

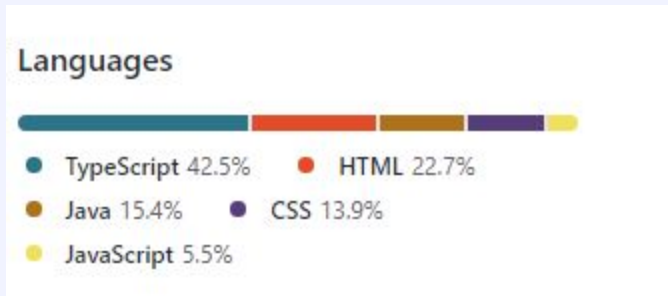
APLICACIÓN GESTIÓN DE TAREAS PARA API UX

Construida por [Erick Ocando Sánchez](#) para las prueba de Desarrollador FullStack

Funcionalidades:

- ❖ Listar tareas
- ❖ Agregar Nueva Tarea
- ❖ Editar tarea
- ❖ Eliminar tarea
- ❖ Dar por completada una tarea
- ❖ Reabrir tarea completada

Tecnologías y características usadas:



- ❖ REST API
- ❖ Backend
 - Java 11
 - Spring Boot 2.2
 - Spring Data JPA
 - PostgreSQL
 - Maven
- ❖ Frontend
 - Angular 11
 - Angular CLI
 - Angular material
 - Typescript
- ❖ Validación de entrada

Ejecutar Aplicación

Backend

1. Instalar PostgreSQL
2. Iniciar el servicio de postgres
3. Ingresar a la consola de postgres
4. Correr el script data.sql dentro de la carpeta backend
 - >> postgres-# \i 'backend/data.sql'

This PC > Local Disk (C:) > Users > eocan > projects > prueba-apiux > backend				
	Name	Date modified	Type	Size
	.idea	12/24/2020 12:15 PM	File folder	
	.mvn	12/23/2020 4:45 PM	File folder	
	src	12/23/2020 4:45 PM	File folder	
	target	12/24/2020 10:38 AM	File folder	
	.gitignore	12/23/2020 4:44 PM	Text Document	1 KB
	data.sql	12/23/2020 6:35 PM	SQL File	1 KB
	HELP.md	12/23/2020 4:44 PM	MD File	2 KB
	mvnw	12/23/2020 4:44 PM	File	11 KB
	mvnw	12/23/2020 4:44 PM	Windows Comma...	7 KB
	pom	12/24/2020 10:35 AM	XML Document	3 KB
	TASK LIST.postman_collection.json	12/23/2020 6:35 PM	JSON File	3 KB
	task-lists.iml	12/21/2020 7:24 PM	IML File	12 KB

data.sql:

```
CREATE DATABASE tasklistdb;

\c tasklistdb;

CREATE TABLE task (
    task_id serial PRIMARY KEY,
    task_name VARCHAR ( 50 ) NOT NULL,
    task_description VARCHAR ( 50 ),
    active BOOLEAN NOT NULL,
    create_time TIMESTAMP NOT NULL,
    update_time TIMESTAMP NOT NULL
);

INSERT INTO "public"."task" ("task_id", "task_name", "task_description", "active
INSERT INTO "public"."task" ("task_id", "task_name", "task_description", "active
INSERT INTO "public"."task" ("task_id", "task_name", "task_description", "active
```

El cual crea la base de datos **tasklistdb**, la tabla **task** y le inserta 3 registros

Luego

- 5. ingresar a la carpeta del backend.
- 6. mvn install
- 7. mvn spring-boot:run

El backend debe quedar arriba en localhost:8080.

La manera alternativa de correr el backend luego de los pasos 1 a 4, hacer:

- cd backend
- java -jar deploy/task-lists-api-0.0.1-SNAPSHOT.jar

Igualmente el backend debe quedar arriba en localhost:8080.

Frontend

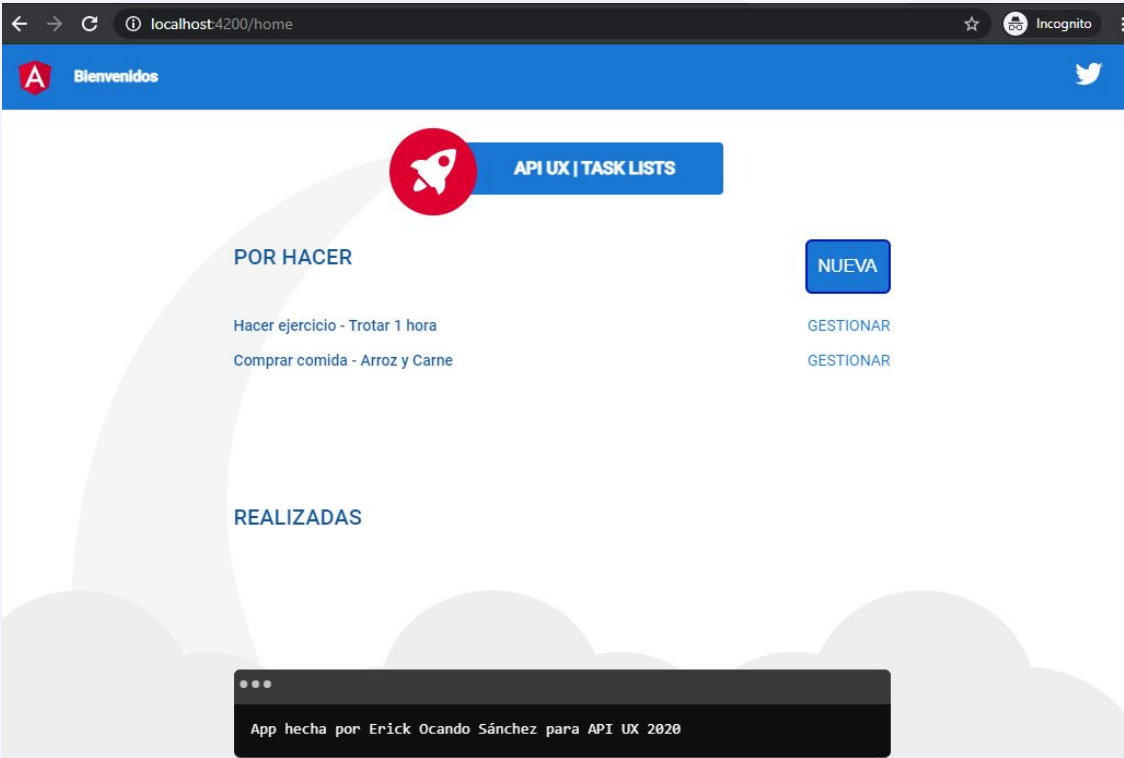
- 8. Instalar Node.js and npm
- 9. cd frontend.
- 10. npm install.
- 11. ng serve
- 12. El frontend debe quedar arriba en localhost:4200.

Repositorio:


<https://github.com/eocandos/prueba-apiux.git>

Pantallas

- Home (Listar tareas)



Nueva tarea




API UX | TASK LISTS

NUEVA TAREA

Nombre tarea *

Descripción tarea *

Guardar



API UX | TASK LISTS

NUEVA TAREA

Nombre tarea *

Tarea de Prueba

Descripción tarea *

Descripción de Prueba

Guardar



API UX | TASK LISTS

POR HACER

Tarea de Prueba - Descripción de Prueba

Hacer ejercicio - Trotar 1 hora

Comprar comida - Arroz y Carne


NUEVA

GESTIONAR

GESTIONAR

GESTIONAR

Opciones para administrar las tareas



API UX | TASK LISTS

POR HACER

Tarea de Prueba - Descripción de Prueba

Hacer ejercicio - Trotar 1 hora

Comprar comida - Arroz y Carne

NUEVA


GESTIONAR

Completada

Editar

Eliminar

Tarea completada



API UX | TASK LISTS

POR HACER

Hacer ejercicio - Trotar 1 hora

Comprar comida - Arroz y Carne

NUEVA

GESTIONAR


GESTIONAR

REALIZADAS

Tarea de Prueba - Descripción de Prueba

GESTIONAR

Editar tarea



API UX | TASK LISTS

EDITAR TAREA


Nombre tarea *

Hacer ejercicio

Descripción tarea *

Trotar 2 horas

Guardar



API UX | TASK LISTS

POR HACER

Hacer ejercicio - Trotar 2 horas

Comprar comida - Arroz y Carne

NUEVA

GESTIONAR

GESTIONAR

Eliminar tarea

POR HACER

Hacer ejercicio - Trotar 2 horas

Comprar comida - Arroz y Carne

NUEVA

GESTIONAR

GESTIONAR

Completada

Editar

Eliminar

REALIZADAS

Tarea de Prueba - Descripción de Prueba

GESTIONAR

POR HACER

Hacer ejercicio - Trotar 2 horas

NUEVA

GESTIONAR

REALIZADAS

Tarea de Prueba - Descripción de Prueba

GESTIONAR

Completar tarea

POR HACER

Hacer ejercicio - Trotar 2 horas

NUEVA

GESTIONAR

Completada

Editar

Eliminar

REALIZADAS

Tarea de Prueba - Descripción de Prueba

GESTIONAR

POR HACER

NUEVA

REALIZADAS

Hacer ejercicio - Trotar 2 horas

Tarea de Prueba - Descripción de Prueba

GESTIONAR

GESTIONAR

Reabrir tarea

POR HACER

NUEVA

REALIZADAS

Hacer ejercicio - Trotar 2 horas

Tarea de Prueba - Descripción de Prueba

GESTIONAR

Reabrir

Eliminar

POR HACER

NUEVA

Hacer ejercicio - Trotar 2 horas

Tarea de Prueba - Descripción de Prueba

GESTIONAR

GESTIONAR

Repositorio:

<https://github.com/eocandos/prueba-apiux.git>

Instancia EC2 AWS

<http://ec2-18-217-92-102.us-east-2.compute.amazonaws.com/>

