

PRÁCTICA EVALUABLE 6



REALIZADO POR:

MIGUEL ANGEL GONZALEEZ SAEZ

ÍNDICE

TAREA 1 FLUX	3
TAREA 2 REDUX	4
TAREA 3 MOBX	4

TAREA 1 FLUX

Flux es una arquitectura la cual implementa una serie de pautas las cuales hacen que las aplicaciones se escalen de forma sencilla y ordenada. Se centra en la unidireccionalidad de los datos, lo cual influye en que los datos solo viajen en una sola direccion atraves de la aplicacion, esto implica que un controlador puede cambiar el estado de otro controlador y propagar esa informacion, en una aplicacion pequeña puede hacer que esto resulte conveniente pero cuando crece la aplicacion y la informacion viaja en las dos direcciones hace que aumente la posibilidad de que algun controlador se desactualice.

Por lo que flux al obligar la direccion de los datos en una sola direccion elimina la posibilidad de que algun controlador rompa el sistema, por lo que esto facilita el mantenimiento y la escalabilidad del código al tener un flujo de datos predecible y controlado.

Flux es una arquitectura basada en eventos los eventos son los siguientes:

Flujo de eventos: Flux se basa en el concepto de eventos para manejar las interacciones del usuario y las actualizaciones del estado de la aplicación.

Acciones: Representan eventos que ocurren en la aplicación, como clics de botón o solicitudes de datos al servidor.

Despachador: Actúa como un centro de distribución de acciones, enviando cada acción a las tiendas relevantes.

Tiendas: Almacenan el estado de la aplicación y actualizan sus datos en respuesta a acciones específicas.

Vistas: Muestran los datos almacenados en las tiendas y se actualizan automáticamente cuando los datos cambian.

Flux proporciona beneficios significativos en el mundo real. Gracias a su estructura no compromete el comportamiento de otras características por lo que es muy escalable. Además, en entornos de desarrollo colaborativo, la gran facilidad que tiene para la separación de contenidos es muy fácil por lo que para grandes trabajos resulta muy sencillo de utilizar. Finalmente, Flux simplifica el mantenimiento del código gracias a su flujo de datos unidireccional y la clara división de responsabilidades, lo que facilita la depuración y el mantenimiento continuo del sistema.

TAREA 2 REDUX

Redux es una biblioteca de gestión de estado para aplicaciones JavaScript, especialmente útil para aplicaciones de una sola página o aplicaciones web con una interfaz de usuario compleja. Redux proporciona un enfoque claro para manejar el estado de la aplicación.

En el corazón de Redux está el concepto de un Store, el cual almacena todos los estados de la misma. Las acciones son eventos que realizan un cambio en el estado y son usadas por componentes de la interfaz de usuario u otras partes de la aplicación.

Los reducers son funciones puras que especifican cómo el estado de la aplicación cambia en respuesta a una acción, tomando el estado actual y una acción como argumentos y devolviendo el nuevo estado de la aplicación. Este flujo de datos unidireccional permite que los datos fluyan de forma coherente a través de la aplicación, asegurando una gestión clara y predecible del estado.

El middleware en Redux proporciona una extensión del comportamiento, permitiendo acciones asíncronas, registros y modificaciones de acciones antes de que lleguen a los reducers, entre otras cosas.

Aunque Redux se utiliza comúnmente con React, también es compatible con otras bibliotecas como Angular o Vue.js. En conjunto, Redux ayuda a gestionar el estado de la aplicación de manera más predecible, facilitando el seguimiento de los cambios y la depuración del código.

TAREA 3 MOBX

MobX, es una librería de gestión de estado para JavaScript que simplifica el estado en aplicaciones mediante un enfoque reactivo. Esto se basa en la idea de que los datos en la aplicación se declaran como "observables", lo que significa que cualquier cambio de los datos se notifica automáticamente a los componentes dependientes, lo que desencadena una actualización de la interfaz de usuario de manera transparente.

El objetivo principal de MobX es facilitar la creación de aplicaciones complejas al reducir la carga cognitiva necesaria para mantener y actualizar el estado de la aplicación. Esto se logra gracias a su modelo reactivo, donde los cambios en los datos se propagan automáticamente a través de la aplicación, eliminando la necesidad de manejar manualmente eventos o suscripciones para actualizar la interfaz de usuario.

MobX se integra de manera fluida con diferentes frameworks y bibliotecas, como React, Vue.js y Angular, lo que lo hace altamente adaptable. Además, su sintaxis declarativa permite escribir de manera más legible y mantenible.

En el mundo real, MobX es especialmente útil en aplicaciones con interfaces dinámicas y datos cambiantes, donde la gestión del estado es fundamental para la experiencia del usuario. La forma en la que manejan los datos entre componentes lo hace una opción sólida para proyectos de gran escala.