

Primera parte de la práctica de Programming for mobile devices Curso 2021/2022

Librería de conexión a los servicios web (2,5 puntos sobre los 10 de la práctica)

En esta primera parte de la práctica se pide desarrollar una librería para conectar con los servicios REST disponibles en la siguiente URL

<https://sanger.dia.fi.upm.es/pmd-task>

Dicha implementación se realizará en Java y sin el uso de librerías adicionales para la conexión con los servicios. Es decir, se pide crear las llamadas HTTP necesarias para el acceso a los servicios, descargar la información recibida, procesarla y transformar los resultados en formato json obtenidos a objetos Java usando la librería GSON.

La librería que se desarrolle será usada en la segunda parte de la práctica como modelo de la aplicación Android a desarrollar.

Los servicios REST a los que se accederá se componen de un recurso principal: el Artículo.

Además, se dispone de un método 'login' de autenticación y un método que permite listar artículos. Las URL y opciones de los servicios REST se describen a continuación:

1) Método de login:

El método de login está accesible en la URI:

<https://sanger.dia.fi.upm.es/pmd-task/login>

Dicho método es una operación POST que recibe un objeto json (application/json) con los atributos 'username' y 'passwd':

```
{ 'username': 'NOMBRE_DE_USUARIO',  
  'passwd': 'CONTRASEÑA' }
```

Si el formato de conexión es correcto, el servicio devolverá:

- Un error 401 si los datos de acceso son incorrectos
- Un objeto json resultado con los datos de sesión para el usuario que se ha autenticado :

```
{ "user": "10",  
  "group": "10",  
  "username": "DEV_TEAM_01",  
  "Authorization": "AUTH_DESCXXX",  
  "apikey": "APIKEY_XYXYXYXYXYXY ",  
  "expires": "2021-11-11 02:39:51" }
```

Estos datos de conexión serán necesarios para el acceso a los recursos de manera autenticada.

- 2) Método de consulta del listado de artículos que está accesible en la URI
`https://sanger.dia.fi.upm.es/pmd-task/articles/{limit}/{offset}`

Dicho método es una operación GET que devuelve un listado de objetos json. Cada objeto corresponde a un artículo. En este listado se obtiene un subconjunto de los atributos de los artículos. Dicho subconjunto se compondrá de un número `{limit}` de artículos, empezando desde el artículo `{offset}`.

- 3) El recurso artículo está accesible en la URI:
`https://sanger.dia.fi.upm.es/pmd-task/article/{id}`

Dicho recurso tiene implementadas las operaciones GET, POST y DEL. que devuelve un listado de objetos json. Cada objeto corresponde a un artículo. En este listado se obtiene.

La operación GET recibe el parámetro `{id}` en la url, y devuelve la información completa del artículo (todos sus campos).

La operación DEL recibe el parámetro `{id}` en la url, y realiza el borrado la información del artículo en el servidor.

La operación POST realiza la creación de un artículo nuevo en el servidor, recibe un objeto json con todos los campos del objeto artículo a excepción del id, que es generado en el servidor. Como resultado devuelve un objeto json que incluye el id del nuevo artículo creado. En esta operación POST si se incluye el id en el objeto enviado al servidor, se intenta hacer la actualización del objeto enviado en el servidor usando los nuevos datos recibidos en el servicio.

Tanto el método 'articles' como el recurso 'article' devuelven dentro del objeto json un atributo que contiene una imagen. Dicha imagen va codificada como un String en base64, por lo que, en esta primera entrega, debe ser tratada como un campo String. Para la segunda entrega se proporcionará el código auxiliar necesario para su transformación en un objeto Bitmap y viceversa.

Además el método 'articles' como el recurso 'article' en la operación GET funcionan sin autenticación, obteniendo artículos del servidor.

Pero los métodos DEL y POST requieren una cabecera de autenticación (Authorization) que incluya como valor un String compuesto con parte de los atributos obtenidos al realizar el login. Ejemplo del valor de la cabecera Authorization:

```
AUTH_DESCXXX apikey=APIKEY_XYXYXYXYXYXY
```

El funcionamiento sin autenticación usando GET obtiene artículos de todos los usuarios. Mientras que si se usa autenticación se obtendrán sólo artículos del usuario autenticado.

Los métodos DEL y POST requieren siempre autenticación (se recibirá un error 401 en caso contrario). Además un usuario sólo puede eliminar y actualizar artículos que le pertenezcan (se recibirá un error 401 en caso contrario). Esto es así para evitar actualizar y/o borrar artículos de otros grupos.

Para esta entrega se pide que cada grupo desarrolle en Java

- 1) Un método login que realice una conexión POST al método del servicio web correspondiente y sea capaz de almacenar los datos de autenticación recibidos (apikey, user, username y authorization). **(0,5 puntos)**
- 2) Un objeto de modelo Artículo que sea capaz de almacenar la información descargada del listado de artículos y de un recurso artículo. Y además que sea capaz de transdor. **(0,5 puntos)**
- 3) Un método de listado de artículos capaz de descargar un listado de n artículos y devolverlo como una lista de objetos java Artículo. **(0,5 puntos)**
- 4) Un método que consulte (GET) un recurso artículo del servidor y lo devuelva como un objeto java de tipo Artículo. **(0,5 puntos)**
- 5) Un método que permita crear y actualizar un artículo en el servidor. Este método recibirá un objeto Java de tipo Artículo, y enviará una petición POST al recurso articulo (requiere la cabecera de autenticación), teniendo en cuenta que si el artículo es de nueva creación el contenido json del artículo (Gson.toJson(objeto)) no debe incluir el id, mientras que si es para actualización sí deberá incluir el campo id. **(0,5 puntos)**

Se recomienda probar los servicios proporcionados con alguna herramienta externa (por ejemplo Advanced Rest Client) para ver su funcionamiento y los objetos necesarios para su correcto funcionamiento así como los objetos que devuelven).