| **Taller PARTE 1: Creación de APIRest con Node y express** |
| --- |
| **Programa: ADSO**  **Competencia:** Modelado de los artefactos del software.  .  **R.A:** Estructurar el modelo de datos del software de acuerdo con las especificaciones del análisis.  **Aprendiz: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Ficha: \_\_\_\_\_\_\_\_\_\_\_\_\_**  **Instructor**: Jairo Augusto Arboleda Londoño (PMP®) **Fecha**:  **Puntuación máxima: 5 Tiempo estimado: 2** horas |
| **Actividad** |
| Usted debe descargar los archivos json productos y órdenes y con base en estos realizar los ejercicios propuestos en este taller. |
| **Enunciados** |
| **Paso a paso:** |
|  |
|  |
| 1. Verificar versión de Node: node –version |
| 1. **Verificar qué versión de npm se instaló por defecto con node:**   **npm –versión** |
| 1. **Desinstalar la versión anterior(SOLO SI ES MUY ANTIGUA)** |
| 1. **Instalar desde: https://nodejs.org/en versión LTS 22.14.0** |
| 1. **Reiniciar** |
| 1. Crear carpeta mkdir primeraapinodejs\* |
| 1. Ingresar a ella y abrir el proyecto con VSc |
| 1. Abrir una terminal con Ctrl shift ñ y |
| 1. Instalar los paquetes: npm init, no tienes que responder a las preguntas que formula, si deseas puedes dar una descripción y un autor, pero no es necesario. Enter a todo y listo. Se crear el archivo   Is this ok?: Yes  Se debe haber creado el **package.json** que es el archivo de configuración y se puede modificar posteriormente. |
| 1. Podrías haber evitado esas preguntas al instalar el proyecto digitando:   npm init –yes |
| 1. Instalar **express** que es un Framework de node para levantar servidores.   npm i express |
| 1. Ir al proyecto y crear el archivo **index.js** para escribir el código del server. |
| 1. Observar qu al instalar express se debe haber creado y generar el archivo: |
| 1. Vaya al **package.json y agregue esta línea para que el index pueda permitir importar librerías que requiere.** |
| 1. Levantar el servidor por el puerto 3000     Probar el server:  Vaya a la terminal del proyecto y dentro del proyecto haga esto:    Autorice al firewall si es necesario y verá esto:  **El servidor está escuchando en el Puerto 3000**  Probar esto: |
|  |
| 1. Configurar los end points para probar la API, agregue las líneas del 5 al 9 y grabe. |
| 1. Vaya a la ventana de comandos y ejecute el proyecto de nuevo, esto detendrá la ejecución del mismo: escriba: node index.js |
| 1. Ahora vaya al navegador para ver el resultado: |
| 1. Para evitar que el server tenga que reiniciarse siempre para reflejar el resultado en el end point. Se debe instalar un paquete: nodemon, se instala como dependencia. O sea se requiere para que cada que corramos el proyecto, el servidor se refresque. |
| 1. Su **package.json** debe quedar así: con dos dependencias: express y nodemon. Si no se refleja el nodemon, inténtalo hasta que se logre ver en el archivo. |
| 1. Vaya el package.json y actualice el código anterior por que está señalado con rojo así:     Con esto garantizamos qué nuestro archivo índex se va a ejecutar siempre con no nodemon. |
| 1. Ir a la ventana del proyecto y ejecutar:   Esto ejecutará el script de nodemon configurado en el package.json. Ya no tendremos que ejecutar más node index.js, cada que grabe se ejecutará de forma automática la dependencia. si te da un error, debes intentarlo de nuevo. Hasta que refresque.  El mensaje devuelto es:    Y si vamos al navegador la app está corriendo:    Si no ve cambios refresque la página. |
| 1. Crear un nuevo archivo  **bd.json** con 10 documentos dentro del proyecto:   Esto es solo una muestra del contenido. |
| 1. Vaya a **index.js** y codifíquelo para que lea el archivo Json. |
| 1. Ejecute y observe que ya debe de cargarse el archivo Json |
| **FIN parte 1** |