

Login completo com Express

Guia passo a passo

Ponto de partida

Para seguir este passo a passo, seu projeto deve ter o seguinte:

1. Um projeto com Node e Express instalados e em execução
2. Uma estrutura MVC onde existem roteadores e controladores
3. Visualizações com EJS ou qualquer outro mecanismo de template

Para este exemplo, vamos imaginar que estamos validando o cadastro de um usuário.

Vamos começar! 🤖👉



Passo 1: habilitar as sessões

Para fazer o login funcionar, a primeira funcionalidade que teremos de ter habilitada é o uso de sessões. Para isso utilizaremos um módulo chamado [Express Session](#). Para instalá-lo, usaremos o mesmo comando que usamos para os módulos anteriores.

```
npm i express-session
```

A seguir, temos de habilitar as sessões: para isso, trabalharemos no app.js (ou no arquivo que temos como ponto de entrada). Como primeiro passo, iremos importar o módulo de sessões e, como segundo passo, iremos habilitá-los na configuração Express.

```
const session = require('express-session');

app.use(session({
  secret: 'Nome do site',

  resave: false,
  saveUninitialized: true,
}));
```



Passo 2: habilitar os cookies

Além disso, para funcionalidades um pouco mais avançadas, como lembrar o usuário, estaremos gerenciando cookies, para isso precisamos de um módulo chamado [Cookie Parser](#). A instalação é como o resto dos módulos.

```
npm i cookie-parser
```

Então, em nosso ponto de entrada, adicionaremos o seguinte código:

```
const session = require('express-session');
app.use(cookieParser());
```



Passo 3: habilitar a encriptação de dados

Finalmente, nosso login não seria completo se não implementássemos um método seguro para salvar as senhas.

Certamente, você já realizou esta etapa no momento da implementação do registro do usuário. Caso ainda não tenha feito, estas são as etapas a serem seguidas:

```
npm i bcryptjs
```

Para encriptar um dado:

```
let hash = bcrypt.hashSync('minhaSenha!');
```

Para verificar um dado:

```
bcrypt.compareSync("minhaSenha!", hash); // true  
bcrypt.compareSync("outraSenha", hash); // false
```



Passo 4: rotas, controller e view do login

Para receber os dados do usuário, será necessário ter:

- Uma visualização que contenha os dados que usaremos para o usuário inserir:
 - Seu e-mail (pode ser um usuário também)
 - Sua senha
- O caminho que o formulário de login mostra, pode ser / users / login
- O método do controlador que processa essa rota



Passo 5: validar os dados

Embora não seja específico para o login, a validação deve estar presente em todos os locais onde recebemos informações do usuário.



Passo 6: verificar o usuário e a senha

Agora que sabemos que os dados enviados são seguros, verificamos se o usuário existe e, se for o caso, se a senha corresponde à que temos em nosso banco de dados.



Passo 7: registrar o usuário na sessão

Se todos os dados anteriores estiverem corretos, significa que nosso usuário entrou corretamente. Nesse caso, devemos lembrar que o usuário está logado durante sua sessão.



Passo 8: criar os middlewares de autenticação

Agora que podemos saber se o usuário existe ou não em sessão, é hora de permitir que nosso site tenha rotas para as quais apenas usuários logados tenham acesso.



Passo extra: lembrar usuário

É sempre útil que o site se lembre de nós, caso não desejemos inserir nossos dados de acesso cada vez que visitarmos o site.

Para isso, colocaremos em uso o segundo módulo que instalamos ao iniciar este guia.

Conclusão

Embora o uso de middleware pareça complexo a princípio, ele simplifica e evita o desenvolvimento de uma grande lógica de validação que teríamos que implementar se não os utilizássemos.

Até a próxima!