Manual de usuario para implementar los archivos estáticos del servidor nginx en python-django

1. Para ver el diseño de la plantilla original abra el navegador de su preferencia, en este caso usare mozilla firefox en ubuntu 12.04, e ingrese la siguiente dirección.

[http://template.tabasco.gob.mx](http://template.tabasco.gob.mx/)

 Ejemplo:

 

1. Ingrese el usuario y la contraseña en el formulario anterior puede ser cualquier carácter solo debe tener carácter el usuario y la contraseña y clic en Aceptar, deberá mostrar la siguiente pagina. 

En el paso dos muestra la pagina principal de la plantilla, si muestra un error como el que se muestra en la imagen siguiente solo recargue la página.



1. Cuando halla seleccionado el o los estilos aplicar en el sistema en cuestión pase al siguiente paso.
2. Ingrese al settings.py del proyecto, para ello busque en la carpeta principal, por lo general el nombre de la carpeta es el mismo nombre que el proyecto. Ejemplo:



1. Habra el archivo settings.py con el editor de su preferencia y busque la linea **STATIC\_URL =,** se encuentra entre la linea 70 y 80 por lo general deberá estar después de **STATIC\_ROOT. Ejemplo:**
2. Se deberá ingresar la ruta de nginx en este caso es [http://template.tabasco.gob.mx](http://template.tabasco.gob.mx/) para que apunte al servidor y no a los estáticos del sistema, deberá quedar tal y como se muestra en la imagen del paso 5.
3. si se desea tener acceso al servidor de nginx a través de un gestor de archivos FTP realice los siguientes paso de lo contrario ignórelo y continúe en el paso 16.

**Pasos para ingresar al servidor nginx utilizando el cliente ftp filezilla.**

1. 1. Descargar el cliente FTP FileZilla para ello accedemos a la siguiente direccion:

<https://filezilla-project.org/download.php?type=client>

**Nota:** La liga anterior es para el sistema operativo Linux.

1. Una vez instalado filezilla ingresamos la dirección IP del servidor el cual es 10.2.10.53 como se muestra a continuación.
2. Nota: el usuario y contraseña deberá ser solicitado al jefe del departamento. El puerto es el **22** como se muestra en la imagen anterior.
3. Si se realizaron los pasos anteriores correctamente, se tendrá acceso al servidor y se podra ver las carpetas similares a las de la imagen siguiente.



1. Borrar la palabra root y dejar la diagonal / y presionar enter y mostrara unas carpetas similares a las que se muestran en la imagen siguiente.



1. Entrar a la carpeta **usr** y seleccionar la carpeta **share** el cual contiene unas carpetas similares a las que se muestran a la siguiente imagen.
2. Entrar a la carpeta **nginx** y buscar la carpeta **DGTICADMIN** el cual contiene los archivos estáticos de la plantilla.
3. Hasta el paso 6 ya se tiene configurado el uso de nginx, para ponerlo a funcionar se debe de agregar en el archivo base la siguiente linea de código **{% load staticfiles %}** para que cargue los archivos estaticos.

**Nota: la linea anterior es una instrucción propia de django y el archivo base hace referencia al documento html donde se cargan todos los js y css.**

1. Para agregar una ruta de un archivo js se debe de hacer de la siguiente forma.

**<script src="{% static "ruta del archivo.js" %}"></script>**

Ejemplo:

Nota: si no se escribe la instrucción **{% static %}** no se podrá acceder a la ruta de nginx.

1. Para agregar un estilo **css** es similar al del paso 17. Ejemplo:



1. verificar que en el archivo base se encuentren todos los codigos de compatibilidad para los navegadores ya que de no agregarlos se podrian presentar problemas con los navegadores. Ejemplo:



1. Las rutas de los css se deben de colocar en el mismo orden en el que están en la plantilla original de lo contrario podrían no funcionar. Ejemplo.
2. Las rutas de los js se deben de implementar hasta el final dentro del body del archivo base a como se declara en la plantilla. Nota los archivos js que maneja la plantilla son los siguientes.

<!-- JS Libs -->

 <script src="{% static "assets/js/libs/jquery-1.8.3.min.js" %}"></script>

 <script src="{% static "assets/js/libs/jquery.placeholder.min.js" %}"></script>

 <script src="{% static "assets/js/libs/jquery.mousewheel.min.js" %}"></script>

 <!-- JS Bootstrap -->

 <script src="{% static "bootstrap/js/bootstrap.min.js" %}"></script>

 <!-- jQuery-UI JavaScript Files -->

 <script src="{% static "assets/jui/js/jquery-ui-1.9.2.min.js" %}"></script>

 <script src="{% static "assets/jui/jquery.ui.timepicker.min.js" %}"></script>

 <script src="{% static "assets/jui/jquery.ui.touch-punch.min.js" %}"></script>

 <link rel="stylesheet" type="text/css" href="{% static "assets/jui/css/jquery.ui.all.css" %}" media="screen">

 <!-- JS Plugins -->

 <!-- Validation Plugin -->

 <script src="{% static "plugins/validate/jquery.validate.min.js" %}"></script>

 <!-- Wizard Plugin -->

 <script src="{% static "assets/js/plugins/wizard/dandelion.wizard.min.js" %}"></script>

 <script src="{% static "assets/js/libs/jquery.form.min.js" %}"></script>

 <link rel="stylesheet" href="{% static "assets/js/plugins/wizard/dandelion.wizard.css" %}" media="screen">

 <script src="{% static "plugins/datatables/jquery.dataTables.js" %}"></script>

 <!-- JS Demo -->

 <script src="{% static "assets/js/demo/demo.tables.js" %}"></script>

 <script src="{% static "assets/js/core/dandelion.core.js" %}"></script>

 <script src="{% static "assets/js/demo/demo.ui.js"%}"></script>

 <!-- JS Customizer -->

 <script src="{% static "assets/js/core/dandelion.customizer.js" %}"></script>

<!--[if lt IE 9]>

 <script src="/static/assets/js/libs/excanvas.min.js"></script>

 <![endif]-->

<script src="{% static"assets/js/plugins/circularstat/dandelion.circularstat.min.js" %}"></script>

 <script src="{% static "plugins/colorpicker/colorpicker.js" %}"></script>

 <link rel="stylesheet" href="{% static "plugins/colorpicker/colorpicker.css" %}" media="screen">

1. Para declarar las imágenes es necesarios utilizar las que están en el servidor de nginx y se deben de especificar las rutas de la siguiente manera.

**<img src="{% static "ruta/imagen" %}" alt="">**

**Ejemplo: Para mostrar el logotipo que se encuentra en nginx dentro de la carpeta logos.**

1. Para implementar los calendarios que se activan al hacer clic en una imagen se deben de declarar de la siguiente manera para que pueda aparecer el icono de lo contrario no se mostrará el icono.

**buttonImage: "{% static "assets/images/icons/led/src/icono" %}".**

 **Ejemplo:**

