**LISTA DE END- POINT.**

1. **Función:** Iniciar sesión.

**Descripción:** Los administradores pueden iniciar sesión en el sistema con su número de usuario y contraseña asignada por el súper administrador que lo registro.

**Método:** Post

**Ruta:** http://localhost:3000/auth/login

**Pre-condiciones**:El usuario debe estar registrado en el sistema**.**

**Restricciones:**

* Datos de inicio de sesión erróneos (Cedula y Password).
* Que el usuario tenga estado a inactivo en el sistema.
* Que el usuario quiera acceder a un rol que no coincida con el suyo.

**Pruebas:**

* **Inicio de sesión** (Usuario estado activo admin\_super, admin\_registrador y admin\_lector)

**R:** Inicia sesión.

* **Inicio de sesión** (admin\_super, admin\_registrador y admin\_lector)

**R:** No inicia sesión.

1. **Función:** Listar todos los administradores.

**Descripción:** El usuario con rol admin\_super podrá listar todos los administradores registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/admin/search-alls

**Pre-condiciones**:El rol del usuario debe ser admin\_super**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super.

**Pruebas:**

* **Listar admins** (Usuario rol admin\_super)

**R:** Lista admins.

* **Listar admins** (Usuario rol admin\_registrador)

**R:** No Lista.

* **Listar admins** (Usuario rol admin\_lector)

**R:** No Lista.

1. **Función:** Listar un administrador específico.

**Descripción:** El usuario con rol admin\_super podrá listar un administrador especifico registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/admin/search/1

**Pre-condiciones**:El rol del usuario debe ser admin\_super**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super.

**Pruebas:**

* **Listar admins** (Usuario rol admin\_super)

**R:** Lista admins.

* **Listar admins** (Usuario rol admin\_registrador)

**R:** No Lista.

* **Listar admins** (Usuario rol admin\_lector)

**R:** No Lista.

1. **Función:** Registrar administradores.

**Descripción:** El usuario con rol admin\_super podrá registrar cualquier tipo de usuario administrador en el sistema.

**Método:** Post

**Ruta:** http://localhost:3000/api/v1/admin/register

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super**.**
* El número de documento no debe estar ligado a un usuario existente en el sistema.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super.
* Si el número de documento está ligado a un usuario existente.

**Pruebas:**

* **Registrar admin** (Usuario rol admin\_super)

**R:** Registra admind.

* **Registrar admin** (Usuario rol admin\_registrador)

**R:** No registra.

* **Registrar admin** (Usuario rol admin\_lector)

**R:** No registra.

{

  "Nombre": "María",

  "Apellido": "Lopez",

  "TipoDocumento\_idTipoDocumento": 2,

  "NumeroDocumento": "23456789",

  "Telefono": "555-1122334",

  "Correo": "maria.lopez@example.com",

  "Password": "contraseñaSegura456",

  "Estado\_idEstado": 1,

  "Rol\_idRol": 3,

  "Sexo\_idSexo": 2

}

1. **Función:** Actualizar administradores.

**Descripción:** El usuario con rol admin\_super podrá actualizar cualquier tipo de usuario administrador registrado en el sistema.

**Método:** Put

**Ruta:** http://localhost:3000/api/v1/admin/search/6

**Pre-condiciones**:El rol del usuario debe ser admin\_super**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super.
* Si el número de documento está ligado a un usuario existente.

**Pruebas:**

* **Actualizar admin** (Usuario rol admin\_super)

**R:** Actualiza admind.

* **Actualizar admin** (Usuario rol admin\_registrador)

**R:** No actualiza.

* **Actualizar admin** (Usuario rol admin\_lector)

**R:** No actualiza.

{

  "Nombre": "Carlos",

  "Apellido": "Martínez",

  "TipoDocumento\_idTipoDocumento": 1,

  "NumeroDocumento": "11223344",

  "Telefono": "555-3344556",

  "Correo": "carlos.martinez@example.com",

  "Password": "contraseñaSegura789",

  "Estado\_idEstado": 1,

  "Rol\_idRol": 2,

  "Sexo\_idSexo": 1

}

1. **Función:** Eliminar administradores.

**Descripción:** El usuario con rol admin\_super podrá eliminar cualquier tipo de usuario administrador registrado en el sistema.

**Método:** Delete

**Ruta:** http://localhost:3000/api/v1/admin/delete/6

**Pre-condiciones**:El rol del usuario debe ser admin\_super**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super.

**Pruebas:**

* **Eliminar admin** (Usuario rol admin\_super)

**R:** Elimina admind.

* **Eliminar admin** (Usuario rol admin\_registrador)

**R:** No elimina.

* **Eliminar admin** (Usuario rol admin\_lector)

**R:** No elimina.

1. **Función:** Registrar beneficiarios.

**Descripción:** El usuario con rol admin\_super y admin\_registrador podrán registrar beneficiarios en el sistema.

**Método:** Post

**Ruta:** http://localhost:3000/api/v1/beneficiary/register

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super o admin\_registrador**.**
* El número de documento no debe estar ligado a un usuario existente en el sistema.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super y admin\_registrador.
* Si el número de documento está ligado a un usuario existente.

**Pruebas:**

* **Registrar beneficiario sin coincidencias en el documento** (Usuario rol admin\_super)

**R:** Registra beneficiario.

* **Registrar beneficiario con coincidencias en el documento** (Usuario rol admin\_super)

**R:** No registra beneficiario.

* **Registrar beneficiario sin coincidencias en el documento** (Usuario rol admin\_registrador)

**R:** Registra beneficiario.

* **Registrar beneficiario con coincidencias en el documento** (Usuario rol admin\_registrador)

**R:** No registra beneficiario.

* **Registrar admin** (Usuario rol admin\_lector)

**R:** No registra.

{

  "Nombre": "Juan",

  "Apellido": "Pérez",

  "TipoDocumento\_idTipoDocumento": 1,

  "NumeroDocumento": "123456789",

  "Telefono": "3001234567",

  "Celular": "3009876543",

  "Correo": "juan.perez@example.com",

  "FechaNacimiento": "1985-05-15",

  "FechaInicio": "2024-01-01",

  "FechaFin": "2024-12-31",

  "CodigoDaneDpmto": "11001",

  "CodigoDaneMunicipio": "11001",

  "Departamento": "Cundinamarca",

  "Municipio": "Bogotá",

  "Direccion": "Calle 123 # 45-67",

  "Barrio": "Chapinero",

  "Anexo": "101",

  "Estado\_idEstado": 1,

  "Estrato\_idEstrato": 3,

  "Sexo\_idSexo": 1

}

1. **Función:** Listar todos los beneficiarios.

**Descripción:** El usuario con rol admin\_super, admin\_registrador o admin\_lector podrán listar todos los beneficiarios registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/beneficiary/search-alls

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super, admin\_registrador o admin\_lector**.**
* Debe haber beneficiarios registrados.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super, admin\_registrador y admin\_lector.
* Si no hay beneficiarios registrados.

**Pruebas:**

* **Listar beneficiarios** (Usuario rol admin\_super)

**R:** Lista beneficiarios.

* **Listar beneficiarios** (Usuario rol admin\_registrador)

**R:** Lista beneficiarios.

* **Listar beneficiarios** (Usuario rol admin\_lector)

**R:** Lista beneficiarios.

1. **Función:** Listar un beneficiario en específico (ID).

**Descripción:** El usuario con rol admin\_super, admin\_registrador o admin\_lector podrán listar un beneficiario en específico registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/beneficiary/search/1

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super, admin\_registrador o admin\_lector**.**
* El beneficiario debe estar registrado.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super, admin\_registrador y admin\_lector.
* Si el beneficiario no está registrado.

**Pruebas:**

* **Listar beneficiarios** (Usuario rol admin\_super)

**R:** Lista beneficiarios.

* **Listar beneficiarios** (Usuario rol admin\_registrador)

**R:** Lista beneficiarios.

* **Listar beneficiarios** (Usuario rol admin\_lector)

**R:** Lista beneficiarios.

1. **Función:** Listar un beneficiario en específico N° documento.

**Descripción:** El usuario con rol admin\_super, admin\_registrador o admin\_lector podrán listar un beneficiario en específico registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/beneficiary/search/beneficiary/123456789

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super, admin\_registrador o admin\_lector**.**
* El beneficiario debe estar registrado.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super, admin\_registrador y admin\_lector.
* Si el beneficiario no está registrado.

**Pruebas:**

* **Listar beneficiarios** (Usuario rol admin\_super)

**R:** Lista beneficiarios.

* **Listar beneficiarios** (Usuario rol admin\_registrador)

**R:** Lista beneficiarios.

* **Listar beneficiarios** (Usuario rol admin\_lector)

**R:** Lista beneficiarios.

1. **Función:** Actualizar beneficiarios.

**Descripción:** El usuario con rol admin\_super y admin\_registrador podrán actualizar beneficiarios en el sistema.

**Método:** Put

**Ruta:** http://localhost:3000/api/v1/beneficiary/update/1

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super o admin\_registrador**.**
* El número de documento no debe estar ligado a un usuario existente en el sistema.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super y admin\_registrador.
* Si el número de documento está ligado a un usuario existente.

**Pruebas:**

* **Actualizar beneficiario sin coincidencias en el documento** (Usuario rol admin\_super)

**R:** Actualiza beneficiario.

* **Actualizar beneficiario con coincidencias en el documento** (Usuario rol admin\_super)

**R:** No actualiza beneficiario.

* **Actualizar beneficiario sin coincidencias en el documento** (Usuario rol admin\_registrador)

**R:** Actualiza beneficiario.

* **Actualizar beneficiario con coincidencias en el documento** (Usuario rol admin\_registrador)

**R:** No actualiza beneficiario.

* **Actualizar beneficiario** (Usuario rol admin\_lector)

**R:** No actualiza.

{

  "Nombre": "Carlos",

  "Apellido": "Rodríguez",

  "TipoDocumento\_idTipoDocumento": 1,

  "NumeroDocumento": "1122334455",

  "Telefono": "3001122334",

  "Celular": "3005566778",

  "Correo": "carlos.rodriguez@example.com",

  "FechaNacimiento": "1982-11-10",

  "FechaInicio": "2024-01-01",

  "FechaFin": "2024-12-31",

  "CodigoDaneDpmto": "11001",

  "CodigoDaneMunicipio": "11001",

  "Departamento": "Cundinamarca",

  "Municipio": "Bogotá",

  "Direccion": "Avenida 68 # 23-45",

  "Barrio": "Centro Internacional",

  "Anexo": "303",

  "Estado\_idEstado": 1,

  "Estrato\_idEstrato": 4,

  "Sexo\_idSexo": 1

}

1. **Función:** Eliminar beneficiarios.

**Descripción:** El usuario con rol admin\_super podrá eliminar cualquier beneficiario registrado en el sistema.

**Método:** Delete

**Ruta:** http://localhost:3000/protected/manage/beneficiary/21

**Pre-condiciones**:El rol del usuario debe ser admin\_super**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super.

**Pruebas:**

* **Eliminar beneficiario** (Usuario rol admin\_super)

**R:** Elimina beneficiario.

* **Eliminar beneficiario** (Usuario rol admin\_registrador)

**R:** No elimina.

* **Eliminar beneficiario** (Usuario rol admin\_lector)

**R:** No elimina.

1. **Función:** Registrar documentos.

**Descripción:** El usuario con rol admin\_super y admin\_registrador podrán registrar documentos en el sistema.

**Método:** Post

**Ruta:** http://localhost:3000/api/v1/document/upload

**Pre-condiciones**:

* El rol del usuario debe ser admin\_super o admin\_registrador**.**
* El documento debe estar ligado a un beneficiario registrado en el sistema.

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super y admin\_registrador.
* Si el documento no está ligado a un beneficiario existente.

**Pruebas:**

* **Registrar documento** (Usuario rol admin\_super)

**R:** Registra documento.

* **Registrar documento** (Usuario rol admin\_registrador)

**R:** Registra documento.

* **Registrar documento** (Usuario rol admin\_lector)

**R:** No registra.

1. **Función:** Listar todos los documentos.

**Descripción:** El usuario con rol admin\_super, admin\_registrador y admin\_lector podrán listar todos los documentos registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/document/search-alls

**Pre-condiciones**:El rol del usuario debe ser admin\_super, admin\_registrador o admin\_lector**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super, admin\_registrador o admin\_lector.

**Pruebas:**

* **Listar documentos** (Usuario rol admin\_super)

**R:** Lista documentos.

* **Listar documentos** (Usuario rol admin\_registrador)

**R:** Lista documentos.

* **Listar documentos** (Usuario rol admin\_lector)

**R:** Lista documentos.

1. **Función:** Listar un documento en específico.

**Descripción:** El usuario con rol admin\_super, admin\_registrador y admin\_lector podrán listar un documento registrados en el sistema.

**Método:** Get

**Ruta:** http://localhost:3000/api/v1/document/search/2

**Pre-condiciones**:El rol del usuario debe ser admin\_super, admin\_registrador o admin\_lector**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super, admin\_registrador o admin\_lector.

**Pruebas:**

* **Listar documentos** (Usuario rol admin\_super)

**R:** Lista documentos.

* **Listar documentos** (Usuario rol admin\_registrador)

**R:** Lista documentos.

* **Listar documentos** (Usuario rol admin\_lector)

**R:** Lista documentos.

1. **Función:** Listar todos los documentos de un beneficiario en específico.

**Descripción:** El usuario con rol admin\_super, admin\_registrador y admin\_lector podrán listar todos los documentos de un beneficiario en específico registrados en el sistema (id).

**Método:** Post

**Ruta:** http://localhost:3000/api/v1/document/search/beneficiary/3

**Pre-condiciones**:El rol del usuario debe ser admin\_super o admin\_registrador**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super, admin\_registrador o admin\_lector.

**Pruebas:**

* **Listar documentos** (Usuario rol admin\_super)

**R:** Lista documentos.

* **Listar documentos** (Usuario rol admin\_registrador)

**R:** Lista documentos.

* **Listar documentos** (Usuario rol admin\_lector)

**R:** Lista documentos.

1. **Función:** Eliminar un documento de un beneficiario en específico.

**Descripción:** El usuario con rol admin\_super o admin\_registrador podrán eliminar un documento de un beneficiario en específico registrado en el sistema (id documento).

**Método:** Delete

**Ruta:** http://localhost:3000/api/v1/document/delete/1

**Pre-condiciones**:El rol del usuario debe ser admin\_super o admin\_registrador**.**

**Restricciones:**

* Si el rol del usuario es diferente a admin\_super o admin\_registrador.

**Pruebas:**

* **Eliminar documentos** (Usuario rol admin\_super)

**R:** Elimina documentos.

* **Eliminar documentos** (Usuario rol admin\_registrador)

**R:** Elimina documentos.

* **Listar documentos** (Usuario rol admin\_lector)

**R:** No elimina documentos.

**Para las siguientes rutas solamente tiene acceso el usuario admin\_super:**

1. **Consultar todos los registros del historial:**

* Get http://localhost:3000/api/v1/change/search-alls

1. **Consultar un registro del historial:**

* Get http://localhost:3000/api/v1/change/search/6

1. **Borrar todos los registros del historial:**

* Del http://localhost:3000/api/v1/change/delete-alls

1. **Borrar un registro del historial:**

* Del http://localhost:3000/api/v1/change/delete/6

1. **Registrar estados:**

* Post http://localhost:3000/api/v1/state/register

{

    "Estado": "Viejo"

}

1. **Modificar un estado:**

* Put http://localhost:3000/api/v1/state/update/5

{

    "Estado": "Viejo"

}

1. **Consultar todos los estados:**

* Get http://localhost:3000/api/v1/state/search-alls

1. **Consultar todos los estados:**

* Get http://localhost:3000/api/v1/state/search/2

1. **Eliminar un estado:**

* Del http://localhost:3000/api/v1/state/delete/5

1. **Registrar estrato:**

* Post http://localhost:3000/api/v1/stratum/register

{

    "Estrato": "Viejo"

}

1. **Modificar estrato:**

* Put http://localhost:3000/api/v1/stratum/update/7

{

    "Estrato": "Viejo"

}

1. **Consultar todos los estratos:**

* Get http://localhost:3000/api/v1/stratum/search-alls

1. **Consultar un estrato:**

* Get http://localhost:3000/api/v1/stratum/search/7

1. **Eliminar estrato:**

* Del http://localhost:3000/api/v1/stratum/delete/7

1. **Registrar un tipo:**

* Post http://localhost:3000/api/v1/document-type/register

{

    "TipoDocumento": "Viejo"

}

1. **Modificar un tipo:**

* Put http://localhost:3000/api/v1/document-type/update/4

{

    "TipoDocumento": "Viejo"

}

1. **Consutar todos los tipos:**

* Get http://localhost:3000/api/v1/document-type/search-alls

1. **Consultar un tipo:**

* Get http://localhost:3000/api/v1/document-type/search/3

1. **Eliminar un tipo:**

* Del http://localhost:3000/api/v1/document-type/delete/4

1. **Registrar rol:**

* Post http://localhost:3000/api/v1/role/register

{

    "Rol": "Viejo"

}

1. **Modificar Rol:**

* Put http://localhost:3000/api/v1/role/update/4

{

    "Rol": "Viejo"

}

1. **Consultar todos los roles:**

* Get http://localhost:3000/api/v1/role/search-alls

1. **Consultar un rol:**

* Get http://localhost:3000/api/v1/role/search/3

1. **Eliminar rol:**

* Del http://localhost:3000/api/v1/role/delete/4

1. **Registrar sexo:**

* Post http://localhost:3000/api/v1/sex/register

{

    "Sexo": "Viejo"

}

1. **Modificar sexo:**

* Put http://localhost:3000/api/v1/sex/update/4

{

    "Sexo": "Viejo"

}

1. **Consultar todos los sexos:**

* Get http://localhost:3000/api/v1/sex/search-alls

1. **Consultar un sexo:**

* Get http://localhost:3000/api/v1/sex/search/3

1. **Eliminar un sexo:**

* Del http://localhost:3000/api/v1/sex/delete/4