Autenticación y Autorización

Usuario por Defecto

Cuando se despliega la aplicación, se crea automáticamente un usuario super administrador con las siguientes credenciales:

- **Email**: `super_admin@parozlabs.com`
- **Contraseña**: `parozlabs`
- **Rol**: `SUPER_ADMIN`

Roles de Usuario

El sistema maneja tres tipos de roles con diferentes niveles de permisos:

SUPER_ADMIN

- **Descripción**: Administrador del sistema con acceso completo
- **Permisos**: Puede gestionar clientes, usuarios, y todos los recursos del sistema
- **Acceso**: Todos los endpoints del sistema

ADMIN

- **Descripción**: Administrador de cliente con permisos limitados a su cliente
- **Permisos**: Puede gestionar usuarios, wizards, certificados, variantes y acciones remotas de su cliente
- **Acceso**: Endpoints relacionados con la gestión de recursos del cliente

USER

- **Descripción**: Usuario estándar con permisos básicos
- **Permisos**: Puede crear y gestionar wizards, certificados, variantes y acciones remotas
- **Acceso**: Endpoints de creación y gestión de recursos básicos

Tipos de Autenticación

1. Autenticación JWT (Bearer Token)

- **Header**: `Authorization: Bearer <token>`
- **Uso**: Para endpoints que requieren autenticación de usuario
- **Token**: Se obtiene mediante el endpoint de login

2. Autenticación API Key

- **Header**: `x-api-key: <api-key>`
- **Uso**: Para endpoints públicos que requieren identificación del cliente
- **Acceso**: Endpoints de wizard-drafts y evaluación de acciones remotas

Endpoints por Rol

Autenticación

Endpoint	Método	Descripción	Roles Permitidos
`/auth/login`	POST	Iniciar sesión y obtener token JWT	Sin restricción

Gestión de Clientes

Endpoint	Método	Descripción	Roles Permitidos
`/clients`	POST	Crear un nuevo cliente	SUPER_ADMIN
`/clients/:id`	PUT	Actualizar un cliente	SUPER_ADMIN
`/clients/all`	GET	Obtener todos los clientes	SUPER_ADMIN
`/clients/:id`	GET	Obtener un cliente por ID	SUPER_ADMIN
`/clients/:id`	DELETE	Eliminar un cliente (soft delete)	SUPER_ADMIN
`/clients/:id/deactivate`	PATCH	Desactivar un cliente	SUPER_ADMIN
`/clients/:id/activate`	PATCH	Activar un cliente	SUPER_ADMIN

Gestión de Usuarios

Endpoint	Método	Descripción	Roles Permitidos
`/users`	POST	Crear un nuevo usuario	ADMIN, SUPER_ADMIN
`/users`	GET	Obtener todos los usuarios	ADMIN, SUPER_ADMIN
`/users/:id`	GET	Obtener un usuario por ID	ADMIN, SUPER_ADMIN
`/users/:id`	PUT	Actualizar un usuario por ID	ADMIN, SUPER_ADMIN
`/users/:id`	DELETE	Eliminar un usuario por ID (soft delete)	ADMIN, SUPER_ADMIN
`/users/:id/deactivate`	PATCH	Desactivar un usuario por ID	ADMIN, SUPER_ADMIN
`/users/:id/activate`	PATCH	Activar un usuario por ID	ADMIN, SUPER_ADMIN
`/users/:id/reset-password`	PATCH	Resetear la contraseña de un usuario por ID	ADMIN, SUPER_ADMIN

* Los usuarios con rol **ADMIN** solo pueden operar dentro del cliente al que pertenecen.

Gestión de Credenciales

Endpoint	Método	Descripción	Roles Permitidos
`/credentials`	GET	Obtener todas las credenciales	ADMIN, SUPER_ADMIN

* Los usuarios con rol **ADMIN** solo pueden operar dentro del cliente al que pertenecen.

Gestión de Wizards

Endpoint	Método	Descripción	Roles Permitidos
`/wizards`	POST	Crear un nuevo wizard	USER, ADMIN
`/wizards`	GET	Obtener todos los wizards	USER, ADMIN
`/wizards/:id`	GET	Obtener un wizard por ID	USER, ADMIN
`/wizards/:id`	PUT	Actualizar un wizard	USER, ADMIN
`/wizards/:id`	DELETE	Eliminar un wizard (soft)	USER, ADMIN
`/wizards/:id/deactivate`	PATCH	Desactivar un wizard	USER, ADMIN
`/wizards/:id/activate`	PATCH	Activar un wizard	USER, ADMIN

Gestión de Variantes

Endpoint	Método	Descripción	Roles Permitidos
`/variants`	POST	Crear una nueva variante	USER, ADMIN
`/variants`	GET	Obtener todas las variantes	USER, ADMIN
`/variants/:id`	GET	Obtener una variante por ID	USER, ADMIN
`/variants/:id`	PUT	Actualizar una variante	USER, ADMIN
`/variants/:id`	DELETE	Eliminar una variante (soft delete)	USER, ADMIN

Gestión de Certificados

Endpoint	Método	Descripción	Roles Permitidos
`/certificates`	POST	Crear un nuevo certificado	USER, ADMIN
`/certificates`	GET	Obtener todos los certificados	USER, ADMIN
`/certificates/:id`	GET	Obtener un certificado por ID	USER, ADMIN
`/certificates/:id`	PUT	Actualizar un certificado	USER, ADMIN
`/certificates/:id`	DELETE	Eliminar un certificado (soft)	USER, ADMIN

Gestión de Acciones Remotas

Endpoint	Método	Descripción	Roles Permitidos

Endpoint	Método	Descripción	Roles Permitidos
`/remote-actions`	POST	Crear una nueva acción remota	USER, ADMIN
`/remote-actions`	GET	Obtener todas las acciones remotas	USER, ADMIN
`/remote-actions/:id`	GET	Obtener una acción remota por ID	USER, ADMIN
`/remote-actions/:id`	PUT	Actualizar una acción remota	USER, ADMIN
`/remote-actions/:id`	DELETE	Eliminar una acción remota (soft)	USER, ADMIN
`/remote-actions/evaluate`	POST	Ejecutar una acción remota	API Key

Wizard Drafts (API Key)

Endpoint	Método	Descripción	Autenticación
`/wizard-drafts`	POST	Crear un nuevo cascarón para un wizard	API Key
`/wizard-drafts/:wizardDraftId/step`	POST	Obtener los pasos de un cascarón por ID	API Key
`/wizard-drafts/:wizardDraftId/field-values`	POST	Guardar los valores de un paso	API Key
`/wizard-drafts/:wizardDraftId/field-values`	GET	Obtener los valores de un paso	API Key
`/wizard-drafts/context`	POST	Obtener los valores de un contexto	API Key

Headers Requeridos

Para endpoints con autenticación JWT:

```
Authorization: Bearer <jwt-token>
```

Para endpoints con API Key:

```
x-api-key: <api-key>
```

Flujo de Autenticación

1. **Login**: El usuario se autentica con email y contraseña en `/auth/login`
2. **Token**: Se recibe un JWT token que contiene información del usuario y cliente

3. **Autorización**: El token se incluye en el header `Authorization` para acceder a endpoints protegidos
4. **Validación**: El sistema valida el token y verifica los permisos del rol del usuario

Notas Importantes

- Los endpoints de wizard-drafts y evaluación de acciones remotas usan autenticación por API Key
- Los roles están jerárquicamente organizados: SUPER_ADMIN > ADMIN > USER
- Los usuarios solo pueden acceder a recursos de su cliente asignado
- Debe revisarse la configuración de CORS. Ante cualquier inconveniente con la API Key, asegúrese de que el dominio consumidor esté correctamente declarado dentro de allowedEndpoints para la credencial en cuestión