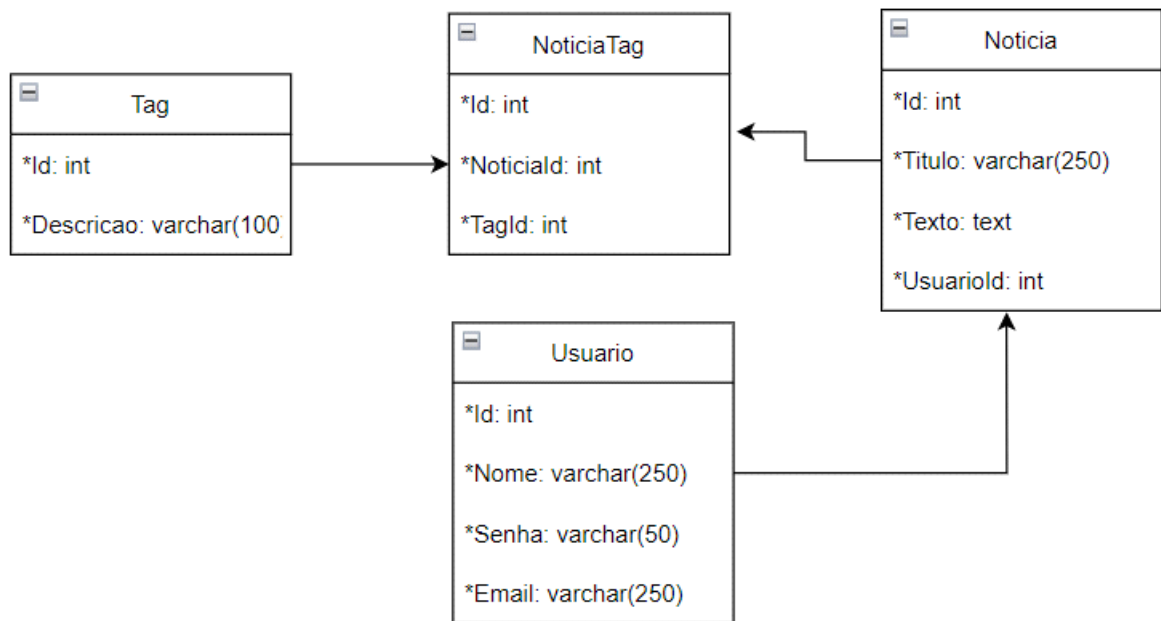


## 1. Apresentação da atividade

Esta prova tem como objetivo avaliar as habilidades do candidato quanto aos seus conhecimentos de c#, .net e suas tecnologias agregadas. O projeto deverá ser do tipo MVC, podendo ou não ser agregado outras arquiteturas, como DDD por exemplo. A versão do .net a ser utilizada deverá ser no mínimo a 7.

## 2. Modelo de Dados a ser criado:



Baseado no Modelo de Dados apresentado desenvolva as questões:

- Criar as entidades de acordo com o Modelo de Dados informado com a tecnologia conhecida como Migrations, utilizando estratégia conhecida como Model First.
- Desenvolver consulta, cadastro, edição, exclusão de tag e noticia. No cadastro e edicao de noticias permitir o vinculo com N Tags. O layout não será avaliado, apenas o domínio sobre HTML e as tags helpers do .net
- O Cadastro e Edição de Noticia devem ser desenvolvidos com o carregamento da view de forma assíncrona, através de requisição Ajax para o back-end, assim como o envio dos dados para persistência. Listagem e Exclusão poderão ser realizadas apenas via processamento de back-end;
- O CRUD de Tag deve ser desenvolvido se utilizando do padrão MVC, realizando as rotinas apenas através do FORM enviando as requisições para o back-end.
- Criar um arquivo resposta.txt no projeto com as respostas das perguntas a seguir:

1) No MVC, o que podemos fazer para enviar uma coleção de dados da classe

Controller para a View?

2) Precisamos validar os dados obrigatórios de um formulário para que não gere exceção no momento de armazenar na fonte de dados. Como podemos fazer isso?

3) Como podemos armazenar configurações (connectionString, urls de apis, chaves de acesso) da aplicação que variam de acordo com o ambiente (debug,desenv,prod) que será publicado a aplicação?

**Pontos que serão avaliados e considerados na entrega do teste:**

- Utilização do Migrations;
- Utilização do Entity Framework;
- Views com Html e Razor;
- Javascript e JQuery;
- Requisição Ajax;
- Manipulação de JSON;
- Rotinas Async;
- Validações Front e Back-end;
- Funcionalidades;
- Estrutura MVC;
- Organização e semântica de código;
- Técnicas de desenvolvimento;
- Baixo acoplamento;
- Reutilização;
- Baixa complexidade;

**Pontos que não serão considerados como critério de avaliação:**

- CRUD de Usuário;
- Design das telas.

**Devolutiva do teste:**

Nos enviar o projeto implementado. Caso não tenha utilizado Migrations, enviar juntamente o arquivo de banco de dados (ou script para criação do mesmo).