

**UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE**



**Computing Department**

Advanced Web Development

**Project URI's Definition**

**Authors:**

Mateo Llumigusin

Gabriel López

Medranda Mateo

NRC: 23284

**Ecuador 2025-06-21**



```
const EnvironmentalPlanRouter = require("./routes/environmentalPlanRoutes");
const activityRouter = require("./routes/activityRoutes");
const controlRouter = require("./routes/controlRoutes");
const profileRoutes = require('./routes/profileRoutes');
const userRoutes = require('./routes/userRoutes');
const actionRoutes = require('./routes/actionRoutes');
const reminderRoutes = require('./routes/remindersRoutes');
const projectRoutes = require('./routes/projectRoutes');
const monitoringRoutes = require('./routes/monitoringRoutes');
const permitRoutes = require('./routes/permitRoutes');

app.use("/projects/:projectId",EnvironmentalPlanRouter);
app.use("/environmental-plans/:planId",activityRouter);
app.use("/activities/:activityId",controlRouter);
app.use('/api/profile', profileRoutes);
app.use('/api/user',userRoutes);
app.use('/api/action', actionRoutes);
app.use('/api/reminder', reminderRoutes);
app.use('/projects', projectRoutes);
app.use('/monitorings', monitoringRoutes);
app.use('/permits', permitRoutes);
```

- 1. Get List of Profiles

Uri name:	/profilesTable
Method	GET
Description	Get the rows of the table profiles to obtain the data of all profiles created
Req	none
Res	Json with the profiles data <pre>{   "profiles_name": "Administrador",   "profiles_state": "ACTIVE",   "profiles_id": 1 }, {   "profiles_name": "Administrador de Planes Ambientales",   "profiles_state": "ACTIVE",   "profiles_id": 3 }, {</pre>

	status 500 on a server error
--	------------------------------

- 
- 2. Get List of Profiles in HTML

Uri name:	/profilesTable
Method	GET
Description	Get the rows of the table profiles to obtain the data of all profiles created
Req	none
Res	<p>html with all the profiles created in the database in HTML format</p> <pre> &lt;tr&gt;   &lt;td&gt;Administrador&lt;/td&gt;   &lt;td&gt;Activo&lt;/td&gt;   &lt;td&gt;     &lt;button class="btn btn-sm btn-primary me-1 edit-profile"       data-id="1" data-name="Administrador"&gt;       &lt;i class="bi bi-pencil"&gt;&lt;/i&gt; Editar     &lt;/button&gt;     &lt;button class="btn btn-sm btn-info me-1 permits_view"       data-id="1" data-name="Administrador"&gt;       &lt;i class="bi bi-eye-fill"&gt;&lt;/i&gt; Ver Permisos     &lt;/button&gt;     &lt;button class="btn btn-sm btn-danger toggle-state"       data-id="1" data-state="ACTIVE"&gt;       &lt;i class="bi bi-check-circle"&gt;&lt;/i&gt; Desactivar     &lt;/button&gt;   &lt;/td&gt; &lt;/tr&gt;  &lt;tr&gt;   &lt;td&gt;Revisiones&lt;/td&gt;   &lt;td&gt;Activo&lt;/td&gt;   &lt;td&gt;     &lt;button class="btn btn-sm btn-primary me-1 edit-profile"       data-id="2" data-name="Revisiones"&gt;       &lt;i class="bi bi-pencil"&gt;&lt;/i&gt; Editar   &lt;/td&gt; &lt;/tr&gt; </pre> <p>status 500 on a server error</p>

- 3. Get Permits

Uri name:	/permits
Method	GET
Description	Get the rows of the table profiles to obtain the data of all permits of a profile, or all existing profiles
Req	id
Res	status 200 Json with all the permits name and values if its an number it searches in database the permits of specific profile

```
"Proyectos": {
  "profiles_readprojects": {
    "permit_name": "Ver Proyectos",
    "value": true
  },
  "profiles_createprojects": {
    "permit_name": "Generar Proyectos",
    "value": true
  },
  "profiles_updateprojects": {
    "permit_name": "Actualizar Proyectos",
    "value": true
  },
  "profiles_deleteprojects": {
    "permit_name": "Eliminar Proyectos",
    "value": true
  }
},
"Planes Ambientales": {
```

if it's anything else, gets the permits with a value of false in them

```
{
  "Proyectos": {
    "profiles_readprojects": {
      "permit_name": "Ver Proyectos",
      "value": false
    },
    "profiles_createprojects": {
      "permit_name": "Generar Proyectos",
      "value": false
    },
    "profiles_updateprojects": {
      "permit_name": "Actualizar Proyectos",
      "value": false
    },
    "profiles_deleteprojects": {
      "permit_name": "Eliminar Proyectos",
      "value": false
    }
  },
  "Planes Ambientales": {
    "profiles_readambientalplans": {
      "permit_name": "Ver Planes Ambientales",
      "value": false
    }
  }
}
```

status 500 on a server error

- 4. Create a Profile

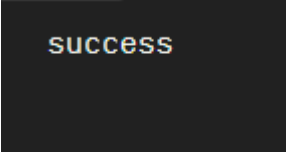
Uri name:	/profiles
Method	POST
Description	Creates a profile
Req	profile_name array selectedPermits
Res	status 200 on profile created <pre>{   "status": "ok" }</pre> status 500 on a server error

- 5. Update a Profile

Uri name:	/profiles/:id
Method	PUT
Description	Update a profile
Req	profile_name array selectedPermits
Res	status 200 on profile updated <pre>{   "message": "Profile updated successfully." }</pre> status 500 on a server error

- 6. Logical delete of a Profile

Uri name:	/profiles/:id/state
Method	PATCH
Description	changes the value of the state of a certain profile

Req	id state
Res	status 200 on profile changed  status 500 on a server error

## Users URIs

- 7. Get Users

Uri name:	/users
Method	GET
Description	Get the rows of the table users to obtain the data of all users created
Req	none
Res	status 200 html with the json of all users  status 500 on a server error

- 8. Get Users Table

Uri name:	/usersTable
Method	GET
Description	Get the rows of the table users to obtain the data of all users created
Req	none
Res	<p>status 200 html with all the users created in the database</p> <pre> &lt;tr&gt;   &lt;td&gt;1&lt;/td&gt;   &lt;td&gt;0650160205&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Administrador&lt;/td&gt;   &lt;td&gt;juan@example.com&lt;/td&gt;   &lt;td&gt;123456789&lt;/td&gt;   &lt;td&gt;Activo&lt;/td&gt;   &lt;td&gt;     &lt;button class="btn btn-sm btn-primary me-1 edit-user"       data-id="1" data-name="Juan"       data-surname="Pérez" data-personalId="0650160205"     &gt;       &lt;i class="bi bi-pencil"&gt;&lt;/i&gt; Editar     &lt;/button&gt;     &lt;button class="btn btn-sm btn-danger toggle-state"       data-id="1" data-state="ACTIVE"&gt;       &lt;i class="bi bi-check-circle"&gt;&lt;/i&gt; Desactivar     &lt;/button&gt;   &lt;/td&gt; &lt;/tr&gt;  &lt;tr&gt;   &lt;td&gt;8&lt;/td&gt;   &lt;td&gt;0650160208&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Revisiones&lt;/td&gt; </pre> <p>status 500 on a server error</p>

- 9. Get the profiles

Uri name:	/profileslist
Method	GET
Description	Get the profiles name to select one for the creation of an user
Req	none
Res	status 200 Json with all profiles in existence

	<pre>[   {     "id": 1,     "name": "Administrador"   },   {     "id": 5,     "name": "Administrador de Acciones"   },   {     "id": 3,     "name": "Administrador de Planes Ambientales"   },   {     "id": 2,     "name": "Administrador de Proyectos"   } ]</pre> <p>status 500 on a server error</p>
--	--

- 10. Get the profiles to choose by the user

Uri name:	/profileslist
Method	GET
Description	Get the profiles name to select one for the creation of an user
Req	none
Res	<p>status 200 html with all the profiles created in the database, name and id value</p> <pre>&lt;option value="1"&gt;Administrador&lt;/option&gt;  &lt;option value="2"&gt;Revisiones&lt;/option&gt;</pre> <p>status 500 on a server error</p>

- 11. Get User

Uri name:	/users/:id
Method	GET
Description	Get the data of an existing user



Req	id
Res	<p>status 200 html with the user data in a Json</p> <pre>{   "user_id": "9",   "personal_id": "1600238257",   "name": "Marco",   "surname": "Diaz",   "profile_id": "2",   "profile_name": "Revisiones",   "email": "juan.perez@example.com",   "borndate": "Thu Jun 15 1995 00:00:00 GMT-0500 (hora de Ecuador)",   "user": "juanp",   "phone_number": "0999999999",   "state": "Activo" }</pre> <p>status 500 on a server error</p>

- 12. Create an user

Uri name:	/users
Method	POST
Description	creates an user in the database with data provided
Req	Json with all the data from a user
Res	<p>status 200 on user created</p> <pre>{   "message": "Usuario registrado exitosamente" }</pre> <p>status 100 on a server personal Id validation triggered</p> <pre>{   "error": "Ya existe un usuario con esa cédula" }</pre> <p>status 100 on a server username validation triggered</p> <pre>{   "error": "Ya existe un usuario con ese nombre" }</pre> <p>status 500 on a server error</p>

- 13. Update an user

Uri name:	/users/:id
Method	PUT
Description	Updates an user in the database with data provided
Req	Json with all the data from a user that changed
Res	<p>status 200 on user updated</p> <pre>{   "message": "Usuario actualizado correctamente" }</pre> <p>status 500 on a server error</p>

- 14. Logical delete username

Uri name:	/users/:id/state
Method	PATCH
Description	changes the value of the state of a certain user
Req	<p>id</p> <p>state</p>
Res	<p>status 200 on user changed</p> <pre>{   "message": "Estado del usuario actualizado correctamente" }</pre> <p>status 500 on a server error</p>

## Actions URIs

- 15. Get actions

Uri name:	/actions
Method	GET

Description	Get the rows of the table actions to obtain the data of all action made in the system and by who
Req	none
Res	<p>html with all the users created in the database</p> <pre> &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Visualizo proyecto Ambiental de AICE&lt;/td&gt;   &lt;td&gt;Fri Jun 27 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt;  &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Generación de un Proyecto&lt;/td&gt;   &lt;td&gt;Sat Jun 28 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt;  &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Generación de un usuario nuevo&lt;/td&gt;   &lt;td&gt;Sat Jun 28 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt; </pre> <p>status 500 on a server error</p>

- 16. Get actions of certain user by personal ID

Uri name:	/user/:personal_id
Method	GET
Description	Get the username that made a certain action
Req	user id
Res	<p>json with the user data and the action made</p> <pre> &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Generación de un usuario nuevo&lt;/td&gt;   &lt;td&gt;Sat Jun 28 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt;  &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Generación de un Proyecto&lt;/td&gt;   &lt;td&gt;Sat Jun 28 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt; </pre> <p>status 500 on a server error</p>

- 17. Get actions of certain date


Uri name:	/date
Method	GET
Description	Get the username that made a certain action
Req	date
Res	<p>json with the user data and the action made</p> <pre> &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Generación de un usuario nuevo&lt;/td&gt;   &lt;td&gt;Sat Jun 28 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt;  &lt;tr&gt;   &lt;td&gt;admin&lt;/td&gt;   &lt;td&gt;Juan Pérez&lt;/td&gt;   &lt;td&gt;Generación de un Proyecto&lt;/td&gt;   &lt;td&gt;Sat Jun 28 2025 00:00:00 GMT-0500 (hora de Ecuador)&lt;/td&gt; &lt;/tr&gt; </pre> <p>status 500 on a server error</p>

- 18. Post Action

Uri name:	/
Method	Post
Description	Post an action of a data, the date is the day the action was made
Req	user id
Res	<p>status 200 action saved</p> <pre> {   "message": "Acción Registrada" } </pre> <p>status 500 on a server error</p>

- 19. Get Report of a user in PDF

Uri name:	/pdf/:personal_id
-----------	-------------------

Method	Get
Description	Pets an downlandable pdf of the report of an users actions
Req	user id
Res	<p>status 200 action saved</p>  <p>status 500 on a server error</p>

## Reminders URIs

- 20. Get all reminders data

Uri name:	/reminders
Method	GET
Description	Get the json of reminders with data of all reminders made for a project and users
Req	date
Res	json with the reminders for a project in the month, just the names and day

	<pre>[   {     "id": 1,     "start": "2025-06-29T05:00:00.000Z",     "project": "HidroAbanico",     "register_date": "2025-04-29T05:00:00.000Z"   } ]</pre> <p>status 500 on a server error</p>
--	---

- 21. Get all reminders data of one project

Uri name:	/:projectId
Method	GET
Description	Get the json of reminders with data of all reminders made for a project
Req	date
Res	<p>json with the reminders for a project in the week, just the names and day</p> <pre>[   {     "id": 1,     "start": "2025-06-29T05:00:00.000Z",     "project": "HidroAbanico",     "register_date": "2025-04-29T05:00:00.000Z"   } ]</pre> <p>status 500 on a server error</p>

- 22. Get one reminder data

Uri name:	/data/:reminder_id
Method	GET
Description	Get the json of a specific reminder selected
Req	reminder_id

Res	<p>json with the reminders for a project in the week</p> <pre>{   "id": 1,   "title": "Auditoria Proyecto de EISA",   "description": "Se realizara una revisión del progreso de la segunda etapa del proyecto",   "start": "2025-06-29T05:00:00.000Z",   "project": "HidroAbanico",   "register_date": "2025-04-29T05:00:00.000Z" }</pre> <p>status 500 on a server error</p>
-----	---

- 23. Create a reminder

Uri name:	/
Method	POST
Description	Generate a reminder
Req	Json with reminder data
Res	<p>status 200 with ok response</p> <pre>{   "message": "Recordatorio creado exitosamente" }</pre> <p>status 500 on a server error</p>

- 24. Update a reminder

Uri name:	/:reminder_id
Method	PUT
Description	Generate a reminder
Req	Json with reminder data for update
Res	status 200 with ok response

	<pre>{   "message": "Recordatorio actualizado exitosamente" }</pre> <p>status 400 with reminder in an inactive state</p> <pre>Recordatorio no se puede actualizar porque no está activo</pre> <p>status 500 on a server error</p>
--	---

- 25. Logic delete of a reminder

Uri name:	/reminder/:id/state
Method	PATCH
Description	Changes the state of the reminder
Req	id of reminder
Res	<p>status 200 with ok response</p> <pre>{   "message": "Recordatorio ha sido modificado exitosamente" }</pre> <p>status 500 on a server error</p>

- 26. Get notifications of the day

Uri name:	/notifications
Method	Get
Description	Get all the reminders that finish this day to notify the user
Req	id of reminder
Res	status 200 with ok response



	<pre>[   {     "title": "Auditoria Proyecto de EISA",     "project": "HidroAbanico"   } ]</pre> <p>status 500 on a server error</p>
--	---

## Environmental Plan

- 27. Consult all the environmental plans of a project

Uri name:	http://localhost:3001/projects/:projectId/environmental-plans/
Method	GET
Description	This URI let the customer get the information all the environmental plans of a specific project, that is the reason we use IdProject as input parameter
Req	projectId
Res	<p>- status 200 (OK) the response will show a json file with the information:</p> <pre> 1  [ 2    { 3      "id": 3, 4      "projectid": 2, 5      "name": "Plan Ambiental C", 6      "description": "Descripción del plan C", 7      "stage": "Final", 8      "process": "Proceso 3", 9      "createdat": "2025-06-28T09:26:17.734Z", 10     "updatedat": "2025-06-28T09:26:17.734Z" 11   }, 12   { 13     "id": 4, 14     "projectid": 2, 15     "name": "Plan Ambiental D", 16     "description": "Descripción del plan D", 17     "stage": "Inicial", 18     "process": "Proceso 4", 19     "createdat": "2025-06-28T09:26:17.734Z", 20     "updatedat": "2025-06-28T09:26:17.734Z" 21   } 22 ] </pre> <p>- status 200(OK) If there is no data then it returns an empty array</p> <pre> 1  [] </pre>

	- status 500 on server error <pre> 1 2 {"message": "Cast to Number failed for value \"NaN\" (type number) at path \"projectId\" for model \"EnvironmentalPlan\""} </pre>
--	---

- 28. Consult one specific environmental plan of a project

Uri name:	http://localhost:3001/projects/:projectId/environmental-plans/:planId
Method	GET
Description	This URI let the customer get a specific plan of a specific project, and let show more information related
Req	planId
Res	- status 200 (OK) The response will show a json file with the information of one environmental plan <pre> 1 2 {"id": 4, 2 3   "projectId": 2, 3 4   "name": "Plan Ambiental D", 4 5   "description": "Descripción del plan D", 5 6   "stage": "Inicial", 6 7   "process": "Proceso 4", 7 8   "createdat": "2025-06-28T09:26:17.734Z", 8 9   "updatedat": "2025-06-28T09:26:17.734Z"} 9 10 </pre> - status 404 on not found <pre> 1 2 {"message": "The environmental plan was not found or not exists"} 2 3 </pre> - status 500 on server error <pre> 1 2 {"message": "Cast to Number failed for value \"NaN\" (type number) at path \"projectId\" for model \"EnvironmentalPlan\""} </pre>

- 29. Create environmental plan

Uri name:	http://localhost:3001/projects/:projectId/environmental-plans/
Method	POST
Description	The URI provides the service to create an environmental plan, using a body to send the data there is no necessary the “createdat, updatedat” values cause the URI will automatically create those two attributes.
Req	- projectId - JSON with the data of the new environmental plan

	<pre> 1  { 2    "name": "Plan Ambiental F", 3    "description": "Descripción del plan F", 4    "stage": "Intermedio", 5    "process": "Proceso 6" 6  }</pre>
Res	<p>- status 201 (created) The response will show a json with the new element created</p> <pre> 1  { 2    "createdat": "2025-06-28T19:53:20.861Z", 3    "updatedat": "2025-06-28T19:53:20.862Z", 4    "id": 8, 5    "projectid": 3, 6    "name": "Plan Ambiental F", 7    "description": "Descripción del plan F", 8    "stage": "Intermedio", 9    "process": "Proceso 6" 10 }</pre> <p>- status 400 on a format incorrect at the body</p> <pre> 1  { 2    "message": "Empty parameters are not allowed or the format is incorrect" 3  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "environmentalPlan.createdd is not a function" 3  }</pre>

### ● 30. Update environmental plan

Uri name:	http://localhost:3001/projects/:projectId/environmental-plans/:planId
Method	PUT
Description	The URI will need an id parameter sent by the body with all the new information, this will update the data from a specific environmental plan.
Req	<p>- planId - JSON with the new data for one specific environmental plan</p> <pre> 1  { 2    "name": "Plan Ambiental G", 3    "description": "Descripción del plan G", 4    "stage": "Construccion", 5    "process": "Proceso 6", 6    "updatedat": "2025-06-30T19:47:02.385Z" 7  }</pre>
Res	- status 200 with ok response to update

	<pre> 1  { 2    "id": 6, 3    "projectId": 3, 4    "name": "Plan Ambiental G", 5    "description": "Descripción del plan G", 6    "stage": "Construccion", 7    "process": "Proceso 6", 8    "createdat": "2025-06-28T19:43:29.270Z", 9    "updatedat": "2025-06-30T19:47:02.385Z" 10 }</pre> <p>- status 404 not found to update</p> <pre> 1  { 2    "message": "Not found the environmental plan or it not exist" 3  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asdas\"" 3  }</pre>
--	---

### ● 31. Delete environmental plan

Uri name:	http://localhost:3001/projects/:projectId/environmental-plans/:planId
Method	DELETE
Description	The URI provides the service to remove an environmental plan by the id of EP to delete that EP of a specific project.
Req	planId
Res	<p>- status 204 it means deleted successfully (not showing nothing)</p> <pre> 1</pre> <p>- status 404 not found</p> <pre> 1  { 2    "message": "Not found the environmental plan or it not exist" 3  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asdasd\"" 3  }</pre>

## Activity

### ● 32. Consult all the activities of a environmental plan

Uri name:	http://localhost:3001/environmental-plans/:planId/activities/
-----------	---

Method	GET
Description	This URI lets the customer get all the activities inside an environmental plan in the form of json to fill the rows of a table.
Req	planId
Res	<p>- The response will show a json file with the information</p> <pre> 1  [ 2    { 3      "activity_id": 1, 4      "environmentalplan_id": 1, 5      "activity_aspect": "Aspecto 1", 6      "activity_impact": "Impacto 1", 7      "activity_measure": "Medida 1", 8      "activity_verification": "Verificación 1", 9      "activity_frequency": "Frecuencia 1", 10     "createdat": "2025-06-28T21:40:08.000Z", 11     "updatedat": "2025-06-28T21:40:08.000Z" 12   }, 13   { 14     "activity_id": 2, 15     "environmentalplan_id": 1, 16     "activity_aspect": "Aspecto 2", 17     "activity_impact": "Impacto 2", 18     "activity_measure": "Medida 2", 19     "activity_verification": "Verificación 2", 20     "activity_frequency": "Frecuencia 2", 21     "createdat": "2025-06-28T21:40:08.000Z", 22     "updatedat": "2025-06-28T21:40:08.000Z" 23   } 24 ] </pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asda\"" 3  } </pre>

- 33. Consult one specific activity of a environmental plan

Uri name:	http://localhost:3001/environmental-plans/:planId/activities/:activityId
Method	GET
Description	This URI lets the customer get the information of a specific activity with the ID
Req	activityId
Res	- The response will show a json file with the information of one activity

	<pre> 1  { 2    "activity_id": 2, 3    "environmentalplan_id": 1, 4    "activity_aspect": "Aspecto 2", 5    "activity_impact": "Impacto 2", 6    "activity_measure": "Medida 2", 7    "activity_verification": "Verificación 2", 8    "activity_frecuency": "Frecuencia 2", 9    "createdat": "2025-06-28T21:40:08.000Z", 10   "updatedat": "2025-06-28T21:40:08.000Z" 11 }</pre> <p>- status 404 not found</p> <pre> 1  { 2    "message": "The activity was not found or does not exist" 3  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asd\"" 3  }</pre>
--	---

- 34. Create an activity

Uri name:	http://localhost:3001/environmental-plans/:planId/activities/
Method	POST
Description	This URI provides the service to create a new activity inside an environmental plan, the information will be sent by the body
Req	<p>- planId</p> <p>- JSON with the data of the new activity</p> <pre> 1  { 2    "aspect": "Construcción de tomas y tubería especializada", 3    "impact": "Degradación ambiental por desconocimiento social", 4    "measure": "Capacitación y señalización de zonas de trabajo", 5    "verification": "Registro de capacitación, fotografías", 6    "frecuency": "Permanente" 7  }</pre>
Res	- status 201 with ok response to create

	<pre> 1  { 2    "createdat": "2025-06-28T22:29:53.174Z", 3    "updatedat": "2025-06-28T22:29:53.174Z", 4    "activity_id": 51, 5    "environmentalplan_id": 1, 6    "activity_aspect": "Construcción de tomas y tubería especializada", 7    "activity_impact": "Degradación ambiental por desconocimiento social", 8    "activity_measure": "Capacitación y señalización de zonas de trabajo", 9    "activity_verification": "Registro de capacitación, fotografías", 10   "activity_frecuency": "Permanente" 11 } </pre> <p>- status 400 on incorrect format</p> <pre> 1  { 2    "message": "Empty parameters are not allowed or the format is incorrect" 3  } </pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "insert or update on table \"activity\" violates foreign key constraint \"fk_generates\"" 3  } </pre>
--	--

### ● 35. Update an activity

Uri name:	http://localhost:3001/environmental-plans/:planId/activities/:activityId
Method	PUT
Description	This URI provides the service to update the data of a specific activity with the id, the information will be sent in the body
Req	<p>- activityId</p> <p>- JSON with the new data of the activity</p> <pre> 1  { 2    "aspect": "Construcción de tomas y tubería especializada actualizado", 3    "impact": "Degradación ambiental por desconocimiento social", 4    "measure": "Capacitación y señalización de zonas de trabajo", 5    "verification": "Registro de capacitación, fotografías", 6    "frecuency": "Permanente" 7  } </pre>
Res	<p>- status 200 with ok response to update</p> <pre> 1  { 2    "activity_id": 51, 3    "environmentalplan_id": 1, 4    "activity_aspect": "Construcción de tomas y tubería especializada actualizado", 5    "activity_impact": "Degradación ambiental por desconocimiento social", 6    "activity_measure": "Capacitación y señalización de zonas de trabajo", 7    "activity_verification": "Registro de capacitación, fotografías", 8    "activity_frecuency": "Permanente", 9    "createdat": "2025-06-28T22:29:53.174Z", 10   "updatedat": "2025-06-28T22:29:53.174Z" 11 } </pre> <p>- status 404 not found</p>

	<pre> 1  ✓  "message": "Activity not found" 2 3 </pre> <p>- status 500 on server error</p> <pre> 1  ✓  "message": "invalid input syntax for type integer: \"asdasd\"" 2 3 </pre>
--	--

- 36. Delete an activity

Uri name:	http://localhost:3001/environmental-plans/:planId/activities/:activityId
Method	DELETE
Description	This URI provides the service to delete a specific activity
Req	activityId
Res	<p>- status 204 with no content but successfully deleted</p> <pre> 1 </pre> <p>- status 404 not found</p> <pre> 1  ✓  "message": "Activity not found" 2 3 </pre> <p>- status 500 on server error</p> <pre> 1  ✓  "message": "invalid input syntax for type integer: \"asdasd\"" 2 3 </pre>

## Control

- 37. Consult all the controls of an activity

Uri name:	http://localhost:3001/activities/:activityId/controls/
Method	GET
Description	This URI lets the customer get all the controls of the activity, to have traceability.
Req	activityId



Res	<p>- status 200 the response will show a json file with the information</p> <pre> 1  { 2    { 3      "control_id": 1, 4      "activity_id": 1, 5      "control_createdby": "Usuario 1", 6      "control_criterion": "Criterio 1", 7      "control_observation": "Observación 1", 8      "control_evidence": "evidencia_1.pdf", 9      "control_verification": "Verificación 1", 10     "createdat": "2025-06-28T21:40:08.000Z", 11     "updatedat": "2025-06-28T21:40:08.000Z" 12   }, 13   { 14     "control_id": 2, 15     "activity_id": 1, 16     "control_createdby": "Usuario 2", 17     "control_criterion": "Criterio 2", 18     "control_observation": "Observación 2", 19     "control_evidence": "evidencia_2.pdf", 20     "control_verification": "Verificación 2", </pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asd\"" 3  } </pre>
-----	--

- 38. Consult one specific control

Uri name:	http://localhost:3001/activities/:activityId/controls/:controlId
Method	GET
Description	This URI lets the customer get the information of a specific control with the ID
Req	<p>- activityId</p> <p>- controlId</p>
Res	<p>- status 200 the response will show a json file with the information of one control</p> <pre> 1  { 2    "control_id": 2, 3    "activity_id": 1, 4    "control_createdby": "Usuario 2", 5    "control_criterion": "Criterio 2", 6    "control_observation": "Observación 2", 7    "control_evidence": "evidencia_2.pdf", 8    "control_verification": "Verificación 2", 9    "createdat": "2025-06-28T21:40:08.000Z", 10   "updatedat": "2025-06-28T21:40:08.000Z" 11 } </pre> <p>- status 404 not found</p>

	<pre> 1  { 2    "message": "The control was not found or does not exist" 3  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asd\"" 3  }</pre>
--	--

### ● 39. Create control

Uri name:	http://localhost:3001/activities/:activityId/controls/
Method	POST
Description	This URI provides the service to create a control for that activity
Req	<p>- activityId - JSON with the data of the new control</p> <pre> 1  { 2    "createdby": "Mateo Medranda", 3    "criterion": "Cumple", 4    "observation": "El proceso cumple con todos los estándares", 5    "evidence": "/evidence/control05062025.pdf", 6    "verification": "Fotografias" 7  }</pre>
Res	<p>- status 201 with ok to create control</p> <pre> 1  { 2    "createdat": "2025-06-28T23:12:03.475Z", 3    "updatedat": "2025-06-28T23:12:03.475Z", 4    "control_id": 152, 5    "activity_id": 1, 6    "control_createdby": "Mateo Medranda", 7    "control_criterion": "Cumple", 8    "control_observation": "El proceso cumple con todos los estándares", 9    "control_evidence": "/evidence/control05062025.pdf", 10   "control_verification": "Fotografias" 11 }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "insert or update on table \"control\" violates foreign key constraint \"fk_contains\"" 3  }</pre>

### ● 40. Update control

Uri name:	http://localhost:3001/activities/:activityId/controls/:controlId
-----------	--

Method	PUT
Description	This URI provides the service update the control of an activity
Req	<p>- controlId</p> <p>- JSON with the new data of the control</p> <pre> 1  { 2    "createdby": "Mateo Medranda", 3    "criterion": "Cumple", 4    "observation": "EL proceso se mantiene sobre los límites, es necesario verificar", 5    "evidence": "/evidence/control0305062025.pdf", 6    "verification": "Informe" 7  }</pre>
Res	<p>- status 200 with ok response</p> <pre> 1  { 2    "control_id": 3, 3    "activity_id": 1, 4    "control_createdby": "Mateo Medranda", 5    "control_criterion": "Cumple", 6    "control_observation": "EL proceso se mantiene sobre los límites, es necesario verificar", 7    "control_evidence": "/evidence/control0305062025.pdf", 8    "control_verification": "Informe", 9    "createdat": "2025-06-28T21:40:08.000Z", 10   "updatedat": "2025-06-28T21:40:08.000Z" 11 }</pre> <p>- status 404 not found</p> <pre> 1  { 2    "message": "Control not found" 3  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asdasd\"" 3  }</pre>

#### ● 41. Delete Control

Uri name:	http://localhost:3001/activities/:activityId/controls/:controlId
Method	PATCH
Description	This URI provides the service to change the state of a control, this is made to have auditory and logical delete.
Req	<p>- controlId</p> <p>- JSON with the new verification of the control</p> <pre> 1  { 2    "verification": "Anulado" 3  }</pre>
Res	- status 200 with ok responde

	<pre> 1  { 2    "control_id": 2, 3    "activity_id": 1, 4    "control_createdby": "Usuario 2", 5    "control_criterion": "Criterio 2", 6    "control_observation": "Observación 2", 7    "control_evidence": "evidencia_2.pdf", 8    "control_verification": "Anulado", 9    "createdat": "2025-06-28T21:40:08.000Z", 10   "updatedat": "2025-06-28T21:40:08.000Z" 11 }</pre> <p>- status 404 not found</p> <pre> 1  { 2    "message": "Control not found" 3 }</pre> <p>status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"asdasd\"" 3 }</pre>
--	---

- 42. Obtain graphic of the control of activities for specific environmental plan at current month

Uri name:	http://localhost:3001/environmental-plans/:planId/compliance
Method	GET
Description	This URI provides the service to get the data to make a graph in the frontend, with the activities that were evaluated and the state of those
Req	planId
Res	<p>- status 200 with ok response</p> <pre> 1  { 2    "totalActivities": 5, 3    "activitiesEvaluated": 5, 4    "activitiesSatisfy": 1, 5    "percentageSatisfy": "20.00" 6 }</pre> <p>- status 500 on server error</p> <pre> {   "message": "Cast to Number failed for value \"asdasd\" (type string) at path \"id\" for model \"EnvironmentalPlan\"" }</pre>

- 43. Obtain the activities that are pending and need a control, based on the date of the last control registered

Uri name:	http://localhost:3001/environmental-plans/:planId/pending
Method	GET
Description	This URI provides the service to get the activities that are pending of a new control, first it will show the total activities for the environmental plan, then the number of pending activities based on the date of last control vs the frequency of the activity, and last the details of activities that are pending with the days since the last control.
Req	planId
Res	<p>- status 200 with ok response</p> <pre> 1  { 2    "totalActivities": 5, 3    "pendingActivities": 1, 4    "details": [ 5      { 6        "activity_id": 1, 7        "activity_measure": "Medida 1", 8        "activity_frequency": "Mensual", 9        "lastControlDate": "2025-05-28", 10       "daysSinceLastControl": 31 11      } 12    ] 13  }</pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "column \"nan\" does not exist" 3  }</pre>

- 44. Obtain all the controls for an environmental plan between two dates

Uri name:	http://localhost:3001/environmental-plans/:planId/reports/controls
Method	GET
Description	<p>This URI provides the service to get all the controls between two dates, showing the activity to which they belong, this is a filter that let the client have a report of the environmental plan based on the controls.</p> <p>Its necessary to send the URI with a query string like “?from=2025-01-01&amp;to=2025-06-01” because the URI uses keys of</p>

	the query string.
Req	- planId
Res	<p>- status 200 with ok response</p> <pre> 1  { 2    "total": 4, 3    "data": [ 4      { 5        "control_id": 152, 6        "activity_id": 1, 7        "control_createdby": "Mateo Medranda", 8        "control_criterion": "Cumple", 9        "control_observation": "El proceso cumple con todos los estándares", 10       "control_evidence": "/evidence/control05062025.pdf", 11       "control_verification": "Fotografias", 12       "createdat": "2025-05-28T23:12:03.475Z", 13       "updatedat": "2025-06-28T23:12:03.475Z", 14       "activity": { 15         "activity_id": 1, 16         "activity_aspect": "Aspecto 1", 17         "activity_impact": "Impacto 1", 18         "activity_measure": "Medida 1", 19         "activity_verification": "Verificación 1", 20         "activity_frecuency": "Mensual" </pre> <p>- status 500 on server error</p> <pre> 1  { 2    "message": "invalid input syntax for type integer: \"sadasd\"" 3  } </pre>

- 45. Obtain a report of an environmental plan in pdf format

Uri name:	http://localhost:3001/environmental-plans/:planId/report
Method	GET
Description	This URI provides the service to get a report in pdf format with one graphic about the activities evaluated, the activities that satisfy and non satisfy, and finally the activities that do not need a control. The report is created for the current month
Req	- planId
Res	- status 200 with ok response



## Reporte Plan De Manejo Ambiental

---

### Plan prevencion y mitigacion de impactos

Fecha de emisión: 3/7/2025



Categoría	Valor
Total actividades	2.0
Evaluadas	1.0
Cumplen	0.0
No cumplen	1.0
No aplica	1.0

#### Actividades Evaluadas

Actividad	Responsable	Evaluación	Observación
Mantener durante la construcción del proyecto personal de SSA responsable del seguimiento de medidas de gestión ambiental y SSO	Permanente	No Cumple	Existe un deficiente control sobre la mitigación de impacto

#### Actividades No Evaluadas

Actividad	Frecuencia	Criterio	Observación
Emitir un permiso de liberación ambiental de actividades y maquinaria previo al inicio de trabajos que involucren el cauce del Río Upano	Previo a la construcción	'N/A'	'N/A'

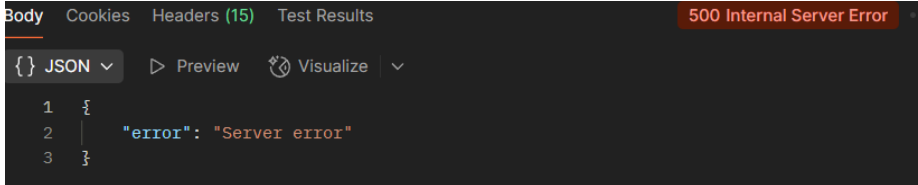
- status 500 on server error

```
1 ✓
2 "message": "invalid input syntax for type integer: \"sadasd\""
3
```

## Project URIs

- 46. Get all Projects

Uri name:	/projects
Method	GET
Description	Get the rows of the table projects to obtain the data of all projects

	created
Req	none
Res	<p>html with all the projects created in the database</p> <pre>[   {     "project_id": 1,     "project_name": "Renovación Centro Histórico",     "project_startdate": "2025-01-15",     "project_state": "Activo",     "project_location": "Bogotá",     "project_image": "/imagenes/centro_historico.jpg",     "project_description": "Renovación del centro histórico de la ciudad"   },   {     "project_id": 2,     "project_name": "Ampliación Autopista Norte",     "project_startdate": "2025-03-01",     "project_state": "En Planificación",     "project_location": "Medellín"   } ]</pre> <p>status 500 on a server error</p> 

- 47. Create a Project

Uri name:	/projects
Method	POST
Description	Creates a project
Req	<p>JSON with project data: name, startDate, image, state, location, description</p> <pre>{\"name\":\"Nuevo Proyecto\",\"startDate\":\"2025-07-01\",\"state\":\"Activo\",\"location\":\"Bucaramanga\",\"image\":\"/imagenes/nuevo.jpg\",\"description\":\"Nuevo desarrollo urbano\"}</pre>
Res	status 200 on project created

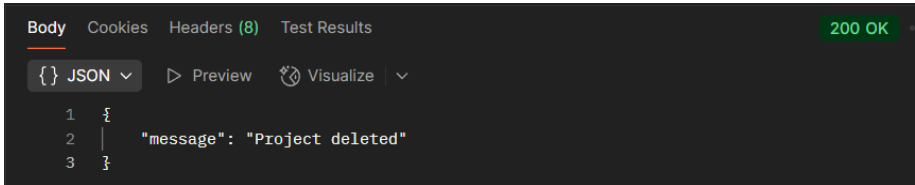
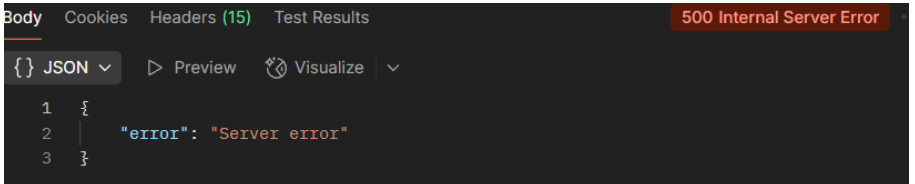


	<div> <div>Body Cookies Headers (15) Test Results 200 OK</div> <div> <div>{ } JSON Preview Visualize</div> <pre> 1  { 2    "project_id": 6, 3    "project_name": "Nuevo Proyecto", 4    "project_startdate": "2025-07-01", 5    "project_image": "/imagenes/nuevo.       jpg 6 7    "project_state": "Activo", 8    "project_location": "Bucaramanga", 9    "project_description": "Nuevo desarrollo urbano"     } </pre> </div> </div> <p>status 500 on a server error</p> <div> <div>Body Cookies Headers (15) Test Results 500 Internal Server Error</div> <div> <div>{ } JSON Preview Visualize</div> <pre> 1  { 2    "error": "Server error" 3  } </pre> </div> </div>
--	---

- 48. Update a Project

Uri name:	/projects/:id
Method	PUT
Description	Updates a project
Req	<p>JSON with project data to update</p> <pre>{   "name": "Renovación Centro Histórico Actualizada",   "startDate": "2025-01-20",   "state": "Activo",   "location": "Bogotá",   "image": "/imagenes/centro_actualizado.jpg",   "description": "Renovación actualizada del centro" }</pre>
Res	<p>status 200 on project updated</p> <div> <div>Body Cookies Headers (15) Test Results 200 OK</div> <div> <div>{ } JSON Preview Visualize</div> <pre> 1  { 2    "message": "Project updated" 3  } </pre> </div> </div> <p>status 500 on a server error</p> <div> <div>Body Cookies Headers (15) Test Results 500 Internal Server Error</div> <div> <div>{ } JSON Preview Visualize</div> <pre> 1  { 2    "error": "Server error" 3  } </pre> </div> </div>

- 49. Delete a Project

Uri name:	/projects/:id
Method	DELETE
Description	Deletes a project
Req	id
Res	<p>status 200 on project deleted</p>  <p>status 500 on a server error</p> 

- 50. Get all Monitorings from a specific project

Uri name:	/projects/:id/monitorings
Method	GET
Description	Gets all monitorings related to a specific project
Req	id of the project
Res	json with all monitorings related to the project

Body

Cookies

Headers (8)

Test Results

200 OK

{ } JSON

Preview

Visualize

```

1  [
2    {
3      "monitoring_id": 1,
4      "project_id": 1,
5      "monitoring_name": "Inspección
        Inicial
6
7      "monitoring_description": "Revisión inicial del sitio en el centro",
8      "monitoring_evidence": "/evidencias/inspeccion1.pdf",
9      "monitoring_observations": "Sin problemas detectados",
10     "monitoring_image": "/imagenes/inspeccion1.
        jpg
11
12     "monitoring_folder": "/carpetas/
        centro_historico

```

status 500 on a server error

Body

Cookies

Headers (15)

Test Results

500 Internal Server Error

{ } JSON

Preview

Visualize

```

1  {
2    "error": "Server error"
3  }

```

- 51. Search a Project by a name or date range

Uri name:	/projects/search
Method	GET
Description	Searches projects by name or date range
Req	query params: name, startDate
Res	json with matched projects

Body
Cookies
Headers (8)
Test Results
200 OK
50 ms
9.71 KB

{} JSON
Preview
Visualize

```

1  [
2    {
3      "project_id": 2,
4      "project_name": "Ampliación Autopista Norte",
5      "project_startdate": "2025-03-01",
6      "project_state": "En Planificación",
7      "project_location": "Medellín",
8      "project_image": "/imagenes/autopista.jpg",
9      "project_description": "Ampliación de la autopista principal"
10   },

```

status 500 on a server error

Body
Cookies
Headers (15)
Test Results
500 Internal Server Error

{} JSON
Preview
Visualize

```

1  {
2    "error": "Server error"
3  }

```

## Monitoring URIs

- 52. Get all Monitorings

Uri name:	/monitorings
Method	GET
Description	Get the rows of the table monitorings to obtain the data of all monitorings created
Req	none
Res	html with all the monitorings created in the database

Body

Cookies

Headers (8)

Test Results

200 OK

{ } JSON

Preview

Visualize

```

1  [
2    {
3      "monitoring_id": 1,
4      "project_id": 1,
5      "monitoring_name": "Inspección
        Inicial",
6      "monitoring_description": "Revisión inicial del sitio en el centro",
7      "monitoring_evidence": "/evidencias/inspeccion1.pdf",
8      "monitoring_observations": "Sin problemas detectados",
9      "monitoring_image": "/imagenes/inspeccion1.
        jpg",
10     "monitoring_folder": "/carpetas/
        centro_historico",
11   },

```

status 500 on a server error

Body

Cookies

Headers (15)

Test Results

500 Internal Server Error

{ } JSON

Preview

Visualize

```

1  {
2    "error": "Server error"
3  }

```

- 53. Create a Monitoring

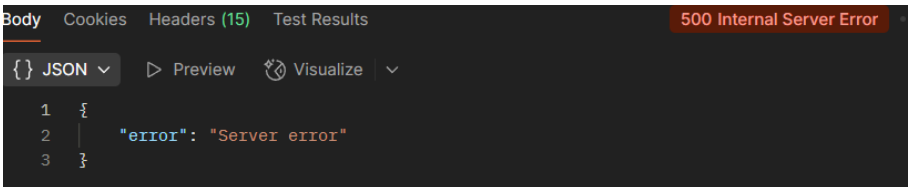
Uri name:	/monitorings
Method	POST
Description	Creates a monitoring
Req	JSON with monitoring data: idProject, name, description, observations, image, file, folder <pre> 1  {"idProject":1,"name":"Inspección Adicional","description":"Revisión adicional del sitio",     "observations":"Todo en orden","image":"/imagenes/adicional.jpg","file":"/evidencias/adicional.pdf",     "folder":"/carpetas/centro_historico"} </pre>
Res	status 200 on monitoring created

	<div> <div>Body Cookies Headers (8) Test Results 200 OK</div> <div> <div>{ } JSON</div> <div>Preview Visualize</div> </div> <div> <pre> 1 { 2   "monitoring_id": 11, 3   "project_id": 1, 4   "monitoring_name": "Inspección       Adicional", 5   "monitoring_description": "Revisión adicional del sitio", 6   "monitoring_observations": "Todo en orden", 7   "monitoring_image": "/imagenes/adicional.       jpg", 8   "monitoring_evidence": "/evidencias/adicional.pdf", 9   "monitoring_folder": "/carpetas/       centro_historico" </pre> </div> </div> <p>status 500 on a server error</p> <div> <div>Body Cookies Headers (15) Test Results 500 Internal Server Error</div> <div> <div>{ } JSON</div> <div>Preview Visualize</div> </div> <div> <pre> 1 { 2   "error": "Server error" 3 } </pre> </div> </div>
--	---

- 54. Update a Monitoring

Uri name:	/monitorings/:id
Method	PUT
Description	Updates a monitoring
Req	<p>JSON with monitoring data to update</p> <pre> {"name": "Inspección Inicial Actualizada", "description": "Revisión inicial actualizada", "observations": "Sin problemas", "image": "/imagenes/inspeccion1_actualizada.jpg", "file": "/evidencias/inspeccion1_actualizada.pdf", "folder": "/carpetas/centro_historico"} </pre>
Res	<p>status 200 on monitoring updated</p> <div> <div>Body Cookies Headers (8) Test Results 200 OK</div> <div> <div>{ } JSON</div> <div>Preview Visualize</div> </div> <div> <pre> 1 { 2   "message": "Monitoring updated" 3 } </pre> </div> </div> <p>status 500 on a server error</p> <div> <div>Body Cookies Headers (15) Test Results 500 Internal Server Error</div> <div> <div>{ } JSON</div> <div>Preview Visualize</div> </div> <div> <pre> 1 { 2   "error": "Server error" 3 } </pre> </div> </div>

- 55. Delete a Monitoring

Uri name:	/monitorings/:id
Method	DELETE
Description	Deletes a monitoring
Req	id
Res	<p>status 200 on monitoring deleted</p>  <p>status 500 on a server error</p> 

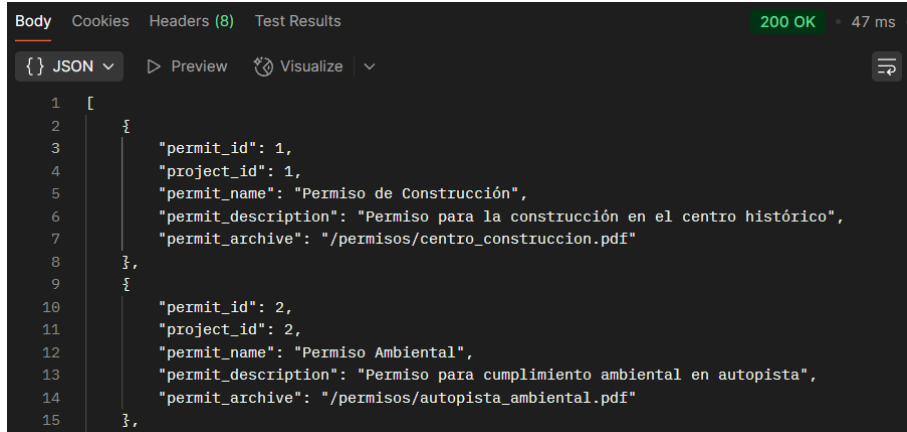
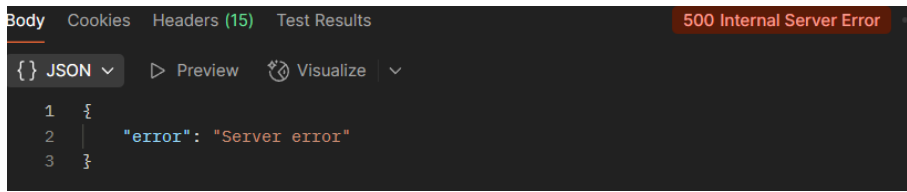
- 56. Get a file from a specific Monitoring

Uri name:	/monitorings/:id/file
Method	GET
Description	Gets the file (PDF/image) of a specific monitoring
Req	id of monitoring
Res	<p>pdf/image of monitoring file</p> <p>status 404 if not found</p> <p>status 500 on a server error</p>

## Permission URIs

- 57. Get all Permissions

Uri name:	/permits
-----------	----------

Method	GET
Description	Get the rows of the table permissions to obtain the data of all permissions created
Req	none
Res	<p>html with all the permissions created in the database</p>  <pre> 1  [ 2    { 3      "permit_id": 1, 4      "project_id": 1, 5      "permit_name": "Permiso de Construcción", 6      "permit_description": "Permiso para la construcción en el centro histórico", 7      "permit_archive": "/permisos/centro_construccion.pdf" 8    }, 9    { 10     "permit_id": 2, 11     "project_id": 2, 12     "permit_name": "Permiso Ambiental", 13     "permit_description": "Permiso para cumplimiento ambiental en autopista", 14     "permit_archive": "/permisos/autopista_ambiental.pdf" 15   } </pre> <p>status 500 on a server error</p>  <pre> 1  { 2    "error": "Server error" 3  } </pre>

- 58. Create a Permission

Uri name:	/permits
Method	POST
Description	Creates a permission
Req	<p>JSON with permission data: name, description, file, idProject</p> <pre> 1  {"name": "Permiso Adicional", "description": "Permiso adicional para obra", "file": "/permisos/adicional.pdf",    "idProject": 1} </pre>
Res	status 200 on permission created



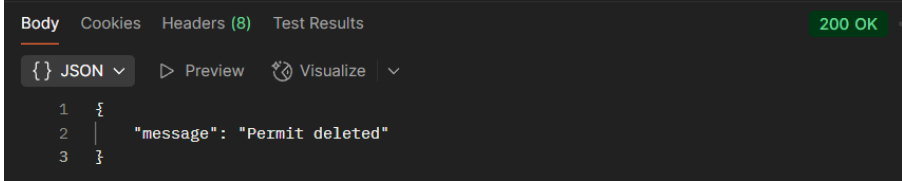
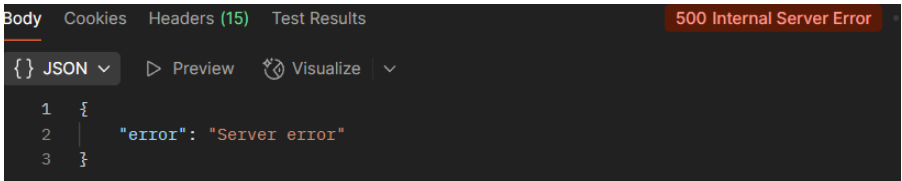
	<div> <div>Body Cookies Headers (8) Test Results 200 OK</div> <div>{ } JSON Preview Visualize</div> <pre> 1 { 2   "permit_id": 11, 3   "permit_name": "Permiso Adicional", 4   "permit_description": "Permiso adicional para obra", 5   "permit_archive": "/permisos/adicional.pdf", 6   "project_id": 1 7 }</pre> </div> <p>status 500 on a server error</p> <div> <div>Body Cookies Headers (15) Test Results 500 Internal Server Error</div> <div>{ } JSON Preview Visualize</div> <pre> 1 { 2   "error": "Server error" 3 }</pre> </div>
--	--

- 59. Update a Permission

Uri name:	/permits/:id
Method	PUT
Description	Updates a permission
Req	<p>JSON with permission data to update</p> <pre>{ "name": "Permiso de Construcción Actualizado", "description": "Permiso actualizado para construcción", "file": "/permisos/centro_actualizado.pdf" }</pre>
Res	<p>status 200 on permission updated</p> <div> <div>Body Cookies Headers (8) Test Results 200 OK</div> <div>{ } JSON Preview Visualize</div> <pre> 1 { 2   "message": "Permit updated" 3 }</pre> </div> <p>status 500 on a server error</p> <div> <div>Body Cookies Headers (15) Test Results 500 Internal Server Error</div> <div>{ } JSON Preview Visualize</div> <pre> 1 { 2   "error": "Server error" 3 }</pre> </div>

- 60. Delete a Permission

Uri name:	/permits/:id
Method	DELETE


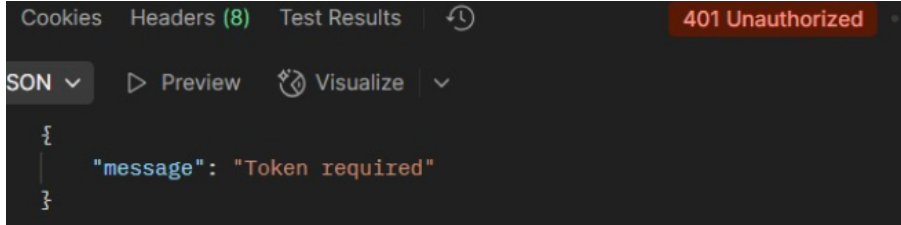
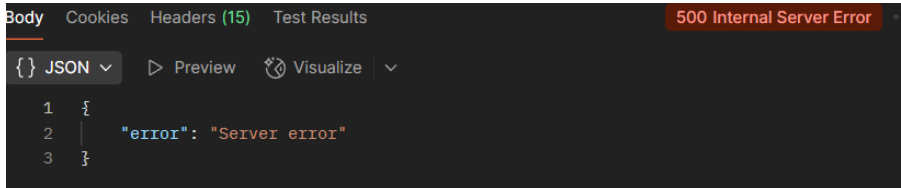
Description	Deletes a permission
Req	id
Res	<p>status 200 on permission deleted</p>  <p>status 500 on a server error</p> 

- 61. Get all Permissions of a specific project

Uri name:	/projects/:projectId/permits
Method	GET
Description	Gets all permissions of a specific project
Req	projectId
Res	<p>json with all permissions related to the project</p>  <p>status 500 on a server error</p>

- 62. Authenticate user and generate token

Uri name:	/api/user/login
-----------	-----------------

Method	POST
Description	Authenticates a user by username and password and returns a JWT token if successful.
Req	JSON body with the username and password
Res	<p>JWT token for authenticated requests and user info (id, username, name, profile).</p>  <pre> 1 { 2   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6OCwidm50bWU0IjkiKRMVybmkZko1LCJyZW11Ijo1SnVhbiIsInBye2ZpbGUiOiJhZG1pbm1zdHJhZG9yIGRlIFB5Ym51cyB8bWJpZW50YWx1cyIsIm1hdCI6MTQyNzY4NywiZmxwIjozNzUxNDMxMjg3fQ.1cBNCmNTSBxcu1VDGEzDnRbtch1-NL83PQRQUM-wwJE", 3   "user": { 4     "id": 8, 5     "username": "JFeinadez", 6     "name": "Juan", 7     "profile": "Administrador de Planes Ambientales" 8   } 9 } </pre> <p>401 Unauthorized</p>  <pre> {   "message": "Token required" } </pre> <p>status 500 on a server error</p>  <pre> 1 { 2   "error": "Server error" 3 } </pre>