

UNIVERSIDAD NACIONAL AUTONOMA DE NICARAGUA



UNAN - León

FACULTAD DE CIENCIAS Y TECNOLOGIA

Carrera: Ingeniería en Telemática

Componente: Software como un servicio

Grupo:1

Docente: Erving Montes

Elaborado por:

- Alli Gissiell Herrera Chow.

Fecha: 28 de agosto de 2024

1. Realizar cada uno de los enunciados de la guía y verificar el correcto funcionamiento.

1. Verificar las versiones de software instaladas.

Es importante verificar las versiones de software instaladas, para así poder continuar con la realización de la guía sin ningún problema.

1.1. Verificar la versión de Rails.rail

```
root@debian:/home/gia# rails -v
rails 7.1.3.4
```

1.2. Verificar la versión de Ruby.

```
root@debian:~# ruby -v
ruby 3.1.2p20 (2022-04-12 revision 4491bb740a) [x86_64-linux]
root@debian:~#
```

1.3. Verificar la versión de sqlite.

```
root@debian:~# sqlite3 --version
3.40.1 2022-12-28 14:03:47 df5c253c0b3dd24916e4ec7cf77d3db5294cc9fd45ae7b9c5e82a
d8197f3alt1
root@debian:~#
```

2. Crear un nuevo proyecto en Rails.

2.1. Haciendo uso del terminal, ubicarse en el directorio donde se va a almacenar el proyecto.

```
root@debian:/home/gia# mkdir Proyectos_RoR
root@debian:/home/gia# ls
Descargas  Escritorio  mi_prooo    Música      prac3        Público  Vídeos
Documentos  Imágenes   mi_proyecto Plantillas  Proyectos_RoR  ruby
root@debian:/home/gia#
```

2.2. Generar un proyecto nuevo utilizando el siguiente comando:

```
root@debian:/home/gia# cd Proyectos_RoR/
root@debian:/home/gia/Proyectos_RoR# rails new first_app
  create
  create  README.md
  create  Rakefile
  create  .ruby-version
  create  config.ru
  create  .gitignore
  create  .gitattributes
  create  Gemfile
  run     git init -b main from "."
Inicializado repositorio Git vacío en /home/gia/Proyectos_RoR/first_app/.git/
  create  app
  create  app/assets/config/manifest.js
  create  app/assets/stylesheets/application.css
  create  app/channels/application_cable/channel.rb
  create  app/channels/application_cable/connection.rb
  create  app/controllers/application_controller.rb
  create  app/helpers/application_helper.rb
  create  app/jobs/application_job.rb
  create  app/mailers/application_mailer.rb
```

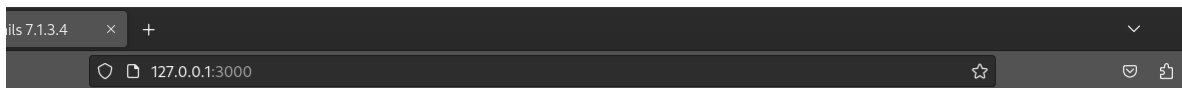
2.3. Entrar a la carpeta del proyecto creado.

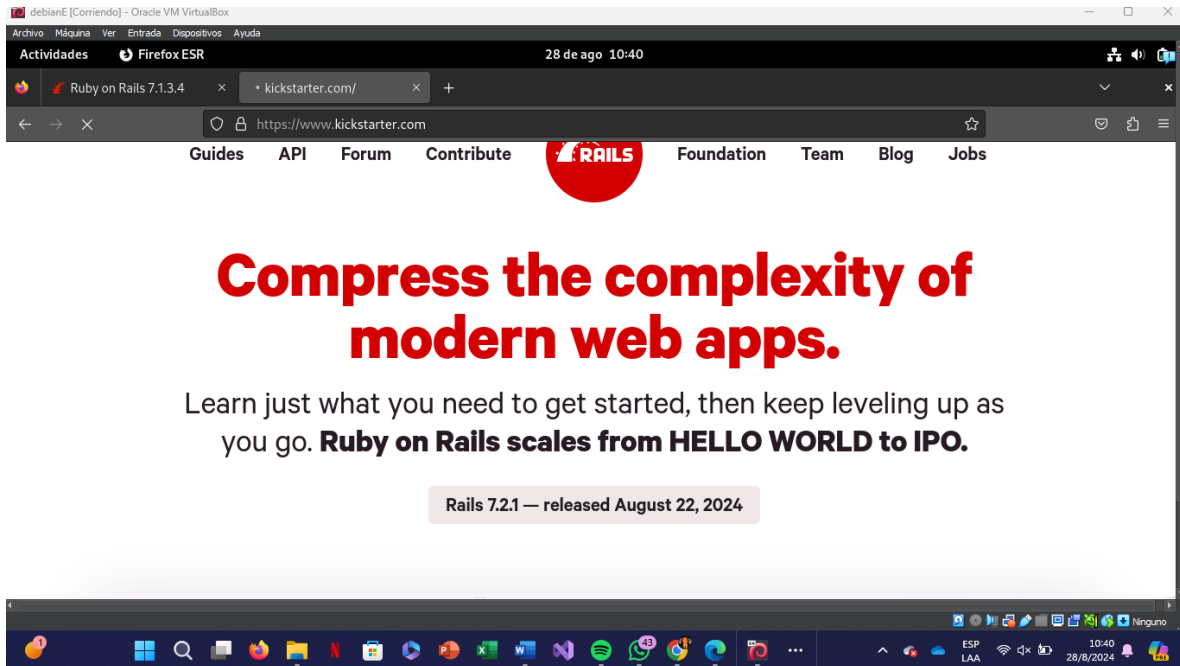
```
root@debian:/home/gia/Proyectos_RoR# cd first_app/
root@debian:/home/gia/Proyectos_RoR/first_app#
```

2.4. Una vez ubicado dentro del directorio del nuevo proyecto creado, iniciar el servidor para comprobar que arranca sin ningún problema.

```
root@debian:/home/gia/Proyectos_RoR/first_app# rails server
=> Booting Puma
=> Rails 7.1.3.4 application starting in development
=> Run `bin/rails server --help` for more startup options
Puma starting in single mode...
* Puma version: 6.4.2 (ruby 3.1.2-p20) ("The Eagle of Durango")
*   Min threads: 5
*   Max threads: 5
*   Environment: development
*         PID: 2899
* Listening on http://127.0.0.1:3000
* Listening on http://[::1]:3000
Use Ctrl-C to stop
█
```

2.5. Abrir el navegador y escribir la siguiente dirección.





3. Agregar una acción al controlador.

3.1. Cuando se crea un proyecto nuevo en Rails, se crea por defecto un controlador llamado `Application_controller`. Para Verificar que existe el controlador, puede ejecutar el siguiente comando en el terminal.

```
root@debian: /home/gia/Proyectos_RoR/first_app# ls app/controllers/ *_controller.rb
app/controllers/application_controller.rb
root@debian: /home/gia/Proyectos_RoR/first_app#
```

3.2. Abrir el archivo del controlador que se encuentra en la dirección `app/controllers/application_controller.rb` en el editor de texto, y definir la acción del controlador.



```
GNU nano 7.2 app/controllers/application_controller.rb
class ApplicationController < ActionController::Base
  protect_from_forgery with: :exception
  def hola
    render html: "hola, mundo!"
  end
end
```

4. Configurar el archivo routes.rb.

El archivo routes.rb es donde se definen las rutas de la aplicación en Rails, cada ruta está compuesta por: verbos HTTP (GET, POST, PATCH, PUT, DELETE), path, controlador y método (acción). Otra parte importante es el método root, este método es por medio del cual se establece la ruta raíz de la aplicación, esta ruta solicita get a una acción, es recomendable que esta ruta siempre esté en la parte superior del archivo routes.

4.1. Abrir el archivo routes (config/routes.rb) y agregar la siguiente línea

```
GNU nano 7.2 routes.rb
rails.application.routes.draw do
  # Define your application routes per the DSL in https://guides.rubyonrails.org/routing.html

  # Reveal health status on /up that returns 200 if the app boots with no exceptions
  # Can be used by load balancers and uptime monitors to verify that the app is up
  get "up" => "rails/health#show", as: :rails_health_check

  # Defines the root path route ("/")
  # root "posts#index"
end
```

```
gia@debian: ~
GNU nano 7.2 routes.rb *
rails.application.routes.draw do
  # Define your application routes per the DSL in https://guides.rubyonrails.org/routing.html

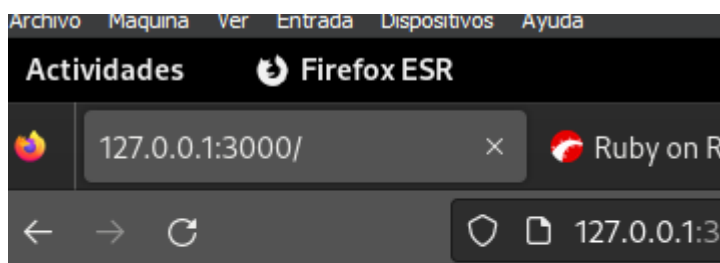
  # Reveal health status on /up that returns 200 if the app boots with no exceptions
  # Can be used by load balancers and uptime monitors to verify that the app is up
  get "up" => "rails/health#show", as: :rails_health_check

  # Defines the root path route ("/")
  # root "posts#index"
  root 'application#hola'
end
```

4.2. Guardar los cambios realizados e iniciar el servidor en el terminal.

```
root@debian:/home/gia/Proyectos_RoR/first_app/config# rails s
=> Booting Puma
=> Rails 7.1.3.4 application starting in development
=> Run `bin/rails server --help` for more startup options
Puma starting in single mode...
* Puma version: 6.4.2 (ruby 3.1.2-p20) ("The Eagle of Durango")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 3861
* Listening on http://127.0.0.1:3000
* Listening on http://[::]:3000
Use Ctrl-C to stop
Started GET "/" for 127.0.0.1 at 2024-08-28 11:13:43 -0600
  ActiveRecord::SchemaMigration Load (0.2ms)  SELECT "schema_migrations"
M "schema_migrations" ORDER BY "schema_migrations"."version" ASC
Processing by ApplicationController#hola as HTML
  Rendering html template
  Rendered html template (Duration: 0.0ms | Allocations: 11)
Completed 200 OK in 31ms (Views: 18.2ms | ActiveRecord: 0.0ms | A
```

4.3. Abrir el navegador en la dirección 127.0.0.1:3000 y verificar que ahora, la página principal presenta la acción hola creada anteriormente.



hola, mundo!

2. Crear una acción nueva que muestre un texto cualquiera.

```
GNU nano 7.2 app/controllers/application_control
class ApplicationController < ActionController::Base
  protect_from_forgery with: :exception
  def algo
    render html: "hola, soy ally!"
  end
end
```

3. Editar el archivo routes.rb, para que la aplicación siempre inicie, por defecto, con la nueva acción creada en el inciso anterior.

```
GNU nano 7.2 routes.rb
Rails.application.routes.draw do
  # Define your application routes per the DSL in https://gui

  # Reveal health status on /up that returns 200 if the app b
  # Can be used by load balancers and uptime monitors to veri
  get "up" => "rails/health#show", as: :rails_health_check

  # Defines the root path route ("/")
  # root "posts#index"
  root 'application#algo'
end
```



hola, soy ally!