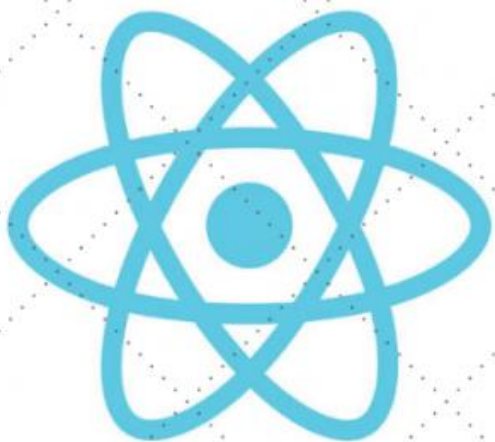


Practica 6 - Estudio de Patrones de Software y Librerías JavaScript



Índice

1. Introducción
2. Tarea 1: Estudio del patrón de software Flux
 - 2.1 Origen
 - 2.2 Objetivo
 - 2.3 Funcionamiento
 - 2.4 Aplicabilidad en el mundo real
 - 2.5 Conclusiones de la Tarea 1
3. Tarea 2: Estudio de la librería Redux
 - 3.1 Origen
 - 3.2 Objetivo
 - 3.3 Funcionamiento
 - 3.4 Aplicabilidad en el mundo real
 - 3.5 Conclusiones de la Tarea 2
4. Tarea 3: Estudio de la librería Mobx
 - 4.1 Origen
 - 4.2 Objetivo
 - 4.3 Funcionamiento
 - 4.4 Aplicabilidad en el mundo real
 - 4.5 Conclusiones de la Tarea 3
5. Bibliografía

1. Introducción

En esta memoria se presenta un estudio sobre los patrones de software Flux y las librerías JavaScript Redux y Mobx. Se va a examinar el origen, los objetivos, el funcionamiento y la aplicabilidad en el mundo real de cada uno de estos conceptos.

2. Tarea 1: Estudio del patrón de software Flux

2.1 Origen

El patrón de software Flux fue desarrollado por Facebook en 2014 para abordar el manejo de datos en aplicaciones web de una manera más predecible y fácil de entender.

2.2 Objetivo

El objetivo principal de Flux es manejar de manera unidireccional el flujo de datos en una aplicación, lo que facilita el manejo de estados y la sincronización de vistas en aplicaciones complejas.

2.3 Funcionamiento

En Flux, los datos fluyen en una sola dirección a través de la aplicación, lo que significa que cualquier cambio en los datos desencadena una serie de eventos que actualizan las vistas de manera consistente.

2.4 Aplicabilidad en el mundo real

Flux es útil en aplicaciones web complejas donde se necesita un manejo eficiente de estados y sincronización de vistas. Es particularmente útil en aplicaciones con interfaces de usuario dinámicas y datos cambiantes.

2.5 Conclusiones de la Tarea 1

Flux ofrece una solución efectiva para manejar el flujo de datos en aplicaciones web, proporcionando un enfoque claro y predecible para el manejo de estados y vistas.

3. Tarea 2: Estudio de la librería Redux

3.1 Origen

Redux es una librería de manejo de estados para aplicaciones JavaScript, desarrollada por Dan Abramov y Andrew Clark en 2015, inspirada en el patrón de Flux.

3.2 Objetivo

Redux tiene como objetivo centralizar y gestionar el estado de la aplicación de una manera predecible y fácilmente comprensible.

3.3 Funcionamiento

En Redux, el estado de toda la aplicación se almacena en un único objeto de estado, llamado *store*. Las vistas se actualizan en respuesta a cambios en este estado, y las actualizaciones del estado solo pueden ocurrir mediante acciones.

3.4 Aplicabilidad en el mundo real

Redux es ampliamente utilizado en aplicaciones web y móviles, especialmente en aplicaciones React, debido a su simplicidad y previsibilidad en la gestión de estados.

3.5 Conclusiones de la Tarea 2

Redux ofrece una solución eficiente y escalable para el manejo de estados en aplicaciones JavaScript, proporcionando una arquitectura clara y fácil de entender.

4. Tarea 3: Estudio de la librería Mobx

4.1 Origen

Mobx es una librería de manejo de estados para aplicaciones JavaScript, desarrollada por Michel Weststrate en 2015, que se centra en la programación reactiva y orientada a objetos.

4.2 Objetivo

El objetivo principal de Mobx es hacer que el manejo del estado sea más sencillo y natural mediante el uso de la programación reactiva.

4.3 Funcionamiento

En Mobx, los estados son representados como objetos observables, que notifican automáticamente a las vistas cuando cambian. Esto permite que las vistas se actualicen de manera automática cuando los datos cambian.

4.4 Aplicabilidad en el mundo real

Mobx es adecuado para aplicaciones que requieren un manejo reactivo y orientado a objetos de estados, especialmente en aplicaciones con una lógica de negocio compleja o en tiempo real.

4.5 Conclusiones de la Tarea 3

Mobx ofrece una alternativa interesante para el manejo de estados en aplicaciones JavaScript, permitiendo una programación más intuitiva y reactiva.

5. Bibliografía

<https://keepcoding.io/blog/aprende-a-usar-mobx-paso-a-paso/>

<https://mobx.js.org/README.html>

<https://carlosazaustre.es/como-funciona-flux>

<https://docs.coreflux.org/flux-lang/>

<https://es.redux.js.org>

<https://redux.js.org>

<https://oxygenacademy.es/que-es-redux-y-por-que-deberias-implementarlo-en-tus-aplicaciones-react/>