



| | | |
|--|--|---|
|  | <p align="center">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p> |  |
| Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación | | |
| Aprobación: 2022/03/01 | Código: GUIA-PRLE-001 | Página: 1 |

INFORME DE LABORATORIO

| INFORMACIÓN BÁSICA | | | | | |
|--|--------------------|-----------------------------|---------|-----------------------|---|
| ASIGNATURA: | Programación WEB 2 | | | | |
| TÍTULO DE LA PRÁCTICA: | | | | | |
| NÚMERO DE PRÁCTICA: | 3 | AÑO LECTIVO: | 2022 | NRO. SEMESTRE: | 1 |
| FECHA DE PRESENTACIÓN | 20/05/2022 | HORA DE PRESENTACIÓN | 11:25am | | |
| INTEGRANTE (s): Diego Alonso Huamani Luque | | | | NOTA: | |
| DOCENTE(s): Carlo Jose Luis Corrales Delgado | | | | | |

| SOLUCIÓN Y RESULTADOS |
|---|
| <p>I. SOLUCIÓN DE EJERCICIOS/PROBLEMAS</p> <p><i>EJERCICIOS RESUELTOS (INDIVIDUAL)</i></p> <p><i>RESUELTO 1.- Cree una aplicación web que ejecute javascript en el cliente (dentro de la carpeta pub) y nodejs en el servidor.</i></p> <p><i>Servidor</i></p> <pre> 1 const path = require('path'); 2 const express = require('express'); 3 const app = express(); 4 app.use(express.static('pub')); 5 6 app.listen(3000, () => { 7 console.log("Escuchando en: http://localhost:3000") 8 }); 9 10 11 app.get('/', (request, response) => { 12 response.sendFile(path.resolve(__dirname, 'index.html')); 13 }); 14 15 16 </pre> |

Cliente (dentro de pub)

```
1 document.addEventListener('DOMContentLoaded',function(){
2     document.getElementById("prueba").innerHTML="Este texto se inserto con un Script dentro de pub ";
3     return false;
4 });
5
```

HTML

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4
5     </head>
6     <body>
7         <h1>EJERCICIO RESUELTO 1 </h1>
8         <div id="prueba"></div>
9         <script src="testEstatico.js"></script>
10    </body>
11 </html>
```

RESULTADO

EJERCICIO RESUELTO 1

Este texto se inserto con un Script dentro de pub

RESUELTO 2.- Cree una aplicación web que realice una petición ajax en el lado del cliente y responda usando nodejs en el lado del servidor.

SERVIDOR

```
1 // En el lado del servidor
2 const fs = require('fs')
3 const path = require('path')
4 const express = require('express')
5 const app = express()
6 app.use(express.static('pub'))
7
8 app.listen(3000, () => {
9     console.log("Escuchando en: http://localhost:3000")
10 });
11
12 app.get('/', (request, response) => {
13     response.sendFile(path.resolve(__dirname, 'index2.html'))
14 })
15
16 app.get('/recitar', (request, response) => {
17     fs.readFile(path.resolve(__dirname, 'priv/poema.txt'), 'utf8',
18         (err, data) => {
19             if (err) {
20                 console.error(err)
21                 response.status(500).json({
22                     error: 'message'
23                 })
24                 return
25             }
26             response.json({
27                 text: data.replace(/\n/g, '<br>')
28             })
29         })
30     //
31 })
```

Cliente

```
1
2 // En el lado del cliente
3 function recitar() {
4     const url = 'http://localhost:3000/recitar'
5     fetch(url).then(
6         response => response.json()
7     ).then(
8         data => {
9             document.getElementById("poema").innerHTML = data.text
10        }
11    )
12 }
13
```

HTML

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4
5     </head>
6     <body>
7         <h1>EJERCICIO RESUELTO 2 </h1>
8         <button type="button" onclick="recitar()" >Recitar!</button>
9         <div id="poema"></div>
10        <script src="recitaCliente.js"></script>
11    </body>
12 </html>
```

RESULTADO

EJERCICIO RESUELTO 2

Recitar!

Hay golpes en la vida, tan fuertes... ¡Yo no sé!
Golpes como del odio de Dios; como si ante ellos,
la resaca de todo lo sufrido
se empozara en el alma... ¡Yo no sé!

Son pocos; pero son... Abren zanjas oscuras
en el rostro más fiero y en el lomo más fuerte.
Serán tal vez los potros de bárbaros Atilas;
o los heraldos negros que nos manda la Muerte.

Son las caídas hondas de los Cristos del alma
de alguna fe adorable que el Destino blasfema.
Esos golpes sangrientos son las crepitaciones
de algún pan que en la puerta del horno se nos quema.

Y el hombre... Pobre... ¡pobre! Vuelve los ojos, como
cuando por sobre el hombro nos llama una palmada;
vuelve los ojos locos, y todo lo vivido
se empoza, como charco de culpa, en la mirada.

Hay golpes en la vida, tan fuertes... ¡Yo no sé!

RESUELTO 3.- Cree una aplicación que haga peticiones AJAX usando POST a un servidor NodeJS

SERVIDOR

```
1 // En el lado del servidor
2 const fs = require('fs')
3 const path = require('path')
4 const express = require('express')
5 const bp = require('body-parser')
6 const MarkdownIt = require('markdown-it'),
7     md = new MarkdownIt();
8 const app = express()
9
10 app.use(express.static('pub'))
11 app.use(bp.json())
12 app.use(bp.urlencoded({
13     extended: true
14 }))
15
16 app.listen(3000, () => {
17     console.log("Escuchando en: http://localhost:3000")
18 })
19
20 app.get('/', (request, response) => {
21     response.sendFile(path.resolve(__dirname, 'index3.html'))
22 })
23
24 app.post('/', (request, response) => {
25     console.log(request.body);
26     let markDownText = request.body.text;
27     console.log(markDownText+"<--- texto a convertir");
28     let htmlText = md.render(String(markDownText));
29     response.setHeader('Content-Type', 'application/json')
30     response.end(JSON.stringify({
31         text: htmlText.replace(/\n/g, '<br>')
32     }))
33 })
34
```

CLIENTE

```
1
2 // En el lado del cliente
3 function recitar(markupText) {
4     const url = 'http://localhost:3000/'
5     const data = {
6         text: markupText
7     }
8     console.log(data)
9     const request = {
10         method: 'POST', // Podría ser GET
11         headers: {
12             'Content-Type': 'application/json',
13         },
14         body: JSON.stringify(data),
15     }
16     http = fetch(url, request)
17     http.then(
18         response => response.json()
19     ).then(
20         data => {
21             document.getElementById("htmlCode").innerHTML = data.text
22         }
23     )
24 }
25 document.addEventListener('DOMContentLoaded', function () {
26     const text = document.getElementById('markupText')
27     document.getElementById('markupForm').onclick = () => {
28         console.log(text.value+"<-- esta es el text value");
29         recitar(text.value)
30         return false;
31     }
32 })
33
```

HTML

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <script src="EstaticoScriptEJER3.js"></script>
5   </head>
6   <body>
7     <h1>Renderizador de MD</h1>
8     <textarea type="text" id="markupText" size="30"></textarea>
9     <input type="submit" id="markupForm" value="Renderizar en MD">
10    <div id="htmlCode">
11    </body>
12 </html>
```

RESULTADO

Renderizador de MD

```
# The largest heading
## The second largest heading
##### The smallest heading
```

Renderizar en MD

The largest heading

The second largest heading

The smallest heading

This is bold text

This text is italicized

~~This was mistaken text~~



This text is *extremely* important

All this text is important

Some basic Git commands are:

```
git status
git add
git commit
```

This site was built using [GitHub Pages](#).

| | | |
|---|---|---|
|  | <p style="text-align: center;">UNIVERSIDAD NACIONAL DE SAN AGUSTIN FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMA</p> |  |
| <p style="text-align: center;">Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación</p> | | |
| <p>Aprobación: 2022/03/01</p> | <p>Código: GUIA-PRLE-001</p> | <p>Página: 6</p> |

| | |
|--|--|
| | |
| <p>II. SOLUCIÓN DEL CUESTIONARIO</p> <p>-</p> | <p>III. CONCLUSIONES</p> <p><i>Node JS nos permite utilizar JavaScript para desarrollar en el lado del servidor. Al hacer esto se consigue contenido dinámico de páginas web antes de que la página se envíe al navegador web del usuario. Por otro lado, la técnica AJAX elimina cualquier retraso, pues funciona directamente en el navegador web y ejecuta operaciones como la validación de datos sin tener que consultar al servidor cada vez. Estas 2 herramientas son muy importantes y, a través de este laboratorio, ha quedado claro su importancia y funcionamiento.</i></p> |

| <p style="text-align: center;">RETROALIMENTACIÓN GENERAL</p> |
|---|
| <p>Los problemas y la cantidad de tiempo estuvieron bien.</p> |

| <p style="text-align: center;">REFERENCIAS Y BIBLIOGRAFÍA</p> |
|--|
| <p>https://github.com/dhuamanilu/LAB03NODEJS-TRABAJO-individual.git</p> |