

Práctica 5 - Login completamente funcional

Ingeniería de Software 2024-1

Rogelio Alcantar Arenas
rogelio-aa@ciencias.unam.mx

Erick Martínez Piza
ermarp@ciencias.unam.mx

Luis Fernando Yang Fong Baeza
fernandofong@ciencias.unam.mx

Valeria García Landa
vale.garcia.landa@gmail.com

Francisco Valdés Souto
fvaldes@ciencias.unam.mx

29 de Septiembre 2023

1. Introducción

El propósito de esta práctica es tener una base para cualquier sistema con una integración completa en arquitectura de microservicios, consumiéndolos vía una aplicación de Front-End, escrita en React y una aplicación de Back-End, escrita en Flask, recordando que los endpoints en esta ocasión, deberán de responder con objetos de tipo **JSON** y la aplicación de Front-End deberá de interpretar ahora estos objetos para convertirlos en componentes de React de acuerdo a la información proporcionada.

Por esta práctica, la entrega deberá de ser en equipos (las especificaciones de entrega se definen al final de este documento).

2. Desarrollo

Para esta práctica se les ha dado el código completamente funcional de una aplicación Back-End con una vista para entender a fondo el funcionamiento de la misma, así mismo, se les ha entregado también el código para que puedan consultar via **Postman** los endpoints que responden un formato **JSON**.

Además se les entrega también una aplicación de React que ya consume algunos de estos endpoints y los transforma en componentes para mayor entendimiento del consumo los microservicios.

El equipo deberá de programar un login completamente funcional para la aplicación que será la base de su sistema web, de manera que no es obligatorio que el login sea para la aplicación **Clonbuster** puesto que este deberá de ser el primer caso de uso del sistema web a desarrollar durante el resto del curso implicando que no necesariamente deberán de hacer uso de la base de datos que se ha trabajado durante todas las prácticas.

Es responsabilidad de los alumnos investigar y arreglar el código proporcionado con el propósito de seguir trabajando sus herramientas de desarrollo.

2.1. Punto extra

El punto extra de esta práctica es para el equipo que logre implementar un caso de uso para reestablecer la contraseña, no necesariamente tiene que incluir toda la aplicación (Front y Back) pero el usuario debe de poder reestablecer la contraseña a su gusto para que el punto extra cuente como válido y se le subirá a todo el equipo.

2.2. Especificaciones de requerimientos

Para que este caso de uso se considere completamente funcional, se deberán de cubrir los 3 flujos que puedan ocurrir y mostrarle al usuario un error específico respecto a qué pasó y como corregirlo si es que fue un error del usuario, tal y como se muestra en la aplicación de Flask pura, si existe dicho correo, si la contraseña es o no correcta, etc. combinando Back-End y Front-End.

2.3. Descargables

El código fuente está disponible en el repositorio oficial de Github en la cuenta del ayudante del laboratorio (consultar el canal de recursos vía Discord)

3. Entregables

A diferencia de las otras prácticas y por cuestiones de espacio, no se subirán los proyectos en Moodle, si no que ambos proyectos se deberán de subir al Github oficial del equipo sin embargo, no deberán de estar en el mismo repositorio, ambos deben de tener su propio repositorio y un README especificando los nombres de los integrantes, el nombre del equipo, el logo, las tecnologías, instalación y cómo se ejecuta dicho proyecto.

Para la entrega en Moodle, simplemente deberá de ser marcado como entregado sin subir nada pero se debe de hacer la invitación a los 4 ayudantes a ver el repositorio del equipo.

4. Fecha de Entrega y Criterios de Evaluación

Para esta práctica se evaluarán los siguientes puntos de manera equitativa (cada rubro vale lo mismo):

1. Conexión a la Base de Datos desde el Front-End consumiendo los servicios de la aplicación Back-End, no se permitirá una conexión directa de la aplicación Front-End con la base de datos.
2. Validación de la contraseña del usuario utilizando las funciones del paquete **CryptoUtils** que se provee para el login.
3. Mostrar la sesión del usuario con una interfaz en caso de ser exitoso el inicio de sesión en la aplicación Front-End.
4. Cierre de sesión exitoso, se considera exitoso si después de haber cerrado la sesión e intentar reingresar a la vista, el sistema vuelve a pedir que se inicie sesión.
5. Si ocurre un flujo alternativo, este deberá de ser mostrado al usuario y cómo corregirlo (contraseña incorrecta, correo no existente, etc...)
6. Si ocurre un flujo excepcional, simplemente indicarle al usuario un error.

La fecha de entrega es para el 26 de Octubre antes de las 23:59:59.