

# Como empezar con Rails 5.1 y Ruby 2.4

En esta guía te explicamos los pasos necesarios para empezar a crear una aplicación Rails 5.1 y alojarla en uno de nuestros planes de hosting.

Ten en cuenta que debes [acceder mediante SSH a tu cuenta de hosting](#) y ejecutar algunos comandos.

## 1.- Crea aplicación Ruby desde el panel de hosting

Accede al panel de Hosting y luego pincha sobre el icono **Aplicaciones Ruby**. Una vez allí rellena y envía el formulario, tal y como se ve en la siguiente imagen:



The screenshot shows a web interface for creating a Ruby application. At the top, there is a breadcrumb trail: Home > Hosting > Hosting Ultimate midominio123.com > Panel de Hosting > Aplicaciones Ruby. The main heading is 'Aplicaciones Ruby' with a green 'MOSTRAR AYUDA' button. Below this is the section 'Crear nueva aplicación Ruby'. The form has three main fields: 'Nombre de la aplicación:' with the value 'miapp' and a note 'Se usará como nombre de directorio de la aplicación, nada de caracteres raros.'; 'Dominio:' with a dropdown menu showing 'ruby.midominio123.com'; and 'Stack:' with a dropdown menu showing 'Ruby Fox (Ruby 2.4 + Rubygems 2.6.x)' and a note 'Se permite un único stack por cuenta de hosting. Una vez creado una app, el resto usarán el mismo stack'. At the bottom, there are two buttons: 'Cancelar' and 'Enviar'.

En este caso, poniendo **miapp** como nombre de la aplicación, la ruta completa de la aplicación será **/home/usuario/ruby/miapp**, donde **usuario** es el nombre de usuario de tu hosting.

Aplicaciones Ruby

La nueva aplicación ruby ha sido configurada correctamente.  
Ahora debes ubicar los archivos de la aplicación en el directorio `/home/midominio/ruby/miapp` de tu cuenta de hosting.

Tus aplicaciones Ruby

| Aplicación             | Protocolo    | Directorio                | Stack   |
|------------------------|--------------|---------------------------|---|
| ruby.midominio123.com/ | HTTP y HTTPS | /home/midomini/ruby/miapp | ruby-fox <a href="#">Reiniciar</a> <a href="#">Borrar</a> |

## 2.- Instala Rails y Bundler

Ahora necesitamos acceder via SSH para empezar a instalar nuestra aplicación Rubi 5.1

Para instalar la versión 5.1 de Rails ejecutamos:

```
<br />  
gem install rails -v5.1<br />
```

En principio, al instalar Rails 5.1 también se habrá instalado la última versión compatible de Bundler. Si por algún motivo no se ha instalado, lo puedes instalar ejecutando:

Y luego instalamos Bundler:

```
<br />  
gem install bundler<br />
```

## 3.- Crea la aplicación Rails en tu hosting

Al configurar la nueva aplicación desde el panel de hosting, te habremos indicado cual es el directorio donde debes ubicar los archivos de la misma. En este caso, el directorio de la aplicación se debe llamar **miapp**.

vamos a acceder al directorio «*ruby*», borrar el directorio «*miapp*» y crear una

nueva instalación de rails con el nombre «*miapp*» y así volver a crear el directorio.

```
<br />  
cd ruby<br />  
rm -fr miapp<br />  
rails new miapp<br />
```

Y seguido entramos al directorio de nuestra aplicación:

```
<br />  
cd miapp<br />
```

## 4.- Edita el archivo Gemfile

Ahora procederemos a realizar las modificaciones en el fichero Gemfile según las necesidades de nuestro proyecto.

Para hacer esto puedes usar el editor *nano*:

```
<br />  
nano Gemfile<br />
```

Para **grabar** el archivo **Gemfile** presionas **CTRL O**, y este se grabara, luego para **salir** del editor presionas **CTRL X**

Ahora puedes elegir entre hacer uso de la gema **The Ruby Racer** o bien **execjs**. Recomendamos la segunda de las opciones.

### The Ruby Racer (Opcional)

Es preferible utilizar EXECJS si nuestra aplicacion lo permite.

En este archivo descomentamos la linea que dice:

```
<br />  
#gem 'therubyracer', platforms: :ruby<br />
```

Donde quitaremos el simbolo #, dejandolo así:

```
<br />  
gem 'therubyracer', platforms: :ruby<br />
```

## NodeJs - execjs (Recomendado)

```
</p>  
<p>nano  
~/ruby/gems/gems/execjs-2.7.0/lib/execjs/runtimes.rb</p>  
<p>
```

```
<br />  
Node = ExternalRuntime.new(<br />  
name: Node.js (V8),<br />  
# command: [nodejs, node],<br />  
command: /opt/nodejs-8/bin/node,<br />  
runner_path: ExecJS.root + /support/node_runner.js,<br />  
encoding: 'UTF-8';<br />  
)<br />
```

*Como usar nodejs en tu aplicación Ruby a través de la gema execjs*

## 5.- Instala dependencias con Bundler

Ahora tendrás que instalar las gemas necesarias para tu aplicación, para ello ejecuta:

```
<br />  
bundle install<br />
```

## 6.- Define el secret\_key\_base

Lo primero es obtener una cadena suficientemente larga y aleatoria, para ello ejecutamos:

```
<br />  
rake secret<br />
```

Se nos mostrará la cadena que usaremos como **secret\_key\_base** de la aplicación. Ahora abrimos el archivo **config/secrets.yml**:

```
<br />  
nano config/secrets.yml<br />
```

Y pegamos nuestra cadena en la sección **production** del archivo.

## 7.- Reinicia la aplicación

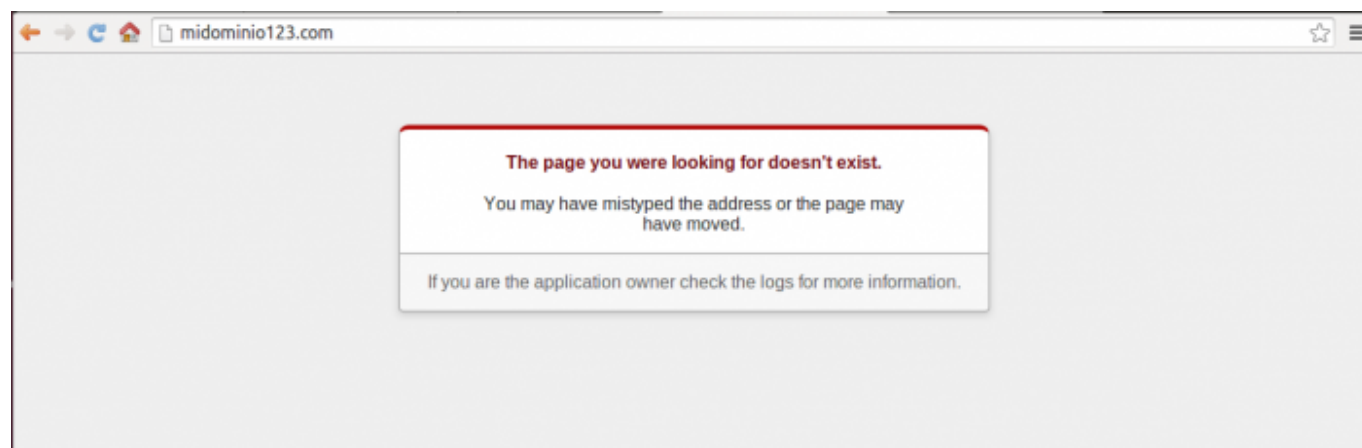
Simplemente ejecuta:

```
<br />  
touch tmp/restart.txt<br />
```

Opcionalmente, puedes reiniciar la aplicación desde el panel de hosting de guebs.

## 8.- Verifica el estado de la instalación

Para verificar el estado de la instalación nos vamos a nuestra pagina web, donde deberá mostrarnos una pantalla como la que se muestra a continuación:



## 9.- Genera el controlador de Bienvenida

Simplemente ejecuta:

```
<br />  
rails generate controller welcome index<br />
```

## 10.- Personalizar el mensaje de bienvenida

Abrimos el archivo de vista **app/views/welcome/index.html.erb** generado en el paso anterior:

```
<br />
nano app/views/welcome/index.html.erb<br />
```

Borramos todo el contenido del archivo y colocamos solamente:

```
</p>
<h1>Hola, Rails!</h1>
<p>
```

En el editor *nano*, para **grabar** el archivo Gemfile presionas **CTRL O**, y este se grabara, luego para **salir** del editor presionas **CTRL X**

## 11.- Configura ruta para bienvenida

Vamos a hacer que Rails muestre en la raíz de la web, es decir, en «/», el controlador y vista generados en los pasos previos.

Para ello abrimos el archivo config/routes.rb :

```
<br />
nano config/routes.rb<br />
```

Y añadimos la siguiente línea antes del operador **end**

```
<br />
root &#8216;welcome#index&#8217;<br />
```

El archivo podría quedar así:

```
<br />
Rails.application.routes.draw do<br />
  get &#8216;welcome/index&#8217;</p>
<p>root &#8216;welcome#index&#8217;<br />
# For details on the DSL available within this file, see
```

```
http://guides.rubyonrails.org/routing.html<br />  
end<br />
```

En el editor Nano, para **grabar** el archivo Gemfile presionas **CTRL O**, y este se grabara, luego para **salir** del editor presionas **CTRL X**

## 12.- Precompilar assets

Ejecutamos la precompilacion de los recursos como CSS:

```
<br />  
RAILS_ENV=production bundle exec rake assets:precompile<br />
```

## 13.- Reinicia la aplicación

Simplemente ejecuta:

```
<br />  
touch tmp/restart.txt<br />
```

Opcionalmente, puedes reiniciar la aplicación desde el panel de hosting de guebs.

## 14.- Accede a la web

Finalmente, después de haber modificado el contenido del controlador de Bienvenida, verificamos en nuestra pagina Web

