



Universidad Tecnológica de Puebla

Tecnologías de la información

Alumno: Leonardo Saínos Pérez.

Matricula: UTP0143830

5to

Grupo: F

Profesor: JENNY ORTEGA GARCIA.

Materia: APLICACIONES WEB PARA I 4.0

Tarea : PRODUCTO 1 AVANCE 2.

Fecha de entrega: 29/01/2021.

Fases de la instalación de un Framework de desarrollo Web para Back-End.

Para explicar las fases de la instalación del framework usaremos nodejs como ejemplo y la forma es la siguiente:

Fase 0: Comenzamos a decidir con que framework queremos trabajar de acuerdo a nuestras necesidades, ver opciones que hay entre los tipos de frameworks que hay y finalmente elegir 1.

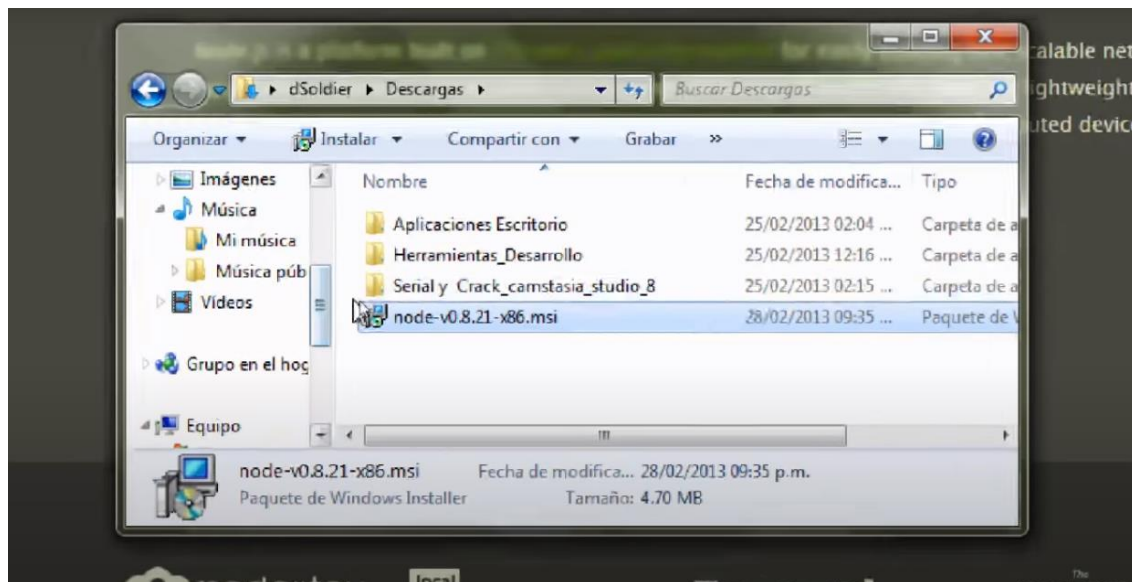


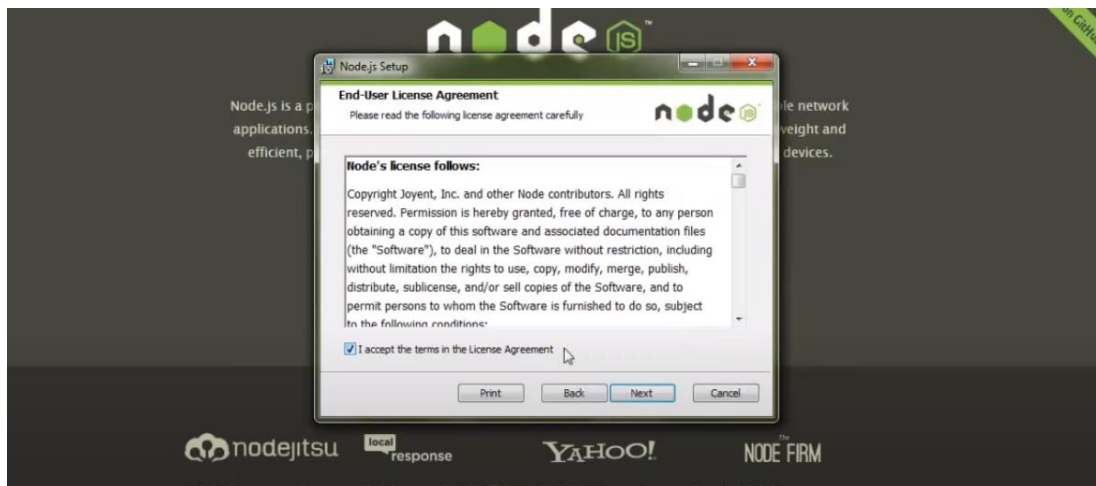
Fase 1: Ir al sitio web oficial donde encontraremos el link de descarga de dicho framework, asegurarse de nuestra versión que vamos a elegir y comenzar con la descarga.

Fase 2: Lo descargamos en donde deseemos guardarlo, posterior a eso ejecutamos el instalador que hemos descargado, para poder completar la instalación de ella .

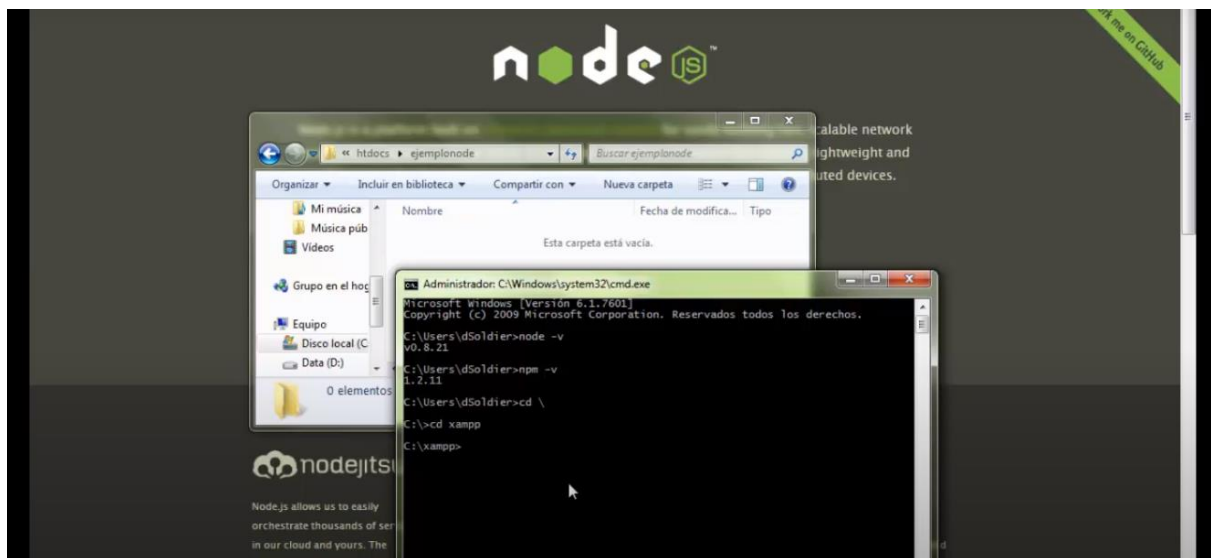


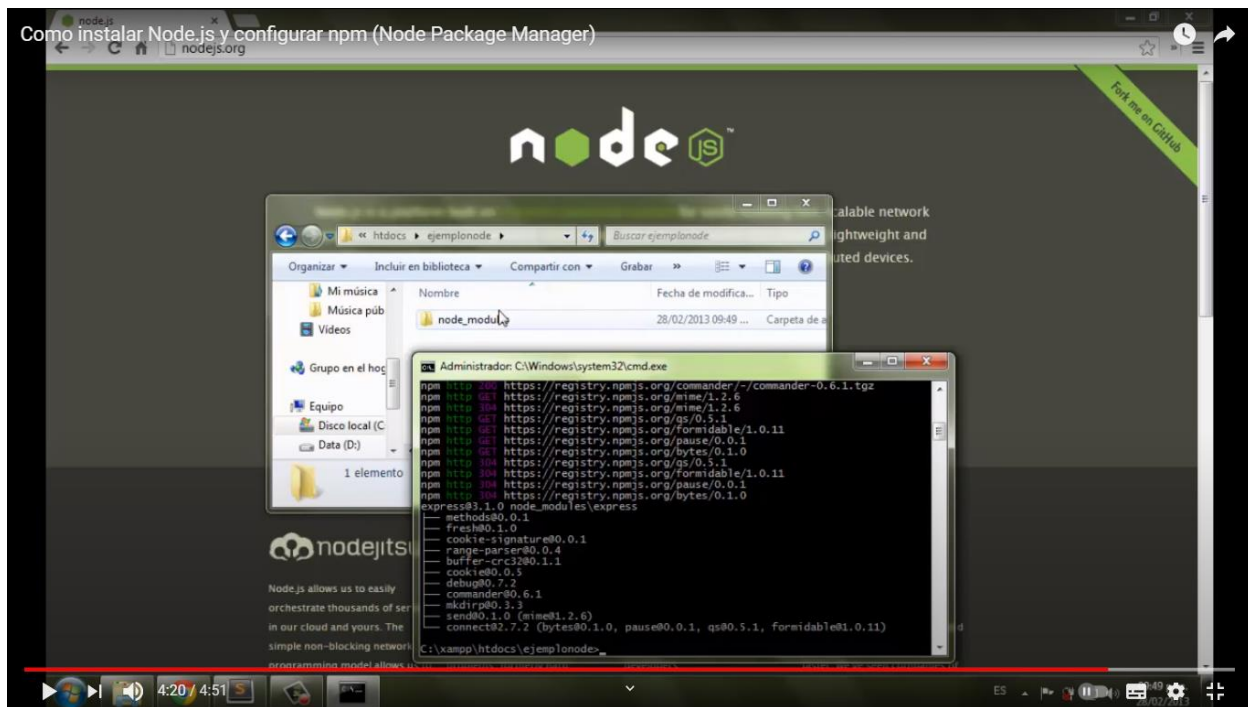
Fase 3: Nos mostrara el cuadro de asistencia, seleccionamos la carpeta donde se instalara (esto es a tu gusto), aceptamos los términos y condiciones y le damos next.





Fase 4: Comprobar que hemos instalado correctamente el framework en cmd, por ejemplo en nodejs usamos el comando “node -v” y si todo sale bien. Nos mostrará la versión que instalamos.





Descripción de elementos de configuración de las aplicaciones Web.

Los elementos para poder configurar nuestras aplicaciones web son :

1. Visibilidad del estado del sistema

Una aplicación debe ser transparente con el usuario e informar en todo momento de los procesos que están teniendo lugar. Es importante proporcionar la mayor cantidad posible de información para que usuario tenga claro el estado de los procesos que pueden estar ejecutándose.

2. Un único lenguaje

La gran mayoría de usuarios no dominan el lenguaje técnico perteneciente a las aplicaciones web. No es una buena opción, por lo tanto, utilizar en los mensajes o comunicaciones conceptos que no van a entender. Debe facilitárseles un mensaje que comprendan y, en caso de ser necesario, valorar la posibilidad de ampliar en detalles más técnicos.

3. La posibilidad de deshacer una acción

Las acciones involuntarias de los usuarios no tienen por qué conducirles a un camino sin salida. Es importante ofrecerles la posibilidad de volver atrás o deshacer una acción tras haber pulsado un botón accidentalmente.

4. La consistencia

Determinados símbolos e iconos tienen un uso y un significado estandarizado. No tiene sentido inventar nuevas funcionalidades para iconos que ya representan algo determinado para los usuarios. Lo más apropiado es atribuirles esa misma acción.

5. La flexibilidad

La aplicación debe adaptarse a los diferentes perfiles de usuario y ser eficiente en el uso para la totalidad de ellos. Es conveniente hacer las cosas sencillas para los usuarios nuevos, pero también establecer ciertos atajos para los más avanzados, para que la navegación no se convierta en una tarea tediosa para ellos.

6. Un diseño minimalista

Menos es más. Limita al máximo el número de opciones en función de la operativa del usuario y evita diseñar una aplicación repleta de iconos, menús y opciones. Busca un diseño minimalista que con pocos botones pueda dar acceso a la funcionalidad necesaria.

7. Ayuda al usuario

Evita configurar mensajes de error que sean de poca utilidad para el usuario. Un mensaje de error oportuno es aquel que contribuye a identificar el problema y que, por lo tanto, tiende la mano al usuario con información de valor que permitirá solucionar el problema rápidamente.

La mayoría de frameworks se basan en el modelo modelo MVC (Controlador => Modelo => Vista), que permiten fragmentar la

programación en estos aspectos básicos en cuanto a la implementación de nuestro sistema:

Modelo: maneja las operaciones lógicas, y de gestión de información para resultar de una forma explicable y sin titubeos.

Vista: le corresponde dibujar, o expresar la última forma de los datos: la interfaz gráfica que interactúa con el usuario final del programa (GUI).

Controlador: Con este apartado podemos controlar el acceso (incluso todo) a nuestra aplicación, y esto puede incluir: archivos, scripts, y/o programas; cualquier tipo de información que permita la interfaz.