



LECTURAS 10 y 11

UNIDAD 5 Implementación.

Materia

**DESARROLLO, APLICACIÓN Y CONSUMO
DE SISTEMAS**

Presenta: Gabriela Esmeralda Sabas Zavala

Docente:

Eduardo Flores Gallegos



INDICE

LECTURAS 10 Y 11..... 1

LECTURA 10 3

LECTURA 11..... 4

REFERENCIAS ¡Error! Marcador no definido.

LECTURA 10 Cómo entender el código JS asíncrono de una manera menos dolorosa.

Cuando comenzamos nuestro desarrollo en JS se tiende a ordenar archivos por carpetas y subdividir todo para que se vea de una mejor manera y tener código limpio sin embargo llega un punto donde tenemos que utilizar código asincronismo, el cual si no se maneja de manera correcta nos puede generar código basura, así como fallos y errores en donde no los había. Sin embargo, este tipo de código se puede adaptar de diferentes maneras para ayudar a obtener los resultados deseados.

Todo se basa en una misma base para los lenguajes de programación,

Santo grail de la separación de código: podemos mover el almacenamiento y la lógica a otro archivo de código, que tratará de forma centralizada con los datos. Este archivo, el servicio, será responsable de los problemas y mecanismos, como la sincronización con el almacenamiento local. Al contrario, los componentes solo servirán como partes de interfaz

JavaScript es un lenguaje muy elegante, una de sus características peculiares es tratar las funciones como cualquier otro valor. Usando la descripción formal, "las funciones son ciudadanos de primera clase". Esto significa que cualquier función se puede asignar a una variable o pasar como un parámetro a otra función.

Podemos definir una función que actualizaría nuestra interfaz de usuario, y luego pasarla a otra parte del código, donde se llamará.

Usando este mecanismo, podemos tomar nuestro renderCities y de alguna manera pasarlo al servicio de datos. Allí, se invocará cuando sea necesario: después de todo, el servicio sabe con precisión cuándo se deben transferir los datos a los componentes, no al revés.

No importa qué método de resolución de problemas asíncronos usará, siempre habrá una variación del mismo principio: algo se suscribe, algo se publica. [1]

LECTURA 11 Red Hat insiste: el futuro de la tecnología es de código abierto

En 2008 a Jim Whitehurst se hizo cargo de la empresa tecnológica Red Hat la cual Microsoft había desarrollado el sistema operativo Linux que comenzó a ser la mayor amenaza para y una empresa de Hackers, 11 años después obtuvo 5 veces mas ingresos que lo obtenido en los últimos 3 años en 2014 se asoció con Microsoft para crear Azure una plataforma que almacena y ayuda al desarrollo de aplicaciones empresariales.

Jim afirma que la empresa creció gracias a la implementación del código abierto, este comenzó con Red Hat ya que los desarrolladores querían crear una forma en como ayudar a otras personas a crear su propio software, también valora mucho el que cualquier persona pueda crear aplicaciones o programas con sus propias ideas.

La empresa cree que todos deben de tener fundamentos básicos de desarrollador.

Red Hat, que pasó de ser la iniciativa de un grupo de hackers a una compañía que cotiza en la Bolsa de Nueva York (siglas RHT), obtuvo ingresos por 814 mdd, 20% más que en el mismo periodo del año pasado.

Para el caso de México, país en el cual Red Hat está presente desde 2007, Whitehurst prevé, además de seguir relacionándose con la mayor cantidad posible de compañías para vender sus soluciones de open source, colocar al país como plataforma desde la cual seguir acercándose a Latinoamérica. [2]

Bibliografía

- [1] H. Zub, «ITNEXT,» 3 10 2018. [En línea]. Available: <https://itnext.io/why-every-beginner-front-end-developer-should-know-publish-subscribe-pattern-72a12cd68d44>. [Último acceso: 10 12 2018].
- [2] A. Medina, «Forbes,» Forbes, 10 10 2018. [En línea]. Available: <https://www.forbes.com.mx/trabaja-red-hat-por-el-futuro-de-codigo-abierto/>. [Último acceso: 10 12 2018].