

Especificação:

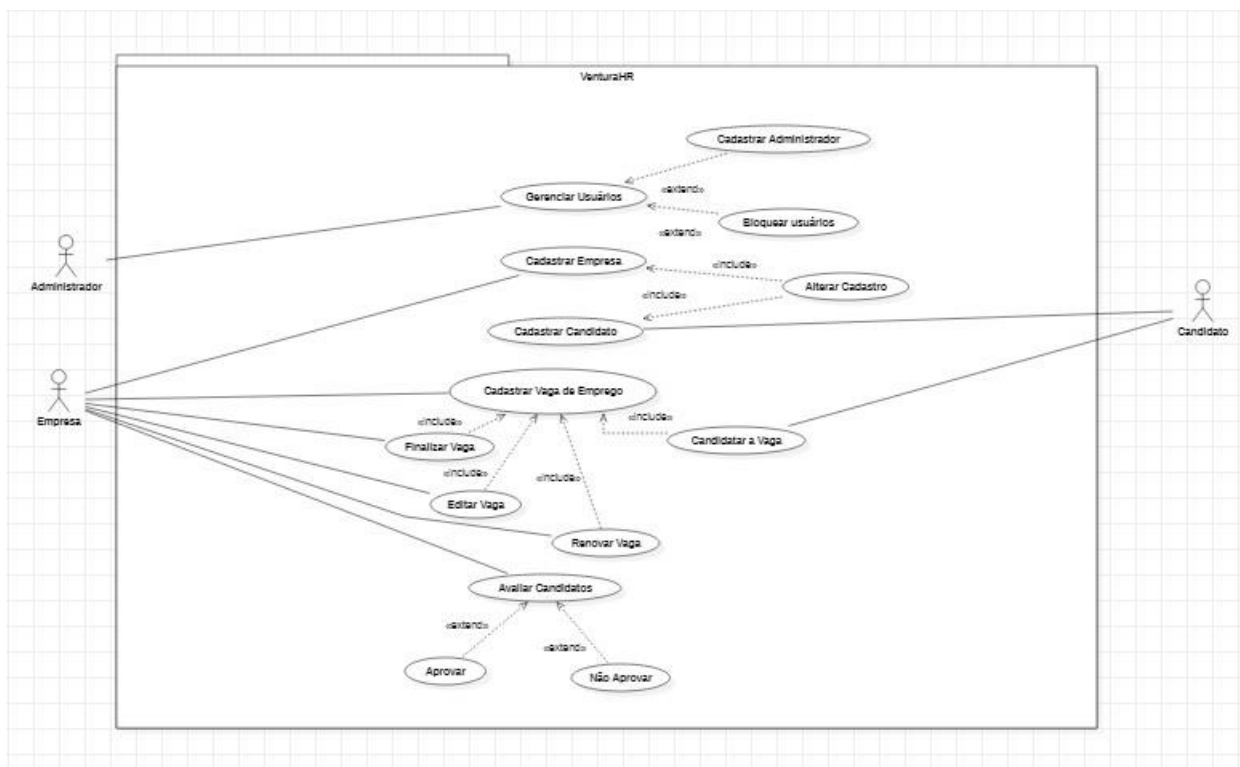
1. Faça uma lista de atores, descrevendo os seus papéis.

Administrador: Vai gerenciar as contas de Usuários e as Vagas cadastradas;

Empresa: Cadastrar Vagas de emprego;

Candidato: Visualizar vagas de emprego e se candidatar;

2. Criar o Modelo de Casos de Uso completo do projeto.



3. Crie uma Matriz de Requisitos (tabela) contendo as seguintes colunas: ID (RF para Requisito Funcional e NF para Requisito Não Funcional), Nome (quando for funcional coloque o nome do Caso de Uso), Descrição, Prioridade (Essencial, Importante, Desejável), Complexidade.
- Faça uma lista com os casos de uso na ordem que acha mais interessante para a implementação. Classifique os casos de uso quanto a sua complexidade: 1 = muito fácil, 2 = fácil, 3 = médio, 4 = complexo, 5 = muito complexo.
 - Relacione os requisitos não funcionais após todos os requisitos funcionais.
 - Justifique o que escreveu através de observações após a tabela.

UC01 - Gerenciar Usuários

UC02 - Cadastrar Empresa

UC03 - Cadastrar Candidato

UC04 - Cadastrar Vaga de Emprego

UC05 - Finalizar Vaga

UC06 - Editar Vaga

UC07 - Renovar Vaga

UC08 - Candidatar a Vaga

UC09 - Avaliar Candidatos

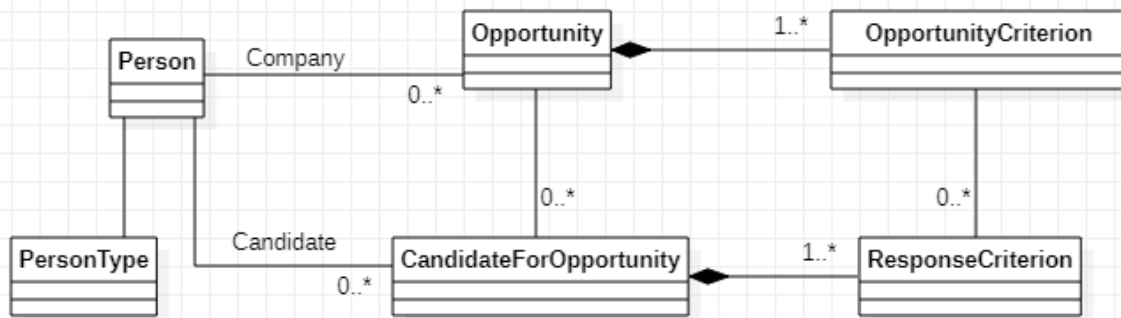
UC10 - Atualizar Cadastro

ID	Nome	Descrição	Prioridade	Complexidade
RF01	UC01	<p>Usuário deverá poder criar uma conta no sistema. Essa conta poderá ser de Empresa ou de Candidato.</p> <p>As contas de Administrador só podem ser criadas por Administradores.O sistema já vem com uma conta de administrador padrão.</p> <p>As contas de usuários não podem ser excluídas.</p>	Essencial	3
RF02	UC10	Usuário deverá poder alterar os dados da sua conta.	Importante	2
RF03	UC08	Todos os usuários deverão poder pesquisar vagas,porém somente Candidatos podem responder uma vaga.	Essencial	4
NF01		Administradores devem visualizar estatísticas operacionais:número de vagas criadas hoje,número de vagas respondidas hoje,número de usuários que criaram contas hoje por tipo.	Desejável	
RF04	UC01	Administradores devem poder pesquisar usuários para bloquear ou desbloquear suas contas.Usuários com a conta bloqueada não conseguem fazer login. No bloqueio, o Administrador deve informar a razão.	Importante	4
NF02		Todos os dias o sistema busca as vagas que têm data de expiração hoje e envia um email para a conta da Empresa correspondente informando que vai expirar para que o usuário possa tomar uma ação.	Desejável	
NF03		Todos os dias o sistema busca as vagas que têm data de expiração há dois dias e finaliza a vaga automaticamente.	Desejável	

RF05	UC04	Os usuários com conta de Empresa podem publicar vagas de emprego com seus dados descritivos e um conjunto de critérios de seleção que têm descrição, perfil mínimo desejado e peso.O peso ordena os critérios entre si. A data de expiração não pode ser maior que 30 dias.	Essencial	5
RF06	UC04	Os usuários com conta de Empresa podem listar as suas vagas publicadas. Essa lista mostra os dados da vaga e o seu estado: publicada, expirada ou finalizada. Selecionando a vaga é possível ver a lista de critérios.	Essencial	3

RF07	UC09	Os usuários com conta de Empresa, a partir de uma vaga publicada, podem consultar o ranking de candidatos que responderam, ordenado pela nota de cada um. Selecionando um candidato no ranking, é possível obter seus dados de contato.	Essencial	5
RF08	UC07	Na lista de vagas publicadas, o usuário com conta de Empresa pode renovar uma vaga Publicada ou Expirada por um período de até 30 dias. Vagas Finalizadas não podem ser mais alteradas.	Importante	5
RF09	UC05	Na lista de vagas publicadas, o usuário com conta de Empresa pode finalizar uma vaga.	Importante	2
RF10	UC06	As vagas publicadas podem ser editadas desde que não tenham nenhum candidato aplicado para elas.	Importante	2
RF11	UC08	Usuários com conta de Candidato podem responder uma vaga.	Essencial	2
NF04		Os usuários com conta de Candidato podem listar as suas vagas respondidas. Essa lista mostra os dados da vaga e o seu estado: publicada, expirada ou finalizada.	Desejável	
NF05		Os usuários com conta de Candidato podem indicar vagas que tenham pesquisado, usando um email para compartilhar a vaga.	Desejável	

4. Criar o modelo de classes com somente classes de entidade, apresentando os nomes das classes, relacionamentos e multiplicidade. Não é necessário detalhar os atributos e métodos neste momento.



Processo de Desenvolvimento de Software:

1. Considerando o que aprendeu sobre DAD, qual seria o ciclo de vida de software mais adequado para o desenvolvimento desse projeto? Justifique a sua escolha.

Ágil - Visto que já existe uma visualização objetiva das necessidades dos usuários, com requisitos bem definidos e risco baixo.

2. Considerando que iremos utilizar o DAD no desenvolvimento desse projeto, Quais elementos do Processo Unificado serão importados para esse projeto?

Serão importados para esse projeto as fases de:

Elaboração: uma vez que os requisitos já foram mitigados, nessa fase seria necessário apenas fazer as estimativas mais realistas sobre prazos e esforços;

Construção: com a implementação iterativa dos requisitos de menor risco; e

Transição: Aplicação dos testes finais e implantação do projeto.

3. Quais seriam as diferentes etapas do desenvolvimento desse projeto?

Conforme o DAD, as diferentes etapas do projeto são:

Construction: através de um ciclo definido de desenvolvimento (incremental e iterativo ou fluxo contínuo e enxuto), onde serão produzidos artefatos consumíveis do projeto.

Transition: Etapa onde o esforço é concentrado na entrega do projeto finalizado e funcional. No método ágil, essa fase é finalizada após total implantação do sistema e sign-off dos stakeholders.

4. Que elementos de desenvolvimento ágil devem ser utilizados nesse projeto?

Serão utilizadas práticas híbridas retiradas de metodologias como Scrum, XP e Agile Modeling. Onde serão apresentados os elementos:

Desenvolvimento iterativo e incremental (Scrum): o desenvolvimento será pautado por ciclos curtos em que, ao final de cada ciclo, seja possível entregar funcionalidades, coletar feedbacks e revisar requisitos.

Integração contínua (XP): atualização constante do sistema, através da implementação de testes unitários garantindo sua funcionalidade a cada incremento.

Releases (Agile Modeling): Implementação das alterações em produção trazendo maior visibilidade e valor para o cliente.

Link do Repositório no Git: https://github.com/caioco007/PB_VenturaHR