



**Curso: Desenvolvimento fullstack  
turma: 2023.1**

**Aluno: Israel dos Santos Hamdan D'Arújo**

**Disciplina: RPG0035 -Nível 5: Software Sem Segurança Não Serve**

**Tutor: Robson Lorbieski**

#### **Objetivos da prática**

- Descrever o controle básico de acesso a uma API Rest;
- Descrever o tratamento de dados sensíveis e log de erros com foco em segurança
- Descrever a prevenção de ataques de acesso não autorizado com base em tokens desprotegidos/desatualizados;
- Descrever o tratamento de SQL Injection em códigos-fonte; Descrever o tratamento de CRLF Injection em códigos-fonte;
- Descrever a prevenção a ataques do tipo CSRF em sistemas web;

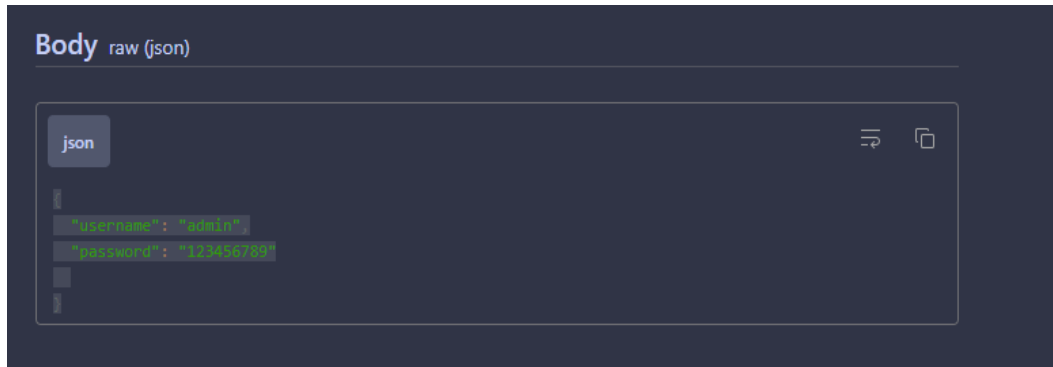
#### **Tecnologias utilizadas**

- [Express.js](https://expressjs.com/)

## Autenticação

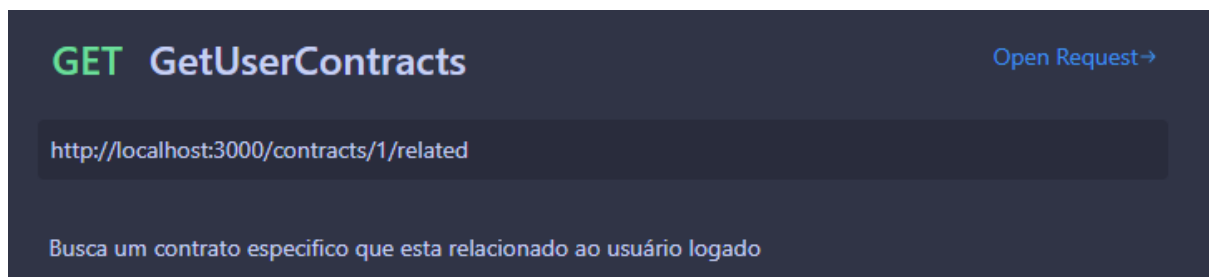
Ao fazer login o usuário recebe automaticamente um token jwt com duração de 24 horas, onde contém o seu Id, o seu nível de acesso e seu nome de usuário de maneira criptografada

- Corpo da requisição



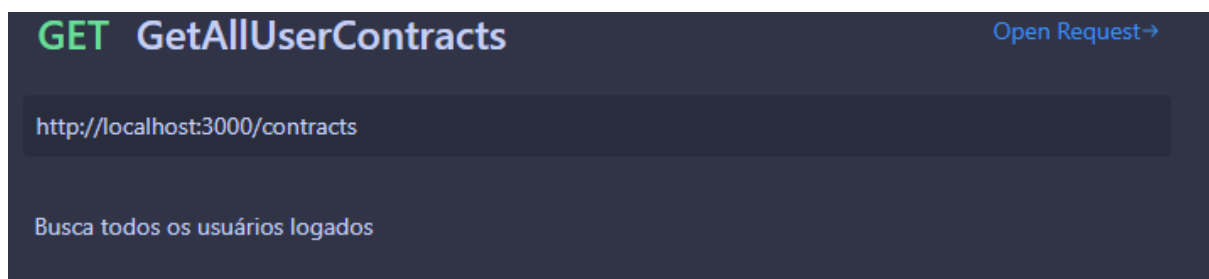
Quando esse token é recebido ele deve ser enviado no header de todas as outras requisições a fim de liberar o acesso

## Busca de contrato de acordo com o usuário logado



Aqui o id de usuário é extraído do token jwt mandado no header da requisição, então a partir desse id, e com o id do contrato na rota, o contrato é localizado

## Busca todos os contratos do usuário



## Criar novo contrato

Ao criar um novo contrato é necessário que seja um usuário com nível de acesso Admin, e o contrato é relacionado ao usuário logado

**POST** newContract

Open Request→

`http://localhost:3000/contracts/new`

Cria novo contrato e relaciona com o usuário logado

Body raw (json)

json

```
{
  "company": "Expert",
  "init": "2023-03-09",
  "end": "2025-07-15"
}
```

Busca informações do usuário logado

**GET** GetUserInfo

Open Request→

`http://localhost:3000/user/`

Add request description...

Authorization Bearer Token

Token <token>

## Criar novo usuário

**POST** newUser [Open Request →](#)

`http://localhost:3000/user/newUser`

Add request description...

**Body** raw (json)

json

```
[{"username": "Israel", "password": "123456", "email": "test@gmail.com", "profile": "admin"}]
```