

## InLock

### 1. StoryTelling

Uma empresa do ramo de games, a **InLock**, deseja realizar a criação de sua base de dados para armazenar os jogos que são vendidos em sua loja. Além disso, como eles já possuem um desenvolvedor front-end atuando na empresa, não será necessário construir a interface com o usuário final. Para isto, deverá ser desenvolvida uma API (trabalhando com JSON), para que o front-end (seja web ou mobile) realize a construção com base nas informações fornecidas.

Os jogos deverão ter as seguintes características:

**Nome, Descrição, Data de Lançamento e Valor.** Além disso, o jogo deverá ter um **estúdio** pelo qual foi desenvolvido.

Por exemplo:

O **Diablo 3** foi lançado em **15 de maio de 2012**, é um jogo que contém bastante ação e é viciante, seja você um novato ou um fã. Além disso, seu estúdio é a **Blizzard**. E o jogo custa **R\$ 99,00**.

**Red Dead Redemption II** é um jogo eletrônico de ação-aventura western desenvolvido pela **Rockstar Studios**. Lançado mundialmente em **26 de outubro de 2018**. E o jogo custa **R\$ 120,00**.

Além disso, somente usuários com o perfil de **ADMINISTRADOR** poderão cadastrar um novo jogo. Qualquer usuário autenticado, com o perfil de **ADMINISTRADOR** ou **CLIENTE**, poderá realizar a listagem de jogos.

Para que o jogo tenha início, HROADS disponibilizou um conteúdo com algumas informações sobre como ele quer que o jogo seja construído.

## Classes

- Bárbaro (Lança Mortal, Escudo Supremo)
- Cruzado (Escudo Supremo)
- Caçadora de Demônios (Lança Mortal)
- Monge (Recuperar Vida, Escudo Supremo)
- Necromante (começa sem habilidades)
- Feiticeiro (Recuperar Vida)
- Arcanista (começa sem habilidades)

## Habilidades

- Lança Mortal (tipo de habilidade: ataque)
- Escudo Supremo (tipo de habilidade: defesa)
- Recuperar Vida (tipo de habilidade: cura)

## Tipos de Habilidades

- Ataque
- Defesa
- Cura
- Magia

## Roteiro BD

Primeiro arquivo: inlock\_01\_DDL.sql

- Criar um banco de dados chamado inlock\_games\_manha/tarde;
- Criar uma tabela de estúdios com os campos de idEstudio e nomeEstudio;
- Criar uma tabela de jogos com os campos idJogo, nomeJogo, descrição, dataLancamento, valor e idEstudio;
- Criar uma tabela de tipos de usuários contendo os campos idTipoUsuario e titulo;
- Criar uma tabela de usuários contendo os campos de idUsuario, email, senha e idTipoUsuario;

Obs.: Atente-se na definição dos tipos de dados.

Segundo arquivo: inlock\_02\_DML.sql

- Inserir um usuário do tipo **ADMINISTRADOR** que tenha o e-mail igual a **admin@admin.com** e a senha igual a **admin**;
- Inserir um usuário do tipo **CLIENTE** que tenha o e-mail igual a [cliente@cliente.com](mailto:cliente@cliente.com) e a senha igual a **cliente**;
- Inserir três estúdios: um com o nome de **Blizzard**, outro com o nome de **Rockstar Studios** e o último com o nome de **Square Enix**;
- Inserir um jogo com o nome de: **Diablo 3**, com data de lançamento de: **15 de maio de 2012**, que contenha a descrição de: **é um jogo que contém bastante ação e é viciante, seja você um novato ou um fã**. Seu estúdio é a **Blizzard**. E o jogo custa **R\$ 99,00**;
- Inserir um jogo com o nome de: **Red Dead Redemption II**, com a descrição de: **jogo eletrônico de ação-aventura western**. Seu estúdio será a **Rockstar Studios**. Lançado mundialmente em **26 de outubro de 2018**. E o jogo custa **R\$ 120,00**;

Terceiro arquivo: inlock\_03\_DQL.sql

- Listar todos os usuários;
- Listar todos os estúdios;
- Listar todos os jogos;
- Listar todos os jogos e seus respectivos estúdios;
- Buscar e trazer na lista todos os estúdios com os respectivos jogos. Obs.: Listar todos os estúdios mesmo que eles não contenham nenhum jogo de referência;
- Buscar um usuário por e-mail e senha (login);
- Buscar um jogo por idJogo;
- Buscar um estúdio por idEstudio;

## Roteiro API

Criar um novo projeto do tipo WebAPI com o seguinte nome: **senai.inlock.webApi**.

Sua API deverá ter as seguintes funcionalidades:

- O usuário do tipo **ADMINISTRADOR** poderá realizar o cadastro dos jogos;
- O usuário do tipo **CLIENTE** ou **ADMINISTRADOR** poderá visualizar a lista de todos os jogos e seus respectivos estúdios;
- Sua autenticação deverá ser feita utilizando JWT (não esquecer de criar um endpoint para gerar o token);
- Utilizar o Swagger para a documentação de sua API;

## Desafio extra da API

Mostrar a lista de todos os estúdios e incluir na lista a lista de jogos daquele determinado estúdio;

## O que deverá ser entregue?

Deverá ser compartilhado o link do GitHub com a pasta do exercício criada dentro do repositório da sprint 2 – back-end, contendo a seguinte estrutura de pastas e respectivos arquivos:

### BD

- inlock\_01\_DDL.sql;
- Inlock\_02\_DML.sql;
- Inlock\_03\_DQL.sql;
- Exportar o diagrama do banco de dados e incluir a imagem dentro deste diretório;

### Back-end

Solução do Projeto (não esqueça que o arquivo .sln **não** armazena os demais arquivos do seu projeto, é preciso entregar toda a estrutura criada);

### Postman

Exportar a coleção de requisições criada com estrutura de pastas:

- Estudios
- Jogos
- Login

## Planejamento

Faça a entrega também do link do quadro do Trello com todo o planejamento das tarefas desenvolvidas.