



Tecnologias Para Desenvolvimento Web

UNIDADE 07

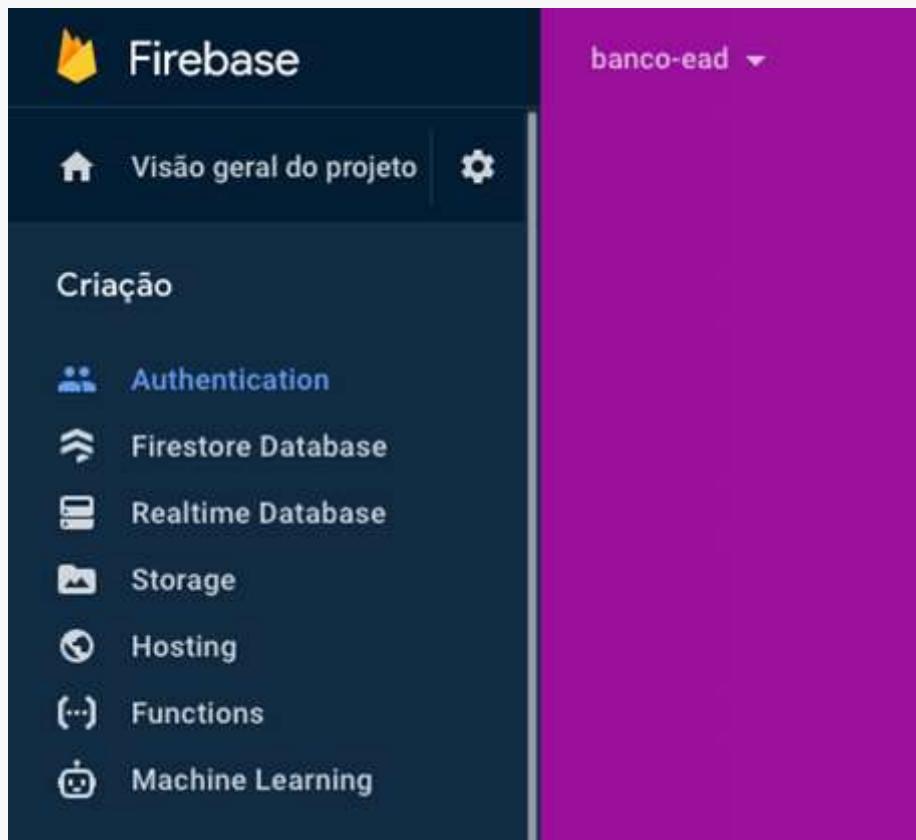
Firebase Authentication

| **Firebase Authentication**

A maioria dos sistemas precisam reconhecer a identidade do usuário. Ter essa informação permite que um sistema salve os dados do usuário na nuvem com segurança e forneça a mesma experiência personalizada em todos os dispositivos do usuário [1]. O Firebase Authentication fornece serviços de back-end, SDKs fáceis de usar e bibliotecas de IU prontas para autenticar usuários no seu sistema. Ele oferece suporte à autenticação usando senhas, números de telefone, provedores de identidade federados conhecidos, como Google, Facebook e Twitter, entre outros [1].

Para isso, precisamos acessar a nossa página do Firebase e entrar no projeto que vamos utilizar. Em seguida, podemos perceber através da Figura 1, que no menu à esquerda, temos a opção **Authentication**, que é a área de autenticação do Firebase. Devemos clicar nesta opção.

Figura 1. Menu do Firebase.



Fonte: o autor, 2021.

Após entrar na área de autenticação do Firebase, devemos acessar a aba **Sign-in method**, que é a aba para métodos de login que podemos utilizar em nossa aplicação React, conforme é ilustrado na Figura 2.

Figura 2. Sign-in method.

A screenshot of the 'Authentication' screen in the Firebase console. At the top, there are tabs for 'Users', 'Sign-in method' (which is underlined in blue, indicating it's active), 'Templates', and 'Usage'. Below the tabs, there's a section titled 'Provedores de login' (Login providers). It contains two columns: 'Provedores nativos' (Native providers) and 'Outros provedores' (Other providers). In the 'Provedores nativos' column, three options are listed: 'E-mail/senha' (Email/password), 'Smartphone' (Smartphone), and 'Anônimo' (Anonymous). A green dashed rectangle highlights the first two options ('E-mail/senha' and 'Smartphone'), and a green arrow points from the text 'E-mail/senha' in the previous paragraph to this highlighted area. In the 'Outros provedores' column, there are ten more providers: Google, Facebook, Play Games, Game Center, Apple, GitHub, Microsoft, Twitter, and Yahoo.

Fonte: o autor, 2021.

Conforme é ilustrado na Figura 2, vamos realizar a autenticação através do provedor de E-mail/senha. Logo em seguida, após clicar no provedor em questão, irá aparecer a página ilustrada na Figura 3.

Figura 3. Permissão de inscrição.



Fonte: o autor, 2021.

Devemos **ativar** a inscrição de usuários com e-mails próprios para a autenticação no Firebase Authentication. Em seguida, devemos clicar em **Salvar**. Pronto, agora este provedor já pode ser utilizado em nossa aplicação React.

Agora, devemos acessar nosso arquivo de configuração do Firebase que está em nosso projeto. Este arquivo, por enquanto, tem apenas a importação do pacote com dados do Firebase e Firestore e ainda não tem do Authentication. Para isso, devemos importá-lo, conforme é ilustrado na Figura 4.

Figura 4. Importação do pacote do Firebase Authentication.

```
import firebase from 'firebase/app';
import 'firebase/firestore';
import 'firebase/auth';
```

Fonte: o autor, 2021.

Agora que foi importado, já podemos realizar a autenticação do usuário em nossa aplicação **React**. Para isso, precisamos inicialmente ter dois **inputs** do tipo **text** e duas variáveis na **state**: **email** e **senha**. Estes dois inputs do tipo **text** devem possuir o atributo **onChange** e fazer com que o valor de cada um, preencha sua variável da **state**. Esses passos você já aprendeu nas aulas anteriores!

Precisamos também, de um botão, que ao ser clicado, execute um método.

Sendo assim, após estes passos, precisamos fazer com que os valores que estão na **state** criem um novo usuário com **email** e **senha** no Firebase Authentication. A Figura 5 ilustra como podemos fazer essa gravação.

Figura 5. Método de criação de usuário.

```
criarUsuario(){  
    firebase.auth().createUserWithEmailAndPassword(this.state.email, this.state.senha);  
}
```

Fonte: o autor, 2021.

Perceba que é muito simples criar um novo usuário, basta executarmos o método **auth()** que é o método de autenticação do Firebase e executar o método **createUserWithEmailAndPassword** (Em português, criar usuário com e-mail e senha). O próprio nome do método já diz claramente o que será feito! Este método exige que tenhamos dois parâmetros, o **email** e **senha**. Como já temos o **email** e **senha** digitados na **state**, basta passarmos como parâmetro e pronto, já teremos um usuário criado no Firebase Authentication.

Após finalizar este processo, podemos acessar a aba chamada **Users**, que fica na área do **Authentication** do Firebase. Podemos perceber, que um novo usuário foi adicionado. A Figura 6 ilustra como essa área ficou.

Figura 6. Usuário adicionado no Firebase Authentication.

Authentication



Users Sign-in method Templates Usage

Desenvolva protótipos e realize testes de ponta a ponta com o Pacote do emulador local, agora com o Firebase Authentication. [Primeiros passos](#)

Pesquise por endereço de e-mail, número de telefone ou UID do usuário					Adicionar usuário		
Identificador	Provedores	Data de criação	Último login	UID do usuário			
eduardo.lino@pucpr.br		17 de jan. d...	17 de jan. d...	rYJMXjH0yLf40PVLoo9CTfUrgK13			

Fonte: o autor, 2021.

Nesta área (Figura 6), temos uma tabela com o identificador que é o e-mail e o provedor, que foi através de E-mail/senha. Temos também a data de criação, último login e o UID do usuário, que é uma chave única deste usuário que podemos utilizar para diversas outras funcionalidades, inclusive relacioná-la com dados do Firebase Firestore, para dizer de quem este login pertence nos dados gravados de usuários no Firestore.

Perceba como foi simples realizar esta gravação e a quantidade de informações que o Firebase nos traz sem que precisemos criar muito código em nossa aplicação, já está tudo pronto dentro da plataforma!

CONCLUSÃO

Nesta unidade, foi apresentado o conceito de componentes em classe e estado do componente. Estes dois conceitos são essenciais para o andamento do nosso conteúdo e entendimento da construção de componentes mais avançados.

REFERÊNCIAS

[1] React. *React – Uma biblioteca JavaScript para criar interfaces de usuário*. Disponível em: <https://pt-br.reactjs.org/>. Acessado em: 12 de outubro de 2021.



© PUCPR - Todos os direitos reservados.