



No __ s19070104 __

INSTITUTO TECNOLÓGICO SUPERIOR DE JEREZ

Libramiento Fresnillo-Tepetongo, Fracc. Los Cardos,
99863 Jerez de García Salinas, Zac.

GARCIA FLORES ANAHI

anahi.gf@tecjerez.edu.mx

INGENIERÍA EN SISTEMAS
COMPUTACIONALES

DevSecOps - Desarrollo, Seguridad y Operaciones de Software

Prof. Salvador Acevedo

9° SEMESTRE

EVALUACION PARCIAL 2

Fecha: 10/10/25

EVALUACIÓN PARCIAL 2

Index

Objetivo.....	2
Desarrollo.....	3
Despliegue de servicios con Kubernetes.....	3
Ejecución en Kubernetes.....	3
Resultados.....	4
Conclusiones.....	5

Objetivo

Realizar la orquestación de servicios para una aplicación web institucional del Área de Tutorías del ITSJ, implementando contenedores para el backend, frontend y base de datos, desplegados mediante Kubernetes (Minikube).

Desarrollo

Link github: <https://github.com/anhiGF/Proyecto.git>

Despliegue de servicios con Kubernetes

Se orquestaron tres servicios: MySQL, Backend, y Frontend, dentro del namespace tutoria.

The screenshot shows the VS Code interface with five tabs open, each containing a YAML configuration file for Kubernetes:

- namespace.yaml**: A Namespace resource named "tutoria".
- frontend.yaml**: A Deployment resource named "frontend" with one replica, using the "luni1993/tutor" image.
- mysql.yaml**: A PersistentVolumeClaim resource named "mysql-pvc" for the "tutoria" namespace.
- backend.yaml**: A Deployment resource named "backend" with one replica, using the "luni1993/tutor" image.
- secrets.yaml**: A Secret resource named "mysql-secret" containing MySQL root password information.

Ejecución en Kubernetes

1. Se aplicaron los YAML.
2. Se verificó el estado de los pods.
3. Se verificaron los servicios y puertos.

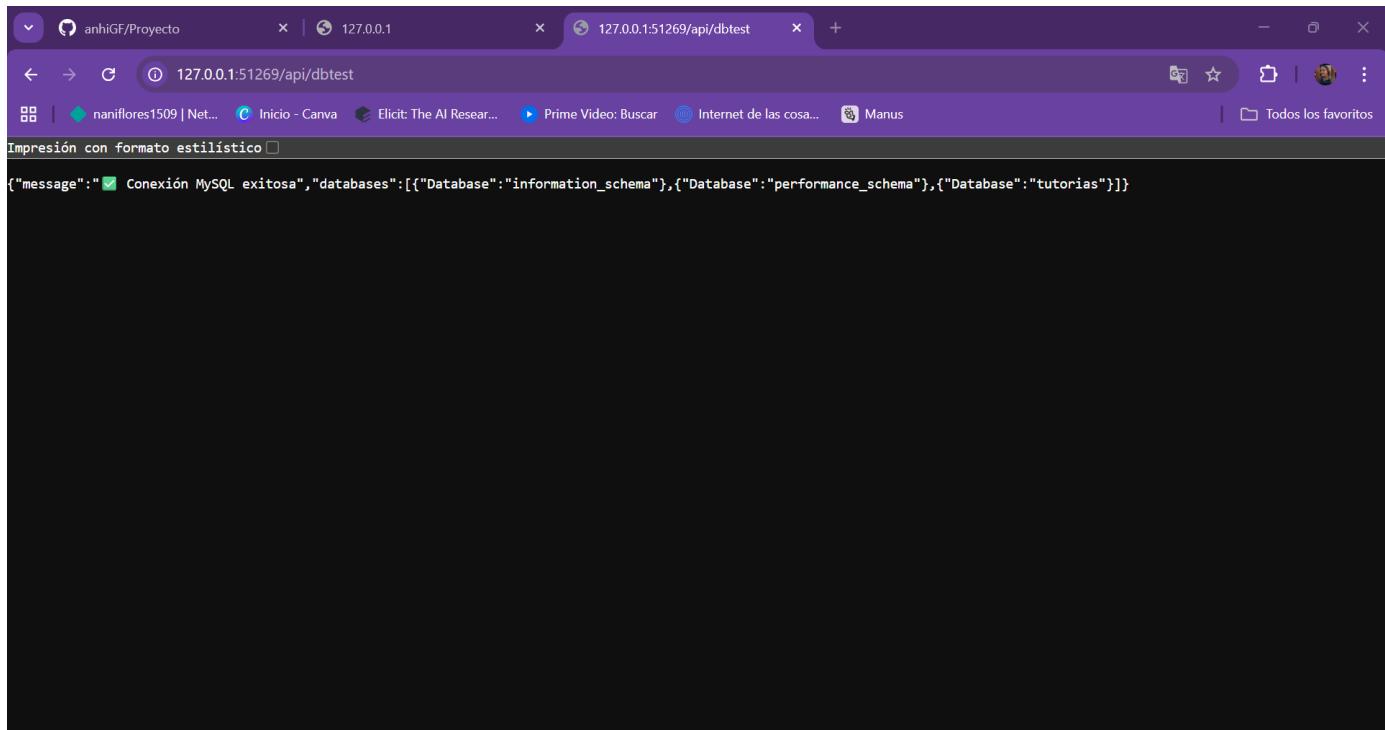
The screenshot shows the VS Code interface with the terminal tab active, displaying the output of running Docker and minikube commands:

```
PS C:\Users\garc1\OneDrive\Escritorio\9_semestre\Pruebas_Docker\Proyecto\proyecto-tutorias\frontend> docker build -t luni1993/tutorias-frontend:latest .
-> [stage-1 2/2] COPY --from=build /app/build /usr/share/nginx/html
=> exporting to image
=> exporting layers
=> => writing image sha256:c3ba48d6b7558d7fd8fe11211485eb29cf63a8977978ef977d1cf96557bee76
=> => naming to docker.io/luni1993/tutorias-frontend:latest

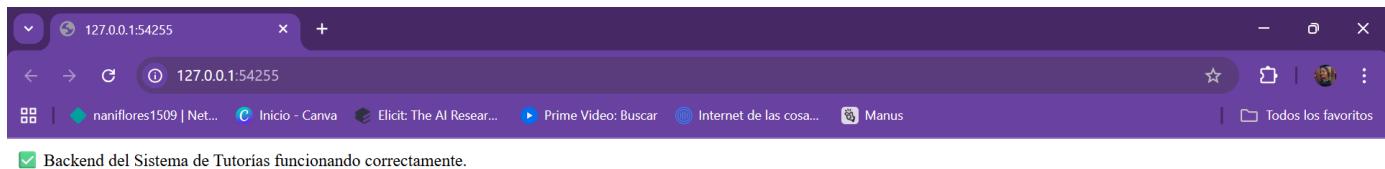
View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/ydoi3hg0ydbqglabam123yyf
PS C:\Users\garc1\OneDrive\Escritorio\9_semestre\Pruebas_Docker\Proyecto\proyecto-tutorias\frontend> minikube start
>> kubectl apply -f k8s/namespace.yaml
>> kubectl apply -f k8s/secrets.yaml
>> kubectl apply -f k8s/mysql.yaml
>> kubectl apply -f k8s/backend.yaml
>> kubectl apply -f k8s/frontend.yaml
>>
=> minikube v1.37.0 on Microsoft Windows 11 Pro 10.0.26200.6725 Build 26200.6725
=> Using the docker driver based on existing profile
=> Starting "minikube" primary control-plane node in "minikube" cluster
=> Pulling base image v0.0.48 ...
=> Restarting existing docker container for "minikube" ...
```

Resultados

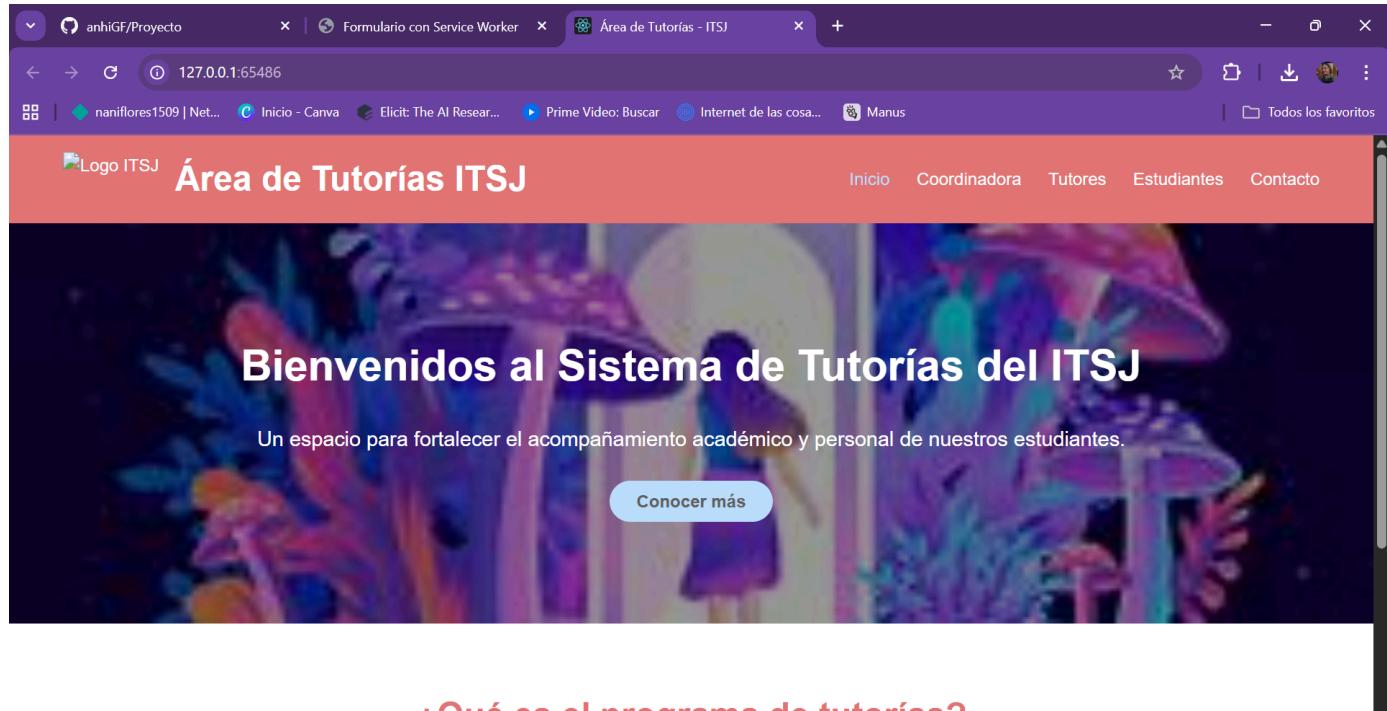
- SQL



- Backend



- Frontend: Acceso exitoso a la aplicación



¿Qué es el programa de tutorías?

Conclusiones

- Se logró desplegar exitosamente una arquitectura multicapa: frontend, backend y base de datos.
- Se fortalecieron las habilidades de configuración de Dockerfiles, YAMLs de despliegue, y administración de pods y servicios en Kubernetes.
- Minikube resultó una herramienta ideal para simular un entorno real de clúster en un entorno local.
- Finalmente, se comprobó la comunicación efectiva entre los servicios y la persistencia de datos mediante MySQL dentro del clúster.