



UNIVERSIDAD
PRIVADA
DEL NORTE

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

Trabajo de Campo 04

Soluciones Web y Aplicaciones distribuidas

Integrantes Grupo:

- Cabrera Rodriguez, Willard Manuel

Lima - Perú
2020

PROPUESTOS

1. Redacte un cuadro de Node.js que lo defina, menciona sus principales características e historial de versiones.

Node Js
¿Qué es?
Node.js® es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, npm, es el ecosistema más grande de librerías de código abierto en el mundo.
Características
Asíncrono y controlado por eventos: todas las API de la biblioteca Node.js son asíncronas, es decir, sin bloqueo. Esencialmente significa que un servidor basado en Node.js nunca espera que una API devuelva datos. El servidor pasa a la siguiente API después de llamarlo y un mecanismo de notificación de Events of Node.js ayuda al servidor a obtener una respuesta de la llamada API anterior.
Muy rápido: al estar construido en el motor JavaScript V8 de Google Chrome, la biblioteca Node.js es muy rápida en la ejecución de código.
Sin almacenamiento en búfer: las aplicaciones Node.js nunca almacenan en búfer ningún dato. Estas aplicaciones simplemente generan los datos en fragmentos.
Procesos en un solo hilo pero altamente escalable: Node.js utiliza un modelo de una solo hilo con bucle de eventos. El mecanismo de eventos ayuda al servidor a responder sin bloqueos y hace que el servidor sea altamente escalable en comparación con los servidores tradicionales que crean hilos limitados para manejar las solicitudes. Node.js utiliza un solo programa de subprocesos y el mismo programa puede proporcionar servicio a un número mucho mayor de solicitudes que los servidores tradicionales como el Servidor Apache HTTP.
Licencia basada en código abierto – Node.js se distribuye gracias a una licencia MIT.

Ventajas	Desventajas
OpenSource: sin duda, hoy el software open source ha tenido la evolución más rápida y con mejor aceptación de los usuarios, además que es totalmente gratis su uso, puede descargar el código fuente para analizarlo y crear tu propio proyecto de Open Source basado en este.	Tipos dinámicos: JavaScript permite enviar como parámetro un objeto que no tiene definición, solo abres y cierras llaves {} y dentro pones lo que sea, lo que hace muy complicado saber que valores puedes enviar. Si eres programador JavaScript seguramente me quedarás colgar vivo diciendo que eso es bueno.

Optimización de los recursos: Dada la naturaleza asíncrona de NodeJS nos permite tener una mucho mejor administración de los recursos.	Callbacks: Si bien, como ya analizamos, esto hace que se utilicen mejor los recursos, la verdad es que las callbacks son un dolor de cabeza, ya que tener funciones para cada paso es incómodo y hace muy complicado seguir el código.
Desarrollo ágiles: NodeJS permite crear aplicaciones de una forma simple y rápida, a diferencia de otros lenguajes de programación como Java o C#, que se requiere de una gran cantidad de clases para echar a volar un proyecto.	Refactor: Si alguna vez has hecho un refactor importante en un proyecto de JavaScript seguro me entenderás, pues casi nunca es posible realizar uno de forma segura, pues el tipeado débil y tipos dinámicos hacen que sea complicado para el IDE determinar cómo refactorizar.
FullStack: Dado que JavaScript se puede ejecutar del lado del servidor y del cliente, es posible crear aplicaciones de BackEnd y FrontEnd con una sola tecnología.	Librerías estándar: Esta es a mi ver una de las más fuertes, es verdad que existen librerías para todos los problemas que se te puedan ocurrir, pero el problema es que existen muchas opciones para lo mismo, haciendo complicado elegir entre una u otra. Adicional que cada una avanza a un ritmo separado en vez de unir fuerzas para mejorar una versión estándar.
Modularidad: Como ya lo hablamos, NodeJS es extremadamente modular, lo cual permite utilizar únicamente los módulos requeridos sin traernos nada de más.	Arquitectura: En mi experiencia las aplicaciones NodeJS tiene un nivel de arquitectura muy por debajo de los lenguajes de programación tradicional como Java o C# pues en NodeJS todo se basa en funciones que son enviadas y exportadas de clases. Y es raro ver que se implementan patrones arquitectónicos o patrones de diseño (desde mi experiencia).

Historial de Versiones de Node Js														
16.4.0	16.3.0	16.2.0	16.1.0	16.0.0	15.14.0	15.13.0	15.12.0	15.11.0	15.10.0	15.9.0	15.8.0	15.7.0	15.6.0	15.5.1
15.5.0	15.4.0	15.3.0	15.2.1	15.2.0	15.1.0	15.0.1	15.0.0	14.17.1	14.17.0	14.16.1	14.16.0	14.15.5	14.15.4	14.15.3
14.15.2	14.15.1	14.15.0	14.14.0	14.13.1	14.13.0	14.12.0	14.11.0	14.10.1	14.10.0	14.9.0	14.8.0	14.7.0	14.6.0	14.5.0
14.4.0	14.3.0	14.2.0	14.1.0	14.0.0	13.14.0	13.13.0	13.12.0	13.11.0	13.10.1	13.10.0	13.9.0	13.8.0	13.7.0	13.6.0
13.5.0	13.4.0	13.3.0	13.2.0	13.1.0	13.0.1	13.0.0	12.22.1	12.22.0	12.21.0	12.20.2	12.20.1	12.20.0	12.19.1	12.19.0
12.18.4	12.18.3	12.18.2	12.18.1	12.18.0	12.17.0	12.16.3	12.16.2	12.16.1	12.16.0	12.15.0	12.14.1	12.14.0	12.13.1	12.13.0
12.12.0	12.11.1	12.11.0	12.10.0	12.9.1	12.9.0	12.8.1	12.8.0	12.7.0	12.6.0	12.5.0	12.4.0	12.3.1	12.3.0	12.2.0
12.1.0	12.0.0	11.15.0	11.14.0	11.13.0	11.12.0	11.11.0	11.10.1	11.10.0	11.9.0	11.8.0	11.7.0	11.6.0	11.5.0	11.4.0
11.3.0	11.2.0	11.1.0	11.0.0	10.24.1	10.24.0	10.23.3	10.23.2	10.23.1	10.23.0	10.22.1	10.22.0	10.21.0	10.20.1	10.20.0
10.19.0	10.18.1	10.18.0	10.17.0	10.16.3	10.16.2	10.16.1	10.16.0	10.15.3	10.15.2	10.15.1	10.15.0	10.14.2	10.14.1	10.14.0
10.13.0	10.12.0	10.11.0	10.10.0	10.9.0	10.8.0	10.7.0	10.6.0	10.5.0	10.4.1	10.4.0	10.3.0	10.2.1	10.2.0	10.1.0
10.0.0	9.11.2	9.11.1	9.11.0	9.10.1	9.10.0	9.9.0	9.8.0	9.7.1	9.7.0	9.6.1	9.6.0	9.5.0	9.4.0	9.3.0
9.2.1	9.2.0	9.1.0	9.0.0	8.17.0	8.16.2	8.16.1	8.16.0	8.15.1	8.15.0	8.14.1	8.14.0	8.13.0	8.12.0	8.11.4
8.11.3	8.11.2	8.11.1	8.11.0	8.10.0	8.9.4	8.9.3	8.9.2	8.9.1	8.9.0	8.8.1	8.8.0	8.7.0	8.6.0	8.5.0
8.4.0	8.3.0	8.2.1	8.2.0	8.1.4	8.1.3	8.1.2	8.1.1	8.1.0	8.0.0	7.10.1	7.10.0	7.9.0	7.8.0	7.7.4
7.7.3	7.7.2	7.7.1	7.7.0	7.6.0	7.5.0	7.4.0	7.3.0	7.2.1	7.2.0	7.1.0	7.0.0	6.17.1	6.17.0	6.16.0

6.15.1	6.15.0	6.14.4	6.14.3	6.14.2	6.14.1	6.14.0	6.13.1	6.13.0	6.12.3	6.12.2	6.12.1	6.12.0	6.11.5	6.11.4
6.11.3	6.11.2	6.11.1	6.11.0	6.10.3	6.10.2	6.10.1	6.10.0	6.9.5	6.9.4	6.9.3	6.9.2	6.9.1	6.9.0	6.8.1
6.8.0	6.7.0	6.6.0	6.5.0	6.4.0	6.3.1	6.3.0	6.2.2	6.2.1	6.2.0	6.1.0	6.0.0	5.12.0	5.11.1	5.11.0
5.10.1	5.10.0	5.9.1	5.9.0	5.8.0	5.7.1	5.7.0	5.6.0	5.5.0	5.4.1	5.4.0	5.3.0	5.2.0	5.1.1	5.1.0
5.0.0	4.9.1	4.9.0	4.8.7	4.8.6	4.8.5	4.8.4	4.8.3	4.8.2	4.8.1	4.8.0	4.7.3	4.7.2	4.7.1	4.7.0
4.6.2	4.6.1	4.6.0	4.5.0	4.4.7	4.4.6	4.4.5	4.4.4	4.4.3	4.4.2	4.4.1	4.4.0	4.3.2	4.3.1	4.3.0
4.2.6	4.2.5	4.2.4	4.2.3	4.2.2	4.2.1	4.2.0	4.1.2	4.1.1	4.1.0	4.0.0	0.12.18	0.12.17	0.12.16	0.12.15
0.12.14	0.12.13	0.12.12	0.12.11	0.12.10	0.12.9	0.12.8	0.12.7	0.12.6	0.12.5	0.12.4	0.12.3	0.12.2	0.12.1	0.12.0
0.11.16	0.11.15	0.11.14	0.11.13	0.11.12	0.11.11	0.11.10	0.11.9	0.11.8	0.11.7	0.11.6	0.11.5	0.11.4	0.11.3	0.11.2
0.11.1	0.11.0	0.10.48	0.10.47	0.10.46	0.10.45	0.10.44	0.10.43	0.10.42	0.10.41	0.10.40	0.10.39	0.10.38	0.10.37	0.10.36
0.10.35	0.10.34	0.10.33	0.10.32	0.10.31	0.10.30	0.10.29	0.10.28	0.10.27	0.10.26	0.10.25	0.10.24	0.10.23	0.10.22	0.10.21
0.10.20	0.10.19	0.10.18	0.10.17	0.10.16	0.10.15	0.10.14	0.10.13	0.10.12	0.10.11	0.10.10	0.10.9	0.10.8	0.10.7	0.10.6
0.10.5	0.10.4	0.10.3	0.10.2	0.10.1	0.10.0	0.9.12	0.9.11	0.9.10	0.9.9	0.9.8	0.9.7	0.9.6	0.9.5	0.9.4
0.9.3	0.9.2	0.9.1	0.9.0	0.8.28	0.8.27	0.8.26	0.8.25	0.8.24	0.8.23	0.8.22	0.8.21	0.8.20	0.8.19	0.8.18
0.8.17	0.8.16	0.8.15	0.8.14	0.8.13	0.8.12	0.8.11	0.8.10	0.8.9	0.8.8	0.8.7	0.8.6	0.8.5	0.8.4	0.8.3
0.8.2	0.8.1	0.8.0	0.7.12	0.7.11	0.7.10	0.7.9	0.7.8	0.7.7	0.7.6	0.7.5	0.7.4	0.7.3	0.7.2	0.7.1
0.7.0	0.6.21	0.6.20	0.6.19	0.6.18	0.6.17	0.6.16	0.6.15	0.6.14	0.6.13	0.6.12	0.6.11	0.6.10	0.6.9	0.6.8
0.6.7	0.6.6	0.6.5	0.6.4	0.6.3	0.6.2	0.6.1	0.6.0	0.5.10	0.5.9	0.5.8	0.5.7	0.5.6	0.5.5	0.5.4
0.5.3	0.5.2	0.5.1	0.5.0	0.4.12	0.4.11	0.4.10	0.4.9	0.4.8	0.4.7	0.4.6	0.4.5	0.4.4	0.4.3	0.4.2
0.4.1	0.4.0	0.3.8	0.3.7	0.3.6	0.3.5	0.3.4	0.3.3	0.3.2	0.3.1	0.3.0	0.2.6	0.2.5	0.2.4	0.2.3
0.2.2	0.2.1	0.2.0	0.1.104	0.1.103	0.1.102	0.1.101	0.1.100	0.1.99	0.1.98	0.1.97	0.1.96	0.1.95	0.1.94	0.1.93
0.1.92	0.1.91	0.1.90	0.1.33	0.1.32	0.1.31	0.1.30	0.1.29	0.1.28	0.1.27	0.1.26	0.1.25	0.1.24	0.1.23	0.1.22
0.1.21	0.1.20	0.1.19	0.1.18	0.1.17	0.1.16	0.1.15	0.1.14							

2. Redacte un cuadro de JQuery que lo defina, menciona sus principales características e historial de versiones

JQuery
¿Qué es JQuery?
JQuery es una librería de JavaScript (JavaScript es un lenguaje de programación muy usado en desarrollo web). Esta librería de código abierto, simplifica la tarea de programar en JavaScript y permite agregar interactividad a un sitio web sin tener conocimientos del lenguaje.
Características
Constructor jQuery. El constructor es una función sobrecargada que nos permite acceder a los elementos del DOM correspondientes a la consulta que realizamos.
Trabajar por grupos(Iteración implícita). Los métodos en jQuery pueden realizar consultas mediante iteración implícita. Es decir, buscando múltiples coincidencias y aplicando modificaciones para cada elemento en una única instrucción

Diseñado para realizar consultas a través del DOM. jQuery nos permite realizar consultas CSS, XPath y transversales.

Es un sistema modular donde podemos anidar consultas o tareas de una manera sencilla.

Ventajas	Desventajas
<ul style="list-style-type: none"> -jQuery es flexible y rápido para el desarrollo web. -Tiene una excelente comunidad de soporte. -Viene con licencia MIT y es Open Source. -Tiene Plugins. -Bugs son resueltos rápidamente. -Excelente integración con AJAX. 	<ul style="list-style-type: none"> -jQuery es fácil de instalar y aprender, inicialmente. Pero no es tan fácil si lo comparamos con CSS. -Si jQuery es implementado inapropiadamente como un Framework, el entorno de desarrollo se puede salir de control.

Historial de Versiones(96)			
jQuery Core 3.6.0	jQuery Core 3.5.1	jQuery Core 3.5.0	jQuery Core 3.4.1
jQuery Core 3.4.0	jQuery Core 3.3.1	jQuery Core 3.3.0	jQuery Core 3.2.1
jQuery Core 3.2.0	jQuery Core 3.1.1	jQuery Core 3.1.0	jQuery Core 3.0.0
jQuery Core 2.2.4	jQuery Core 2.2.3	jQuery Core 2.2.2	jQuery Core 2.2.1
jQuery Core 2.2.0	jQuery Core 2.1.4	jQuery Core 2.1.3	jQuery Core 2.1.2
jQuery Core 2.1.1	jQuery Core 2.1.0	jQuery Core 2.0.3	jQuery Core 2.0.2
jQuery Core 2.0.1	jQuery Core 2.0.0	jQuery Core 1.12.4	jQuery Core 1.12.3
jQuery Core 1.12.2	jQuery Core 1.12.1	jQuery Core 1.12.0	jQuery Core 1.11.3
jQuery Core 1.11.2	jQuery Core 1.11.1	jQuery Core 1.11.0	jQuery Core 1.10.2
jQuery Core 1.10.1	jQuery Core 1.10.0	jQuery Core 1.9.1	jQuery Core 1.9.0
jQuery Core 1.8.3	jQuery Core 1.8.2	jQuery Core 1.8.1	jQuery Core 1.8.0
jQuery Core 1.7.2	jQuery Core 1.7.1	jQuery Core 1.7.0	jQuery Core 1.7.0
jQuery Core 1.6.4	jQuery Core 1.6.3	jQuery Core 1.6.2	jQuery Core 1.6.1
jQuery Core 1.6.0	jQuery Core 1.5.2	jQuery Core 1.5.1	jQuery Core 1.5.0
jQuery Core 1.4.4	jQuery Core 1.4.3	jQuery Core 1.4.2	jQuery Core 1.4.1
jQuery Core 1.4.0	jQuery Core 1.3.2	jQuery Core 1.3.1	jQuery Core 1.3.0
jQuery Core 1.2.6	jQuery Core 1.2.5	jQuery Core 1.2.4	jQuery Core 1.2.3
jQuery Core 1.2.2	jQuery Core 1.2.1	jQuery Core 1.2.0	jQuery Core 1.1.4
jQuery Core 1.1.3	jQuery Core 1.1.2	jQuery Core 1.1.1	jQuery Core 1.1.0
jQuery Core 1.0.4	jQuery Core 1.0.3	jQuery Core 1.0.2	jQuery Core 1.0.1
jQuery Core 1.0.0	jQuery Migrate 3.3.2	jQuery Migrate 3.3.1	jQuery Migrate 3.3.0
jQuery Migrate 3.2.0	jQuery Migrate 3.1.0	jQuery Migrate 3.0.1	jQuery Migrate 3.0.0
jQuery Migrate 1.4.1	jQuery Migrate 1.4.0	jQuery Migrate 1.3.0	jQuery Migrate 1.2.1
jQuery Migrate 1.2.0	jQuery Migrate 1.1.1	jQuery Migrate 1.1.0	jQuery Migrate 1.0.0

3. Realice un ejemplo de aplicación de JQuery (código)

https://github.com/willardcabrera/soluciones_web/tree/main/TC04/jQuery

4. Redacte un cuadro de Angular que lo defina, menciona sus principales características e historial de versiones

Angular
¿Qué es Angular?
Angular es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página.
Características
Velocidad y rendimiento
Generación de código: Angular convierte tus plantillas en código altamente optimizado para las máquinas virtuales de JavaScript de hoy en día, ofreciéndote todas las ventajas del código escrito a mano con la productividad de un framework.
Universal: Ejecuta la primera vista de tu aplicación en node.js, .NET, PHP, y otros servidores para renderizado de forma casi instantánea obteniendo solo HTML y CSS. También abre posibilidades para la optimización del SEO del sitio, incluyendo configuración.
División del código: Las aplicaciones de Angular se cargan rápidamente gracias al nuevo enrutador de componentes. Este ofrece una división automática de códigos para que los usuarios solo carguen el código necesario para procesar la vista que solicitan.
Productividad
Plantillas: Permite crear rápidamente vistas de interfaz de usuario con una sintaxis de plantilla simple y potente.
Angular CLI: Las herramientas de línea de comandos permiten empezar a desarrollar rápidamente, añadir componentes y realizar test, así como previsualizar de forma instantánea la aplicación.
IDEs: Obtén sugerencias de código inteligente, detección de errores y otros comentarios en la mayoría de los editores populares e IDEs.
Historia completa del desarrollo
Testing: Utiliza Karma para realizar pruebas unitarias, y Protractor para realizar pruebas end-to-end de forma rápida, estable y eficaz.
Animación: Permite crear animaciones complejas y de alto rendimiento con muy poco código a través de la intuitiva API de Angular.

Accesibilidad: Posee características para crear aplicaciones accesibles con los componentes disponibles para ARIA.

Otros:

DESARROLLAR EN TODAS LAS PLATAFORMAS: Aprenda una forma de crear aplicaciones con Angular y reutilice su código y sus habilidades para crear aplicaciones para cualquier objetivo de implementación. Para web, web móvil, móvil nativo y escritorio nativo.

VELOCIDAD Y RENDIMIENTO: Logre la máxima velocidad posible en la plataforma web hoy y llévelo más allá, a través de los trabajadores web y la representación del lado del servidor. Angular te permite controlar la escalabilidad. Satisface requerimientos enormes de data construyendo modelos de data en RxJS, Immutable.js o algún otro modelo-de-empuje.

HERRAMIENTAS INCREÍBLES: Cree funciones rápidamente con plantillas declarativas simples. Amplíe el lenguaje de la plantilla con sus propios componentes y utilice una amplia gama de componentes existentes. Obtenga ayuda y comentarios inmediatos específicos de Angular con casi todos los IDE y editores. Todo esto se combina para que pueda concentrarse en crear aplicaciones increíbles en lugar de intentar hacer que el código funcione.

AMADO POR MILLONES: Desde el prototipo hasta la implementación global, Angular ofrece la productividad y la infraestructura escalable que admite las aplicaciones más grandes de Google.

VERSIÓN	ESTADO	LIBERADO	EXTREMOS ACTIVOS	FINALIZA LTS
^ 12.0.0	Activo	12/05/2021	12/11/2021	12/11/2022
^ 11.0.0	LTS	11/11/2020	11/05/2021	11/05/2022
^ 10.0.0	LTS	24/06/2020	24/12/2020	24/12/2021
^ 9.0.0	LTS	06/02/2020	06/08/2020	06/08/2021
Las versiones angulares v4, v5, v6, v7 y v8 ya no tienen soporte.				

5. Desarrolle el siguiente ejercicio usando Angular: Desarrolle un programa en angular que permita calcular el sueldo de un trabajador:
Sueldo=horas Trabajadas* precio
Dado que, si las horas trabajadas es mayor a 40, se le da un bono de 300 soles; de ser menores solo recibe el sueldo, en función al número de horas y precio por hora.

https://github.com/willardcabrera/soluciones_web/tree/main/TC04/angular