confirmation button: "São Paulo".

On the right side of the interface, there is a progress indicator showing "2/2" and a right arrow button.

Fonte: Facebook da Atento

Após o candidato submeter o interesse pela vaga, independentemente do meio que ele utilizou para se inscrever, o sistema gera um arquivo em formato .pdf contendo todas as informações do processo anterior. A partir deste momento, cabe ao analista de recrutamento

realizar manualmente a análise dos requisitos da vaga correspondente e comparar com as informações disponibilizadas pelo candidato.

Esta análise gera um grande gargalo no processo inteiro de contratação, pois é quando se descobre que a maioria dos interessados não estão aptos.

Quando há compatibilidade para a vaga, o RH envia o acesso do portal por e-mail. O portal provê ao candidato uma plataforma onde é possível realizar provas de conhecimento geral e/ou provas técnicas, isto é, ele pode realizar uma prova de aproximadamente 10 perguntas e logo em seguida já recebe um feedback. Caso aprovado, pode realizar o agendamento da entrevista final. Caso o candidato seja reprovado duas vezes no teste e ainda tem interesse pela vaga, o candidato deve contatar a Atento para marcar uma prova presencial.

1.2 Objetivo

A premissa da nova solução é obter um software que gerencie o processo de recrutamento ligando a pessoa certa para a vaga adequada, pois perde-se muito tempo fazendo esse processo manualmente tendo em vista a exorbitante quantidade de candidatos inscritos.

- Ele contemplará no seu escopo:
- Cadastro de departamento
- Cadastramento de vagas
- Cadastramento de candidatos
- Candidatar para vagas
- Cadastramento de provas
- Aplicação de provas online
- Seleção manual
- Gerenciar datas para entrevista
- Marcar entrevistas
- Destruir conta
- Finalizar processo de seleção
- Gerar relatórios
- Enviar e-mails com atualização do processo

Detalhamento dos requisitos (itens de desenvolvimento e entrega):

- **Cadastro de departamento** - Inclui o cadastro de departamento pelo Analista de RH com o nome e o funcionário responsável.
- **Cadastro de funcionário** - Inclui o cadastro de um funcionário com nome, sobrenome, email, senha para o acesso ao sistema e tipo de funcionário: Analista de RH, Recrutador ou outro.
- **Cadastramento de vaga** - Inclui o cadastro em si por um Analista de RH com título, quantidade de vagas, data desejada, status da vaga (Suspendido, Cancelado, Inativo, Preenchida), área de atuação, tags de habilidade necessárias, pesos das tags no score do candidato, local de atuação (Endereço do local de trabalho com CEP, estado e país), salário (não obrigatório), departamento, data de abertura, tempo experiência profissional, descrição da vaga, tipo (tempo integral, etc), requisitos, benefícios e quais provas devem ser realizadas.

Se uma nova vaga está sendo criada deve existir a opção de copiar os dados de uma vaga já encerrada para os campos da nova.

Será possível escolher um número específico de questões de cada prova que o analista selecionará para compor a prova unificada do candidato.

Cada componente integrante do score (link do Facebook, nº de amigos, link do Twitter, nº de seguidores, link do LinkedIn, nº de conexões e link do vídeo do YouTube) terá um peso selecionada pelo analista de RH na hora de criar a vaga, (1 - 50 pontos, 2 - 40 pontos, 3 - 30 pontos, 4 - 20 pontos, 5 - 10 pontos).

O Analista tem a opção de publicar vaga (e enviar para o painel do site) ou salvar como rascunho (status inativo). Após publicar é possível alterar o status cancelar / suspender / concluir o processo de recrutamento.

- **Cadastramento de candidato** - O candidato deve poder se cadastrar no sistema, ele terá três etapas de cadastro, uma inicial e obrigatória com os dados: email, nome, sobrenome, telefone, celular, senha e confirmação de senha. Não haverá duas contas com o mesmo endereço de e-mail e deverá haver uma confirmação do e-mail informado.

Ao candidatar-se a uma vaga será obrigatório preencher a segunda e terceira etapa com endereço (CEP, Rua, Cidade, Estado, País), data de nascimento, Anos de experiência, cargo atual, pretensão salarial, informações adicionais (Facebook, Twitter, LinkedIn (URL, nº de amigos / seguidores e se usa com frequência)). O candidato terá a opção de anexar um link do youtube de um vídeo de até 1 minuto para uma apresentação pessoal. Na terceira etapa deve ser possível fazer upload do currículo e carta de recomendação/apresentação nos formatos PDF ou DOCX.

- **Candidatar para vaga** - O sistema deve permitir se candidatar às vagas por meio de um painel eletrônico de vagas na aba trabalhe conosco do site com os dados: Título da vaga, área de atuação, requisitos, descrição, salário, benefícios, local de trabalho (Cidade e região), data desejada.

Após escolher uma vaga o candidato deverá realizar login/cadastro, atualizar seus dados se necessário e submeter sua intenção para vaga.

Ao se candidatar a uma vaga o candidato terá um score de compatibilidade.

Cada vaga terá tags de habilidade necessárias, o candidato selecionará na criação do perfil as que ele mais se identifica. Ao se recrutar em uma vaga, para cada tag que coincidir com as tags da vaga será somado uma quantidade específica de pontos ao score do candidato.

Ao preencher os campos: Link do vídeo, Links da Rede social (Facebook, Twitter, LinkedIn) também será somado pontos ao score. Para cada 50 amigos / seguidor na rede social o candidato ganhará uma quantidade de pontos, se o candidato informar que não utiliza a rede social a pontuação do score nesse quesito será desconsiderada.

O sistema deve comparar se o score das tags de habilidade é maior que 50% do total de tags. Caso o score seja menor que 50% do score total da vaga o candidato será automaticamente reprovado.

- **Cadastramento de prova** - O recrutador deve poder cadastrar provas no sistema para o candidato realizar eletronicamente. A prova tem um título e suas questões de múltipla escolha. A prova realizada pelo candidato terá o número de questões que estará cadastrado na vaga e serão aleatórias, geradas na hora de iniciar a prova.

O recrutador necessita cadastrar as questões de cada prova informando um título para exibição na área do recrutador, o texto da questão e as alternativas apontando a correta.

- **Aplicação de provas online** - O candidato pode realizar provas pelo portal após se candidatar a alguma vaga. De acordo com o perfil do candidato as provas podem ficar disponíveis instantaneamente para serem realizadas até a data desejada da vaga.

Deve ser armazenado o tempo que o candidato demorou para responder a prova e quantas vezes acessou o sistema. Isso ajuda a identificar tentativas de “cola”.

O score da prova será somado ao do perfil e caso o score seja menor que 50% do score total da vaga o candidato será automaticamente reprovado.

- **Seleção manual** - O recrutador deverá ter uma interface para ver os perfis dos candidatos cadastrados que já tenham realizado a prova e estejam aprovados nas provas, podendo filtrar por vaga específica, nome, status (Reprovado / Aguardando aprovação / Aprovado para entrevista / entrevista marcada / Aprovado para vaga), score mínimo e local de moradia.

Será possível ver todos os dados do candidato, incluindo as provas realizadas e os scores, currículo e outros documentos. Se o status for aguardando aprovação será possível aprovar o candidato para marcar entrevista ou reprovar.

Se o candidato estiver na fase de entrevista, haverá um botão no perfil para aprovação na vaga.

Nos perfis do candidato deve ter um campo de anotações para o recrutador realizar apontamentos úteis.

- **Gerenciar datas para entrevistas** - O recrutador poderá selecionar em um calendário datas para entrevistas e quais horários estão disponíveis por dia.

O calendário ficará disponível automaticamente para os candidatos aprovados. O recrutador poderá ver quais as entrevistas marcadas (data / hora / candidato / vaga).

- **Marcar entrevista** - Após ser aprovado na seleção manual o candidato poderá escolher uma das datas / horários disponíveis para realizar entrevista online ou

presencial, isso será definido na etapa de seleção manual ou posteriormente no perfil do candidato.

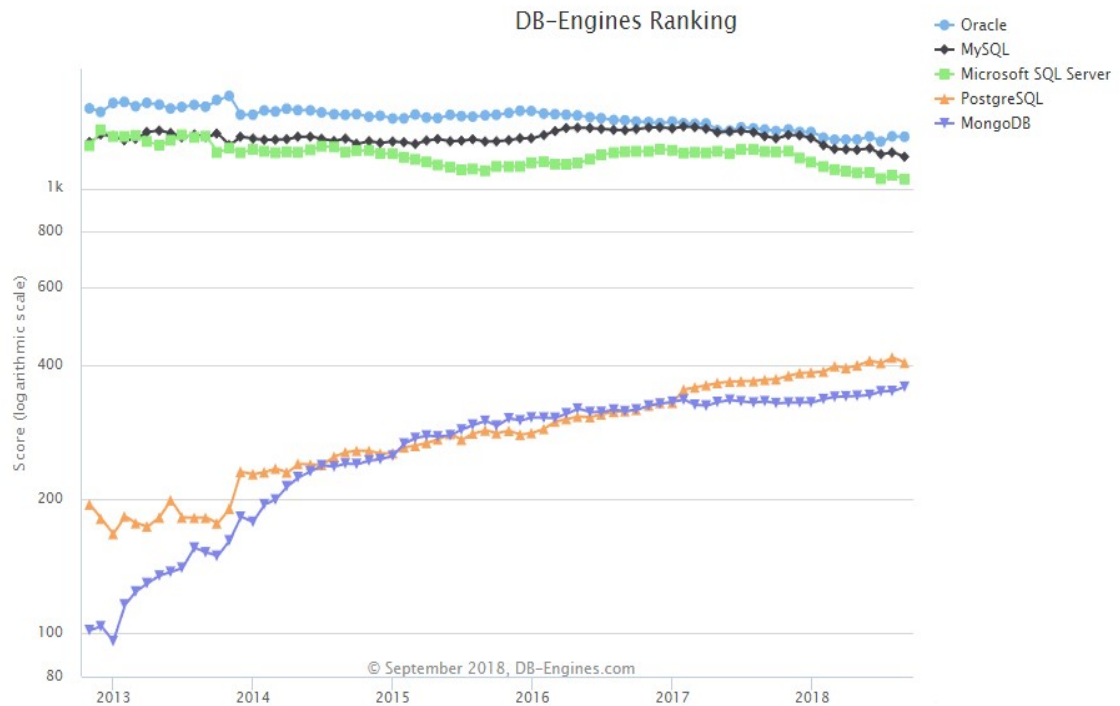
- **Destruir conta** - O candidato terá uma opção no seu perfil de destruir conta, apagando assim todos os dados e documentos do seu perfil.
- **Finalizar processo de seleção** - O analista de RH pode alterar o status da vaga para Preenchida e finalizar o processo de seleção.
- **Gerar relatórios** - O sistema deve possuir quatro relatórios: Um relatórios de todos candidatos por vaga com nome, e-mail e telefone dos candidatos; Um relatório de todos os candidatos aprovados para uma vaga com nome, e-mail e telefone dos candidatos; Um relatório de vagas abertas com título, data desejada, data de abertura, departamento e número de candidatos e um relatório de entrevistas com nome, e-mail e telefone dos candidatos junto com a data / hora da entrevista.
- **Enviar e-mails com atualização do processo** - A cada fase do processo deverá ser enviado um e-mail para o candidato. Os eventos são: Criar conta, Candidatar a vaga, Provas disponíveis, Disponibilidade para marcar entrevista e Entrevista marcada, Reprovado na seleção manual.

1.3 Escolha de SGBD

A solução de banco de dados escolhida é o Oracle pois o nosso software será utilizado por um grande número de pessoas na WEB e o SGBD precisa estar preparado para efetuar muitas operações de leitura e gravação simultaneamente. O Oracle tem a capacidade de lidar com essa alta carga de trabalho de forma eficiente e rápida, não deixando de lado a segurança dos dados.

O banco de dados Oracle (Oracle DB) é um sistema de gerenciamento de banco de dados relacional da Oracle Corporation. Originalmente desenvolvido em 1977 por Lawrence Ellison, o Oracle DB é um dos mecanismos de banco de dados relacionais mais confiáveis e amplamente usados.

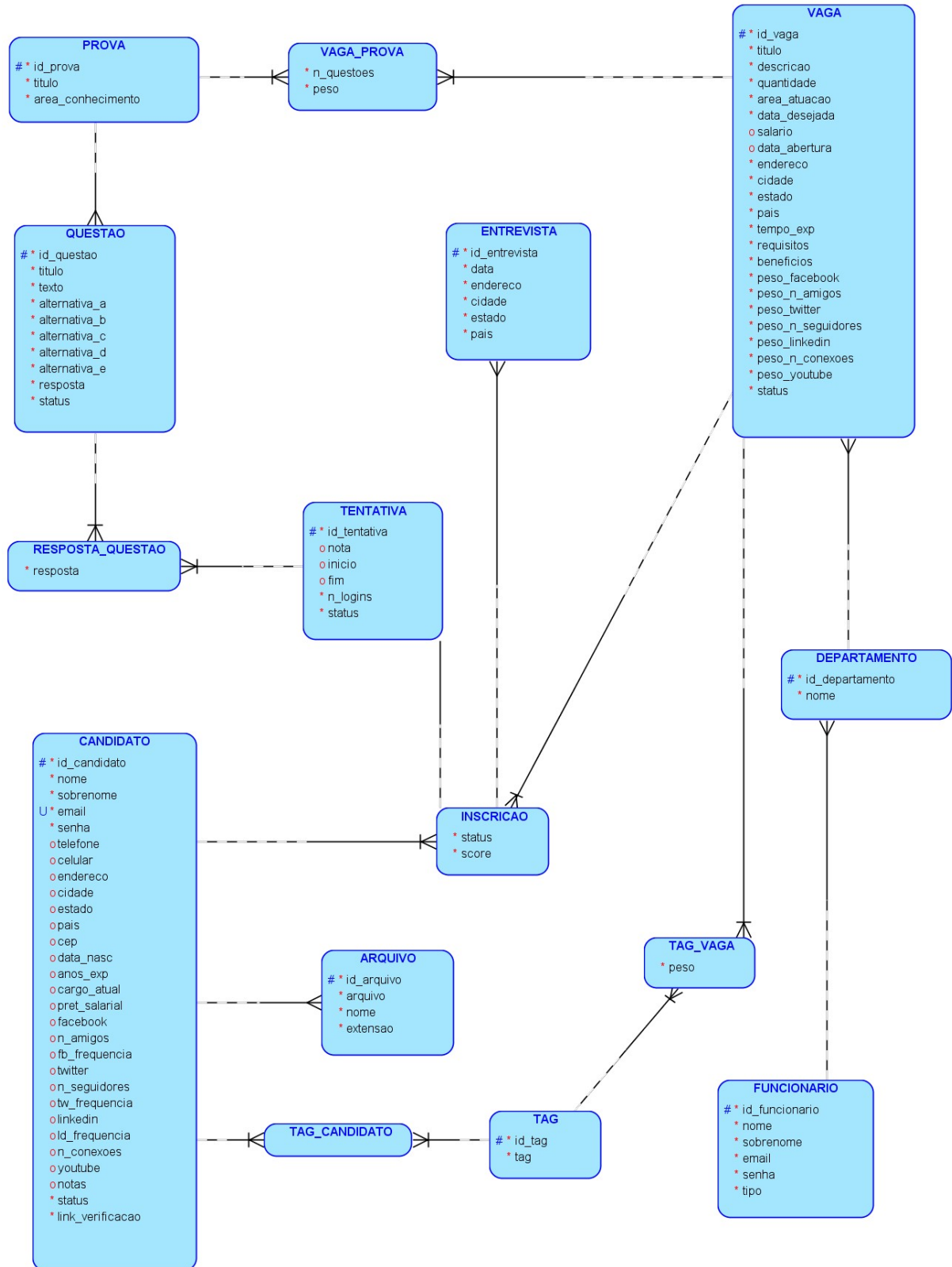
Figura 2 - Gráfico de comparação de uso entre o Oracle e outros SGBD



O sistema é construído em torno de uma estrutura de banco de dados relacional na qual os dados podem ser acessados diretamente pelos usuários (ou um front-end de aplicativo) por meio da linguagem de consulta estruturada (SQL).

2 MODELO CONCEITUAL

Figura 3 – DER



2.1 Dicionário de dados

Tabela 1 - Legenda da tabela

ENTIDADE	
Entidade/Tabela	Nome da Entidade
Descrição	Texto que descreve a entidade e os dados representados por ela, além de circunstâncias relevantes ao modelo de dados. Trata-se da descrição do conceito mais próximo da regra de negócio, sem preocupação com as características técnicas que serão implementadas no banco de dados.
ATRIBUTOS	
Atributo/Coluna	Nome do Atributo/Coluna
Tipo de Dado	Tipo de Dado que será armazenado (CHARACTER/NUMBER/INTEGER ETC.)
Tamanho	Precisão dos dados (quantidade de caracteres)
Restrição	Regras que deverão ser implementadas na criação das tabelas (PK-PRIMARY KEY/FK-FOREIGN KEY/U-UNIQUE/NN-NOT NULL/C-CHECK)
Descrição	Descrição do atributo
Constraint/Restrição	DESCRIÇÃO
PK	Primary Key-Chave Primária. Identifica de forma única e exclusiva uma ocorrência de uma entidade, ou, uma linha de uma tabela. Uma PK é sempre U/NN
FK	Foreign Key-Chave Estrangeira. É a coluna que permite a associação entre duas entidades ou entre ela mesma. Uma FK sempre irá referenciar uma PK.
U	Unique Key-Chave Única. Define que o conteúdo de uma coluna deve ser único/distinto/sem repetições.
NN	Not Null-Não Nulo. Define a obrigatoriedade de preenchimento de uma coluna.
C	Check-Validação. Valida os valores que serão armazenados em uma coluna por meio da definição de uma regra.

ENTIDADE				
Entidade/Tabela	CANDIDATO			
Descrição	Entidade que armazena todos os dados de um candidato que poderá no futuro participar de um processo de seleção e realizar provas.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_candidato	number	5	PK	Código de identificação do candidato
nome	varchar	30		Coluna que armazena o nome do candidato
location_id	number	5	FK	ID do local onde um departamento está localizado. Chave estrangeira para a coluna LOCATION_ID da tabela de LOCATIONS.

Tabela 2 - Entidade: Candidato

ENTIDADE				
Entidade/Tabela	CANDIDATO			
Descrição	Entidade que armazena todos os dados de um candidato que possa realizar login no sistema.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_candidato	number	5	PK	Código de identificação do candidato
nome	varchar	50	NN	Nome do candidato
sobrenome	varchar	50	NN	Sobrenome do candidato
email	varchar	60	NN, U	Email do candidato. Ela deve ser única pois será usada para realizar login.
senha	varchar	20	NN	Senha do candidato
telefone	varchar	16		Número de telefone do candidato.
celular	varchar	16		Número de celular do candidato.
endereco	varchar	100		Rua que o candidato mora
cidade	varchar	40		Nome da Cidade do candidato
estado	varchar	20		Nome do estado do candidato
pais	varchar	20		Nome do país do candidato
cep	varchar	20		Código postal da rua do candidato.
data_nasc	date			Data de nascimento do candidato
anos_exp	number	2		Número de anos de experiência profissional o candidato possui.
cargo_atual	varchar	50		Nome do cargo que o candidato ocupa atualmente em outra empresa.
pret_salarial	number	6,2		Valor da pretensão salarial que o candidato deseja.
facebook	varchar	50		Link para o perfil do facebook do candidato.
n_amigos	number	5		Número de amigos no facebook
fb_frequencia	number	1	C	Frequência de uso: 0 para nenhum uso, 1 para pouco uso, 2 para uso moderado e 3 para muito uso
twiter	varchar	30		Armazena o @nome do twitter do candidato
n_seguidores	number	5		Número de seguidores no twitter
tw_frequencia	number	1	C	Frequência de uso: 0 para nenhum uso, 1 para pouco uso, 2 para uso moderado e 3 para muito uso
linkedin	varchar	50		Link para o perfil do linkedin
n_conexoes	number	5		Número de conexões no linkedin
ld_frequencia	number	1	C	Frequência de uso: 0 para nenhum uso, 1 para pouco uso, 2 para uso moderado e 3 para muito uso
youtube	varchar	50		Link para o vídeo do youtube.
status	number	1	NN, C	Status da conta do candidato: 1 para Aguardando ativação, 3 para ativada e 2 para ativada com perfil incompleto.
link_verificacao	varchar	30	NN	Link que será usado para verificar o e-mail do candidato. Será comparado com o link acessado pelo candidato na hora da validação.

notas	varchar	1000		Texto qualquer que poderá ser informado pelo recrutador onde ele apontar informações que ele desejar.
-------	---------	------	--	---

Tabela 3 - Entidade: Arquivo

ENTIDADE				
Entidade/Tabela	ARQUIVO			
Descrição	Armazenar dados do arquivo que o candidato fez upload.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_arquivo	number	5	PK	Código de identificação do arquivo.
arquivo	varchar	200	NN	URL do arquivo
nome	varchar	100	NN	Nome do arquivo.
extensao	varchar	10	NN	Extensão do arquivo.
id candidato	number	5	FK/NN	Id do candidato que enviou o arquivo.

Tabela 4 - Entidade: Vaga

ENTIDADE				
Entidade/Tabela	VAGA			
Descrição	Entidade que armazena todos os dados de uma vaga.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_vaga	number	5	PK	Código de identificação da vaga
titulo	varchar	100	NN	Título curto da vaga
descricao	varchar	500	NN	Descrição da vaga detalhada.
quantidade	number	4	NN	Número de vagas disponíveis.
area_atuacao	varchar	30	NN	Nome da área da empresa que a vaga pertence
data_desejada	date		NN	Data final do processo de seleção
salario	number	6.2		Salário da vaga.
data_abertura	date			Data em que a vaga foi marcada como ativa.
endereco	varchar	100	NN	Rua da do local de trabalho da vaga.
cidade	varchar	40	NN	Nome da Cidade do local de trabalho da vaga.
estado	varchar	20	NN	Nome do estado do local de trabalho da vaga.
pais	varchar	20	NN	Nome do país do local de trabalho da vaga.
tempo_exp	number	3	NN	Tempo de experiência profissional que o candidato deve ter.
requisitos	varchar	500	NN	Lista de requisitos para a vaga.
beneficios	varchar	500	NN	Lista de benefícios para a vaga
peso_facebook	number	5	NN, C	Peso para cadastro de link do facebook no perfil do candidato: 1 para 50 pontos, 2 para 40 pontos,

				3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
peso_n_amigos	number	5	NN, C	Peso para cada 50 amigos no facebook do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
peso_twitter	number	5	NN, C	Peso para cadastro de link do twitter no perfil do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
peso_n_seguidores	number	5	NN, C	Peso para cada 50 seguidores no twitter do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
peso_linkedin	number	5	NN, C	Peso para cadastro de link do LinkedIn no perfil do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
peso_n_conexoes	number	5	NN, C	Peso para cada 50 conexões no LinkedIn do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
peso_youtube	number	5	NN, C	Peso para cadastro do link de um vídeo no youtube para apresentação pessoal: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
status	number	1	NN, C	Status da vaga: 1 para Preenchida, 2 para Ativa, 3 para Suspensa e 4 para Inativa.
id_departamento	number	5	FK/NN	Id do departamento que a vaga pertence.

Tabela 5 - Entidade: Tag

ENTIDADE				
Entidade/Tabela	TAG			
Descrição	Armazenar as tags da vaga			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_tag	number	5	PK	Código de identificação da tag.
tag	varchar	30	NN	Tag em si.

Tabela 6 - Entidade: Tag_Vaga

ENTIDADE	
Entidade/Tabela	TAG_VAGA
Descrição	Associa tags de habilidade com a vagas
ATRIBUTOS	

Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
peso	number	1	NN, C	Peso da tag no perfil do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
id_tag	number	5	FK, PK	Código de identificação da tag.
id_vaga	number	5	FK, PK	Código de identificação da vaga.

Tabela 7 - Entidade: Tag_Candidato

ENTIDADE				
Entidade/Tabela	TAG_CANDIDATO			
Descrição	Associa tags de habilidade com candidatos			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_tag	number	5	FK, PK	Código de identificação da tag.
id_candidato	number	5	FK, PK	Código de identificação do candidato.

Tabela 8 - Entidade: Departamento

ENTIDADE				
Entidade/Tabela	DEPARTAMENTO			
Descrição	Armazenar dados principais do departamento			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_departamento	number	5	PK	Código de identificação do departamento.
nome	varchar	50	NN	Nome do departamento.
id_funcionario	number	5	FK, NN	Id do funcionário responsável pelo departamento.

Tabela 9 - Entidade: Inscricao

ENTIDADE				
Entidade/Tabela	INSCRICAO			
Descrição	Guardar a inscrição de um candidato para uma vaga.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
status	number	1	NN, C	Status do candidato no processo de seleção: 1 para Verificando compatibilidade, 2 para Realizando provas, 3 para Aguardando seleção manual, 4 para Aprovado na seleção manual, 5 para Entrevista marcada, 6 para Aprovado e 7 para Reprovado

score	number	5	NN	Guarda o score total do candidato para a vaga específica.
id_candidato	number	5	PK/FK	Id do candidato associado.
id_vaga	number	5	PK/FK	Id da vaga associada.

Tabela 10 - Entidade: Entrevista

ENTIDADE				
Entidade/Tabela	ENTREVISTA			
Descrição	Guardar data/hora da entrevista e processo associado.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_entrevista	varchar	20	PK	Código de identificação da entrevista.
data	date		NN	Data/ hora para a entrevista.
endereco	varchar	100	NN	Endereço do local da entrevista.
cidade	varchar	40	NN	Cidade do local da entrevista.
estado	varchar	20	NN	Estado do local da entrevista.
pais	varchar	20	NN	País do local da entrevista.
id_candidato	number	5	FK, NN	Id do candidato associado.
id_vaga	number	5	FK, NN	Id da vaga associada.

Tabela 11 - Entidade: Prova

ENTIDADE				
Entidade/Tabela	PROVA			
Descrição	Guardar dados de uma prova.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_prova	varchar	20	PK	Código de identificação da prova.
título	varchar	100	NN	Título da prova
area_conhecimento	varchar	50	NN	Área do conhecimento que a prova pertence.

Tabela 12 - Entidade: Questão

ENTIDADE				
Entidade/Tabela	QUESTAO			
Descrição	Guardar alternativas de uma questão do tipo de múltipla escolha.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_questao	number	5	PK	Código de identificação da questão.

titulo	varchar	100	NN	Título qualquer para identificar a questão
texto	varchar	500	NN	Texto da questão.
alternativa_a	varchar	300	NN	Texto da alternativa.
alternativa_b	varchar	300	NN	Texto da alternativa.
alternativa_c	varchar	300	NN	Texto da alternativa.
alternativa_d	varchar	300	NN	Texto da alternativa.
alternativa_e	varchar	300	NN	Texto da alternativa.
resposta	char	1	NN	Alternativa correta.
id_prova	number	5	NN, FK	Id da prova que a questão faz parte
status	varchar	1	NN, C	Status da questão: 1 para ativa e 2 para inativa.

Tabela 13 - Entidade: Vaga_Prova

ENTIDADE				
Entidade/Tabela	VAGA_PROVA			
Descrição	Guardar as provas necessárias para se candidatar a uma vaga.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
n_questoes	number	5	NN	Número de questões que serão selecionadas da prova.
peso	number	1	NN,C	Peso das questões no perfil do candidato: 1 para 50 pontos, 2 para 40 pontos, 3 para 30 pontos, 4 para 20 pontos e 5 para 10 pontos
id_prova	number	5	FK/PK	Código de identificação da prova.
id_vaga	number	5	FK/PK	Código de identificação da vaga.

Tabela 14 - Entidade: Tentativa

ENTIDADE				
Entidade/Tabela	TENTATIVA			
Descrição	Guardar dados te uma tentativa de realização de prova pelo candidato.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_tentativa	number	5	PK	Código de identificação da tentativa.
nota	number	2		Nota que o candidato adquiriu na prova
inicio	date			Horário / data de início da prova
fim	date			Horário / data de fim da prova
n_logins	number	5	NN	Número de logins que o candidato realizou no sistema de provas.
id_candidato	number	5	FK, NN	Id do candidato associado.
id_vaga	number	5	FK, NN	Id da vaga associada.

status	number	1	NN, C	Status da tentativa: 1 para aberta, 2 para concluída e 3 para não realizada.
--------	--------	---	-------	--

Tabela 15 - Entidade: Resposta_Questao

ENTIDADE				
Entidade/Tabela	RESPOSTA_QUESTAO			
Descrição	Guardar dados da resposta que um candidato forneceu para uma questão.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_tentativa	number	5	FK/PK	Código de identificação da tentativa.
id_questao	number	5	FK/PK	Código de identificação da questão que está sendo respondida.
resposta	char	1	NN	Resposta informada pelo candidato (A, B, C, D ou E).

Tabela 16 - Entidade: Funcionario

ENTIDADE				
Entidade/Tabela	FUNCIONARIO			
Descrição	Armazenar dados de um funcionário.			
ATRIBUTOS				
Atributo/Coluna	Tipo de Dado	Tamanho	Restrição	Descrição
id_funcionario	number	5	PK	Código de identificação do funcionário
nome	varchar	50	NN	Nome do funcionário.
sobrenome	varchar	50	NN	Sobrenome do funcionário.
senha	varchar	20	NN	Senha do funcionário.
email	varchar	60	NN	E-mail do do funcionário.
tipo	varchar	5	NN, C	Tipo de conta: 1 para Analista RH, 2 para recrutador e 0 para outra.

2.2 Teste de conceito

Tabela 17 - Entidade: Arquivo

ARQUIVO				
ID_ARQUIVO	ARQUIVO	NOME	EXTENSAO	ID_CANDIDATO
5	b26e7488c2ac06e347556181001085912	curriculo	docx	1
6	b26e7488c2ac06e634750181001092117	apresentacao	docx	1

Tabela 18 - Entidade: Departamento

DEPARTAMENTO		
ID_DEPARTAMENTO	NOME	id_funcionario
2	Telemarketing	2
1	Vendas	3

Tabela 19 - Entidade: Funcionario

FUNCIONARIO					
ID_FUNCIONARIO	NOME	SOBRENOME	EMAIL	SENHA	TIPO
1	Mauricio	Rodrigues	mauricio@gmail.com	123	1
2	Jonathan	Salu	jonathan@gmail.com	456	0
3	Caio	Bianchini	caio@gmail.com	789	0

Tabela 20 – Entidade: Inscricao

INSCRICAO			
STATUS	ID_VAGA	ID_CANDIDATO	SCORE
1	1	1	1500
7	2	1	650

Tabela 21 - Entidade: Entrevista

ENTREVISTA							
ID_ENTREVISTA	DATA	ENDEREÇO	CIDADE	ESTADO	PAIS	ID_VAGA	ID_CANDIDATO
1	02/out	Rua X	São Paulo	São Paulo	Brasil	2	1

Tabela 22 - Entidade: Tag

TAG	
ID_TAG	TAG
1	Flexível
2	Comunicativo
3	Graduação
4	Ensino médio
5	Curso de administração
6	Vendedor
7	Habilitado CNH
8	Conhecimentos de computador

Tabela 23 - Entidade: Tag_Vaga

TAG_VAGA		
ID_TAG	ID_VAGA	PESO
3	1	3
9	1	4
7	2	5
6	2	4
5	2	4
3	2	5
2	2	2
8	2	1
1	1	1
2	1	2

Tabela 24 - Entidade: Tag_Candidato

TAG_CANDIDATO	
ID_TAG	ID_CANDIDATO
1	1
2	1
3	1
6	1
7	1
10	1
4	1

Tabela 25 - Entidade: Vaga_Prova

VAGA_PROVA			
ID_PROVA	ID_VAGA	N_QUESTIONES	PESO
1	1	5	1
1	2	5	3

Tabela 26 - Entidade: Candidato

CANDIDATO				
ID_CANDIDATO	NOME	SOBRENOME	EMAIL	SENHA
1	Mateus	Igreja	mateus@gmail.com	1485541
TELEFONE	CELULAR	ENDERECO	CIDADE	ESTADO
969898123	2134	Rua X	São paulo	São paulo
PAIS	CEP	DATA_NASC	ANOS_EXP	CARGO_ATUAL
Brasil	08210546	03/12/99	10	Analista

PRET_SALARIAL	FACEBOOK	TWITTER	LINKDIN	NOTAS
1000	http://fb.com/x	@mateus	http://linkedin.com/x	
STATUS	N_AMIGOS	FB_FREQUENCIA	N_SEGUIDORES	TW_FREQUENCIA
2	500	3	400	2
LD_FREQUENCIA	N_CONEXOES	YOUTUBE	LINK_VERIFICACAO	
1	577	yy	c4be1c05ed01a	

Tabela 27 - Entidade: Vaga

VAGA				
ID_VAGA	TITULO	DESCRICAO	QUANTIDADE	AREA_ATUACAO
1	Especialista de Remuneração Corporativo	Atuar em nossa estrutura de Recursos Humanos Corporativo/Global (13 países na América Latina, Europa e Estados Unidos).	12	Recursos humanos
2	Assistente de vendas I	Apresentar propostas de investimentos. Receber contatos para realizar os mesmos procedimentos.	3	Vendas
DATA_DESEJADA	SALARIO	DATA_ABERTURA	ENDEREÇO	CIDADE
25/09/18			Rua Y	São Paulo
25/09/18	1200	26/09/18	Rua Z	Guarulhos
ESTADO	PAIS	REQUISITOS	TEMPO_EXP	BENEFICIOS
São Paulo	Brasil	Graduação Superior	1	A combinar
São Paulo	Brasil	Ensino superior	5	A combinar
PESO_FACEBOOK	PESO_N_AMIGOS	PESO_TWITTER	PESO_N_SEGUIDORES	PESO_LINKEDIN
1	1	1	1	1
2	2	5	4	2
PESO_N_CONEXOES	PESO_YOUTUBE	STATUS	ID_DEPARTAMENTO	
1	2	1	1	
1	1	2	2	

Tabela 28 - Entidade: Tentativa

TENTATIVA							
ID_TENTATIVA	NOTA	INICIO	FIM	N_LOGINS	STATUS	ID_CANDIDATO	ID_VAGA
5	10	10/09/18	27/09/18	20	2	1	2
6	0	12/09/18	01/10/18	16	2	1	1

Tabela 29 - Entidade: Prova

PROVA		
ID_PROVA	TITULO	AREA_CONHECIMENTO
1	Conhecimentos gerais	Conhecimentos gerais

Tabela 30 - Entidade: Resposta_questao

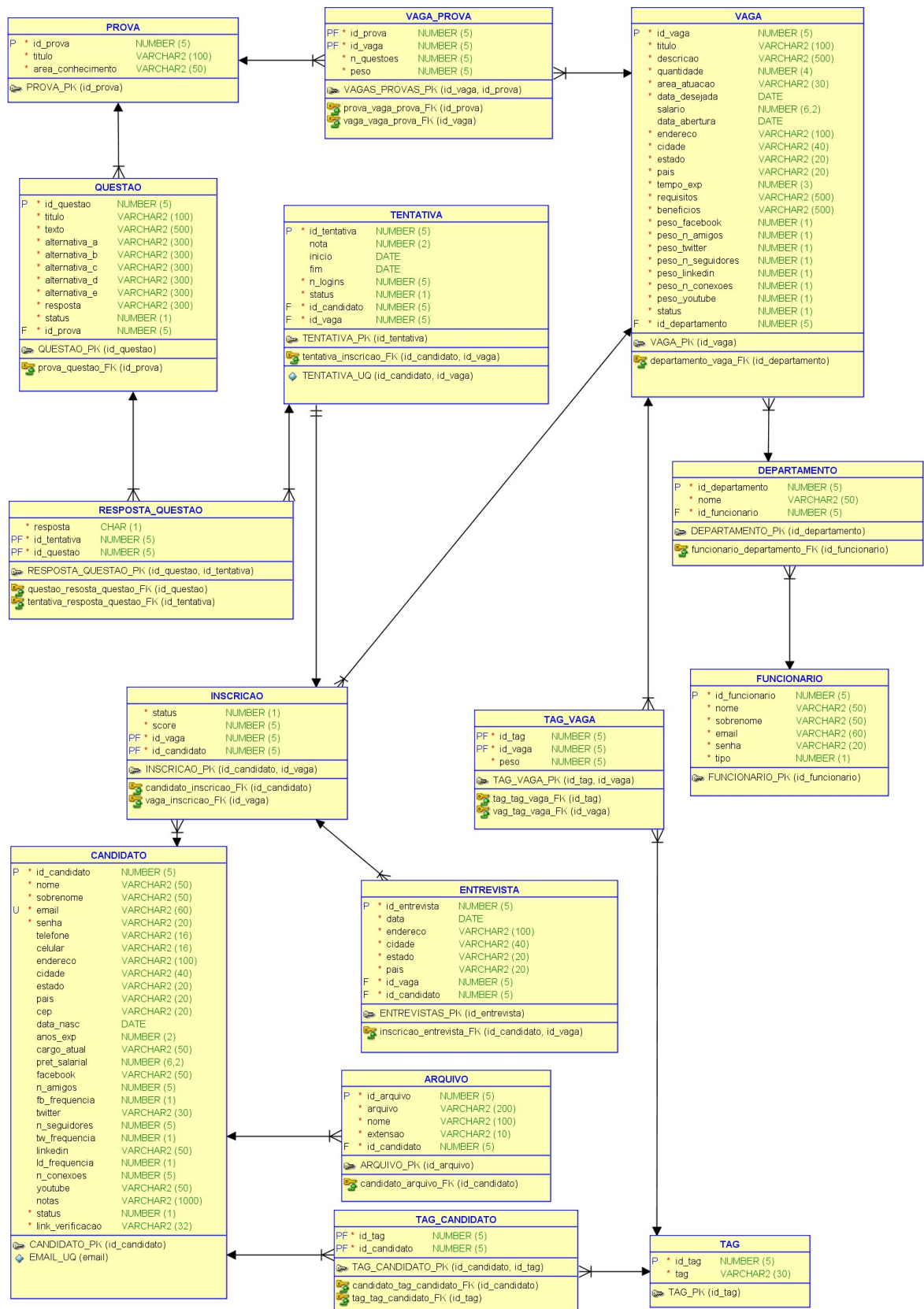
RESPOSTA_QUESTAO		
RESPOSTA	ID_TENTATIVA	ID_QUESTAO
a	6	1
c	6	2
c	6	3
d	6	4
e	6	5
a	6	6
e	6	7
b	6	8

Tabela 31 - Entidade: Questao

QUESTAO										
ID_QUESTAO	TITULO	TEXT O	ALTERNATIVA_A	ALTERNATIVA_B	ALTERNATIVA_C	ALTERNATIVA_D	ALTERNATIVA_E	RESPOSTA	STATUS	ID_PROVA
1	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
2	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
3	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
4	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
5	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
6	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
7	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1
8	Lorem	ad	Vestibulu	Ut	Aliquam	Nullam	Nullam	E	1	1

3 MODELO LÓGICO

Figura 4 - MER



4 MODELO FÍSICO

```
CREATE TABLE arquivo (
  id_arquivo      NUMBER(5) NOT NULL,
  arquivo         VARCHAR2(200) NOT NULL,
  nome            VARCHAR2(100) NOT NULL,
  extensao        VARCHAR2(10) NOT NULL,
  id_candidato    NUMBER(5) NOT NULL
);
```

```
ALTER TABLE arquivo ADD CONSTRAINT arquivo_pk PRIMARY KEY (
id_arquivo );
```

```
CREATE TABLE candidato (
  id_candidato    NUMBER(5) NOT NULL,
  nome            VARCHAR2(50) NOT NULL,
  sobrenome       VARCHAR2(50) NOT NULL,
  email           VARCHAR2(60) NOT NULL,
  senha           VARCHAR2(20) NOT NULL,
  telefone        VARCHAR2(16),
  celular         VARCHAR2(16),
  endereco        VARCHAR2(100),
  cidade          VARCHAR2(40),
  estado          VARCHAR2(20),
  pais            VARCHAR2(20),
  cep             VARCHAR2(20),
  data_nasc       DATE,
  anos_exp        NUMBER(2),
  cargo_atual     VARCHAR2(50),
  pret_salarial   NUMBER(6,2),
  facebook        VARCHAR2(50),
  n_amigos        NUMBER(5),
  fb_frequencia   NUMBER(1),
  twitter         VARCHAR2(30),
  n_seguidores    NUMBER(5),
  tw_frequencia   NUMBER(1),
  linkedin        VARCHAR2(50),
  ld_frequencia   NUMBER(1),
  n_conexoes      NUMBER(5),
  youtube         VARCHAR2(50),
  notas           VARCHAR2(1000),
  status          NUMBER(1) NOT NULL,
  link_verificacao VARCHAR2(32) NOT NULL
);
```

```
ALTER TABLE candidato
  ADD CONSTRAINT candidato_fb_frequencia_chk CHECK ( fb_frequencia
```

```
BETWEEN 0 AND 3 );
```

```
ALTER TABLE candidato
  ADD CONSTRAINT candidato_tw_frequencia_chk CHECK ( tw_frequencia
BETWEEN 0 AND 3 );
```

```
ALTER TABLE candidato
  ADD CONSTRAINT candidato_lk_frequencia_chk CHECK ( ld_frequencia
BETWEEN 0 AND 3 );
```

```
ALTER TABLE candidato
  ADD CONSTRAINT candidato_status_chk CHECK ( status BETWEEN 1 AND
3 );
```

```
ALTER TABLE candidato ADD CONSTRAINT candidato_pk PRIMARY KEY (
id_candidato );
```

```
ALTER TABLE candidato ADD CONSTRAINT email_uq UNIQUE ( email );
```

```
CREATE TABLE departamento (
  id_departamento  NUMBER(5) NOT NULL,
  nome              VARCHAR2(50) NOT NULL,
  id_funcionario    NUMBER(5) NOT NULL
);
```

```
ALTER TABLE departamento ADD CONSTRAINT departamento_pk PRIMARY KEY
( id_departamento );
```

```
CREATE TABLE entrevista (
  id_entrevista    NUMBER(5) NOT NULL,
  data             DATE NOT NULL,
  endereco         VARCHAR2(100) NOT NULL,
  cidade           VARCHAR2(40) NOT NULL,
  estado           VARCHAR2(20) NOT NULL,
  pais             VARCHAR2(20) NOT NULL,
  id_vaga          NUMBER(5) NOT NULL,
  id_candidato     NUMBER(5) NOT NULL
);
```

```
ALTER TABLE entrevista ADD CONSTRAINT entrevistas_pk PRIMARY KEY (
id_entrevista );
```

```
CREATE TABLE funcionario (
  id_funcionario  NUMBER(5) NOT NULL,
  nome            VARCHAR2(50) NOT NULL,
  sobrenome       VARCHAR2(50) NOT NULL,
  email           VARCHAR2(60) NOT NULL,
  senha           VARCHAR2(20) NOT NULL,
  tipo            NUMBER(1) NOT NULL
```

```

);

ALTER TABLE funcionario
    ADD CONSTRAINT funcionario_tipo_chk CHECK ( tipo BETWEEN 0 AND 2
);

ALTER TABLE funcionario ADD CONSTRAINT funcionario_pk PRIMARY KEY (
id_funcionario );

CREATE TABLE inscricao (
    status          NUMBER(1) NOT NULL,
    score           NUMBER(5) NOT NULL,
    id_vaga         NUMBER(5) NOT NULL,
    id_candidato    NUMBER(5) NOT NULL
);

ALTER TABLE inscricao
    ADD CONSTRAINT inscricao_status_chk CHECK ( status BETWEEN 1 AND
7 );

ALTER TABLE inscricao ADD CONSTRAINT inscricao_pk PRIMARY KEY (
id_candidato, id_vaga );

CREATE TABLE prova (
    id_prova        NUMBER(5) NOT NULL,
    titulo          VARCHAR2(100) NOT NULL,
    area_conhecimento VARCHAR2(50) NOT NULL
);

ALTER TABLE prova ADD CONSTRAINT prova_pk PRIMARY KEY ( id_prova );

CREATE TABLE questao (
    id_questao      NUMBER(5) NOT NULL,
    titulo          VARCHAR2(100) NOT NULL,
    texto           VARCHAR2(500) NOT NULL,
    alternativa_a   VARCHAR2(300) NOT NULL,
    alternativa_b   VARCHAR2(300) NOT NULL,
    alternativa_c   VARCHAR2(300) NOT NULL,
    alternativa_d   VARCHAR2(300) NOT NULL,
    alternativa_e   VARCHAR2(300) NOT NULL,
    resposta        VARCHAR2(300) NOT NULL,
    status          NUMBER(1) NOT NULL,
    id_prova        NUMBER(5) NOT NULL
);

ALTER TABLE questao ADD CONSTRAINT questao_pk PRIMARY KEY (
id_questao );

CREATE TABLE resposta_questao (

```

```

        resposta          CHAR(1) NOT NULL,
        id_tentativa       NUMBER(5) NOT NULL,
        id_questao         NUMBER(5) NOT NULL
    );

ALTER TABLE resposta_questao ADD CONSTRAINT resposta_questao_pk
PRIMARY KEY ( id_questao, id_tentativa );

CREATE TABLE tag (
    id_tag    NUMBER(5) NOT NULL,
    tag       VARCHAR2(30) NOT NULL
);

ALTER TABLE tag ADD CONSTRAINT tag_pk PRIMARY KEY ( id_tag );

CREATE TABLE tag_candidato (
    id_tag        NUMBER(5) NOT NULL,
    id_candidato  NUMBER(5) NOT NULL
);

ALTER TABLE tag_candidato ADD CONSTRAINT tag_candidato_pk PRIMARY
KEY ( id_candidato, id_tag );

CREATE TABLE tag_vaga (
    id_tag    NUMBER(5) NOT NULL,
    id_vaga   NUMBER(5) NOT NULL,
    peso      NUMBER(5) NOT NULL
);

ALTER TABLE tag_vaga
    ADD CONSTRAINT tag_vaga_peso_chk CHECK ( peso BETWEEN 1 AND 5 );

ALTER TABLE tag_vaga ADD CONSTRAINT tag_vaga_pk PRIMARY KEY (
id_tag, id_vaga );

CREATE TABLE tentativa (
    id_tentativa  NUMBER(5) NOT NULL,
    nota          NUMBER(2),
    inicio        DATE,
    fim           DATE,
    n_logins      NUMBER(5) NOT NULL,
    status        NUMBER(1) NOT NULL,
    id_candidato  NUMBER(5) NOT NULL,
    id_vaga       NUMBER(5) NOT NULL
);

ALTER TABLE tentativa
    ADD CONSTRAINT tentativa_status_chk CHECK ( status BETWEEN 1 AND
3 );

```

```
CREATE UNIQUE INDEX tentativa_uq ON
  tentativa (
    id_candidato
  ASC,
    id_vaga
  ASC );
```

```
ALTER TABLE tentativa ADD CONSTRAINT tentativa_pk PRIMARY KEY (
  id_tentativa );
```

```
CREATE TABLE vaga (
  id_vaga          NUMBER(5) NOT NULL,
  titulo           VARCHAR2(100) NOT NULL,
  descricao        VARCHAR2(500) NOT NULL,
  quantidade       NUMBER(4) NOT NULL,
  area_atuacao     VARCHAR2(30) NOT NULL,
  data_desejada    DATE NOT NULL,
  salario          NUMBER(6,2),
  data_abertura    DATE,
  endereco         VARCHAR2(100) NOT NULL,
  cidade           VARCHAR2(40) NOT NULL,
  estado           VARCHAR2(20) NOT NULL,
  pais             VARCHAR2(20) NOT NULL,
  tempo_exp        NUMBER(3) NOT NULL,
  requisitos       VARCHAR2(500) NOT NULL,
  beneficios       VARCHAR2(500) NOT NULL,
  peso_facebook    NUMBER(1) NOT NULL,
  peso_n_amigos    NUMBER(1) NOT NULL,
  peso_twitter     NUMBER(1) NOT NULL,
  peso_n_seguidores NUMBER(1) NOT NULL,
  peso_linkedin    NUMBER(1) NOT NULL,
  peso_n_conexoes  NUMBER(1) NOT NULL,
  peso_youtube     NUMBER(1) NOT NULL,
  status           NUMBER(1) DEFAULT 1 NOT NULL,
  id_departamento NUMBER(5) NOT NULL
);
```

```
ALTER TABLE vaga
  ADD CONSTRAINT peso_facebook_chk CHECK ( peso_facebook BETWEEN 1
  AND 5 );
```

```
ALTER TABLE vaga
  ADD CONSTRAINT peso_n_amigos_chk CHECK ( peso_n_amigos BETWEEN 1
  AND 5 );
```

```
ALTER TABLE vaga
  ADD CONSTRAINT peso_twitter_chk CHECK ( peso_twitter BETWEEN 1
  AND 5 );
```

```

ALTER TABLE vaga
    ADD CONSTRAINT peso_n_seguidores_chk CHECK ( peso_n_seguidores
BETWEEN 1 AND 5 );

ALTER TABLE vaga
    ADD CONSTRAINT peso_linkedin CHECK ( peso_linkedin BETWEEN 1 AND
5 );

ALTER TABLE vaga
    ADD CONSTRAINT peso_n_conexoes_chk CHECK ( peso_n_conexoes
BETWEEN 1 AND 5 );

ALTER TABLE vaga
    ADD CONSTRAINT peso_youtube_chk CHECK ( peso_youtube BETWEEN 1
AND 5 );

ALTER TABLE vaga
    ADD CONSTRAINT vaga_status_chk CHECK ( status BETWEEN 1 AND 4 );

ALTER TABLE vaga ADD CONSTRAINT vaga_pk PRIMARY KEY ( id_vaga );

CREATE TABLE vaga_prova (
    id_prova      NUMBER(5) NOT NULL,
    id_vaga       NUMBER(5) NOT NULL,
    n_questoes    NUMBER(5) NOT NULL,
    peso          NUMBER(5) NOT NULL
);

ALTER TABLE vaga_prova
    ADD CONSTRAINT vaga_prova_pesos_chk CHECK ( peso BETWEEN 1 AND 5
);

ALTER TABLE vaga_prova ADD CONSTRAINT vagas_provas_pk PRIMARY KEY (
id_vaga, id_prova );

ALTER TABLE arquivo
    ADD CONSTRAINT candidato_arquivo_fk FOREIGN KEY ( id_candidato )
REFERENCES candidato ( id_candidato );

ALTER TABLE inscricao
    ADD CONSTRAINT candidato_inscricao_fk FOREIGN KEY ( id_candidato
)
REFERENCES candidato ( id_candidato );

ALTER TABLE tag_candidato
    ADD CONSTRAINT candidato_tag_candidato_fk FOREIGN KEY (
id_candidato )
REFERENCES candidato ( id_candidato );

```



```

ALTER TABLE vaga
    ADD CONSTRAINT departamento_vaga_fk FOREIGN KEY (
id_departamento )
    REFERENCES departamento ( id_departamento );

ALTER TABLE departamento
    ADD CONSTRAINT funcionario_departamento_fk FOREIGN KEY (
id_funcionario )
    REFERENCES funcionario ( id_funcionario );

ALTER TABLE entrevista
    ADD CONSTRAINT inscricao_entrevista_fk FOREIGN KEY (
id_candidato, id_vaga )
    REFERENCES inscricao ( id_candidato, id_vaga );

ALTER TABLE questao
    ADD CONSTRAINT prova_questao_fk FOREIGN KEY ( id_prova )
    REFERENCES prova ( id_prova );

ALTER TABLE vaga_prova
    ADD CONSTRAINT prova_vaga_prova_fk FOREIGN KEY ( id_prova )
    REFERENCES prova ( id_prova );

ALTER TABLE resposta_questao
    ADD CONSTRAINT questao_resosta_questao_fk FOREIGN KEY (
id_questao )
    REFERENCES questao ( id_questao );

ALTER TABLE tag_candidato
    ADD CONSTRAINT tag_tag_candidato_fk FOREIGN KEY ( id_tag )
    REFERENCES tag ( id_tag );

ALTER TABLE tag_vaga
    ADD CONSTRAINT tag_tag_vaga_fk FOREIGN KEY ( id_tag )
    REFERENCES tag ( id_tag );

ALTER TABLE tentativa
    ADD CONSTRAINT tentativa_inscricao_fk FOREIGN KEY (
id_candidato, id_vaga )
    REFERENCES inscricao ( id_candidato, id_vaga );

ALTER TABLE resposta_questao
    ADD CONSTRAINT tentativa_resposta_questao_fk FOREIGN KEY (
id_tentativa )
    REFERENCES tentativa ( id_tentativa );

ALTER TABLE tag_vaga
    ADD CONSTRAINT vag_tag_vaga_fk FOREIGN KEY ( id_vaga )

```

```
REFERENCES vaga ( id_vaga );

ALTER TABLE inscricao
  ADD CONSTRAINT vaga_inscricao_fk FOREIGN KEY ( id_vaga )
    REFERENCES vaga ( id_vaga );

ALTER TABLE vaga_prova
  ADD CONSTRAINT vaga_vaga_prova_fk FOREIGN KEY ( id_vaga )
    REFERENCES vaga ( id_vaga );
```

5 CONCLUSÃO

Propomos nesse documento uma solução para um grande problema do setor de RH da empresa Atento. Eles possuem um grande gargalo no processo de seleção de candidatos para vagas pois eles recebem 17.000 candidatos mensais e apenas 17% são contratados.

A nossa proposta é o desenvolvimento e implantação de um sistema de recrutamento, provas e entrevistas, tudo de forma online. Para sustentar esse software é necessário um banco de dados bem modelado e que atenda aos requisitos do cliente. Esse documento abordou com detalhes a parte conceitual, lógica e física dessa modelagem.

Esperamos iniciar, a partir dessa entrega, a fase de desenvolvimento do sistema que após implantado será de grande valor para o cliente.

REFERÊNCIAS

Prado, Fabio. Qual é o melhor Banco de Dados: ORACLE ou SQL SERVER. 2012. Disponível em: < <http://www.fabioprado.net/2012/01/qual-e-o-melhor-banco-de-dados-oracle.html>>. Acesso em: 28 set 2018.

Oracle Database. Disponível em: < https://en.wikipedia.org/wiki/Oracle_Database>. Acesso em: 28 set 2018.

Atento Brasil. Disponível em: <<https://www.facebook.com/AtentoBrasilOficial/>>. Acesso em: 28 set 2018.