

Entrega individual

Presentación

El objetivo de este trabajo es poner en práctica los conocimientos sobre técnicas de gestión y orquestación de contenedores. Cada estudiante deberá realizar el trabajo de manera individual. La calificación obtenida con esta entrega corresponde al 40% de la nota de la asignatura.

Este documento describe los objetivos del trabajo, tareas obligatorias y opcionales y requisitos de entrega. Como parte del trabajo, se deberá redactar y entregar una documentación.

Plazo de entrega: Domingo, 23 de noviembre de 2025 a las 23:55.

Objetivo

Cada estudiante debe desarrollar una aplicación web compuesta, al menos, por un servidor web, un servidor de BBDD y un servicio adicional de libre elección. La temática de la aplicación es libre, pero debe incluir o utilizar datos obtenidos de un repositorio público, p.e. Kaggle (<https://www.kaggle.com/datasets>) o portales de Open Data (p.e. <https://opendata.euskadi.eus/>, <https://datos.gob.es/>, <https://www.data.gov.uk/>, ...).

Para el desarrollo, cada estudiante debe elegir 3 imágenes de Docker Hub que formen la base de la aplicación:

- Un servidor web. Puede ser cualquiera de los disponibles.
- Un servidor de BBDD. Puede ser cualquiera de los disponibles, excepto Redis.
- Una imagen a libre elección de cada estudiante. No puede ser un servidor web o de BBDD, salvo que esté debidamente justificado. No pueden ser gestores de BBDD como phpMyAdmin o Adminer, ni herramientas de visualización como Grafana o Kibana. La imagen elegida debe ser coherente con el objetivo de la aplicación.

La complejidad de la aplicación queda a criterio de cada estudiante. Inicialmente, el foco del desarrollo debe estar en la correcta configuración de servicios/contenedores y no en crear una aplicación realista con el mayor número de funcionalidades de usuario posible.

Independientemente de la temática de la aplicación, es necesario gestionar la persistencia, al menos, del servidor de BBDD. Los datos de la aplicación deben guardarse fuera de un contenedor para permitir copias de seguridad.

Entrega

La entrega del trabajo será una documentación en formato PDF, que se deberá subir a la tarea correspondiente en el apartado Evaluación de eGela. Los apartados que debe tener la documentación se describen más adelante en este documento.

Los ficheros que formen la aplicación deberán entregarse en un repositorio de código, preferiblemente GitHub. Este repositorio de código debe ser privado y se debe dar acceso al usuario *ulopeznova*. En caso de no utilizar GitHub, ponerse en contacto con el profesor. Hacer público el repositorio de código o entregar el trabajo de forma diferente puede suponer una penalización de hasta 4 puntos en la nota.

Administración de Sistemas - Curso 2025 / 2026

Requisitos mínimos para obtener un 5

- Desarrollar una aplicación funcional. Debe ser posible acceder a todas sus funcionalidades desde un navegador web actual.
- Crear las imágenes Docker que sean necesarias. Cada imagen creada debe subirse a un repositorio público de Docker Hub. En la documentación, se debe describir el propósito de cada imagen utilizada, indicar si se usa una versión provista en Docker Hub o si se ha creado como parte del trabajo, describir brevemente cada Dockerfile creado y aportar la URL de las imágenes en Docker Hub.
- Crear un entorno Docker Compose para poder ejecutar la aplicación desarrollada. Se debe crear un fichero YAML de Docker Compose y describirlo en la documentación.

Tareas adicionales

- Crear un despliegue Kubernetes para ejecutar la aplicación desarrollada. El resultado debe ser equivalente al despliegue creado con Docker Compose. Es obligatorio utilizar objetos *Deployment*. En la documentación, describir los objetos Kubernetes creados y las conexiones entre ellos. Se deben entregar los ficheros de configuración creados (**3 puntos**).
 - Para conseguir los 3 puntos, se debe gestionar correctamente la persistencia de la BBDD.
- Incluir imágenes/contenedores adicionales que aporten nuevas funcionalidades a la aplicación. Pueden integrarse como parte del despliegue Docker Compose o del despliegue Kubernetes. Deben comunicarse con alguna de las imágenes existentes y su inclusión debe ser coherente con el sentido de la aplicación. Los gestores de BBDD como phpMyAdmin o Adminer no cuentan. En la documentación, describir cada imagen adicional utilizada y la funcionalidad que provee (**hasta 1 punto por imagen**).
- Utilizar funcionalidades de Kubernetes no vistas en clase. Se deben describir en la documentación (**hasta 1 punto**).

Respecto a la parte con Kubernetes, el despliegue puede realizarse en gestores locales (p.e. minikube o Docker Desktop) o con Google Kubernetes Engine (GKE). Se debe indicar en la documentación cuál se ha utilizado y por qué. La elección de gestor no afecta a la nota.

Documentación

La documentación de la aplicación debe tener los siguientes apartados:

- Descripción de la aplicación desarrollada y explicación de sus características. Este apartado debe constar, al menos, de 300 palabras e incluir:
 - Breve descripción y enlace al repositorio público de datos elegido.
 - Motivación del trabajo, por qué se ha elegido la temática de la aplicación.
- Enlace a repositorio de código con los ficheros creados (p.e. archivos YAML de Docker/Kubernetes).
- Listado de las tareas realizadas (obligatorias y opcionales).
- Breve manual de configuración/despliegue/instalación.
 - Parte Docker Compose: obligatorio.
 - Parte Kubernetes: sólo si se ha realizado el despliegue con Kubernetes.
- Breve manual de uso de la aplicación.
 - No es necesario cubrir todas las funcionalidades, sólo las más relevantes.
- Explicaciones requeridas por cada tarea realizada.
- Preguntas técnicas. Ver descripción a continuación.

Administración de Sistemas - Curso 2025 / 2026

En el apartado “*preguntas técnicas*”, se debe dar respuesta a las preguntas presentadas a continuación. Se recomienda contestarlas tras finalizar el desarrollo de la entrega. Se valorará la capacidad de análisis y la reflexión técnica. Se valorará ser directo y conciso en cada respuesta (1 o 2 párrafos es suficiente). No se aceptarán respuestas genéricas o teóricas.

- Describe un problema concreto que tuviste al construir una de las imágenes Docker y cómo lo resolviste.
- Explica cómo se comunican los contenedores entre sí, y qué configuraciones han sido necesarias. Es suficiente explicarlo para uno de los 2 despliegues (Docker Compose o Kubernetes).
- ¿Qué pasos has seguido para verificar el funcionamiento de la aplicación, antes de entregarla?
- ¿Cómo has organizado el repositorio para facilitar la comprensión del proyecto? ¿Has seguido alguna guía o convención?

Evaluación

Para obtener una calificación mínima de Aprobado (5.0) es obligatorio realizar correctamente las tareas obligatorias. La calificación del trabajo será Suspenso si las tareas obligatorias no se cumplen, aunque las opcionales se cumplan.

En la evaluación se valorarán, entre otros:

- La calidad de la documentación entregada, incluyendo el formato y la redacción. Los fallos en la documentación, tanto de redacción como de formato, resultarán en penalizaciones en la nota.
- Las configuraciones realizadas, más allá de lo visto en la asignatura. Las configuraciones inseguras (p.e. utilizar contraseñas débiles) resultarán en penalizaciones en la nota.

La evaluación de la parte obligatoria (Docker y Docker Compose) se hará en una máquina virtual remota. Por este motivo, se recomienda probarla y verificar que funciona correctamente en una ubicación diferente a la que se utilice para desarrollo (p.e. en una máquina virtual de Google Cloud Platform).

Respecto a la documentación, no es necesario aportar una explicación detallada de cada fichero creado (p.e. no es necesario explicar un Dockerfile línea a línea). Es suficiente con indicar el objetivo de cada fichero creado, una explicación general y, si procede, una justificación de las decisiones tomadas.