🔐 Aula – Autenticação com JWT (JSON Web Token)

📌 Objetivo da Aula

Explicar como funciona autenticação utilizando JWT em uma aplicação Mobile + Backend.

🧠 O que é JWT?

JWT (JSON Web Token) é um token assinado digitalmente usado para autenticação e autorização.

Ele é composto por `3` **partes**:

```
HEADER.PAYLOAD.SIGNATURE
```

🧱 Estrutura do JWT

1️⃣ Header

Contém metadados do token.

Exemplo:

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

- `alg` → algoritmo de assinatura
- `typ` → tipo do token

2️⃣ Payload (Body)

Contém os dados do usuário (claims).

Exemplo:

```
{
  "id": "1",
  "exp": 1712345678
}
```

Pode conter:

- id do usuário
- email
- permissões
- data de expiração (`exp`)

🚨 Nunca colocar:

- senha
- dados sensíveis
- cartão
- CPF

O payload é apenas codificado em Base `6 4` , qualquer pessoa pode decodificar.

3️⃣ Signature

Garante que o token não foi alterado.

Exemplo conceitual:

```
assinatura = HMACSHA256(
   base64(header) + "." + base64(payload),
   SECRET_KEY
)
```

Se alguém modificar o payload, a assinatura deixa de ser válida.

🔁 Fluxo Completo da Aplicação

📱 1️⃣ Login (Mobile → Backend)

Usuário envia:

```
email + senha
```

Backend:

- valida credenciais
- gera JWT
- retorna o token para o app

🔐 2️⃣ Armazenamento do Token

No Mobile:

- Utilizar `expo-secure-store`

❌ Não usar AsyncStorage em produção.

🌐 3️⃣ Requisição Autenticada

Exemplo:

```
GET /users/scores
Authorization: Bearer <JWT>
```

Backend:

1. Extrai o token
2. Valida assinatura usando a SECRET_KEY
3. Decodifica o payload
4. Obtém o `id` do usuário
5. Busca dados no banco

🔑 Onde fica a chave secreta?

No backend:

```
.env
SECRET_KEY=chave_super_secreta
```

- Nunca enviar para o frontend
- Nunca expor no código público

📚 Conceitos Fundamentais

- JWT é assinado, para evitar que seja alterado.
- Payload não deve conter dados sensíveis
- A SECRET_KEY fica apenas no backend
- Token deve ser enviado no header Authorization
- Backend é responsável por validar e identificar o usuário

🎯 Resumo Final

JWT permite que o backend identifique o usuário sem armazenar sessão no servidor.

Fluxo resumido:

1. Usuário faz login
2. Backend gera JWT
3. App armazena o token com segurança
4. Requisições enviam o token no header
5. Backend valida e retorna os dados do usuário autenticado