Endpoints

1. Autenticación

Login

Descripción: Permite a los usuarios iniciar sesión y obtener un token de acceso.

Método: POST

URL: http://localhost:19090/auth/login

Autenticación: No requiere token

Cuerpo de la solicitud:

json

{

"userName": "jperez",

"email": "leinadcastilq@gmail.com",

"password": "12345"

}

2. Usuarios

Obtener usuario

Descripción: Obtiene información de un usuario específico por su ID.

Método: GET

URL: http://localhost:19090/api/v1/usuarios/{id}

Autenticación: Requiere token Bearer

Ejemplo: GET http://localhost:19090/api/v1/usuarios/1

Actualizar usuario

Descripción: Actualiza la información de un usuario existente.

Método: PUT

URL: http://localhost:19090/api/v1/usuarios/{id}

Autenticación: Requiere token Bearer

Cuerpo de la solicitud:

json

{

"userName": "usuario1234",

"password": "12345",

"email": "usuario4@example.com",

"nombre": "Juan",

"apellidos": "Pérez Gómez",

"role": "USER"

}

Eliminar usuario

Descripción: Elimina un usuario del sistema.

Método: DELETE

URL: http://localhost:19090/api/v1/usuarios/{id}

Autenticación: Requiere token Bearer

Ejemplo: DELETE http://localhost:19090/api/v1/usuarios/2

3. Votos

Contar votos de fotografía

Descripción: Obtiene el número de votos que ha recibido una fotografía específica.

Método: GET

URL: http://localhost:19090/votos/{idFotografia}

Ejemplo: GET http://localhost:19090/votos/1

Registrar voto

Descripción: Permite a un usuario votar por una fotografía.

Método: POST

URL: http://localhost:19090/votos

Cuerpo de la solicitud:

json

{

"idFotografia": 11,

"ip": "192.168.1.15"

}

4. Fotografías

Obtener fotografía

Descripción: Obtiene los detalles de una fotografía específica.

Método: GET

URL: http://localhost:19090/api/v1/fotografias/{id}

Autenticación: Requiere token Bearer

Ejemplo: GET http://localhost:19090/api/v1/fotografias/1

Agregar fotografía

Descripción: Permite a un participante subir una nueva fotografía.

Método: POST

URL: http://localhost:19090/api/v1/fotografias/Participantes/Add

Autenticación: Requiere token Bearer

Cuerpo de la solicitud:

json

{

"descripcion": "dcastillo",

"idParticipante": 1,

"link": "12345",

"titulo": "12345",

"usuCre": "12345"

}

Actualizar fotografía

Descripción: Permite modificar los datos de una fotografía existente.

Método: PUT

URL: http://localhost:19090/api/v1/fotografias/Participantes/Update

Autenticación: Requiere token Bearer

Cuerpo de la solicitud:

json

{

"id": 6,

"descripcion": "dcastillo cambiado",

"idParticipante": 1,

"link": "12345 cambiado",

"titulo": "12345 cambiado",

"usuMod": "12345"

}

Eliminar fotografía

Descripción: Elimina una fotografía del sistema.

Método: DELETE

URL: http://localhost:19090/api/v1/fotografias/Participantes/Delete

Autenticación: Requiere token Bearer

Cuerpo de la solicitud:

json

{

"id": 8,

"usuMod": "jperez"

}

Obtener todas las fotografías

Descripción: Obtiene un listado de todas las fotografías en el sistema.

Método: GET

URL: http://localhost:19090/api/v1/fotografias/All

Obtener fotografías de un participante

Descripción: Obtiene todas las fotografías subidas por un participante específico.

Método: GET

URL: http://localhost:19090/api/v1/fotografias/Participantes/All/{idParticipante}

Autenticación: Requiere token Bearer

Ejemplo: GET http://localhost:19090/api/v1/fotografias/Participantes/All/1

Obtener fotografías por estado (ADMIN)

Descripción: Obtiene fotografías filtradas por estado (solo para administradores).

Método: GET

URL: http://localhost:19090/api/v1/fotografias/State/{idAdmin}/{estado}

Autenticación: Requiere token Bearer de administrador

Ejemplo: GET http://localhost:19090/api/v1/fotografias/State/11/PENDIENTE

Buscar fotografías por título o descripción

Descripción: Busca fotografías que coincidan con un término en su título o descripción.

Método: GET

URL: http://localhost:19090/api/v1/fotografias/All/{terminoBusqueda}

Ejemplo: GET http://localhost:19090/api/v1/fotografias/All/ca

Aprobar/Rechazar fotografía

Descripción: Permite a un administrador aprobar o rechazar una fotografía.

Método: PUT

URL: http://localhost:19090/api/v1/fotografias/Aprobar/{idAdmin}/{idFotografia}/{aprobada}

Autenticación: Requiere token Bearer de administrador

Ejemplo: PUT http://localhost:19090/api/v1/fotografias/Aprobar/13/17/false

Contar fotografías subidas por participante

Descripción: Obtiene el número de fotografías que ha subido un participante.

Método: GET

URL: http://localhost:19090/api/v1/fotografias/Contar/{idParticipante}

Autenticación: Requiere token Bearer

Ejemplo: GET http://localhost:19090/api/v1/fotografias/Contar/15

5. Concurso

Obtener información del concurso

Descripción: Obtiene los detalles del concurso actual.

Método: GET

URL: http://localhost:19090/concurso

Actualizar información del concurso

Descripción: Permite a un administrador actualizar los parámetros del concurso.

Método: PUT

URL: http://localhost:19090/concurso/Update/{idAdmin}

Autenticación: Requiere token Bearer de administrador

Cuerpo de la solicitud:

json

{

"descripcion": "Concurso de fotografía paisajística",

"fechaInicioEnvio": "2025-06-01",

"fechaFinEnvio": "2025-06-10",

"fechaInicioVotacion": "2025-06-11",

"fechaFinVotacion": "2025-06-20",

"fechaAnuncio": "2025-06-22",

"numeroFotografias": 3

}