

Universidad ORT Uruguay
Facultad de Ingeniería

Diseño de Aplicaciones 2

Obligatorio 1

Evidencia del diseño y especificación de la API

Mauro Teixeira(211753) Rodrigo Hirigoyen (197396)

Docentes: Gabriel Piffaretti, Nicolas Fierro, Nicolas Hernandez

<https://github.com/mauroteix/Teixeira-Hirigoyen>

Entregado como requisito de la materia Diseño de Aplicaciones 2

6 de mayo de 2021

Índice

Criterios seguidos	2
Mecanismo de autenticación	2
Manejo del token	2
Encabezados para autenticación	3
Filtro de autenticación	3
Códigos de error	3
Controladores	4
CategoryController	4
PlaylistController	4
TrackController	4
PsychologistController	5
UserController	5
AdministratorController	5
SessionController	5
MedicalConditionController	5
Descripción de los resources	6
Administrator	6
Category	7
MedicalCondition	8
Playlist	9
Psychologist	12
Session	15
Track	15
User	18
Evidencia REST	19

Criterios seguidos

La aplicación fue diseñada considerando un conjunto de estándares establecidos por REST. Los mismos se listan a continuación:

- **Los endpoint respetan las siguientes convenciones:**
 - Se utilizan sustantivos antes que verbos. (ej. .../category)
 - Se busca que sea intuitiva y simple. Para esto los sustantivos utilizados son lo más nemotécnico en función de lo que se está solicitando.
 - No se utilizan más de dos niveles
- Se utilizan los 4 verbos más comunes, estos son: GET, POST, PUT y DELETE.
- Se utilizan status codes de HTTP para el manejo de errores y excepciones donde la gran mayoría vienen acompañados de un mensaje que describe el error.

Mecanismo de autenticación

Para nuestra aplicación, utilizamos una autenticación basada en usuarios. Hoy en día se identifican dos tipos de usuarios:

- Administrador
- Usuario(Paciente)

El usuario(Paciente) puede realizar operaciones de ver las principales categorías y navegar entre su contenido y playlists, reproduciendo el contenido y agendar consultas con psicólogos en nuestro sistema. Para dichos endpoints, no se agregaron filtros de autenticación, dado que no requiere ningún nivel de seguridad.

Para los usuarios administradores, los cuales van a llevar el mantenimiento del sistema, se realizó un modelo el cual tiene un nombre, email, password y token de validación. Dicho token es único para cada usuario, el cual se va a utilizar en cada petición a los endpoints para validarse que el usuario tiene los permisos necesarios para realizar dicha operación.

Manejo del token

Nuestros administradores necesitan realizar un Login en nuestro sistema mediante email y password, y en caso de ser correctos, el mismo devuelve un token de autenticación, el cual debe ser usado en los Headers de las request, mediante el identificador 'Authentication'.

Encabezados para autenticación

```
[ServiceFilter(typeof(AuthorizationFilter))]  
[HttpPost()]  
5 referencias | 4/4 pasando | mauroteix, Hace 1 hora | 2 autores, 5 cambios  
public IActionResult Add([FromBody] Administrator admin)  
{  
    try  
    {
```

Como se muestra, cada endpoint que requieran de autenticación, se la anotación de ServiceFilter, con la clase que realiza dicha autorización.

Filtro de autenticación

Creamos una sección de Filtros en nuestro proyecto de webApi, dado que se va a utilizar en dicho proyecto, específicamente en los controllers. En dicha sección se va a crear nuestra clase de AuthorizationFilter, la cual se le inyecta nuestra interface que va a manejar todo el proceso de validación de token.

Códigos de error

Se utilizaron los siguientes códigos de errores:

Code	Description
200	Ok. Return the request object
401	Unauthorized. You do not have permissions to perform this action
404	NotFound. The requested object was not found
422	Unprocessable. Error in the semantics
501	InternalServerError. The server could not handle an exception in the system.

Nos permite indicar errores concretos, con mensajes para los usuarios a fin de comprender la situación actual de su petición. Los mismos se pueden observar en la documentación luego de realizar algún endpoint.

Controladores

De acuerdo a las funcionalidades solicitadas en la letra del problema planteado optamos por ocho controladores en nuestro sistema. Cada uno para las diferentes funcionalidades con cada recurso y serán presentados a continuación:

BetterCalm <small>v1</small> <small>OpenAPI 3.0</small> <small>/swagger/v1/swagger.json</small>	
Administrator	>
Category	>
MedicalCondition	>
Playlist	>
Psychologist	>
Session	>
Track	>
User	>

- **CategoryController**

Vimos que era un controlador necesario ya que por letra los usuarios del sistema acceden a ver todas las categorías de forma pública.

- **PlaylistController**

Por letra y verificado en el foro, los administradores deben realizar el mantenimiento de las playlist por lo que nuestro sistema va a tener las funciones necesarias para realizar lo solicitado, y es por ello que se decidió implementarlo. A su vez el usuario puede navegar entre las playlist

- **TrackController**

También por letra indica que el administrador debe realizar alta y baja de un contenido reproducible de este modo es que decidimos incluir este controlador. También el usuario puede navegar entre el contenido reproducible.

- **PsychologistController**

En este caso también podemos ver que los psicólogos deben ser mantenidos en nuestro sistema, por lo que era sumamente necesario tener entre controlador así los administradores podrán mantener dicha entidad.

- **UserController**

El problema solicita agendar la consulta con un psicólogo ingresando los datos del usuario, por lo que vimos conveniente tener un controlador específico para realizar la agenda de su consulta. Y que ésta permita realizar esa operación.

- **AdministratorController**

Como la letra lo indica un administrador va realizar las tareas del mantenimiento del sistema.

- **SessionController**

Debido a que el sistema solo permite realizar ciertas funcionalidades a los administradores es de aquí que se decide tener entre controlador para manejar la sesión.

- **MedicalConditionController**

Los usuarios del sistema deben poder ver las medical condition para luego poder hacer la agenda para esa problemática.

Descripción de los resources

Administrator

POST `/api/administrator` Add a administrator

Try it out

Parameters

No parameters

Request body

application/json-patch+json

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "email": "string",
  "password": "string",
  "token": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

DELETE `/api/administrator/{id}` Delete a administrator by id

Try it out

Parameters

Name	Description
id * required integer(\$int32) (path)	id

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

PUT

/api/administrator/{id} Update a administrator by id

Parameters

Try it out

Name	Description
id * required integer(\$int32) (path)	<input type="text" value="id"/>

Request body

application/json-patch+json

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "email": "string",
  "password": "string",
  "token": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

Category

GET

/api/category Get all categories

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

GET

/api/category/{id} Get category by id

Try it out

Parameters

Name	Description
id * required integer(\$int32) (path)	id

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

MedicalCondition

GET

/api/medicalcondition Get all the medical condition

Try it out

Parameters

No parameters

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

GET

/api/medicalcondition/{id} Get medical condition by id

Try it out

Parameters

Name	Description
id * required integer(\$int32) (path)	id

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

Playlist

GET `/api/playlist/{id}` Get a playlist by id

Parameters Try it out

Name	Description
id * required integer(int32) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

DELETE `/api/playlist/{id}` Delete a playlist by id

Parameters Try it out

Name	Description
id * required integer(int32) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

PUT

/api/playlist/{id} Update a playlist by id

Try it out

Parameters

Name	Description
id * required integer(int32) (path)	id

Request body

application/json-patch+json

Example Value | Schema

```

{
  "id": 0,
  "name": "string",
  "description": "string",
  "image": "string",
  "playlistTrack": [
    {
      "id@playlist": 0,
      "playlist": "string",
      "id@track": 0,
      "track": {
        "id": 0,
        "name": "string",
        "author": "string",
        "image": "string",
        "hour": 0,
        "minSeconds": 0,
        "sound": "string",
        "category/track": [
          {
            "idCategory": 0,
            "category": "string",
            "id@track": 0,
            "track": "string"
          }
        ]
      },
      "playlistTrack": [
        "string"
      ]
    }
  ]
}

```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

POST

/api/playlist Add a playlist

Try it out

Parameters

No parameters

Request body

application/json-patch+json

Example Value | Schema

```
{
  "track": {
    "id": 0,
    "name": "string",
    "author": "string",
    "image": "string",
    "hour": 0,
    "minSeconds": 0,
    "sound": "string",
    "categoryTrack": [
      {
        "idCategory": 0,
        "category": "string",
        "idtrack": 0,
        "track": "string"
      }
    ],
    "playlistTrack": [
      "string"
    ]
  },
  "playlistCategory": [
    {
      "idPlaylist": 0,
      "playlist": "string",
      "idCategory": 0,
      "category": {
        "id": 0
      }
    }
  ]
}
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

GET

/api/playlist Get all playlist

Try it out

Parameters

No parameters

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

Psychologist

GET /api/psychologist Get all the psychologist		
Parameters		
No parameters		
Responses		
Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

POST /api/psychologist Add a psychologist		
Parameters		
No parameters		
Request body		
application/json-patch+json		
Example Value Schema		
<pre>{ "id": 0, "name": "string", "meetingType": 1, "addressMeeting": "string", "expertise": [{ "idMedicalCondition": 0, "medicalCondition": { "id": 0, "name": "string", "expertise": ["string"] } }], "idPsychologist": 0, "psychologist": "string" }, { "meeting": [{ "user": { "id": 0, "name": "string", "surname": "string", "birthday": "2021-05-06T15:02:34.440Z", "email": "string", "cellphone": "string", "meeting": [</pre>		
Responses		
Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

GET

/api/psychologist/{id}

Get a psychologist by id

Parameters

Try it out

Name	Description
id * required	
integer(\$int32)	id
(path)	

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

DELETE

/api/psychologist/{id}

Delete a psychologist by id

Parameters

Try it out

Name	Description
id * required	
integer(\$int32)	id
(path)	

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

PUT

/api/psychologist/{id} Update a psychologist by id

Parameters

Try it out

Name	Description
id * required	id

integer(\$int32)
(path)

Request body

application/json-patch+json

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "meetingType": "",
  "addressMeeting": "string",
  "expertise": [
    {
      "idMedicalCondition": 0,
      "medicalCondition": {
        "id": 0,
        "name": "string",
        "expertise": [
          "string"
        ]
      }
    }
  ],
  "idPsychologist": 0,
  "psychologist": "string"
},
"meeting": [
  {
    "user": {
      "id": 0,
      "name": "string",
      "surname": "string",
      "birthday": "2021-05-06T15:03:52.456Z",
      "email": "string",
      "cellphone": "string",
      "meeting": {
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

Session

POST /api/session Login a administrator

Parameters Try it out

No parameters

Request body application/json-patch+json ▾

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "email": "string",
  "password": "string",
  "token": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

Track

GET /api/track/{id} Get track by id

Parameters Try it out

Name	Description
id <small>required</small> integer(int32) (path)	<input type="text" value="id"/>

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

PUT

/api/track/{id} Update a track by id

Try it out

Parameters

Name	Description
id * required	id
integer(\$int32)	(path)

Request body

application/json-patch+json

Example Value | Schema

```

{
  "id": 0,
  "name": "string",
  "author": "string",
  "image": "string",
  "hour": 0,
  "minSeconds": 0,
  "sound": "string",
  "categoryTrack": [
    {
      "idCategory": 0,
      "category": {
        "id": 0,
        "name": "string",
        "categoryTrack": [
          "string"
        ]
      },
      "playlistCategory": [
        {
          "idPlaylist": 0,
          "playlist": "string",
          "idCategory": 0,
          "category": "string"
        }
      ]
    }
  ],
  "idTrack": 0,
  "track": "string"
}

```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

DELETE

/api/track/{id} Delete a track by id

Try it out

Parameters

Name	Description
id * required	id
integer(\$int32)	(path)

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

POST

/api/track

Add a track

Parameters

Try it out

No parameters

Request body

application/json-patch+json

Example Value | Schema

```
{  "id": 0,  "name": "string",  "author": "string",  "image": "string",  "hour": 0,  "minSeconds": 0,  "sound": "string",  "categoryTrack": [    {      "idCategory": 0,      "category": {        "id": 0,        "name": "string",        "categoryTrack": [          "string"        ]      },      "playlistCategory": [        {          "idPlaylist": 0,          "playlist": "string",          "idCategory": 0,          "category": "string"        }      ]    },    {      "idTrack": 0,      "track": "string"    }  ]}
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
401	Unauthorized. You do not have permissions to perform this action.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

GET

/api/track

Get all the tracks

Parameters

Try it out

No parameters

Responses

User

POST

/api/user

Create a meeting for a user

Parameters

Try it out

No parameters

Request body

application/json-patch+json

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "surname": "string",
  "birthday": "2021-05-06T15:07:28.266Z",
  "email": "string",
  "cellphone": "string",
  "meeting": [
    {
      "user": "string",
      "idUser": 0,
      "psychologist": {
        "id": 0,
        "name": "string",
        "meetingType": "",
        "addressMeeting": "string",
        "expertise": [
          {
            "idMedicalCondition": 0,
            "medicalCondition": "string",
            "idPsychologist": 0,
            "psychologist": "string"
          }
        ]
      },
      "meeting": {
        "string"
      }
    }
  ],
  "idPsychologist": 0,
  "birthday": "2021-05-06T15:07:28.266Z"
}
```

Responses

Code	Description	Links
200	OK. Returns the requested object.	No links
404	NotFound. The requested object was not found.	No links
422	UnprocessableEntity. Error in the semantics.	No links
501	InternalServerError. The server could not handle an exception in the system.	No links

Evidencia REST

- Cliente-servidor

Tenemos por un lado un servidor el cual le brinda al cliente servicios, expone una serie de endpoint los cuales el usuario puede usar para poder hacer operaciones. Estos endpoint son los descritos anteriormente.

- Sin estado

El servidor cada vez que recibe una nueva request crea los servicios nuevamente, esto se lo especificamos en el startup de nuestra app, lo cual hace que cada request que le llegue no tenga una trazabilidad sino que se trata siempre como una nueva request que llega.

- Se puede almacenar en caché

Al tener una aplicación sin estado, el usuario debe ser capaz de almacenar los datos que considere necesarios para poder llevar un estado, como por ejemplo para la autenticación del mismo debe guardar el token que el servidor le provee y utilizarlo en el header de las request que hace.

- Interfaz uniforme

- Mensajes autodescriptivos
- Identificación de recursos
- Manipulación de recursos a través de representaciones

En nuestra aplicación cumplimos con estas cuatro condiciones, esto se puede apreciar viendo cada endpoint que se describió anteriormente, como identificamos los recursos, como serializamos los recursos mapearlos con entidades de nuestro sistema, como manejamos los errores brindándole al usuario un texto informativo para que pueda realmente darse cuenta de lo que pasó.

- Sistema en capas

Como se muestra en el documento de descripción de diseño tenemos una aplicación que está dividida en capas, en el documento mostramos y describimos cada una de estas capas. Por otra parte utilizamos todos los verbos http (GET / PUT / POST / DELETE) dependiendo la operación que queramos realizar, identificándose correctamente.