Universidad ORT Uruguay

Facultad de Ingeniería

# Obligatorio Diseño Aplicaciones 2

## Diseño Api REST

Luis Sempolis 185664

Entregado como requisito de la materia Diseño Aplicaciones 2

6 de mayo de 2021

# Declaraciones de autoría

Nosotros, Luis Sempolis y Martin Vergara ; declaramos que el trabajo que se presenta en esa obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizábamos el primer obligatorio de Probabilidad y Estadística;
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad;
- Cuando hemos citado obras de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, hemos acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotros;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

# Índice general

## 0.1. Discusión de los criterios REST

- **Interfaz Uniforme** Siempre nos basamos en un recurso en particular Audio,Playlist, Cita etc. y estos se mapean con algo del entorno de la problemática.

- **Stateless** No manejamos ningún tipo de estado, como podemos ver lo único que requeriría un estado seria la autenticación pero esta se espera que se maneje con una cookie del lado del cliente en una futura implementación.
- **Cacheable** No manejamos ningún tipo de cacheo, ni siquiera en las partes fijas de la aplicación, esto es un debe a corregir en próximas entregas.
- **Implementación** En ningún momento le damos a entender al cliente si esta conectado a un servidor final o intermediario, simplemente manejamos URL y estos se conectan por alli con la aplicacion.
- **Sistema Separdado en Capas** Como se puede ver en los diagramas de la documentación nro.1 nuestro sistema se basa 100
- **Las llamada a los recursos se hace mediante sustantivos y se deja que la accion la identifique el HTTP REQUEST.**
- **Uso del Plural.**

## 0.2. Descripción del mecanismo de autenticación de requests

Simplemente se realiza de la siguiente forma, poseemos una tabla en la base de datos llamada tokens. Lo que se hace es cuando un administrador se loguea es brindarle un token (cadena string) y agregar el mismo a la tabla tokens junto a la fecha de ingreso. Se da por valido que cuando un cliente posea un token ya sea almacenado en una cookie etc. y este brinda el mismo en la consulta este se da por valido si simplemente esta agregado en la tabla tokens. Es decir estamos cambiando las credenciales Usuario y Contraseña por una cadena de String.
La fecha en la tabla es para que el administrador de base de datos pueda definir un Assert que elimine automaticamente los tokens con mas de determinado tiempo de antigüedad (ej 1 hora). haciendo que las sesiones se cierren y los administradores deben volver a loguearse para renovar su token.

## 0.3. Descripción general de códigos de error (1xx, 2xx, 4xx, 3xx, 5xx

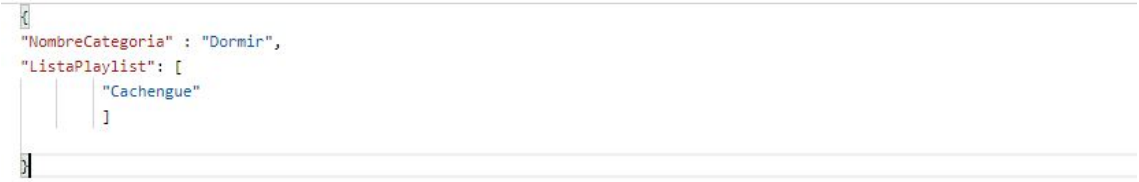
Se describen uno a uno en cada caso.

# 1. Descripción de los Resources

## 1.1. HTTP POST api/Categoria

### 1.1.1. Headers

### 1.1.2. Body de entrada




### 1.1.3. Códigos de Respuesta

- **201 Created** La categoria es correcta y se agrega.
- **400 Bad Request** Formato de JSON Incorrecto.
- **404 Not Found** No existe el nombre de la Categoría o No existe el nombre alguna Playlist enviada en el body.
- **409 Conflict** Alguna playlist enviada en el body ya fue agregada a la categoría.
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.2. HTTP GET api/Categoria

### 1.2.1. Headers

### 1.2.2. Params

Nombre = Dormir.

### 1.2.3. Códigos de Respuesta

- **200 OK** La categoria es correcta y devuelve el nombre de la misma junto a su lista de playlist en formato JSON.

  **Ejemplo de salida:**

  "ListaPlaylist":[
  "ListaAudios":[],"Nombre":Çachengue","Descripcion":"Para Escabiar y pasarla bien con tus panas",Ürl":"...

  ]
  ,"NombreCategoria":0

  En este caso devolvió la Categoria 0 que se mapea a Dormir(Enumerado) en dicha categoria se encuentran la playlist Cachengue que aun no tiene ningún audio.

- **400 Bad Request** Formato de Query Param Incorrecto.
- **404 Not Found** No existe el nombre de la Categoría.
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.3. HTTP POST api/Audios

### 1.3.1. Headers

Niguno.

### 1.3.2. Body de entrada

JSON con el siguiente formato:


```
{
 "Nombre": "Besame",
 "Duracion":3,
 "UnidadDeTiempo": "Minuto",
 "NombreCreador": "El Reja",
  "UrlImagen": "https://res.cloudinary.com/bidoware/image/upload/v1618153761/ImagenesAudios/elrejabesame_s4zdgd.jpg",
  "UrlMp3" : "https://res.cloudinary.com/bidoware/video/upload/v1618153735/Audios/El_Reja_-_Besame_dxvxwg.mp3"
}
```


### 1.3.3. Códigos de Respuesta

- **201 Created** El audio es correcto y se agrega.
- **400 Bad Request** Formato de JSON Incorrecto.
- **404 Not Found** Parte de la información del audio es errónea, La unidad de tiempo posiblemente no sea ni minuto ni hora o directamente esta vacia.
- **409 Conflict** El Audio enviado en el body ya fue agregado anteriormente.
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.4. HTTP GET api/Audios

### 1.4.1. Headers

Ninguno.

### 1.4.2. Body de entrada

Ninguno.

### 1.4.3. Códigos de Respuesta

- **200 OK** Devuelve la lista de audios del sistema en formato JSON.

  **Ejemplo de salida:**


```
[
    {
        "nombre": "Besame",
        "duracion": 3,
        "unidadDeTiempo": 0,
        "nombreCreador": "El Reja",
        "urlImagen": "https://res.cloudinary.com/bidoware/image/upload/v1618153761/ImagenesAudios/elrejabesame_s4zdgd.jpg",
        "urlMp3": "https://res.cloudinary.com/bidoware/video/upload/v1618153735/Audios/El_Reja_-_Besame_dxvxwg.mp3"
    },
    {
        "nombre": "Soltero Hasta la Tumba",
        "duracion": 4,
        "unidadDeTiempo": 0,
        "nombreCreador": "El Reja",
        "urlImagen": "https://res.cloudinary.com/bidoware/image/upload/v1618094458/ImagenesAudios/ElRejaSolteroHastaLaTumba_ajfvmk.jpg",
        "urlMp3": "https://res.cloudinary.com/bidoware/video/upload/v1618091230/Audios/ElRejaSolteroHastaLaTumba_uirmx4.mp3"
    }
]
```


  En este caso la lista de Audios esta compuesta por las canciones Besame y Soltero Hasta la Tumba ambas del reja.

- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.5. HTTP POST api/Playlist

### 1.5.1. Headers

Niguno.

### 1.5.2. Body de entrada

JSON con el siguiente formato:


```
{
 "Nombre": "Besame",
 "Duracion":3,
 "UnidadDeTiempo": "Minuto",
 "NombreCreador": "El Reja",
  "UrlImagen": "https://res.cloudinary.com/bidoware/image/upload/v1618153761/ImagenesAudios/elrejabesame_s4zdgd.jpg",
  "UrlMp3" : "https://res.cloudinary.com/bidoware/video/upload/v1618153735/Audios/El_Reja_-_Besame_dxvxwg.mp3"
}
```


### 1.5.3. Códigos de Respuesta

- **201 Created** La playlist es correcta y se agrega.
- **400 Bad Request** Formato de JSON Incorrecto.
- **404 Not Found** Audios pertenecientes a la playlist a agregar no existen. Antes de agregar una playlist se deben agregar todos los audios asociados.
- **409 Conflict** la playlst enviado en el body ya fue agregado anteriormente.
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.6. HTTP GET api/Playlist

### 1.6.1. Headers

Ninguno.

### 1.6.2. Body de entrada

Ninguno.

### 1.6.3. Códigos de Respuesta

- **200 OK** Devuelve la lista de platylist del sistema en formato JSON.

  **Ejemplo de salida:**


```
[
    {
        "nombre": "Besame",
        "duracion": 3,
        "unidadDeTiempo": 0,
        "nombreCreador": "El Reja",
        "urlImagen": "https://res.cloudinary.com/bidoware/image/upload/v1618153761/ImagenesAudios/elrejabesame_s4zdgd.jpg",
        "urlMp3": "https://res.cloudinary.com/bidoware/video/upload/v1618153735/Audios/El_Reja_-_Besame_dxvxwg.mp3"
    },
    {
        "nombre": "Soltero Hasta la Tumba",
        "duracion": 4,
        "unidadDeTiempo": 0,
        "nombreCreador": "El Reja",
        "urlImagen": "https://res.cloudinary.com/bidoware/image/upload/v1618094458/ImagenesAudios/ElRejaSolteroHastaLaTumba_ajfvmk.jpg",
        "urlMp3": "https://res.cloudinary.com/bidoware/video/upload/v1618091230/Audios/ElRejaSolteroHastaLaTumba_uirmx4.mp3"
    }
]
```


  En este caso la lista de Audios esta compuesta por las canciones Besame y Soltero Hasta la Tumba ambas del reja.

- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.7. HTTP POST api/Psicologo

### 1.7.1. Headers

Ninguno.

### 1.7.2. Body de entrada

```
{
  "Email":"elviejopsicologo@gmail.com",
   "Nombre": "Roku",
   "TipodeConsulta": "VideoLlamada",
   "ProblematicasEspecializadas": [
         "Depresion",
         "Estres",
         "Ansiedad"
         ],
  "FechaIngreso": "1995-05-05",
  "DireccionFisica": "yaguaron 1415"

}

```

### 1.7.3. Códigos de Respuesta

- **201 Created** Crea un psicologo nuevo.
- **401 Token Inexistente** El token enviado no es valido.
- **404 Not Found** No se Reconoce la problemática el Modo de Consulta.
- **409 Conflict** Ya existe un psicologo con ese nombre.

- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.8. HTTP POST api/Psicologo

### 1.8.1. Headers

Token obtenido por Administrador previamente.

### 1.8.2. Body de entrada

```
{
  "Email":"elviejopsicologo@gmail.com",
   "Nombre": "Roku",
   "TipodeConsulta": "VideoLlamada",
   "ProblematicasEspecializadas": [
         "Depresion",
         "Estres",
         "Ansiedad"
         ],
  "FechaIngreso": "1995-05-05",
  "DireccionFisica": "yaguaron 1415"

}

```

### 1.8.3. Códigos de Respuesta

- **201 Created** Crea un psicologo nuevo.
- **401 Token Inexistente** El token enviado no es valido.
- **404 Not Found** No se Reconoce la problemática el Modo de Consulta.
- **409 Conflict** Ya existe un psicologo con ese nombre.

- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.9. HTTP DELETE api/Psicologo

### 1.9.1. Headers

Token obtenido por Administrador previamente.

### 1.9.2. Body de entrada

Ninguno.

### 1.9.3. Query Param

El Email Del Psicologo a Obtener.

### 1.9.4. Códigos de Respuesta

- **200 OK** borro el psicologo indicado.
- **401 Token Inexistente** El token enviado no es valido.
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.10. HTTP POST api/Solicitud

### 1.10.1. Headers

Niguno.

### 1.10.2. Body de entrada

```
{
   "Celular": "094556723",
   "Email":"maxi@gmail.com",
   "Nombre": "Maxi",
   "FechaNacimiento":  "1982-05-05 ",
   "Problematica": "Depresion"
}
```

### 1.10.3. Códigos de Respuesta

- **201 Created** Crea una nueva CITA.
- **404 Not Found** No se Reconoce la problemática o no se encontro ningun psicologo disponible que la trate
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.11. HTTP POST api/Login

### 1.11.1. Headers

Niguno.

### 1.11.2. Body de entrada

### 1.11.3. Query Parmas

Email y Username del Administrador.

```
{
    "Celular": "094556723",
    "Email":"maxi@gmail.com",
    "Nombre": "Maxi",
    "FechaNacimiento":  "1982-05-05 ",
    "Problematica": "Depresion"
}
```

### 1.11.4. Códigos de Respuesta

- **200 OK** Las credenciales son correctas y devuelve un token unico.
- **404 Not Found** No se Reconoce la problemática o no se encontro ningun psicologo disponible que la trate
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

## 1.12. HTTP POST api/Administrador

### 1.12.1. Headers

Niguno.

### 1.12.2. Body de entrada

### 1.12.3. Query Parmas

Email y Username del Administrador.

```
{
  "Nombre": "Carlos",
  "Email":"calrlitos@gmail.com",
  "Password": "cerrolargo"
}

```

### 1.12.4. Códigos de Respuesta

- **201 Created** Se creo el administrador con dichas credenciales.
- **409 Conflict** Ya existe un administrador con dichas credenciales.
- **500 Internal Server Error** Ocurrió un error interno como puede ser que el motor de base de datos este caido.

Incluimos en este documento las funcionalidades que nos parecian claves.

# Bibliografía

[1] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, eight ed. McGraw-Hill, 2014.

[2] I. Sommerville, *Software Engineering*, 10th ed. Pearson, 2015.