

 POLITÉCNICO COLOMBIANO JAIME ISAZA CADAVID	POLITÉCNICO COLOMBIANO JAIME ISAZA CADAVID FORMATO PARA REALIZACIÓN DE ACTIVIDADES EVALUATIVAS		Código: FD-GC195		
	FACULTAD DE INGENIERÍAS PROGRAMA: Tecnología en Sistematización de Datos Propedéuticos Con la Técnica Profesional		Versión: 02		
ASIGNATURA	CÓDIGO:ING00846	NOMBRE: Construcción de Elementos de Software 3			
PROFESOR:	OSCAR MESA		FECHA: 23-11-2023		
TIPO DE EVALUACIÓN	TALLER	QUIZ	PARCIAL	FINAL	OTRO – CUÁL?
		X			

NOMBRE: Diego Alejandro Muñoz Robayo CÉDULA: 1023622903 NOTA: _____

El siguiente examen cuenta con dos puntos que deben ser subidos a un repositorio en GitHub con el nombre “Mi Momento Valorativo”, en el repositorio debe estar el archivo de Word con la solución del primer punto y el segundo punto debe ser los archivos del proyecto de Java Web que debe estar en la raíz de repositorio.

1. VALOR 30%

1 - ¿Cuál es el propósito principal de HTTP en el desarrollo de software?

- a. Manipulación de archivos
- b. Transferencia de datos en la web
- c. Seguridad de red
- d. Compilación de código

2 - ¿Qué comando de Linux se utiliza para cambiar los permisos de un archivo?

- a. chmod
- b. chown
- c. ls
- d. cp



3 - ¿Qué hace el comando "git commit" en Git?

- a. Crea una nueva rama
- b. Guarda los cambios en el repositorio**
- c. Muestra el historial de confirmaciones
- d. Fusiona ramas

4 - En Java, ¿cuál es el propósito de un constructor?

- a. Declarar variables
- b. Inicializar objetos**
- c. Crear funciones
- d. Definir clases

5 - ¿Cuál es la diferencia entre HTTP y HTTPS?

- a. HTTP es más rápido
- b. HTTPS es más seguro debido al cifrado**
- c. No hay diferencia
- d. HTTPS no se utiliza en el desarrollo web

6 - ¿Cuál de los siguientes es un protocolo de comunicación utilizado comúnmente para API RESTful?

- a. SOAP
- b. HTTP**
- c. FTP
- d. UDP

7 - ¿Qué es el polimorfismo en programación orientada a objetos?

- a. Capacidad de una clase para heredar de múltiples clases
- b. Capacidad de una clase para tener múltiples métodos con el mismo nombre**



- c. Capacidad de una clase para ocultar datos
- d. Capacidad de una clase para acceder a datos privados de otra clase

8 - ¿Cuál de las siguientes colecciones en Java utiliza una estructura de datos tipo "pila"?

- a. ArrayList
- b. LinkedList
- c. HashSet
- d. Stack

9 - ¿Qué hace el comando "git commit" en Git?

- a. Crea una nueva rama
- b. Guarda los cambios en el repositorio
- c. Muestra el historial de confirmaciones
- d. Fusiona ramas

10 - ¿Qué es un objeto en el contexto de la programación orientada a objetos (POO)?

- a. Una instancia de una clase que puede tener atributos y métodos.
- b. Un archivo que almacena código fuente.
- c. Una función específica en un programa.
- d. Un tipo de dato primitivo.

11- ¿Cuál es el propósito de la sentencia "super" en Java?

- a. Invocar al constructor de la clase base.

- b. Acceder a variables locales de un método.
- c. Declarar una variable estática.
- d. Finalizar la ejecución de un bucle.

12 - ¿Qué es un servlet en el contexto de desarrollo web con Java?

- a. Una biblioteca para el manejo de JSON.
- b. Un componente de interfaz de usuario.
- c. Un programa que se ejecuta en el servidor web y gestiona solicitudes HTTP.
- d. Una herramienta para el control de versiones.

13 - ¿Cuál es el propósito principal de HTTP en el desarrollo de software?

- a. Manipulación de archivos
- b. Transferencia de datos en la web
- c. Seguridad de red
- d. Compilación de código

14 - ¿Qué es un objeto JSON válido?

- a. {nombre: "John", edad: 25, ciudad: "New York"}
- b. ["John", 25, "New York"]
- c. 'nombre: John, edad: 25, ciudad: New York'
- d. (nombre: "John", edad: 25, ciudad: "New York")

2. VALOR 70%

Nota: Se debe colocar dentro del proyecto el script sql, que cuente mínimamente con 10 insert. De igual forma, también se tendrá muy en



cuenta emplear todo lo visto hasta el momento, tal como herencia, interfaces, DTO, etc.

Supongamos que estás construyendo un sistema para gestionar usuarios, cada usuario tiene un nombre, un correo electrónico y una contraseña.

1. Obtener todos los usuarios (GET):
 - a. Crea un endpoint que devuelva la lista completa de usuarios registrados en el sistema.
 - b. Utiliza un método HTTP GET para recuperar la información de todos los usuarios almacenados en la tabla de la base de datos.
2. Obtener información detallada de un usuario (GET):
 - a. Implementa un endpoint que permita obtener información detallada de un usuario mediante su identificador único.
 - b. Utiliza un método HTTP GET y un parámetro en la URL para identificar al usuario.
 - c. Recupera la información del usuario almacenada en la base de datos.
3. Eliminar un usuario por su identificador único (DELETE):
 - a. Implementa un endpoint que permita eliminar un usuario del sistema mediante su identificador único.
 - b. Utiliza un método HTTP DELETE y el identificador único del usuario para realizar la eliminación.
 - c. Asegúrate de actualizar la tabla de la base de datos después de eliminar al usuario.
4. Validación de credenciales (POST):
 - a. Desarrollar un endpoint para verificar la duplicidad de usuarios por correo. El endpoint debe indicar si ya existe en la base de datos un usuario con el correo que se le envíe (devuelve un mensaje de error adecuado).

