

# MANUAL DE INSTALACIÓN

## CONTENIDO

|  |    |
|--|----|
| INTRODUCCIÓN .....   | 5  |
| REQUERIMIENTOS MÍNIMOS DEL SISTEMA .....                         | 5  |
| PROGRAMAS NECESARIOS .....                                       | 5  |
| COMPOSER.....  | 5  |
| XAMMP.....   | 7  |
| NODE.JS .....  | 9  |
| GOOGLE CHROME.....   | 11 |
| INSTALACIÓN .....  | 12 |
| Uso de XAMPP para ubicar la carpeta del proyecto .....           | 12 |
| Instalación de Composer en la carpeta del proyecto .....         | 14 |
| Creación de la base de datos .....                               | 18 |
| Modificación del archivo example.env .....                       | 19 |
| Configuración del repositorio de MAIL.....                       | 22 |
| Configuración de Pusher para notificaciones en tiempo real ..... | 26 |
| Cambiar nombre archivo exmaple.env.....                          | 31 |
| Llave de acceso .....  | 33 |
| Creación ruta a carpeta Storage .....                            | 33 |
| Instalación de Vue de Node.js en la carpeta del proyecto .....   | 33 |
| Migraciones.....   | 35 |

|                           |    |
|---------------------------|----|
| Seeders .....             | 37 |
| Correr el aplicativo..... | 38 |
| Acceso al login.....      | 39 |
| REFERENCIAS .....         | 40 |

## ILUSTRACIONES

|  |    |
|--|----|
| Ilustración 1 Página principal de Composer .....                     | 5  |
| Ilustración 2 Documentación de instalación de Composer .....         | 6  |
| Ilustración 3 Instalación de Composer .....                          | 7  |
| Ilustración 4 Página principal de XAMPP .....                        | 8  |
| Ilustración 5 Recomendaciones antes de la instalación de XAMPP ..... | 9  |
| Ilustración 6 Proceso de la instalación de XAMPP .....               | 9  |
| Ilustración 7 Página principal de node.js .....                      | 10 |
| Ilustración 8 Proceso de instalación de Node.js .....                | 11 |
| Ilustración 9 Página de descarga de Google Chrome .....              | 12 |
| Ilustración 10 Ruta de la carpeta principal de Calisoft .....        | 13 |
| Ilustración 11 Interfaz XAMPP .....                                  | 13 |
| Ilustración 12 Interfaz XAMPP con servicios activados .....          | 14 |
| Ilustración 13 Ventana ejecutar.....                                 | 14 |
| Ilustración 14 Simbolo del sistema .....                             | 15 |
| Ilustración 15 Ejecución del comando “cd..” .....                    | 15 |
| Ilustración 16 Ejecución del comando “cd..” .....                    | 15 |
| Ilustración 17 Cmd ubicación de la carpeta principal.....            | 16 |

|   |    |
|---|----|
| Ilustración 18 Composer install en CMD .....  | 16 |
| Ilustración 19 Procedimiento de la instalación de Composer en carpeta principal ..... | 16 |
| Ilustración 20 Finalización de Composer en carpeta principal.....                     | 17 |
| Ilustración 21 Comando Composer update en CMD .....                                   | 17 |
| Ilustración 22 Actualización Composer en CMD .....                                    | 18 |
| Ilustración 23 Creación base de datos.....  | 19 |
| Ilustración 24 Ubicación archivo "example.env" .....                                  | 20 |
| Ilustración 25 Contenido archivo "example.env" .....                                  | 21 |
| Ilustración 26 Configuración de MAIL .....  | 22 |
| Ilustración 27 Página principal de Mailtrap.....                                      | 22 |
| Ilustración 28 Inicio de sesión Mailtrap.....   | 23 |
| Ilustración 29 Backend Mailtrap .....   | 23 |
| Ilustración 30 Administración de mensajes de Mailtrap .....                           | 24 |
| Ilustración 31 Apartado Integrations Mailtrap.....                                    | 25 |
| Ilustración 32 Atributos de Mailtrap en Laravel .....                                 | 26 |
| Ilustración 33 Apartado de Mail en archivo example.env .....                          | 26 |
| Ilustración 34 Página principal de Pusher .....                                       | 27 |
| Ilustración 35 Welcome to Pusher.....   | 28 |
| Ilustración 36 Modificación de Pusher .....   | 29 |
| Ilustración 37 Backend Pusher.....  | 30 |
| Ilustración 38 Pestañas Pusher .....  | 30 |
| Ilustración 39 App Keys Pusher.....   | 31 |
| Ilustración 40 Configuración Pusher en archivo example.env .....                      | 31 |

|   |    |
|---|----|
| Ilustración 41 Cambiar nombre del archivo example.env .....               | 32 |
| Ilustración 42 Archivo. env .....   | 32 |
| Ilustración 43 Php artisan key:generate .....                             | 33 |
| Ilustración 44 Proceso exitoso del comando Php artisan storage:link ..... | 33 |
| Ilustración 45 Proceso de instalación de Vue .....                        | 34 |
| Ilustración 46 Finalización exitosa de Vue en carpeta del proyecto .....  | 35 |
| Ilustración 47 Comando php artisan migrate.....                           | 36 |
| Ilustración 48 Migraciones exitosas.....                                  | 36 |
| Ilustración 49 Comando php artisan db:seed exitoso .....                  | 37 |
| Ilustración 50 Registros de la tabla TBL_Usuarios .....                   | 37 |
| Ilustración 51 Datos de los seeders .....                                 | 38 |
| Ilustración 52 Ip del servidor local.....                                 | 39 |
| Ilustración 53 Página principal de Calisoft .....                         | 39 |

# INTRODUCCIÓN

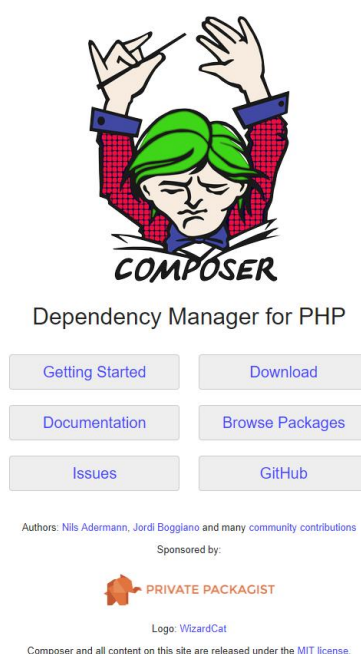
Este documento es una guía por la cual el técnico que dará soporte a la aplicación Calisoft, debe tener en cuenta a la hora de realizar la precisa instalación para que todos los componentes y extensiones del aplicativo funcionen correctamente. A continuación, se darán a conocer los programas necesarios para este proceso.

## REQUERIMIENTOS MÍNIMOS DEL SISTEMA

### PROGRAMAS NECESARIOS

#### COMPOSER

“Composer es una herramienta para la administración de dependencias en PHP. Le permite declarar las bibliotecas de las que depende su proyecto y las administrará (las instalará / actualizará).”(Composer, n.d.) Para realizar la instalación es necesario dirigirse a la página; <https://getcomposer.org/> , en la ilustración 1 podrá observar como es la interfaz.



*Ilustración 1 Página principal de Composer*

Dentro de la interfaz, usted deberá seleccionar la opción de “Download” y se redirigirá a la siguiente página;

[Home](#) | [Getting Started](#) | [Download](#) | [Documentation](#) | [Browse Packages](#)

## Download Composer

### Windows Installer

The installer will download composer for you and set up your PATH environment variable so you can simply call `composer` from any directory.

Download and run [Composer-Setup.exe](#) - it will install the latest composer version whenever it is executed.

### Command-line installation

To quickly install Composer in the current directory, run the following script in your terminal. To automate the installation, use [the guide on installing Composer programmatically](#).

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('SHA384', 'composer-setup.php') === '544e09ee996cdf60ece3804abc52599c22b1f40f4323403c44d4'
php composer-setup.php
php -r "unlink('composer-setup.php');"
```

This installer script will simply check some php.ini settings, warn you if they are set incorrectly, and then download the latest composer.phar in the current directory. The 4 lines above will, in order:

- Download the installer to the current directory
- Verify the installer SHA-384 which you can also [cross-check here](#)
- Run the installer
- Remove the installer

**WARNING:** Please do not redistribute the install code. It will change with every version of the installer. Instead, please link to this page or check [how to install Composer programmatically](#).

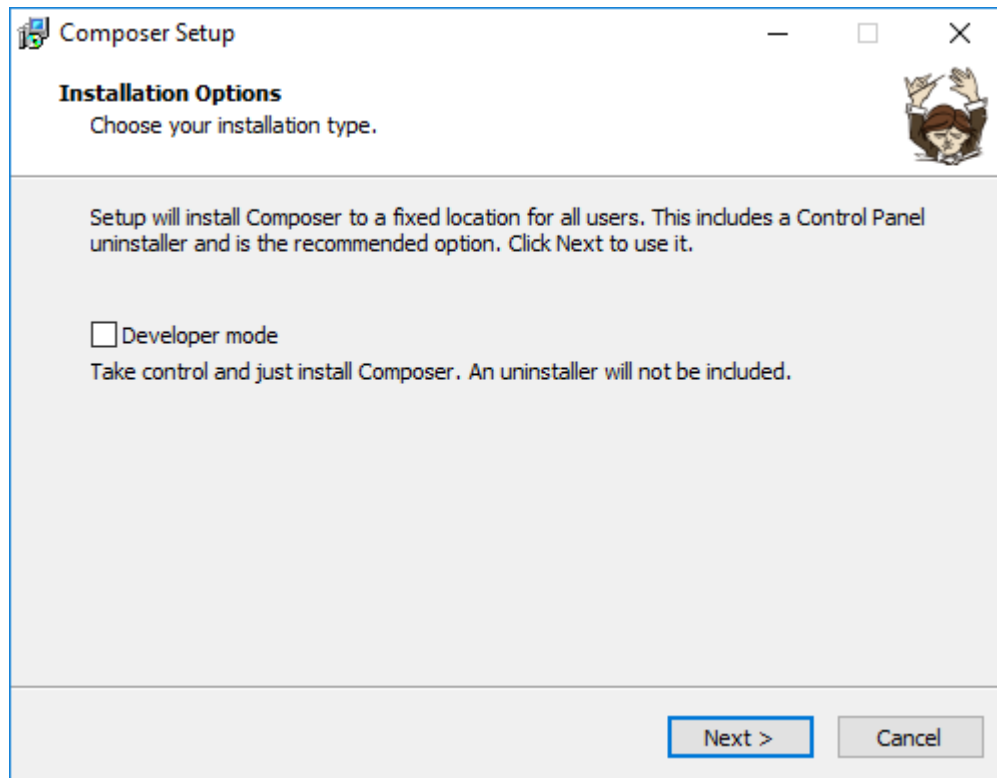
### Installer Options

#### --install-dir

You can install composer to a specific directory by using the `--install-dir` option and providing a target directory. Example:

*Ilustración 2 Documentación de instalación de Composer*

Se procede a dar clic en el hipervínculo “Composer.Setup.exe” y se iniciará la descarga automáticamente. Una vez descargado el programa, se debe ejecutar en modo administrador para evitar errores de instalación y aparecerá la siguiente ventana;



*Ilustración 3 Instalación de Composer*

Es necesario que omita la opción de “Developer mode” y proceda a pulsar el botón “Next”. Debe realizar el mismo procedimiento en las siguientes ventanas hasta que finalice la instalación.

Una vez finalizada la instalación se cierra la ventana y de esta forma Composer queda instalado en el ordenador.

## **XAMMP**

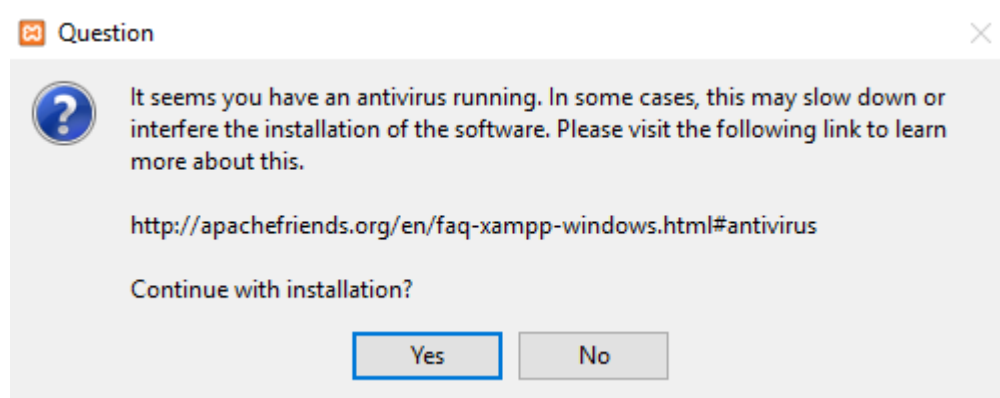
“El objetivo de XAMPP es crear una distribución fácil de instalar para desarrolladores que se están iniciando en el mundo de Apache. XAMPP viene configurado por defecto con todas las opciones activadas”. (Apache Friends, 2015) Para proceder a su instalación debe ingresar a la siguiente página; <https://www.apachefriends.org/es/index.html> que corresponde a la página principal de XAMPP



Ilustración 4 Página principal de XAMPP

Estando dentro página principal, debe seleccionar la opción que corresponda a su sistema operativo y empezará la descarga automáticamente.

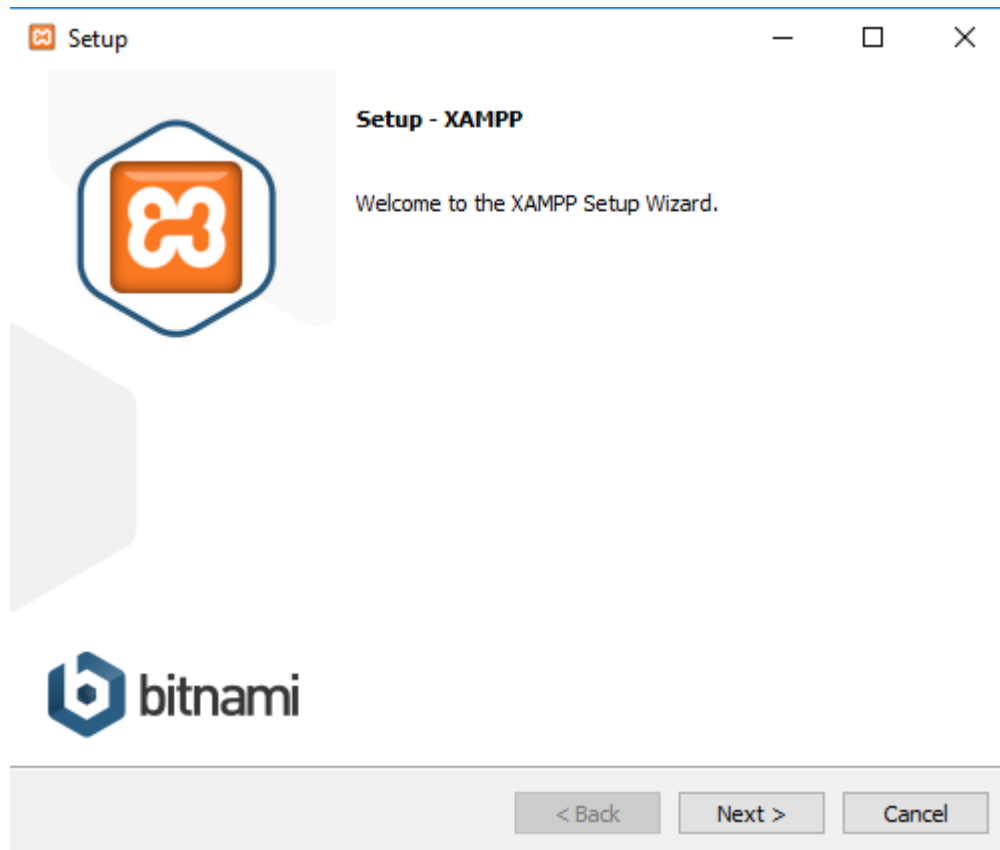
Una vez haya finalizado la descarga de XAMPP, debe ejecutar el programa en modo administrador y aparecerá la siguiente ventana;





*Ilustración 5 Recomendaciones antes de la instalación de XAMPP*

Es una ventana informativa, la cual explica que para evitar algún inconveniente se debe pausar el antivirus que corresponda a su ordenador, al realizar la recomendación anterior se procede a pulsar el botón “Yes”. Una vez realizado este proceso aparecerá la siguiente ventana;



*Ilustración 6 Proceso de la instalación de XAMPP*

En la cual debe dar clic en el botón “Next” y sin modificar ninguna opción debe repetir el proceso hasta que la instalación finalice, al finalizar la instalación el programa quedará disponible en nuestro ordenador.

## **NODE.JS**

“Node.js® es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, npm, es el ecosistema más grande de librerías de código abierto en el mundo”.(Node.js, n.d.)

Para realizar su correcta instalación se debe dirigir a la página principal por medio del siguiente enlace; <https://nodejs.org/es/>



Node.js® es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, npm, es el ecosistema mas grande de librerías de código abierto en el mundo.

### Spectre and Meltdown in the context of Node.js.

## Descargar para Windows (x64)

**8.9.4 LTS**

Recomendado para la mayoría

**9.7.1 Actual**

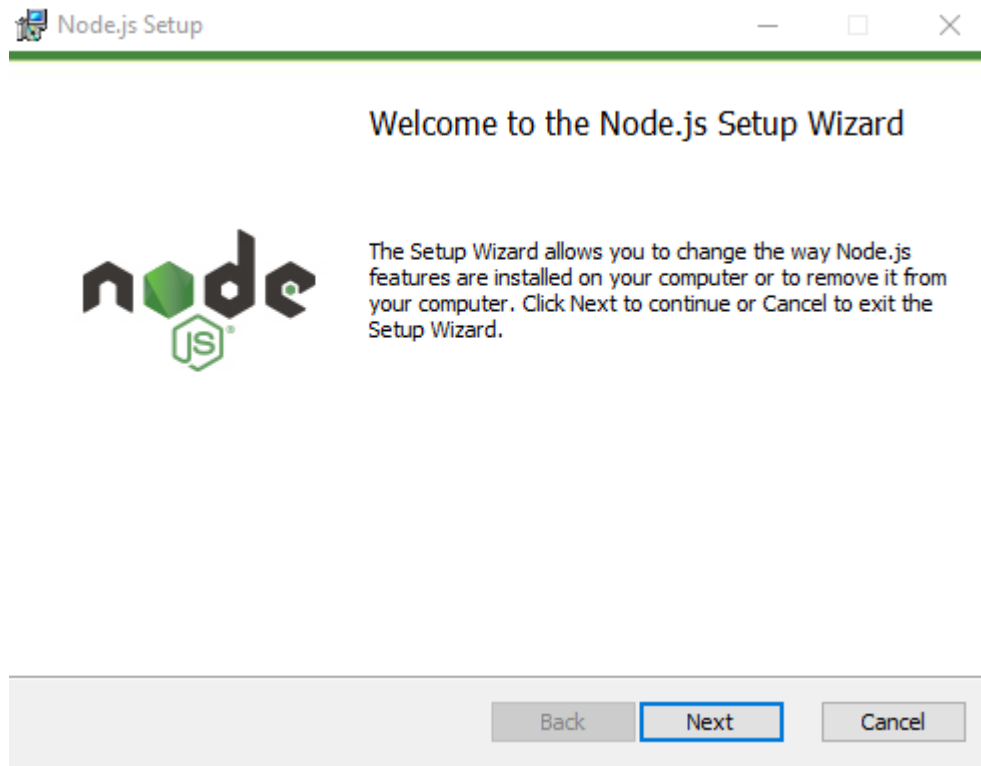
Últimas características

[Otras Descargas](#) | [Cambios](#) | [Documentación del API](#) [Otras Descargas](#) | [Cambios](#) | [Documentación del API](#)

Ó revise la [Agenda de LTS](#).

*Ilustración 7 Página principal de node.js*

En donde seleccionará la opción “8.9.4 LTS (recomendado para la mayoría)” e iniciará la descarga del programa automáticamente. Una vez finalizada la descarga del programa, debe ejecutarlo como administrador y aparecerá la siguiente ventana;



*Ilustración 8 Proceso de instalación de Node.js*

En donde dará clic en el botón “Next” y sin modificar ninguna opción procederá a realizar el mismo proceso hasta finalizar la instalación, de esta forma Node.js quedará instalado en nuestro ordenador.

## **GOOGLE CHROME**

Google Chrome es un navegador diseñado por Google, el cual es requerido en este proyecto, porque Calisoft cuenta con una extensión desarrollada bajo este navegador, dicha extensión es usada en el módulo de Testing para generar las pruebas automatizadas, por esta razón es indispensable para su correcto funcionamiento.

El proceso de instalación de Google Chrome, consiste en dirigirse a la página de su descarga por medio del siguiente enlace; <https://www.google.com.mx/intl/es-419/chrome/#eula>

# Obtén un navegador web gratuito y veloz

Un solo navegador para tu computadora, teléfono y tablet

Descargar Chrome

Para Windows 10/8.1/8/7 64-bit

Descargar Chrome para otra plataforma

Ilustración 9 Página de descarga de Google Chrome

Dentro de la página se debe pulsar el botón “Descargar Chrome” y automáticamente iniciará su descarga. Una vez descargado el programa, se debe iniciar como administrador. El proceso no requiere de alguna decisión por el usuario, sólo se ejecuta el programa y su instalación es automática.

## INSTALACIÓN

### Uso de XAMPP para ubicar la carpeta del proyecto

Una vez se hayan instalado los anteriores programas, se procederá a realizar la instalación del proyecto en el ordenador. El proyecto se puede obtener vía online en el siguiente repositorio Github; <https://github.com/Calisoft2017/ModuloCentral>. Ya con el proyecto en el ordenador, se procederá a almacenar en la carpeta Htdocs de Xampp, esta carpeta se encuentra en la siguiente ruta; C:\xampp\htdocs

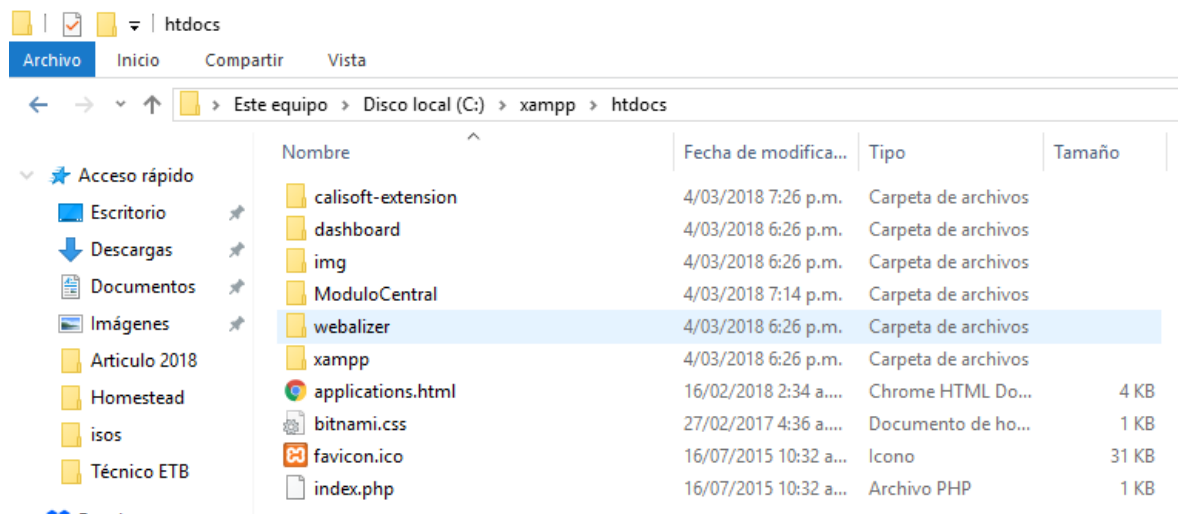


Ilustración 10 Ruta de la carpeta principal de Calisoft

En este caso, “ModuloCentral” es la carpeta del proyecto Calisoft. Una vez ubicada la carpeta procedemos a ejecutar la aplicación Xampp como administrador y pulsamos en el botón “Start” en la casilla de Apache y MySQL, como se ve a continuación;

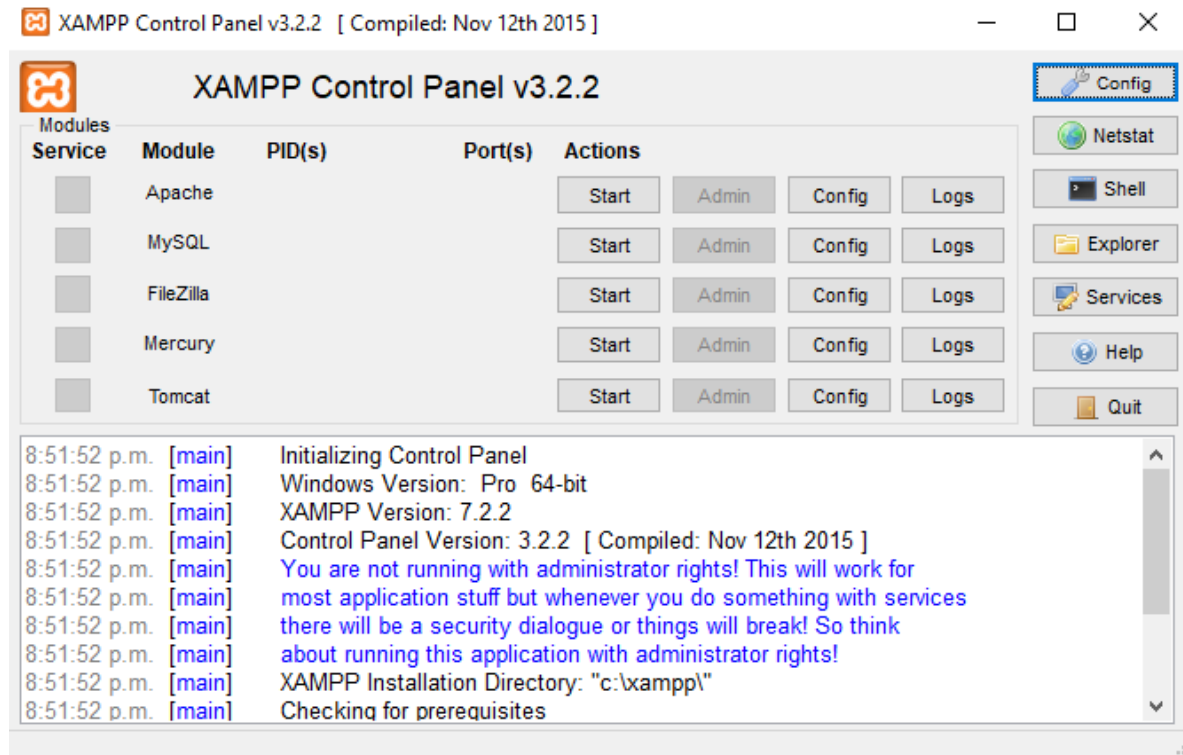


Ilustración 11 Interfaz XAMPP

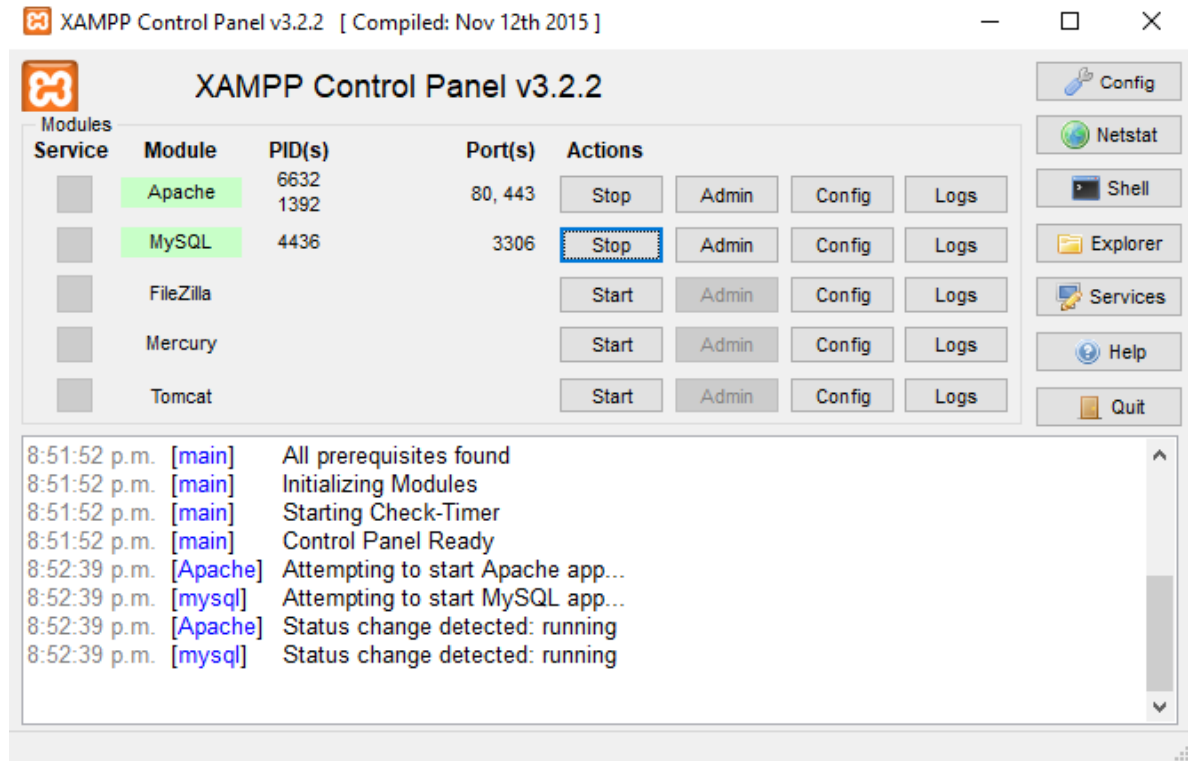


Ilustración 12 Interfaz XAMPP con servicios activados

Las casillas Apache y MySQL deben quedar en color verde, lo que significa que el procedimiento fue exitoso, esto se hace para habilitar un servidor local y encender la base de datos que usará Calisoft.

### Instalación de Composer en la carpeta del proyecto

Ahora, es necesario abrir un símbolo del sistema, para ello realizamos el siguiente comando; “Windows” + “R”

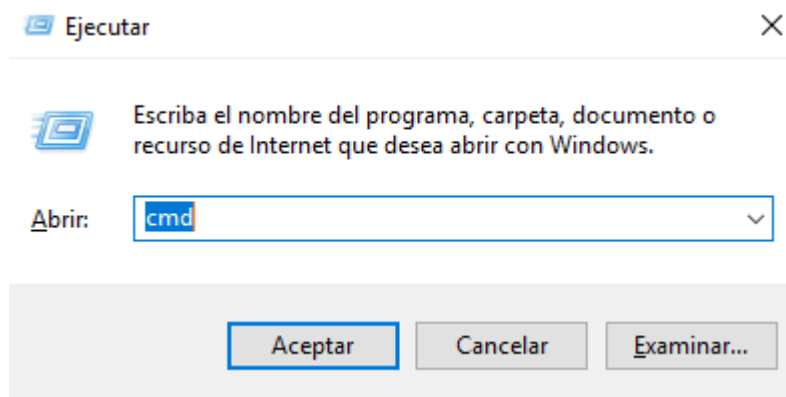
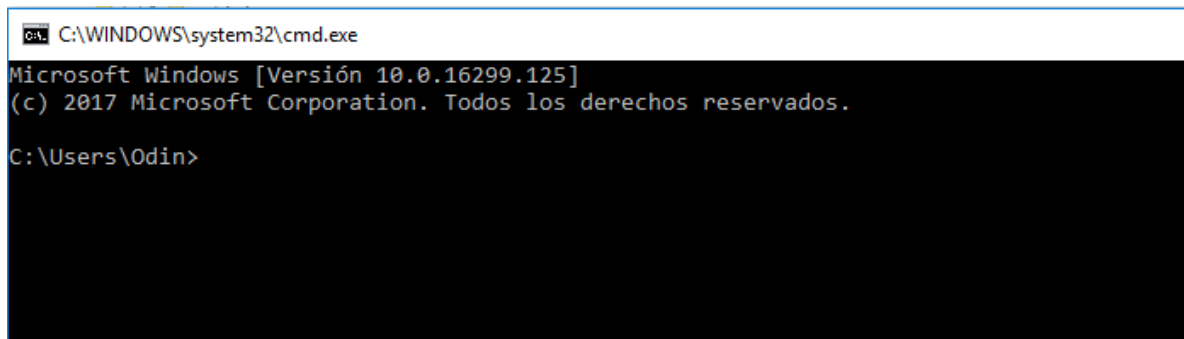


Ilustración 13 Ventana ejecutar

Aparecerá la ventana de “Ejecutar”, digitamos el comando “cmd” y damos clic en el botón “Aceptar”.

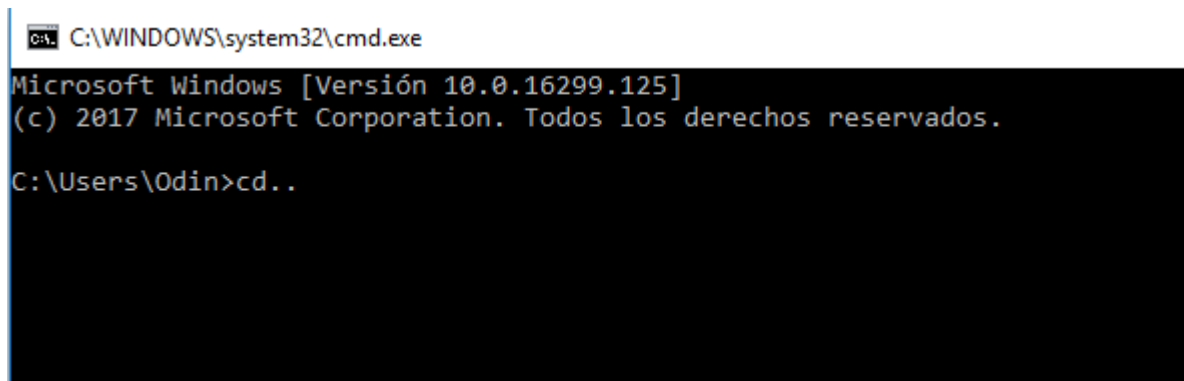


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Versión 10.0.16299.125]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Odin>
```

*Ilustración 14 Simbolo del sistema*

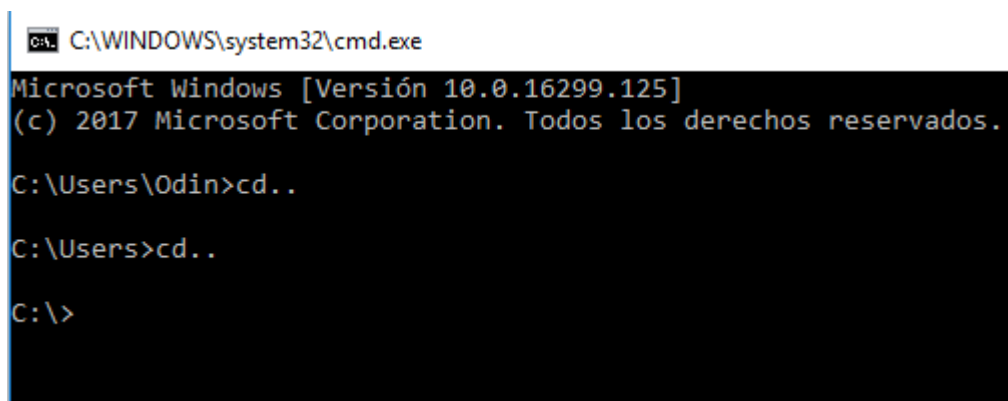
Se nos abrirá el símbolo del sistema ubicado en la carpeta del usuario del ordenador, en este caso el usuario principal es “Odin”, por medio del comando “cd..” retrocedemos al disco local “C”.



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Versión 10.0.16299.125]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Odin>cd..
```

*Ilustración 15 Ejecución del comando “cd..”*



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Versión 10.0.16299.125]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

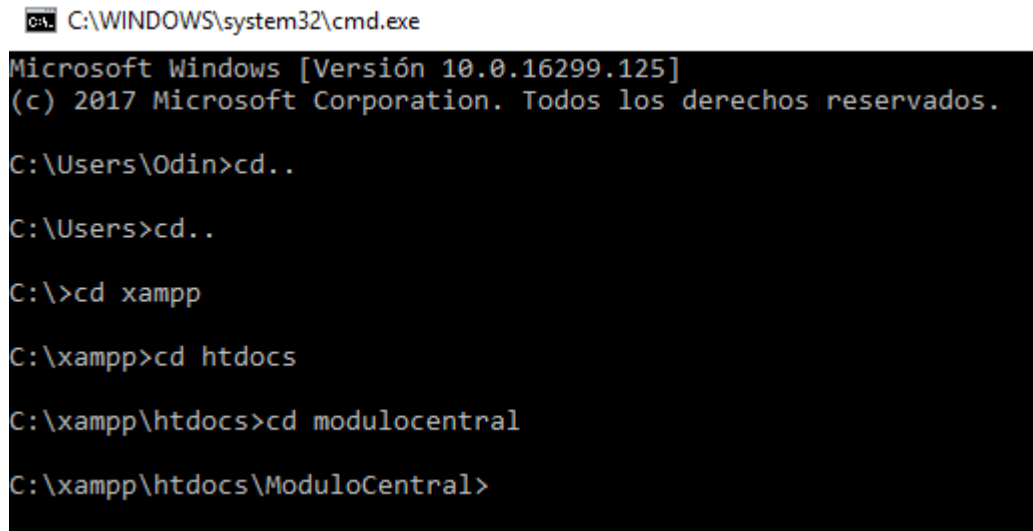
C:\Users\Odin>cd..

C:\Users>cd..

C:\>
```

*Ilustración 16 Ejecución del comando “cd..”*

Una vez ubicados en el disco local “C” procedemos a buscar por medio del comando “cd nombreDeLaCarpeta” la carpeta en donde se alojó el proyecto Calisoft, la ruta en este caso es la siguiente;



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Versión 10.0.16299.125]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Odin>cd..

C:\Users>cd..

C:\>cd xampp

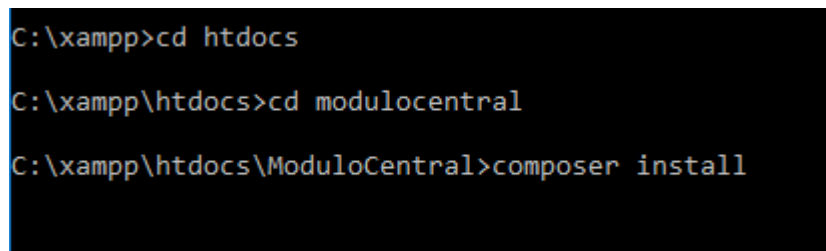
C:\xampp>cd htdocs

C:\xampp\htdocs>cd modulocentral

C:\xampp\htdocs\ModuloCentral>
```

*Ilustración 17 Cmd ubicación de la carpeta principal*

Una vez estemos ubicados en la carpeta del proyecto, procederemos a ejecutar el comando “composer install” y pulsamos la tecla enter.

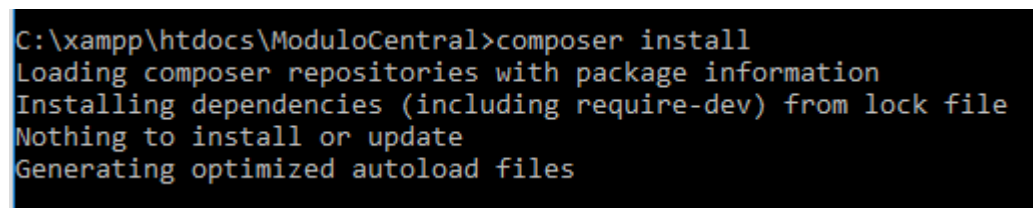


```
C:\xampp>cd htdocs

C:\xampp\htdocs>cd modulocentral

C:\xampp\htdocs\ModuloCentral>composer install
```

*Ilustración 18 Composer install en CMD*



```
C:\xampp\htdocs\ModuloCentral>composer install
Loading composer repositories with package information
Installing dependencies (including require-dev) from lock file
Nothing to install or update
Generating optimized autoload files
```

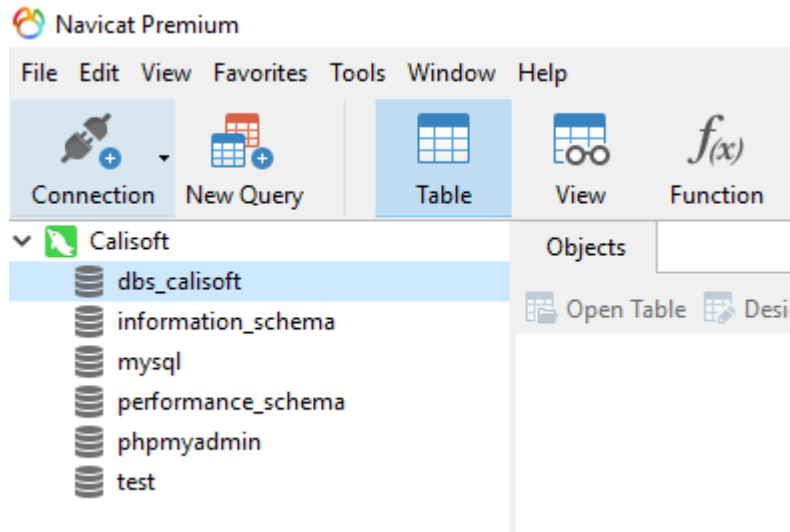
*Ilustración 19 Procedimiento de la instalación de Composer en carpeta principal*

Este comando procederá a ejecutar la instalación de Composer que procederá a instalar automáticamente en nuestra carpeta, esto es indispensable para el correcto funcionamiento de Calisoft.









*Ilustración 23 Creación base de datos*

Tengan en cuenta el usuario y la contraseña con el que se crea la conexión, ya que posteriormente serán necesarios. En este paso sólo es necesario crear la base de datos.

#### **Modificación del archivo example.env**

Ahora, debemos ir a la carpeta principal del proyecto y buscamos el archivo "example.env".

Este equipo > Disco local (C:) > xampp > htdocs > ModuloCentral

|  | Nombre         | Fecha de modifica... | Tipo                | Tamaño |
|--|----------------|----------------------|---------------------|--------|
|  | .git           | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | app            | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | bootstrap      | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | config         | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | database       | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | public         | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | resources      | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | routes         | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | storage        | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | tests          | 4/03/2018 9:41 p.m.  | Carpeta de archivos |        |
|  | vendor         | 4/03/2018 9:55 p.m.  | Carpeta de archivos |        |
|  | .gitattributes | 4/03/2018 9:41 p.m.  | Documento de tex... | 1 KB   |
|  | .gitignore     | 4/03/2018 9:41 p.m.  | Documento de tex... | 1 KB   |
|  | artisan        | 4/03/2018 9:41 p.m.  | Archivo             | 2 KB   |
|  | composer.json  | 4/03/2018 9:41 p.m.  | Archivo JSON        | 2 KB   |
|  | composer.lock  | 4/03/2018 9:55 p.m.  | Archivo LOCK        | 144 KB |
|  | example.env    | 4/03/2018 4:49 p.m.  | Archivo ENV         | 1 KB   |
|  | jsconfig.json  | 4/03/2018 9:41 p.m.  | Archivo JSON        | 1 KB   |
|  | package.json   | 4/03/2018 9:41 p.m.  | Archivo JSON        | 2 KB   |
|  | phpunit.xml    | 4/03/2018 9:41 p.m.  | Documento XML       | 2 KB   |
|  | readme.md      | 4/03/2018 9:41 p.m.  | Archivo MD          | 2 KB   |
|  | server.php     | 4/03/2018 9:41 p.m.  | Archivo PHP         | 1 KB   |
|  | webpack.mix.js | 4/03/2018 9:41 p.m.  | Archivo JavaScript  | 3 KB   |

Ilustración 24 Ubicación archivo "example.env"

Para modificar el archivo "example.env" es necesario abrirlo por medio de un editor de texto, en este caso usaremos el editor de texto Visual Studio Code para editar el archivo, usted puede usar el editor de texto de su preferencia.

```
example.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=
4 APP_DEBUG=true
5 APP_LOG_LEVEL=debug
6 APP_URL=http://localhost
7
8 DB_CONNECTION=mysql
9 DB_HOST=127.0.0.1
10 DB_PORT=3306
11 DB_DATABASE=dbs_calisoft
12 DB_USERNAME=root
13 DB_PASSWORD=
14
15 BROADCAST_DRIVER=log
16 CACHE_DRIVER=file
17 SESSION_DRIVER=file
18 QUEUE_DRIVER=database
19
20 REDIS_HOST=127.0.0.1
21 REDIS_PASSWORD=null
22 REDIS_PORT=6379
23
24 MAIL_DRIVER=smtp
25 MAIL_HOST=smtp.mailtrap.io
26 MAIL_PORT=2525
27 MAIL_USERNAME=26d4e5da1fb4e7
28 MAIL_PASSWORD=f82f27b88f9aeb
29 MAIL_ENCRYPTION=null
30
31 PUSHER_APP_ID=376363
32 PUSHER_APP_KEY=13fba520219a11216ae1
33 PUSHER_APP_SECRET=e6c109277502493e15e1
34 PUSHER_APP_CLUSTER=mt1
35 |
```

*Ilustración 25 Contenido archivo "example.env"*

Se debe modificar la sección "DB\_DATABASE" con forme al nombre con el que se creó la base de datos, el username y el password deben corresponder a su configuración.

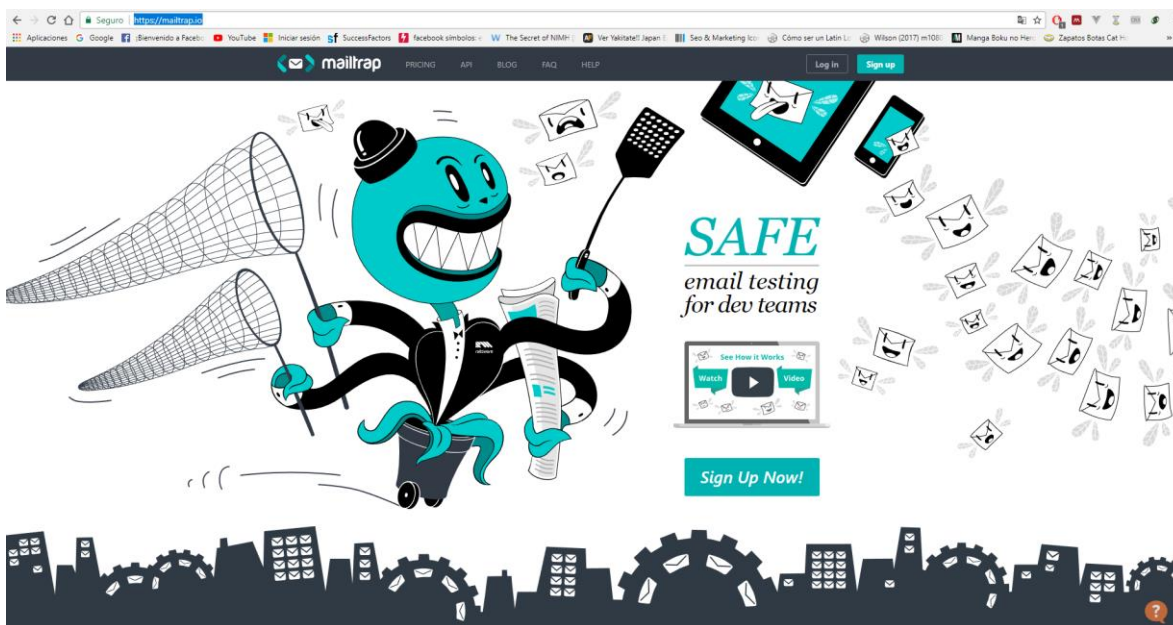
## Configuración del repositorio de MAIL

Se debe configurar la sección de MAIL para que el aplicativo envíe mensajes vía correo, esta configuración debe realizarse de la siguiente manera;

```
MAIL_DRIVER=smtp
MAIL_HOST=smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=26d4e5da1fb4e7
MAIL_PASSWORD=f82f27b88f9aeb
MAIL_ENCRYPTION=null
```

*Ilustración 26 Configuración de MAIL*


Es necesario que se dirija a la página <https://mailtrap.io/> y hacer uso de sus servicios, “Mailtrap es un servidor SMTP falso para que los equipos de desarrollo prueben, vean y compartan los correos electrónicos enviados desde los entornos de desarrollo y puesta en escena sin enviar spam a clientes reales.” (mailtrap, n.d.)




*Ilustración 27 Página principal de Mailtrap*

Una vez estando en la página principal, procederemos a dar clic en el botón “Sing up” el cual nos redirigirá a la siguiente ventana;


## Create Free Forever Account

 Use Google account

 Use Github account

or

[Signup with Your Email](#)

 No credit card needed



 No Contract needed


Ilustración 28 Inicio de sesión Mailtrap

Donde pulsaremos en cualquiera de las tres opciones para crear una cuenta. Una vez cree el perfil e inicie sesión en la página, nos saldrá una ventana como la siguiente;



[Shared Inboxes](#)
[Billing](#)

[Upgrade](#)


Johan Suarez

[Home](#)
[New Company](#)

My Inboxes

[Create Inbox](#)








| Inboxes  | Messages  | Max size | Last message | Action  |
|--|---|----------|--------------|---|
|  Demo inbox |  42 / 50 | 50       | 3 hours ago  |    |

Ilustración 29 Backend Mailtrap

Por defecto se crea el repositorio de mensajes llamado; “Demo inbox”, daremos clic en el para ingresar a sus ajustes.


[Shared Inboxes](#)
[Billing](#)



[Home](#) / [Demo inbox](#)

Se te ha asignado un evaluador

To: <fred@joya.joya>

4 hours ago

Se te ha asignado un proyecto

To: <paiza@mail.com>

4 hours ago

Su proyecto fue aceptado ¡Felicitaciones!

To: <fred@joya.joya>

4 hours ago

Se ha enviado un formulario de un caso prueba

To: <paiza@mail.com>

7 days ago

Se ha creado un caso prueba para su proyecto

To: <fred@joya.joya>

7 days ago

Proyecto listo para evaluación

To: <paiza@mail.com>

7 days ago

Se ha activado su proyecto

To: <fred@joya.joya>

8 days ago

Se ha enviado un formulario de un caso prueba

To: <paiza@mail.com>

8 days ago

Se ha creado un caso prueba para su proyecto

To: <fred@joya.joya>

8 days ago

Se ha enviado un formulario de un caso prueba

To: <paiza@mail.com>

8 days ago

Se ha creado un caso prueba para su proyecto

To: <fred@joya.joya>

8 days ago

Se ha enviado un formulario de un caso prueba

To: <paiza@mail.com>

8 days ago

Se ha creado un caso prueba para su proyecto

To: <fred@joya.joya>

8 days ago

Se ha enviado un formulario de un caso prueba

To: <fred@joya.joya>

8 days ago

[SMTP Settings](#)
[Email Address](#)
[Auto Forward](#)
[Manual Forward](#)
[Users](#)

## Credentials

[Reset SMTP/POP3](#)

### SMTP

Host: smtp.mailtrap.io

Port: 25 or 465 or 2525

Username: 26d4e5da1fb4e7

Password: f82f27b88f9aeb

Auth: PLAIN, LOGIN and CRAM-MD5

TLS: Optional

## Integrations

In config/environments/\*.rb specify ActionMailer defaults for your development or staging servers:

```

config.action_mailer.delivery_method = :smtp
config.action_mailer.smtp_settings = {
  :user_name => '26d4e5da1fb4e7',
  :password => 'f82f27b88f9aeb',
  :address => 'smtp.mailtrap.io',
  :domain => 'smtp.mailtrap.io',
  :port => '2525',
  :authentication => :cram_md5
}

```

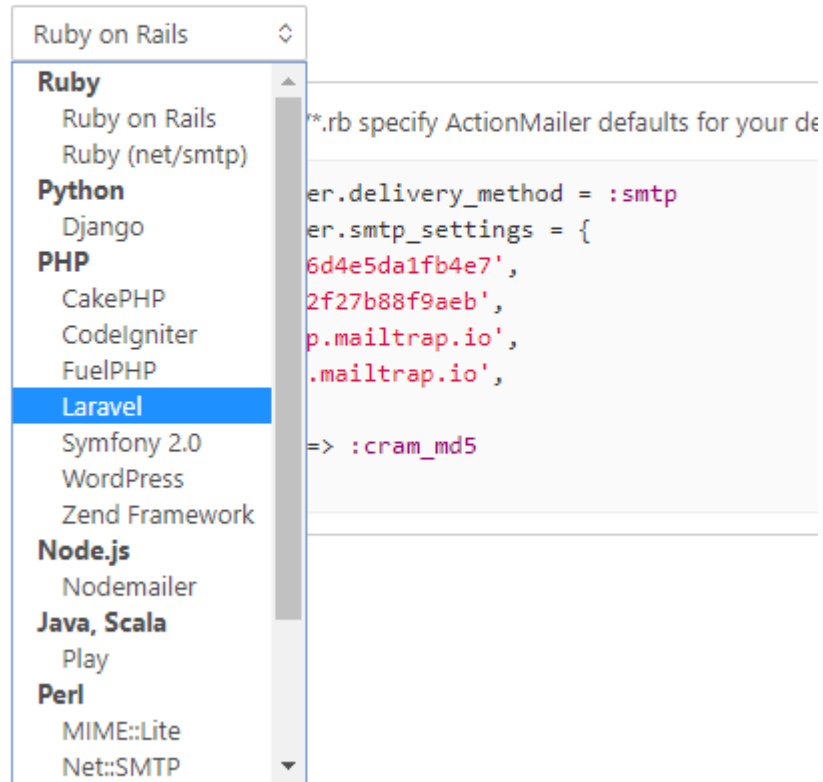
Ilustración 30 Administración de mensajes de Mailtrap

Una vez dentro, iremos a la sección “Integrations” y en el input selec llamado; “Ruby on Rails” buscaremos y seleccionaremos la opción de “Laravel”



Auth: PLAIN, LOGIN and CRAM-MD5  
TLS: Optional

## Integrations



*Ilustración 31 Apartado Integrations Mailtrap*

En la parte de abajo se actualizarán los datos necesarios para configurar mailtrap en Laravel

```
return array(
    "driver" => "smtp",
    "host" => "smtp.mailtrap.io",
    "port" => 2525,
    "from" => array(
        "address" => "from@example.com",
        "name" => "Example"
    ),
    "username" => "26d4e5da1fb4e7",
    "password" => "f82f27b88f9aeb",
    "sendmail" => "/usr/sbin/sendmail -bs",
    "pretend" => false
);
```

*Ilustración 32 Atributos de Mailtrap en Laravel*

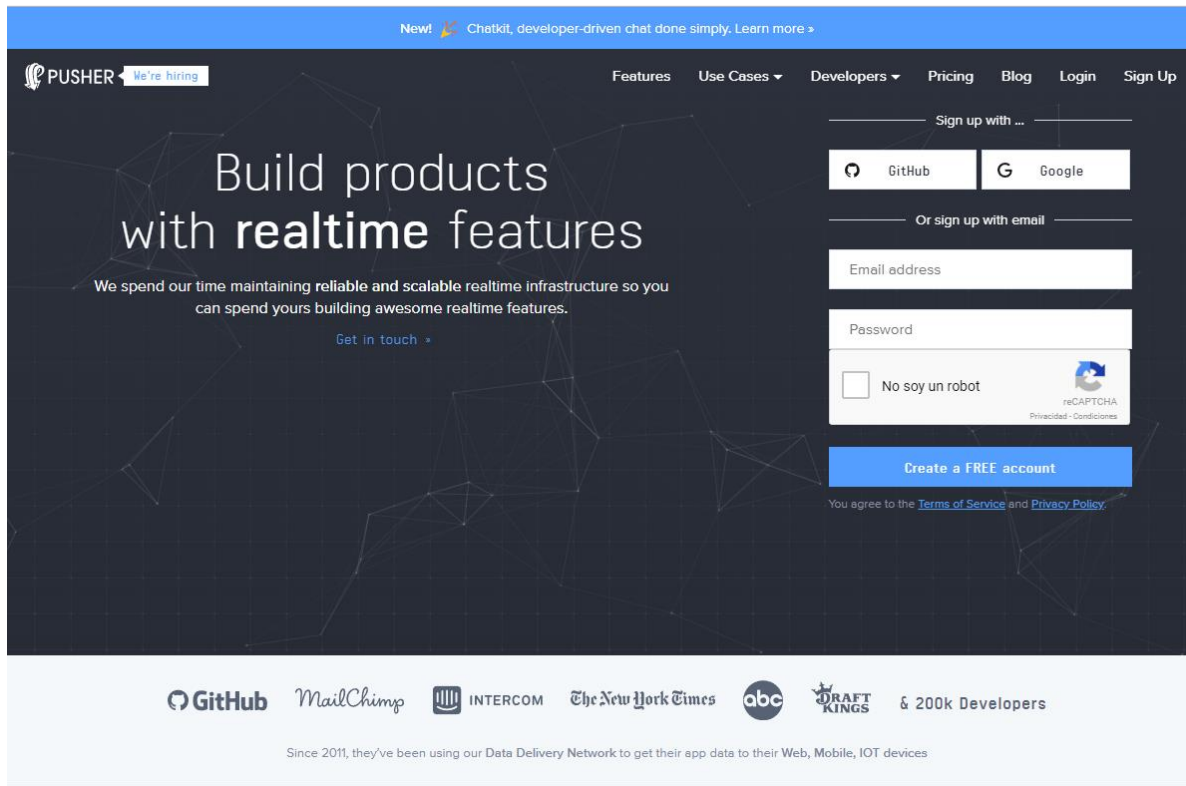
Estos datos debes ser usados en el archivo `example.env` en la sección de MAIL. Al realizar las correspondientes modificaciones, debe quedar de la siguiente forma;

```
MAIL_DRIVER=smtp
MAIL_HOST=smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=26d4e5da1fb4e7
MAIL_PASSWORD=f82f27b88f9aeb
MAIL_ENCRYPTION=null
```

*Ilustración 33 Apartado de Mail en archivo `example.env`*

### Configuración de Pusher para notificaciones en tiempo real

Ahora, procederemos a editar la sección de PUSHER, que es la necesaria para que las notificaciones lleguen en tiempo real, para ello iremos a la página <https://pusher.com/>



*Ilustración 34 Página principal de Pusher*

Para usar este servicio es necesario crear una cuenta, esto se puede hacer por medio de un usuario en Github, Google o digitando un correo y aportando una contraseña. Una vez creada la cuenta e iniciada la sesión, la primera pestaña en aparecer será la siguiente;

## Welcome to Pusher!

Create an app to generate your unique credentials. You can create as many apps as you need.

1 Name your app ?

dry-gift-365








2 Select a cluster ?















us2 (US East (Ohio))

☐ Create apps for multiple environments?

3 What's your front-end tech?

4 What's your back-end tech?





Skip

Create my app

Ilustración 35 Welcome to Pusher

En donde se debe modificar el nombre de la “App” a preferencia del usuario y es necesario seleccionar el icono de Laravel como se ve en la siguiente imagen.

Today's messages sent today

### Welcome to Pusher!

Create an app to generate your unique credentials. You can create as many apps as you need.

**1** Name your app ⓘ

Calisoft








**2** Select a cluster ⓘ

us2 (US East (Ohio)) ▼
























☐ Create apps for multiple environments?

Customize your getting started guide (optional).

**3** What's your front-end tech?



**4** What's your back-end tech?



Skip

Create my app

*Ilustración 36 Modificación de Pusher*

Para continuar se debe oprimir el botón “Create my app” y la página redirigirá a un Dashboard en donde contiene todo el contenido y la administración de nuestra herramienta de tiempo real. Para entrelazar Pusher con Laravel, es necesario que se dirija a la pestaña “App Keys”

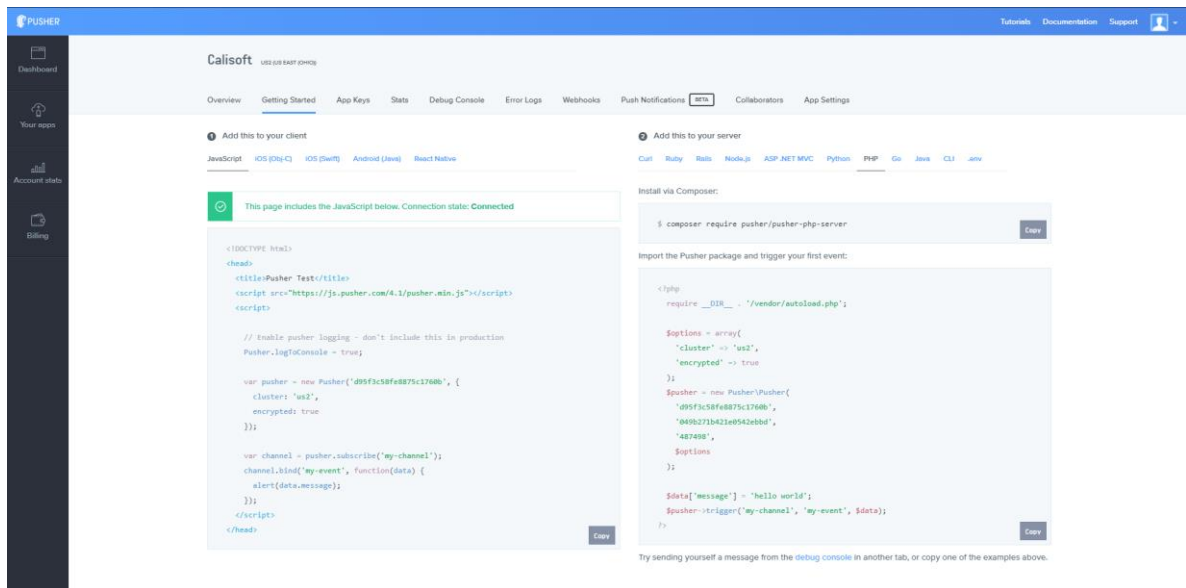


Ilustración 37 Backend Pusher

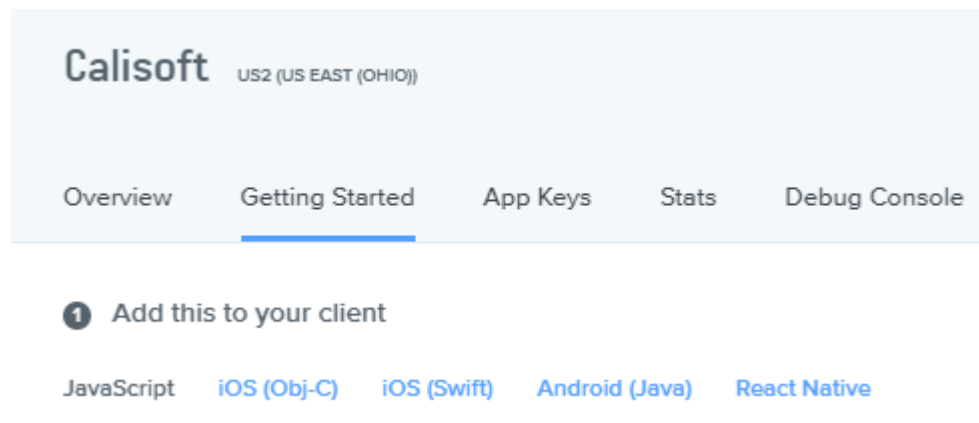


Ilustración 38 Pestañas Pusher

Seleccionamos la pestaña “App Keys” y una vez dentro, el sistema nos arrojará las diferentes llaves necesarias para que Calisoft conecte satisfactoriamente con Pusher

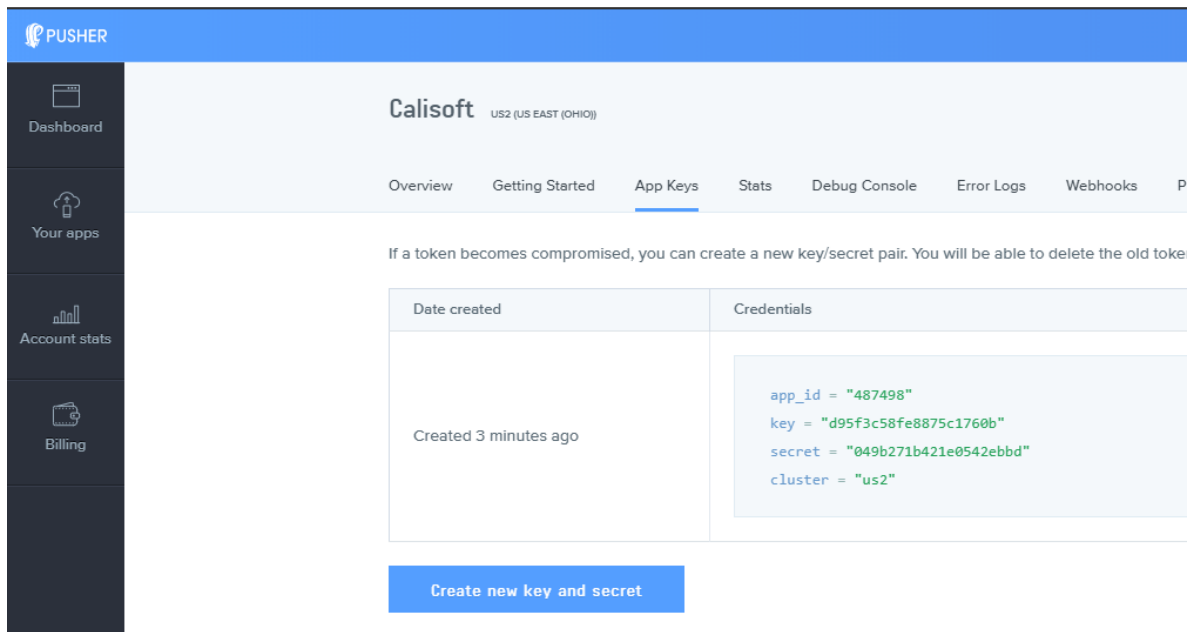


Ilustración 39 App Keys Pusher

Teniendo en cuenta las llaves, se procede a dejar la sección de Pusher del archivo `example.env`, de la misma forma en la que se halla en la página.

```
PUSHER_APP_ID=376363
PUSHER_APP_KEY=13fba520219a11216ae1
PUSHER_APP_SECRET=e6c109277502493e15e1
PUSHER_APP_CLUSTER=mt1
```

Ilustración 40 Configuración Pusher en archivo `example.env`

### Cambiar nombre archivo `example.env`

Con el anterior proceso finalizamos la edición del archivo `example.env`, ahora, procedemos a modificar el nombre de dicho archivo, se recomienda realizar este proceso desde un editor de texto. Para este ejemplo se usó Visual Studio Code, dando clic derecho sobre el archivo y yendo a la opción de modificar nombre.

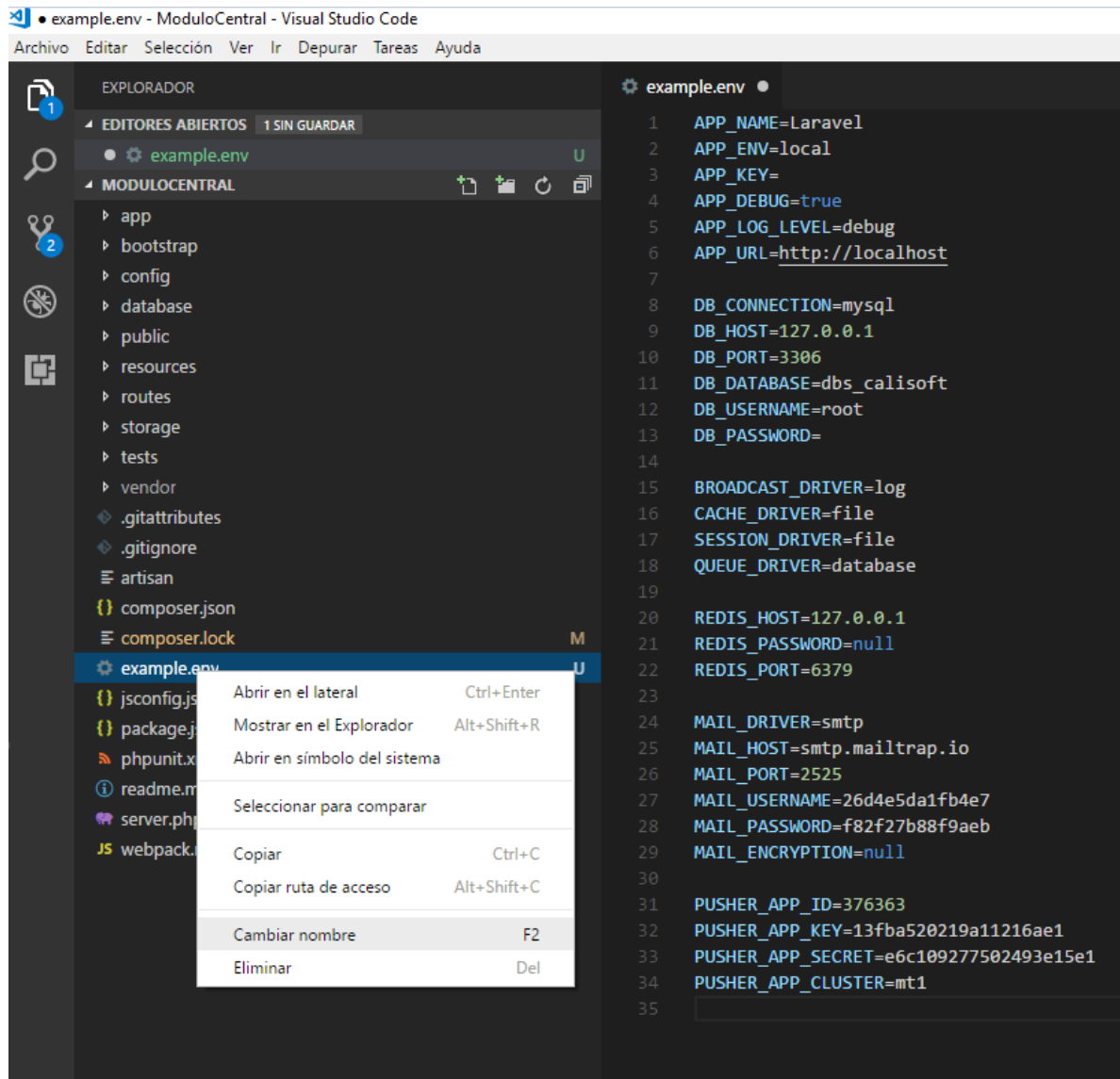


Ilustración 41 Cambiar nombre del archivo example.env

El archivo debe quedar como “.env”, sólo con la extensión como se aprecia en la siguiente ilustración;

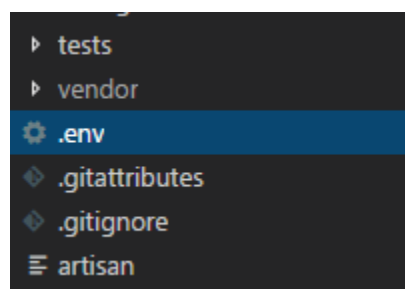


Ilustración 42 Archivo. env



## Llave de acceso

Al finalizar este paso, procedemos a ajustar la llave de acceso del aplicativo, y para ello, usamos en el símbolo del sistema que anteriormente teníamos abierto, o simplemente abrimos uno nuevo como se explicó anteriormente y digitamos el código “php artisan key:generate”, como se ve en la siguiente ilustración;

```
C:\>cd xampp  
C:\xampp>cd htdocs  
C:\xampp\htdocs>cd modolocentral  
C:\xampp\htdocs\ModuloCentral>php artisan key:generate
```

*Ilustración 43 Php artisan key:generate*

Presionamos la tecla enter y automáticamente asignará una Key para nuestro aplicativo.

## Creación ruta a carpeta Storage

A Continuación procedemos a digitar el comando “php artisan storage:link” con el objetivo de crear una carpeta storage de acceso directo dentro del contenido public, esto se hace para generar seguridad al momento de realizar una lectura a los archivos almacenados en la nube. Presionamos la tecla enter y si el proceso es satisfactorio deberá aparecer de la siguiente forma como se aprecia en la siguiente ilustración;

```
C:\xampp\htdocs\ModuloCentral>php artisan storage:link  
The [public/storage] directory has been linked.
```

*Ilustración 44 Proceso exitoso del comando Php artisan storage:link*

Si al ejecutar el comando “php artisan storage:link” generar un error, se aconseja abrir el símbolo del sistema como administrador y volver a intentar.

## Instalación de Vue de Node.js en la carpeta del proyecto

Al finalizar el paso anterior, se procede a instalar Vue de Node.js en el aplicativo, lo cual se realiza digitando el siguiente comando en el símbolo del sistema; “npm install --only=dev”, este proceso tarda entre alrededor de 5 minutos. Una vez finalizado, debe mostrar una ventana como la siguiente;

```

C:\xampp\htdocs\ModuloCentral>npm install --only=dev
npm WARN deprecated gulp-util@3.0.8: gulp-util is deprecated - replace it, following the guidelines at https://medium.com/gulpjs/gulp-util-ca3b1f9f9ac5

> node-sass@4.7.2 install C:\xampp\htdocs\ModuloCentral\node_modules\node-sass
> node scripts/install.js

Cached binary found at C:\Users\Odin\AppData\Roaming\npm-cache\node-sass\4.7.2\win32-x64-57_binding.node

> uglifyjs-webpack-plugin@0.4.6 postinstall C:\xampp\htdocs\ModuloCentral\node_modules\webpack\node_modules\uglifyjs-webpack-plugin
> node lib/post_install.js

> node-sass@4.7.2 postinstall C:\xampp\htdocs\ModuloCentral\node_modules\node-sass
> node scripts/build.js

Binary found at C:\xampp\htdocs\ModuloCentral\node_modules\node-sass\vendor\win32-x64-57_binding.node
Testing binary
Binary is fine

> gifsicle@3.0.4 postinstall C:\xampp\htdocs\ModuloCentral\node_modules\gifsicle
> node lib/install.js

  ✓ gifsicle pre-build test passed successfully

> mozjpeg@5.0.0 postinstall C:\xampp\htdocs\ModuloCentral\node_modules\mozjpeg
> node lib/install.js

  ✓ mozjpeg pre-build test passed successfully

> optipng-bin@3.1.4 postinstall C:\xampp\htdocs\ModuloCentral\node_modules\optipng-bin
> node lib/install.js

  ✓ optipng pre-build test passed successfully

> pngquant-bin@3.1.1 postinstall C:\xampp\htdocs\ModuloCentral\node_modules\pngquant-bin
> node lib/install.js

  ✓ pngquant pre-build test passed successfully

> undefined postinstall C:\xampp\htdocs\ModuloCentral
> npm run prod

> @ prod C:\xampp\htdocs\ModuloCentral
> npm run production

> @ production C:\xampp\htdocs\ModuloCentral
> cross-env NODE_ENV=production node_modules/webpack/bin/webpack.js --progress --hide-modules --config=node_modules/laravel-mix/setup/webpack.config.js

91% additional asset processing

```

*Ilustración 45 Proceso de instalación de Vue*

```
C:\WINDOWS\system32\cmd.exe
DONE Compiled successfully in 147842ms

  Asset      Size  Chunks             Chunk Names
  \js\evaluator-proyectos.js 261 kB    14 [emitted] [big] \js\evaluator-proyectos
    \js\evaluator-script.js 1.17 MB    0 [emitted] [big] \js\evaluator-script
    \js\archivo-sql.js 1.13 MB    2 [emitted] [big] \js\archivo-sql
  \js\scripts-codificacion.js 1.13 MB    3 [emitted] [big] \js\scripts-codificacion
    \js\proyectos.js 269 kB    4 [emitted] [big] \js\proyectos
    \js\admin-proyectos.js 267 kB    5 [emitted] [big] \js\admin-proyectos
    \js\plataforma-student.js 330 kB    6 [emitted] [big] \js\plataforma-student
    \js\escenario.js 411 kB    7 [emitted] [big] \js\escenario
    \js\componentes.js 346 kB    8 [emitted] [big] \js\componentes
    \js\plataforma.js 257 kB    9 [emitted] [big] \js\plataforma
    \js\usuarios.js 407 kB   10 [emitted] [big] \js\usuarios
    \js\modelacion.js 258 kB   11 [emitted] [big] \js\modelacion
    \js\evaluator-docs.js 265 kB   12 [emitted] [big] \js\evaluator-docs
    \js\documentos.js 260 kB   13 [emitted] [big] \js\documentos
    \js\base-datos.js 1.22 MB    1 [emitted] [big] \js\base-datos
    \js\categorias.js 256 kB   15 [emitted] [big] \js\categorias
  \js\evaluacion-modelado.js 263 kB   16 [emitted] [big] \js\evaluacion-modelado
    \js\codificacion.js 255 kB   17 [emitted] [big] \js\codificacion
    \js\tipo-nomenclatura.js 192 kB   18 [emitted] [big] \js\tipo-nomenclatura
    \js\nomenclaturas-show.js 253 kB   19 [emitted] [big] \js\nomenclaturas-show
    \js\items-codificacion.js 192 kB   20 [emitted] [big] \js\items-codificacion
    \js\categorias-show.js 254 kB   21 [emitted] [big] \js\categorias-show
    \js\tipo-documento.js 192 kB   22 [emitted] [big] \js\tipo-documento
    \js\semilleros.js 193 kB   23 [emitted] [big] \js\semilleros
    \js\items-show.js 190 kB   24 [emitted] [big] \js\items-show
    \js\notificaciones.js 189 kB   25 [emitted] [big] \js\notificaciones
    \js\invitaciones.js 189 kB   26 [emitted] [big] \js\invitaciones
    \js\bootstrap.js 189 kB   27 [emitted] [big] \js\bootstrap

npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN uiv@0.11.11 requires a peer of bootstrap@^3.3.7 but none is installed. You must install
npm WARN ajv-keywords@3.1.0 requires a peer of ajv@^6.0.0 but none is installed. You must install
npm WARN @vue/test-utils@1.0.0-beta.12 requires a peer of vue-server-renderer@2.x but none is installed
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.1.3 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.1.3: wanted {
  "arch": "x64",
  "os": "darwin",
  "cpu": "x64"
}

added 1532 packages in 652.237s

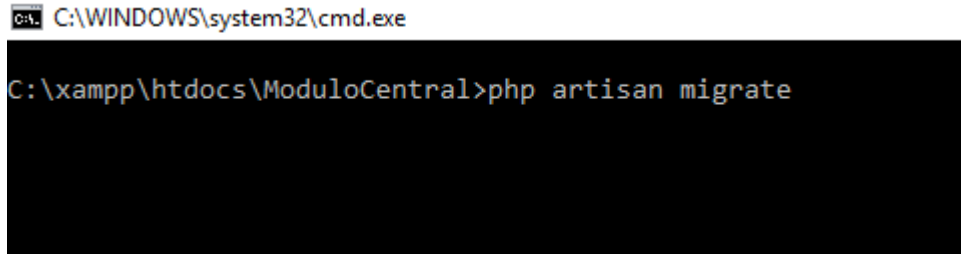
C:\xampp\htdocs\ModuloCentral>
```

Ilustración 46 Finalización exitosa de Vue en carpeta del proyecto

Esto significa que todos los componentes se instalaron correctamente. Se recomienda que después de la instalación, se ejecute el comando “npm update” para que actualice los componentes instalados.

## Migraciones

El paso siguiente se centra en generar los datos necesarios para agregar valores automáticamente a la base de datos, estos datos permitirán acceder a los diferentes módulos del aplicativo. El comando es “php artisan migrate”, se debe realizar sobre la carpeta del proyecto como se ha realizado con los comandos anteriores. (Recuerde que Apache y MySQL deben estar activados en el programa Xampp).

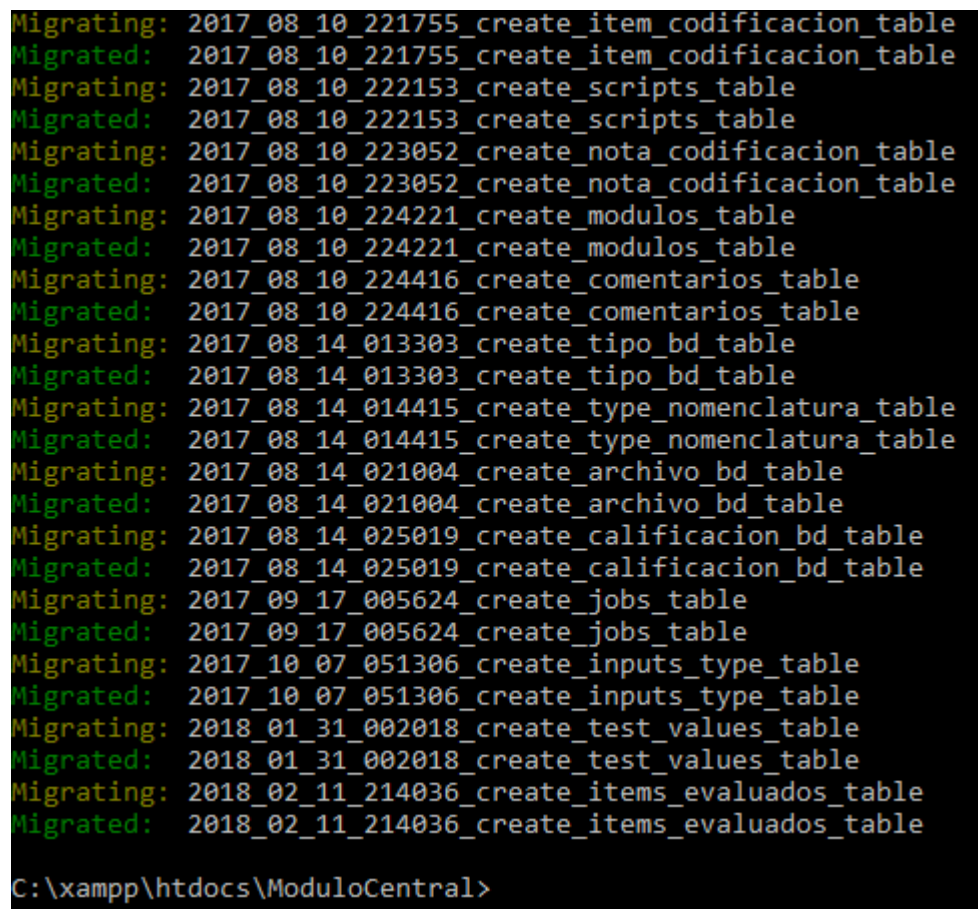


```
C:\WINDOWS\system32\cmd.exe

C:\xampp\htdocs\ModuloCentral>php artisan migrate
```

*Ilustración 47 Comando php artisan migrate*

Se pulsa la tecla enter y si el proceso es satisfactorio, deberá aparecer la siguiente ventana;



```
Migrating: 2017_08_10_221755_create_item_codificacion_table
Migrated:  2017_08_10_221755_create_item_codificacion_table
Migrating: 2017_08_10_222153_create_scripts_table
Migrated:  2017_08_10_222153_create_scripts_table
Migrating: 2017_08_10_223052_create_nota_codificacion_table
Migrated:  2017_08_10_223052_create_nota_codificacion_table
Migrating: 2017_08_10_224221_create_modulos_table
Migrated:  2017_08_10_224221_create_modulos_table
Migrating: 2017_08_10_224416_create_comentarios_table
Migrated:  2017_08_10_224416_create_comentarios_table
Migrating: 2017_08_14_013303_create_tipo_bd_table
Migrated:  2017_08_14_013303_create_tipo_bd_table
Migrating: 2017_08_14_014415_create_type_nomenclatura_table
Migrated:  2017_08_14_014415_create_type_nomenclatura_table
Migrating: 2017_08_14_021004_create_archivo_bd_table
Migrated:  2017_08_14_021004_create_archivo_bd_table
Migrating: 2017_08_14_025019_create_calificacion_bd_table
Migrated:  2017_08_14_025019_create_calificacion_bd_table
Migrating: 2017_09_17_005624_create_jobs_table
Migrated:  2017_09_17_005624_create_jobs_table
Migrating: 2017_10_07_051306_create_inputs_type_table
Migrated:  2017_10_07_051306_create_inputs_type_table
Migrating: 2018_01_31_002018_create_test_values_table
Migrated:  2018_01_31_002018_create_test_values_table
Migrating: 2018_02_11_214036_create_items_evaluados_table
Migrated:  2018_02_11_214036_create_items_evaluados_table

C:\xampp\htdocs\ModuloCentral>
```

*Ilustración 48 Migraciones exitosas*

Para confirmar la acción de este código, puede dirigirse a la base de datos y comprobar la creación de dichas tablas.

Si durante el proceso se genera un error, verifique que el nombre, contraseña y usuario de la base de datos corresponda a los datos que se encuentran en el archivo “.env”.

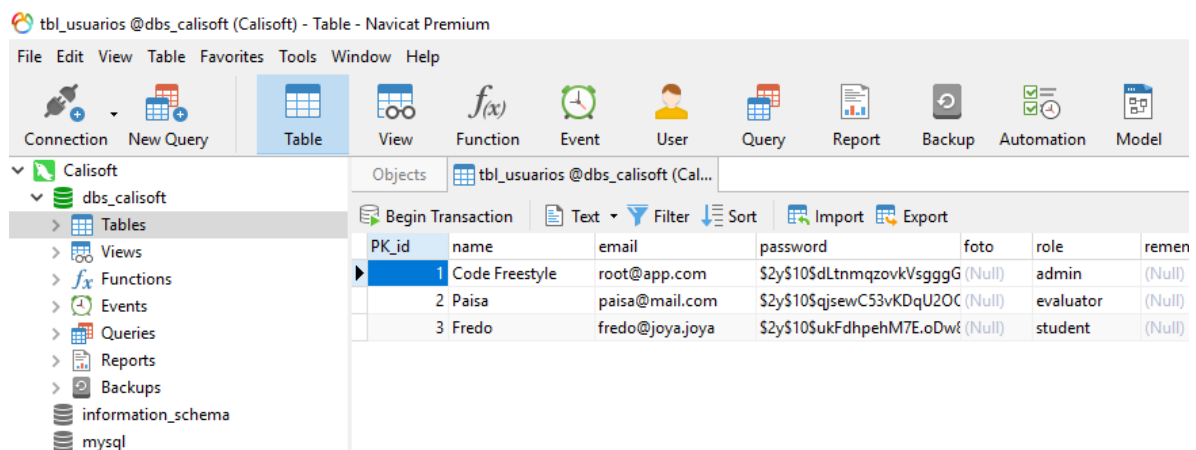
## Seeders

El paso siguiente al finalizar este proceso, es generar los seeders. Los seeders son un método de Laravel para generar registros automáticos en las tablas de la base de datos, esto es principal para definir los primeros usuarios del aplicativo o los registros principales con los que se quiere iniciar el software. El código es “php artisan db:seed”. Al dar enter se ejecutará el comando y al finalizar deberá aparecer la siguiente información;

```
C:\xampp\htdocs\ModuloCentral>php artisan db:seed
Seeding: UsersTableSeeder
Seeding: CategoriesTableSeeder
Seeding: TiposDocumentoTableSeeder
Seeding: SemillerosTableSeeder
Seeding: GrupoInvestigacionTableSeeder
Seeding: ItemsCodificacionTableSeeder
Seeding: ModulosTableSeeder
Seeding: TipoNomenclaturaTableSeeder
Seeding: TipoBdTableSeeder
Seeding: ComponenteTableSeeder
Seeding: TipoInputsTestingSeeder
Seeding: TestValueSeeder
C:\xampp\htdocs\ModuloCentral>
```

Ilustración 49 Comando php artisan db:seed exitoso

Para confirmar si los registros se insertaron correctamente, diríjase a la base de datos y compruebe si las tablas, por ejemplo, TBL\_Usuarios contienen registros.



The screenshot shows the Navicat Premium interface. On the left, the 'Objects' tree is expanded to show the 'tbl\_usuarios' table. The main window displays the table's data. The table has columns: PK\_id, name, email, password, foto, role, and remen. There are three records in the table.

| PK_id | name           | email           | password                   | foto   | role      | remen  |
|-------|----------------|-----------------|----------------------------|--------|-----------|--------|
| 1     | Code Freestyle | root@app.com    | \$2y\$10\$dLtnmqzovkVsgggG | (Null) | admin     | (Null) |
| 2     | Paiza          | paiza@mail.com  | \$2y\$10\$qjsewC53vKDqU2OC | (Null) | evaluator | (Null) |
| 3     | Fredo          | fredo@joya.joya | \$2y\$10\$ukFdhpehM7E.oDw  | (Null) | student   | (Null) |

Ilustración 50 Registros de la tabla TBL\_Usuarios

Como puede observar, el password se almacena con un Has, lo que no permite su lectura a simple vista, para saber las contraseñas de los diferentes usuarios, usted deberá dirigirse mediante su editor de texto a la siguiente carpeta siguiendo la ruta;

C:\xampp\htdocs\ModuloCentral\database\seeds\UsersTableSeeder.php y dentro del código usted podrá observar las correspondientes contraseñas de los usuarios creados mediante los métodos Seeders.

```
DB::table('TBL_Usuarios')->insert([
    [
        'name' => 'Code Freestyle',
        'email' => 'root@app.com',
        'role' => 'admin',
        'password' => bcrypt('12345')
    ],
    [
        'name' => 'Paísa',
        'email' => 'paísa@mail.com',
        'role' => 'evaluator',
        'password' => bcrypt('12345')
    ],
    [
        'name' => 'Fredo',
        'email' => 'fredo@joya.joya',
        'role' => 'student',
        'password' => bcrypt('12345')
    ]
])
```

*Ilustración 51 Datos de los seeders*

### Correr el aplicativo

Hasta este punto usted ya logró configurar correctamente el aplicativo Calisoft, ahora usted deberá correr los siguientes códigos en diferentes símbolos del sistema, los códigos son;

- “php artisan serve”: Es necesario para crear acceder al servidor local mediante la dirección Ip que este comando genera.
- “npm run watch”: Este comando permite que los componentes .js se mantengan compilando.
- “php artisan queue:work --tries=3”: Es el encargado de realizar las notificaciones en tiempo real.

Recuerde que cada código debe ser ejecutado sobre la carpeta del aplicativo y en diferentes símbolos del sistema.

## Acceso al login

Para acceder al Login, debe copiar la dirección Ip que generó el código “php artisan serve”  
Y pegarlo en el navegador de Google Chrome

```
C:\WINDOWS\system32\cmd.exe - php artisan serve

C:\xampp\htdocs\ModuloCentral>php artisan serve
Laravel development server started: <http://127.0.0.1:8000>
```

Ilustración 52 Ip del servidor local

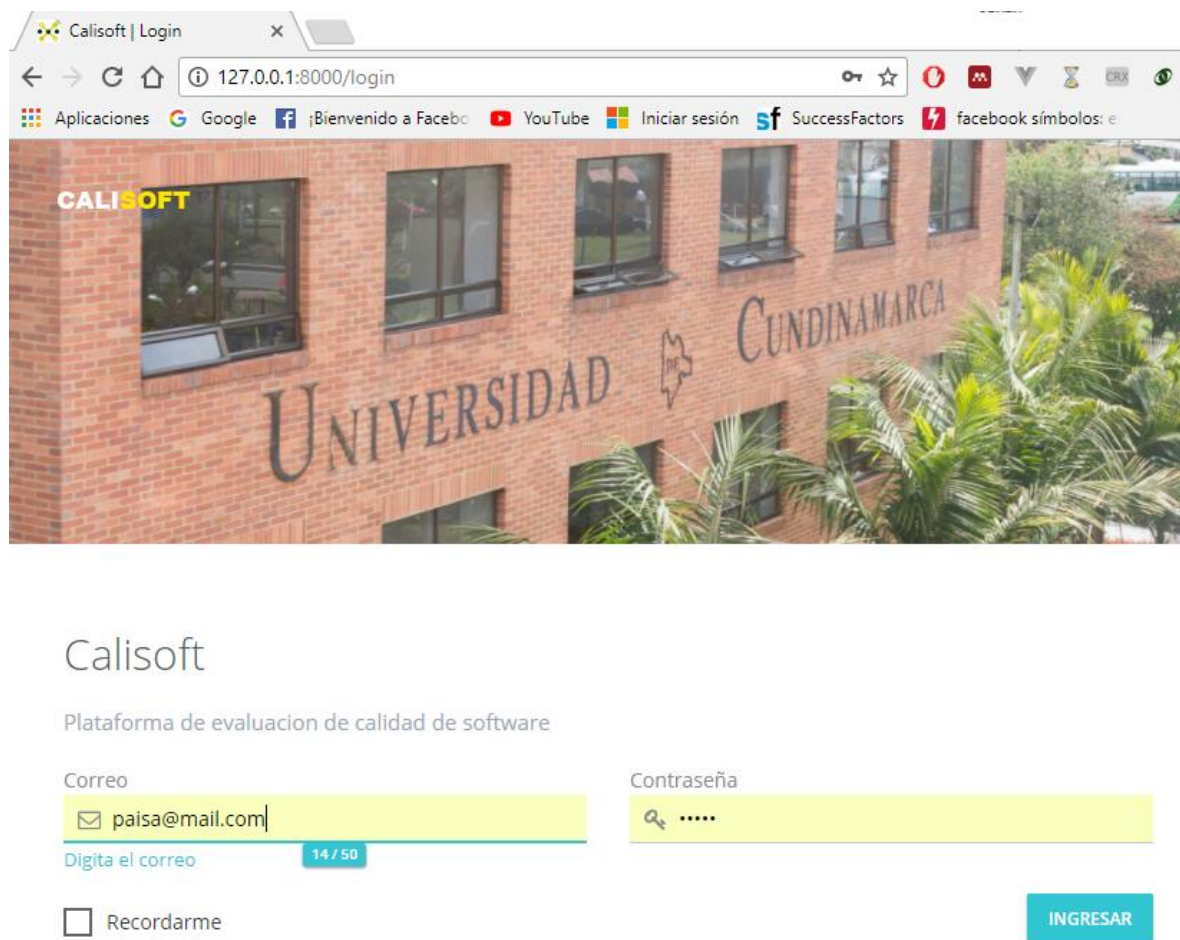


Ilustración 53 Página principal de Calisoft

Con este último paso procedemos a finalizar el proceso de instalación de Calisoft, para obtener información de los diferentes módulos del aplicativo, proceda a leer el Manual De Usuario de Calisoft.

## REFERENCIAS

Apache Friends. (2015). About the XAMPP project. Retrieved March 4, 2018, from <https://www.apachefriends.org/de/about.html>

Composer. (n.d.). Introduction - Composer. Retrieved March 4, 2018, from <https://getcomposer.org/doc/00-intro.md>

mailtrap. (n.d.). mailtrap.io. Retrieved March 8, 2018, from <https://mailtrap.io/>

Node.js. (n.d.). Node.js. Retrieved March 4, 2018, from <https://nodejs.org/es/>