

UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA

UNAN-León.



Facultad de Ciencias y Tecnología

Carrera: Ingeniería en Telemática.

Componente: Software como un Servicio.

Grupo: #2.

Docente: Ervin Montes.

Nombre:

- Rosa Arellys Orozco Lira.

Fecha: 25/08/2024.

¡A la libertad por la universidad!

1. Creación de un proyecto nuevo.

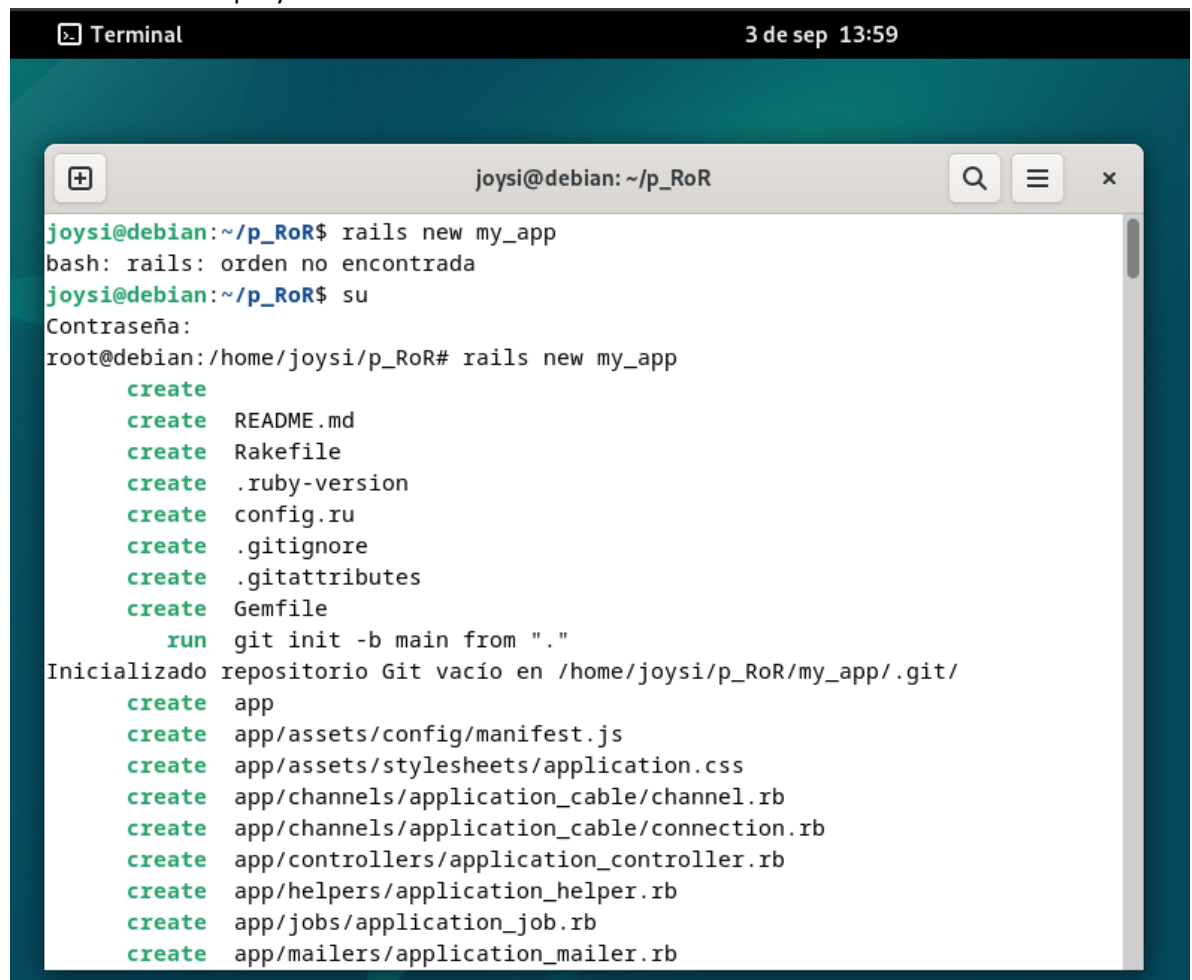
Ubicarse en el directorio donde se va almacenar el proyecto.



A terminal window titled 'Terminal' with a timestamp of '3 de sep 13:53'. The window shows a user 'joysi@debian' in the home directory '~'. The user runs 'mkdir p_RoR' and then 'ls'. The 'ls' command outputs a list of files and directories: 'array.rb', 'Escritorio', 'mi_proyecto', 'practica3', 'Público', 'trabajoGit', 'Descargas', 'first_app', 'Música', 'p_RoR', 'trabajo2', 'TrabajoRuby2', 'Documentos', 'Imágenes', 'Plantillas', 'proyectos_RoR', 'Trabajo2Ruby', and 'Videos'. The prompt returns to 'joysi@debian:~\$'.

```
joysi@debian:~$ mkdir p_RoR
joysi@debian:~$ ls
array.rb  Escritorio  mi_proyecto  practica3    Público    trabajoGit
Descargas first_app   Música       p_RoR        trabajo2   TrabajoRuby2
Documentos Imágenes   Plantillas  proyectos_RoR Trabajo2Ruby Videos
joysi@debian:~$
```

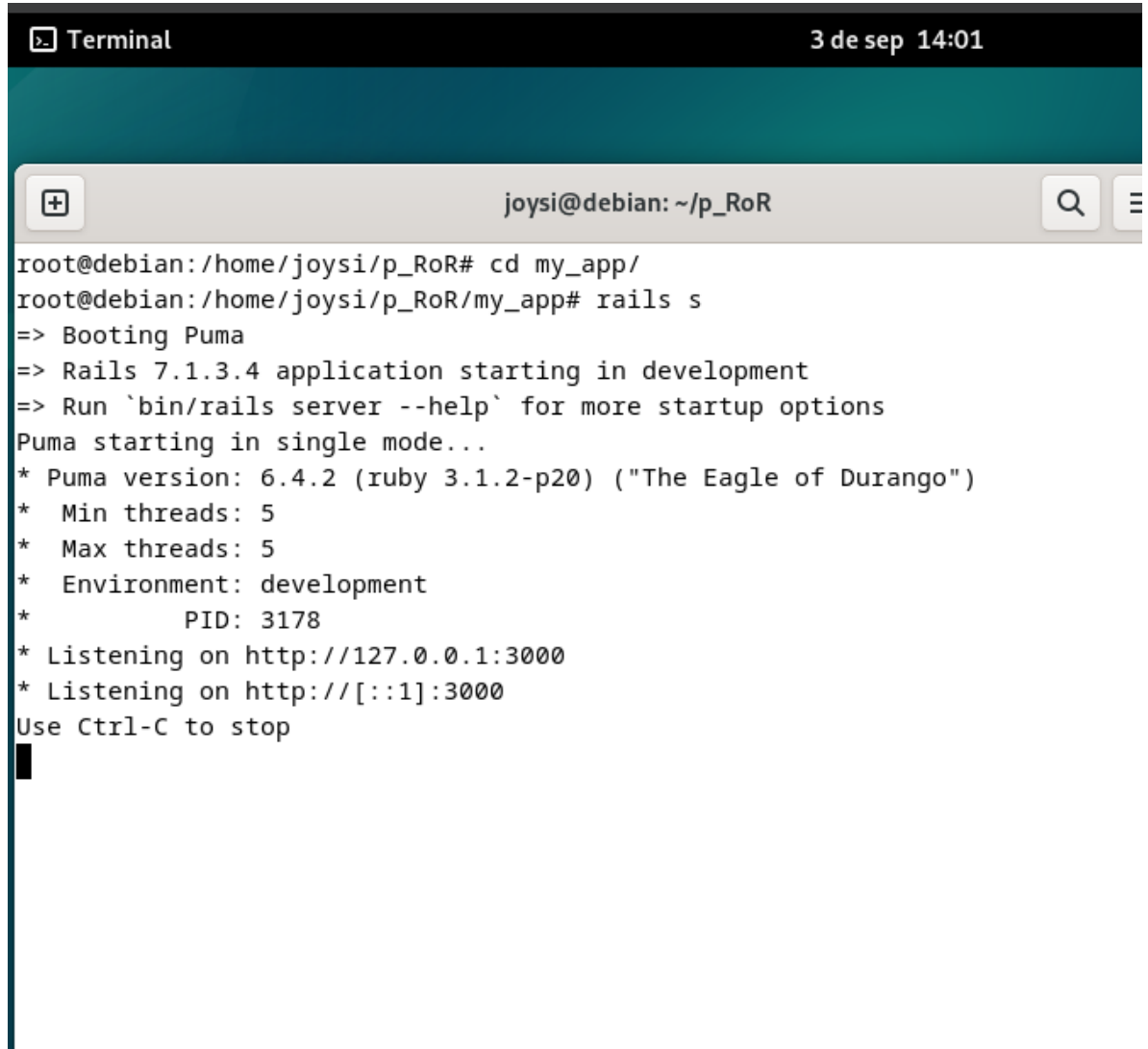
2. Generar un nuevo proyecto.



A terminal window titled 'Terminal' with a timestamp of '3 de sep 13:59'. The window shows a user 'joysi@debian' in the directory '~/p_RoR'. The user runs 'rails new my_app', which fails with the message 'bash: rails: orden no encontrada'. The user then runs 'su' to become root. As root, the user runs 'rails new my_app'. The output shows the creation of several files and directories: '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/joysi/p_RoR/my_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', and 'create app/mailers/application_mailer.rb'.

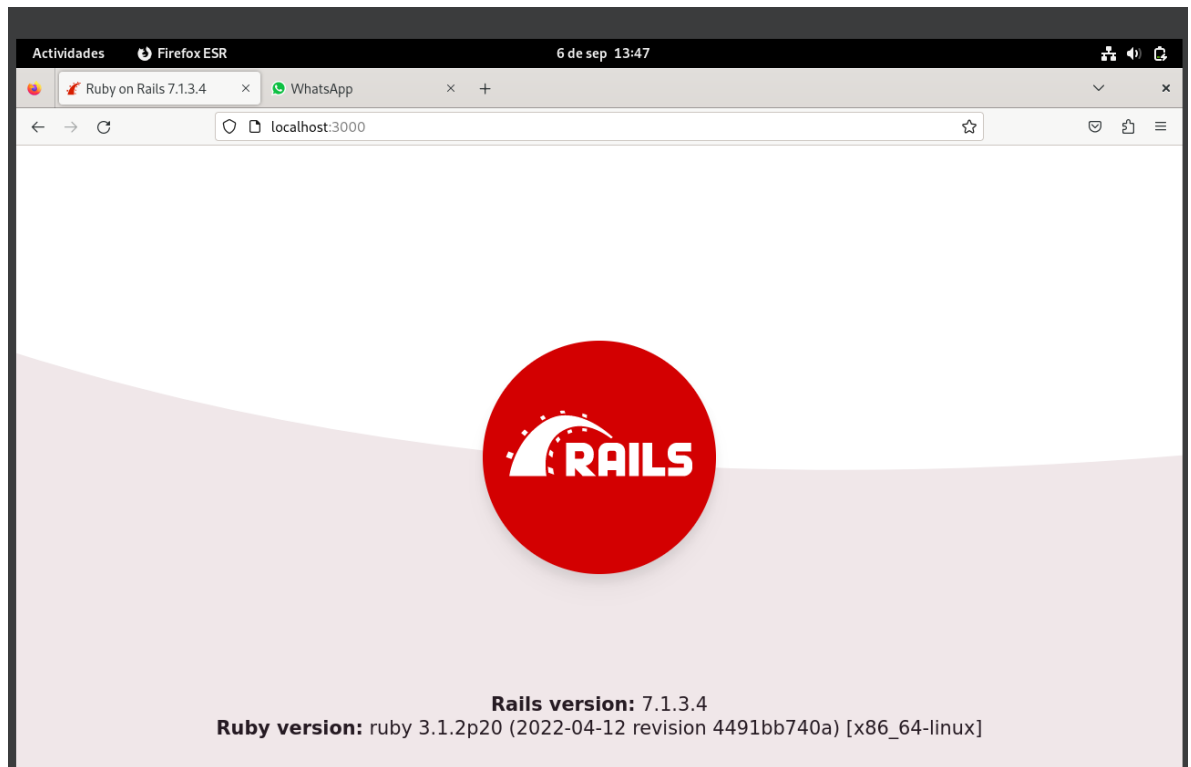
```
joysi@debian:~/p_RoR$ rails new my_app
bash: rails: orden no encontrada
joysi@debian:~/p_RoR$ su
Contraseña:
root@debian:/home/joysi/p_RoR# rails new my_app
create
create  README.md
create  Rakefile
create  .ruby-version
create  config.ru
create  .gitignore
create  .gitattributes
create  Gemfile
run     git init -b main from "."
Inicializado repositorio Git vacío en /home/joysi/p_RoR/my_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
```

3. Ubicarse dentro del proyecto e iniciar el servidor para comprobar que funciona sin ningún problema.

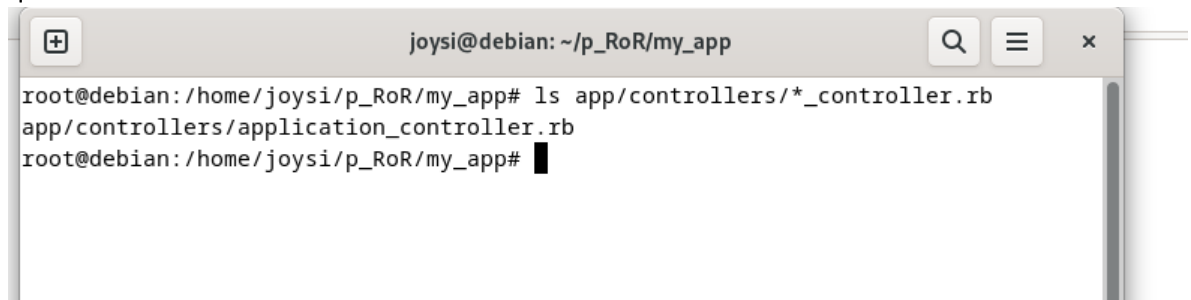
A terminal window titled "Terminal" with a timestamp "3 de sep 14:01". The window shows a user "joysi@debian" in the directory "~/p_RoR". The user navigates to the "my_app" directory and runs "rails s". The output shows the Rails application starting in development mode, booting Puma, and listening on http://127.0.0.1:3000 and http://[::1]:3000. The user is prompted to use Ctrl-C to stop the server.

```
Terminal 3 de sep 14:01
joysi@debian: ~/p_RoR
root@debian:/home/joysi/p_RoR# cd my_app/
root@debian:/home/joysi/p_RoR/my_app# 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: 3178
* Listening on http://127.0.0.1:3000
* Listening on http://[::1]:3000
Use Ctrl-C to stop
█
```

4. Abrir el navegador y escribir la siguiente dirección.



5. Generar un nuevo controlador al proyecto. Verificar cuál es el controlador existente dentro del proyecto. De la siguiente forma se obtendrá el nombre de los controladores que existen hasta el momento



6. Generar el nuevo controlador y una acción correspondiente a ese controlador.

```
joysi@debian: ~/p_RoR/my_app
root@debian:/home/joysi/p_RoR/my_app# rails generate controller welcome index
   create  app/controllers/welcome_controller.rb
   route   get 'welcome/index'
  invoke   erb
   create  app/views/welcome
   create  app/views/welcome/index.html.erb
  invoke   test_unit
   create  test/controllers/welcome_controller_test.rb
  invoke   helper
   create  app/helpers/welcome_helper.rb
  invoke   test_unit
root@debian:/home/joysi/p_RoR/my_app#
```

7. Verificar nuevamente los controladores del proyecto. Podrá observar que ahora ya existe uno nuevo, y que se corresponde con el creado anteriormente.

```
joysi@debian: ~/p_RoR/my_app
root@debian:/home/joysi/p_RoR/my_app# ls app/controllers/*_controller.rb
app/controllers/application_controller.rb
app/controllers/welcome_controller.rb
root@debian:/home/joysi/p_RoR/my_app#
```

8. Abrir el archivo `welcome_controller.rb` que se encuentra en el directorio (`app/controllers/`) y verificar que existe el método `index`, que fue creado al momento de generar el controlador, este es el método (acción) que tiene relación con la vista `index.html.erb`.

```
6 de sep 13:52
welcome_controller.rb - proyectos_RoR - Visual Studio Code

Terminal Ayuda
application_controller.rb welcome_controller.rb x routes.rb
home > joysi > p_RoR > my_app > app > controllers > welcome_controller.rb
1 class WelcomeController < ApplicationController
2   def index
3   end
4 end
5
```

9. Trabajar con la vista. Para comenzar a trabajar con las vistas se debe ubicar el directorio (`app/view/`), que es donde se almacenan las vistas del proyecto, ahí se observa un archivo

llamado index.html.erb, este es por medio del cual los usuarios finales interactúan con la aplicación.

Editar el archivo index.html.erb del nuevo controlador y escribir el siguiente código html.

```
welcome_controller.rb  <> index.html.erb x
```

```
home > joysi > p_RoR > my_app > app > views > welcome > <> index.html.erb
1  <h1>Bienvenido</h1>
2  <p>Pagina de inicio</p>
3  <p>UNAN-Leon</p>
4
```

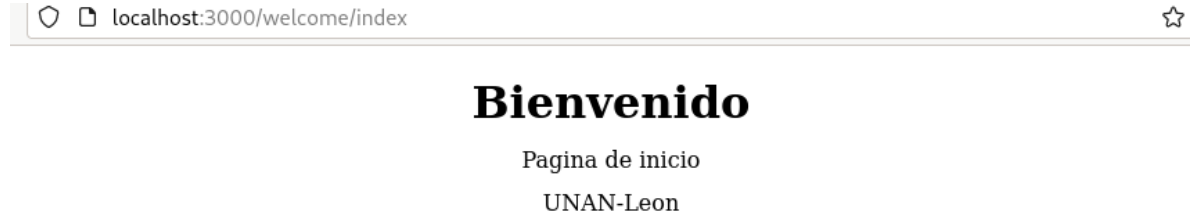
10. . Editar el archivo welcome.scss, que se encuentra en el directorio (app/assets/stylesheets/), en este directorio es donde se almacenan todas las hojas de estilos que contiene el proyecto. Todos los archivos de estilos en rails terminan en “.scss” debido a que el framework permite utilizar el lenguaje de estilos llamado Sass (Syntactically Awesome StyleSheets), es una extensión de css que agrega potencia y elegancia al lenguaje básico. Para darle un poco de formato a las vistas, agregar el siguiente código.

```
Terminal Ayuda
```

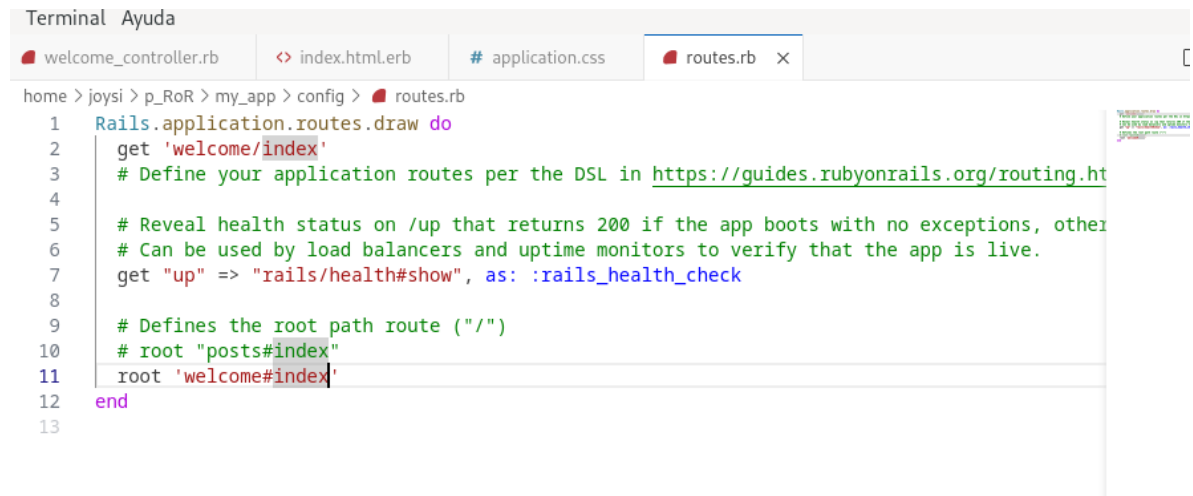
```
welcome_controller.rb  <> index.html.erb  # application.css x
```

```
home > joysi > p_RoR > my_app > app > assets > stylesheets > # application.css > p
1  h1
2  {
3      color: black;
4      height: 30px;
5      text-align: center;
6  }
7  p
8  {
9      height: 15px;
10     text-align: center;
11 }
```

11. Ir a la dirección (<http://localhost:3000/welcome/index>), podrá obtener una vista como la de la figura 52.



12. Si navega directamente en <http://localhost:3000/> no se tendrá acceso a la vista antes modificada, para esto, se debe modificar el archivo routes.rb, de manera que inicie (root) con la vista de nombre index.html.erb.



13. Actualizar el navegador en la dirección <http://localhost:3000/> Al momento de actualizar el navegador ya se ha obtenido el acceso a la vista en la cual se ha estado trabajando a como se puede observar en la figura , esto es debido a que se ha especificado la ruta del controlador, como la ruta raíz o principal del proyecto.



14. Para ver el funcionamiento ir al archivo `welcome_controller.rb` y crear un array global con los días de la semana dentro del método `index`.

```
welcome_controller.rb x index.html.erb # application.css routes.rb
home > joysi > p_RoR > my_app > app > controllers > welcome_controller.rb
1 class WelcomeController < ApplicationController
2   def index
3     @dias_semana=
4     ["Lunes", "Martes", "Miercoles", "Jueves", "Viernes", "Sabado", "Domingo"]
5   end
6 end
7
```

15. Para poder hacer uso del array dentro de la vista, se debe utilizar lenguaje ERB en el archivo de la vista `index.html.erb`; editar el archivo y escribir el siguiente.

```
welcome_controller.rb x index.html.erb # application.css routes.rb
home > joysi > p_RoR > my_app > app > views > welcome > index.html.erb
1 <h1>Bienvenido</h1>
2 <p>Pagina de inicio</p>
3 <p>UNAN-Leon</p>
4
5 <h3>Imprimiendo un array Ruby dentro de una vista</h3>
6 <% @dias_semana.each do |dia| %>
7   <%= dia %> <br>
8 <% end %>
9
```

Bienvenido
Pagina de inicio
UNAN-Leon

Imprimiendo un array Ruby dentro de una vista

Lunes
Martes
Miercoles
Jueves
Viernes
Sabado
Domingo

16. De igual forma se puede hacer uso de los distintos métodos propios de los array, en el caso anterior se utilizó el método each para recorrer el array. Escribir el siguiente código en el archivo index.html.erb.

```
welcome_controller.rb  index.html.erb x  application.css  routes.rb

home > joysi > p_RoR > my_app > app > views > welcome > <> index.html.erb
1  <h1>Bienvenido</h1>
2  <p>Pagina de inicio</p>
3  <p>UNAN-Leon</p>
4
5  <h3>Imprimiendo un array Ruby dentro de una vista</h3>
6  <%= @dias_semana.each do |dia| %>
7  <%= dia %> <br>
8  <%= end %>
9
10 <h3> Método last</h3>
11 <%= @dias_semana.last %>
12
```

Bienvenido

Pagina de inicio

UNAN-Leon

Imprimiendo un array Ruby dentro de una vista

Lunes
Martes
Miercoles
Jueves
Viernes
Sabado
Domingo

Método last

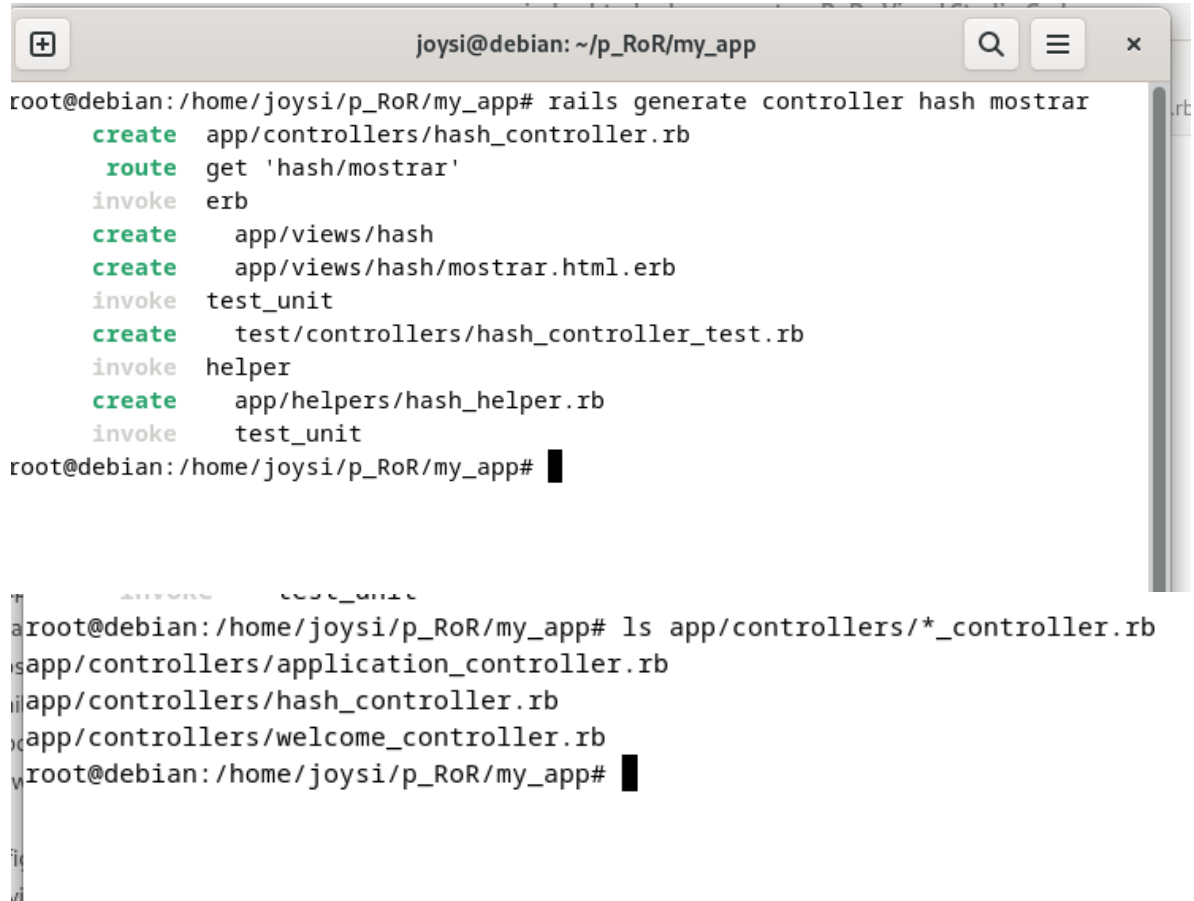
Domingo

17. Muestre en la vista index.html.erb el primer dato almacenado y el tamaño completo del array, haga uso de métodos similares a los vistos anteriormente.

Primer campo del array:Lunes

Tamaño del array:7

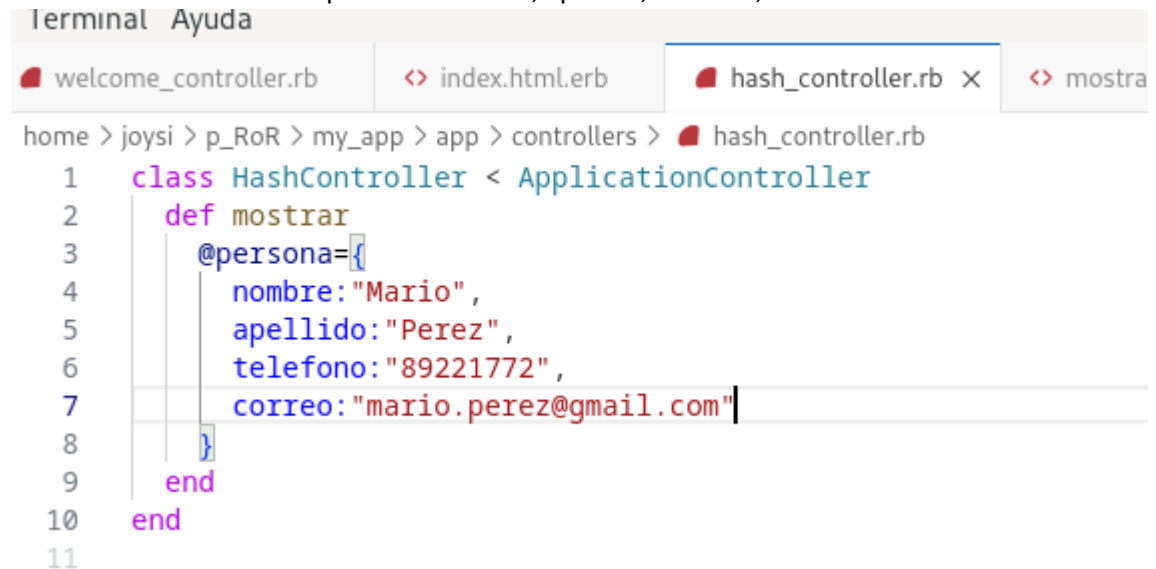
18. Generar un nuevo controlador de nombre “hash” y una acción que es la que tendrá relación con la vista llamada “mostrar”, similar al realizado al inicio de esta guía.



```
joysi@debian: ~/p_RoR/my_app
root@debian:/home/joysi/p_RoR/my_app# rails generate controller hash mostrar
   create  app/controllers/hash_controller.rb
   route   get 'hash/mostrar'
   invoke  erb
   create  app/views/hash
   create  app/views/hash/mostrar.html.erb
   invoke  test_unit
   create  test/controllers/hash_controller_test.rb
   invoke  helper
   create  app/helpers/hash_helper.rb
   invoke  test_unit
root@debian:/home/joysi/p_RoR/my_app#
```

```
root@debian:/home/joysi/p_RoR/my_app# ls app/controllers/*_controller.rb
app/controllers/application_controller.rb
app/controllers/hash_controller.rb
app/controllers/welcome_controller.rb
root@debian:/home/joysi/p_RoR/my_app#
```

19. Dentro de la acción mostrar que se creó junto al nuevo controlador del proyecto, crear un hash con los datos de una persona: nombre, apellido, teléfono, correo.



```
Terminal Ayuda
welcome_controller.rb  index.html.erb  hash_controller.rb x  mostra

home > joysi > p_RoR > my_app > app > controllers > hash_controller.rb
1  class HashController < ApplicationController
2      def mostrar
3          @persona={
4              nombre:"Mario",
5              apellido:"Perez",
6              telefono:"89221772",
7              correo:"mario.perez@gmail.com"
8          }
9      end
10 end
11
```

20. Modificar la vista del controlador creado anteriormente y mostrar los datos del hash en una tabla html.

lcome_controller.rb <> index.html.erb hash_controller.rb <> mostrar.htn

```
home > joysi > p_RoR > my_app > app > views > hash > <> mostrar.html.erb
1  <h1>Datos de la persona</h1>
2  <table border="1" cellpadding="10" cellspacing="0">
3    <tr>
4      <th>Campo</th>
5      <th>Valor</th>
6    </tr>
7    <tr>
8      <td><strong>Nombre</strong></td>
9      <td><%= @persona[:nombre] %></td>
10   </tr>
11   <tr>
12     <td><strong>Apellido</strong></td>
13     <td><%= @persona[:apellido] %></td>
14   </tr>
15   <tr>
16     <td><strong>Teléfono</strong></td>
17     <td><%= @persona[:telefono] %></td>
18   </tr>
19   <tr>
20     <td><strong>Correo</strong></td>
21     <td><%= @persona[:correo] %></td>
22   </tr>
23 </table>
```

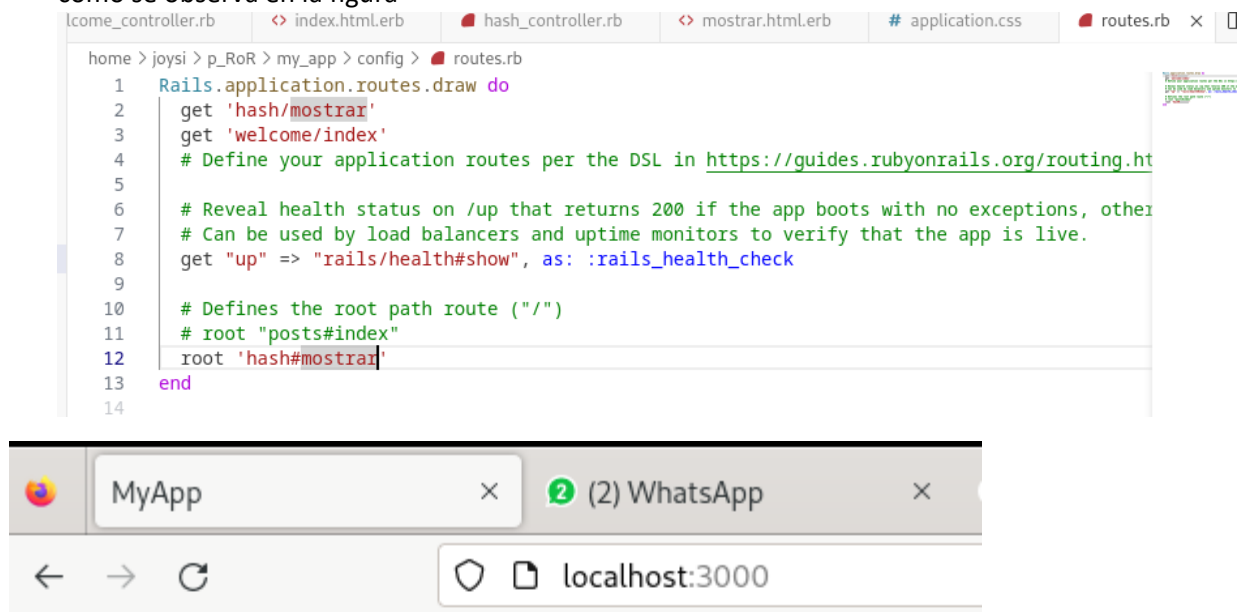
21. Agregar código css a la tabla, se mostrará en la vista como en la figura que se puede ver que la tabla contiene formato, colores, bordes.

```
lcome_controller.rb  <> index.html.erb  hash_controller.rb  <> mostrar.html.erb

home > joyasi > p_RoR > my_app > app > assets > stylesheets > # application.css > td

1  table {
2      width: auto;
3      border-collapse: collapse;
4      font-family: Arial, sans-serif;
5      margin-top: 20px;
6  }
7
8  th {
9      background-color: #07539e; /* Azul oscuro */
10     color: white;
11     padding: 10px;
12     text-align: left;
13     border: 1px solid black;
14 }
15
16 td {
17     background-color: #85cbf3; /* Amarillo claro */
18     padding: 10px;
19     border: 1px solid black;
20     text-align: left;
21 }
22
23 h1 {
24     font-size: 24px;
25 }
26
27 h3 {
28     font-size: 18px;
```

22. Configurar el archivo routes.rb para cambiar la página de inicio del proyecto y ahora hacerlo apuntar al nuevo controlador junto con su respectiva vista, la cual deberá quedar como se observa en la figura



Datos de la persona

Campo	Valor
Nombre	Mario
Apellido	Perez
Teléfono	89221772
Correo	mario.perez@gmail.com