

App de Tareas - MVP Básico

Descripción

Este es un MVP básico de una aplicación de gestión de tareas. Incluye:

- Un backend en Spring Boot con una API REST.
- Una base de datos H2 en memoria.
- Una página estática de "En construcción".

Instrucciones

1. Clona el Repositorio

Clona el repositorio en tu máquina local:

```
git clone https://github.com/tu-usuario/taskmanager.git
cd taskmanager
```

2. Ejecuta la Aplicación Spring Boot

Ejecuta la aplicación usando Maven:

```
./mvnw spring-boot:run
```

- La aplicación estará disponible en <http://localhost:8080>.

3. Prueba la API con Postman o curl

PROF

La API expone los siguientes endpoints:

Listar todas las tareas

- **Método:** GET
- **URL:** <http://localhost:8080/tasks>
- **Ejemplo con curl:**

```
curl http://localhost:8080/tasks
```

- **Respuesta esperada:**

```
[]
```

Crear una nueva tarea

- **Método:** POST
- **URL:** <http://localhost:8080/tasks>
- **Cuerpo de la solicitud** (JSON):

```
{
  "title": "Comprar leche",
  "completed": false
}
```

- **Ejemplo con curl:**

```
curl -X POST -H "Content-Type: application/json" -d '{"title":
"Comprar leche", "completed": false}' http://localhost:8080/tasks
```

- **Respuesta esperada:**

```
{
  "id": 1,
  "title": "Comprar leche",
  "completed": false
}
```

Listar tareas después de crear una

- **Método:** GET
- **URL:** <http://localhost:8080/tasks>
- **Ejemplo con curl:**

```
curl http://localhost:8080/tasks
```

- **Respuesta esperada:**

```
[
  {
    "id": 1,
    "title": "Comprar leche",
    "completed": false
  }
]
```

```
}  
]
```

4. Accede a la Consola H2

Puedes ver la base de datos en tiempo real usando la consola H2:

1. Abre tu navegador y ve a <http://localhost:8080/h2-console>.
2. Conéctate con:
 - **JDBC URL:** `jdbc:h2:mem:taskdb`
 - **User:** `sa`
 - **Password:** `password`
3. Ejecuta la siguiente consulta SQL para ver las tareas:

```
SELECT * FROM TASK;
```

5. Abre la Página Estática

La página estática de "En construcción" está disponible en:

- **URL:** <http://localhost:8080>
- **Contenido:**

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>App de Tareas - En Construcción</title>  
</head>  
<body>  
  <h1>¡Estamos trabajando en algo increíble!</h1>  
  <p>La aplicación estará disponible pronto. Mientras tanto,  
  puedes probar nuestra API REST.</p>  
</body>  
</html>
```

El frontend de la aplicación también está desplegado en GitHub Pages:

- **URL:** <https://paulmclennon.github.io/taskmanager/>

Documentación

- El anteproyecto del proyecto está disponible en [Releases](#).

Estructura del Proyecto

```
taskmanager/
├── src/
│   ├── main/
│   │   ├── java/
│   │   │   ├── com/
│   │   │   │   ├── example/
│   │   │   │   │   ├── taskmanager/
│   │   │   │   │   │   ├── TaskmanagerApplication.java
│   │   │   │   │   │   ├── Task.java
│   │   │   │   │   │   ├── TaskRepository.java
│   │   │   │   │   │   └── TaskController.java
│   │   │   └── resources/
│   │   │       ├── application.properties
│   │   │       └── static/
│   │   │           └── index.html
│   └── test/
├── docs/
│   ├── index.html
│   └── Anteproyecto.pdf
└── pom.xml
```

<-- Carpeta para GitHub Pages
<-- Página estática del frontend
<-- Archivo PDF del anteproyecto

Tecnologías Usadas

- **Backend:** Spring Boot, H2 Database.
- **Frontend:** Página estática HTML.
- **Herramientas:** Maven, Postman, curl.

Próximos Pasos

- Implementar un frontend interactivo.
- Migrar a una base de datos remota (MySQL, PostgreSQL).
- Desplegar la aplicación en un entorno de producción (Kubernetes, Docker).