

# API de Usuarios

## Listar Usuarios

**Endpoint:** GET /users

Obtiene la lista de usuarios con paginación.

**Permiso necesario:** view\_users

### Request HTTP

```
GET /users?_start=0&_end=10&_sort=nombre&_order=ASC HTTP/1.1
Host: localhost:3000
Authorization: Bearer tu-token-aqui
```

### Response HTTP

```
HTTP/1.1 200 OK
Content-Type: application/json
X-Total-Count: 50

[
  {
    "_id": "507f1f77bcf86cd799439011",
    "nombre": "María",
    "apellidos": "García Ramírez",
    "correo_electronico": "maria@paramedia.com",
    "role": "paramedic",
    "telefono": "5512345678",
    "turnos": ["LV-8am3pm"]
  }
]
```

### Ejemplo JavaScript

```
async function obtenerUsuarios() {
  const token = localStorage.getItem('token');

  const response = await fetch('http://localhost:3000/users?_start=0&_end=10', {
    headers: {
      'Authorization': `Bearer ${token}`
    }
  });

  const total = response.headers.get('X-Total-Count');
  const usuarios = await response.json();

  console.log(`Total de usuarios: ${total}`);
}
```

```
    return usuarios;
}
```

## Obtener Usuario por ID

**Endpoint:** GET /users/:id

**Permiso necesario:** view\_users

### Request HTTP

```
GET /users/507f1f77bcf86cd799439011 HTTP/1.1
Host: localhost:3000
Authorization: Bearer tu-token-aqui
```

### Ejemplo JavaScript

```
async function obtenerUsuario(id) {
    const token = localStorage.getItem('token');

    const response = await fetch(`http://localhost:3000/users/${id}`, {
        headers: {
            'Authorization': `Bearer ${token}`
        }
    });

    return await response.json();
}
```

## Crear Usuario

**Endpoint:** POST /users

**Permiso necesario:** create\_users

### Request HTTP

```
POST /users HTTP/1.1
Host: localhost:3000
Authorization: Bearer tu-token-aqui
Content-Type: application/json

{
  "nombre": "Juan",
  "apellidos": "Pérez López",
  "fecha_nacimiento": "1995-03-15",
  "telefono": "5512345678",
  "correo_electronico": "juan@paramedia.com",
  "curp": "PELJ950315HDFRPN01",
```

```
"direccion": "Calle Principal 123",
"role": "paramedic",
"turnos": ["LV-8am3pm"],
"contrasenia": "password123",
"permissions": ["view_reports", "create_reports"]
}
```

## Ejemplo JavaScript

```
async function crearUsuario(datos) {
  const token = localStorage.getItem('token');

  const response = await fetch('http://localhost:3000/users', {
    method: 'POST',
    headers: {
      'Authorization': `Bearer ${token}`,
      'Content-Type': 'application/json'
    },
    body: JSON.stringify(datos)
  });

  return await response.json();
}

// Usar
const nuevoUsuario = {
  nombre: 'Juan',
  apellidos: 'Pérez López',
  fecha_nacimiento: '1995-03-15',
  telefono: '5512345678',
  correo_electronico: 'juan@paramedia.com',
  curp: 'PELJ950315HDFRPN01',
  direccion: 'Calle Principal 123',
  role: 'paramedic',
  turnos: ['LV-8am3pm'],
  contrasenia: 'password123',
  permissions: ['view_reports', 'create_reports']
};

crearUsuario(nuevoUsuario).then(usuario => {
  console.log('Usuario creado:', usuario._id);
});
```

## Actualizar Usuario

**Endpoint:** PUT /users/:id

**Permiso necesario:** edit\_users

## Request HTTP

```
PUT /users/507f1f77bcf86cd799439011 HTTP/1.1
Host: localhost:3000
Authorization: Bearer tu-token-aqui
Content-Type: application/json

{
  "nombre": "Juan Carlos",
  "apellidos": "Pérez López",
  "telefono": "5587654321",
  "role": "turnchief"
}
```

### Ejemplo JavaScript

```
async function actualizarUsuario(id, cambios) {
  const token = localStorage.getItem('token');

  const response = await fetch(`http://localhost:3000/users/${id}`, {
    method: 'PUT',
    headers: {
      'Authorization': `Bearer ${token}`,
      'Content-Type': 'application/json'
    },
    body: JSON.stringify(cambios)
  });

  return await response.json();
}

// Usar
actualizarUsuario('507f1f77bcf86cd799439011', {
  telefono: '5587654321',
  role: 'turnchief'
});
```

## Eliminar Usuario

Endpoint: DELETE /users/:id

Permiso necesario: delete\_users

### Request HTTP

```
DELETE /users/507f1f77bcf86cd799439011 HTTP/1.1
Host: localhost:3000
Authorization: Bearer tu-token-aqui
```

### Ejemplo JavaScript

```
async function eliminarUsuario(id) {  
  const token = localStorage.getItem('token');  
  
  const response = await fetch(`http://localhost:3000/users/${id}`, {  
    method: 'DELETE',  
    headers: {  
      'Authorization': `Bearer ${token}`  
    }  
  });  
  
  return await response.json();  
}
```

## Roles Disponibles

- admin - Administrador del sistema
- turnchief - Jefe de turno
- paramedic - Paramédico
- dispatcher - Despachador
- coordinator - Coordinador

## Permisos Disponibles

- view\_reports , create\_reports , edit\_reports , delete\_reports
- view\_users , create\_users , edit\_users , delete\_users
- view\_my\_reports , view\_turn\_reports , view\_turn\_users
- \* - Todos los permisos