

**UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA
(UNAN-LEÓN)**

FACULTAD DE CIENCIAS Y TECNOLOGÍAS.

INGENIERIA EN TELEMATICA.



COMPONENTE: SaaS.

Grupo 2 Telemática

INTEGRANTES:

➤ Cecilia Elohi Escoto Sandoval.

DOCENTE: Ervin Montes.

**¡A LA LIBERTAD POR LA UNIVERSIDAD!
“AÑO LECTIVO 2024”**

1.

```
ceci@debian12:~$ rails new Instagram
  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/ceci/Instagram/.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
```

Proyecto 2 Proyecto 1.pdf Proyecto1_SAAS - Documen Ruby on Rails / 7.2.1.2

localhost:3000



Rails version: 7.2.1.2
Rack version: 3.1.8
Ruby version: ruby 3.1.2p20 (2022-04-12 revision 4491bb740a) [x86_64-linux]

2.

```
ceci@debian12:~/Instagram$ nano Gemfile
ceci@debian12:~/Instagram$ bundle install
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies...
Bundle complete! 18 Gemfile dependencies, 104 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
ceci@debian12:~/Instagram$
```

3.

```
ceci@debian12:~/Instagram$ rails generate devise:install
  create  config/initializers/devise.rb
  create  config/locales/devise.en.yml

=====

Depending on your application's configuration some manual setup may be required:

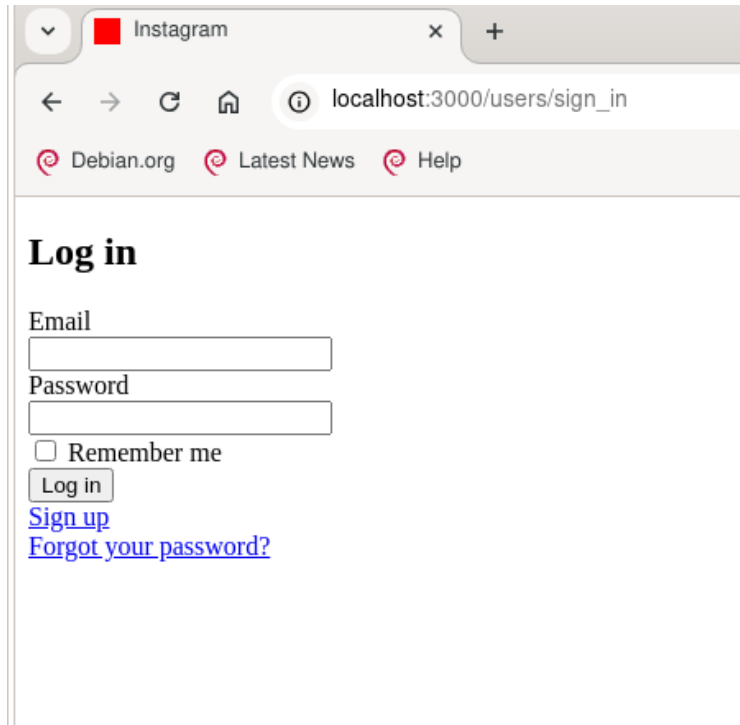
  1. Ensure you have defined default url options in your environments files. Here
  is an example of default_url_options appropriate for a development environment
  in config/environments/development.rb:

      config.action_mailer.default_url_options = { host: 'localhost', port: 3000 }

  In production, :host should be set to the actual host of your application.
```

```
ceci@debian12:~/Instagram$ rails generate devise User
  invoke  active_record
  create  db/migrate/20241023231107_devise_create_users.rb
  create  app/models/user.rb
  invoke  test_unit
  create  test/models/user_test.rb
  create  test/fixtures/users.yml
  insert  app/models/user.rb
  route   devise_for :users
```

```
ceci@debian12:~/Instagram$ rake db:migrate
== 20241023231107 DeviseCreateUsers: migrating =====
-- create_table(:users)
   -> 0.0050s
-- add_index(:users, :email, {:unique=>true})
   -> 0.0155s
-- add_index(:users, :reset_password_token, {:unique=>true})
   -> 0.0016s
== 20241023231107 DeviseCreateUsers: migrated (0.0288s) =====
```



4.

```
ceci@debian12:~/Instagram$ nano Gemfile
ceci@debian12:~/Instagram$ bundle install
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies...
Fetching execjs 2.10.0
Installing execjs 2.10.0
Bundle complete! 19 Gemfile dependencies, 112 gems now installed.
Use `bundle info [gemname]` to see where a bundled gem is installed.
```

```
app > assets > stylesheets > application.scss
1  /*
2   * This is a manifest file that'll be compiled into application.css, which is placed in the public
3   * listed below.
4   *
5   * Any CSS (and SCSS, if configured) file within this directory, and any subdirectory thereof, will
6   * vendor/assets/stylesheets directory can be compiled into this file.
7   *
8   * You're free to add application-wide styles to this file, but it's generally better to create a new
9   * compiled file so the styles you add here take precedence over those from default files.
10  * files in this directory. Styles in this file will be a lower priority than the styles defined in
11  * It is generally better to create a new file for each component or group of components.
12  *
13  *= require_tree .
14  *= require_self
15  */
16  @import "bootstrap-sprockets";
17  @import "bootstrap";
```

```

app > javascript > js application.js
1 // Configure your import map in config/importmap.rb
2 import "@hotwired/turbo-rails"
3 import "controllers"
4 //require bootstrap-sprockets

```

5.

```

app > views > layouts > <> application.html.erb
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <title>Instagram</title>
5   <%= stylesheet_link_tag "application", media: "all", "data-turbolinks-track"
6   <%= javascript_include_tag "application", "data-turbolinks-track" => true %>
7   <%= csrf_meta_tags %>
8 </head>
9 <body>
10   <!-- Navigation -->
11   <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
12     <div class="container">
13       <!-- Brand and toggle get grouped for better mobile display -->
14       <div class="navbar-header">
15         <button type="button" class="navbar-toggle" data-toggle="collapse"
16         target="#bs-example-navbar-collapse-1">
17           <span class="sr-only">Toggle navigation</span>
18           <span class="icon-bar"></span>
19           <span class="icon-bar"></span>
20           <span class="icon-bar"></span>
21         </button>
22       </div>
23       <!-- Collect the nav links, forms, and other content for toggling -->
24       <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
25         <ul class="nav navbar-nav">
26           <li><a href="#">About</a></li>
27           <li><a href="#">Services</a></li>
28           <li><a href="#">Contact</a></li>
29         </ul>
30       </div>

```

```

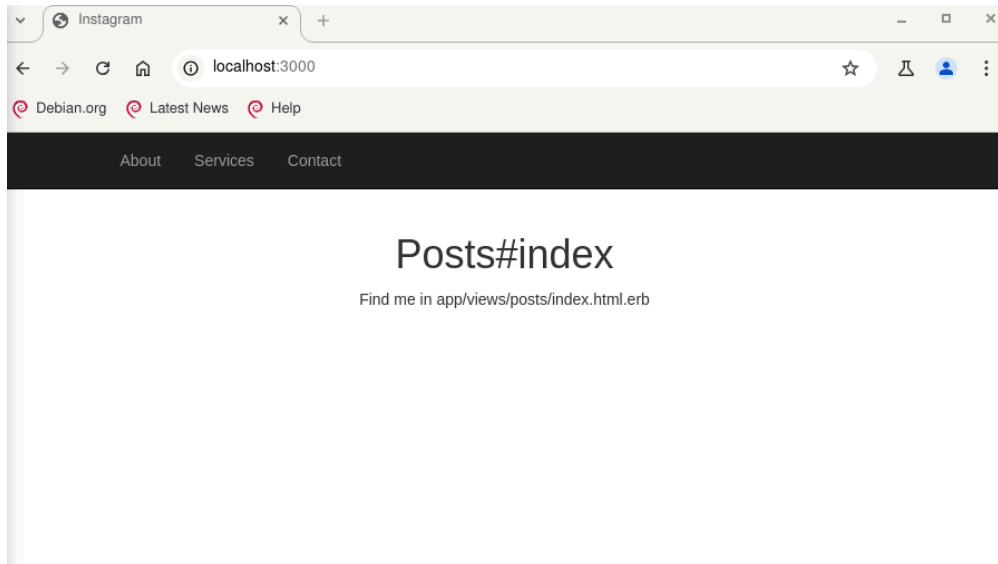
ceci@debian12:~/Instagram$ rails generate model Post description:text
  invoke  active_record
  create  db/migrate/20241024000531_create_posts.rb
  create  app/models/post.rb
  invoke  test_unit
  create  test/models/post_test.rb
  create  test/fixtures/posts.yml

```

```

ceci@debian12:~/Instagram$ rails generate controller Posts new index show
  create  app/controllers/posts_controller.rb
  route   get "posts/new"
  route   get "posts/index"
  route   get "posts/show"
  invoke  erb
  create  app/views/posts
  create  app/views/posts/new.html.erb
  create  app/views/posts/index.html.erb
  create  app/views/posts/show.html.erb
  invoke  test_unit
  create  test/controllers/posts_controller_test.rb
  invoke  helper
  create  app/helpers/posts_helper.rb
  invoke  test_unit

```



6. `ceci@debian12:~/Instagram$ rails routes`

Prefix	Verb	URI Pattern	Controller#Action
posts_new	GET	/posts/new(.:format)	posts#new
posts_index	GET	/posts/index(.:format)	posts#index
posts_show	GET	/posts/show(.:format)	posts#show
root	GET	/	posts#index
posts	GET	/posts(.:format)	posts#index
	POST	/posts(.:format)	posts#create
new_post	GET	/posts/new(.:format)	posts#new
edit_post	GET	/posts/:id/edit(.:format)	posts#edit
post	GET	/posts/:id(.:format)	posts#show
	PATCH	/posts/:id(.:format)	posts#update
	PUT	/posts/:id(.:format)	

7.

```
llect the nav links, forms, and other content for toggling -->
<div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
  <ul class="nav navbar-nav">
    <li><%= link_to "New Post", new_post_path %></li>
    <%- if current_user %>
    <li>
      <%= link_to "Logout", destroy_user_session_path, method: :delete,
        data: { confirm: "Are you sure?" } %>
    </li>
    <% else %>
    <li><%= link_to "Login", new_user_session_path %></li>
    <li><%= link_to "Register", new_user_registration_path %></li>
    <% end %>
  </ul>
</div>
```

localhost:3000/posts/new

Debian.org Latest News Help

New Post Login Register

Posts#new

Find me in app/views/posts/new.html.erb

New Post Login Register

Log in

Email

Password

☐ Remember me

Log in

New Post Login Register

Sign up

Email

Password (6 characters minimum)

Password confirmation

Sign up

[Log in](#)

8.

```
class PostsController < ApplicationController
  GET /posts/new(.:format) | Jump to view

  def new
    @post = Post.new
  end
end
```

[Instagram](#)
[New Post](#)
[Login](#)
[Register](#)

New Post

Description

9.

[Instagram](#)
[New Post](#)
[Logout](#)

Welcome! You have signed up successfully.

10.

```
<div class="container">
  <div class="row">
    <% flash.each do |a,b| %>
      <div class="alert alert-success"><%= b%> </div>
    <%end%>
    <div class="col-lg-12 text-center">
      <%= yield %>
    </div>
  </div>
<!-- /.row -->
</div>
```

11.

```
ceci@debian12:~/Instagram$ bundle install
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies...
Fetching multi_json 1.15.0
Fetching jmespath 0.5.1
Installing multi_json 1.15.0
Fetching climate_control 0.2.0
Installing jmespath 0.5.1
Fetching multi_xml 0.7.1
Installing climate_control 0.2.0
Fetching mimemagic 0.3.10
Installing multi_xml 0.7.1
Fetching cocaine 0.5.8
```


12.

```
ceci@debian12:~/Instagram$ rails g paperclip post image
      create db/migrate/20241024022729_add_attachment_image_to_posts.rb
```

13.

```
class AddAttachmentImageToPosts < ActiveRecord::Migration[5.1]
  def self.up
    change_table :posts do |t|
      t.attachment :image
    end
  end

  def self.down
    remove_attachment :posts, :image
  end
end
```

```
ceci@debian12:~/Instagram$ rake db:migrate
== 20241024022729 AddAttachmentImageToPosts: migrating =====
-- change_table(:posts)
rake aborted!
StandardError: An error has occurred, this and all later migrations canceled:

wrong number of arguments (given 3, expected 2)
/home/ceci/Instagram/db/migrate/20241024022729_add_attachment_image_to_posts.rb:4:in `block in up'
/home/ceci/Instagram/db/migrate/20241024022729_add_attachment_image_to_posts.rb:3:in `up'

Caused by:
ArgumentError: wrong number of arguments (given 3, expected 2)
/home/ceci/Instagram/db/migrate/20241024022729_add_attachment_image_to_posts.rb:4:in `block in up'
/home/ceci/Instagram/db/migrate/20241024022729_add_attachment_image_to_posts.rb:3:in `up'
Tasks: TOP => db:migrate
See full trace by running task with --trace
```

14.

```
class Post < ApplicationRecord
  has_attached_file :image, styles: {large: "600x600>", medium: "300x300>",
    thumb: "150x150>" }, default_url: "/images/:style/missing.png"
  validates_attachment_content_type :image, content_type: /\Aimage\/.*\z/
end
```

15.

```
<h2>New Post</h2>
<hr>
<%= form_for @post, html: { multipart: true } do |f| %>
  <%= f.label :image %>
  <%= f.file_field :image%>
  <br>
  <%= f.label :description %>
  <%= f.text_area :description %>
  <br>
  <%= f.submit %>
<%end%>
```

16.

```
app > views > posts > <> show.html.erb
1  <%= image_tag(@post.image.url(:large)) %>
2  <br>
3  <%= @post.description %>
4  <div class="link">
5    <%= link_to "Destroy", @post , method: :delete,confirm:"Are you sure?" %>
6    <%= link_to "Edit",edit_post_path(@post) %>
7  </div>
```

17.

```
app > views > posts > <> index.html.erb
1  <% @posts.each do |x| %>
2  <div>
3    <%= image_tag x.image.url(:medium) %>
4    <br>
5    <%= x.description %>
6    <div class="link">
7      <%= link_to "Destroy", x , method: :delete,confirm:"Are you sure?" %>
8      <%= link_to "Edit", edit_post_path(x) %>
9      <%= link_to "Show", x %>
10   </div>
11 </div>
12 <% end %>
```

El uso de paperclip esta discontinuado y desde rails 6 se usa Active record que es nativo de Rails tuve que usar active record para avanzar la guia, al igual que el destroy y el path logout deben ser botones o requieren mas bibliotecas de js como query yo decidi hacerlos botones


Instagram

localhost:3000

Debian.orgLatest NewsHelp

INSTAGRAMNew PostLogout


All Posts



Kurosaki-Kun~

Destroy

Edit Show



Kuchiki Rukia

Destroy

Instagram

localhost:3000/users/sign_in

Debian.orgLatest NewsHelp

INSTAGRAMNew PostLoginRegister

You need to sign in or sign up before continuing.

Log in

Email

Password

☐ Remember me

Log in

[Sign up](#)

[Forgot your password?](#)

["Image can't be blank", "Description can't be blank", "Description is too short (minimum is 10 characters)"]

New Post

Image

Seleccionar archivo Sin archivos seleccionados

Description

Create Post

```
class PostsController < ApplicationController
  before_action :authenticate_user!
  before_action :set_post, only: [:show, :edit, :update, :destroy]
  before_action :authorize_user!, only: [:edit, :destroy]

  # GET /posts
```

```

def create
  @post = Post.new(post_params)
  @post.user = current_user # Asignar el usuario actual al post
  if @post.save
    flash[:success] = "Success!"
    redirect_to post_path(@post)
  else
    flash[:error] = @post.errors.full_messages.join(", ")
    redirect_to new_post_path
  end
end

```

```

def set_post
  @post = Post.find(params[:id])
end

def authorize_user!
  unless @post.user_id == current_user.id
    flash[:alert] = "No tienes permiso para realizar esta acción."
    redirect_to posts_path
  end
end

```

[INSTAGRAM](#)
[New Post](#)
[Logout](#)

No tienes permiso para realizar esta acción.

All Posts



El post fue actualizado exitosamente.

Post



Usooooo~

Destroy

[Edit](#)