

PRESENTADO POR ALEXANDER DAVID PAZ
INGENIERO DE SISTEMAS
ESPECIALISTA EN PROCESOS DE DESARROLLO DE SOFTWARE.

REQUERIMIENTO:

Usuario solicita crear una aplicación 360 implementando mejores prácticas de arquitectura y codificación para una aplicación web que permita a los usuarios ver las propiedades disponibles según criterios de búsqueda (precio, nombre, dirección).

Solución.

Se crea una API en .net Core para implementar el Backend que soporte las peticiones de la aplicación web aplicando clean arquitectura y el patrón CQRS.

Se crea base de datos en mongoDB y se crean fakedata para realizar pruebas desde la aplicación.

se crea app web con react.js para visualizar las propiedades aplicando estilos Css para una mejor visualización.

Adicional, se configura un certificado de seguridad SSL local para poder usar https en las peticiones web y se monta al contenedor de la api.

Se utiliza **Swagger** para la documentación de la API.

La implementación realizada se coloca en contenedores de Docker para una mayor facilidad al momento de desplegar los proyectos por medio de un docker-compose.yml.

contenedores:

mongoDB

inmuebles-api <https://localhost:5001/swagger/index.html>

react-app <http://localhost:3000/>

Para ejecutar el proyecto, se debe hacer lo siguiente.

1. descomprimir el archivo .zip en una ubicación del disco.
2. Abrir el CMD en la ubicación donde se encuentra el archivo docker-compose.yml - utilizar command-prompt
3. ejecutar el comando para levantar los contenedores.

`docker-compose up -d`

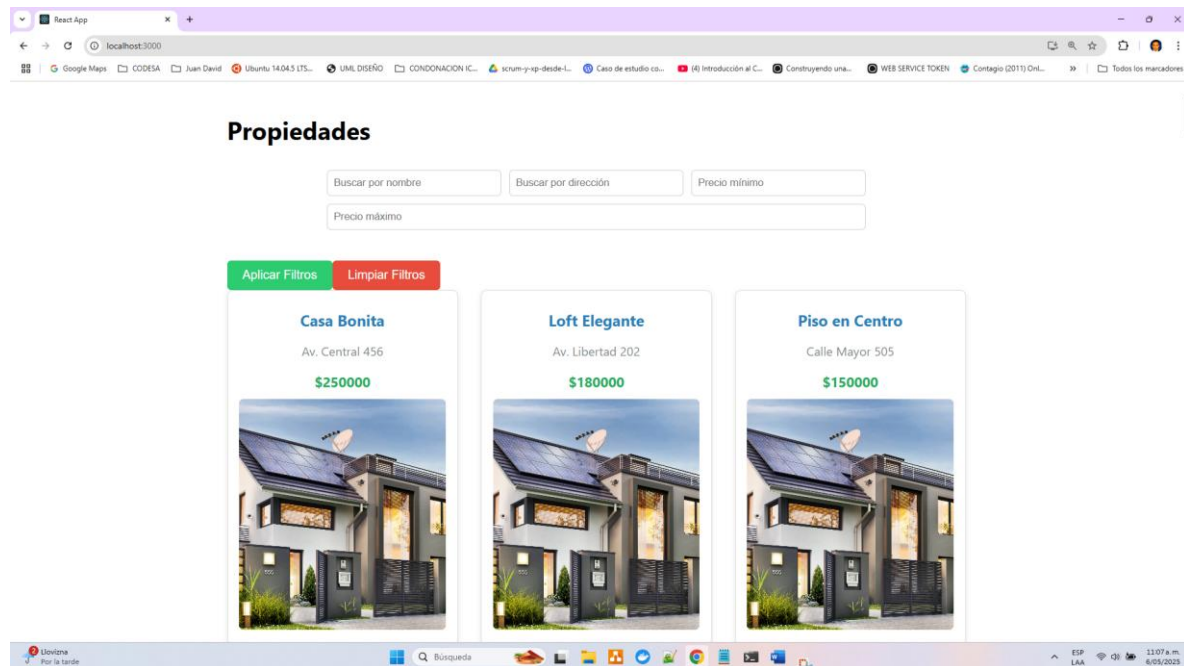
4. ejecutar el comando para cargar los fakedata en la base de datos de mongo.

`Docker exec -i mongo_db mongosh < DB_init.js`

Los contenedores deben estar ejecutándose para poder pasar el paso 5.

```
PS C:\Users\dapri> docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
83a7d13db442   inmublessolution-react-app         "docker-entrypoint.s..." 26 minutes ago Up 26 minutes 0.0.0.0:3000->3000/tcp             react_app
1d2c3cd8229c   inmublessolution-inmuebles-api     "dotnet Inmuebles.Api..." 26 minutes ago Up 26 minutes 8080-8081/tcp, 0.0.0.0:5001->443/tcp inmuebles_api
7e569bb70583   mongo:7.0                          "docker-entrypoint.s..." 26 minutes ago Up 26 minutes 0.0.0.0:27017->27017/tcp           mongo_db
```

5. ingresar a `http://localhost:3000/` y se deben visualizar las propiedades existentes en la base de datos.



Requisito:

Debe estar instalado Docker en el computador según el sistema operativo.