

O banco de dados utilizado foi SQL SERVER, utilizando o componente Dapper pela simplicidade e rapidez para consultar uma base já existente.

Vide o modelo relacional no arquivo modelo\_dados.pdf

Os scripts do banco de dados estão no repositório, pasta script

Para conexão ao banco de dados é necessário criar uma variável de ambiente no sistema operacional com o nome de “`DbConnection_locacaoAPI`” com a string de conexão.

Todos os endpoints estão setados para autorização por token.

Exceto endpoint de Login, onde fornece o token de acesso de um operado e/ou cliente.

Para acesso inicial aos endpoints é necessário consumir inicialmente o endpoint /Login e usar umas das informações de acesso abaixo.

Operador para teste

matricula: 123456

senha: 123456

perfil: 1

Cliente para teste

CPF: 44032721009

senha: 123456

perfil: 2

### Considerações sobre os tópicos

- O cadastro de usuário terá os campos login, senha, onde o campo login pode ser o CPF do cliente (perfil CLIENTE) ou a matrícula (perfil OPERADOR)

Um usuário é criado quando é cadastro um OPERADOR ou CLIENTE

- Disponibilizar um serviço de simulação de locação com base no veículo selecionado, valor hora do veículo, total de horas.

Considere que, por tratar de uma simulação. O front-end poderia simular com base nas informações do veículo, consumindo o endpoint de veículos, selecionado utilizando um formulário.

- Serviço de agendamento de um veículo calculando o valor final considerando a categoria do veículo, valor hora, total de horas

Não ficou claro o detalhamento do agendamento. Adicionei outros campos que julguei necessário para um agendamento completo de um veículo. Adicionado o campo status, data agendamento, data retirada e data devolução.

Sendo o status: 0 – agendando, 1-Retirado , 2-Entregue. Data de retirada é a data que o cliente retirou o veiculo (status=1), Data devolução é a data que o cliente entregou o veiculo (status =2)

- Disponibilizar um modelo de contrato de locação no formato .PDF

Criado um endpoint para download de PDF modelo.

### Testes unitários

Devido ao tempo muitos testes não foram implementados.

A opção escolhida foi o xUnit pela facilidade e simplicidade de uso