



# TFG DE UNA API REST CON SPRING BOOT

CURSO 2022-23

## ENDPOINTS - USER

En las siguientes tablas se describen los diferentes endpoints que serás capaz de realizar con el user-controller.

HTTP	URL	Descripción
POST	/auth/login	<p>Este método sirve para logear a un usuario ya creado</p> <p>Si los datos son correctos se loguea correctamente, devuelve <i>201</i> si el usuario se ha logueado correctamente y devuelve sus datos.</p> <p>Si hay algún error en los datos que nos envían, debe devolver <i>401 No Authenticate</i>.</p> <p>El modelo de entrada será:</p> <pre>{   "username": david,   "password": 1234 }</pre> <p>El modelo de respuesta será:</p> <pre>{</pre>

		<pre> "id": "681903b4-dc43-43c2-942c-86b81b15d1fc", "username": "marta", "email": "marta@gmail.com", "avatar": "marta.jpg", "biography": "I am Marta, a marketing specialist who loves traveling and trying new foods from different cultures.", "phone": "406038371", "birthday": "17/07/2000", "myPost": [   { "image": "REAL_SOCIEDAD.jpg" },   { "image": "joao_felix.jpg" },   { "image": "salah.jpg" },   { "image": "van_dike.jpg" } ], "roles": [ "ADMIN", "USER" ], "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJzdWIiOiI2ODE5MDNiNC1kYzQzLTQzYzItOTQyYy04NmI4MWIxNWQxZmMiLCJpYXQiOiJlOTYzMTY4NjU5NzIyOX0.t0sbXdUeL4Gn2SOqe-UgB-OaHyiqcJVMV7ZZrEO3OKbeyrrUSrJ6Lmr9peNDuAClu2VVNf829gzA4wIkKO2DPA", "refreshToken": "f346bd39-297f-4a38-bc47-6e530bfe5e7a" } </pre>
POST	/auth/register	<p>Este método envía un código de verificación al email del usuario para poder crear su cuenta.</p> <p>El json de entrada:</p> <pre> {   "username": "maylor",   "email": "maylorbustamante2001@gmail.com",   "phone": "609835692",   "password": "Maylor15",   "verifyPassword": "Maylor15" } </pre> <p>Código de respuesta 200 : Se ha enviado correctamente el código de verificación.</p> <p>El modelo de respuesta será un texto:</p> <pre> The code was send to email: maylorbustamante2001@gmail.com. Please check your message email or spam folder. </pre> <p>Código de respuesta 400 : No se han introducidos los datos correctamente o el email no existe. Observa la lista de errores.</p> <p>Código de respuesta 401: Has intentado realizar la petición borrando un atributo del post.</p>

		<p>Código de respuesta 405 : Estás intentado hacer la petición de POST a otra distinta, ejemplo GET.</p>
POST	/auth/verifyCode/{code}	<p>Este método crea la cuenta de un usuario.</p> <p>Código de respuesta 201: Se ha creado un nuevo usuario.</p> <pre>{   "id": "c0a800a3-88b0-1af5-8188-b10ba78a0000",   "username": "maylor",   "avatar": "avatar.png",   "roles": [     "USER"   ] }</pre> <p>Código de respuesta 400: El código introducido no pertenece a ningún usuario o es incorrecto.</p>
POST	/refresh/token	<p>Este método sirve para realizar el token de refresco al usuario logueado.</p> <p>El json de entrada será:</p> <pre>{   "refreshToken": "{{refresh_token}}" }</pre> <p>201: El token se ha refrescado correctamente. El modelo de respuesta será:</p> <pre>{   "id": "681903b4-dc43-43c2-942c-86b81b15d1fc",   "username": "marta",   "email": "marta@gmail.com",   "avatar": "marta.jpg",   "biography": "I am Marta, a marketing specialist who loves traveling and trying new foods from different cultures.",   "phone": "406038371",   "birthday": "17/07/2000",   "myPost": [     { </pre>

		<pre> "image": "REAL_SOCIEDAD.jpg" }, { "image": "joao_felix.jpg" }, { "image": "salah.jpg" }, { "image": "van_dike.jpg" } ], "roles": [ "ADMIN", "USER" ], "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJzdWIiOiI2ODE5MDNiNC1kYzQzLTQzYzltOTQyYy04NmI4MWlxNWQxZmMiLCJpYXQiOiJlOTc3MTk5LmV4cCI6MTY4NjU5ODYxOX0._rEeXUIWzyJ87imvudMguasaaCySpzUIJX2EZSehszT8jlsI7m3t7mtgRqpWU5i0bD8fLiNIEw-BX08Kj5N2aA", "refreshToken": "d0e958f7-ab57-44a4-97f2-26d0e854e94c" } </pre> <p>401: No estás logueado u autenticado o el token de refresco ha expirado.</p>
POST	/user/follow/{id}	<p>Este método implementa seguir a un usuario o dejar de seguirlo.</p> <p>Código de respuesta 201: Se ha dado/quitado el follow al usuario destinatario. El modelo de respuesta será:</p> <pre> {   "Now, you follow at @username" } </pre> <p>401: No estás logueado.</p> <p>404: El id del usuario destinatario a darle follow / unfollow no existe.</p> <p>403: No puedes seguirte a ti mismo.</p>
POST	/banUserByAdmin/{id}	<p>Este método banea o quita el ban a un usuario de la aplicación.</p> <p>Código de respuesta 201: El usuario se ha baneado/desbanear correctamente. El modelo de respuesta será:</p>

		<pre>{   "id": "7495ac21-9cef-4655-a816-29aleee80841",   "username": "alejandro",   "avatar": "alejandro.jpg",   "enabled": true,   "roles": [     "USER"   ] }</pre> <p>404: No se ha encontrado el usuario</p> <p>401: No estás logueado</p> <p>403: No tienes permisos para realizar esta petición</p>
POST	/changeRole/{id}	<p>Este método añade o elimina el rol de ADMIN a un usuario</p> <p>Código de respuesta 201: Al usuario se le ha añadido o quitado el rol de ADMIN. El modelo de respuesta será:</p> <pre>{   "id": "7495ac21-9cef-4655-a816-29aleee80841",   "username": "alejandro",   "avatar": "alejandro.jpg",   "enabled": true,   "roles": [     "USER"   ] }</pre> <p>404: No se ha encontrado el usuario</p> <p>401: No estás logueado</p> <p>403: No tienes permisos para realizar esta petición</p>
POST	/edit/avatar	<p>Este método lo que hace es cambiar a tu avatar.</p> <p>Código de respuesta 201: Se ha cambiado el avatar con éxito. El modelo de respuesta será:</p> <pre>{   "avatar": "5842fe0ea6515b1e0ad75b3c.png" }</pre> <p>400: No has proporcionado la imagen nueva.</p> <p>401: No estás logueado.</p>
PUT	/user/c	<p>Este método lo que hace es cambiar tu contraseña.</p>

	changePassword	<p>El json de entrada será:</p> <pre>{   "oldPassword": "M1n2345",   "newPassword": "MaylorDaa",   "verifyNewPassword": "MaylorDaa" }</pre> <p>Código de respuesta 201: Se ha cambiado la contraseña con éxito. El modelo de respuesta será:</p> <pre>{   "username": "wbeetham0",   "avatar": "avatar.png",   "enabled": true,   "roles": [     "ADMIN",     "USER"   ] }</pre> <p>400: Estás otorgando datos erróneos.</p> <p>401: No estás logueado.</p>
PUT	/edit/bio	<p>Este método lo que hace es editar tu biografía.</p> <p>El json de entrada:</p> <pre>{"biography": "I am new here guys"}</pre> <p>Código de respuesta 201: Se ha cambiado la biografía con éxito. El modelo de respuesta será:</p> <pre>{   "descripcion": "este es mi nueva descripción" }</pre> <p>400: Estás otorgando datos erróneos.</p> <p>401: No estás logueado.</p>
PUT	/edit/phone	<p>Este método lo que hace es editar tu número de teléfono.</p> <p>El json de entrada será:</p> <pre>{   "phone": "608991690" }</pre>

		<p>Código de respuesta 201: Se ha cambiado el número de teléfono con éxito. El modelo de respuesta será:</p> <pre>{   "phone": "609835694" }</pre> <p>400: Estás otorgando datos erróneos.</p> <p>401: No estás logueado.</p>
PUT	/edit/birthday	<p>Este método lo que hace es editar tu fecha de cumpleaños.</p> <p>El json de entrada será:</p> <pre>{   "birthday": "10/12/2023" }</pre> <p>Código de respuesta 201: Se ha cambiado la fecha de cumpleaños con éxito. El modelo de respuesta será:</p> <pre>{   "birthday": "17/12/2001" }</pre> <p>400: Estás otorgando datos erróneos.</p> <p>401: No estás logueado.</p>
DELETE	/user/{idU}	<p>Este método elimina o da de baja a un usuario.</p> <p>Código de respuesta 204: El usuario se ha eliminado correctamente. El modelo de respuesta será void.</p> <p>404: No se ha encontrado el usuario</p> <p>403: No tienes permisos para realizar esta petición</p> <p>401: No estás logueado.</p>
GET	/user/followers/{id}	<p>Este método obtiene una lista paginada de los followers de un determinado usuario.</p> <p>Código de respuesta 200: Se ha obtenido la lista paginada correctamente. El modelo de respuesta será:</p> <pre>{   "content": [     {       "id": "d8825758-d02a-4bcc-8146-95fb6fa3ded7",       "username": "bmacalester1",       "avatar": "avatar.png"     }   ],   "totalPages": 1 }</pre>

		<p>404: El id del usuario al que quieres ver los followers no existe o no tienes ningún seguidor en esta página o el usuario no existe.</p> <p>401: No estás logueado</p>
GET	/fav	<p>Este método obtiene los posts favoritos del usuario logueado.</p> <p>Código de respuesta 200: Se han devuelto los posts que el usuario le ha dado a favoritos. El modelo de respuesta será:</p> <pre>[   {     "id": 1,     "tag": "#pacheco",     "description": "RCD ESPANYOL",     "image": "PACHECO.jpg",     "uploadDate": "15/07/1998",     "author": "alejandro",     "idAuthor": "7495ac21-9cef-4655-a816-29aleeee80841",     "authorFile": "alejandro.jpg",     "likesByAuthor": [       "marta"     ],     "countLikes": 1,     "commentaries": [       {         "message": "Proin eu mi. Nulla ac enim. In tempor, turpis nec euismod scelerisque, quam turpis adipiscing lorem, vitae mattis nibh ligula nec sem.",         "authorName": "alejandro",         "authorFile": "alejandro.jpg",         "uploadCommentary": "23/04/2023"       }     ]   } ]</pre> <p>401: No estás logueado.</p> <p>404: La lista se encuentra vacía.</p>
GET	/user/follows/{id}	<p>Este método obtiene una lista paginada de los follows de un determinado usuario</p> <p>Código de respuesta 200: Se ha obtenido la lista paginada correctamente. El modelo de respuesta será:</p> <pre>{   "content": [     {       "id":</pre>



		<pre>"d8825758-d02a-4bcc-8146-95fb6fa3ded7",   "username": "bmacalester1",   "avatar": "avatar.png" }, {   "totalPages": 1 }</pre> <p>404: El id del usuario al que quieres ver los follows no existe o No sigues a nadie en esta página o el usuario no existe.</p> <p>401: No estás logueado.</p>
GET	/user	<p>Este método obtiene todos los datos de los usuarios creados.</p> <p>Código de respuesta 200: Se han devuelto los datos correctamente de todos los usuarios sólo para el administrador. El modelo de respuesta será:</p> <pre>[ {   "content": [     {       "id": "abb9feac-f0ec-45cf-91a9-5d21c789da2d",       "username": "ccliss6",       "createdAt": "22/02/2023 21:43:35",       "avatar": "avatar.png",       "birthday": "06/12/1997",       "enabled": true,       "roles": [         "USER"       ]     },     {       "id": "6e245baa-4865-45be-99f0-33d548b16887",       "username": "mdeverell7",       "createdAt": "22/02/2023 21:43:35",       "avatar": "avatar.png",       "birthday": "15/12/1991",       "enabled": true,       "roles": [         "USER"       ]     },     {       "id": "563d2700-0c3c-4276-a4c4-55f861ebe90a",       "username": "nchoppen8",       "createdAt": "22/02/2023 21:43:35",       "avatar": "avatar.png",       "birthday": "12/10/1988",       "enabled": true,       "roles": [         "USER"       ]     }   ] }</pre>

		<pre>         }       ],       "totalPages": 4,       "totalElements" : 3     }   ] </pre> <p>403: No tienes permisos para acceder a esta petición</p> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere</p> <p>404: La lista está vacía</p> <p>401: No estás logueado</p>
GET	/state/follow/user/{id}	<p>Este método devuelve true en caso de que el usuario logueado siga al usuario pasado el id o false en caso de que no.</p> <p>El código de respuesta 200: Se obtiene una respuesta correcta. El modelo de respuesta será:</p> <pre> true </pre> <p>401: No estas logueado</p> <p>400: El usuario pasado por id no existe</p> <p>403: No puedes consultar si te sigues a ti mismo ya que no es posible.</p>
GET	/me	<p>Este método obtiene todos los datos del usuario logueado, id, nombre, posts, comentarios de posts...</p> <p>Código de respuesta 200: Se han devuelto los datos correctamente del usuario logueado. El modelo de respuesta será:</p> <pre> [   {     "id": "d8825758-d02a-4bcc-8146-95fb6fa3ded7",     "username": "bmacalester1",     "createdAt": "22/02/2023 19:36:30",     "email": "bmacalester1@hotmail.com",     "avatar": "avatar.png",     "phone": "3011096944",     "birthday": "02/04/2002",     "enabled": true,     "myPost": [       {         "id": 3,         "tag": "#CR7",         "image": "cr7.jpg",         "uploadDate": "22/02/2023",         "author": "bmacalester1",         "idAuthor": </pre> <p>"d8825758-d02a-4bcc-8146-95fb6fa3ded7",</p>

		<pre> "authorFile": "avatar.png", "countLikes": 0, "commentaries": [   {     "id": 46,     "message": "engineer rich schemas",     "authorName": "cc1692e1-f031-4675-a0b1-96aeeada21aa",     "uploadCommentary": "22/02/2023"   } ], {   "id": 4,   "tag": "#CR7",   "image": "cr7.jpg",   "uploadDate": "22/02/2023",   "author": "bmacalester1",   "idAuthor": "d8825758-d02a-4bcc-8146-95fb6fa3ded7",   "authorFile": "avatar.png",   "countLikes": 0,   "commentaries": [     {       "id": 50,       "message": "engage 24/7 users",       "authorName": "cc1692e1-f031-4675-a0b1-96aeeada21aa",       "uploadCommentary": "22/02/2023"     }   ] }, "roles": [   "USER" ], "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9. eyJzdWIiOiJkODgyNTc1OC1kMDJhLTJY2MtODE0Ni05NWZiNmZhM 2RlZDciLCJpYXQiOiJlZ2NzcwOTFwMDgsImV4cCI6MTY3NzY5NTgwOH 0. XJofqY6kDfku9wWjK4r4Xfa1wN8QdTsf8_fXn6tDxOlNoCrZueh_ 6xVry1vybjmUYDvbrdNmHPc6-X_-PDK8g" } ]  403: No puedes acceder a esta petición  400: Estás intentando pasar cuerpo a la petición </pre>
GET	/info/user/{id}	<p>Este método trae el perfil con los datos de un usuario</p> <p>El código de respuesta 200: Se ha obtenido los datos correctamente. El modelo de respuesta será:</p>

		<pre>[   {     "id": "51057cde-9852-4cd5-be5e-091979495656",     "username": "wbeetham0",     "createdAt": "23/02/2023 19:47:32",     "email": "wbeetham0@gmail.com",     "avatar": "avatar.png",     "biography": "Hi i am a new in that!",     "phone": "3908006159",     "birthday": "29/12/2004",     "enabled": true,     "myPost": [       {         "id": 1,         "tag": "#CR7",         "image": "cr7.jpg",         "uploadDate": "23/02/2023",         "author": "wbeetham0",         "idAuthor": "51057cde-9852-4cd5-be5e-091979495656",         "authorFile": "avatar.png",         "countLikes": 0,         "commentaries": [           {             "id": 41, models",             "message": "innovate intuitive             "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d",             "authorFile": "wbeetham0.jpg"             "uploadCommentary": "23/02/2023"           },         ]       },       {         "id": 2,         "tag": "#CR7",         "image": "cr7.jpg",         "uploadDate": "23/02/2023",         "author": "wbeetham0",         "idAuthor": "51057cde-9852-4cd5-be5e-091979495656",         "authorFile": "avatar.png",         "countLikes": 0,         "commentaries": [           {             "id": 45, initiatives",             "message": "morph web-enabled             "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d",             "uploadCommentary": "23/02/2023"           },           {             "id": 70,             "message": "harness </pre>
--	--	--



		<pre> "uploadDate": "23/02/2023", "author": "wbeetham0", "idAuthor": "51057cde-9852-4cd5-be5e-091979495656", "authorFile": "avatar.png", "countLikes": 0, "commentaries": [ { "id": 41, "message": "innovate intuitive models", "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d", "authorFile": "wbeetham0.jpg" "uploadCommentary": "23/02/2023" }, ] }, { "id": 2, "tag": "#CR7", "image": "cr7.jpg", "uploadDate": "23/02/2023", "author": "wbeetham0", "idAuthor": "51057cde-9852-4cd5-be5e-091979495656", "authorFile": "avatar.png", "countLikes": 0, "commentaries": [ { "id": 45, "message": "morph web-enabled initiatives", "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d", "uploadCommentary": "23/02/2023" }, { "id": 70, "message": "harness transparent platforms", "authorName": "cc1692e1-f031-4675-a0b1-96aeeada21aa", "uploadCommentary": "23/02/2023" } ] }, { "id": 11, "tag": "#MESSI", "image": "messi.jpg", "uploadDate": "23/02/2023", "author": "wbeetham0", "idAuthor": "51057cde-9852-4cd5-be5e-091979495656", </pre>
--	--	---







		<p>]</p> <p>401: No estas logeado</p> <p>400: Estás otorgando datos erróneos</p> <p>404: Usuario no encontrado</p>
GET	/user/list/comments/{id}	<p>Este método obtiene una lista de los 3 últimos comentarios realizados por un usuario determinado.</p> <p>El código de respuesta 200: Se ha obtenido la lista correctamente. El modelo de respuesta será:</p> <pre>[   {     "id": 49,     "message": "Duis bibendum. Morbi non quam nec dui luctus rutrum. Nulla tellus.",     "authorName": "marta",     "author File": "marta.jpg",     "uploadCommentary": "23/04/2023"   },   {     "id": 50,     "message": "Praesent blandit. Nam nulla. Integer pede justo, lacinia eget, tincidunt eget, tempus vel, pede.",     "authorName": "marta",     "authorFile": "marta.jpg",     "uploadCommentary": "23/04/2023"   },   {     "id": 51,     "message": "Nulla ut erat id mauris vulputate elementum. Nullam varius. Nulla facilisi.",     "authorName": "marta",     "authorFile": "marta.jpg",     "uploadCommentary": "23/04/2023"   } ]</pre> <p>404: El id del usuario al que quieres ver los comentarios recientes no existe o El usuario no ha realizado ningún comentario aún.</p> <p>403: No tienes permiso para realizar esta petición</p> <p>401: No estas logeado</p>
GET	/user/list/state/now	<p>Este método obtiene una lista con la cantidad de usuarios baneados y no baneados.</p> <p>El código de respuesta 200: Se ha obtenido la lista correctamente. El modelo de respuesta será:</p> <pre>[   {</pre>

		<pre>         "state": "active",         "value":5       },       {         "state": "inactive/ban",         "value":5       },     ],   ] </pre> <p>401: No estás logueado</p> <p>403: No tienes permiso para realizar esta petición</p> <p>404: No hay usuarios en la base de datos</p>
GET	/user/create/year/{year}	<p>Este método obtiene una lista de los meses en la que se ha creado la cuenta de usuarios por el año que indiquemos.</p> <p>El código de respuesta 200: Se ha obtenido la lista correctamente.</p> <pre> [   {     "month": 4,     "total":10   } ] </pre> <p>401: No estás logueado.</p> <p>403: No tienes permiso para realizar esta petición.</p> <p>404: No hay usuarios en la base de datos.</p>

## ENDPOINTS - POST

En las siguientes tablas se describen los diferentes endpoints que serás capaz de realizar con el post-controller

HTTP	URL	Descripción
POST	/post/	<p>Este método sirve para subir un post</p> <p>Se necesita un file y un json que será:</p> <pre> {   "tag": "nuevoPost",   "description": "hola buenas" } </pre> <p>Si los datos son correctos se loguea correctamente, devuelve 201 si</p>

		<p>todo es correcto. El modelo de respuesta será:</p> <pre>{   {     "id": 1000,     "tag": "#messi",     "description": "Lorem Ipsum Dolor Sit Amet....",     "image": "5842fe0ea6515b1e0ad75b3c_317828.png",     "uploadDate": "23/02/2023",     "author": "wbeetham0",     "idAuthor": "51057cde-9852-4cd5-be5e-091979495656",     "authorFile": "avatar.png",     "countLikes": 0   } }</pre> <p>401: No estás logueado</p> <p>404: No estas enviando los datos correctamente</p>
POST	/post/like/{id}	<p>Este método envía un código de verificación al email del usuario para poder crear su cuenta.</p> <p>Código de respuesta 201: Se ha dado/quitado el like al post.</p> <p>El modelo de respuesta será:</p> <pre>{   {     "id": 15,     "tag": "#MESSI",     "image": "messi.jpg",     "uploadDate": "23/02/2023",     "author": "hpitt2",     "idAuthor": "eabbe12c-363e-49ff-8b0f-cad8daca9bf9",     "authorFile": "avatar.png",     "likesByAuthor": [       "wbeetham0"     ],     "countLikes": 1   } }</pre> <p>Código de respuesta 400: El id del post no existe</p> <p>Código de respuesta 201: No estás logueado</p> <p>Código de respuesta 405: Estás intentado hacer la petición de POST a otra distinta, ejemplo GET.</p>

POST	/commentary/{id}	<p>Este método agrega un comentario a un post</p> <p>El json de entrada será:</p> <pre>{   "message": "This is my first commentary here aaa!" }</pre> <p>Se ha agregado correctamente el comentario al post</p> <p>El modelo de respuesta será:</p> <pre>{   {     "id": 20,     "tag": "#MESSI",     "image": "messi.jpg",     "uploadDate": "23/02/2023",     "author": "bmacalester1",     "idAuthor":     "d8825758-d02a-4bcc-8146-95fb6fa3ded7",     "authorFile": "avatar.png",     "countLikes": 0,     "commentaries": [       {         "message": "This is my first commentary here aaa!",         "authorName": "wbeetham0",         "authorFile": "javier.jpg",         "uploadCommentary": "23/02/2023"       }     ]   } }</pre> <p>Código de respuesta 401: No estás logueado</p> <p>Código de respuesta 404: El id del post no existe</p>
DELETE	/user/{id}/user/{idU}	<p>Este método elimina un post de un usuario</p> <p>Código de respuesta 204: El post se ha eliminado correctamente. El modelo de respuesta será void.</p> <p>404: No se ha encontrado el usuario u post.</p> <p>403: No tienes permisos para realizar esta petición</p> <p>401: No estás logueado.</p>
GET	/user/followers/{id}	<p>Este método obtiene una lista paginada de los followers de un determinado usuario.</p> <p>Código de respuesta 200: Se ha obtenido la lista paginada correctamente. El modelo de respuesta será:</p>

		<pre>{   "content": [     {       "id": "d8825758-d02a-4bcc-8146-95fb6fa3ded7",       "username": "bmacalester1",       "avatar": "avatar.png"     }   ],   "totalPages": 1 }</pre> <p>404: El id del usuario al que quieres ver los followers no existe o no tienes ningún seguidor en esta página o el usuario no existe.</p> <p>401: No estás logueado</p>
DELETE	/delete/ commentary/{id}	<p>Este método obtiene una lista paginada de los followers de un determinado usuario.</p> <p>Código de respuesta 200: Se ha obtenido la lista paginada correctamente. El modelo de respuesta será:</p> <pre>{   "content": [     {       "id": "d8825758-d02a-4bcc-8146-95fb6fa3ded7",       "username": "bmacalester1",       "avatar": "avatar.png"     }   ],   "totalPages": 1 }</pre> <p>404: El id del usuario al que quieres ver los followers no existe o no tienes ningún seguidor en esta página o el usuario no existe.</p> <p>401: No estás logueado</p>
GET	/post/	<p>Este método obtiene todos los posts de los usuarios</p> <p>Código de respuesta 200: Se han devuelto los datos correctamente de todos los posts de los usuarios. El modelo de respuesta será:</p> <pre>[ {   "content": [     {       "id": 1,       "tag": "#CR7",       "image": "cr7.jpg",       "uploadDate": "23/02/2023",       "author": "wbeetham0",       "idAuthor":</pre>

		<pre> "51057cde-9852-4cd5-be5e-091979495656",   "authorFile": "avatar.png",   "countLikes": 0,   "commentaries": [     {       "message": "innovate intuitive models",       "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d",       "authorFile" : "mario.jpg"       "uploadCommentary": "23/02/2023"     }   ] }, {   "id": 2,   "tag": "#CR7",   "image": "cr7.jpg",   "uploadDate": "23/02/2023",   "author": "wbeetham0",   "idAuthor": "51057cde-9852-4cd5-be5e-091979495656",   "authorFile": "avatar.png",   "countLikes": 0,   "commentaries": [     {       "message": "morph web-enabled initiatives",       "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d",       "uploadCommentary": "23/02/2023"     },     {       "message": "harness transparent platforms",       "authorName": "cc1692e1-f031-4675-a0b1-96aeeada21aa",       "uploadCommentary": "23/02/2023"     }   ] }, {   "id": 3,   "tag": "#CR7",   "image": "cr7.jpg",   "uploadDate": "23/02/2023",   "author": "bmacalester1",   "idAuthor": "d8825758-d02a-4bcc-8146-95fb6fa3ded7",   "authorFile": "avatar.png",   "countLikes": 0,   "commentaries": [     {       "message": "engineer rich schemas",       "authorName": "cc1692e1-f031-4675-a0b1-96aeeada21aa",       "uploadCommentary": "23/02/2023"     },     { </pre>
--	--	---

		<pre> "message": "monetize best-of-breed eyeballs", "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d", "uploadCommentary": "23/02/2023" }, { "message": "strategize cross-media deliverables", "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d", "uploadCommentary": "23/02/2023" } ] } ], "totalPages": 8 } ] </pre> <p>401: No estás logueado.</p> <p>404: La lista se encuentra vacía.</p> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere.</p>
GET	/user/{username}	<p>Este método obtiene todos los posts de un usuario concreto de forma paginada.</p> <p>Código de respuesta 200: Se han devuelto los datos correctamente de todos los posts del usuario en cuestión. El modelo de respuesta será:</p> <pre> [ { "content": [ { "id": 1, "tag": "#CR7", "image": "cr7.jpg", "uploadDate": "23/02/2023", "author": "wbeetham0", "idAuthor": "51057cde-9852-4cd5-be5e-091979495656", "authorFile": "avatar.png", "countLikes": 0, "commentaries": [ { "message": "innovate intuitive models", "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d", "uploadCommentary": "23/02/2023" } ] } ] </pre>

		<pre> }, {   "id": 2,   "tag": "#CR7",   "image": "cr7.jpg",   "uploadDate": "23/02/2023",   "author": "wbeetham0",   "idAuthor": "51057cde-9852-4cd5-be5e-091979495656",   "authorFile": "avatar.png",   "countLikes": 0,   "commentaries": [     {       "message": "morph web-enabled initiatives",       "authorName": "abb9feac-f0ec-45cf-91a9-5d21c789da2d",       "uploadCommentary": "23/02/2023"     },     {       "message": "harness transparent platforms",       "authorName": "cc1692e1-f031-4675-a0b1-96aeeada21aa",       "uploadCommentary": "23/02/2023"     }   ] } ], "totalPages": 8 } ] </pre> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere</p> <p>404: La lista está vacía o el usuario no existe</p> <p>401: No estás logueado.</p>
GET	/user/post/grid/{username}	<p>Este método obtiene todas las imágenes de los posts de un usuario concreto.</p> <p>Código de respuesta 200: Se han devuelto los datos correctamente de todos los posts del usuario en cuestión. El modelo de respuesta será:</p> <pre> [   {     "image": "SPANYOL.jpg"   },   {     "image": "CAMPANIA.jpg"   },   {     "image": "de_bruyne.jpg"   },   {     "image": "toni_kross.jpg"   } ] </pre>



		<pre>} ]</pre> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere</p> <p>404: La lista está vacía</p> <p>401: No estás logueado</p>
GET	/details/{id}	<p>Este método obtiene los detalles de un post</p> <p>El código de respuesta 200: Se han devuelto los datos correctamente. El modelo de respuesta será:</p> <pre>{   "id": 1,   "tag": "#pacheco",   "description": "RCD ESPANYOL",   "image": "PACHECO.jpg",   "uploadDate": "15/07/1998",   "author": "alejandro",   "idAuthor": "7495ac21-9cef-4655-a816-29aleeee80841",   "authorFile": "alejandro.jpg" }</pre> <p>401: No estas logueado.</p> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere.</p> <p>404: El id del post no es correcto</p>
GET	/details/{id}/commentaries	<p>Este método obtiene todos los comentarios de un post .</p> <p>Código de respuesta 200: Sa han devuelto los datos correctamente de todos los comentarios de un post. El modelo de respuesta será:</p> <pre>{   "content": [     {       "id": 45,       "message": "Proin eu mi. Nulla ac enim. In tempor, turpis nec euismod scelerisque, quam turpis adipiscing lorem, vitae mattis nibh ligula nec sem.",       "id_author": "7495ac21-9cef-4655-a816-29aleeee80841",       "authorName": "alejandro",       "post_id": 1,       "uploadCommentary": "23/04/2023"     }, </pre>

		<pre> {   "id": 46,   "message": "Etiam vel augue. Vestibulum rutrum rutrum neque. Aenean auctor gravida sem.",   "id_author": "7495ac21-9cef-4655-a816-29aleeee80841",   "authorName": "alejandro",   "post_id": 1,   "uploadCommentary": "23/04/2023" }, {   "id": 52,   "message": "Aenean fermentum. Donec ut mauris eget massa tempor convallis. Nulla neque libero, convallis eget, eleifend luctus, ultricies eu, nibh.",   "id_author": "681903b4-dc43-43c2-942c-86b81b15d1fc",   "authorName": "marta",   "post_id": 1,   "uploadCommentary": "23/04/2023" } ], "totalPages": 1, "totalElements": 3 } </pre> <p>404: La lista está vacía</p> <p>401: No estas logeado</p>
GET	/all/comentarios	<p>Este método obtiene todos los comentarios</p> <p>El código de respuesta 200: Se han devuelto los datos correctamente de todos mis posts. El modelo de respuesta será:</p> <pre> [ {   "content": [     {       "id": 41,       "message": "Maecenas ut massa quis augue luctus tincidunt. Nulla mollis molestie lorem. Quisque ut erat.",       "id_author": "7495ac21-9cef-4655-a816-29aleeee80841",       "authorName": "alejandro",       "post_id": 17,       "uploadCommentary": "23/04/2023"     },     {       "id": 42,       "message": "Curabitur at ipsum ac tellus semper interdum. Mauris ullamcorper purus sit amet nulla. Quisque arcu libero, rutrum ac, lobortis vel, dapibus at, diam.", </pre>

		<pre>         "id_author":         "7495ac21-9cef-4655-a816-29aleeee80841",         "authorName": "alejandro",         "post_id": 9,         "uploadCommentary": "23/04/2023"     },     ],     "totalPages": 2,     "totalElements": 30 } ] </pre> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere.</p> <p>404: La lista está vacía.</p> <p>401: No estás logueado.</p>
GET	/total/post/{month}	<p>Este método obtiene el total de post en un mes concreto</p> <p>El código de respuesta 201: Se ha obtenido el total de post. El modelo de respuesta será:</p> <pre> {   {     10   } } </pre> <p>404: El mes no existe.</p> <p>401: No estás logueado.</p>

## ENDPOINTS - MESSAGE

En las siguientes tablas se describen los diferentes endpoints que serás capaz de realizar con el message-controller

HTTP	URL	Descripción
GET	/message/{idUser}	<p>Este método obtiene todos los mensajes de un chat</p> <p><b>Código de respuesta 200:</b> Se han devuelto los datos correctamente de todos mensajes de un chat. El modelo de respuesta será:</p>

		<pre>[ {   "content": [     {       "id": 1001,       "idUserWhoSendMessage": "7495ac21-9cef-4655-a816-29aleeee80841",       "usernameWhoSendMessage": "alejandro",       "avatarWhoSendMessage": "alejandro.jpg",       "bodyMessage": "Hola buenas, segunda mensaje",       "timeWhoSendMessage": "26/05/2023 21:46"     },     {       "id": 1002,       "idUserWhoSendMessage": "681903b4-dc43-43c2-942c-86b81b15d1fc",       "usernameWhoSendMessage": "marta",       "avatarWhoSendMessage": "marta.jpg",       "bodyMessage": "Hola buenas, segunda mensaje",       "timeWhoSendMessage": "26/05/2023 21:46"     }   ],   "totalPages": 8,   "totalElements":20 } ]</pre> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere.</p> <p>404: La lista está vacía</p> <p>401: No estás logueado</p>
POST	/message/{id}	<p>Este método envía un mensaje a otro usuario.</p> <p><b>Código de respuesta 201:</b>Se ha enviado el mensaje correctamente. El modelo de respuesta será:</p> <pre>{   "id": 1004,   "idUserWhoSendMessage": "681903b4-dc43-43c2-942c-86b81b15d1fc",   "usernameWhoSendMessage": "marta",   "avatarWhoSendMessage": "marta.jpg",   "bodyMessage": "Hola buenas",   "timeWhoSendMessage": "28/05/2023 00:56" }</pre> <p>401: No estas logeado u autenticado</p> <p>403: No puedes mandarte mensajes a ti mismo</p>

		404: No existe el usuario destinatario
DELETE	/message/{id}	<p>Este método elimina el mensaje de un chat.</p> <p>204: Se ha eliminado el mensaje correctamente</p> <p>404: El mensaje no existe</p> <p>401: No estás logueado</p> <p>403: No tienes permisos para realizar esta petición porque no se encuentra el mensaje dentro del chat</p>

## ENDPOINTS - CHAT

En las siguientes tablas se describen los diferentes endpoints que serás capaz de realizar con el chat-controller

HTTP	URL	Descripción
GET	/chat/	<p>Este método obtiene todos los chats de un usuario</p> <p><b>Código de respuesta 200:</b> Se han devuelto los datos correctamente de todos los chats del usuario. El modelo de respuesta será:</p> <pre>[   {     "content": [       {         "id": 1000,         "avatarUserDestinity": "avatar.jpg",         "usernameUserDestinity": "Unknown",         "lastMessage": "mensaje no borrar"       },       {         "id": 1003,         "avatarUserDestinity": "david.jpg",         "usernameUserDestinity": "david",         "lastMessage": "mensaje no borrar"       }     ],     "totalPages": 8   } ]</pre> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere.</p>

		<p>404: La lista está vacía</p> <p>401: No estás logueado</p>
DELETE	/chat/{id}	<p>Este método elimina el chat de un usuario</p> <p>204: Se ha eliminado el chat correctamente</p> <p>404: El chat no existe</p> <p>401: No estás logueado</p> <p>403: No tienes permisos para realizar esta petición porque no se encuentra el mensaje dentro del chat</p>

## ENDPOINTS - FILE

En las siguientes tablas se describen los diferentes endpoints que serás capaz de realizar con el file-controller

HTTP	URL	Descripción
GET	/download/{filename:..+}/	<p>Este obtiene la imagen de posts por su nombre.</p> <p><b>Código de respuesta 200:</b> Se han devuelto la imagen correctamente.</p> <p>El modelo de respuesta será un file.</p> <p>400: Estás intentando pasar cuerpo a la petición que no lo requiere.</p> <p>404: No se ha encontrado la imagen.</p> <p>401: No estás logueado.</p>