



DIGITAL
HUBO SAC

Spring Boot JAVA

OPTIMIZACIÓN DEL PROCESO DE GESTIÓN DEL
SOPORTE A LOS CLIENTES



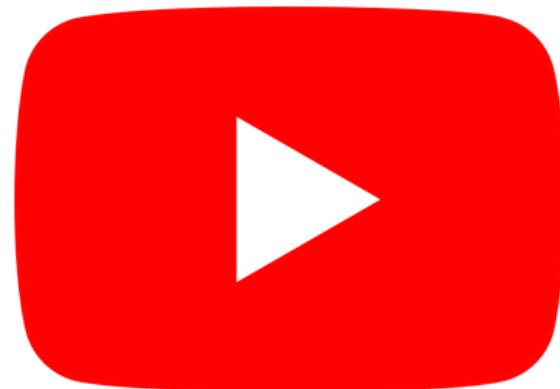
CURSOS INACCESIBLES

No se encontraron cursos gratuitos de Spring Boot con Java que permitan comprobar fechas de inicio o fin. Los pocos disponibles son costosos o carecen de detalle.



CURSOS ALTERNATIVOS

Ante la falta de cursos gratuitos, detallados o verificables de Spring Boot con Java, opté por buscar alternativas en YouTube y en la guía oficial de Spring para seguir avanzando.



INFORMACIÓN SOBRE **SPRING**



¿QUÉ ES SPRING?

Spring es el nombre que recibe todo un ecosistema de tecnologías orientadas al desarrollo de aplicaciones modernas en Java. Se trata de un proyecto principal que integra varios subproyectos, entre ellos: Spring Framework, Spring Boot, Spring Data, entre otros.



spring
by Pivotal™

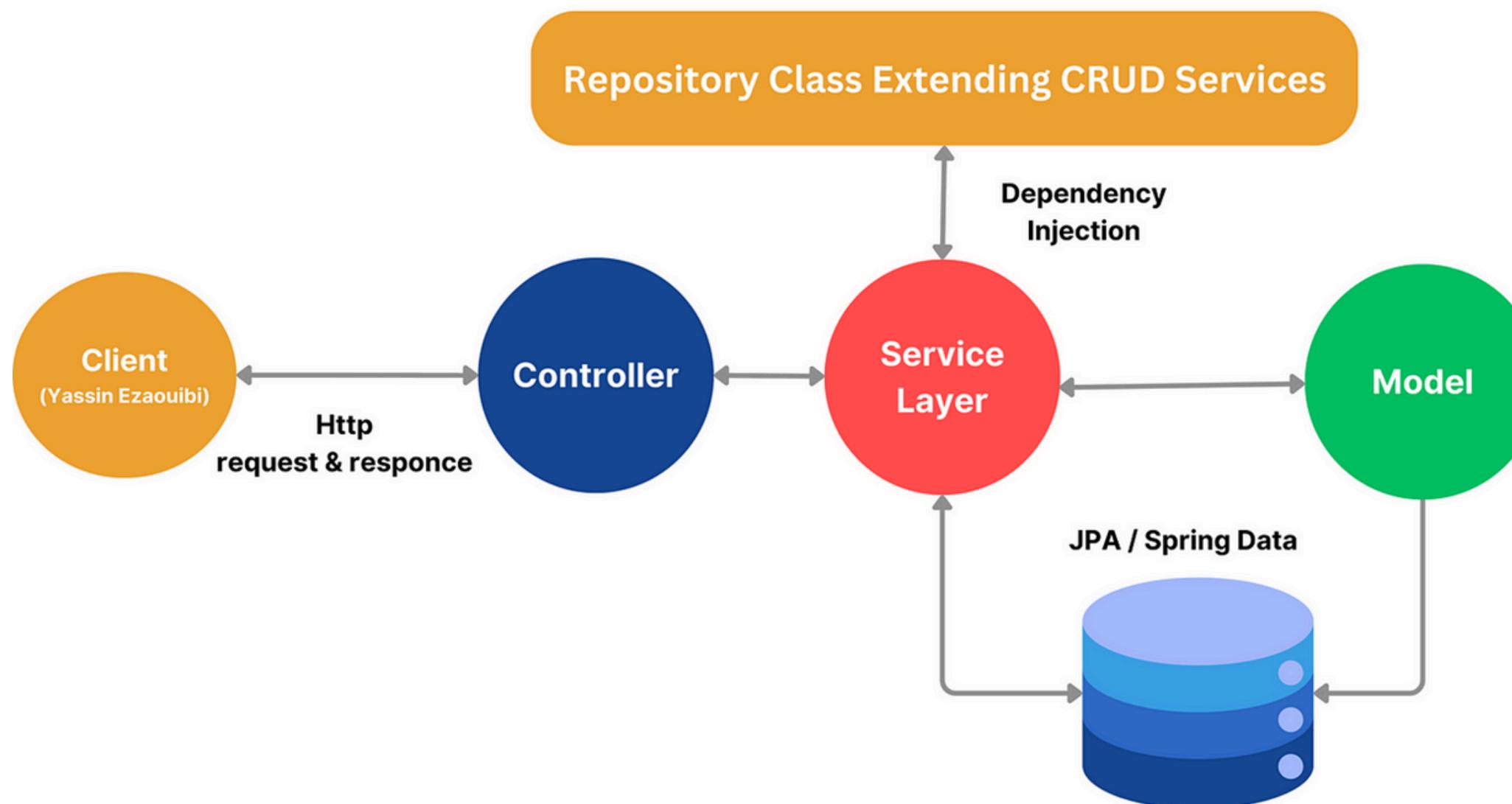
¿QUÉ ES SPRING BOOT?

Spring Boot es una herramienta que hace más fácil y rápido crear aplicaciones en Java con Spring. Elimina configuraciones complicadas y permite crear apps listas para ejecutarse con:

- Configuración automática
- Ajustes predeterminados inteligentes
- Aplicaciones que se ejecutan solas (sin servidor externo)



ARQUITECTURA SPRING BOOT

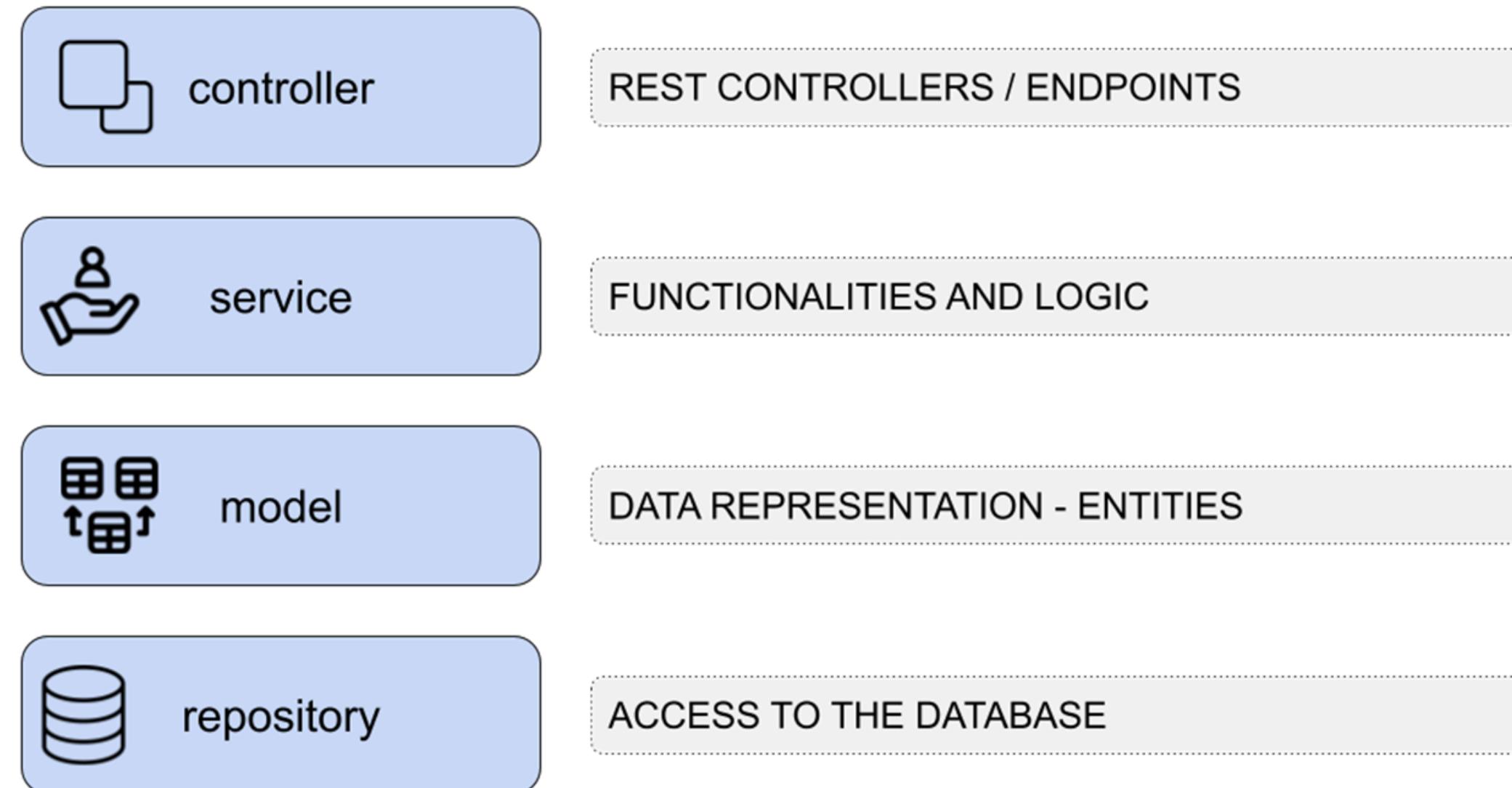


CARACTERÍSTICAS DE SPRING BOOT

Spring Boot permite crear aplicaciones Java que se ejecutan solas, sin necesidad de servidores externos. Entre sus principales ventajas:

- Incluye servidores como Tomcat o Jetty integrados (no necesitas archivos WAR).
- Usa dependencias preconfiguradas para facilitar la instalación y configuración.
- Configura automáticamente Spring y otras bibliotecas.
- Ofrece herramientas listas para producción: métricas, monitoreo y configuración externa.
- No requiere XML ni generación de código extra.

ESTRUCTURA SPRING BOOT



INFORMACIÓN DEL PROYECTO



IDENTIFICACIÓN DEL PROBLEMA

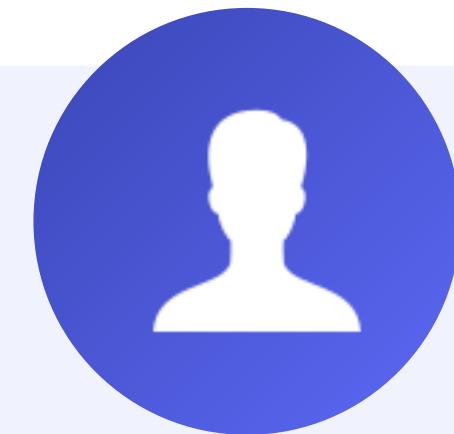


El problema ocurre en el sector de servicios digitales, específicamente en el área de soporte técnico de la empresa Digital Buho S.A.C., donde se gestionan las solicitudes de atención de los clientes.



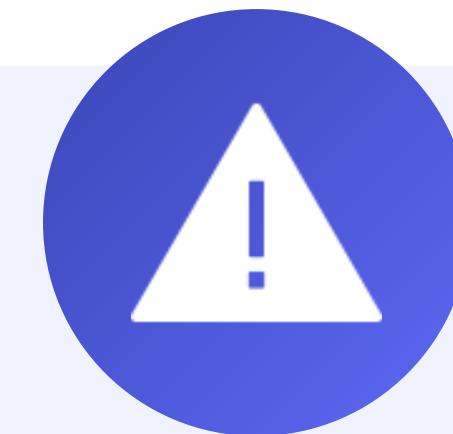
DONDE OCURRE?

En el Área de Soporte Técnico de la Empresa



A QUIEN AFECTA?

Al personal encargado de atender las solicitudes de soporte técnico y, de forma indirecta, a los clientes que esperan atención y, de forma directa a la empresa.



LA CAUSA PRINCIPAL?

Es el desconocimiento del personal, sobre el estado actual de cada solicitud de soporte

EN UNA SOLA FRASE EL PROBLEMA



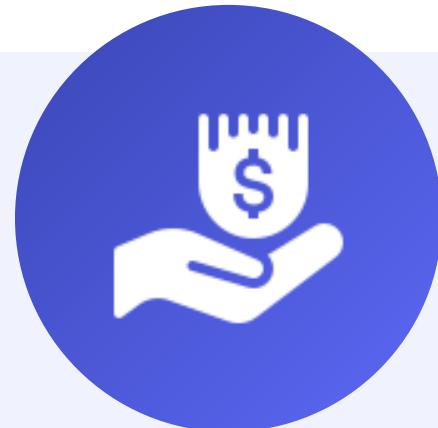
**DEFICIENCIA EN LA GESTIÓN DEL SOPORTE A LOS
CLIENTES EN LA EMPRESA DIGITAL BUHO S.A.C**

IMPACTO EN LOS INDICADORES DE LA EMPRESA



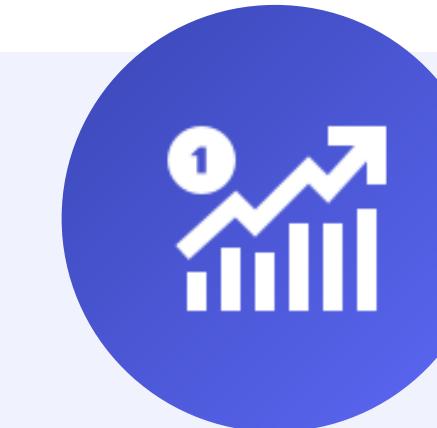
COSTOS

Se incrementan los costos operativos por tener que repetir tareas, hacer seguimientos manuales, por la mala gestión de solicitudes.



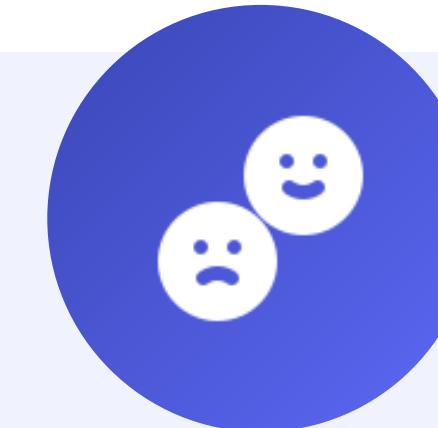
INGRESOS

Se pierden oportunidades de ingresos, porque los clientes insatisfechos pueden dejar de contratar servicios o no renovar contratos.



VENTAS

Disminuyen las ventas por la mala reputación o recomendaciones negativas de clientes que recibieron atención tardía o ineficiente.



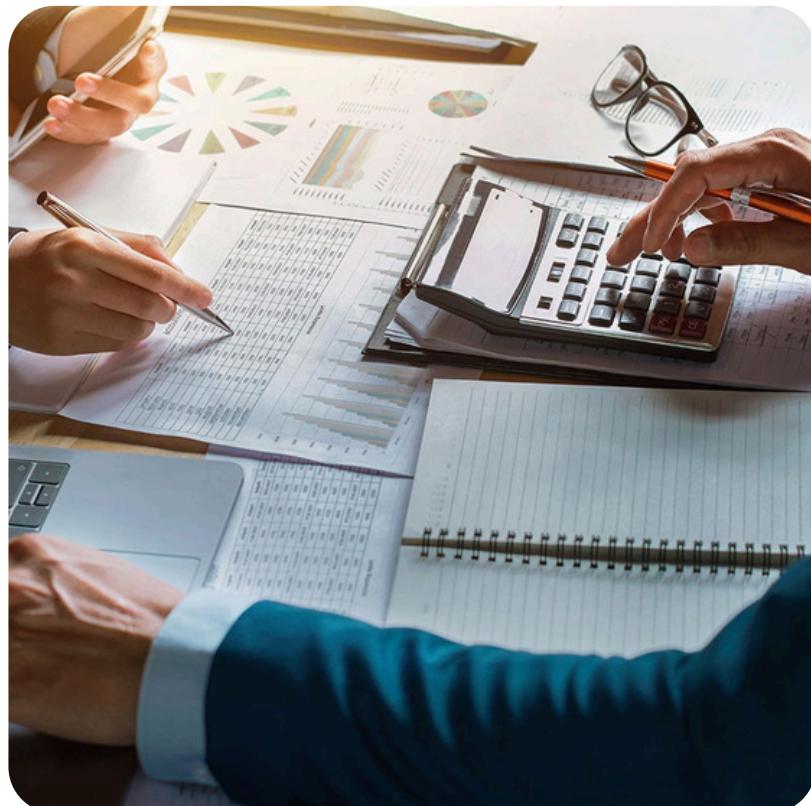
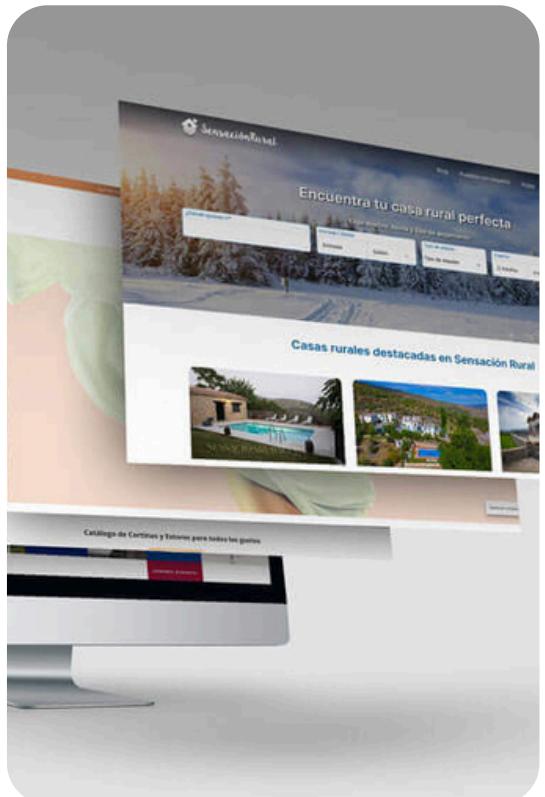
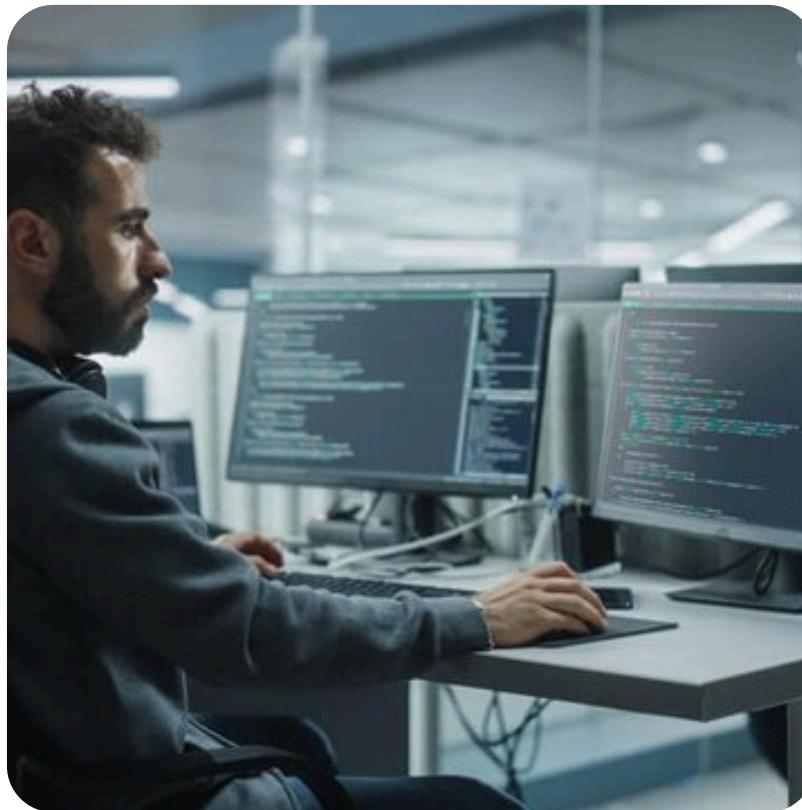
SATISFACCIÓN AL CLIENTE

Baja la satisfacción porque los clientes no reciben respuestas rápidas ni seguimiento claro a sus solicitudes.

EN UNA SOLA FRASE EL OBJETIVO GENERAL



**OPTIMIZACIÓN EN LA GESTIÓN DEL SOPORTE A LOS
CLIENTES EN LA EMPRESA DIGITAL BUHO S.A.C**



OBJETIVOS ESPECÍFICOS

- **Reducir los costos operativos** mejorando el control y seguimiento automatizado de las solicitudes, para evitar retrabajos y tareas manuales innecesarias.
- **Evitar la pérdida de ingresos** asegurando una gestión oportuna de las solicitudes que mantenga a los clientes satisfechos y fidelizados.
- **Mejorar las ventas** fortaleciendo la imagen de la empresa a través de un servicio técnico más rápido, organizado y confiable.
- **Aumentar la satisfacción del cliente** brindando respuestas más rápidas y con mejor seguimiento del estado de sus solicitudes.

SOLUCIÓN PROPUESTA

Desarrollar una aplicación web con API RESTful en Spring Boot para gestionar en tiempo real las solicitudes de soporte técnico, mejorando la organización del equipo y la atención al cliente.

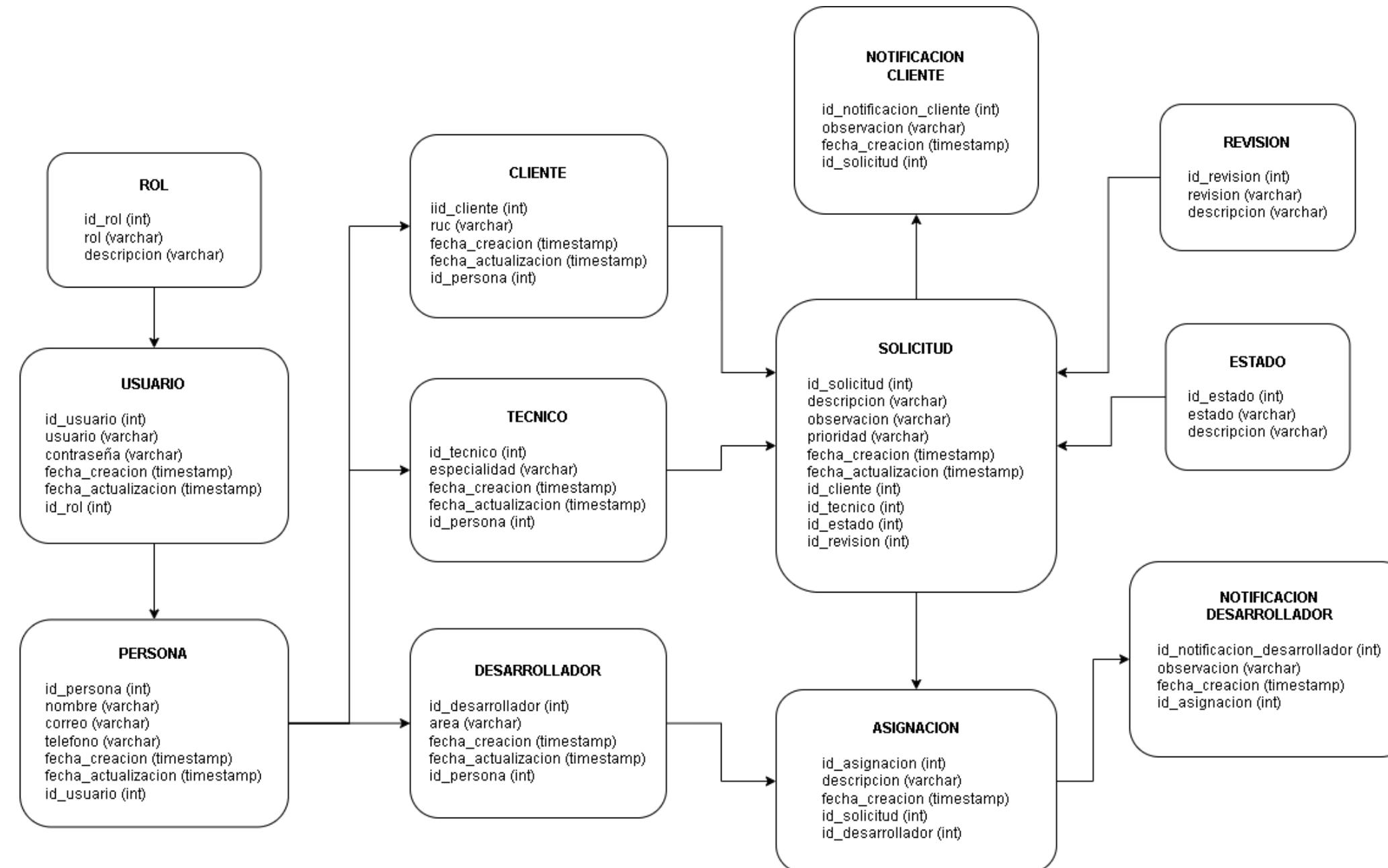


CARACTERISTICAS DEL PROYECTO



MODELO LÓGICO

(Base de Datos)



MODELO FISICO

(Base de Datos)



ENDPOINTS RESTFUL



Todos los endpoints siguen una estructura común que facilita su uso y organización:

METODO	RUTA	ACCIÓN
GET	/api/{tabla}/listar	Listar
POST	/api/{tabla}/crear	Crear
GET	/api/{tabla}/listar/{id}	Listar por ID
PUT	/api/{tabla}/actualizar/{id}	Modificar
DELETE	/api/{tabla}/eliminar/{id}	Eliminar

PRUEBAS POSTMAN



METODO	RUTA	ACCIÓN
GET	/api/{tabla}/listar	Listar

The screenshot shows the Postman interface for a 'GET Listar Usuarios' request. The URL is https://digital-buho-api-soporte.onrender.com/api/roles/listar. The 'Body' tab is selected, showing a raw JSON response:

```
[{"id_rol": 1, "rol": "CLIENTE", "descripcion": "Rol para clientes registrados"}, {"id_rol": 2, "rol": "TECNICO", "descripcion": "Rol para técnicos registrados"}, {"id_rol": 3, "rol": "DESARROLLADOR", "descripcion": "Rol para desarrolladores registrados"}]
```

A screenshot of the Postman interface showing the 'Body' tab with a JSON response. The response is a list of three roles: CLIENTE, TECNICO, and DESARROLLADOR, each with an ID, role name, and description.

```
1 [ 2 { 3 "id_rol": 1, 4 "rol": "CLIENTE", 5 "descripcion": "Rol para clientes registrados" 6 }, 7 { 8 "id_rol": 2, 9 "rol": "TECNICO", 10 "descripcion": "Rol para técnicos registrados" 11 }, 12 { 13 "id_rol": 3, 14 "rol": "DESARROLLADOR", 15 "descripcion": "Rol para desarrolladores registrados" 16 } ]
```

GET

PRUEBAS POSTMAN



METODO	RUTA	ACCIÓN
POST	/api/{tabla}/crear	Crear

The screenshot shows the Postman interface with a POST request to `https://digital-buho-api-soporte.onrender.com/api/roles/crear`. The 'Body' tab is selected, showing a raw JSON payload:

```
1 {  
2   "rol": "CLIENTE",  
3   "descripcion": "Para cliente"  
4 }
```

The screenshot shows the Postman response window with a 200 OK status. The JSON response is displayed:

```
1 {  
2   "id_rol": 4,  
3   "rol": "CLIENTE",  
4   "descripcion": "Para cliente"  
5 }
```

POST

PRUEBAS POSTMAN



MÉTODO	RUTA	ACCIÓN
GET	/api/{tabla}/listar/{id}	Listar por ID

GET Listar Usuario +

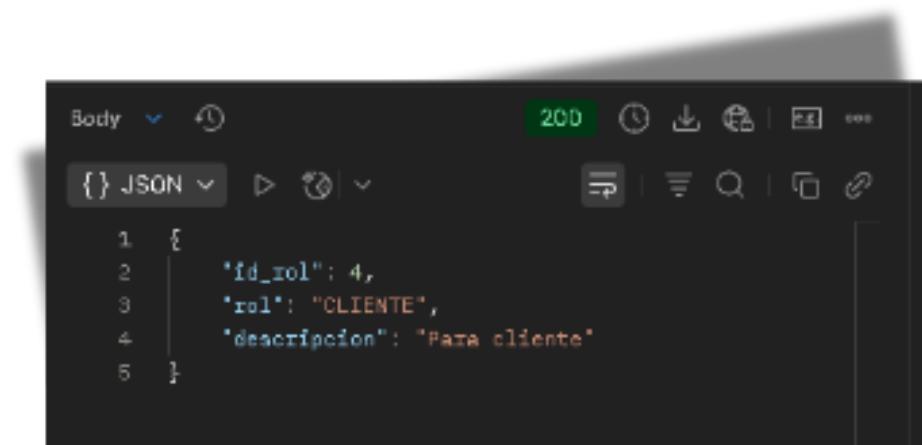
HTTP Digital-Buho-SAC / API-Roles / Listar Usuario

GET https://digital-buho-api-soporte.onrender.com/api/roles/listar/4

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL **JSON** ▾

1



GET / ID

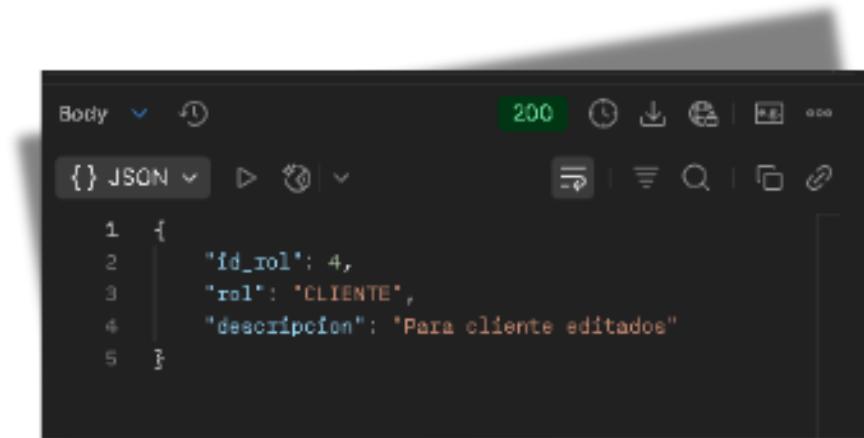
PRUEBAS POSTMAN



METODO	RUTA	ACCIÓN
PUT	/api/{tabla}/actualizar/{id}	Modificar

Screenshot of the Postman application interface. The title bar says "PUT Actualizar Usuario". The URL field shows "https://digital-buho-api-soporte.onrender.com/api/roles/actualizar/4". The "Body" tab is selected, showing the raw JSON payload:

```
1 {  
2   "rol": "CLIENTE",  
3   "descripcion": "Para cliente editados"  
4 }
```



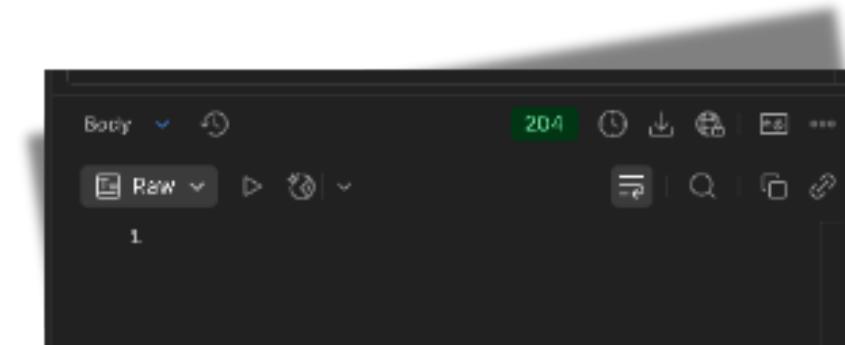
PUT

PRUEBAS POSTMAN



METODO	RUTA	ACCIÓN
DELETE	/api/{tabla}/eliminar/{id}	Eliminar

A screenshot of the Postman application interface showing a DELETE request to the endpoint https://digital-buho-api-soporte.onrender.com/api/roles/eliminar/4. The request method is set to DELETE, and the URL is specified. The 'Body' tab is selected, showing the raw JSON input: "1". Other tabs like 'Headers' and 'Authorization' are visible but not active.



DELETE

DOCUMENTACIÓN CON SWAGGER



Se integró Swagger en el proyecto para la documentación automática de la API RESTful, incluyendo la configuración y las dependencias necesarias.

The screenshot shows the Swagger UI interface for a RESTful API. At the top, it displays the URL `https://digital-buho-api-soporte.onrender.com/swagger-ui/index.html`. The main content area is titled "OpenAPI definition v0 OAS 3.0". It includes a "Servers" section with the URL `https://digital-buho-api-soporte.onrender.com - Generated server url`. Below this, there are two sections: "usuario-rest-controller" and "tecnico-rest-controller". Each section lists several API endpoints with their corresponding HTTP methods and URLs. The "usuario-rest-controller" section contains:

- PUT /api/usuarios/actualizar/{id}
- POST /api/usuarios/crear
- GET /api/usuarios/listar
- GET /api/usuarios/listar/{id}
- DELETE /api/usuarios/eliminar/{id}

The "tecnico-rest-controller" section contains:

- PUT /api/tecnicos/actualizar/{id}
- POST /api/tecnicos/crear
- GET /api/tecnicos/listar
- GET /api/tecnicos/listar/{id}

CONTROLADOR PARA WEB



Se implementó un controlador web con operaciones CRUD para un manejo más rápido y eficiente de los recursos.

The screenshot shows a web application interface titled 'Listado de Roles'. At the top, there is a search bar labeled 'Buscar rol por ID:' with a 'Buscar' button. Below the search bar is a table with three rows of data. The table has columns for 'ID', 'Rol', 'Descripción', and 'Acciones'. The data is as follows:

ID	Rol	Descripción	Acciones
2	TECNICO	Rol para técnicos registrados	Editar Eliminar
3	DESARROLLADOR	Rol para desarrolladores registrados	Editar Eliminar
5	CLIENTE	Rol para clientes registrados.	Editar Eliminar

At the bottom of the table, there are two buttons: 'Restablecer lista' and 'Agregar nuevo rol'.

REPOSITORIO EN GITHUB



El proyecto fue subido a GitHub para su acceso, revisión y colaboración.

The screenshot shows a GitHub repository page for 'Digital-Buho-API-SOPORTE'. The repository is public and has 1 branch and 0 tags. The main branch contains several commits from 'JuanYslaCadenas' made 3 days ago, including changes to Dockerfile, .mvnw, .src, .gitattributes, .gitignore, mvnw, mvnw.cmd, and pom.xml. The repository has 0 stars, 1 watching, and 0 forks. It includes sections for Activity, Releases (none published), Packages (none published), Languages (HTML 67.2%, Java 32.6%, Dockerfile 0.2%), and Suggested workflows (Java with Gradle). A button to 'Add a README' is visible at the bottom left.

PROYECTO PUBLICADO EN LA NUBE



DESPLIEGUE Y ALOJAMIENTO DEL PROYECTO

El proyecto está completamente en la nube: la base de datos se aloja en Supabase, la aplicación está desplegada en Render, y el código fuente se encuentra disponible en GitHub.



 Render

ENLACES DEL PROYECTO



- **Link del Proyecto:** <https://digital-buho-api-soporte.onrender.com>
- **Repositorio en GitHub:** <https://github.com/SiologoDr/Digital-Buho-API-SOPORTE>

NOTA: Se debe activar SUPABASE cada ves que se quiera entrar al LINK DEL PROYECTO.