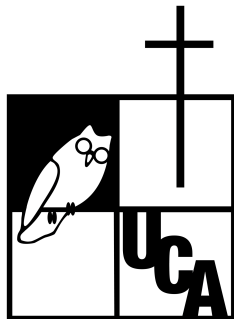


# UNIVERSIDAD CENTROAMERICANA JOSÉ SIMEON CAÑAS



## Manual de Usuario

Carlos Fernando Vásquez Morataya

00060414

Regina Beatriz Viscarra Cortes

00060514

## **Edición y Administración para Sitio de Centro Escolar Nuevo Lourdes**

11 de diciembre de 2018

# Índice

<b>1. Sitio Oficial Nuevo Lourdes</b>	<b>2</b>
1.1. Obteniendo Archivos del Sitio . . . . .	2
1.2. Añadiendo Nuevo Contenido . . . . .	3
1.2.1. Cambiando Contenido de Página frontal . . . . .	3
1.2.2. Añadiendo Noticias . . . . .	4
1.2.3. Markdown . . . . .	5
1.3. Probando cambios antes de subirlos . . . . .	6
1.4. Subiendo cambios . . . . .	6

# 1. Sitio Oficial Nuevo Lourdes

Esta es una guía básica para la edición y publicación de contenido en este sitio web, los resultados serán visibles en el siguiente link <https://nuevolourdes.github.io/>.

Todos los archivos presentes en esta entrega deberán ser editados con un editor de texto convencional, algunos ejemplos de estos son:

- notepad
- notepad++
- sublime text
- visual studio code

Todos estos editores se encuentran disponibles completamente gratis para todos los sistemas operativos, Windows, Mac OS y Linux.

## 1.1. Obteniendo Archivos del Sitio

Para obtener los archivos del sitio para modificarlo, es necesario tener la última versión de **git** instalada en su computadora, a continuación se presentan guías para hacerlo en distintos sistemas operativos, el sistema recomendado es Ubuntu.

- Windows: [Descargar aquí](#)
- Ubuntu: [Seguir guía](#)

Luego de instalar **git** deberíamos de tener una consola especial llamada **git Bash** en el caso de Windows, en Linux se puede utilizar la consola usual para los siguientes pasos, deberá ejecutar los siguientes comandos (cada \$ es un comando distinto, no incluirlo al digitar los comandos):

```
$ git clone https://github.com/nuevolourdes/sitio
$ cd sitio
$ git submodule add -b master \
https://github.com/nuevolourdes/nuevolourdes.github.io.git public
```

Luego de ejecutar estos comandos se deberá tener un folder cuyos contenidos serán parecidos a lo siguiente bloque, (se han dejado de fuera los folders que no editaremos o no importan para los propósitos de esta guía), dentro de estos se definen todo el contenido del sitio en texto plano en archivos ya sea de formato **.md** o formato **.yaml**:

```
.
./auto
./config.toml
./content
./data
./deploy.sh
./public
./static
./themes
```

## 1.2. Añadiendo Nuevo Contenido

Para añadir nuevo contenido o remover contenido viejo, deberemos tener instalado un programa llamado **hugo**, descarga el archivo respectivo para poder correrlo en tu sistema operativo:

- Windows: [Hugo para Windows](#)
- Ubuntu: [Hugo para Linux](#)

### 1.2.1. Cambiando Contenido de Página frontal

```
data/
  carousel/
    confiable.yaml
    valores.yaml
  features/
    ambiente.yaml
    educacion.yaml
    valores.yaml
```

Podemos cambiar el contenido de la pagina frontal en varios aspectos, dentro del folder **carousel** se cambian los contenidos ilustrados por la figura 3:



Figura 1: carousel

Dentro el folder **features** se cambian los aspectos de la página ilustrados por la figura 2, dentro de estos archivos se encontraran las definiciones de visión, misión y valores, los cuales podrán ser editados y guardados para actualizar el sitio:



Figura 2: features

## 1.2.2. Añadiendo Noticias

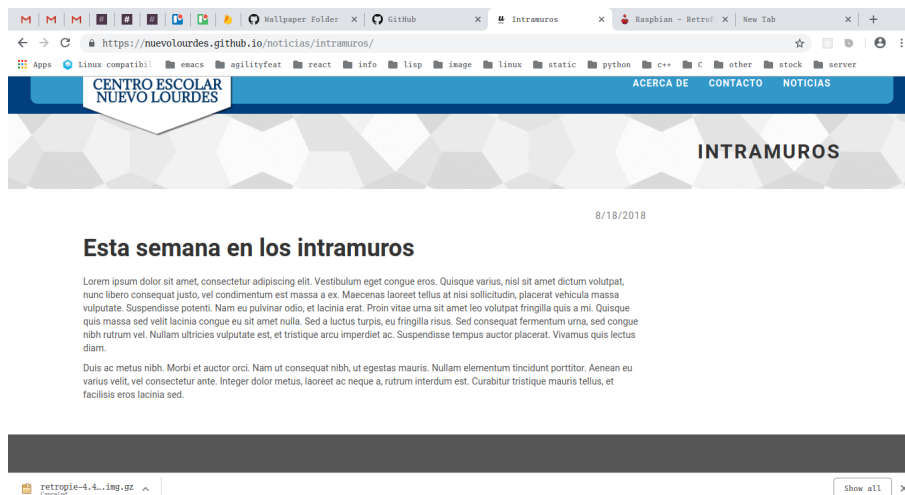


Figura 3: Ejemplo de una Noticia

Podemos hallar el contenido actual bajo esta estructura, en el folder `contents`:

```
content/
  acercade.md
  noticias/
    entreganotas.md
    intramuros.md
    nuevopost.md
```

Podemos crear un nuevo archivo con extensión `.md` para crear una nueva página con el contenido de noticias, podemos simplemente copiar uno existente, renombrar y editar los contenidos del nuevo archivo, los contenidos se verán a continuación.

```
---
title: "Vacaciones"
date: 2018-08-19T17:53:29-06:00
draft: false
type: "blog"
---
```

```
# Vacaciones semana santa
```

```
Sed dolor sapien, rhoncus nec urna vitae, rhoncus congue urna.  
Duis quis vehicula massa. Ut scelerisque consectetur justo.  
Nam in libero at dui lacinia scelerisque molestie id eros.  
Curabitur est felis, viverra in mauris eu, tristique efficitur  
neque. Vestibulum lorem ex, consectetur consequat justo ac, scelerisque  
dignissim sapien. In eu lobortis tellus.
```

```
![texto](./img/unaimagen.jpg)
```

El texto bajo **title**, sera el titulo de nuestra noticia, **date** la fecha, **draft** indica si este es un borrador, siempre deberá ser **false** para asegurarnos de que nuestras noticias se publiquen siempre, finalmente en la configuración tenemos **type**, el cual en el caso de las noticias deberá estar siempre en **blog**.

Podemos añadir imágenes a la carpeta **static/img/** dentro de la carpeta principal, luego podemos incluir las imágenes en las noticias de la forma que se utiliza arriba.

### 1.2.3. Markdown

este es el formato en la que la mayoría de contenido estara escrito, se explicara brevemente que significa cada elemento en este formato, a continuación un ejemplo:

```
# Este es un titulo  
## Esto es un subtitulo  
### Esto es un subsubtitulo
```

```
===
```

El bloque de arriba "===" indica una linea separadora en nuestro contenido, también podemos crear listas:

```
* este es un elemento  
* este es otro elemento  
* este es un tercer elemento
```

```
===
```

Para insertar imágenes podemos hacerlo fácilmente usando el siguiente formato:

```
![texto auxiliar](/url/o/ruta/al/archivo)
```

Si hacemos esto sin el signo de exclamación, el resultado final será un link a la imagen, es decir, esta no se mostrara.

```
===
```

esto es texto en **\*\*negrita\*\*** y esto es texto en *\*cursivo\**

```
<p>
```

```
como una función avanzada también podemos utilizar
HTML dentro de esta versión de markdown.
</p>
```

Para una lista más comprensiva de las funcionalidades del formato `markdown` se recomienda revisar este sitio: <https://markdown.es/sintaxis-markdown/>

### 1.3. Probando cambios antes de subirlos

Para probar los cambios, se puede utilizar `hugo`, de la siguiente forma:

```
$ hugo serve
```

Este comando ocasionara que los archivos sean interpretados y que el resultado final sea colocado en la carpeta `public`, también actuara como un servidor local, el cual podremos ver funcionando en la consola desde la cual ejecutamos el comando, a continuación, se puede dirigir a la dirección <http://localhost:1313> para ver el resultado, cualquier cambio que se haga mientras corre el servidor será automáticamente actualizado en la versión local, cuando se este satisfecho, es hora de subir cambios.

### 1.4. Subiendo cambios

Subir los cambios es muy sencillo, solo se requiere ejecutar el siguiente comando, luego de eso los cambios estarán arriba en cuestión de minutos:

```
$ bash ./deploy.sh
```

Luego se puede navegar a <https://nuevolourdes.github.io/>, para observar el sitio actualizado.