

Examen Temas 5 y 6 (Modelo)

Nombre y apellidos:

Esta página contiene los requisitos y condiciones del examen, además de unas pautas sobre su entrega. Los ejercicios a realizar se encuentran en las páginas siguientes.

Pre-requisitos:

Tener una máquina virtual de Ubuntu Server funcional. Se recomienda crear una nueva máquina virtual en lugar de utilizar una en la que se hayan realizado pruebas o ejercicios de la asignatura. Se recomienda revisar la configuración de puertos y utilizar la máquina a través de SSH desde la consola de vuestro equipo.

Además, dos ejercicios requieren utilizar las aplicaciones Elasticsearch y Logstash, por lo que se recomienda disponer de un entorno Docker Compose que permita poner ambos en marcha. Uno de los ejercicios requiere tener una cuenta en GitHub para utilizar su funcionalidad *Actions* de CI/CD.

Condiciones del examen:

- Tiempo total: 100 minutos
- Está permitido utilizar los apuntes de la asignatura y buscar información en foros, manuales o documentación en internet.
- Los teléfonos móviles deberán estar apagados o en silencio y no se permite su uso.
- No se permite utilizar sistemas de mensajería o asistentes virtuales durante el examen. En caso de copia o uso de sistemas de mensajería, la nota del examen será 0.

Entrega:

Al terminar el examen, se debe subir un único fichero comprimido .tar.gz llamado DNI_NOMBRE_APELLIDO.tar.gz a la actividad de eGela, donde DNI será vuestro número de DNI sin letra. Ejemplo: 12345678_Unai_Lopez.tar.gz.

El enunciado de cada uno de los ejercicios del examen indica qué ficheros se deben entregar y con qué nombre. El fichero comprimido .tar.gz debe contener los ficheros de solución a los ejercicios resueltos.

NOTA: Este documento es un modelo de examen. El examen tendrá un número diferente de ejercicios.

Administración de Sistemas - Curso 2023 / 2024

Ejercicio 1 (X puntos)

El repositorio de código <https://github.com/ulopeznovoa/AS-CI-CD-ej1> contiene el siguiente código Python: un fichero "wallet.py" con la implementación de una clase Wallet y un fichero "test.py" con los tests unitarios asociados.

Crear un repositorio privado en vuestra cuenta de GitHub que contenga el código del repositorio anterior (hacer *fork* o clonar). En este repositorio, crear un fichero Dockerfile para empaquetar el código, sus pasos deben ser:

- Utilizar como base la imagen "python:slim".
- Utilizar como directorio de trabajo /code.
- Copiar los ficheros de código Python en la carpeta /code.

A continuación, crear un *workflow* de GitHub Actions con las siguientes características:

- Nombre: examen-workflow
- Se debe disparar al hacer Push en una rama llamada "nueva-caracteristica".
- Se debe ejecutar en un entorno "ubuntu-latest".

El *workflow* debe hacer lo siguiente:

- 1) Descargar el código.
- 2) Utiliza la acción "actions/setup-python@v4" (<https://github.com/actions/setup-python>) para configurar un entorno Python de versión 3.10.
- 3) Ejecutar los siguientes comandos.
 - a. pip install pytest
 - b. pytest test.py
- 4) Utiliza la acción "docker/build-push-action@v5" (<https://github.com/marketplace/actions/build-and-push-docker-images>) para construir una imagen usando el Dockerfile. No es necesario subir la imagen a ningún registro.

Entregar:

- El fichero Dockerfile y el fichero YAML con el workflow.
- Una captura de pantalla de la sección "Actions" del repositorio que muestre la correcta ejecución del workflow.

Administración de Sistemas - Curso 2023 / 2024

Ejercicio 2 (Y puntos)

Realizar las siguientes operaciones en Elasticsearch utilizando su API REST:

- Crear un índice "usuarios" con las siguientes características:
 - Campo "nombre", tipo text.
 - Campo "apellido", tipo text.
 - Campo "edad", tipo integer.
- Añadir los siguientes elementos al índice:

ID	Nombre	Apellido	Edad (años)
10	Jon	Blanco	27
11	Amaia	Lopez	28
12	Hodei	Bilbao	33

- Realizar las siguientes búsquedas:
 - Obtener los datos de los usuarios cuyo apellido empiece por la letra B.
 - Recuperar los datos de "Amaia Lopez", pero utilizando "Loepz" como término de búsqueda.

Entregar las consultas REST utilizadas para completar estas tareas. Elegir una de las dos siguientes formas para entregar las consultas:

- Un fichero llamado "Ej2_comandos.txt" con los comandos *curl* utilizados para estas tareas. Cada comando puede estar acompañado de una breve explicación, si fuese necesario.
- Rellenar el documento "Hoja de respuestas para consultas REST" disponible en eGela. Entregarlo con el nombre "Ej2_comandos.doc".

Ejercicio 3 (Z puntos)

Configurar un pipeline Logstash con las siguientes características:

- Recibir datos en formato JSON a través de conexiones HTTP en el puerto 9870.
- Eliminar el campo "ciudad" de cada objeto JSON recibido.
- Escribir cada objeto JSON en el índice "usuarios" de Elasticsearch recién creado.

Entregar el fichero de configuración de Logstash. El nombre del fichero debe ser "Ej3_logstash.conf". El fichero puede contener comentarios explicativos, si fuese necesario.