



**Curso:**

Bases de Datos

**Tarea Corta:** L3M - Super

**Tema:**

Programacion imperativa

**Estudiantes:**

Karina Martínez Guerrero, 2017102001

Gerald Valverde Mckenzie, 2017112234

Darío Rodríguez Obando, 2017117105

Jonathan Guzmán Araya, 2013041216

**Profesor:**

Marco Rivera

**Grupo 01**

# Tabla de contenidos

<b>Tabla de contenidos</b>	<b>2</b>
<b>Resumen</b>	<b>3</b>
<b>Descripción de los métodos implementados</b>	<b>4</b>
<b>Descripción detallada de los algoritmos desarrollados</b>	<b>6</b>
<b>Problemas sin solución</b>	<b>7</b>
<b>Problemas encontrados</b>	<b>7</b>
<b>Modelo Entidad-Relación</b>	<b>8</b>
<b>Plan de trabajo</b>	<b>8</b>
<b>Github</b>	<b>8</b>
<b>Conclusiones</b>	<b>9</b>
<b>Recomendaciones del proyecto.</b>	<b>9</b>
<b>Bibliografía</b>	<b>10</b>

# Resumen

Consiste en crear un prototipo de una página web que permita a toda una compañía Las 3 Marías, tener una aplicación que le permite manipular de una forma correcta y más segura sus datos ya sea en cualquiera de los ámbitos, además, de permitirle tener diferentes vistas de cada uno de los sectores de la empresa, tal como vendedores, administradores, y clientes, por lo dicho anteriormente, esto es solo un prototipo, lo cual significa que sólo simula cómo sería una aplicación como tal si se llevara a cabo su implementación.

Está se compone de diferentes vistas:

1. **Administración:** es un conjunto de vistas destinadas al uso de los administradores del local, de tal forma que estos pueden acceder y modificar los datos, además pueden observar más detalles de los trabajadores como sus horas laboradas, las sucursales que hay y la lista de productos respectiva de cada una de ellas.
2. **Reportes:** se encuentran las vistas de los productos por sucursal o los productos totales que hay en todo la marca.
3. **Venta:** especial para la generación de facturas , ya que se escoge el cajero y la sucursal donde trabaja y la lista de productos por la que este paso, además calcula un valor total por producto y el total por pagar.

En general, las páginas en html, que contienen el controlador y manejan todos los eventos en dichas páginas, además carga los datos que se necesitan por ver y todo lo que es gráfico.

El objetivo principal de la tarea es utilizar correctamente las nuevas herramientas que se necesitan para el curso, tales como la programación en un typescript, web server , c#, entre otras aplicaciones que son necesarias para la implementación de bases de datos.

## Descripción de los métodos implementados

- **add():** se utiliza mucho en la mayoría de los controller la función de agregar , para que se represente el nuevo valor que el usuario acaba de agregar a la lista de objetos , la mayoría consiste en agregar las características de este y por medio de un botón se llama a la función para que esta la agregue a la tabla de datos. Es necesario tener en cuenta que hay una función para cada vista, que en teoría hace lo mismo pero, recibe diferentes valores, ya que cada vista requiere diferentes datos para formar una instancia de lo que se quiere ejemplificar.
- **eliminar():** es un método que se llama a la hora de querer eliminar una lista de datos de la tabla, se llama por medio de un botón a la derecha de esta y tiene forma de x, la función se crea en cada uno de los controllers necesarios para que se eliminen los datos que se desean, además , se actualiza la tabla para que el resultado se vea gráficamente.
- **populateForm():** por medio del servidor de http, solicita al servidor los datos de la lista para poder ponerlo en las tablas de datos.
- **mobileFunction():** se encarga de detectar si el dispositivo del cual se accede es una PC o un celular, para que la vista se adapte según sea necesario siendo un sistema responsive.

## Descripción de las estructuras de datos (entidades) desarrolladas

- **Producto:** entidad encargada de almacenar información básica sobre los artículos del supermercado, sus atributos básicos son el código de barras, precio, nombre, descuento, descripción, impuesto y el proveedor que lo suministra.  
Se encuentra relacionada con la entidad proveedor y la entidad sucursal.
- **Sucursal:** entidad encargada de almacenar información sobre las distintas sucursales de L3M. Sus atributos básicos son nombre, teléfono y dirección.  
Se encuentra relacionada con la entidad débil compra y con la entidad trabajador.
- **Rol:** entidad encargada almacenar información sobre los distintos roles que pueden tener los trabajadores de una sucursal de L3M. Sus atributos son nombre y descripción.  
Se encuentra relacionada con la entidad trabajador.
- **Proveedor:** entidad encargada de almacenar información sobre los distintos proveedores de las sucursales de L3M. Sus atributos son nombre y cédula.  
Se encuentra relacionada con la entidad producto y con la entidad débil compra.
- **Trabajador:** entidad encarga de almacenar información sobre los trabajadores de las distintas sucursales de L3M, sus atributos son nombre, fecha de ingreso, salario por hora, fecha de nacimiento y cédula.  
Se encuentra relacionada con las entidades rol y sucursal, además de las entidades débiles jornada laboral y ventas.

## Descripción detallada de los algoritmos desarrollados

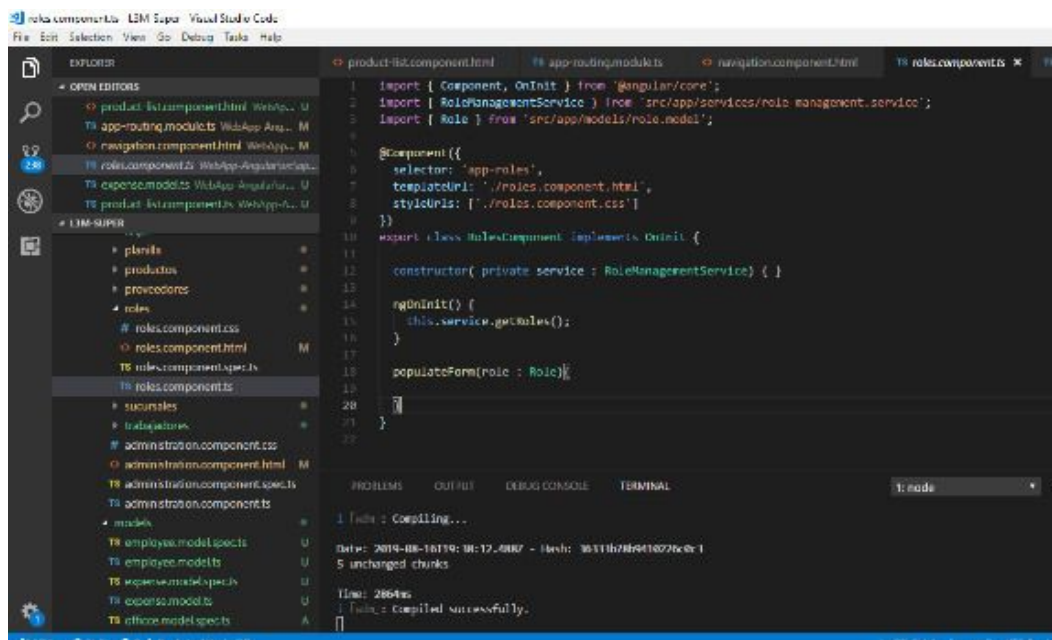
**Ngfor:** se encarga de cargar a la tabla todos los datos que se necesitan de la vista y los pone en una tabla por medio de la función `platForm`, la cual se explica más adelante.

```

1 <div class="container">
2   <h1 style="color: #263238">Gestión de Roles</h1>
3 </div>
4 <div>
5
6 <div class="table-responsive">
7   <table class="table table-hover">
8     <tr>
9       <th>Puesto</th>
10      <th>Descripción</th>
11      <th></th>
12    </tr>
13    <tr *ngFor="let Role of service.roleList" class="active">
14      <td (click)="populateForm(Role)"> {{Role.Name}}</td>
15      <td> {{Role.Description}}</td>
16      <td><button class="btn btn-sm btn-outline-danger">X</button></td>
17    </tr>
18  </table>
19 </div>
20
21 <div></div>
22 <div class="container">
23   <h3>Agregar roles <span class="label label-warning">Nuevo</span></h3>
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

**populateForm:** Se encarga de solicitar los XML al servidor, los cuales contiene los datos que se necesitan para la tabla.



## Problemas sin solución

1. Modificar datos: esto debido a que las tablas una vez que se introducen los datos, no se pueden modificar, a menos que se borre y se inserte con el cambio.
2. No se logró implementar los reportes con Crystal Reports debido a problemas de compatibilidad en Visual Studio.

## Problemas encontrados

1. Al inicio, se estaba implementando con archivos JSON, pero el servidor no se prestaba para eso, entonces se tuvo que hacer el cambio a XML.
2. Además, algunos archivos JSON no se leían, pero era que estaban con la gramática incorrecta.
3. Problema al enviar el XML al servidor, se solucionó con un cambio en el código del servidor.
4. Implementación del método para detectar dispositivos móviles: ya que se pudo crear una función pero no se sabe manejar la llamada de ésta.

# Modelo Entidad-Relación

[Link](#)

## Plan de trabajo

Nombre	Actividad	Horas	Fecha
Dario	Vistas	10	09/8/2019
	Reporte	2	14/08/2019
	Documentacion	3	15/08/2019
	Investigación	10	08/08/2019
Gerald	Vistas	20	14/08/2019
	Bootstrap	5	16/08/2019
	Documentacion	5	20/08/2019
Jonathan	Vistas	12	16/08/2019
	Reporte	8	20/08/2019
	Documentacion	5	21/08/2019
Karina	Servidor	20	16/08/2019
	Documentacion	5	19/08/2019

## Github

- <https://github.com/geraldvm/L3M-Super>
- <https://github.com/karinsmgquerrero/L3M>



# Conclusiones

- Es necesario manejar algunos datos con XML ya que el servidor los maneja más fácil.
- Para facilitar la creación de las vistas, se puede utilizar un .Css del agrado para mantener un formato en la página.
- Se debe llevar un buen manejo de los controller.
- IIS es una buena herramienta para publicar páginas web localmente, evitando así que se requiera de un IDE de un lenguaje para poder visualizarse.

# Recomendaciones del proyecto.

1. Tener en cuenta que es solamente un prototipo, el cual no funciona como tal, sino que representa la “carcasa” de la aplicación.
2. Saber que vista es la que quiero probar, así de esta forma se debe escoger si se quiere administración, reportes o ventas.
3. Cuando se trata de realizar una página web que sea responsive con el usuario es recomendable utilizar Bootstrap ya que está nos ahorra tiempo que intentar hacerlo uno mismo.
4. Angular nos permite tener un mejor manejo del modelo vista controlador para aplicaciones basadas en navegadores.

## Bibliografía

- Angular, «Angular,» [En línea]. Available: <https://angular.io/docs>. [Último acceso: 14 agosto 2019].
- Bootstrap, “Bootstrap,” [Online]. Available: <https://getbootstrap.com/docs/4.3/getting-started/introduction/>. [Accessed 09 agosto 2019].
- Microsoft, “Microsoft,” [Online]. Available: <https://docs.microsoft.com/en-gb/iis/get-started/whats-new-in-iis-10-version-1709/new-features-introduced-in-iis-10-1709>. [Accessed 18 agosto 2019].