

Teste de Desenvolvedor Back-End

Visão Geral

Parabéns por ter chegado nessa etapa, para este teste você precisa criar uma api em C#.net para simular algumas funcionalidades de um jogo de poker.

Requisitos

- Crie um projeto ASP.NET Core Web API.
- Utilize Entity framework ou Dapper com banco em memória.
- Implemente a autenticação JWT para proteger os endpoints da API.
- Documentação Swagger

Endpoints da API

Endpoint de Login:

- Rota: /api/auth/login
- Método: POST
- Parâmetros: { "username": "string", "password": "string" }
- Retorna um token JWT válido se as credenciais forem autenticadas com sucesso.

Endpoint para Gravar Dados do Usuário:

- Rota: /api/users
- Método: POST
- Parâmetros: { "name": "string", "cpf": "string", "phone": "string", "password": "string", "username": "string" }
- Retorno: { "userId": "int" }
- Grava os dados do usuário no banco de dados.

Endpoint para Criar uma Mesa de Poker:

- Rota: /api/poker-tables
- Método: POST
- Parâmetros: { "name": "string" }
- Retorno: { "tableId": "int" }
- Cria uma mesa de poker com até 8 jogadores.

Endpoint para Adicionar um Usuário à Mesa:

- Rota: /api/poker-tables/{tableId}/players
- Método: POST
- Parâmetros: { "userId": "int" }
- Adiciona um usuário à mesa especificada pelo tableId.

Endpoint para Simular um Ganhador:

- Rota: /api/poker-tables/{tableId}/winner

- Método: POST
- Parâmetros: { "tableId": "int" }
- Retorno: { "userName": "string", "userId": "int" }
- Retorna o ganhador da mesa de poker especificada pelo tableId.

Regras de Negócio

- Garanta que o mesmo usuário não possa ser inserido duas vezes na mesma mesa.
- Uma mesa precisa ter no mínimo 3 jogadores para que haja um ganhador.
- Validar minimamente os dados imputados como cpf, phone.

CrITÉrios de Avaliação

- Implementação correta da autenticação JWT.
 - Implementação correta dos endpoints da API.
 - Correta configuração e uso do Entity Framework Core e Dapper.
 - Aplicação correta das regras de negócio.
 - Documentação completa e precisa da API usando Swagger.
 - Implementação de Testes é um diferencial
 - Aplicação estar preparada para Docker é um diferencial
- aneira eficiente.

Entrega

Todo material que você julgar necessário ser entregue deverá ser realizada para os e-mails abaixo até dia 27/03/2024:

- italo.amorim@suprema.group
- pedro.braganick@suprema.group

Lembre-se usar as boas práticas de programação e organização.

Boa sorte!