**Documentación**

Aprenda a configurar y extender TastyIgniter para tomar pedidos en línea y administrar su restaurante, ya sea que sea un novato o un experto.

**Instalación**

Requerimientos mínimos

Estos son los requisitos para que su alojamiento web ejecute TastyIgniter:

PHP 7.2+

MySQL 5.6.10+

Extensión PHP PDO

Extensión PHP cURL

Extensión PHP OpenSSL

Extensión GD PHP

Extensión PHP Mcrypt

Extensión PHP Mbstrings

**Instalar TastyIgniter**

Hay dos formas de instalar TastyIgniter, ya sea usando las instrucciones de instalación rápida o desde la línea de comandos.

**Instalación rápida**

1. Descargue y descomprima el asistente de instalación de TastyIgniter en un directorio vacío en su servidor.
2. Cree una base de datos de usuario MySQL para TastyIgniter en su servidor de base de datos.
3. Sube las carpetas y archivos de TastyIgniter a tu servidor. Normalmente, el archivo setup.php estará en el directorio raíz de la web.
4. Otorgue permisos de escritura en el directorio de instalación, sus subdirectorios y archivos.
5. Ejecute el script de configuración de TastyIgniter accediendo a setup.php en su navegador web. Ejemplo, http://example.com/setup.php o http://example.com/folder/setup.php
6. Siga todas las instrucciones en pantalla y asegúrese de que todos los requisitos de instalación estén marcados.

**Instalación desde la línea de comandos**

TastyIgniter también usa composer para administrar sus dependencias, necesitará tener composer instalado en su máquina antes de usar la instalación de línea de comandos de TastyIgniter. Ejecute este comando en una ubicación vacía en la que desee instalar TastyIgniter:



Después de ejecutar el comando anterior, ejecute el comando de instalación y siga las instrucciones para completar la instalación



El comando de instalación lo guiará a través del proceso de configuración de TastyIgniter por primera vez. Le pedirá la configuración de la base de datos, la URL de la aplicación y los detalles del administrador.

**Pasos posteriores a la instalación**

Por razones de seguridad, se utilizó el Asistente de configuración de instalación rápida, debe eliminar los archivos de configuración. TastyIgniter nunca eliminará automáticamente archivos de su sistema, por lo que estos archivos y directorios deben eliminarse manualmente:



**Configurar el programador de tareas**

Debe agregar la siguiente entrada Cron a su servidor para que las tareas programadas funcionen correctamente. La edición de Crontab generalmente se realiza con el comando 



Asegúrese de reemplazar  con la ruta absoluta al archivo artesanal en su directorio raíz de TastyIgniter. Este Cron llamará al programador de comandos cada minuto. Al ejecutar el comando, TastyIgniter evaluará sus tareas programadas y ejecutará las tareas pendientes.

**Configurar el demonio de cola**

De forma predeterminada, la cola en TastyIgniter es sincrónica e intentará ejecutar tareas como enviar correos electrónicos en tiempo real. Este comportamiento se puede establecer en un método asincrónico cambiando el  parámetro en el 

Si estás usando el cola, debe ejecutar el proceso de cola como un servicio demonio. Utilice el siguiente comando: 

Se recomienda encarecidamente que ejecute este comando al iniciar el sistema. Si Cron está disponible, esto se puede lograr usando el siguiente formato: 

**Configuración del servidor web**

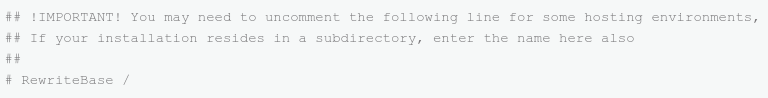
TastyIgniter tiene una configuración básica que debe aplicarse a su servidor web. Los servidores web comunes y su configuración se pueden encontrar a continuación.

**Configuración de Apche**

Existen algunos requisitos adicionales del sistema si su servidor web ejecuta Apache,  debe estar instalado y habilitado y el la opción debe estar activada.

TastyIgniter incluye un archivo: asegúrese de que se haya cargado correctamente.

Deberá descomentar esta línea en el archivo en algunos casos.



Si ha creado un subdirectorio, también puede agregar el nombre del subdirectorio:

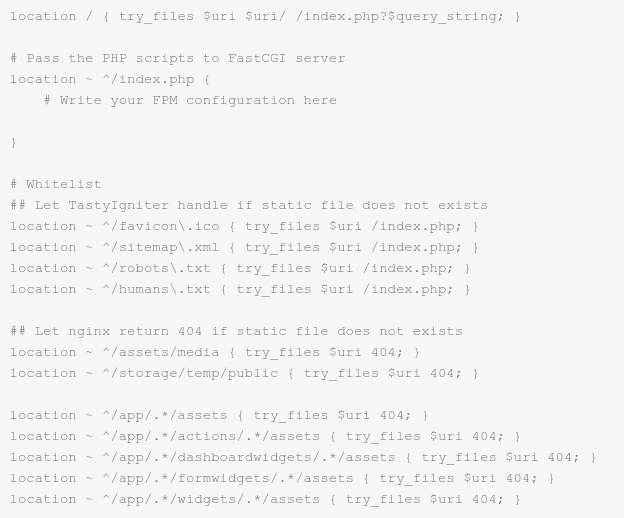


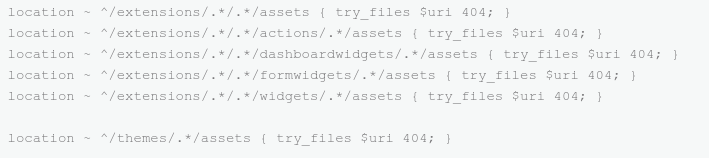
**Configuración de Nginx**

Agregue las siguientes líneas al bloque de configuración de su servidor:



Utilice el siguiente código en la sección del servidor. Si ha instalado TastyIgniter en un subdirectorio, reemplace las primeras directivas de ubicación / in con el directorio en el que se instaló TastyIgniter:





## **Configuración de la aplicación**

**Modo de depuración**

La configuración de depuración se encuentra en el archivo de configuración con el parámetro, y está habilitado de forma predeterminada.

Cuando está habilitada, esta configuración mostrará mensajes de error detallados cuando ocurran junto con otras funciones de depuración. El modo de depuración siempre debe estar deshabilitado en un sitio de producción en vivo. Esto evita la visualización de información potencialmente sensible al usuario final.

**Protección CSRF**

TastyIgniter ofrece un método simple para proteger su aplicación de falsificaciones de solicitudes entre sitios.

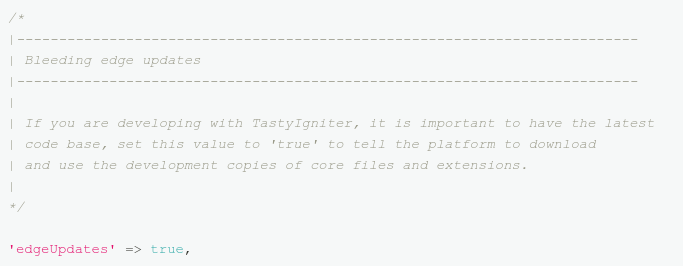
Para cada sesión de usuario activa administrada por la aplicación, TastyIgniter genera automáticamente un "token" CSRF. Este token se utiliza para comprobar que el usuario autenticado es el que realmente hace las solicitudes del cliente.

Aunque la seguridad CSRF está habilitada de forma predeterminada, puede deshabilitarla en el  archivo de configuración usando elparámetro.

**Actualizaciones del borde sangrante**

El núcleo de TastyIgniter y algunas extensiones del mercado introducirán cambios en dos etapas para garantizar la estabilidad y la integridad generales del código base. Esto significa que, además de las versiones estables normales, tienen una versión de prueba.

Modificando el parámetro en el archivo de configuración, puede indicarle a la aplicación que elija versiones de prueba del mercado.



Cuando utilice Composer para manejar las actualizaciones, reemplace los requisitos predeterminados de TastyIgniter en su archivo con lo siguiente para recibir actualizaciones de la rama de desarrollo directamente.



**Empezando**

Puede acceder al panel de administrador desde with your username and password asked during the setup process. After you’ve logged in you’ll be able to access the administration panel to configure your site.



## **Getting help**

* Si cree que ha encontrado un error, infórmelo utilizando el rastreador de problemas de GitHub, o mejor aún, bifurque el repositorio y envíe una solicitud de extracción.
* Si tiene comentarios para compartir, ideas que le gustaría implementar, por supuesto, compártalas en los foros de la comunidad de TastyIgniter.
* Únase a nosotros en nuestro canal de Slack para charlar con nosotros.

**Solución de problemas**

1. **Se muestra una página de error 404:** esto podría ser el resultado de que el módulo mod\_rewrite no se haya activado / instalado o configurado correctamente. Active mod\_rewrite para el servidor web Apache. Si ya está activado, verifique el archivo htaccess raíz en , para asegurarse de que el  el valor está configurado correctamente.
2. **Se muestra una pantalla en blanco al abrir la aplicación:** Verifique que los permisos de archivo estén configurados correctamente en el archivos y carpetas y grabables para el servidor web.
3. **La configuración se realizó correctamente, pero los enlaces de la tienda no funcionan:** compruebe que todas las extensiones necesarias del tema estén instaladas.



**TEMAS**

**Introducción**

Los temas de TastyIgniter son archivos que trabajan juntos para crear el diseño y la funcionalidad de un sitio TastyIgniter. Cada tema puede ser diferente, ofreciendo a los propietarios de sitios muchas opciones para cambiar el aspecto de su sitio web al instante. Este artículo trata sobre el desarrollo de temas para TastyIgniter. Los temas son subdirectorios predeterminados en el directorio / themes. El subdirectorio del tema contiene todas las páginas, parciales, diseños y archivos de activos y el archivo PHP de tema opcional (theme.php).

\*\*\*El tema activo se establece en el archivo config / system.php con el parámetro defaultTheme o en la página Diseño> Administración de temas con el Selector de temas.\*\*\*

**Estructura de directorios**

A continuación se muestra un ejemplo de una estructura de directorio de temas. Cada tema de TastyIgniter está representado por medio de un directorio separado y normalmente se activa un tema para mostrar el sitio web.



TastyIgniter admite un subdirectorio de un solo nivel para diseños, páginas, parciales y archivos de contenido (se puede usar cualquier estructura en el directorio de activos), lo que simplifica la organización de sitios web grandes.

Un tema también puede contener cualquier número de otros subdirectorios.

**Archivo de manifiesto del tema (opcional)**

Un  archivo tiene este aspecto:



|  |  |
| --- | --- |
| **Campo** | **Descripción** |
| Código | El código del tema, obligatorio. El valor se utiliza en el mercado TastyIgniter para establecer el valor del código del tema. |
| Nombre | Especifica el nombre del tema, obligatorio. |
| Descripción | La descripción del tema, obligatoria. |
| Autor | Especifica el nombre del autor, obligatorio. |
| Versión | Especifica la versión del tema, requerida. |
| Página principal | Especifica la URL del sitio web del autor, opcional. |
| Exigir | Una matriz de nombres de extensión de los que depende el tema, opcional. |

**Archivo de manifiesto de activos**

Antes de agregar sus archivos al archivo de manifiesto de activos del tema, debe colocarlos dentro de su tema en la estructura de directorio correcta, como se muestra en la estructura de directorio del tema.

Un archivo tiene este aspecto:



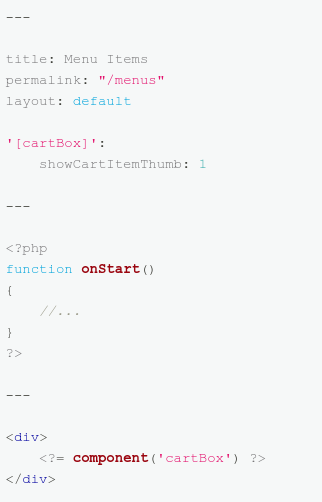
El código JavaScript debe colocarse en archivos externos siempre que sea posible. Utilizar para cargar sus scripts.

**Captura de pantalla**

Crea una captura de pantalla para tu tema. La captura de pantalla debe llamarse y debe colocarse en el directorio de nivel superior. El tamaño de imagen recomendado es de 1200 px de ancho por 900 px de alto. La captura de pantalla suele ser más pequeña, pero la imagen de gran tamaño permite una visualización de alta resolución en pantallas HiDPI.

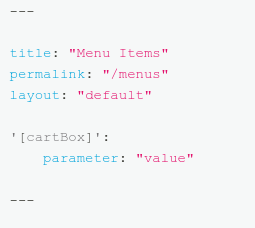
**Estructura de la plantilla**

Una plantilla es cómo se organiza una página y cómo se muestra el contenido de la página. Tanto los diseños como las plantillas de páginas pueden contener hasta tres secciones: contenido inicial, código PHP y marcado HTML. Las secciones están separadas por la secuencia ---. Por ejemplo:



**Sección de contenido preliminar**

La portada debe ser lo primero en el archivo y debe tomar la forma de un conjunto YAML válido entre líneas de tres puntos. Puede establecer variables predefinidas entre estas líneas de tres puntos (consulte a continuación para obtener una referencia) o incluso crear sus propias variables personalizadas. Estas variables estarán disponibles para que pueda acceder utilizando etiquetas PHP tanto más abajo en el archivo como en cualquier plantilla referenciada. A continuación, se muestra un ejemplo básico:



**Sección de código PHP**

El código dentro de la sección PHP se ejecuta cada vez antes de que se procese la plantilla. La sección PHP es opcional y el contenido depende del tipo de plantilla en la que se define. Las etiquetas de apertura y cierre de PHP siempre deben establecerse dentro del separador de sección --- en una línea diferente. Por ejemplo:



La sección PHP se convierte en una clase PHP cuando se analiza la plantilla, solo puede definir funciones y espacios de nombres, no se permite ningún otro código PHP. Puede acceder a  este objeto para leer y establecer variables.

**Marcado HTML**

La sección HTML define el marcado que debe representar la plantilla. El contenido de la sección HTML puede contener etiquetas HTML y PHP, funciones, el contenido depende de si la plantilla es una página, diseño o parcial.

**Incluyendo archivos de plantilla**

Para cargar un archivo parcial o de contenido desde otro archivo de plantilla, puede usar para facilitar que un tema reutilice secciones de código. Las rutas de la plantilla son siempre absolutas. Si renderiza un parcial del mismo subdirectorio aún es necesario especificar el nombre del subdirectorio, .

**Variables globales**

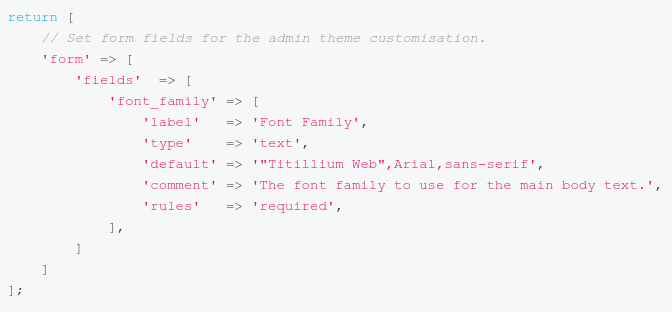
|  |  |
| --- | --- |
| **Variable** | **Descripción** |
| Tema | Objeto de tema para leer la configuración de personalización. |
| Página | Objeto de página . Las variables personalizadas configuradas a través de la página principal en las páginas estarán disponibles aquí. |
| Diseño | Objeto de diseño Las variables personalizadas configuradas a través de la parte delantera en los diseños estarán disponibles aquí. |
| Controlador | Acceder al objeto controlador subyacente |

**API de personalización**

La función de personalización de temas está disponible de forma predeterminada para casi todos los temas de TastyIgniter desde Diseño> Temas. La página de administración de personalización de temas se completa automáticamente con campos de formulario que un tema registra utilizando el archivo:

\*\*\*Tenga en cuenta que la pantalla Personalizar solo está disponible si el tema activo admite Personalización. Además, esta pantalla probablemente será diferente para cada tema que la habilita y la construye.\*\*\*

Ejemplo de un archivo.



Luego se puede acceder al valor dentro de cualquiera de las plantillas de tema:



**Mejores prácticas**

* Nombra tus archivos CSS y JavaScript principales con el mismo nombre que tu tema.
* Todos los archivos de plantilla (diseños, páginas, parciales, contenido) deben usar 
* Utilice rutas relativas en sus archivos CSS, por ejemplo: 
* Utilice nombres de archivo en minúsculas

**EXTENSIONES**

**Introducción**

Las extensiones son la base para agregar nuevas funciones a TastyIgniter extendiéndolo. El núcleo de TastyIgniter está diseñado para ser delgado y liviano, para maximizar la flexibilidad y minimizar el exceso de código. Con las extensiones, puede agregar funcionalidad a su sitio, en lugar de piratear el código central de TastyIgniter.

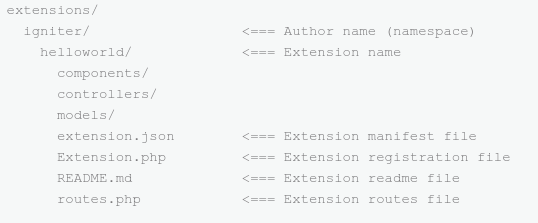
Una extensión se puede agregar al conjunto específico de características o servicios de TastyIgniter, tales como: definir componentes, definir permisos de personal, agregar o extender elementos de menú, listas y formularios, crear o alterar estructuras de tablas de base de datos y datos iniciales, alterar el núcleo u otra funcionalidad de extensión, proporcionar servicios, controladores de administración, vistas, activos y otros archivos.

**Cómo carga TastyIgniter las extensiones**

Cuando TastyIgniter carga la lista de extensiones en la página Sistema -> Extensiones del administrador de TastyIgniter, busca en la carpeta de (y sus subdirectorios de segundo nivel) para encontrar el archivo PHP Class 

**Estructura de directorios**

Las extensiones se encuentran en el subdirectorio / extensions del directorio de la aplicación. A continuación se muestra un ejemplo de una estructura de directorio de extensión.



**Creando una extensión**

Esta sección del artículo cubre los pasos que debe seguir, y algunas cosas que debe considerar, al crear una extensión de TastyIgniter bien estructurada.

**Nombrar su extensión**

El primer paso para crear una extensión es seleccionar un espacio de nombres y un nombre corto para ella. Este nombre de extensión Namespace.ShortName se usa para referirse a su extensión por el núcleo TastyIgnier. El espacio de nombres se utilizará como su código de autor al publicar sus extensiones en el mercado de TastyIgniter.

Tanto el espacio de nombres como el nombre de la extensión deben seguir estas reglas importantes:

* Solo se deben proporcionar cartas.
* Los nombres de las carpetas deben estar en minúsculas, como se muestra en el ejemplo de estructura de directorios.
* No debe contener espacios.
* Debe ser único. El nombre de su extensión no debe ser el mismo que el de ninguna otra extensión o tema.

Aquí tienes un ejemplo: **Igniter.HelloWorld**

Dado el ejemplo anterior **Igniter.HelloWorld,** crear una extensión es bastante simple y directo, simplemente llame al siguiente comando desde el directorio de la aplicación para generar un directorio de extensión básico y archivos:

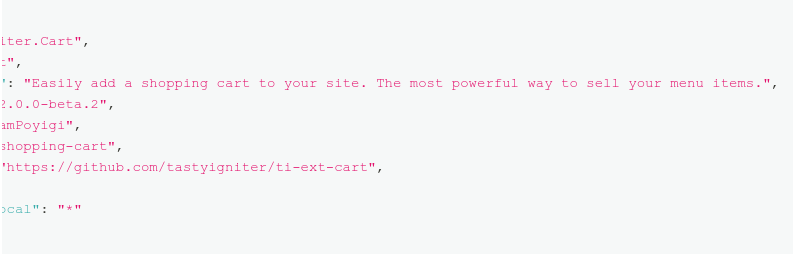


\*\*\*Le recomendamos encarecidamente que siga los estándares de codificación de TastyIgniter al crear sus propias extensiones. Es un requisito para cualquier cambio en el código central de TastyIgniter.\*\*\*

**Archivo de manifiesto de extensión**

Una , archivo de manifiesto es una parte importante de la extensión TastyIgniter para almacenar metadatos sobre la extensión. El archivo de manifiesto se puede utilizar para definir las dependencias de la extensión.





\*\*\*El parámetro  de un archivo de manifiesto de extensión define las dependencias de la extensión. En el ejemplo anterior, la extensión igniter.cart depende de la extensión igniter.local.\*\*\*

**Archivo Léame**

Si desea publicar su extensión en el mercado de TastyIgniter, debe crear un archivo readme.md en un formato estandarizado en su directorio de extensiones.

**Archivo de rutas**

Las extensiones también pueden proporcionar un archivo llamado route.php con lógica de enrutamiento personalizada como se describe en el artículo de enrutamiento.

**Registro**

Un archivo Extension.php (también conocido como archivo de registro de extensión) es una parte esencial de la extensión TastyIgniter para proporcionar métodos para extender el núcleo de TastyIgniter.

El script de registro debe definir una clase de nombre de extensión que amplíe la clase  El siguiente es un ejemplo de archivo de registro de extensión.





\*\*\*Se recomienda definir los metadatos de la extensión utilizando el archivo de manifiesto en lugar de anular el método .

**Métodos de clases de registro**

Los siguientes métodos son compatibles con la clase de registro de extensiones:

|  |  |
| --- | --- |
| **Método** | **Descripción** |
| Registro ( ) | Método de registro, llamado cuando el complemento se registra por primera vez |
| Boot ( ) | Método de arranque, llamado justo antes de la ruta de solicitud. |
| ExtensiónMeta ( ) | Devuelve metadatos sobre la extensión, cuando no se proporciona un extension.json. |
| RegisterComponents ( ) | Registra cualquier componente de interfaz proporcionado por esta extensión. |
| RegisterNavigation ( ) | Registra los elementos del menú de navegación del administrador para esta extensión, consulte a continuación, por ejemplo. |
| RegisterPermissions ( ) | Registra cualquier permiso de personal proporcionado por esta extensión. |
| RegisterSettings ( ) | Registra cualquier página de configuración de administrador proporcionada por esta extensión. |
| RegisterDashboardWidgets ( ) | Registra cualquier widget del panel de administración, proporcionado por esta extensión. |
| RegisterFormWidgets ( ) | Registra los widgets de formulario de administración proporcionados por esta extensión. |
| RegisterMailLayouts ( ) | Registra cualquier diseño de vista de correo proporcionado por esta extensión. |
| RegisterMailTemplates ( ) | Registra cualquier plantilla de vista de correo proporcionada por esta extensión, ver más abajo por ejemplo. |
| RegisterMailPartials ( ) | Registra todos los parciales de vista de correo proporcionados por esta extensión. |
| RegisterPaymentGateways ( ) | Registra cualquier pasarela de pago proporcionada por esta extensión. |

Un ejemplo de registro de archivo de vista de plantillas de correo 



Un ejemplo de registro de un elemento de menú de navegación de menú secundario y de nivel superior:



Para registrar un middleware personalizado, puede llamar a cualquiera de los siguientes en su método de arranque.



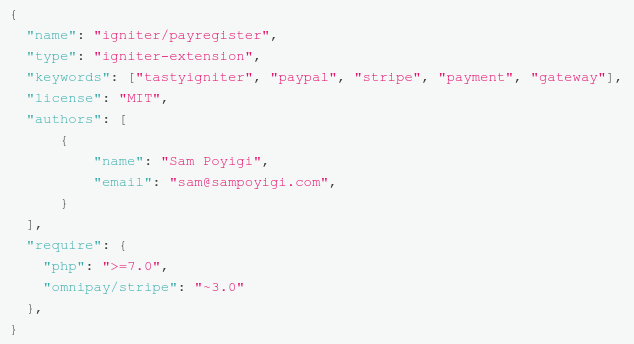
Un ejemplo de registro de comandos de consola proporcionados por la extensión Igniter.Api:



**Usando Composer**

Hay una serie de escenarios que requieren que el desarrollador agregue un archivo composer.json a su extensión. Algunos de estos escenarios dependen de si la extensión dependerá de paquetes de terceros.

Aquí hay un ejemplo completo de cómo la extensión Igniter.PayRegister usa composer.json para depender del paquete de terceros 



El nombre y el tipo son propiedades importantes que se utilizan para decirle al paquete del cómo instalar realmente la dependencia.

\*\*\*No confirme el directorio , el archivo  debe estar comprometido con el repositorio.\*\*\*

**Proceso de instalación de dependencias**

El mercado toma los siguientes pasos adicionales al preparar extensiones que definen dependencias de terceros mediante .

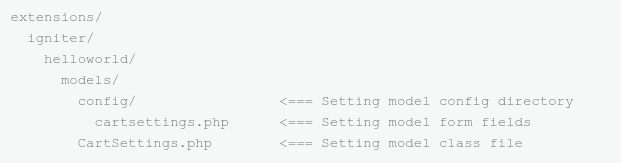
* Inyecta dependencias ya incluidas en el núcleo de TastyIgniter en la propiedad de de .
* Ejecuta la  en el directorio de extensiones.
* Elimina los archivos  y para evitar la duplicación de dependencias con el núcleo , ya que ya existe una carpeta  en el directorio de extensión.
* Por último, el resultado final está empaquetado y listo para ser consumido a través de la API de actualización por el Administrador de actualizaciones de TastyIgniter.

**Ajustes y configuración**

Hay dos formas de configurar extensiones: con configuraciones de administrador (configuraciones de base de datos) y archivos de configuración.

**Usando el modelo de configuración**

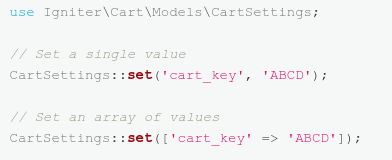
Los modelos de configuración, como cualquier otro modelo, residen en el subdirectorio / modelos del directorio de extensiones.



Al implementar la clase de acción  y extender la  de modelo base en una clase de modelo, puede crear modelos para almacenar configuraciones en la base de datos.



Un ejemplo de escritura en un modelo de configuración:



Un ejemplo de lectura de un modelo de configuración:



**Usando la configuración basada en archivos**

Las extensiones pueden tener un archivo de configuración en el subdirectorio del directorio de extensiones. Los archivos de configuración son scripts PHP que definen y devuelven una matriz, p. Ej. Archivo de configuración :

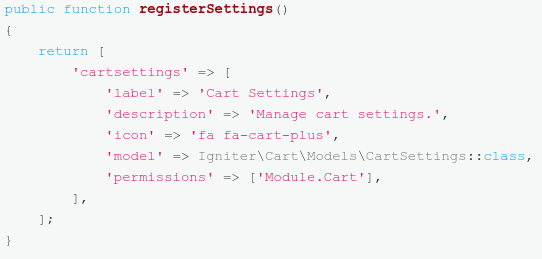


Utilice la clase para acceder a los valores de configuración definidos en el archivo de configuración.



**Registro de enlace de configuración**

Un ejemplo que muestra cómo crear un enlace de configuración de administrador a un modelo de configuración. Los enlaces de configuración registrados aparecerán en la página **Sistema>Administrador** de configuración.



**Extender una extensión**

¿Por qué ampliar una extensión con otra extensión? Para mantener los cambios compatibles, si la extensión recibe una actualización. Las extensiones se extienden principalmente para inyectar o modificar la funcionalidad de las clases principales y otras extensiones que utilizan el servicio de eventos.

El lugar más común para suscribirse a un evento es el método de inicio del archivo de registro de extensión.

**Pruebas de escritura**

Si desea comenzar a escribir una nueva extensión, recuerde que las pruebas unitarias lo ayudan a entregar extensiones de mayor calidad.

**COMPONENTES**

**Introducción**

TastyIgniter Components implementa contenido y características que amplían su sitio web. Se pueden agregar, eliminar y reorganizar desde Diseño> Temas> Editor en el Panel de administración. Además de mostrar el marcado HTML en una página, los componentes pueden implementar el manejo de solicitudes AJAX, el manejo de devoluciones de datos y el manejo del ciclo de vida de ejecución de la página, lo que permite inyectar páginas. Este artículo describe el uso de componentes dentro de los temas y no explica la creación de componentes como parte de las extensiones.

**Adjuntar componentes**

Utilice la interfaz de administración para adjuntar componentes a páginas y diseños. Puede adjuntar un componente a una página o diseño manualmente utilizando un editor de archivos siguiendo el ejemplo siguiente:



El componente cartBox en el ejemplo anterior inicializa la propiedad showCartItemThumb con el valor 1.

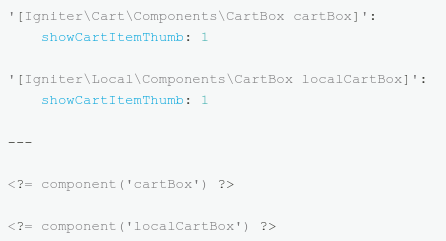
Cuando adjunta un componente, se crea automáticamente una variable de página que coincide con el nombre del componente ($ cartBox en el ejemplo anterior). Renderice el marcado HTML de los componentes en una página o diseño de la siguiente manera:



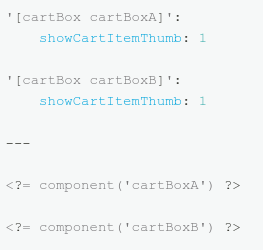
**Usar varias instancias del mismo componente**

Si dos componentes con el mismo nombre se asignan a una página y diseño juntos, el componente de página anula cualquier propiedad del componente de diseño.

Puede adjuntar componentes del mismo nombre registrados dentro de dos extensiones utilizando su nombre de clase totalmente calificado y asignándole un alias:

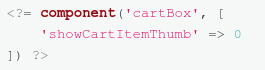


Defina varias instancias de un mismo componente:



**Variables componentes**

La etiqueta .Acepta una matriz de variables como segundo parámetro. Las variables especificadas estarán disponibles en el momento de la renderización y anularán explícitamente los valores de propiedad del componente:



El valor de la propiedad showCartItemThumb del componente se establecerá en 0 cuando se procese el componente.

**Anulación de parciales de componentes**

Todos los parciales de componentes se pueden anular creando parciales de tema. Un componente definido con alias **cartBox** puede tener su **control.php** parcialmente anulado creando un archivo de tema llamado **\_partials / cartBox / control.php.**

**PERMISOS**

**Introducción**

El sistema de permisos controla el acceso y la restricción a todas las partes de una aplicación TastyIgniter. Hay superadministradores en el nivel más bajo (usuarios con la marca establecida en) y administradores (personal) y permisos. Los modelos y son los contenedores que contienen toda la información importante sobre un administrador.

Los superadministradores tienen acceso a todo en el sistema y solo pueden ser administrados por ellos mismos o por otros superadministradores; no son accesibles ni editables por los administradores habituales, incluso si un administrador tiene el permiso de .

**Cómo funcionan los permisos**

Los permisos son claves de cadena en forma de que se le otorga a un administrador por herencia a través del Rol del personal.

Al determinar si un personal tiene un permiso específico, se aplicará la configuración de permisos para la función de ese personal. Por ejemplo, si el administrador Sam pertenece al rol de **Chef**, y el rol de **Chef** tiene permiso de , **Sam** llegará a****.

Los roles del personal son grupos de permisos con un nombre que se utiliza para identificar el rol de un administrador. Solo se puede asignar un rol a un personal a la vez. Se puede asignar una función a varios miembros del personal. TastyIgniter se envía con cuatro roles de sistema por defecto; y . Se puede crear y aplicar cualquier número de roles personalizados a los administradores con sus propias combinaciones de permisos.

Los grupos de personal no tienen nada que ver con los permisos y son estrictamente una herramienta administrativa para agrupar administradores. Por ejemplo, los grupos se pueden utilizar para asignar un pedido a todos los administradores de la cocina.

**Registro de permisos**

Los permisos de administrador se pueden registrar por extensiones anulando el método dentro de la clase de registro de Extensión. Los permisos se definen como una matriz con claves que coinciden con las claves de permisos y los valores que contienen el grupo de permisos y las descripciones. Las claves de permiso consisten en el nombre del autor, el nombre de la extensión y el nombre de la función. Aquí hay un ejemplo:

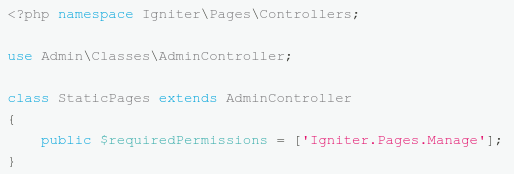


Los controladores de administración pueden usar permisos definidos por extensiones para restringir el acceso de administrador a páginas o funciones. Los permisos se definen con una clave de permiso y una descripción. El siguiente ejemplo muestra cómo registrar elementos con permisos de administrador.



**Restringir el acceso a las páginas de administración**

Dentro de una clase de controlador de administrador, puede especificar qué permisos se requieren para acceder a las páginas del controlador, utilizando la propiedad del controlador . Esta propiedad debe contener una matriz de claves de permisos. El sistema permitiría al administrador acceder a las páginas del controlador si los permisos del administrador coinciden con CUALQUIER permiso de la lista.



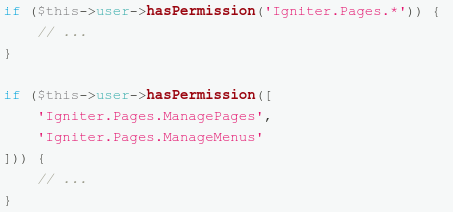
El símbolo de asterisco también se puede utilizar para indicar la condición de "todos los permisos". Las páginas del controlador están disponibles en el siguiente ejemplo para todos los administradores que tienen permisos que comienzan con "Igniter.Pages". cuerda:



**Restringir el acceso a las funciones**

El modelo de usuario admin tiene el método para determinar si el administrador tiene permisos específicos. Para restringir la funcionalidad de la interfaz de usuario de administrador, puede utilizar esta función. Este método toma dos parámetros: la cadena de claves de permisos (o una matriz de cadenas de claves) y el parámetro opcional que especifica que todos los permisos especificados con los primeros parámetros son obligatorios.

El método devuelve para cualquier permiso si el administrador es un superadmin ( establecido en) o si el administrador tiene los permisos especificados a través de su grupo. El siguiente ejemplo ilustra cómo utilizar los métodos en el código del controlador:



**LOCALIZACIÓN**

**Introducción**

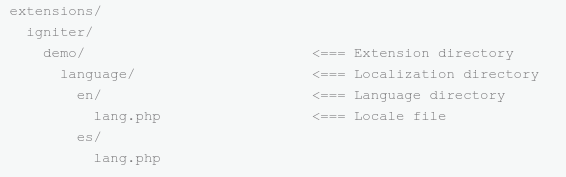
TastyIgniter tiene un potente sistema de traducción que permite que su sitio admita contenido multilingüe. Al desarrollar su extensión, debe considerar el uso de cadenas de configuración regional incluso si no tiene la intención de usarlas en más de un idioma. Nunca se sabe: si decide que su extensión esté disponible para usuarios de todo el mundo, puede ser útil más adelante.

Recomendamos encarecidamente incluir un conjunto completo de recursos en inglés con cada extensión, también si decide publicar su extensión en el mercado de TastyIgniter.

Los archivos de configuración regional se almacenan en el subdirectorio de **extensión / idioma.**

**Estructura de directorios**

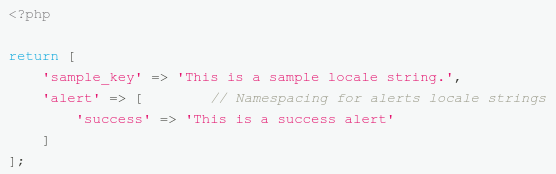
A continuación se muestra un ejemplo del directorio de idiomas de la extensión:



\*\*\*Cada directorio de idioma debe ser nombrado usando el código ISO 639-1 para el idioma que contiene.\*\*\*

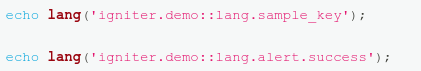
**Definición de cadenas de configuración regional**

Todos los archivos de configuración regional devuelven una matriz de cadenas de configuración regional con clave. Por ejemplo:



**Acceder a cadenas de configuración regional**

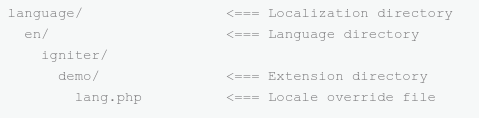
Puede utilizar la función auxiliar para obtener cadenas de archivos de configuración regional. El método acepta el archivo y la clave de la cadena de configuración regional como primer argumento. Por ejemplo, usemos la cadena de configuración regional de las extensiones de archivo de configuración 



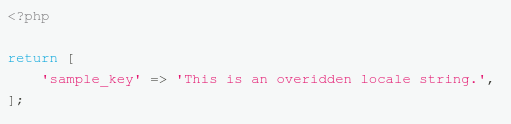
**Anular cadenas de configuración regional**

Los usuarios del sistema pueden anular las cadenas de la configuración regional de la extensión sin alterar los archivos de extensión para modificar estas cadenas. Por ejemplo, debe crear el archivo de configuración regional  en la siguiente ubicación para anular la cadena de configuración regional dentro del archivo lang.php de la extensión .

\*\*\*También puede editar fácilmente las cadenas de configuración regional principales y / o de extensión desde la interfaz de administración en la página Localización> Idiomas> Traducciones.\*\*\*



Solo debe definir las cadenas de configuración regional que desea anular en este archivo. Cualquier cadena de configuración regional que no anule se seguirá cargando desde los archivos de configuración regional originales de la extensión.



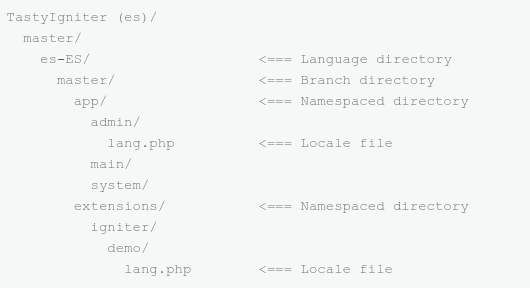
**Haciendo su sitio multilingüe**

En esta sección del artículo, encontrará una descripción general completa de los pasos involucrados en la creación de un sitio TastyIgniter multilingüe.

**Descargar manualmente un paquete de idioma**

Siga estos pasos para descargar manualmente un paquete de idioma traducido por la comunidad:

* Únase a nuestra página del proyecto Crowdin de traducciones.
* Elija el idioma que desea instalar. Por ejemplo, elijamos español (ES). Descarga y descomprime el paquete de idioma.
* Para descargar, debe hacer clic en el botón en la parte superior derecha de la página de idiomas de Crowdin.
* La estructura de carpetas y archivos del paquete de idioma extraído debería tener este aspecto.



Copie los archivos y carpetas dentro de los directorios con espacio de nombres en su directorio de idioma TastyIgniter, vea a continuación. Si no tiene un directorio de idioma en la raíz de su aplicación, cree uno nuevo.



\*\*\*Observe que el nombre del directorio de idiomas usa un guión bajo en lugar de un guión (por ejemplo, "es\_ES").\*\*\*

**Instalación de un paquete de idioma**

Después de descargar un paquete de idioma, debe crear un nuevo idioma en la interfaz de administración para registrar el idioma en el sistema.

* Cree un nuevo idioma desde la página **Localización> Idiomas** de la interfaz de administración.
* Rellene el formulario. El valor del de configuración regional debe coincidir con el nombre del directorio de idioma. Usando el ejemplo anterior, el valor será 
* Por último, cambie el interruptor de **Estado** a y luego guarde el formulario.
* Asegúrese de que el idioma aparezca en el campo **Idiomas admitidos** en la página **Sistema> Configuración> General** de la interfaz de administración

**Configuración del idioma predeterminado**

Es posible que desee configurar el paquete de idioma instalado como el idioma predeterminado para los nuevos usuarios y visitantes una vez que esté seguro de que funciona. Puede hacer esto en la página **Sistema> Configuración> General** de la interfaz de administración, en la pestaña **Sitio** seleccione el idioma que desea usar como predeterminado.

**Habilitación de la detección de idioma**

TastyIgniter incluye soporte para la negociación de idiomas lista para usar utilizando una variedad de métodos sin obligar al usuario a elegir su idioma. El idioma se detecta en la siguiente secuencia:

* Determine el idioma a partir del parámetro de **solicitud / sesión**.
* Determine el idioma en la configuración de idioma del navegador del usuario.

**Extensiones de terceros**

Si bien los paquetes de idiomas descargados desde la página **examinar idiomas** de la interfaz de administración pueden proporcionar traducciones para todas las extensiones recomendadas incluidas con TastyIgniter, como regla general, no cubrirán ninguna extensión de terceros que pueda haber instalado. Los desarrolladores son responsables de proporcionar y mantener las traducciones de sus extensiones.

Por lo tanto, antes de instalar una extensión de terceros, debe verificar que incluya traducciones para cada paquete de idioma que haya instalado. Si descubre que una extensión no es compatible con un idioma que necesita, comuníquese directamente con el desarrollador y haga arreglos para que se agreguen las traducciones necesarias.