

# Informe de Laboratorio 03

## Tema: Ajax y NodeJS

| Nota |
|------|
|      |

| Estudiante   | Escuela  | Asignatura  |
|--|--|---|
| Rutbel Carlos Ttito Campos<br>rstitoca@unsa.edu.pe | Escuela Profesional de<br>Ingeniería de Sistemas | Programación Web 2<br>Semestre: III<br>Código: 20231001 |

| Laboratorio | Tema          | Duración |
|-------------|---------------|----------|
| 03          | Ajax y NodeJS | 04 horas |

| Semestre académico | Fecha de inicio   | Fecha de entrega |
|--------------------|-------------------|------------------|
| 2023 - A           | Del 10 Abril 2023 | Al 17 Abril 2023 |

### 1. Equipos, materiales y temas utilizados

- Sistema Operativo Ubuntu 22.04.2 LTS
- Git 2.39.2.
- Cuenta en GitHub con el correo institucional.
- Python
- Arreglos en python

### 2. Tarea

- Ejercicio: Cree una aplicación NodeJS con express, para administrar una agenda personal.
- Home (“/”) : Página Principal
- Preferiblemente trabaje todo en una misma interfaz.
- Ejemplo de estructura de la agenda cuando se quiere “Ver eventos”.

Listing 1: Estructura de archivos

```
agenda DIR
|
|----- 2023-05-15 DIR
| |----- 10-00.txt FILE
| |----- 13-30.txt FILE
|----- 2023-05-20 DIR
| |----- 09-10.txt FILE
| |----- 20-40.txt FILE
```

- La aplicación debe permitir:
- Crear evento: fecha y hora. Si ya existe el archivo no debería ingresar el evento La primera línea es el título del evento, las demás líneas son la descripción del evento.
- Preferiblemente trabaje todo en una misma interfaz.
- Editar evento -¿Se muestran el archivo donde esta el detalle del evento
- Ver eventos. Utilizar el formato árbol especificado anteriormente, donde debería incluirse sólo el título del evento.
- Recomendaciones:
- Interfaz que trabaje todo un CRUD en una sola vista. Se pueden usar ventanas emergentes

### 3. Solucion de Ejercicios

#### 3.1. Creando proyecto NodeJS

```
carlos@core:~/pw2-lab-d-23a/Lab4$ npm init -y
Wrote to /home/carlos/pw2-lab-d-23a/Lab4/package.json:

{
  "name": "lab4",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

#### 3.2. Instalando Express

```
carlos@core:~/pw2-lab-d-23a/Lab4$ npm install express
added 58 packages, and audited 59 packages in 4s

8 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

### 3.3. Archivo HTML

Listing 2: index.html

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Agenda Personal</title>
5   </head>
6   <body>
7     <div id="main">
8       <h1>PWEB2 - LAB 3 Ajax y NodeJS</h1>
9       <button onclick="mostrarLista()">Mostrar Lista de eventos</button>
10    </div>
11    <script src="script.js"></script>
12  </body>
13 </html>
```

### 3.4. Archivo JS para AJAX

Listing 3: script.js

```
1 function mostrarLista() {
2   //fetch (GET) para obtener JSON de estructura de eventos y mostrarlos como lista HTML
3 }
4 function formCrearEvento(){
5   //formulario html para crear un evento
6 }
7 function crearEvento(){
8   //fetch para crear un evento
9   console.log("hola llego a crear evento");
10
11   let date = document.getElementById("date").value;
12   let time = document.getElementById("time").value;
13   let title = document.getElementById("title").value;
14   let description = document.getElementById("description").value;
15   const data = {
16     date:date,
17     time:time,
18     title:title,
19     description:description
20   }
21   console.log(data);
22   const request = {
23     method: 'POST',
24     headers: {
25       'Content-Type': 'application/json',
26     },
27     body: JSON.stringify(data)
28   };
29   const url = 'http://localhost:3000/event';
30
31   fetch(url,request)
32   .then(function(response) {
33     if(response.ok){
```

```
34     mostrarLista();
35   }else{
36     console.log('Error al enviar los datos');
37   }
38 })
39 .catch(error =>{
40   console.log('Error al crear archivo:',error);
41 });
42
43 }
44 function verEvento(date,time){
45   //fetch (GET) para ver un evento
46 }
47 function editarEvento(date,time){
48   //formulario para editar un evento
49 }
50 function actualizarEvento(date,time){
51   //fetch (PUT) para actualizar un evento
52 }
53 function borrarEvento(date,time){
54   //fetch (DELETE)para borrar un evento
55 }
```

### 3.5. Archivo JS para el lado del Servidor

Listing 4: app.js

```
1 //Usando framework express
2 const express = require('express');
3 const bodyParser = require('body-parser');
4 const fs = require('fs');
5 const path = require('path');
6
7 const app = express();
8
9 //Configuracion de la plantilla de vista
10 app.use(bodyParser.json());
11 app.use(bodyParser.urlencoded({ extended: false }));
12 app.use(express.static('public'));
13
14 //Ruta principal
15 app.get('/', (req, res) => {
16   res.sendFile(path.join(__dirname, 'index.html'));
17 });
18
19 //lista de los eventos creados
20 app.get('/list', (req, res) => {
21   //Lee los eventos guardados y devuelve un JSON
22   const eventTree = getEventTree();
23   res.json(eventTree);
24 });
25
26 //Ruta para crear un evento
27 app.post('/event', (req, res) => {
28   const { date, time, title, description } = req.body;
```

```
29 const eventPath = path.join(__dirname, 'events', date);
30 const eventFile = path.join(eventPath, `${time}.txt`);
31
32 //Verificar si el archivo del evento ya existe
33 if (fs.existsSync(eventFile)) {
34   res.send('El evento ya existe.');
```

```
35 } else {
36   //Crear el directorio de la fecha si no existe
37   if (!fs.existsSync(eventPath)) {
38     fs.mkdirSync(eventPath);
39   }
40   //Guardar el evento en el archivo
41   fs.writeFileSync(eventFile, `${title}\n${description}`);
42   res.end();
43 }
44 });
45
46 app.get('/event/:date/:time', (req, res) => {
47   //Obtener la descripcion del contenido de un evento
48 });
49
50 app.put('/event/:date/:time/update', (req, res) => {
51   //Ruta para actualizar un evento (procesamiento del formulario de edicion
52 });
53
54 app.delete('/event/:date/:time/delete', (req, res) => {
55   //Ruta para eliminar un evento
56 });
57
58 function getEventTree() {
59   // Obtener la estructura JSON de eventos
60 }
61 function getTitleFromContent(content) {
62   //Obtener el titulo del contenido de un evento
63 }
64 }
65 function getDescriptionFromContent(content) {
66   //Obtener la descripcion del contenido de un evento
67 }
68 //Iniciar el servidor
69 app.listen(3000, () => {
70   console.log("Servidor iniciado en http://localhost:3000");
71 });
```

### 3.6. Capturas de Ejecucion

## PWEB2 - LAB 3 Ajax y NodeJS

Mostrar Lista de eventos

Figura 1: Ruta Raiz

## Agenda Personal


### Ver eventos


- 2023-05-20
  - 21:02->[PWEB2] [Editar](#) [Eliminar](#)
- 2023-05-24
  - 19:45->[Fiesta] [Editar](#) [Eliminar](#)
- 2023-05-26
  - 12:01->[io] [Editar](#) [Eliminar](#)

CREAR NUEVO EVENTO

Figura 2: Lista de eventos

### Crear evento

Fecha:  

Hora:  

Título:

Descripción:

Figura 3: Crear Evento

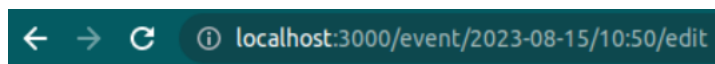
←
→
↻
ℹ
localhost:3000

## Agenda Personal

### Ver eventos

- 2023-05-20
  - 21:02->[PWEB2] [Eliminar](#) [Editar](#)
- 2023-05-24
  - 19:45->[Fiesta] [Eliminar](#) [Editar](#)
- 2023-08-15
  - 10:50->[Festividad] [Eliminar](#) [Editar](#)

Figura 4: Lista con nuevo Evento



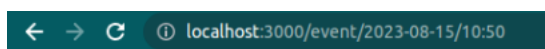
## Editar evento

Título:

Descripción:

[Cancelar](#)

Figura 5: editar evento



## Detalles del evento

**Fecha:** 2023-08-15

**Hora:** 10:50

**Contenido:**

Festividad 483 ---- Aniversario de arequipa

[Volver](#)

Figura 6: Evento editado

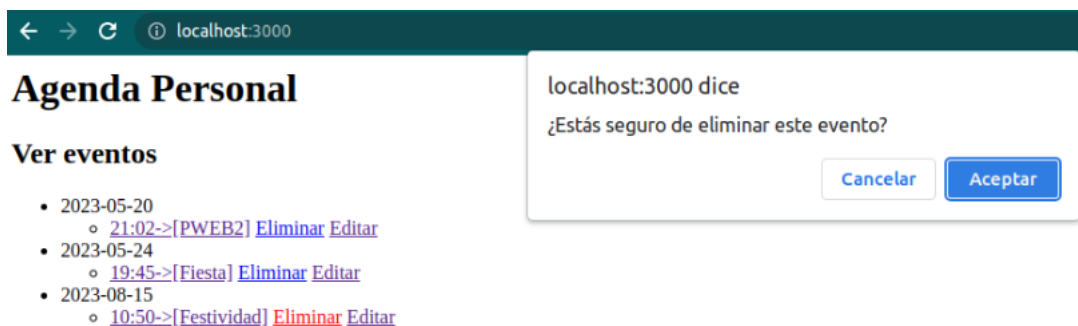


Figura 7: Elminar evento



Figura 8: Lista de eventos final

## 4. URL de Repositorio Github

- URL del Repositorio GitHub para clonar o recuperar:
- <https://github.com/RutbelCarlosTC/pw2-lab-d-23a.git>
- URL para el laboratorio 03 en el Repositorio GitHub.
- <https://github.com/RutbelCarlosTC/pw2-lab-d-23a/tree/main/Lab3>

## 5. Actividades con el repositorio GitHub

### 5.1. Commits

URL de todos los commits realizados: <https://github.com/RutbelCarlosTC/pw2-lab-d-23a/commits/main/Lab3>

- Commits iniciales

```
commit 9963c9618505ed7b3a71a7cf2928617269546491
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Fri May 26 12:14:01 2023 -0500

    se agrego script inicial en el servidor para mostrar la lista de eventos

commit 4f3de090ae8958b9fdc5bf2cec9dfe456180e8dc
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Fri May 26 12:10:28 2023 -0500

    se agrego carpeta statica public con archivo script.js para mostrar la lista de eventos

commit ba67d4ea3b525b6189673d7393d294ff8082cbd3
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date:   Fri May 26 12:07:43 2023 -0500

    se agrego index.html inicial
```

Figura 9: crear proyecto NodeJs Y archivos iniciales

- Crear y modificar Rutas



```
commit 4c1699c3e702b935cc1ebfb2352fea2a95a0e232
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:33:13 2023 -0500

    se modifiko ruta para actualizar un evento

commit ae8c2568c5d30975c295fbfb8186d5233fad48f7
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:28:28 2023 -0500

    se agrego funcion verEvento() que recibe un JSON y muestra los detalles de u

commit e2539e4642a59eff362e6143ef0c159a04a5e511
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:22:32 2023 -0500

    se modifiko ruta de ver detalles de evento

commit 7fbc2cd27358781d60f7d5ff712051515d23f818
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:20:33 2023 -0500

    se modifiko ruta para ver detalles de un evento para devolver un JSON de arc

commit bf86abfd13513d9c5bba28c994ee52c877dbb435
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:15:25 2023 -0500

    se agrego funcion para agregar un formulario de creacion de un nuevo evento

commit 2460b99341f60b8ceaa20dc75359c4ef10e30156
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:13:04 2023 -0500

    se modifiko ruta para crear un evento

commit f17308a59a66ed7f52634d06715bd6b3c5233979
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:08:19 2023 -0500

    se modifiko la funcion mostrarLista() para recibir un JSON con la estructura
```

Figura 10: modificando rutas en el lado del servidor

- Crear, editar y borrar un evento

```
commit 58d5d1488909e6c5ff5a62f965a5362e895848d1
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 18:01:27 2023 -0500

    se elimino evento 12:18.txt

commit bf29a2f82d0bde738138271e0f82c00484e3ca6f
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:57:31 2023 -0500

    se modifiko funcion borrarEvento

commit 44977797f2aaa2e7345853ae48e51b6b199216bc
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:55:29 2023 -0500

    se modifiko un evento

commit 0e45668ede8ecc46f81805907124f287b9aa858e
Author: RutbelCarlosTC <rttitoca@unsa.edu.pe>
Date: Sun May 28 17:52:40 2023 -0500

    se agrego un nuevo evento
```

Figura 11: probando CRUD con nuevo evento

## 6. Rúbricas

### 6.1. Entregable Informe

Tabla 1: Tipo de Informe

| <b>Informe</b> |   |
|----------------|---|
| <b>Latex</b>   | El informe está en formato PDF desde Latex, con un formato limpio (buena presentación) y fácil de leer. |

### 6.2. Rúbrica para el contenido del Informe y demostración

- El alumno debe marcar o dejar en blanco en celdas de la columna **Checklist** si cumple con el ítem correspondiente.
- Si un alumno supera la fecha de entrega, su calificación será sobre la nota mínima aprobada, siempre y cuando cumpla con todos los ítems.
- El alumno debe autocalificarse en la columna **Estudiante** de acuerdo a la siguiente tabla:

Tabla 2: Niveles de desempeño

| <b>Puntos</b> | <b>Nivel</b>         |                 |                    |                     |
|---------------|----------------------|-----------------|--------------------|---------------------|
|               | Insatisfactorio 25 % | En Proceso 50 % | Satisfactorio 75 % | Sobresaliente 100 % |
| <b>2.0</b>    | 0.5                  | 1.0             | 1.5                | 2.0                 |
| <b>4.0</b>    | 1.0                  | 2.0             | 3.0                | 4.0                 |

Tabla 3: Rúbrica para contenido del Informe y demostración

| Contenido y demostración |  | Puntos | Checklist | Estudiante | Profesor |
|--------------------------|--|--------|-----------|------------|----------|
| <b>1. GitHub</b>         | Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.   | 2      | X         | 2          |          |
| <b>2. Commits</b>        | Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).   | 4      | X         | 3          |          |
| <b>3. Código fuente</b>  | Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.   | 2      | X         | 2          |          |
| <b>4. Ejecución</b>      | Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.   | 2      | X         | 2          |          |
| <b>5. Pregunta</b>       | Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).  | 2      | X         | 2          |          |
| <b>6. Fechas</b>         | Las fechas de modificación del código fuente estan dentro de los plazos de fecha de entrega establecidos.  | 2      | X         | 2          |          |
| <b>7. Ortografía</b>     | El documento no muestra errores ortográficos.  | 2      | X         | 1          |          |
| <b>8. Madurez</b>        | El Informe muestra de manera general una evolución de la madurez del código fuente, explicaciones puntuales pero precisas y un acabado impecable. (El profesor puede preguntar para refrendar calificación). | 4      | X         | 3          |          |
| <b>Total</b>             |  | 20     |           | 17         |          |

## 7. Referencias

- <https://www.w3schools.com/java/default.asp>
- <https://www.geeksforgeeks.org/insertion-sort/>