

# Manual de usuario



---

# Índice

<b>1. Introduction</b>	<b>1</b>
<b>2. Manual para la empresa</b>	<b>1</b>
2.1. Antes de empezar . . . . .	1
2.2. Identificarse en Mimir . . . . .	1
2.3. Crear una nueva aplicación . . . . .	3
2.4. Modificar y ver detalles de la aplicación . . . . .	5
2.5. Borrar aplicación . . . . .	7
2.6. Ver y modificar detalles de tu cuenta . . . . .	8
2.7. Borrar cuenta . . . . .	10
<b>3. Manual de uso de la cuenta aplicación</b>	<b>11</b>
3.1. Antes de empezar . . . . .	11
3.2. Obtener token de acceso . . . . .	11
3.3. Crear nueva entidad . . . . .	12
3.4. Listar entidades . . . . .	13
3.5. Acceder a un atributo de una entidad . . . . .	14
3.6. Obtener datos de una entidad . . . . .	15
3.7. Modificar un campo de una entidad . . . . .	15
3.8. Borrar entidad . . . . .	16

---

# 1. Introduction

El presente manual adopta un enfoque práctico basado en ejemplos para que los usuarios logren familiarizarse con Mimir de un modo rápido y ameno.

Este documento está dividido para los dos roles de usuarios admitidos por Mimir: las empresas y sus aplicaciones.

## 2. Manual para la empresa

### 2.1. Antes de empezar

Como la gestión de las cuentas de empresa es realizada comúnmente por personas, Mimir implementa una API navegable para que las operaciones listadas en este manual puedan ser realizadas de una forma sencilla. Por ello se proporcionará tanto la vista de navegador como los comandos para realizar todas las operaciones.

Para poder hacer uso de Mimir es necesario tener una cuenta en el sistema. Esta cuenta le será proporcionada por el ayuntamiento tras concretar su contrato con dicha institución.

Los datos de usuario le serán proporcionados junto con la dirección de acceso a Mimir tras la firma del contrato.

Para los ejemplos de este manual supondremos que la url de acceso es localhost en el puerto 8000, el usuario Business1 y la contraseña qazwsxed

Mimir utiliza JSON tanto para los datos en las solicitudes como para las respuestas.

***ATENCIÓN!: Recuerde que todas las urls deben terminar en '/'***

### 2.2. Identificarse en Mimir

Desde la página inicial de Mimir haga click en el botón login ubicado arriba a la derecha, tras lo cual será redirigido a la página de login donde deberá meter sus credenciales. Una vez hecho esto ya estará identificado en el sistema.

Mediante comando la única forma de loguearse en el sistema es mediante un token de acceso. Para obtenerlo ejecute el siguiente comando:

1 `http POST localhost:8000/api-token-auth/ username=Business1 password=qazwsxed`

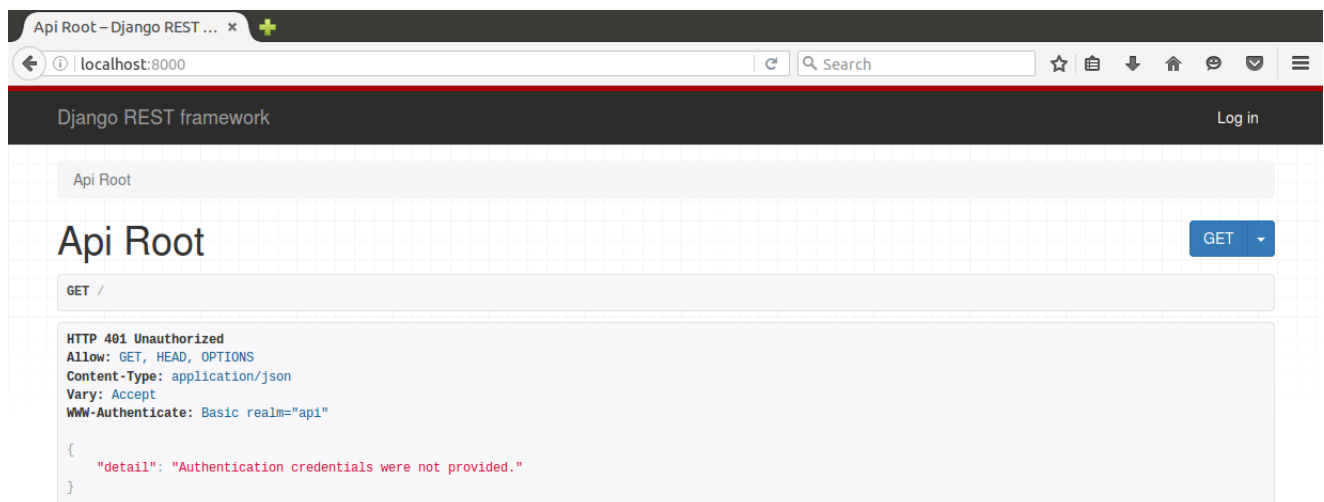


Figura 1: Página inicial

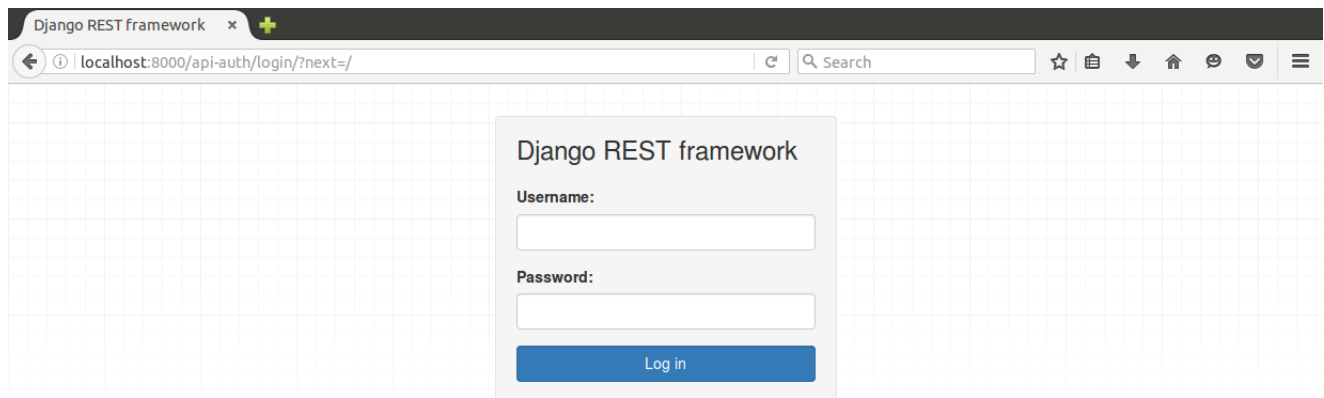


Figura 2: Página de login

```
tfg@tfg-VirtualBox:~$ http POST localhost:8000/api-token-auth/ username=Business
1 password=qazwsxed
HTTP/1.1 200 OK
Allow: POST, OPTIONS
Content-Length: 52
Content-Type: application/json
Date: Sun, 08 Jul 2018 22:35:30 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Cookie
X-Frame-Options: SAMEORIGIN

{
  "token": "cad7358a50df07d034ccc2f8306332820c6d03c8"
}

tfg@tfg-VirtualBox:~$
```

Figura 3: Obtener token de acceso

Como respuesta recibirá el token de acceso que deberá pasar en el resto de peticiones a Mimir

### 2.3. Crear una nueva aplicación

Diríjase a la página **localhost:8000/apps/** en ella se listarán todas sus aplicaciones y podrá crear nuevas.

Rellene los datos del formulario poniendo especial atención en que los permisos escogidos deben ser compatibles con los acordados con el ayuntamiento. Para este ejemplo escogeremos los permisos `createPerson` que nos permite crear nuevas entidades de tipo persona, `updatePerson` para modificar sus datos, `deletePerson` para borrar la entidad y `givenNamePerson` para acceder al atributo nombre de la entidad.

El comando equivalente es el siguiente:

```
1 http POST localhost:8000/apps/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
3 username=App1 description="App 1" password=qazwsxed
4 permissions='["createPerson","deletePerson","updatePerson","namePerson"]'
```

Si la aplicación se crea correctamente recibirá una respuesta con código 201 y los datos de la aplicación creada. Una vez creada una aplicación deberá esperar a que le den de paso los permisos escogidos y le activen la cuenta de la aplicación.

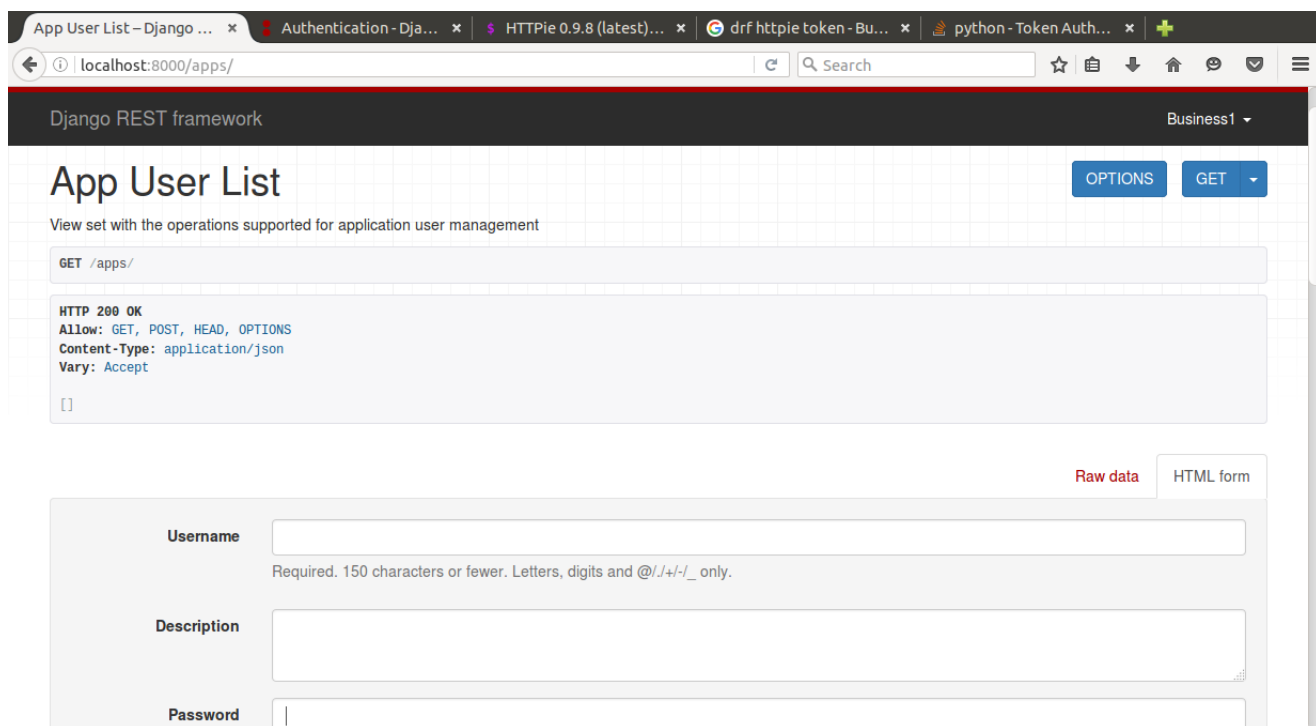


Figura 4: Pagina de listado y creación de aplicaciones

```
tfg@tfg-VirtualBox:~$ http POST localhost:8000/apps/ "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8" username=App1 description="App 1" password=qazwsxed permissions:=['createPerson','deletePerson','updatePerson','namePerson']
HTTP/1.1 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Length: 242
Content-Type: application/json
Date: Sun, 08 Jul 2018 22:59:29 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "active": false,
  "description": "App 1",
  "owner": "Business1",
  "password": "pbkdf2_sha256$100000$28US1yko123q$MwztFNWdMx0ujoQ2qDxGawV8s6Enzein0DCSEpjj1RI=",
  "permissions": [
    "createPerson",
    "deletePerson",
    "updatePerson",
    "namePerson"
  ],
  "username": "App1"
}
```

Figura 5: Crear nueva aplicación

## 2.4. Modificar y ver detalles de la aplicación

Diríjase a la página ***localhost:8000/apps/App1*** para ver los detalles de la aplicación. Desde esta página podrá además modificar los datos de la aplicación. Tenga en cuenta que una vez modificada la cuenta de la aplicación volverá al estado inactivo tendrá que ser activada de nuevo.

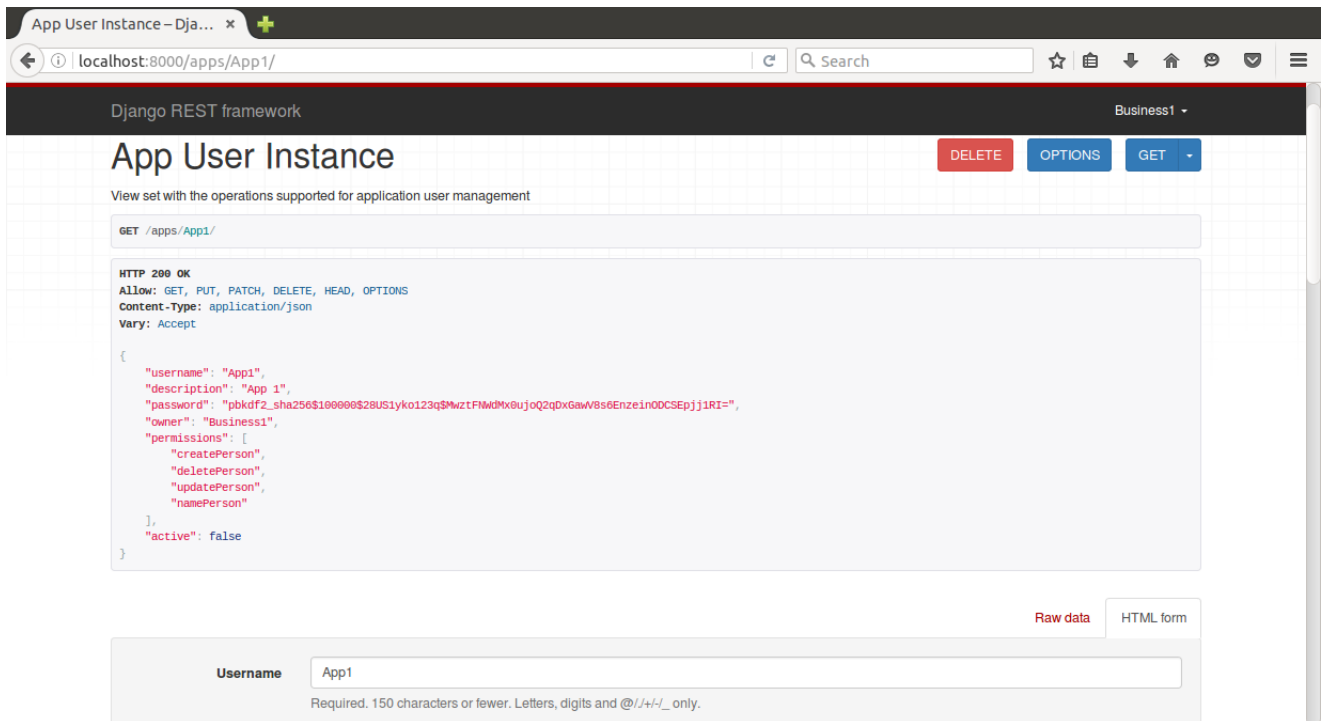


Figura 6: Pagina detalles y actualización de aplicaciones

Los comandos equivalentes son los siguientes:

Para ver detalles

```
1 http GET localhost:8000/apps/App1/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
```

Este comando devolverá una respuesta con código 200 y los datos de la aplicación

Para modificar

```
1 http PUT localhost:8000/apps/App1/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
3 username=App1 description="App 1 modificada" password=qazwsxed
4 permissions='["createPerson", "deletePerson", "updatePerson", "namePerson"] '
```

Si la aplicación se modifica correctamente recibirá una respuesta con código 200 y los nuevos datos de la aplicación.

```
tfg@tfg-VirtualBox:~$ http GET localhost:8000/apps/App1/ "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
HTTP/1.1 200 OK
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Length: 253
Content-Type: application/json
Date: Sun, 08 Jul 2018 23:45:48 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "active": false,
  "description": "App 1 modificada",
  "owner": "Business1",
  "password": "pbkdf2_sha256$100000$P1uolG509CqIS$Q1FdxCFUb0pcpcL5PyINn8tq0j6lt4QOxmnujjw2kmm=",
  "permissions": [
    "createPerson",
    "deletePerson",
    "updatePerson",
    "namePerson"
  ],
  "username": "App1"
}

tfg@tfg-VirtualBox:~$
```

Figura 7: Ver detalles de aplicación

```
tfg@tfg-VirtualBox:~$ http PUT localhost:8000/apps/App1/ "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8" username=App1 description="App 1 modificada" password=qazwsxed permissions='["createPerson","deletePerson","updatePerson","namePerson"]'
HTTP/1.1 200 OK
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Length: 253
Content-Type: application/json
Date: Sun, 08 Jul 2018 23:31:16 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "active": false,
  "description": "App 1 modificada",
  "owner": "Business1",
  "password": "pbkdf2_sha256$100000$P1uolG509CqIS$Q1FdxCFUb0pcpcL5PyINn8tq0j6lt4QOxmnujjw2kmm=",
  "permissions": [
    "createPerson",
    "deletePerson",
    "updatePerson",
    "namePerson"
  ],
  "username": "App1"
}

tfg@tfg-VirtualBox:~$
```

Figura 8: Modificar aplicación



## 2.5. Borrar aplicación

Desde la página *localhost:8000/apps/App1* también podrá borrar la cuenta de la aplicación. Para ello únicamente deberá pulsar en el botón rojo de la izquierda y confirmar en la ventana emergente.

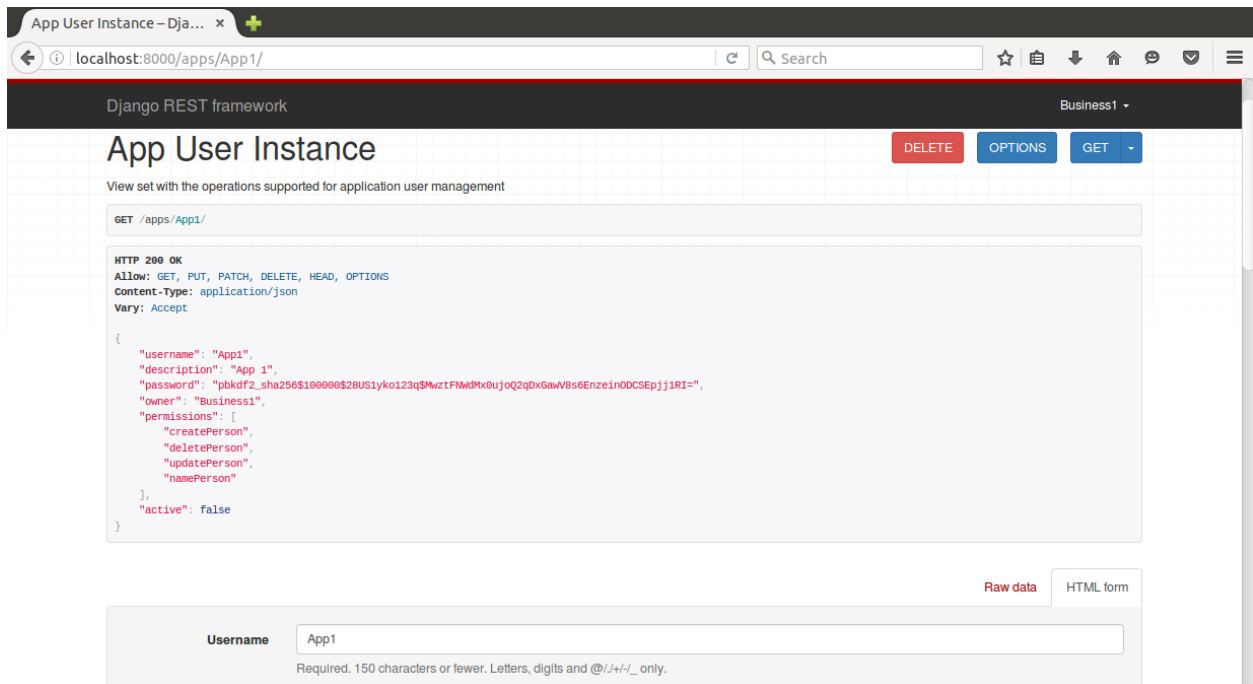


Figura 9: Borrar aplicación

El comando equivalente es el siguiente:

```
1 http DELETE localhost:8000/apps/App1/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
```

```
tfg@tfg-VirtualBox:~$ http DELETE localhost:8000/apps/App1/ "Authorizati
n cad7358a50df07d034ccc2f8306332820c6d03c8"
HTTP/1.1 204 No Content
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Length: 0
Date: Sun, 08 Jul 2018 23:48:31 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

tfg@tfg-VirtualBox:~$
```

Figura 10: Comando para borrar aplicación

Si la aplicación se borra correctamente recibirá una respuesta con código 204.

## 2.6. Ver y modificar detalles de tu cuenta

Diríjase a la página *localhost:8000/users/Business1* para ver los detalles de su cuenta. Desde esta página podrá además modificar sus datos.

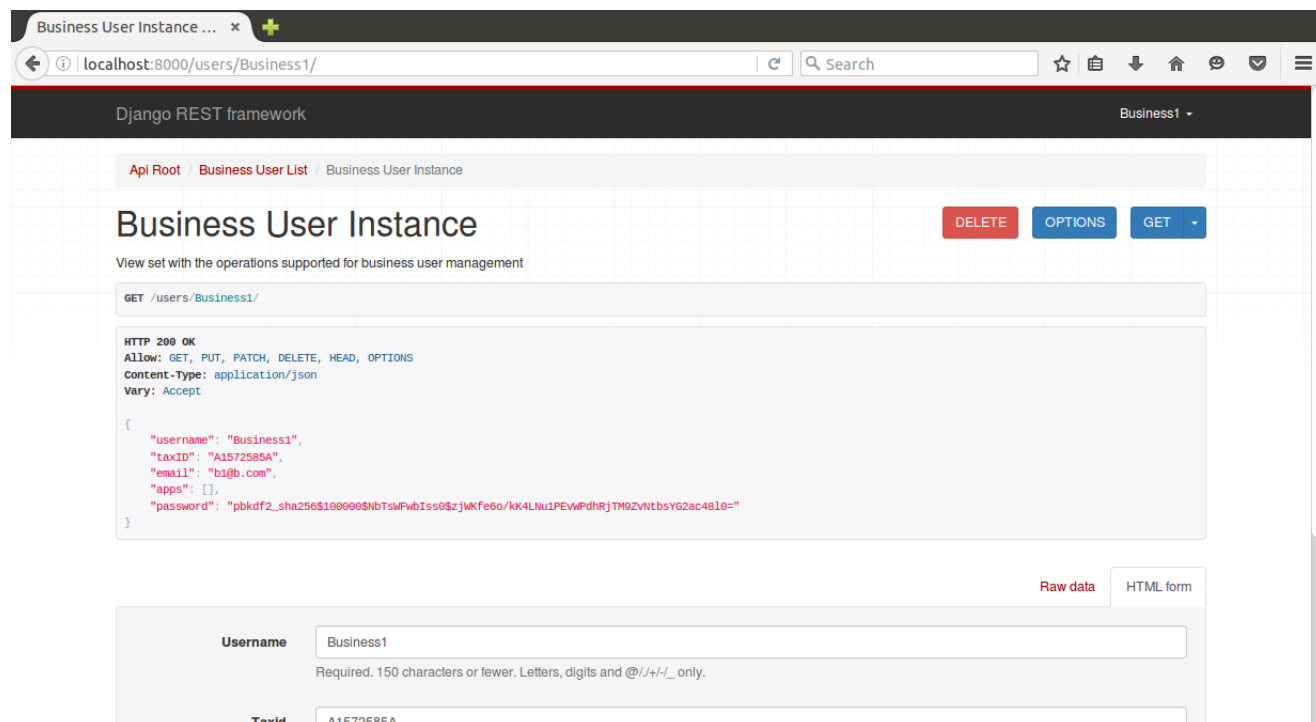


Figura 11: Ver y modificar tu cuenta

Los comandos equivalentes son los siguientes:

Para ver detalles

```
1 http GET localhost:8000/users/Business1/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
```

Este comando devolverá una respuesta con código 200 y los datos de tu cuenta

Para modificar

```
1 http PUT localhost:8000/users/Business1/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
3 username=Business1 email=modificado@mod.com password=qazwsxed taxID=B1234567B
```

Si su cuenta se modifica correctamente recibirá una respuesta con código 200 y los nuevos datos de la aplicación.

```
tfg@tfg-VirtualBox:~$ http GET localhost:8000/users/Business1/ "Authorization: cad7358a50df07d034ccc2f8306332820c6d03c8"
HTTP/1.1 200 OK
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Length: 165
Content-Type: application/json
Date: Sun, 08 Jul 2018 23:57:25 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "apps": [],
  "email": "b1@b.com",
  "password": "pbkdf2_sha256$100000$NbTsWFwbIss0$zjWKfe6o/kK4LNU1PEvWPFvNtbsYG2ac48l0=",
  "taxID": "A1572585A",
  "username": "Business1"
}

tfg@tfg-VirtualBox:~$
```

Figura 12: Ver detalles de tu cuenta

```
tfg@tfg-VirtualBox:~$ http PUT localhost:8000/users/Business1/ "Authorization: cad7358a50df07d034ccc2f8306332820c6d03c8" username=Business1 email=do@mod.com password=qazwsxed taxID=B1234567B
HTTP/1.1 200 OK
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Length: 175
Content-Type: application/json
Date: Sun, 08 Jul 2018 23:59:54 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "apps": [],
  "email": "modificado@mod.com",
  "password": "pbkdf2_sha256$100000$ZEFrfTyYMJ1Z$HmixeLkcHrc/NRpkB7IdwQXuQGTLVJ1G5Pg=",
  "taxID": "B1234567B",
  "username": "Business1"
}

tfg@tfg-VirtualBox:~$
```

Figura 13: Modificar datos de tu cuenta

## 2.7. Borrar cuenta

Desde la página *localhost:8000/users/Business1* también podrá borrar su cuenta de usuario. Para ello únicamente deberá pulsar en el botón rojo de la izquierda y confirmar en la ventana emergente.

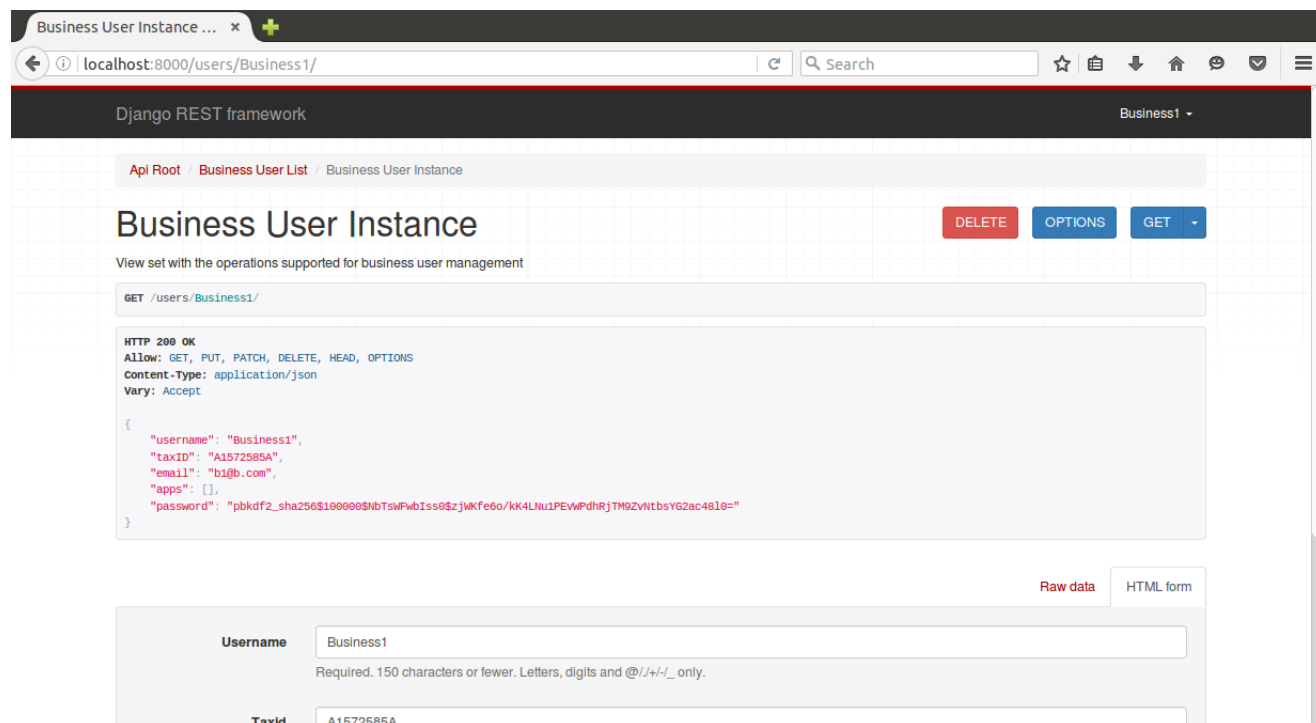


Figura 14: Borrar tu cuenta

El comando equivalente es el siguiente:

```
1 http DELETE localhost:8000/users/Business1/
2 "Authorization: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
```

```
tfg@tfg-VirtualBox:~$ http DELETE localhost:8000/users/Business1/ "Autho
: Token cad7358a50df07d034ccc2f8306332820c6d03c8"
HTTP/1.1 204 No Content
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Length: 0
Date: Mon, 09 Jul 2018 00:03:03 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN
tfg@tfg-VirtualBox:~$
```

Figura 15: Borrar tu cuenta

Si la cuenta de usuario se borra correctamente recibirá una respuesta con código 204.

---

## 3. Manual de uso de la cuenta aplicación

### 3.1. Antes de empezar

Para poder seguir los pasos detallados en este manual es necesario tener una cuenta de aplicación activada en el sistema.

El usuario objetivo de esta sección del manual es el programador de la aplicación que hará uso de esta cuenta, por lo tanto en estos ejemplos se presentarán únicamente las acciones via comando.

Para los ejemplos de este manual supondremos que la url de acceso es localhost en el puerto 8000, el usuario App1 y la contraseña qazwsxed

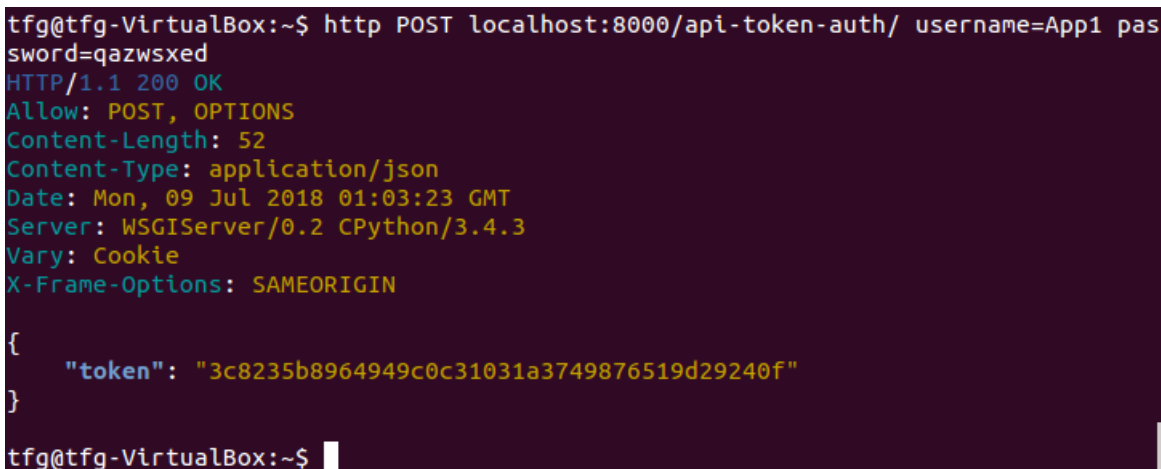
App1 tendrá los permisos necesarios para crear actualizar y borrar entidades de tipo persona así como el permiso para acceder al atributo *name* de las entidades de tipo persona. Mimir utiliza JSON tanto para los datos en las solicitudes como para las respuestas.

**ATENCIÓN!:** Recuerde que todas las urls deben terminar en ‘/’

### 3.2. Obtener token de acceso

Mimir utiliza como modo de autenticación via comando tokens de acceso. Para obtener un token de acceso ejecute el siguiente comando.

```
1 http POST localhost:8000/api-token-auth/ username=App1 password=qazwsxed
```



```
tfg@tfg-VirtualBox:~$ http POST localhost:8000/api-token-auth/ username=App1 password=qazwsxed
HTTP/1.1 200 OK
Allow: POST, OPTIONS
Content-Length: 52
Content-Type: application/json
Date: Mon, 09 Jul 2018 01:03:23 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Cookie
X-Frame-Options: SAMEORIGIN

{
  "token": "3c8235b8964949c0c31031a3749876519d29240f"
}
tfg@tfg-VirtualBox:~$
```

Figura 16: Obtener token de acceso

Como respuesta recibirá el token de acceso que deberá pasar en el resto de peticiones a Mimir.

### 3.3. Crear nueva entidad

Para crear una nueva entidad utilice el siguiente comando con el mismo formato para los datos que el mostrado.

```
1 http POST localhost:8000/v2/entities/  
2 "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f"  
3 id=12345678D type=Person  
4 givenName:='{ "value": "TEST", "type": "String" }'  
5 familyName:='{ "value": "TEST TEST", "type": "String" }'  
6 birthDate:='{ "value": "1999-02-15", "type": "String" }'  
7 birthPlace:='{ "value": "Valladolid, Valladolid", "type": "String" }'  
8 address:='{ "value": "TEST", "type": "String" }'  
9 gender:='{ "value": "Masculino", "type": "String" }'  
10 nationality:='{ "value": "española", "type": "String" }'  
11 parkingCard:='{ "value": 0, "type": "Integer" }'
```

```
tfg@tfg-VirtualBox:~$ http POST localhost:8000/v2/entities/ "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f" id=12345678D type=Person givenName:='{ "value": "TEST", "type": "String" }' familyName:='{ "value": "TEST TEST", "type": "String" }' birthDate:='{ "value": "1999-02-15", "type": "String" }' birthPlace:='{ "value": "Valladolid, Valladolid", "type": "String" }' address:='{ "value": "TEST", "type": "String" }' gender:='{ "value": "Masculino", "type": "String" }' nationality:='{ "value": "española", "type": "String" }' parkingCard:='{ "value": 0, "type": "Integer" }'  
HTTP/1.1 201 Created  
Allow: GET, POST, HEAD, OPTIONS  
Content-Length: 397  
Content-Type: application/json  
Date: Mon, 09 Jul 2018 01:10:30 GMT  
Server: WSGIServer/0.2 CPython/3.4.3  
Vary: Accept, Cookie  
X-Frame-Options: SAMEORIGIN  
  
{  
  "address": {  
    "type": "String",  
    "value": "TEST"  
  },  
  "birthDate": {  
    "type": "String",  
    "value": "1999-02-15"  
  },  
  "birthPlace": {  
    "type": "String",  
    "value": "Valladolid, Valladolid"  
  },  
  "familyName": {  
    "type": "String",  
    "value": "TEST TEST"  
  },  
  "gender": {
```

Figura 17: Crear nueva entidad

Si la aplicación se crea correctamente recibirá una respuesta con código 201 y los datos de la entidad creada.

### 3.4. Listar entidades

Para listar todas las entidades de un determinado tipo, utilice el siguiente comando. Utilice `type` para seleccionar el tipo de las entidades a listar. No es posible listar entidades de diversos tipos a la vez.

```
1 http GET localhost:8000/v2/entities/
2 "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f"
3 type==Person
```

```
tffg@tffg-VirtualBox:~$ http GET localhost:8000/v2/entities/ "Authorization: Token
3c8235b8964949c0c31031a3749876519d29240f" type==Person
HTTP/1.1 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Length: 254
Content-Type: application/json
Date: Mon, 09 Jul 2018 01:23:13 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

[
  {
    "givenName": "Maria"
  },
  {
    "givenName": "Diego"
  },
  {
    "givenName": "Nerea"
  },
  {
    "givenName": "Jose Francisco"
  },
  {
    "givenName": "Guillermo"
  },
  {
    "givenName": "Celia"
  },
  {
    "givenName": "Saray"
  }
]
```

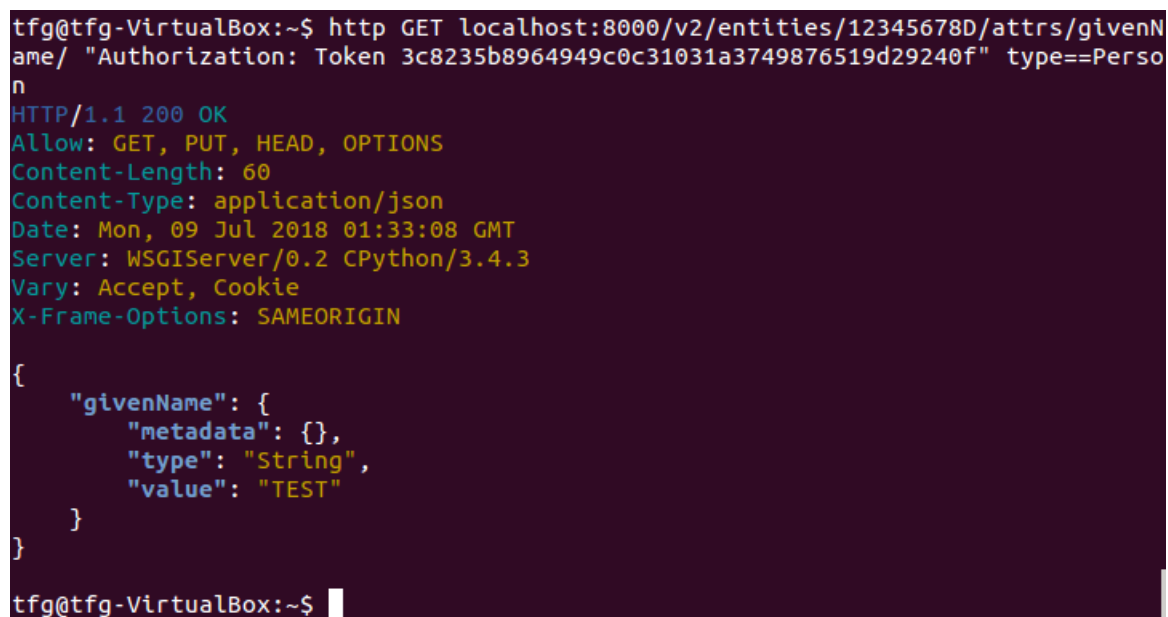
Figura 18: Listar entidades de un determinado tipo

Recibirá una respuesta con código 200 y los datos de la entidad a los que tenga acceso. Como podemos ver, App1 recibe únicamente el campo ***givenName*** ya que es el único al que tiene permitido el acceso

### 3.5. Acceder a un atributo de una entidad

Para acceder a un atributo de una entidad, utilice el siguiente comando. Es obligatorio el uso de type para seleccionar el tipo de la entidad. Opcionalmente puede usar options=keyValues para mostrar únicamente el valor del atributo.

```
1 http GET localhost:8000/v2/entities/12345678D/attrs/givenName
2 "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f"
3 type==Person
```



```
tfg@tfg-VirtualBox:~$ http GET localhost:8000/v2/entities/12345678D/attrs/givenName/ "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f" type==Person
HTTP/1.1 200 OK
Allow: GET, PUT, HEAD, OPTIONS
Content-Length: 60
Content-Type: application/json
Date: Mon, 09 Jul 2018 01:33:08 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "givenName": {
    "metadata": {},
    "type": "String",
    "value": "TEST"
  }
}
tfg@tfg-VirtualBox:~$
```

Figura 19: Acceder atributo de una entidad

Si el atributo existe dentro de la entidad y tienes permiso para verlo recibirá una respuesta con código 200 y el atributo.



---

### 3.6. Obtener datos de una entidad

Para acceder a los datos una única entidad, utilice el siguiente comando. Es obligatorio el uso de type para seleccionar el tipo de la entidad.

```
1 http GET localhost:8000/v2/entities/12345678D/
2 "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f"
3 type==Person
```



```
tfg@tfg-VirtualBox:~$ http GET localhost:8000/v2/entities/12345678D/ "Authorizat
ion: Token 3c8235b8964949c0c31031a3749876519d29240f" type==Person
HTTP/1.1 200 OK
Allow: GET, DELETE, HEAD, OPTIONS
Content-Length: 20
Content-Type: application/json
Date: Mon, 09 Jul 2018 01:46:46 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

{
  "givenName": "TEST"
}

tfg@tfg-VirtualBox:~$
```

Figura 20: Obtener datos de una entidad

Recibirá una respuesta con código 200 y los atributos de la entidad a los que tenga acceso.

### 3.7. Modificar un campo de una entidad

Para modificar un campo de una entidad, utilice el siguiente comando. Es obligatorio el uso de type para seleccionar el tipo de la entidad.

```
1 http PUT localhost:8000/v2/entities/12345678D/attrs/givenName/
2 "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f"
3 type==Person
4 givenName:='{ "value": "Mod", "type": "String" }'
```

Recibirá una respuesta con código 204 si el campo se ha modificado satisfactoriamente.

```
tfg@tfg-VirtualBox:~$ http PUT localhost:8000/v2/entities/12345678D/attrs/g
ame/ "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f" type==
n givenName:={"value":"Mod","type":"String"}'
HTTP/1.1 204 No Content
Allow: GET, PUT, HEAD, OPTIONS
Content-Length: 0
Date: Mon, 09 Jul 2018 01:56:41 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

tfg@tfg-VirtualBox:~$
```

Figura 21: Modificar datos de una entidad

### 3.8. Borrar entidad

Para borrar una entidad existente, utilice el siguiente comando. Es obligatorio el uso de type para seleccionar el tipo de la entidad.

```
1 http DELETE localhost:8000/v2/entities/12345678D/
2 "Authorization: Token 3c8235b8964949c0c31031a3749876519d29240f"
3 type==Person
```

```
tfg@tfg-VirtualBox:~$ http DELETE localhost:8000/v2/entities/12345678D/ "Au
zation: Token 3c8235b8964949c0c31031a3749876519d29240f" type==Person
HTTP/1.1 204 No Content
Allow: GET, DELETE, HEAD, OPTIONS
Content-Length: 0
Date: Mon, 09 Jul 2018 02:00:16 GMT
Server: WSGIServer/0.2 CPython/3.4.3
Vary: Accept, Cookie
X-Frame-Options: SAMEORIGIN

tfg@tfg-VirtualBox:~$
```

Figura 22: Borrar una entidad

Recibirá una respuesta con código 204 si la entidad se ha borrado satisfactoriamente.