

MANUAL TÉCNICO

Presentado por
Grupo8

Marcos Arnoldo Itzep Ixmay 201907156
Josué Noel Mazariegos Gramajo 201900092
Byron Enrique Rumpich Sal 201907769
Carlos Alberto Rodas Galvez 201700420

Índice

GENERAL	3
JSON	4
HTML	5
JS	6

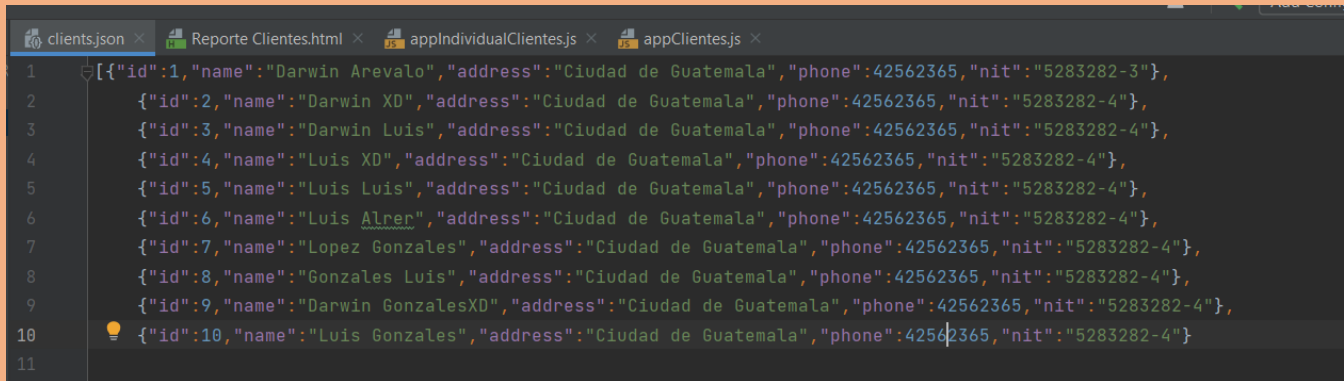
GENERAL

Dentro de este manual se encuentra una breve descripción del código de la creación de reportes del proyecto final.

Se manejaron tres clases de archivos distintos.

- JSON
- HTML
- JS

JSON



The screenshot shows a code editor with four tabs: 'clients.json', 'Reporte Clientes.html', 'appIndividualClientes.js', and 'appClientes.js'. The 'clients.json' tab is active, displaying a JSON array of 10 client objects. Each object contains 'id', 'name', 'address', 'phone', and 'nit' fields. The names include 'Darwin Arevalo', 'Darwin XD', 'Darwin Luis', 'Luis XD', 'Luis Luis', 'Luis Alrer', 'Lopez Gonzales', 'Gonzales Luis', 'Darwin GonzalesXD', and 'Luis Gonzales'. The addresses are all 'Ciudad de Guatemala', and the phone numbers are '42562365'. The NIT values are either '5283282-3' or '5283282-4'. A cursor is positioned at the end of the 10th object.

```
1 [{"id":1,"name":"Darwin Arevalo","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-3"},
2   {"id":2,"name":"Darwin XD","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
3   {"id":3,"name":"Darwin Luis","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
4   {"id":4,"name":"Luis XD","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
5   {"id":5,"name":"Luis Luis","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
6   {"id":6,"name":"Luis Alrer","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
7   {"id":7,"name":"Lopez Gonzales","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
8   {"id":8,"name":"Gonzales Luis","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
9   {"id":9,"name":"Darwin GonzalesXD","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"},
10  {"id":10,"name":"Luis Gonzales","address":"Ciudad de Guatemala","phone":42562365,"nit":"5283282-4"}
11]
```

Los archivos .json contienen los datos que utilizara nuestra aplicación al iniciar su ejecución.

Para el funcionamiento de este software para un restaurante se utilizan cinco archivos de esta clase: clientes, users, config, invoices y products.

HTML

```
clients.json x Reporte Clientes.html x appIndividualClientes.js x appClientes.js x
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLI
8   <title>Reporte Noel</title>
9 </head>
```

Los reportes finales se trabajan en archivos html.

```
10 <body>
11
12 <div class="container">
13   <h1>Clientes</h1>
14   <input id="inputJson" type="file" />
15
16   <div id="todoslosclientes" class="d-flex justify-content-around flex-wrap" >
17   </div>
18
19 </div>
20
21   <script src="appClientes.js"></script>
22 </body>
23 </html>
```

El cuerpo del reporte hace la lectura del archivo appClientes.js

JS

```
clients.json × Reporte Clientes.html × appIndividualClientes.js × appClientes.js ×
*.js files are supported by IntelliJ IDEA Ultimate
1  const inputJson = document.getElementById('inputJson')
2  localStorage.setItem('jsonCont', document.getElementById('inputJson'))
3
4  const divtodoslosclientes = document.getElementById('todoslosclientes')
5  localStorage.setItem('clientesCont', document.getElementById('todoslosclientes'))
6
7  function leerJSON(){
8      const fileReader = new FileReader();
9
10
11  function miOnLoad(){
12      const json = JSON.parse(fileReader.result)
13      crearLista(json)
14  }
15
16  fileReader.readAsText(inputJson.files[0])
17  fileReader.onload = miOnLoad;
18
19  }
```

Los archivos JavaScript, leen los archivos .json y crean las respectivas listas con los datos obtenidos.

```
20
21  function crearLista(clientes){
22      let html = ''
23      for (const cliente of clientes){
24          html += crearCliente(cliente)
25      }
26
27      divtodoslosclientes.innerHTML = html;
28
29  }
30
31
```

```

31
32     function crearCliente(cliente){
33         let html = `
34             <div class="card mt-4" style="width: 18rem;">
35                 <h5 class="card-title"> ID# ${cliente.id}</h5>
36                 
37                 <div class="card-body">
38                     <div style='text-align:center'><h5 class="card-title">${cliente.name}</h5></div>
39                     <p class="card-text"></p>
40                 </div>
41                 <ul class="list-group list-group-flush">
42                     <li class="list-group-item">Dirección: ${cliente.address}</li>
43                     <li class="list-group-item">Telefono: ${cliente.phone}</li>
44                     <li class="list-group-item">Nit: ${cliente.nit}</li>
45                 </ul>
46                 <div class="card-body">
47                     <div style='text-align:center'><a href="file:///C:/Users/EPA/Desktop/Reportes/Reporte%20Individual%20Cliente.html" class="card-link">Vista individual</a></div>
48                 </div>
49             </div>
50         `
51     }
52     return html
53 }
54 inputJson.addEventListener('change', leerJSON)
55
56
57

```

Dentro se crea un archivo html ordenando los datos obtenidos anteriormente.

```

function crearCliente(cliente){
    let html = `

<div class="card mb-3" style="max-width: 540px;">
    <div class="row g-0">
        <div class="col-md-4">

            
        </div>
        <div class="col-md-8">
            <div class="card-body">
                <ul class="list-group list-group-flush">
                    <div style='text-align:center'><h2 class="card-title">${alert(globalVariable.contenidoCliente)}</h2></div>
                    <li class="list-group-item">Dirección: ${cliente.address}</li>
                    <li class="list-group-item">Telefono: ${cliente.phone}</li>
                    <li class="list-group-item">Nit: ${cliente.nit}</li>
                </ul>
            </div>
        </div>
    </div>
</div>
    `
    return html
}
inputJson.addEventListener('change', leerJSON)

```


Los reportes individuales se manejan de la misma forma con la diferencia que muestran la información de un código específico.