

1. Manual Técnico

Este manual le permitirá aprender a configurar el Sistema de Gestión de Incidencias Técnicas (SIGIT) para ponerlo en producción.

Contenido

1. Manual Técnico	1
1.1. Módulo de Seguridad	2
1.1.1. Permisos	2
1.2. Configuración de la aplicación.....	2
1.2.1. Enrutamiento y dirección http de la aplicación	2
1.2.2. Configurar App/config.php	2
1.3. Configurar base de datos	2
1.4. Configurar Correo electrónico para el envío de incidencias	3
1.5. Configurar logos de aplicación	3
1.6. Manejador de dependencias para PHP	4
1.6.1. Instalación de manejador de dependencias PHP	4
1.6.2. Configurar el archivo composer.json	4
1.7. Clonar proyecto BitBucket.....	5
1.8. Carpetas del proyecto	5
1.9. Cotejamiento Base de Datos	6

1.1. Módulo de Seguridad

1.1.1. Permisos

Los permisos se guardan en la tabla usuPermisos los cuales están listados por módulos y permisos como (AGR = Agregar, (ACT = Actualización),(EST = Estado) o (CON =Consultar) además del tipo del usuario al que pertenecen

1. Administrador

2. Supervisor

3. Técnico

1.2. Configuración de la aplicación

1.2.1. Enrutamiento y dirección http de la aplicación

SI se desea cambiar de dominio o el nombre de la carpeta contenedora del proyecto se debe modificar dos archivos:

1.2.2. Configurar App/config.php

Cambiar la constante PHP

```
define('URL', 'dominio/SIGIT/');
```

1.2.3. Configurar URL para comprobar conexión a internet App/config.php

- Esto permite al usuario de intranet ingresar a la plataforma si este no posee internet mediante su número de cedula y correo electrónico.

```
define('URL_comprobar_conexion', 'https://www.google.com.co/');
```

1.2.4. Configurar public/js/config.js

Cambiar la constante JavaScript para el dominio

```
const URL = 'http://'+document.domain+'/SIGIT/';
```

Nota: Si desea quitar la carpeta de la url solo debe dejar el dominio terminado en “/”

1.3. Configurar base de datos

1.3.1. Configurar App/config.php

- Host o nombre del servidor

```
define('DB_HOST','nombre_servidor');
```

- Nombre de base de datos
`define('DB_NAME','nombre_basedatos');`
- Usuario de la base de datos SQL server
`define('DB_USER','usuario');`
- Contraseña de usuario de base de datos
`define('DB_PASS','contraseña');`

1.4. Configurar Correo electrónico para el envío de incidencias

1.4.1. Configurar App/config.php

- Host o nombre del servidor
`define('MAIL_HOST','mail.carnation.arvixe.com');`
- Correo electrónico para el envío de correos
`define('MAIL_USER','info@sigit.fusagasugadigital.gov.co');`
- Nombre de usuario para el envío de correos
`define('MAIL_USER_NAME','Oficina TIC');`
- Contraseña para el envío de correos
`define('MAIL_PASS','contraseña');`
- Correo de duplicación
`define('JEFE_TIC','jefetic@fusagasuga-cundinamarca.gov.co');`

1.5. Configurar logos de aplicación

- Logo Pestaña aplicación

```
define('logo_aplicacion', URL.'public/img/logo-sigit.png');
```

- Logo de la Alcaldía Municipal

```
define('logo_administracion', URL.'public/img/logo.png');
```

- Escudo de Fusagasugá

```
define('escudo_fusagasuga', URL.'public/img/escudo-fusagasuga.png');
```

- Escudo de Colombia

```
define('escudo_colombia', URL.'public/img/escudo-colombia.png');
```

1.6. Manejador de dependencias para PHP

1.6.1. Instalación de manejador de dependencias PHP

- Descargar e instalar <https://getcomposer.org/>
- Requiere: PHP 5 o 7(Proyecto en 5.6)
- Ejecuta el comando: **composer update o composer install** en la raíz del proyecto por CMD ,esto permitirá actualizar las librerías o descargarlas si no existen la carpeta vendor .

1.6.2. Configurar el archivo composer.json

Este archivo se puede modificar si se desea cambiar de versión alguna de las siguientes librerías.

```
{  
"require": {  
    "twig/twig": "~1.0",  
    "kanellov/slim-twig-flash": "^0.1.0",  
    "vlucas/valitron": "^1.3",  
    "spipu/html2pdf": "^4.6",  
    "phpoffice/phpexcel": "^1.8",  
    "swiftmailer/swiftmailer": "^5.4",  
    "phpmailer/phpmailer": "^5.2"
```

```
}  
  
}
```

1.7. Clonar proyecto BitBucket

Git Clone <https://nricaurte@bitbucket.org/sigitfusa/sigit.git>

Nota: Repositorio privado la oficina tic tiene acceso con su cuenta

1.8. Carpetas del proyecto

App = Están las configuraciones de la aplicación

Archivos = son las imágenes y documentos cargados en la aplicación,

Controllers = Están todas las clases encargadas de la lógica de negocio

Models = Clases encargadas de enviar los datos a la base de datos

Public = imágenes, videos y librerías css y js

Test = casos de prueba JunitTest

Vendor = Librería o dependencias instaladas con Composer

View = vistas htm5 y twig

- Si se desea crear otra funcionalidad se debe crear la función en la clase controlador y para llamarla en el navegador

<http://server02/SIGIT/nombreControlador/nombreFuncion/parametros>

Ejemplo de Funcion:

```
public function nombreFuncion()  
{  
    //Permisos para el usuario  
    if(!$this->getPermisosProveedor('categoria','AGR')){$this->redirectIndex();}  
  
    $this->_view->renderView('nombreCarpetaView/nombreDeVista.html.twig',  
        array(  
  
            'Titulo' => 'Titulo de la funcionalida()  
        ));
```

```
}
```

- Si la función necesita acceder a un modelo para guardar en la base de datos debe inicializarlo en el constructor de- controlador

```
$this->_nombreObjetoManipuladorDeModelo=$this->  
>loadModel('nombreClaseModelo');
```

1.9. Cotejamiento Base de Datos

Si se restaura la base de datos es necesario cambiar el cotejamiento puesto que el software hace uso de este para trabajar las fechas

Cambio de cotejamiento Sql Server 2014

```
COLLATE SQL_Latin1_General_CP1_CI_AS;
```

Realizado por:

Néstor Fernando Ricarte Aguirre

Ingeniería de Sistemas

Pasantía: Sistema de Gestión de Incidencias Técnicas

Código: 161211143