XX DE DICIEMBRE DE 2021 - [COM XX]

# Testing I

# **Examen integrador**

Les pedimos que lean atentamente las siguientes consignas y respondan a las preguntas de acuerdo a lo solicitado.

**No se aceptarán links de Drive, solo documentos adjuntos. Caso contrario, el examen no será considerado para su corrección.**

Nota aclaratoria: al enviar el formulario con el adjunto se debe esperar la confirmación del profesor **antes de salir de la sala de Zoom** para garantizar que se recibió correctamente para posterior corrección. Caso contrario, no se recibirá la evaluación y el alumno deberá recuperar esta instancia de evaluación. **Solo se recibirá 1 (un) documento por alumno**.

**Duración**: 1 hora 30 minutos.

**Nombre y Apellido:** Juan Manuel Arango Tobón

Parte teórica

1. Mencionar **2 características** que debe tener **un buen caso de prueba**.

(Sugerencia: utilizar 2 a 5 renglones)

1. ¿Qué **ventajas** presentan las **pruebas automatizadas**?

(Sugerencia: utilizar 2 a 5 renglones)

1. Explicar la diferencia entre una **prueba funcional** y una **prueba no funcional**.

(Sugerencia: utilizar 2 a 5 renglones)

1. Explicar la **técnica de prueba de caja negra**.

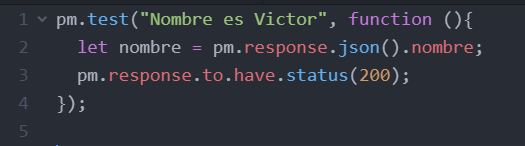
(Sugerencia: utilizar 2 a 5 renglones)

1. Explicar el **nivel de prueba de componente**.

(Sugerencia: utilizar 2 a 5 renglones)

Parte práctica

1. ¿Este **test de Postman** es correcto para validar si el contenido devuelto es igual al esperado? Justificar tu respuesta.



1. Continuamos trabajando con nuestra app **Comida Ya!** Esta se conecta con un servicio back-end. Si realizo una petición **GET en Postman** a la siguiente URL: <https://ctd-api-resto.herokuapp.com/v1/products>. ¿Qué resultado arroja y por qué?
2. Detallar **1 caso de prueba** que aplicarías en la página de [Comida Ya!](https://ctd-app-resto.herokuapp.com/admin/dashboard/), solo explicando su descripción (**no utilizar el template**).
3. Mencionar **1 defecto** que encuentres en el sitio de [Comida Ya!](https://ctd-app-resto.herokuapp.com/admin/dashboard/) (**no utilizar el template**).
4. Si estoy trabajando con Jest y quiero validar que el resultado devuelto sea **true**. ¿Qué matcher puedo utilizar? Dar un ejemplo de un posible test.

**SOLUCIÓN:**

* **Parte teórica:**

1. Mencionar **2 características** que debe tener **un buen caso de prueba**.
   1. Se presenta como un documento escrito que brinda la información necesaria sobre qué y como se debe probar algo en el software el cual está enfocado en cumplir los requisitos de un cliente.
   2. Un caso de prueba debe ser simple y concreto para que sea comprensivo y entendible por alguien que no sea el autor y que lo llegue a ejecutar
2. ¿Qué **ventajas** presentan las **pruebas automatizadas**?

Cuando se habla de pruebas automatizadas se entiende que es el proceso mediante el cual se realizan varias pruebas sin necesidad de que hay una intervención humana y que se realizan mediante una herramienta generalmente de terceros que permite la automatización de estos procesos.

Las ventajas que se presentan con este tipo de pruebas en el desarrollo de software se pueden aplicar a varios aspectos como el costo de produccion o los recursos y tiempos que se disponen para las pruebas, entre algunas de las ventajas están:

* + Reducir costos en las pruebas
  + Se pueden ejecutar pruebas más complejas
  + Ahorro en el tiempo de ejecución
  + Se pueden reducir los errores por parte del tester

1. Explicar la diferencia entre una **prueba funcional** y una **prueba no funcional**.

Una prueba funcional se encarga de probar la funcionalidad que se implementa en un sistema o en un componente en si mismo, estas verifican que se cumplan con los requisitos especificados, mientras que una prueba no funcional se encarga de cosas totalmente contrarias que no tengan nada que ver con la entrega final de un componente o sistema de software, como la seguridad, compatibilidad, mantenimiento etc. Sin embargo, se complementan entre si haciéndolas primordial para el desarrollo de software

1. Explicar la **técnica de prueba de caja negra**.

Una técnica de pruebas de caja negra es una técnica en la cual se verifica la funcionalidad de un componente o un sistema en si mismo, sin tener en cuenta la estructura interna con la que esta compuesta, es decir el código, tampoco se tienen en cuenta posibles escenarios o casos en los que se pueda implementar.

1. Explicar el **nivel de prueba de componente**.

En el desarrollo de software, una prueba de tipo de componente es una técnica que se basa en probar o testear cada componente de un sistema de manera separada o individualmente, y esto sin tener en cuenta la integración con el sistema final en si mismo o con otros componentes, la interpretación de estas pruebas puede variar según la arquitectura de un sistema, siendo desde pequeños fragmentos de código hasta módulos enteros.

El resultado que se arroja por parte de la petición es un error **401 unauthorized**, puesto que, para poder acceder a los datos de la consulta en el URL, se debe haber logrado en la aplicación, y no están libres para el público.

Un caso de prueba que se podría llevar a cabo es la prueba de componente a nivel de ingreso de formulario, donde se pueda verificar que se ingresan los datos correctos o de forma correcta, por ejemplo verificar que haya un límite establecido de caracteres, también verificar que no se puedan enviar inyecciones SQL, y que los datos sean enviados con el método REST correcto, en este caso POST

sfsf