

Universidad ORT Uruguay

Facultad de Ingeniería



Obligatorio 1 Diseño de Aplicaciones 2

Enzo Izquierdo (283145)

Manuel Graña (285727)

Martín Salaberry (294238)

[Repositorio](#)

2024

Índice

Cumplimiento de los Criterios REST	4
Uniform Interface	4
Stateless	4
Cacheable	4
Client-Server	4
Layered System	5
Autenticación de Requests	5
Funcionamiento del Mecanismo de Autenticación	5
¿Qué es "Bearer"?	5
Validación del Token	6
Códigos de Estado HTTP	6
Body de Código de Error	7
Elección de nombres de endpoints	7
Elección de métodos HTTP	7
Base URL	8
Endpoints para Administradores	8
1) Crear cuenta de administrador	8
2) Eliminar cuenta de administrador	9
3) Crear cuenta de dueño de empresa	11
4) Listar todas las cuentas	12
5) Listar todas las empresas	14
Manejo de notificaciones	15
6) Obtener las notificaciones de un usuario	15
Endpoints para Dueños de Empresas	17
7) Crear empresa	17
8) Registrar cámara de seguridad	18
9) Registrar sensor de ventana	20
Endpoints para Personas No Autenticadas	22
10) Autenticación	22
11) Crear cuenta de dueño de hogar	23
Endpoints para Personas Autenticadas	24
12) Listar dispositivos	24
13) Listar tipos de dispositivos soportados	26
Endpoints para Dueños de Hogar	27
14) Crear hogar	27
15) Agregar miembros al hogar	28
16) Asociar dispositivos al hogar	30
17) Listar miembros del hogar	31
18) Listar dispositivos del hogar	32
19) Configurar notificaciones para miembros	33
Endpoints para Dispositivos	34
20) Conectar/Desconectar sensor	34

21) Crear notificación de ventana abierta de sensor	35
22) Crear notificación de ventana cerrada de sensor	36
23) Conectar/Desconectar cámara	36
24) Crear notificación de detección de movimiento de cámara	37
25) Crear notificación de detección de persona de cámara	38

Evidencia del diseño y especificación de la API

Cumplimiento de los Criterios REST

Uniform Interface

La API sigue el principio de tener una interfaz uniforme, donde cada recurso tiene una representación clara y accesible mediante un endpoint consistente. Por ejemplo, si quiero acceder a los administradores o a los dueños de empresa, voy directamente a los endpoints `/admins` o `/business owners`, ya que aunque en el código ambos sean instancias de `User`, estos son recursos distintos.

Stateless

Cada solicitud del cliente contiene toda la información necesaria para que el servidor la procesa, sin requerir que el servidor mantenga estado de sesión entre las solicitudes.

- **Ejemplo:**
 - Endpoint: [POST /admins](#)
 - Descripción: El cliente envía la solicitud junto con toda la información en el cuerpo, y el servidor no necesita mantener datos de la solicitud anterior.

Cacheable

Las respuestas de la API incluyen encabezados de control de caché, lo que permite al cliente almacenar respuestas y reutilizarlas para mejorar el rendimiento.

- **Ejemplo:**
 - Endpoint: [GET /device_types](#)
 - Descripción: La respuesta se puede cachear por un mes.

Client-Server

La separación de responsabilidades entre cliente y servidor está clara. El cliente no conoce la lógica del servidor y solo interactúa con él a través de los servicios.

Los controladores reciben “requests”, es decir, structs con la información necesaria para hacer la consulta, y retornan “responses”, también structs solo con la información que necesita el cliente y compuestos de tipos primitivos.

Layered System

La API sigue una arquitectura en capas, permitiendo la existencia de intermediarios entre el cliente y el servidor, ya que el cliente no tiene manera de saber si la respuesta le está llegando del servidor final o de un intermediario. Actualmente no hay ninguno.

Autenticación de Requests

Cuando un usuario inicia sesión utilizando el endpoint de autenticación ([POST /auth](#)), se genera un token único (GUID) que se almacena en la base de datos junto con su fecha de expiración, la cual está configurada para una duración de **una hora**. Este token actúa como un identificador de sesión que permite al usuario acceder a recursos protegidos de la API sin necesidad de volver a autenticarse. Dicho Token guarda además una referencia a User para poder validar posteriormente sus permisos.

Funcionamiento del Mecanismo de Autenticación

1. Inicio de Sesión:

- El usuario envía sus credenciales al endpoint **/auth**.
- Si las credenciales son válidas, el servidor genera un token GUID y lo almacena en la base de datos con un tiempo de expiración de una hora.

2. Uso del Token en Solicitudes:

- Cada vez que el usuario realiza una petición a un endpoint que requiere autenticación, debe incluir el token en el **header** de la solicitud.
- El formato del header es **Authorization: Bearer [token]**.

¿Qué es "Bearer"?

El término **Bearer** es un tipo de esquema de autorización utilizado en el estándar OAuth 2.0. En este contexto, el esquema Bearer indica que el cliente debe presentar el token de acceso

(en este caso, el GUID) para ser autenticado. El término "bearer" implica que cualquier entidad que posea el token tiene derecho a acceder a los recursos protegidos, lo que significa que no se requiere información adicional para verificar la identidad del cliente.

Ejemplo de Header:

Authorization: Bearer a1b2c3d4-e5f6-7g8h-9i10-j11k12l13m14

Validación del Token

- Cuando el servidor recibe una solicitud con el header **Authorization**, verifica el token para asegurarse de que:
 - Es un token válido y corresponde a un usuario en la base de datos.
 - No ha expirado.

Códigos de Estado HTTP

Siguiendo las recomendaciones de *Brian Mulloy en "Web API Design" (apigee)*, limitamos nuestra API a 8 códigos de estado HTTP. Esta decisión se basa en que, de los más de 70 códigos existentes, la mayoría de desarrolladores no los conocen todos. Restringir los códigos a un subconjunto esencial permite que los consumidores de la API se enfoquen en desarrollar sus aplicaciones, en lugar de aprender numerosos códigos de error. Los códigos de error retornados son los siguientes:

- 200 OK
- 304 Not Modified
- 400 Bad Request
- 401 Unauthorized
- 403 Forbidden
- 404 Not Found
- 409 Conflict
- 500 Internal Server Error

Body de Código de Error

Para cualquier endpoint de la API, utilizamos un body de error unificado, como se describe a continuación. Se procura que esté alineado con el código de estado HTTP que se retorna. De esta forma, los clientes de la API pueden anticipar y manejar los errores de manera uniforme, sin tener que implementar diferentes estructuras de validación para cada caso. En el campo message, ofrecemos una descripción legible del error, facilitando la comprensión de lo que ocurrió sin necesidad de hacer suposiciones basadas únicamente en el código de error.

```
{ "innerCode": string, "message": string }
```

Elección de nombres de endpoints

Los nombres de los endpoints fueron elegidos para reflejar de manera clara y directa los recursos que gestionan, alineándose con los principios REST. Por ejemplo, los endpoints para Administradores (/admins) y Dueños de Empresas (/business_owners) indican los roles de usuario específicos que se van a crear. Los nombres como Negocios (/businesses) y Hogares y Miembros (/homes, /homes/{homeId}/members) representan claramente sus respectivos recursos, mientras que Dispositivos y Sensores (/devices, /sensors, /cameras) se enfocan en tipos de dispositivos manejables. El endpoint de Autenticación (/auth) permite obtener tokens para futuras solicitudes, y Notificaciones (/notifications) comunica que el recurso es sobre eventos de dispositivos.

Elección de métodos HTTP

Los métodos HTTP fueron seleccionados basándose en las operaciones que realizan sobre los recursos de la API:

- **POST** se utiliza en la creación de nuevas entidades, como cuentas de administrador o dueño de empresa, así como para acciones que generan recursos, tales como notificaciones o registros de dispositivos.
- **GET** se utiliza para obtener recursos como listar cuentas, empresas o dispositivos. No altera los recursos.
- **DELETE** está especificado para eliminar recursos, lo utilizamos para la eliminación de cuentas de administrador

- **PATCH** se emplea para realizar actualizaciones parciales de recursos, como configurar notificaciones para miembros del hogar. No usamos PUT ya que se suele utilizar para modificar el recurso entero, y en este caso solo estamos alterando la configuración de las notificaciones.

Base URL

<https://api.homeconnect.com/v1>

Esta es la URL base para todos los endpoints de la API. A continuación, se describen los endpoints específicos con sus respectivos métodos y rutas.

Endpoints para Administradores

1) Crear cuenta de administrador

POST /admins

Ejemplo: **POST** <https://api.homeconnect.com/v1/admins>

Header: Authorization

Crea una nueva cuenta de administrador.

Cuerpo de la solicitud (Request Body):

- name (requerido)
 - Descripción: nombre del administrador
 - Tipo: cadena de texto
 - Ejemplo: firstName=John
- surname (requerido)
 - Descripción: apellido del administrador
 - Tipo: cadena de texto
 - Ejemplo: lastName=Doe
- email (requerido)
 - Descripción: email del administrador, debe cumplir con formato predefinido
 - Tipo: cadena de texto

- Ejemplo: email=johnndoe@ejemplo.com
- password (requerido)
 - Descripción: contraseña para la cuenta del administrador, debe cumplir requisitos de seguridad predefinidos
 - Tipo: cadena de texto
 - Ejemplo: password=JohnDoe#2001

Respuestas:

- 200 OK
 - Descripción: la cuenta del administrador fue creada exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "id": string
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para crear cuentas de administrador
- 409 Conflict
 - Descripción: ya existe una cuenta de administrador con el mismo email
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

2) Eliminar cuenta de administrador

DELETE /admins/{adminId}

Header: Authorization

Elimina una cuenta de administrador basada en su ID.

Ejemplo:

DELETE <https://api.homeconnect.com/v1/admins/{adminId}>

Parámetros de ruta:

- adminId (requerido)
 - Descripción: identificador del administrador que se desea eliminar
 - Tipo: GUID
 - Ejemplo: /admins/1

Respuestas:

- 200 OK
 - Descripción: la cuenta de administrador fue eliminada exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{  
  "id": string  
}
```

- 400 Bad Request
 - Descripción: la solicitud no es válida; el adminId no está en el formato correcto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para eliminar la cuenta de administrador
- 404 Not Found
 - Descripción: la cuenta de administrador con el adminId especificado no fue encontrada
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

3) Crear cuenta de dueño de empresa

POST `/business_owners`

Ejemplo:

POST `https://api.homeconnect.com/v1/business_owners`

Header: Authorization

Crea una cuenta para un dueño de empresa.

Cuerpo de la solicitud (Request Body):

- name (requerido)
 - Descripción: nombre del administrador
 - Tipo: cadena de texto
 - Ejemplo: firstName=John
- surname (requerido)
 - Descripción: apellido del administrador
 - Tipo: cadena de texto
 - Ejemplo: lastName=Doe
- email (requerido)
 - Descripción: email del administrador, debe cumplir con formato predefinido
 - Tipo: cadena de texto
 - Ejemplo: email=jhondoe@ejemplo.com
- password (requerido)
 - Descripción: contraseña para la cuenta del administrador, debe cumplir requisitos de seguridad predefinidos
 - Tipo: cadena de texto
 - Ejemplo: password=JohnDoe#2001

Respuestas:

- 200 OK
 - Descripción: la cuenta de dueño de empresa fue creada exitosamente
 - Cuerpo de la respuesta (Response body):

```
{
  "id": string
}
```

- 400 Bad Request
 - Descripción: la solicitud no es válida; campos faltantes o mal formateados
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene permisos para crear cuentas de dueño de empresa
- 409 Conflict
 - Descripción: ya existe una cuenta de dueño de empresa con este email
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

4) Listar todas las cuentas

GET /users

Ejemplo:

GET https://api.homeconnect.com/v1/users

Header: Authorization

Obtiene una lista paginada de todas las cuentas registradas.

Parámetros de Consulta (Query Parameters):

- fullName (opcional):
 - Descripción: Filtra las cuentas por el nombre completo del usuario
 - Tipo: cadena de texto
 - Ejemplo: fullName=John Doe
- role (opcional):
 - Descripción: Filtra las cuentas por el rol del usuario
 - Tipo: cadena de texto
 - Ejemplo: role=Business Owner

- page (opcional)
 - Descripción: Especifica el número de la página de resultados que se desea obtener. Utilizado para la paginación.
 - Tipo: Entero
 - Valor predeterminado: 1
 - Ejemplo: page=2
- pageSize (opcional):
 - Descripción: Especifica el número de resultados por página.
 - Tipo: Entero.
 - Valor predeterminado: 10.
 - Ejemplo: pageSize=50

Respuestas:

- 200 OK
 - Descripción: la lista de cuentas fue obtenida exitosamente
 - Cuerpo de la respuesta de ejemplo (Response Body):

```
{
  "users": [
    {
      "id": string,
      "name": string,
      "surname": string,
      "role": string,
      "createdAt": string
    }
  ],
  "pagination": {
    "page": int,
    "pageSize": int,
    "totalPages": int
  }
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, el formato de algún parámetro es incorrecto o la combinación de parámetros no es válida
- 401 Unauthorized

- Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para listar las cuentas de usuario
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

5) Listar todas las empresas

GET /businesses

Ejemplo:

GET https://api.homeconnect.com/v1/businesses

Header: Authorization

Parámetros de Consulta (Query Parameters):

- name (opcional):
 - Descripción: Filtra las empresas por el nombre de la empresa.
 - Tipo: Cadena de texto.
 - Ejemplo: name=SmartTech
- ownerName (opcional):
 - Descripción: Filtra las empresas por el nombre completo del dueño.
 - Tipo: Cadena de texto.
 - Ejemplo: ownerName=John Doe
- page (opcional):
 - Descripción: Especifica el número de la página de resultados que se desea obtener. Utilizado para la paginación.
 - Tipo: Entero.
 - Valor predeterminado: 1.
 - Ejemplo: page=2
- pageSize (opcional):
 - Descripción: Especifica el número de resultados por página.
 - Tipo: Entero.
 - Valor predeterminado: 10.
 - Ejemplo: pageSize=50

Respuestas:

- 200 OK
 - Descripción: la lista de empresas fue obtenida exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "businesses": [
    {
      "name": string,
      "ownerName": string,
      "ownerSurname": string,
      "ownerEmail": string,
      "rut": string
    }
  ],
  "pagination": {
    "page": int,
    "pageSize": int,
    "totalPages": int
  }
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, el formato de algún parámetro es incorrecto o la combinación de parámetros no es válida
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para listar las empresas
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

Manejo de notificaciones

6) Obtener las notificaciones de un usuario

GET /notifications

Ejemplo:

GET https://api.homeconnect.com/v1/notifications

Header: Authorization

Obtiene una lista de notificaciones que pueden ser filtradas por casa, tipo de dispositivo, fecha de creación, leídas/no leídas.

Parámetros de Consulta (Query Parameters):

- device (opcional):
 - Descripción: Filtra por tipo de dispositivo
 - Tipo: Cadena de texto
 - Valores posibles: device = camera, device = sensor
- dateCreated (opcional):
 - Descripción: Filtra por fecha de creación.
 - Tipo: Cadena de texto formato dd/mm/yyyy.
 - Ejemplo: dateCreated = 12/11/2024
- read (opcional):
 - Descripción: Filtra por fecha de creación.
 - Tipo: Booleano
 - Ejemplo: read = True

Respuestas:

- 200 OK
 - Descripción: la lista de notificaciones fue obtenida exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "notifications": [
    {
      "event": string,
      "deviceId": string,
      "read": bool,
      "dateCreated": string
    }
  ]
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, el formato de algún parámetro es incorrecto o la combinación de parámetros no es válida

- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para listar las notificaciones
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

Endpoints para Dueños de Empresas

7) Crear empresa

Header: Authorization

POST /businesses

Ejemplo:

POST <https://api.homeconnect.com/v1/businesses>

Crea una empresa y la asocia a la cuenta de un dueño, correspondiente al token de Authorization

Cuerpo de la solicitud (Request Body):

- name (requerido)
 - Descripción: Nombre de la empresa a crear
 - Tipo: Cadena de texto
 - Ejemplo: name=OSE
- logo (requerido)
 - Descripción: URL o ruta a el logotipo de la empresa
 - Tipo: Cadena de texto
 - Ejemplo:

logo=http://www.ose.com.uy/media/images/logo_entero_banner.svg?timestamp=20171206191445
- rut (requerido):
 - Descripción: Rol Unico Tributario de la empresa, actua como identificador unico
 - Tipo: Entero

- Ejemplo: RUT=306869575

Respuestas:

- 200 OK
 - Descripción: la empresa fue creada exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{  
  "rut": string  
}
```
- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para crear la empresa
- 409 Conflict
 - Descripción: ya existe una empresa con el mismo RUT
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

8) Registrar cámara de seguridad

Header: Authorization

POST /cameras

Ejemplo:

POST <https://api.homeconnect.com/v1/cameras>

Registra una cámara de seguridad.

Cuerpo de la solicitud (Request Body):

- name (requerido):
 - Descripción: Nombre de la cámara de seguridad.

- Tipo: Cadena de texto.
 - Ejemplo: name=TPLink1234
- modelName (requerido):
 - Descripción: Número de modelo de la cámara a registrar.
 - Tipo: Entero
 - Ejemplo: modelName=1234
- interior (requerido):
 - Descripción: Indica si la cámara es para uso interior.
 - Tipo: Booleano.
 - Ejemplo: interior=true
- exterior (requerido):
 - Descripción: Indica si la cámara es para uso exterior.
 - Tipo: Booleano.
 - Ejemplo: exterior=false
- description (requerido):
 - Descripción: Descripción detallada de la cámara de seguridad.
 - Tipo: Cadena de texto.
 - Ejemplo: description=Cámara de seguridad para exteriores con detección de movimiento y vision nocturna
- mainPhoto (requerido):
 - Descripción: URL de imagen principal utilizada para comercializar la cámara.
 - Tipo: Cadena de texto.
 - Ejemplo: mainPhoto = https://ejemplo1.com/foto.jpg
- secondaryPhotos (opcional):
 - Descripción: Lista de URLs de imágenes secundarias utilizadas para comercializar la cámara.
 - Tipo: Array de cadenas de texto.
 - Ejemplo: secondaryPhotos=[https://ejemplo1.com/foto.jpg, https://ejemplo2.com/foto2.jpg]
- motionDetection (requerido):
 - Descripción: Capacidad de detección de movimiento de la cámara.
 - Tipo: Booleano

- Ejemplo: motionDetection = true
- personDetection (requerido):
 - Descripción: Capacidad de detección de personas de la cámara.
 - Tipo: Booleano.
 - Ejemplo: personDetection = false

Respuestas

- 200 OK
 - Descripción: la cámara fue registrada exitosamente
 - Cuerpo de la respuesta (Response Body):


```
{
  "id": string
}
```
- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para registrar la cámara
- 409 Conflict
 - Descripción: ya existe una cámara con el mismo número de modelo
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

9) Registrar sensor de ventana

Header: Authorization

POST /sensors

Ejemplo:

POST <https://api.homeconnect.com/v1/sensors>

Registra un sensor de ventana.

Cuerpo de la solicitud (Request Body):

- name (requerido):
 - Descripción: Nombre del sensor de ventana.
 - Tipo: Cadena de texto.
 - Ejemplo: name=Sensor Ventana Principal
- modelNumber (requerido):
 - Descripción: Número de modelo del sensor.
 - Tipo: Entero.
 - Ejemplo: modelNumber=5678
- description (requerido):
 - Descripción: Descripción del sensor de ventana.
 - Tipo: Cadena de texto.
 - Ejemplo: description=Sensor de ventana gran angular
- mainPhoto (requerido):
 - Descripción: Imagen principal del sensor de ventana
 - Tipo: Cadena de texto
 - Ejemplo: mainPhoto=https://ejemplo.com/foto.jpg
- secondaryPhotos (opcional):
 - Descripción: Lista de URLs de imágenes utilizadas para comercializar el sensor de ventana.
 - Tipo: Array de cadenas de texto.
 - Ejemplo: secondaryPhotos=https://ejemplo1.com/foto.jpg,
<https://ejemplo2.com/foto2.jpg>

Respuestas:

- 200 OK
 - Descripción: el sensor de ventana fue registrado exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{  
  "id": string  
}
```

- 400 Bad Request

- Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para registrar el sensor
- 409 Conflict
 - Descripción: ya existe un sensor con el mismo número de modelo
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

Endpoints para Personas No Autenticadas

10) Autenticación

POST /auth

Autentica a un usuario mediante correo y contraseña.

Cuerpo de la solicitud (Request Body):

- email (requerido):
 - Descripción: Correo electrónico del usuario que se está autenticando.
 - Tipo: Cadena de texto.
 - Ejemplo: email=johndoe@example.com
- password (requerido):
 - Descripción: Contraseña del usuario.
 - Tipo: Cadena de texto.
 - Ejemplo: password=JohnDoe#2001

Respuestas:

- 200 OK
 - Descripción: autenticación exitosa, el usuario ha sido autenticado correctamente
 - Cuerpo de la respuesta (Response Body):

```
{  
  "token": string  
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el correo electrónico o la contraseña son incorrectos
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

11) Crear cuenta de dueño de hogar

POST /home_owners

Ejemplo:

POST https://api.homeconnect.com/v1/home_owners

Crea una cuenta de dueño de hogar.

Cuerpo de la solicitud (Request Body):

- name (requerido):
 - Descripción: Nombre del dueño de hogar.
 - Tipo: Cadena de texto.
 - Ejemplo: firstName=John
- surname (requerido):
 - Descripción: Apellido del dueño de hogar.
 - Tipo: Cadena de texto.
 - Ejemplo: lastName=Doe
- profilePicture (requerido):
 - Descripción: URL de la foto de perfil del dueño de hogar.
 - Tipo: Cadena de texto.
 - Ejemplo: profilePicture=https://ejemplo.com/foto.jpg
- email (requerido):
 - Descripción: Correo electrónico único del dueño de hogar. Debe cumplir con formato predefinido.

- Tipo: Cadena de texto.
- Ejemplo: email=johndoe@ejemplo.com
- password (requerido):
 - Descripción: Contraseña para la cuenta del dueño de hogar. Debe cumplir con los requisitos de seguridad predefinidos.
 - Tipo: Cadena de texto.
 - Ejemplo: password=JohnDoe#2001

Respuestas:

- 200 OK
 - Descripción: la cuenta del dueño de hogar fue creada exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{  
  "id": string  
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 409 Conflict
 - Descripción: ya existe una cuenta de dueño de hogar con el mismo email
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

Endpoints para Personas Autenticadas

12) Listar dispositivos

GET `/devices`

Header: Authorization

Ejemplo:

GET `https://api.homeconnect.com/v1/devices`

Obtiene una lista paginada de los dispositivos registrados.

Parámetros de Consulta (Query Parameters):

- name (opcional):
 - Descripción: Filtra por nombre de dispositivo.
 - Tipo: Cadena de texto.
 - Ejemplo: name=example_cam1
- type (opcional):
 - Descripción: Filtra por tipo de dispositivo.
 - Tipo: Cadena de texto.
 - Valores posibles: type=Camera
- model (opcional):
 - Descripción: Filtra por número de modelo.
 - Tipo: Entero.
 - Ejemplo: model=123
- businessName (opcional):
 - Descripción: Filtra por nombre de la empresa.
 - Tipo: Cadena de texto
 - Ejemplo: businessName=ExampleBusiness
- page (opcional):
 - Descripción: Especifica el número de la página de resultados que se desea obtener. Utilizado para la paginación.
 - Tipo: Entero.
 - Valor predeterminado: 1.
 - Ejemplo: page=2
- pageSize (opcional):
 - Descripción: Especifica el número de resultados por página.
 - Tipo: Entero.
 - Valor predeterminado: 10.
 - Ejemplo: pageSize= 50

Respuestas:

- 200 OK
 - Descripción: la lista de dispositivos fue obtenida exitosamente
 - Cuerpo de la respuesta (Response Body):

```

{
  "devices": [
    {
      "id": string,
      "name": string,
      "businessName": string,
      "type": string,
      "modelName": int,
      "photo": string
    }
  ],
  "pagination": {
    "page": int,
    "pageSize": int,
    "totalPages": int
  }
}

```

- 400 Bad Request
 - Descripción: solicitud no válida, el formato de algún parámetro es incorrecto o la combinación de parámetros no es válida
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

13) Listar tipos de dispositivos soportados

GET /device_types

Header: Authorization

Obtiene una lista de los tipos de dispositivos soportados

Cache-Control: public, max-age=2592000 (cacheable por un mes)

Respuestas:

- 200 OK
 - Descripción: la lista de tipos de dispositivos soportados fue obtenida exitosamente
 - Cuerpo de la respuesta (Response Body):

```

{

```

```
"deviceTypes": [  
  "Sensor",  
  "Camera"  
]
```

- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

Endpoints para Dueños de Hogar

14) Crear hogar

Header: Authorization

POST /homes

Ejemplo:

POST <https://api.homeconnect.com/v1/homes>

Registra un hogar.

Cuerpo de la solicitud (Request Body):

- address (requerido):
 - Descripción: Dirección física del hogar.
 - Tipo: Cadena de texto.
 - Ejemplo: address=123 Avenida Italia, Montevideo, Uruguay
- latitude (requerido):
 - Descripción: Latitud en formato de coordenadas
 - Tipo: Double.
 - Ejemplo: 20.00
- longitude (requerido):
 - Descripción: Latitud en formato de coordenadas
 - Tipo: Double.
 - Ejemplo: 20.00
- maxMembers (requerido):

- Descripción: El número máximo de miembros que residen en el hogar.
- Tipo: Enter positivo.
- Ejemplo: maxMembers=4

Respuestas:

- 200 OK

- Descripción: el hogar fue registrado exitosamente
- Cuerpo de la respuesta (Response Body):

```
{
  "id": string
}
```

- 400 Bad Request

- Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto

- 401 Unauthorized

- Descripción: el token de autorización es inválido o está ausente

- 403 Forbidden

- Descripción: el usuario autenticado no tiene los permisos necesarios para registrar un hogar

- 409 Conflict

- Descripción: ya existe un hogar registrado con la misma dirección

- 500 Internal Server Error

- Descripción: ocurrió un error inesperado en el servidor

15) Agregar miembros al hogar

Header: Authorization

POST /homes/{homesId}/members

Ejemplo:

POST <https://api.homeconnect.com/v1/homes/12345/members>

Agrega un home owner existente a un hogar.

Cuerpo de la solicitud (Request Body):

- **userId (requerido):**
 - Descripción: Id del dueño de hogar a agregar.
 - Tipo: GUID
 - Ejemplo: userId=123456789
- **canAddDevices (opcional, por defecto *false*):**
 - Descripción: Permiso para agregar dispositivos al hogar.
 - Tipo: bool
 - Ejemplo: canAddDevices=true
- **canListDevices (opcional, por defecto *false*):**
 - Descripción: Permiso para listar los dispositivos del hogar.
 - Tipo: bool
 - Ejemplo: canListDevices=false

Respuestas:

- **200 OK**
 - Descripción: el miembro fue agregado exitosamente al hogar
 - Cuerpo de la respuesta (Response Body):

```
{
  "homeId": string,
  "memberId": string
}
```
- **400 Bad Request**
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- **401 Unauthorized**
 - Descripción: el token de autorización es inválido o está ausente
- **403 Forbidden**
 - Descripción: el usuario autenticado no tiene los permisos necesarios para agregar miembros al hogar
- **404 Not Found**
 - Descripción: no se encontró el hogar o el dueño de hogar indicado
- **409 Conflict**
 - Descripción: el miembro ya está agregado al hogar o la casa está llena

- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

16) Asociar dispositivos al hogar

Header: Authorization

POST /homes/{homeId}/devices

Ejemplo:

POST https://api.homeconnect.com/v1/homes/12345/devices

Asocia dispositivos al hogar.

Cuerpo de la solicitud (Request Body):

- deviceId (requerido):
 - Descripción: IDs de los dispositivos que se pueden asociar al hogar.
 - Tipo: Array de GUIDs.
 - Ejemplo: deviceId=[“a472c849-06a5-41d8-9b6b-154ea66bf7542”, “a472c849-06a5-41d8-9b6b-154ea66bf754”]

Respuestas:

- 200 OK
 - Descripción: los dispositivos fueron asociados exitosamente al hogar
 - Cuerpo de respuesta (Response body):

```
{
  "homeId": string,
  "deviceIds": [string]
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden

- Descripción: el usuario autenticado no tiene los permisos necesarios para asociar dispositivos al hogar
- 404 Not Found
 - Descripción: no se encontró el hogar
- 409 Conflict
 - Descripción: uno o más dispositivos ya están asociados a otro hogar
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

17) Listar miembros del hogar

Header: Authorization

GET /homes/{homeId}/members

Obtiene una lista de los miembros del hogar.

Respuestas:

- 200 OK
 - Descripción: la lista de miembros del hogar fue obtenida exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "members": [
    {
      "id": string,
      "name": string,
      "surname": string,
      "photo": string,
      "canAddDevices": bool,
      "canListDevices": bool,
      "shouldBeNotified": bool
    }
  ]
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, el formato del homeId es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente

- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para listar los miembros del hogar
- 404 Not Found
 - Descripción: no se encontró el hogar indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

18) Listar dispositivos del hogar

Header: Authorization

GET /homes/{homeId}/devices

Ejemplo:

GET https://api.homeconnect.com/v1/homes/12345/devices

Obtiene una lista de los dispositivos asociados a un hogar.

Respuestas:

- 200 OK
 - Descripción: la lista de dispositivos asociados al hogar fue obtenida exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "devices": [
    {
      "hardwareId": string,
      "name": string,
      "businessName": string,
      "type": string,
      "modelNumber": int,
      "photo": string
    }
  ]
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, el formato del homeId es incorrecto
- 401 Unauthorized

- Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para listar los dispositivos del hogar
- 404 Not Found
 - Descripción: no se encontró el hogar indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

19) Configurar notificaciones para miembros

Header: Authorization

PATCH /members/{memberId}/notifications

Ejemplo:

PATCH

<https://api.homeconnect.com/v1/members/12345/notifications>

Configura si un miembro recibirá notificaciones de los dispositivos o no.

Cuerpo de la solicitud (Request Body):

- shouldBeNotified (requerido):
 - Descripción: Permiso para recibir notificaciones de los dispositivos del hogar.
 - Tipo: bool
 - Ejemplo: shouldBeNotified=false

Respuestas:

- 200 OK
 - Descripción: la configuración de notificaciones para el miembro fue actualizada exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "memberId": id,
```

```
"shouldBeNotified": bool
}
```

- 400 Bad Request
 - Descripción: solicitud no válida, faltan campos requeridos o el formato del request body es incorrecto
- 401 Unauthorized
 - Descripción: el token de autorización es inválido o está ausente
- 403 Forbidden
 - Descripción: el usuario autenticado no tiene los permisos necesarios para modificar la configuración de notificaciones
- 404 Not Found
 - Descripción: no se encontró el miembro indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

Endpoints para Dispositivos

20) Conectar/Desconectar sensor

Header: Ninguno

POST /sensors/{hardwareId}/toggle

Ejemplo:

POST

<https://api.homeconnect.com/v1/sensors/{hardwareId}/open>

Cambia el estado de conexión del sensor entre conectado y desconectado.

Cuerpo de la solicitud (Request Body): Ninguno

Respuestas:

- 200 OK
 - Descripción: el estado de conexión del sensor fue actualizado exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
```

```
"connected": bool,  
"hardwareId": string,  
}
```

- 404 Not Found
 - Descripción: no se encontró el dispositivo indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

21) Crear notificación de ventana abierta de sensor

Header: Ninguno

POST /sensors/{hardwareId}/open

Ejemplo:

POST

<https://api.homeconnect.com/v1/sensors/{hardwareId}/open>

Crea una notificación informando que se abrió la ventana correspondiente al sensor.

Cuerpo de la solicitud (Request Body): Ninguno

Respuestas:

- 200 OK
 - Descripción: la notificación se creó y envió correctamente.
 - Cuerpo de la respuesta (Response Body):

```
{  
  "hardwareId": string  
}
```

- 400 Bad Request
 - Descripción: el dispositivo está desconectado
- 404 Not Found
 - Descripción: no se encontró el dispositivo indicado

- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

22) Crear notificación de ventana cerrada de sensor

Header: Ninguno

POST /sensors/{hardwareId}/close

Ejemplo:

POST

<https://api.homeconnect.com/v1/sensors/{hardwareId}/open>

Crea una notificación informando que se cerró la ventana correspondiente al sensor.

Cuerpo de la solicitud (Request Body): Ninguno

Respuestas:

- 200 OK
 - Descripción: la notificación se creó y envió correctamente.
 - Cuerpo de la respuesta (Response Body):

```
{
  "hardwareId" = string
}
```

- 400 Bad Request
 - Descripción: el dispositivo está desconectado
- 404 Not Found
 - Descripción: no se encontró el dispositivo indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

23) Conectar/Desconectar cámara

Header: Ninguno

POST /cameras/{hardwareId}/toggle

Ejemplo:

POST

<https://api.homeconnect.com/v1/sensors/{hardwareId}/open>

Cambia el estado de conexión de la cámara entre conectado y desconectado.

Cuerpo de la solicitud (Request Body): Ninguno

Respuestas:

- 200 OK
 - Descripción: el estado de conexión de la cámara fue actualizado exitosamente
 - Cuerpo de la respuesta (Response Body):

```
{
  "connected": bool,
  "hardwareId": string,
}
```

- 404 Not Found
 - Descripción: no se encontró el dispositivo indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

24) Crear notificación de detección de movimiento de cámara

Header: Ninguno

POST /cameras/{hardwareId}/movement-detected

Ejemplo:

POST

<https://api.homeconnect.com/v1/cameras/{hardwareId}/movement-detected>

Crea una notificación informando que la cámara detectó movimiento.

Cuerpo de la solicitud (Request Body): Ninguno

Respuestas:

- 200 OK
 - Descripción: la notificación se creó y envió correctamente.
 - Cuerpo de la respuesta (Response Body):

```
{
  "hardwareId" = string
}
```

- 400 Bad Request
 - Descripción: el dispositivo está desconectado
- 404 Not Found
 - Descripción: no se encontró el dispositivo indicado
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor

25) Crear notificación de detección de persona de cámara

Header: Ninguno

POST /cameras/{hardwareId}/person-detected

Ejemplo:

POST

<https://api.homeconnect.com/v1/cameras/{hardwareId}/person-detected>

Crea una notificación informando que la cámara detectó una persona registrada.

Cuerpo de la solicitud (Request Body):

- userId (requerido):
 - Descripción: Id del usuario registrado que la cámara detectó.
 - Tipo: Guid
 - Ejemplo: userId = a99feb27-7dac-41ec-8fd2-942533868689

Respuestas:

- 200 OK
 - Descripción: la notificación se creó y envió correctamente.
 - Cuerpo de la respuesta (Response Body):

```
{
  "hardwareId" = string
}
```

- 400 Bad Request
 - Descripción: el dispositivo está desconectado o el usuario no es válido / no fue encontrado
- 404 Not Found
 - Descripción: no se encontró el dispositivo indicado, falta el userId o es invalido
- 500 Internal Server Error
 - Descripción: ocurrió un error inesperado en el servidor