

## 1.2 احراز هویت (Authentication) و مدیریت توکن

### 1 مکانیزم احراز هویت

سیستم از JWT (JSON Web Token) مبتنی بر Bearer Token برای احراز هویت استفاده می‌کند.

#### نوع Authentication

Bearer Token (JWT)

#### نحوه ارسال توکن در درخواست‌ها

تمام Endpoint های محافظت شده نیازمند ارسال هدر زیر هستند:

Authorization: Bearer <access\_token>

### 2 تنظیمات JWT (بر اساس appsettings.json)

تنظیمات JWT از بخش Jwt در فایل appsettings.json خوانده می‌شود.

کلید	توضیح	مقدار فعلی
Jwt:Secret	کلید امضای توکن (Signing Key)	مقدار تنظیم شده در پروژه
Jwt:Issuer	صادرکننده توکن	PayanehaTicket
Jwt:Audience	دریافت کننده معتبر	PayanehaTicket

#### مدت اعتبار توکن (Token Expiration)

مدت زمان اعتبار توکن در کد به صورت صریح تعیین شده است:

DateTime.UtcNow.AddHours(1)

بنابراین:

- ⌚ Access Token Validity: 1 ساعت
- 🕒 UTC: مبنای زمان

### 3 فلو کامل احراز هویت (OTP-Based Authentication Flow)

سیستم از مکانیزم OTP برای صدور توکن استفاده می‌کند.

#### مرحله 1 — ارسال کد OTP

POST /api/Auth/loginOtp

Request Body:

```
{
  "mobile": "09123456789"
}
```

#### مرحله 2 — تأیید OTP و دریافت توکن

POST /api/Auth/VerifyOtpCode

Request Body:

```
{
  "mobile": "09123456789",

```

```
"otpCode": "12345"
}
```

Response (Success):

```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "expireAt": "2026-02-15T15:30:00Z"
}
```

### مرحله 3 — استفاده از توکن

در تمام درخواست‌های محافظت‌شده:

```
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

## 4 Endpoint های بدون نیاز به توکن (AllowAnonymous)

در وضعیت فعلی کد، Endpoint های زیر نیاز به JWT ندارند:

- GetServiceDetail
- SaleTicketsNew
- GetListBusSummarys
- GetListPassengerInfo (دارای چک دستی توکن)
- AllCities
- getListCities
- GetTerminals
- برخی Endpoint های Terminal

⚠ نکته مهم: برخی Endpoint ها به صورت رسمی [AllowAnonymous] هستند و برخی دیگر دارای بررسی دستی توکن در داخل متد می‌باشند که از نظر معماری پیشنهاد نمی‌شود.

### ارسال خودکار توکن در تمام درخواست‌ها

مثال (Axios Interceptor):

```
axios.interceptors.request.use((config) => {
  const token = localStorage.getItem("token");
  if (token) {
    config.headers.Authorization = `Bearer ${token}`;
  }
  return config;
});
```

### مدیریت خطای 401

در صورت دریافت پاسخ:

```
HTTP 401 Unauthorized
```

باید:

1. توکن حذف شود
2. کاربر به صفحه Login ریدایرکت شود
3. State احراز هویت ریست گردد

## Comments