



PROYECTO SPRING BOOT

**G E S T O R D E T A R E A S
(T A S K M A N A G E R)**

Realizado por Azeneth Bonilla Franco

Fecha: 03/11/2025
Semillero de Talento Hitss: Java

Tabla de contenido

1. Objetivo y Alcance del Proyecto.....	1
2. Arquitectura y Tecnologías Clave.....	1
3. Entidades Principales y Relaciones.....	2
4. Endpoints API REST (TaskController).....	2
5. Evidencias del proyecto	3
5.1 Base de Datos (MySQL)	3
5.2 Funcionalidad: Creación de Usuarios (Registro)	4
5.3 Funcionalidad: Autenticación (Login)	6
5. 4 Endpoints REST (TaskController).....	8
6.Conclusión del Proyecto.....	13

1. Objetivo y Alcance del Proyecto

Nombre del Proyecto: Gestor de Tareas (Task Manager)

Objetivo: Desarrollar una API RESTful completa utilizando **Spring Boot** para la gestión de tareas personales. El proyecto servirá como vehículo de aprendizaje para implementar los fundamentos del *framework* (Controladores, Servicios, Repositorios, Entidades, Configuración de Base de Datos) y la implementación de seguridad con **Spring Security + JWT**.

2. Arquitectura y Tecnologías Clave

- **Backend Framework:** Spring Boot (última versión estable).
- **Lenguaje de Programación:** Java.
- **Base de Datos:** MySQL.
- **Persistencia:** Spring Data JPA / Hibernate (CRUD con Spring Data JPA).
- **Seguridad:** Spring Security con JSON Web Tokens (JWT).
- **Documentación API:** SpringDoc / Swagger UI.
- **Herramienta de Construcción:** Maven o Gradle.
- **Estructura de Capas:**
 - **Controller:** Manejo de solicitudes HTTP y mapeo de endpoints (anotación @RestController).
 - **Service:** Lógica de negocio y uso de programación funcional/excepciones (anotación @Service).
 - **Repository:** Interacción con la base de datos (anotación @Repository).
 - **Model/Entity:** Clases que representan las tablas de la DB (anotación @Entity).

3. Entidades Principales y Relaciones

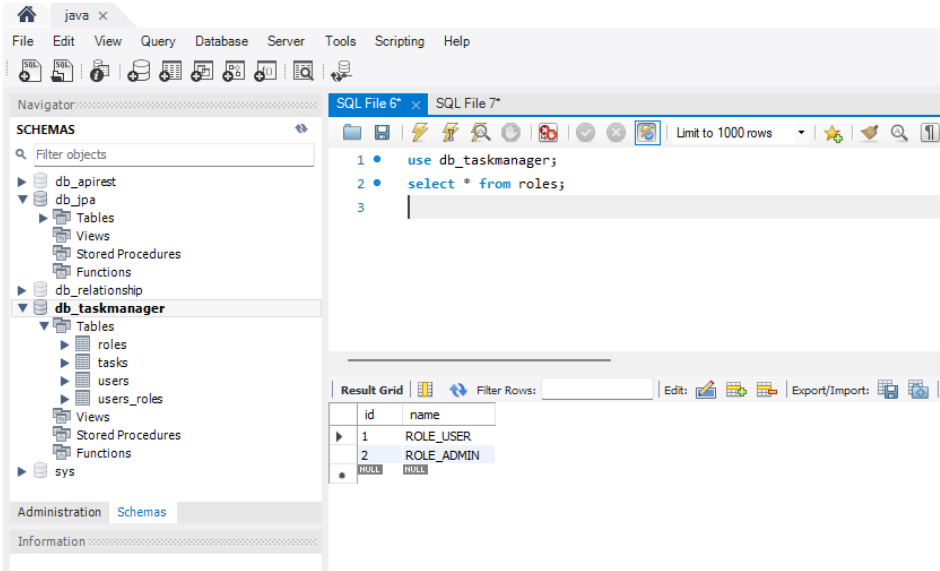
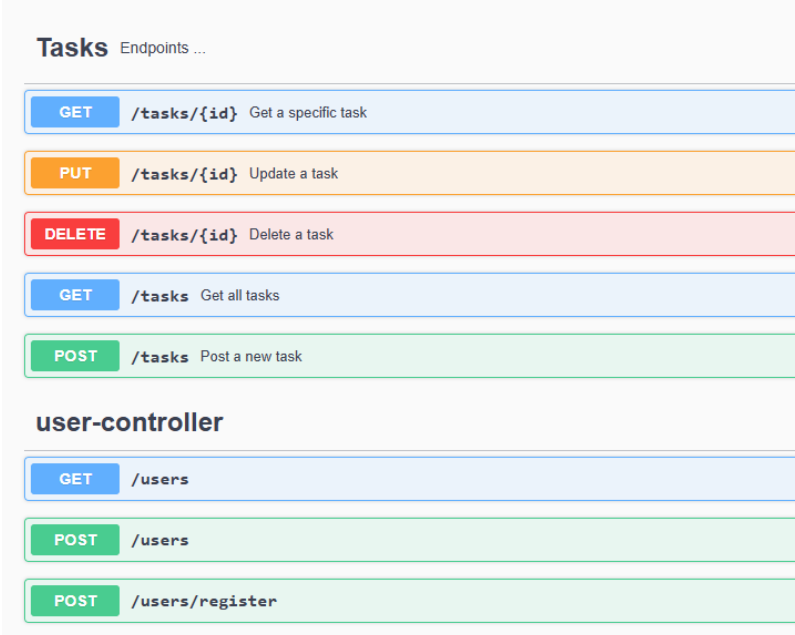
Entidad	Propósito	Anotación JPA	Relaciones
User	Almacena la información del usuario (username, password, roles).	@Entity	Uno a Muchos a Muchos con Role y Uno a Muchos con Task.
Role	Define los roles del usuario (e.g., USER, ADMIN).	@Entity	
Task	Representa una tarea (título, descripción, fecha de creación, estado).	@Entity	Muchos a Uno con User (la tarea pertenece a un usuario).

4. Endpoints API REST (TaskController)

MÉTODO	RUTA	DESCRIPCIÓN	Requisitos de Rol
GET	/tasks	Listar todas las tareas (o todas las del usuario autenticado).	USER, ADMIN
GET	/tasks/{id}	Obtener una tarea específica por ID.	USER (solo su tarea), ADMIN (cualquier tarea)
POST	/tasks	Crear una nueva tarea.	USER , ADMIN
PUT	/tasks/{id}	Actualizar una tarea existente.	ADMIN
DELETE	/tasks/{id}	Eliminar una tarea.	ADMIN
POST	/auth/login	<i>Endpoint</i> de autenticación (genera JWT).	Público
POST	/auth/register	<i>Endpoint</i> de registro (crea un nuevo USER).	Público

5. Evidencias del proyecto

5.1 Base de Datos (MySQL)

Evidencia	Detalle de Requisito Cumplido						
Requisito	Uso de base de datos MySQL y configuración de conexión.						
Petición	Ejecución de la <i>Query</i> para crear la base de datos e insertar los datos de los roles.						
Resultado Esperado	La base de datos (task_manager_db) y las tablas se visualizan correctamente en el gestor de MySQL.						
Evidencia	 <p>The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' pane displays a tree view of databases, with 'db_taskmanager' expanded to show its tables: roles, tasks, users, users_roles, and Views. The main editor shows an SQL query: <code>use db_taskmanager;</code> followed by <code>select * from roles;</code>. Below the query, the 'Result Grid' displays the results of the query:</p> <table border="1"> <thead> <tr> <th>id</th> <th>name</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>ROLE_USER</td> </tr> <tr> <td>2</td> <td>ROLE_ADMIN</td> </tr> </tbody> </table>	id	name	1	ROLE_USER	2	ROLE_ADMIN
id	name						
1	ROLE_USER						
2	ROLE_ADMIN						
Petición	Navegar a la URL de Swagger UI (http://localhost:8080/swagger-ui.html).						
Evidencia: Interfaz de Swagger cargada, mostrando los endpoints	 <p>The screenshot shows the Swagger UI interface. It lists two main sections of endpoints:</p> <p>Tasks Endpoints ...</p> <ul style="list-style-type: none"> GET <code>/tasks/{id}</code> Get a specific task PUT <code>/tasks/{id}</code> Update a task DELETE <code>/tasks/{id}</code> Delete a task GET <code>/tasks</code> Get all tasks POST <code>/tasks</code> Post a new task <p>user-controller</p> <ul style="list-style-type: none"> GET <code>/users</code> POST <code>/users</code> POST <code>/users/register</code> 						

5.2 Funcionalidad: Creación de Usuarios (Registro)

Evidencia	Detalle de Requisito Cumplido
Requisito	Crear al menos dos usuarios con roles: USER y ADMIN.
Petición	POST a /auth/register (o el endpoint de registro) enviando credenciales para un ROLE ADMIN.
Resultado Esperado	Usuario creado en la DB. Código 201 (Created).
Evidencia	<div><div><div>localhost:8080/users</div><div>POSTlocalhost:8080/users</div><div>ParamsAuthorizationHeaders (9)BodyScriptsSettings</div><div>noneform-datax-www-form-urlencodedorawbinaryGraphQL JSON</div><div><pre>1 { 2 "username": "Azeneth", 3 "email": "azenethbonilla@gmail.com", 4 "password": "admin123", 5 "enabled": true, 6 "admin": true 7 }</pre></div><div>BodyCookiesHeaders (11)Test Results201 Created564 ms58</div><div>JSONPreviewVisualize</div><div><pre>1 { 2 "id": 1, 3 "username": "Azeneth", 4 "email": "azenethbonilla@gmail.com", 5 "password": "\$2a\$10\$GVceXx.SB0kCCbwerF3eUmNkBDG8TfDM/2ZWdzZgb6Nm9h85jva", 6 "tasks": null, 7 "roles": [8 { 9 "id": 1, 10 "name": "ROLE_USER" 11 } 12], 13 "enabled": true, 14 "admin": true 15 }</pre></div></div><div><div>localhost:8080/users/register</div><div>POSTlocalhost:8080/users/register</div><div>ParamsAuthorizationHeaders (9)BodyScriptsSettings</div><div>noneform-datax-www-form-urlencodedorawbinaryGraphQL JSON</div><div><pre>1 { 2 "username": "Juliana", 3 "email": "juliana@gmail.com", 4 "password": "admin123", 5 "enabled": true 6 }</pre></div><div>BodyCookiesHeaders (11)Test Results201 Created417 ms550 B</div><div>JSONPreviewVisualize</div><div><pre>2 { 3 "id": 2, 4 "username": "Juliana", 5 "email": "juliana@gmail.com", 6 "password": "\$2a\$10\$DP9z6j5G1KmEVupLwgGLM03a@UbyBKTNBxRGVfGtmHDSvyAKRu2cm", 7 "tasks": null, 8 "roles": [9 { 10 "id": 1, 11 "name": "ROLE_USER" 12 } 13], 14 "enabled": true, 15 "admin": false 16 }</pre></div></div></div>

Evidencia

Consulta Query, para verificar la existencia de los usuarios creados.

The screenshot shows a SQL query editor with two tabs: 'SQL File 6*' and 'SQL File 7*'. The active tab 'SQL File 7*' contains the following SQL code:

```
1 • use db_taskmanager;  
2 • select * from users;  
3
```

Below the query editor is the 'Result Grid' showing the results of the query. The grid has columns: enabled, id, username, email, and password. The results are as follows:

enabled	id	username	email	password
1	1	Azeneth	azeneth@gmail.com	\$2a\$10\$ROWdCqRGgJr3QNe0j.od.S0zE2fpDB...
1	2	Juliana	juliana@gmail.com	\$2a\$10\$DP9rGj5GKmfVUpLwgGLMO3a0UbyBKT...
NULL	NULL	NULL	NULL	NULL

Tabla user_roles, la cual muestra la relación entre la tabla users y roles.

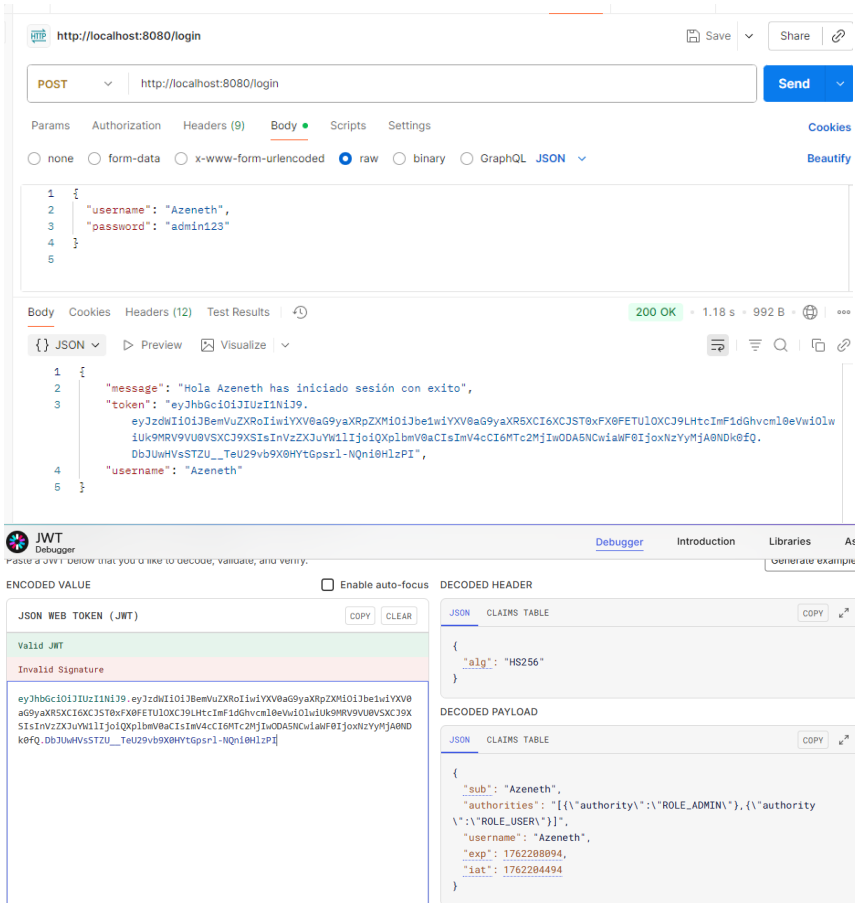
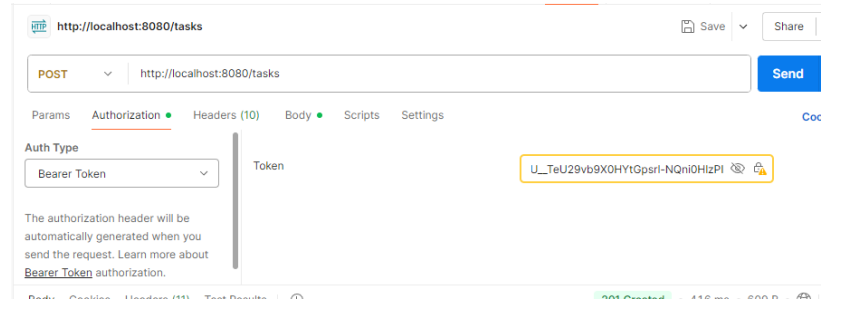
The screenshot shows a SQL query editor with two tabs: 'SQL File 6*' and 'SQL File 7*'. The active tab 'SQL File 6*' contains the following SQL code:

```
1 • use db_taskmanager;  
2 • select * from users_roles;  
3
```

Below the query editor is the 'Result Grid' showing the results of the query. The grid has columns: role_id and user_id. The results are as follows:

role_id	user_id
1	1
1	2
2	1
NULL	NULL

5.3 Funcionalidad: Autenticación (Login)

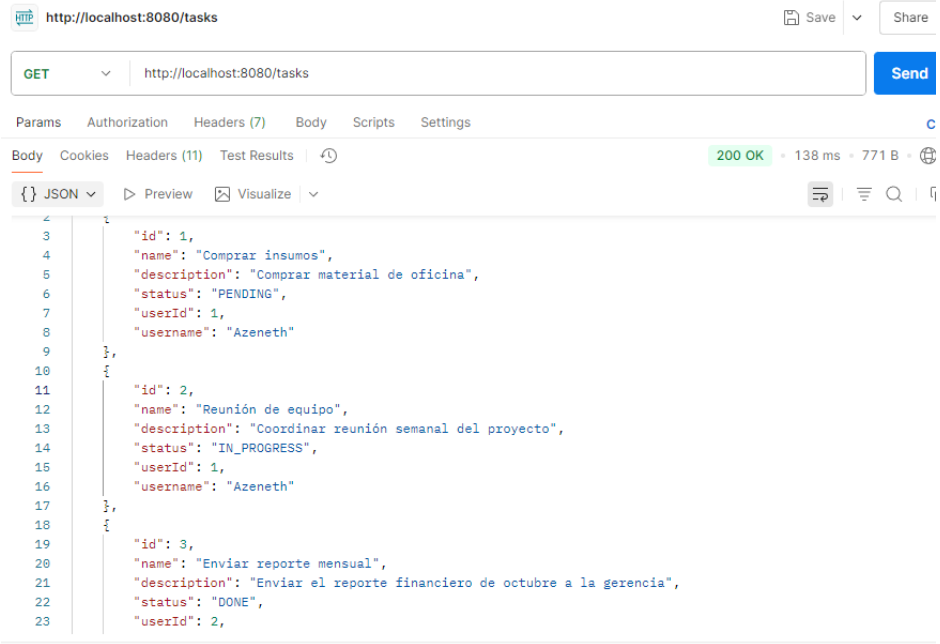
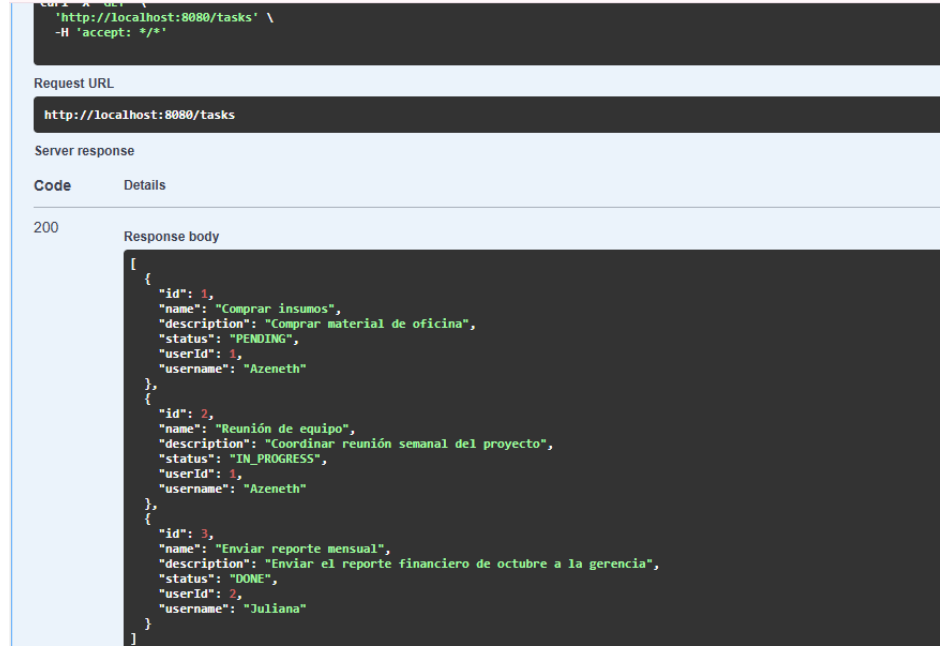
Campo de Evidencia	Detalle de Requisito Cumplido
Requisito	Generar un JWT para un usuario válido .
Petición (Postman)	POST a /auth/login con credenciales de un usuario.
Resultado Esperado	Respuesta exitosa con el Token JWT en el cuerpo o <i>Header</i> . Código 200 (OK).
Evidencia	
Requisito	Login con Usuario “ADMIN”
Petición (Postman)	POST a http://localhost:8080/tasks
Resultado Esperado	Se ejecuta correctamente la petición. Código 200 (OK).
Evidencia	

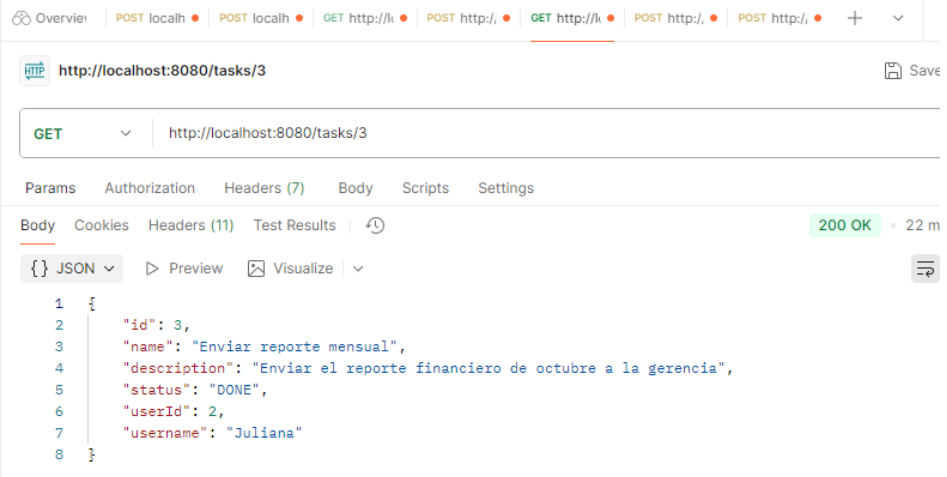
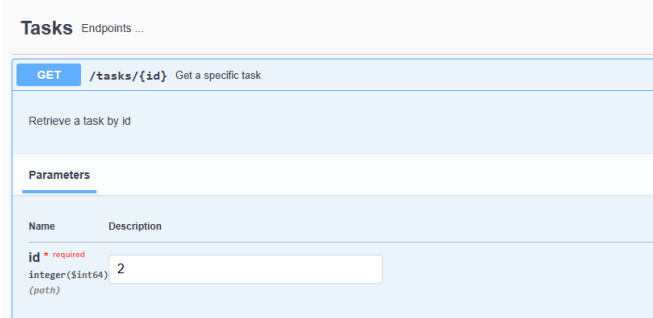
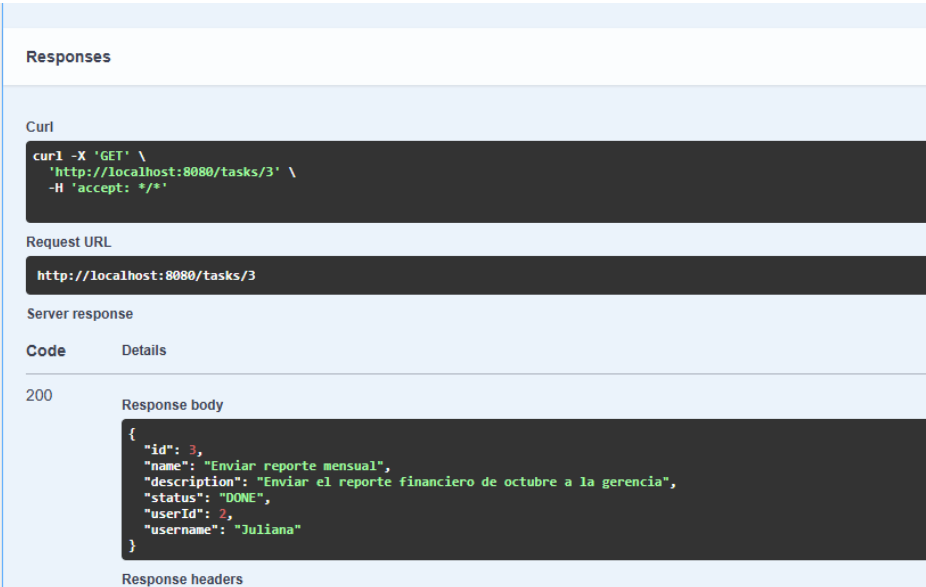
	
Requisito	Login con Usuario “USER”
Petición (Postman)	Como solo puede crear tareas y visualizarlas. Intentaremos hacer un DELETE a la tarea creada anteriormente.
Resultado Esperado	No se ejecutará la petición por que no es usuario ADMIN. Código 403(Forbidden)
Evidencia	<div>  </div>

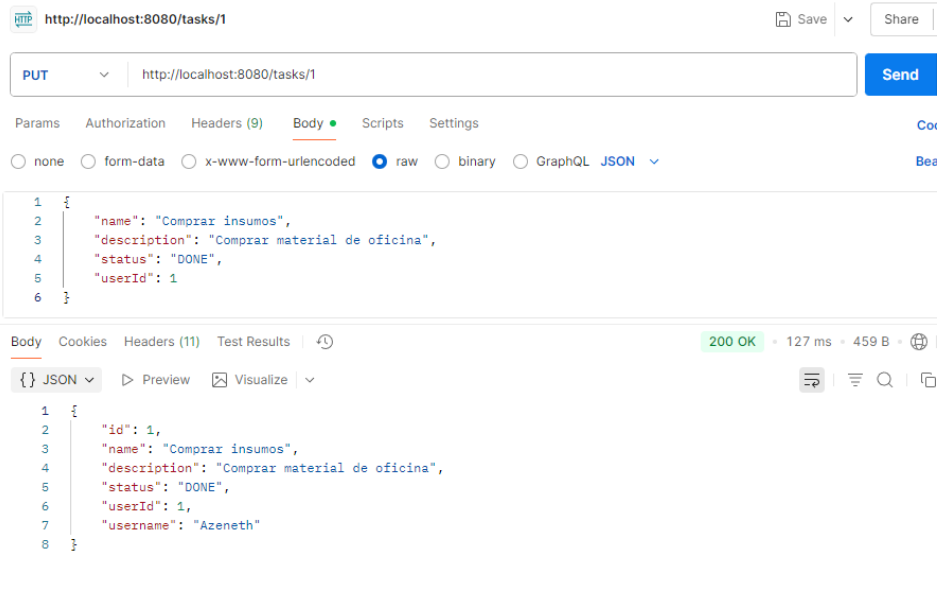
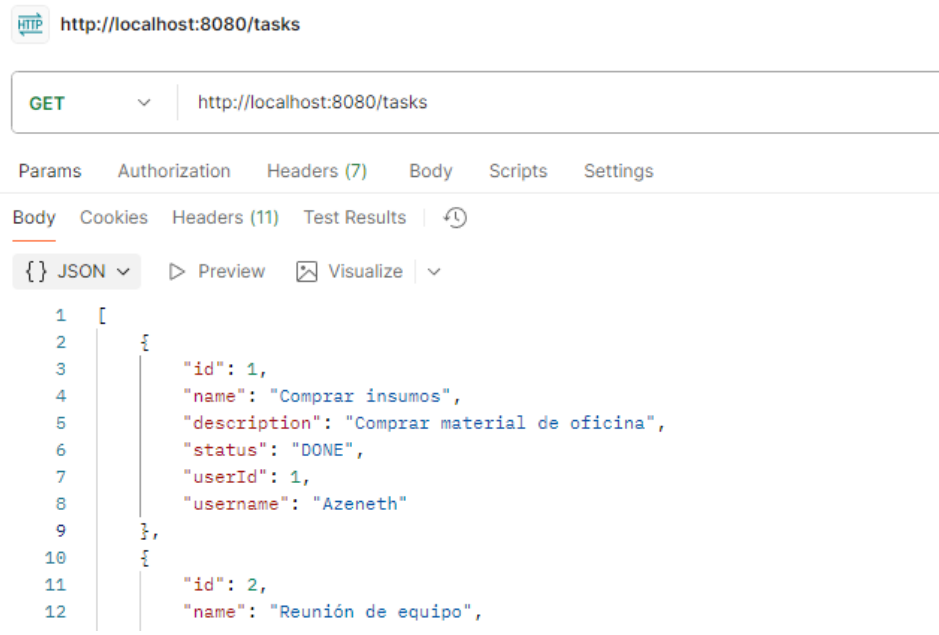
Y cuando usamos el usuario Admin, ya es posible eliminar.
Code(200 OK)

5. 4 Endpoints REST (TaskController)

Campo de Evidencia	Detalle de Requisito Cumplido
Requisito	CREAR UNA NUEVA TAREA
Petición	POST a /tasks. Usar el JWT del user en el <i>Header</i> Authorization: Bearer <token>.
Resultado Esperado	Tarea creada con éxito. Respuesta del objeto creado. Código 201 (Created).
Evidencia	<div><div><div>http://localhost:8080/tasks</div><div>POST http://localhost:8080/tasks</div><div>Params Authorization Headers (9) Body Scripts Settings</div><div>none form-data x-www-form-urlencoded raw binary GraphQL JSON</div><div><pre>1 { 2 "name": "Comprar insumos", 3 "description": "Comprar material de oficina", 4 "status": "PENDING", 5 "user": { 6 "id": 1 7 } 8 } 9</pre></div><div>Body Cookies Headers (11) Test Results 201 Created 35 ms 552 B</div><div>JSON Preview Visualize</div><div><pre>1 { 2 "id": 1, 3 "name": "Comprar insumos", 4 "description": "Comprar material de oficina", 5 "status": "PENDING", 6 "user": { 7 "id": 1, 8 "username": null, 9 "password": null 10 } 11 }</pre></div><div>POST http://localhost:8080/tasks</div><div>Params Authorization Headers (9) Body Scripts Settings</div><div>none form-data x-www-form-urlencoded raw binary GraphQL JSON</div><div><pre>1 { 2 "name": "Reunión de equipo", 3 "description": "Coordinar reunión semanal del proyecto", 4 "status": "IN_PROGRESS", 5 "user": { 6 "id": 1 7 } 8 } 9</pre></div><div>Body Cookies Headers (11) Test Results 201 Created 37 ms 571 B</div><div>POST http://localhost:8080/tasks</div><div>Params Authorization Headers (9) Body Scripts Settings</div><div>none form-data x-www-form-urlencoded raw binary GraphQL JSON</div><div><pre>1 { 2 "name": "Enviar reporte mensual", 3 "description": "Enviar el reporte financiero de octubre a la gerencia", 4 "status": "DONE", 5 "user": { 6 "id": 2 7 } 8 } 9</pre></div><div>Body Cookies Headers (11) Test Results 201 Created 36 ms 582 B</div></div></div>

REQUISITO	LISTAR LAS TAREAS EXISTENTES
Petición	GET /tasks Listar todas las tareas
Resultado Esperado	Consulta con éxito. Respuesta de la petición. Código 201 .
Evidencia	 <pre> { "id": 1, "name": "Comprar insumos", "description": "Comprar material de oficina", "status": "PENDING", "userId": 1, "username": "Azeneth" }, { "id": 2, "name": "Reunión de equipo", "description": "Coordinar reunión semanal del proyecto", "status": "IN_PROGRESS", "userId": 1, "username": "Azeneth" }, { "id": 3, "name": "Enviar reporte mensual", "description": "Enviar el reporte financiero de octubre a la gerencia", "status": "DONE", "userId": 2, "username": "Juliana" } </pre>  <pre> curl -X GET 'http://localhost:8080/tasks' \ -H 'accept: */*' Request URL http://localhost:8080/tasks Server response Code Details 200 Response body [{ "id": 1, "name": "Comprar insumos", "description": "Comprar material de oficina", "status": "PENDING", "userId": 1, "username": "Azeneth" }, { "id": 2, "name": "Reunión de equipo", "description": "Coordinar reunión semanal del proyecto", "status": "IN_PROGRESS", "userId": 1, "username": "Azeneth" }, { "id": 3, "name": "Enviar reporte mensual", "description": "Enviar el reporte financiero de octubre a la gerencia", "status": "DONE", "userId": 2, "username": "Juliana" }] </pre>

REQUISITO	OBTENER UNA TAREA ESPECIFICA.
Petición	GET a /tasks/{Id}
Resultado Esperado	Respuesta exitosa con el objeto JSON de la tarea solicitada. Código 200 (OK).
Evidencia	 <p>Usando Swagger, también pide el id, de forma más visual.</p>  

REQUISITO	ACTUALIZAR UNA TAREA
Petición (Postman)	PUT a /tasks/{ID_EXISTENTE} con el JSON de actualización. Usar el JWT del ADMIN .
Resultado Esperado	Tarea modificada con éxito. El JSON de respuesta refleja los cambios. Código 200 (OK) .
Evidencia (Captura de Postman)	 <p>The screenshot shows a Postman interface for a PUT request to <code>http://localhost:8080/tasks/1</code>. The request body is a JSON object: <code>{ "name": "Comprar insumos", "description": "Comprar material de oficina", "status": "DONE", "userId": 1 }</code>. The response is a 200 OK status with a response time of 127 ms and a body size of 459 B. The response body is a JSON object: <code>{ "id": 1, "name": "Comprar insumos", "description": "Comprar material de oficina", "status": "DONE", "userId": 1, "username": "Azeneth" }</code>.</p>  <p>The screenshot shows a Postman interface for a GET request to <code>http://localhost:8080/tasks</code>. The response is a 200 OK status with a response time of 127 ms and a body size of 459 B. The response body is a JSON array of two task objects: <code>[{ "id": 1, "name": "Comprar insumos", "description": "Comprar material de oficina", "status": "DONE", "userId": 1, "username": "Azeneth" }, { "id": 2, "name": "Reunión de equipo", "description": "Reunión de equipo", "status": "PENDING", "userId": 1, "username": "Azeneth" }]</code>.</p>

REQUISITO	ELIMINAR UNA TAREA EXISTENTE.										
Petición (Postman)	DELETE a /tasks/{ID_A_ELIMINAR}. Usar el JWT del ADMIN .										
Resultado Esperado	Tarea eliminada con éxito. Código 204 (No Content) o 200 (OK). (El 204 es el más RESTful para eliminaciones).										
Evidencia (Captura de Postman)	<div><div><div><div><div><div>http://localhost:8080/tasks/3</div><div>Save</div><div>Share</div></div><div><div>DELETE</div><div>http://localhost:8080/tasks/3</div><div>Send</div></div><div><div>Params</div><div>Authorization</div><div>Headers (7)</div><div>Body</div><div>Scripts</div><div>Settings</div></div><div>Query Params</div><table><thead><tr><th>Key</th><th>Value</th><th>Description</th><th></th><th>Bulk I</th></tr></thead><tbody><tr><td>Key</td><td>Value</td><td>Description</td><td></td><td></td></tr></tbody></table><div><div>Body</div><div>Cookies</div><div>Headers (11)</div><div>Test Results</div><div></div></div><div><div>200 OK</div><div>105 ms</div><div>492 B</div><div></div></div><div><div>{}</div><div>JSON</div><div></div><div>Preview</div><div>Visualize</div><div></div></div><div><div>1</div><div>{</div><div>2</div><div> "id": 3,</div><div>3</div><div> "name": "Enviar reporte mensual",</div><div>4</div><div> "description": "Enviar el reporte financiero de octubre a la gerencia",</div><div>5</div><div> "status": "DONE",</div><div>6</div><div> "userId": 2,</div><div>7</div><div> "username": "Juliana"</div><div>8</div><div>}</div></div></div></div></div><div><div><div>SQL File 6*</div><div>SQL File 7*</div></div><div><div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></div><div></</div></div></div></div></div>	Key	Value	Description		Bulk I	Key	Value	Description		
Key	Value	Description		Bulk I							
Key	Value	Description									

6.Conclusión del Proyecto

El desarrollo del proyecto **Gestor de Tareas (Task Manager)** ha logrado exitosamente el objetivo de construir una **API RESTful funcional** y robusta sobre **Spring Boot**.

Se ha demostrado un dominio sólido de los fundamentos del *framework*, abarcando la arquitectura de capas (Controller, Service, Repository) y la persistencia de datos mediante **Spring Data JPA** con una base de datos **MySQL**.

1. **CRUD Completo y Funcional:** Se implementaron los cinco *endpoints* REST (Crear, Listar, Obtener, Actualizar y Eliminar Tareas), verificando su correcto funcionamiento con **Postman**.
2. **Seguridad y Roles Implementados:** La integración de **Spring Security** y **JWT** fue crítica para establecer un modelo de autorización basado en roles, cumpliendo con la regla de negocio: el rol **USER** está restringido a la creación y visualización, mientras que el rol **ADMIN** mantiene control total (CRUD completo).
3. **Buenas Prácticas:** Se incorporaron prácticas de código limpio mediante el uso de anotaciones específicas (`@RestController`, `@Service`, `@Entity`), manejo profesional de errores con excepciones personalizadas y la aplicación de programación funcional (*Streams* y *Optional*) en la capa de lógica de negocio.
4. **Documentación:** La API está debidamente documentada e interactiva gracias a **Swagger UI**, facilitando su uso y futura expansión.