

**POR**

**BAUTISTA GONZALEZ JOSE ARMANDO**

Trabajo: Bitácora de evidencias Inteligencia Artificial(programación web).

Grupo: 8SA

Carrera: Ing. Sistemas computacionales

Profesor(a): Guillermo Alejandro Chávez Sánchez

Fecha: 02/06/2017

## Contenido

<b>Instalación LAMP.....</b>	<b>1</b>
<b>APACHE.....</b>	<b>1.1</b>
<b>MYSQL.....</b>	<b>1.2</b>
<b>PHPMYADMIN.....</b>	<b>1.3</b>
<b>PROYECTO</b>	
<b>DIAGRAMA FLUJO.....</b>	<b>2</b>
<b>UML.....</b>	<b>2.2</b>
<b>DIAGRAMA WEB.....</b>	<b>2.3</b>
<b>_SESIONES ACCESO LOGIN.....</b>	<b>2.4</b>
<b>Utilizando Slimframework</b>	
<b>INSTALACION COMPOSER.....</b>	<b>3</b>
<b>CONFIGURACION SLIMFRAMEWORK.....</b>	<b>3.1</b>
<b>CONEXIÓN BD Y SLIMFRAMEWORK.....</b>	<b>3.2</b>
<b>CHROME Y POSTMAN.....</b>	<b>3.3</b>
<b>CONCLUSIONES.....</b>	<b>4</b>

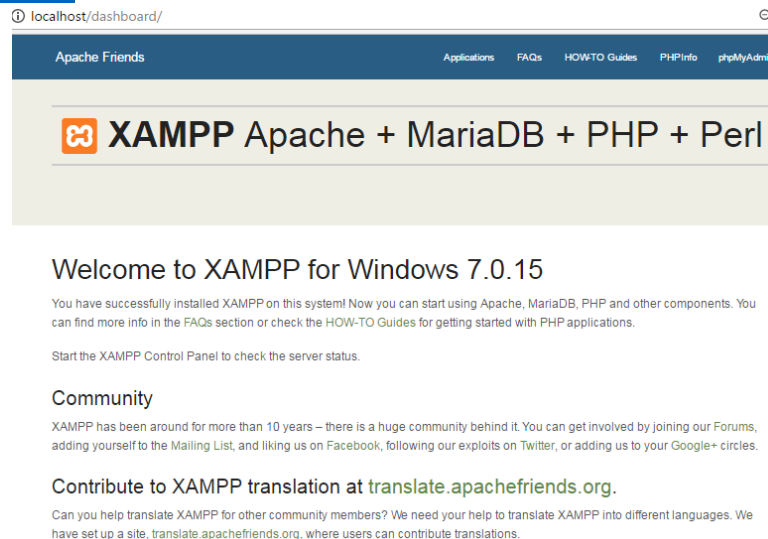
## 1. INSTALACION LAMP LINUX MINT

Abrimos la terminal de linux mint y escribimos a continuación:

- `Sudo apt-get install apache2`

Este comando nos sirve para emular un servidor ficticio para ejecutar nuestros proyectos, después de poner este comando picamos enter y verificamos en el navegador y escribimos:

- <http://localhost>



### 1.1. APACHE

Después ponemos este siguiente comando para la instalación de php

- `sudo apt-get install php libapache2-mod-php`

después de haber salido todo bien de este comando ponemos el siguiente para que se reinicien los servicios de apache y reconozca a phpmyadmin.

- `sudo service apache2 restart`

después verificar si esta el archivo info.php para su visualización si no se encuentra hacer un archivo con extensión php ejemplo:

hacer un archvo .php

- `<? Info(); ?> -- dentro su contenido`



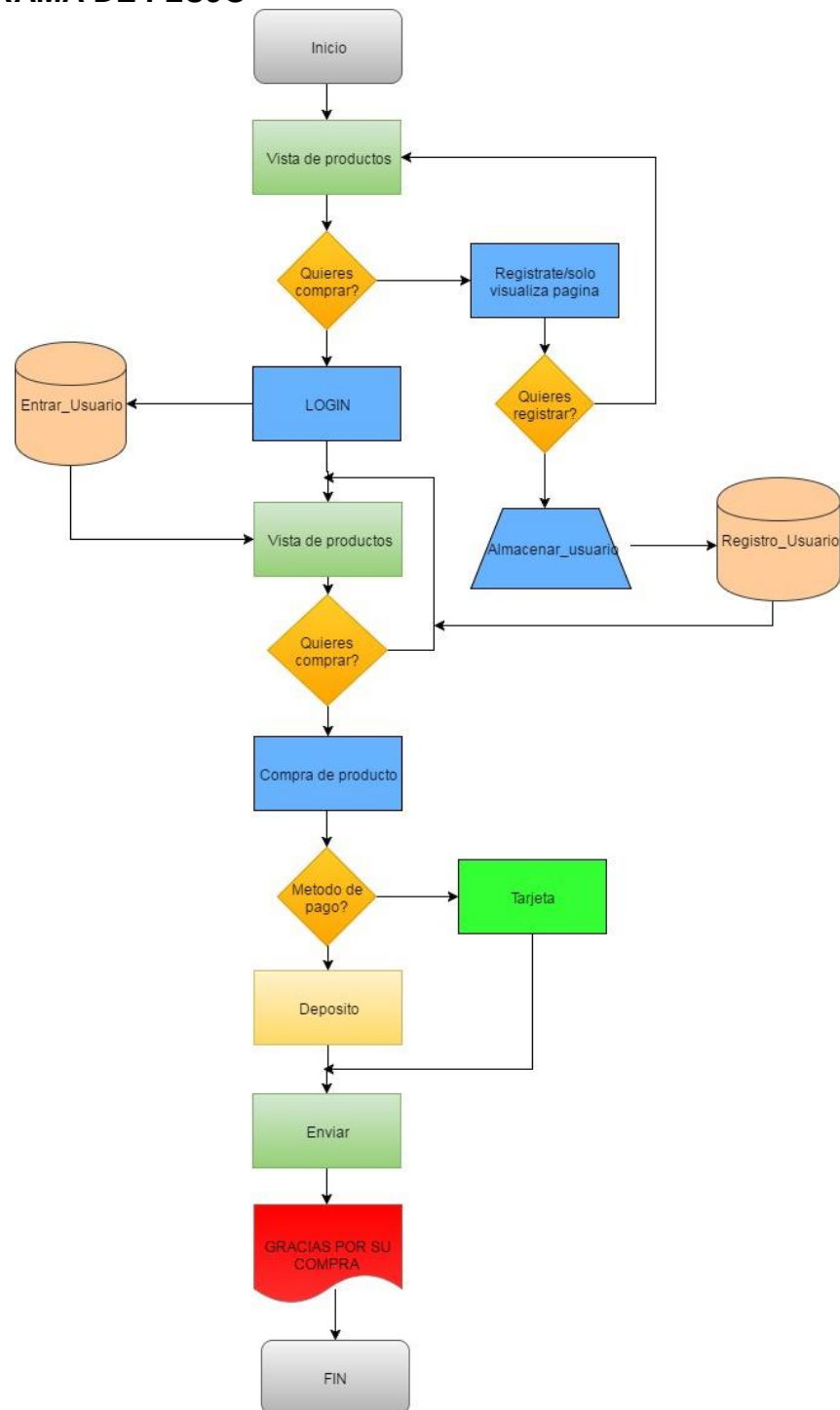
## 1.2. Instalando MYSQL SERVER

- `sudo apt-get install mysql-server mysql-common mysql-client`

- ## Manejo de base de datos con php

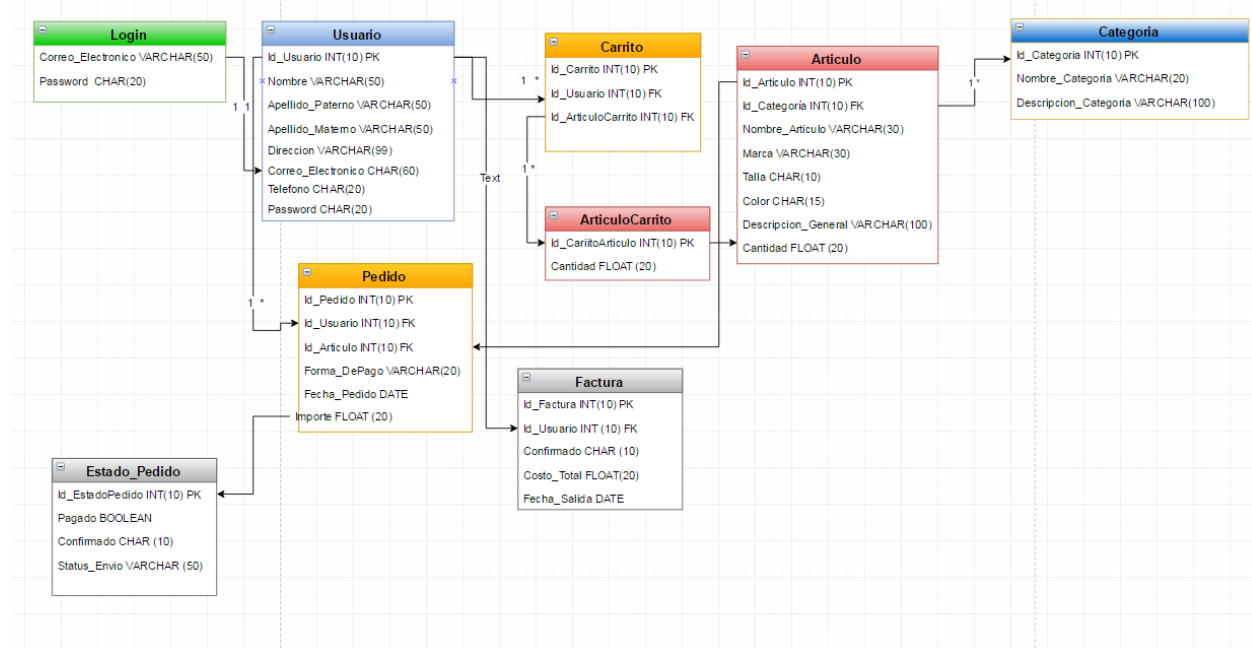
- art para reiniciar servicios

## 2. DIAGRAMA DE FLUJO

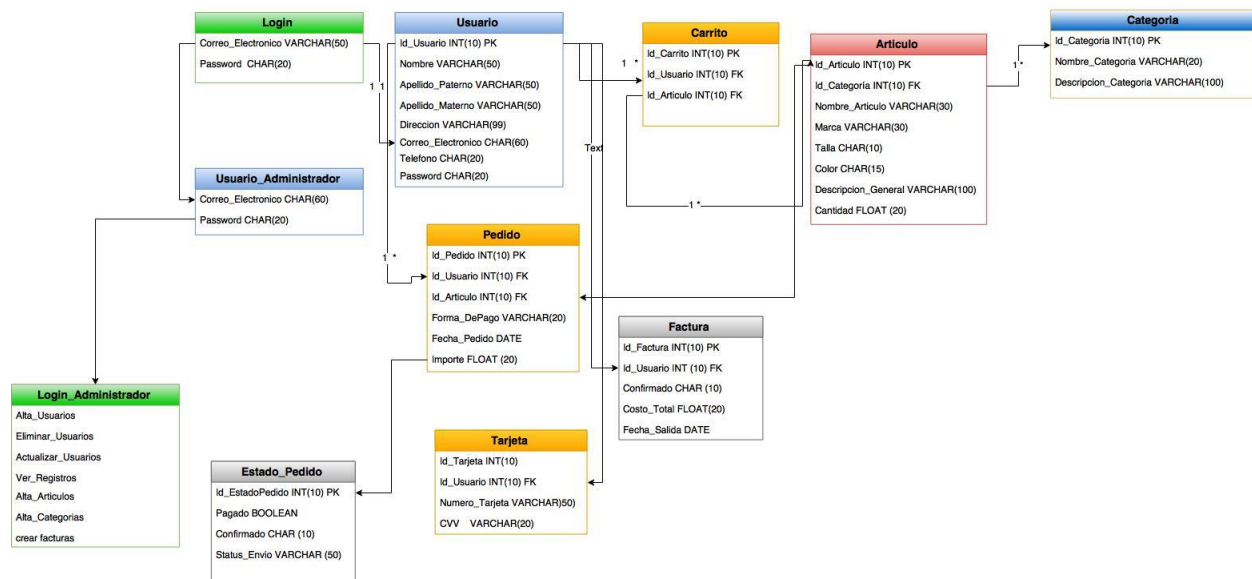


## 2.1. Creación de la base de datos (UML)

1er Diagrama UML que fue su creación.



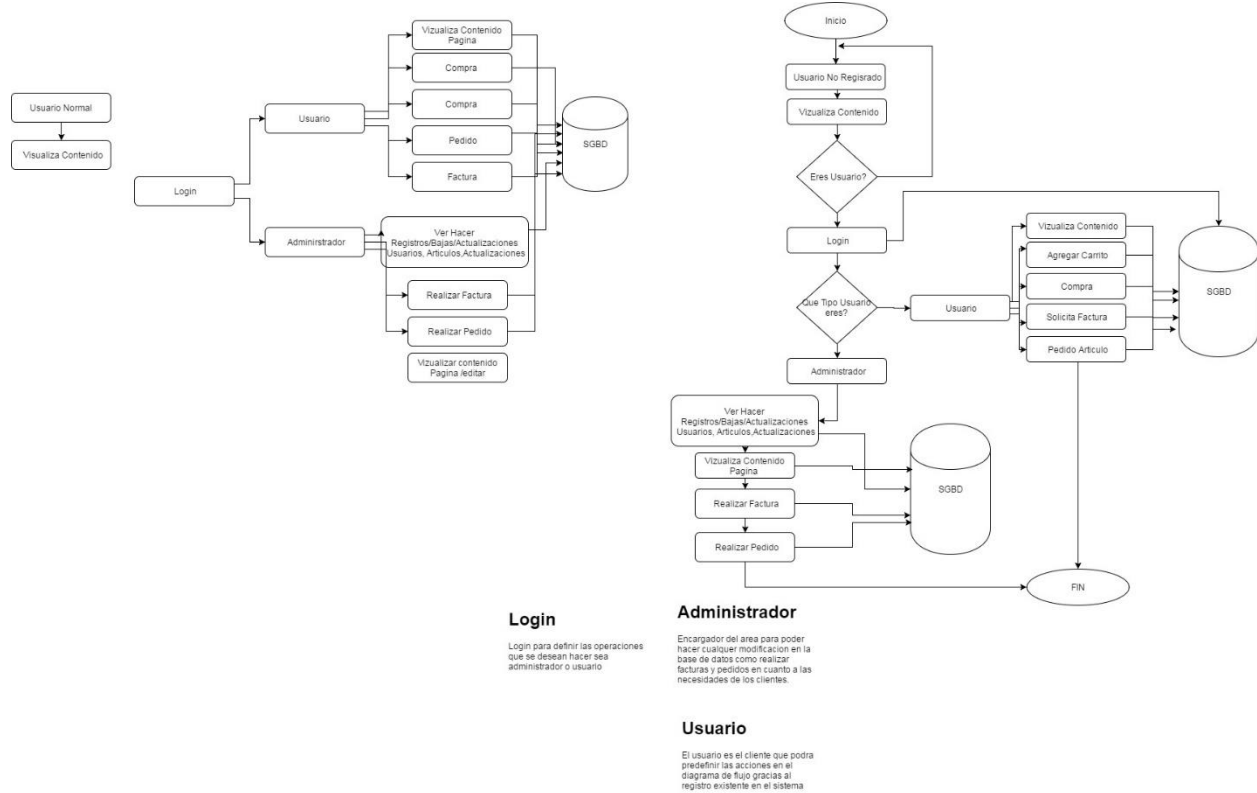
Después surgió una modificación en el sistema quedando de la siguiente manera para su funcionamiento



Se agregaron los campos:

- Usuario\_Administrador para las operaciones del sistema en la tabla Login\_Administrador.
- Tarjeta para el pago de la transacción.

## 2.2. Diagrama de Microservicios.



## 2.3. \_SESIONES

Maneje las sesiones en php para validar en la base de datos un usuario y su contraseña pero surgió algún problema con la consulta. Esto fue antes de utilizar slimframework

```
<?php
include('ConexionDataBase.php');

$Correo_Electronico=$_POST["Correo_Electronico"]; //Acedemos a las variables
$Contraseña=$_POST["Contraseña"];

/* Realizamos una consulta por cada tabla para buscar en que tabla se encuentra
el usuario que está intentando acceder */
$Usuario = $Conexion->query("SELECT * FROM usuario WHERE Correo_Electronico = '$Correo_Electronico' AND
Contraseña = '$Contraseña'");
$Administrador = $Conexion->query("SELECT * FROM usuario_administrador WHERE Correo_Electronico = '$
Correo_Electronico' AND Contraseña = '$Contraseña'");

if(mysqli_num_rows($Usuario) > 0) /* Ahora comprobamos que variable contiene al usuario*/
{
    session_start(); /* Si entra en este if significa que el que intenta acceder es un alumno, por lo
tanto le creamos una sesión */

    $_SESSION['Usuario']="$Correo_Electronico";
    header("Location: logueado.php"); /* Nos dirigimos al espacio de los alumnos usando header que nos
redireccionará a la página que le indiquemos */

    /* terminamos la ejecución ya que si redireccionamos ya no nos interesa
seguir ejecutando código de este archivo */
}
else if (mysqli_num_rows($Administrador) > 0)
{
    session_start();
    $_SESSION['Administrador']="$Correo_Electronico";
    header("Location: Logueado_Admin.php"); # code...
    exit();
}
else
{
    echo "<script type='text/javascript'>
alert(\"Usuario incorrecto\");
</script>";
}
?>
```

La solución de este error fue este

```
$Usuario = $Conexion->query("SELECT Correo_Electronico,Contrasena FROM usuario WHERE Correo_Electronico = '$Correo_Electronico' AND Contraseña = '$Contraseña'");  
$Administrador = $Conexion->query("SELECT Correo_Electronico,Contrasena FROM usuario_administrador WHERE Correo_Electronico = '$Correo_Electronico' AND Contraseña = '$Contraseña'");
```



### 3. Instalación composer

Instalación en Linux

1. Abre una consola de comandos y accede al directorio raíz de tu proyecto Symfony2:
2. Ejecuta el siguiente comando:

- `Curl -s https://getcomposer.org/installer | php`

Si todo ha funcionado bien, en el directorio raíz de tu proyecto verás un nuevo archivo llamado `composer.phar` y después comprobar que se ha instalado correctamente el siguiente comando:

- `Php composer.phar`

Para instalar Composer globalmente, mueve el archivo `composer.phar` a algún directorio ejecutable del sistema, como por ejemplo:

- `Sudo mv composer.phar/usr/local/bin/composer`

Comprueba que todo funciona bien ejecutando el comando `composer` en la terminal de Linux. Después poner `composer self-update` para que actualice a la versión más reciente.

- `Sudo composer self-update`

#### 3.1. Creación de proyectos SlimFramework 2.6 en Linux mint

Entrar a esta dirección para descargar el slimframework versión 2.6.2

- <https://www.versioneye.com/php/slim:slim/2.6.2>

pegar en la dirección `htdocs/www/Aquí raíz de archivo localhost`





A continuación en navegador escribir localhost y donde esta ubicada la carpeta slimframework y saldrá por default un index que es este.

Editamos el archivo

Hasta abajo del archivo saldrá esta ruta que get es un método para recolectar datos entonces:

```
$app->get('/hello/:name', function ($name) {  
    echo "Hello, $name";  
});
```

En el navegador nos vamos ala ruta de Slimframerwork por ejemplo:

<https://localhost/index.php/hello/jose>

En la salida de el navegador nos compilara esto: “Hello Jose”

### 3.2. Conexión con la BD y slimframework:

Necesitamos editar el archivo apache2.conf nos vamos ala siguiente dirección y abrimos el terminal

```
vi /etc/apache2/apache2.conf

Replace "AllowOverride None" to "AllowOverride All"

<Directory /usr/share>
AllowOverride all
Require all granted
</Directory>

y reiniciamos el servicio de apache2
```

Esto es para que Slimframework funciones con todas las rutas que le indicamos de esta manera siguiente veremos para utilizarla.

```
require 'Slim/Slim.php';
require 'includes/meekrodb.2.2.class.php';
require 'includes/functions.php';

\Slim\Slim::registerAutoloader();
DB::$user = 'root';
DB::$password = 'linux2017';
DB::$dbName = 'tiendaenlinea';
DB::$host = 'localhost';
DB::$encoding = 'utf8';
```

De la siguiente manera incluimos los dos archivos

- meekrodb.2.2.class.php
- functions.php

y los agregamos en la carpeta includes

Despues creamos las rutas correspondientes para su consulta con el método GET para obtener los datos registrado en la base de datos.

```

$app->get('/Registros_Usuarios',function () use ($app)
{
    $query = DB::query("Select * from Usuario");
    $app->response()->header('content-Type','application/json; charset=utf8');
    array_walk ($query,'utf8_encode_array');
    echo json_encode($query);
});
/*****Registros Articulos*****/
$app->get('/Registros_Articulos',function () use ($app)
{
    $query = DB::query("Select * from Articulo");
    $app->response()->header('content-Type','application/json; charset=utf8');
    array_walk ($query,'utf8_encode_array');
    echo json_encode($query);
});
/*****Registros Categorias*****/
$app->get('/Registros_Categorias',function () use ($app)
{
    $query = DB::query("Select * from Categoria");
    $app->response()->header('content-Type','application/json; charset=utf8');
    array_walk ($query,'utf8_encode_array');
    echo json_encode($query);
});

```

Y algunas de las rutas para su insercion de datos con el método POST.

```

$app->post('/Alta_Registros',function () use ($app)
{
    $Id_Usuario = $_POST ['Id_Usuario'];
    $Nombre = $_POST ['Nombre'];
    $Apellido_Paterno = $_POST ['Apellido_Paterno'];
    $Apellido_Materno = $_POST ['Apellido_Materno'];
    $Direccion = ($_POST ['Direccion']);
    $Correo_Electronico = $_POST ['Correo_Electronico'];
    $Telefono = $_POST ['Telefono'];
    $Contrasena = $_POST ['Contrasena'];

    $Insertar = DB::query("INSERT INTO Usuario (Id_Usuario,Nombre,Apellido_Paterno,Apellido_Materno
    ,Direccion,Correo_Electronico,Telefono,Contrasena) VALUES('$Id_Usuario','$Nombre','$
    Apellido_Paterno','$Apellido_Materno','$Direccion','$Correo_Electronico','$Telefono','$
    Contrasena')");

    $app->response()->header('content-Type','application/json; charset=utf8');

    echo json_encode($Insertar);
});
/*****Alta Articulos*****/
$app->post('/Alta_Articulos',function () use ($app)
{
    $Id_Articulo = $_POST ['Id_Articulo'];
    $Id_Categoria = $_POST ['Id_Categoria'];
    $Nombre_Articulo = $_POST ['Nombre_Articulo'];
    $Marca = $_POST ['Marca'];
    $Talla= $_POST ['Talla'];
    $Color = $_POST ['Color'];
    $Descripcion = $_POST ['Descripcion'];
    $Cantidad = $_POST ['Cantidad'];

```

### 3.3. Descarga Chrome para su extensión POSTMAN

Abrir terminal y escribir el siguiente comando

- `wget -q -O - https://dl-ssl.google.com/linux/linux_signing_key.pub | sudo apt-key add -`

este comando siguiente para agregarlo al repositorio.

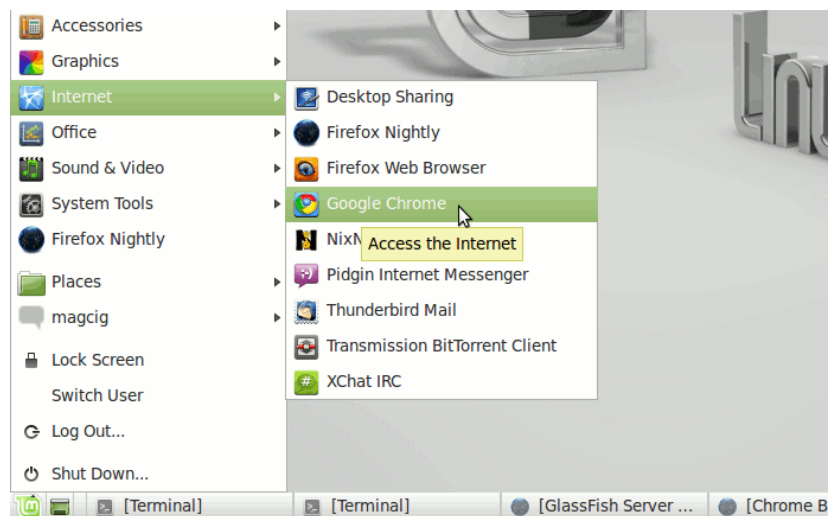
- `sudo sh -c 'echo "deb http://dl.google.com/linux/chrome/deb/ stable main" >> /etc/apt/sources.list.d/google-chrome.list'`

Ahora actualize el repositorio e instale el navegador.

- `sudo apt-get update`

Instalación de versión estable

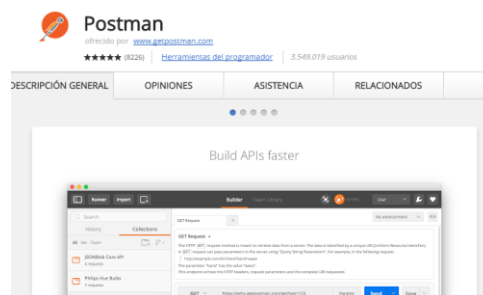
- `sudo apt-get install google-chrome-stable`



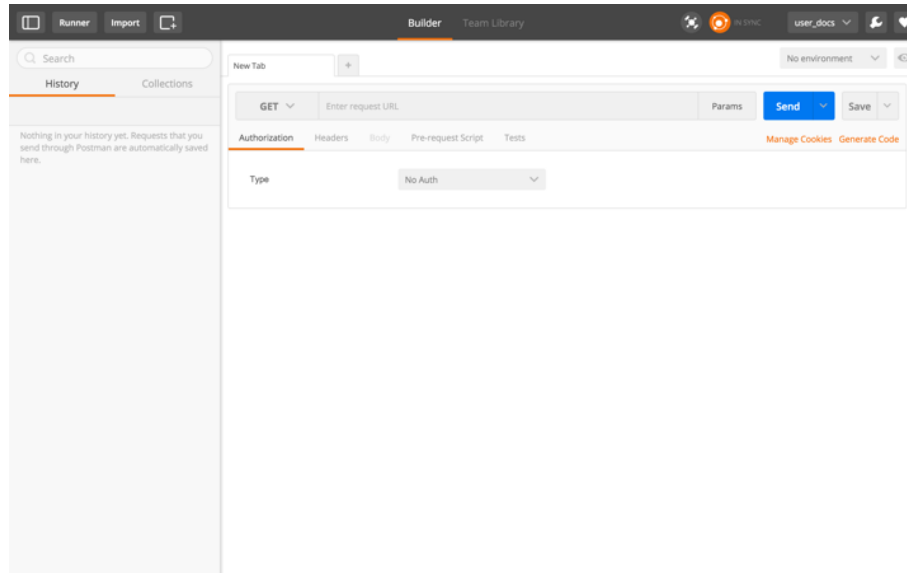
### 3.4. AGREGAR EXTENSION POSTMAN

ir a chrome configuracion -> extensions ->

Despues teclear en la barra POSTMAN y saldrá



## Abriendo la aplicación.

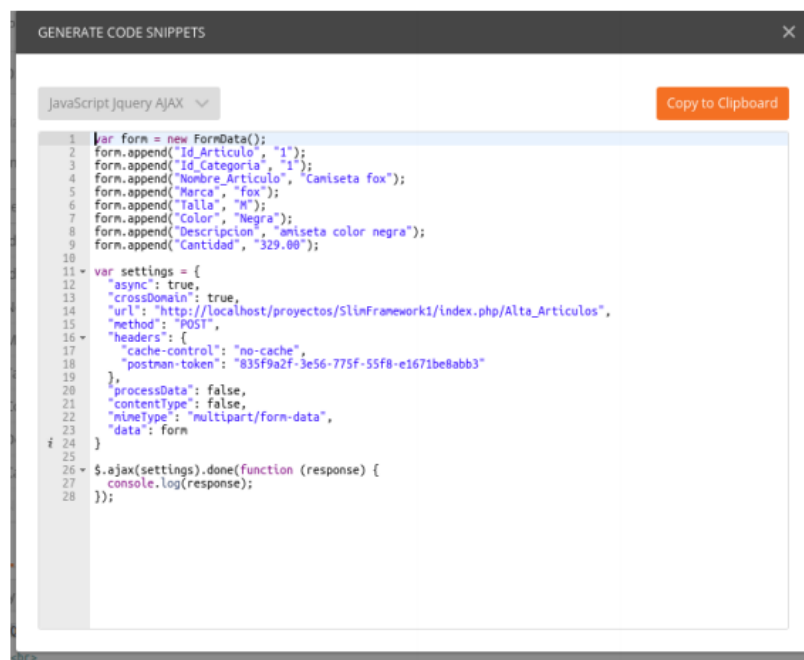


Visualizamos la aplicación dentro ahí un cuadro que dice get en el insertamos la ruta de la api para poder leer el contenido entre ellos esta el GET, PUT, POST,DELETE Y PATCH.

Después seleccionamos la opción de body y dentro saldrán campos vacíos para poner los nombres de las tablas como su valor una vez llenado esto le damos en send.

Después le damos donde dice code para su devolución de código en diferentes lenguajes en este caso yo los solicite con javascript.

Resultados de mis consultas e inserciones.



## INTELIGENCIA ARTIFICIAL

### Insertión en tabla artículos

GENERATE CODE SNIPPETS

JavaScript jQuery AJAX

Copy to Clipboard

```
1 var form = new FormData();
2
3 var settings = {
4   "async": true,
5   "crossDomain": true,
6   "url": "http://localhost/proyectos/SlimFramework1/Registros_Articulos",
7   "method": "GET",
8   "headers": {
9     "cache-control": "no-cache",
10    "postman-token": "b07cfe99-6c4b-5045-7971-2aa30573d525"
11  },
12  "processData": false,
13  "contentType": false,
14  "mimeType": "multipart/form-data",
15  "data": form
16 }
17
18 $.ajax(settings).done(function (response) {
19   console.log(response);
20 });
```

Aquí se consulto los registros existentes después sigue el resultado obtenido

GET http://localhost/proyectos/SlimFramework1/Registros\_Articulos

Authorization Headers Body Pre-request Script Tests

Type No Auth

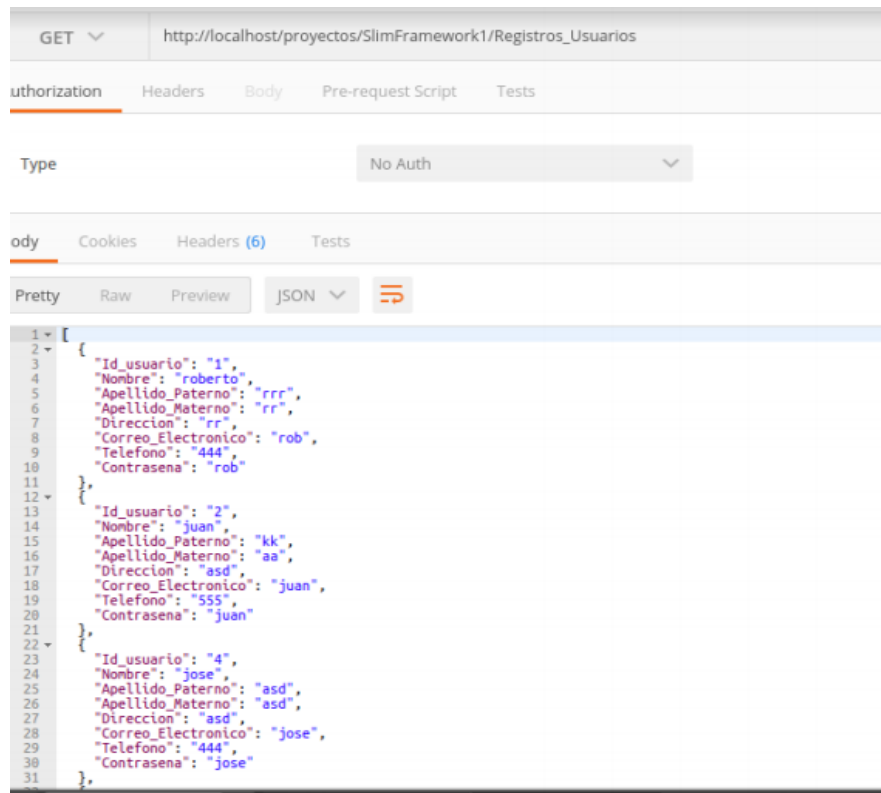
Body Cookies Headers (6) Tests

Pretty Raw Preview JSON

```
1 [
2   {
3     "Id_Articulo": "1",
4     "Id_Categoria": "1",
5     "Nombre_Articulo": "Camiseta fox",
6     "Marca": "fox",
7     "Talla": "M",
8     "Color": "Negra",
9     "Descripcion": "amiseta color negra",
10    "Cantidad": "329"
11  }
12 ]
```

## INTELIGENCIA ARTIFICIAL

Después hice lo mismo para los usuarios



The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** http://localhost/proyectos/SlimFramework1/Registros\_Usuarios
- Authorization:** No Auth
- Body:** Pretty, JSON
- Response:** A JSON array of three user objects.

```
1 {  
2   {  
3     "Id_usuario": "1",  
4     "Nombre": "roberto",  
5     "Apellido_Paterno": "rrr",  
6     "Apellido_Materno": "rr",  
7     "Direccion": "rr",  
8     "Correo_Electronico": "rob",  
9     "Telefono": "444",  
10    "Contraseña": "rob"  
11  },  
12  {  
13    "Id_usuario": "2",  
14    "Nombre": "juan",  
15    "Apellido_Paterno": "kk",  
16    "Apellido_Materno": "aa",  
17    "Direccion": "asd",  
18    "Correo_Electronico": "juan",  
19    "Telefono": "555",  
20    "Contraseña": "juan"  
21  },  
22  {  
23    "Id_usuario": "4",  
24    "Nombre": "jose",  
25    "Apellido_Paterno": "asd",  
26    "Apellido_Materno": "asd",  
27    "Direccion": "asd",  
28    "Correo_Electronico": "jose",  
29    "Telefono": "444",  
30    "Contraseña": "jose"  
31  }  
}
```

## CONCLUSIONES:

Descubrí una manera mas fácil de insertar datos con una nueva tecnología llamada slimframework ya que proporciona mas flexibilidad en manejo de datos como su interfaz mas facil en llamado de datos me ah gustado el manejo porque aprendi una manera mas en programación ya que me intereso un lenguaje que se llama laravel que trabaja con frameworks además que utilizamos una red social de programadores en una plataforma llamada "git hub" además que trabaje con sesiones en php que nunca había manejado me intereso la web.