

# ■ Aula – Autenticação com Firebase (Login com E-mail e Senha)

## ■ Objetivo da Aula

- Implementar autenticação utilizando Firebase Authentication.
- Criar usuários no portal do Firebase.
- Configurar o projeto Firebase no React Native.
- Implementar login com e-mail e senha.
- Gerar e utilizar o token JWT do usuário autenticado.

## ■ Parte 1 – Criando Login e Senha no Portal do Firebase

- 1. Acesse: <https://console.firebaseio.google.com>
- 2. Clique em 'Adicionar projeto'.
- 3. Defina um nome para o projeto.
- 4. Finalize a criação do projeto.
- 5. Vá em Authentication > Get Started.
- 6. Na aba 'Sign-in method', ative 'Email/Password'.
- 7. Vá na aba 'Users' e clique em 'Add user'.
- 8. Informe email e senha e salve.

## ■ Parte 2 – Configuração do Firebase no App

Arquivo firebase.ts:

```
import { initializeApp } from "firebase/app";
import { getAuth } from "firebase/auth";

const firebaseConfig = {} // Cole aqui o config do seu projeto

const app = initializeApp(firebaseConfig);
export const auth = getAuth(app);
```

- initializeApp(firebaseConfig) conecta o app ao projeto criado no Firebase.

- O firebaseConfig é obtido nas configurações do projeto no console.
- getAuth(app) cria a instância de autenticação.
- A constante auth será usada para login e logout.

## ■ Parte 3 – Implementação do Login no App

Arquivo index.tsx:

```
import { auth } from '@/services.firebaseio';
import { signInWithEmailAndPassword } from 'firebase/auth';
import { useState } from 'react';
import { Button, StyleSheet, TextInput, View } from 'react-native';

export default function Login() {
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');

  const login = async () => {
    try {
      const credential = await signInWithEmailAndPassword(auth, email, password)
      alert(`Usuário logado: ${email}`)
      const token = await credential.user.getIdToken()
      console.log(token)
    } catch (error: any) {
      alert(`Usuário NÃO logado`)
      console.log(error)
    }
  }

  return (
    <View style={ styles.container }>
      <TextInput
        placeholder='Digite seu E-mail'
        onChangeText={(text) => setEmail(text)}
      />
      <TextInput
        placeholder='Digite sua senha'
        onChangeText={(text) => setPassword(text)}
      />
      <Button title='Login' onPress={login}/>
    </View>
  )
}
```

## ■ Fluxo Completo da Autenticação

- 1. Usuário digita email e senha.

- 2. O botão executa a função login().
- 3. signInWithEmailAndPassword(auth, email, password) envia os dados ao Firebase.
- 4. O Firebase valida as credenciais no servidor.
- 5. Se válido, retorna um objeto UserCredential.
- 6. credential.user.getIdToken() gera o token JWT.
- 7. O token pode ser enviado para APIs próprias para validação.

## ■ Conceitos Importantes

- Authentication: processo de verificação de identidade.
- JWT (JSON Web Token): token que identifica o usuário autenticado.
- UserCredential: objeto retornado após login bem-sucedido.
- getIdToken(): método que gera o token do usuário.
- try/catch: estrutura para tratamento de erros assíncronos.

## ■ Conclusão

- Criamos um projeto no Firebase.
- Ativamos login com e-mail e senha.
- Criamos usuário manualmente.
- Configuramos Firebase no React Native.
- Implementamos login utilizando signInWithEmailAndPassword.
- Geramos o token JWT do usuário autenticado.