

Práctica Calificada 2

PARTE 3 RAILS

CC3S2

Cesar Jesus Lara Avila

Calagua Mallqui Jairo Andre

20210279F

Octubre del 2023

Procedemos a verificar si tenemos instaladas las versiones correctas de Ruby, Rails y Render para mi caso, ya que fue la herramienta utilizada para mi primera calificada, entonces:

Ejecutamos **ruby -v** para verificar nuestra versión de Ruby:

```
PS C:\Users\calag> ruby -v
ruby 3.0.6p216 (2023-03-30 revision 23a532679b) [x64-mingw32]
```

La versión de la que dispongo es la 3.0.6.

Ejecutamos **Rails -v** para comprobar la versión de Rails:

```
PS C:\Users\calag> rails -v
Rails 7.0.8
PS C:\Users\calag>
```

La versión de la que dispongo es la 7.0.8.

Ahora, ejecutamos **bundle -v** para comprobar la versión de nuestro Bundler:

```
PS C:\Users\calag> bundle -v
Bundler version 2.4.20
PS C:\Users\calag>
```

La versión de la que dispongo es la 2.4.20

No instalaré la versión que se me sugiere, porque no quiero cometer el error de que algo falle; sin embargo, puedo mostrar el proceso en el caso hipotético en el que si lo hubiera hecho:

```
PS C:\Users\calag> gem install bundler --version=1.17.3 --no-document
Successfully installed bundler-1.17.3
1 gem installed
PS C:\Users\calag> |
```

Ya que tengo más de una versión del bundle instalada, ejecutamos el comando **gem uninstall bundler**.

```
PS C:\Users\calag> gem uninstall bundler
Gem bundler-2.4.20 cannot be uninstalled because it is a default gem

Select gem to uninstall:
 1. bundler-1.17.3
 2. bundler-2.4.20
 3. All versions
> 2
Successfully uninstalled bundler-2.4.20
There was both a regular copy and a default copy of bundler-2.4.20. The regular copy was successfully uninstalled, but the default copy was left around because default gems can't be removed.
PS C:\Users\calag>
```

Ahora, ya que no usaremos Heroku (y en su lugar usaremos Render), no hace falta verificar versión alguna por medio de comandos, pero si podemos ver un VS realizado por el mismo Render:

Render vs Heroku

We've built Render to help developers and businesses avoid the cost and inflexibility traps of legacy Platform-as-a-Service solutions like Heroku. Our customers often tell us Render is what Heroku *could* have been. This page explains why so many former Heroku customers consider Render to be the best Heroku alternative.

Este VS contiene mucha más información detallada que se puede encontrar en el siguiente URL:

<https://render.com/render-vs-heroku-comparison>

Creación de una nueva aplicación Rails

Utilizaremos el comando **rails new rottenpotatoes --skip-test-unit --skip-turbolinks --skip-spring** para crear una nueva aplicación Rails vacía:

```
PS C:\Users\calag> rails new rottenpotatoes --skip-test-unit --skip-turbolinks --skip-spring
```

```
identical app/assets/config/manifest.js
identical app/assets/stylesheets/application.css
identical app/channels/application_cable/channel.rb
identical app/channels/application_cable/connection.rb
identical app/controllers/application_controller.rb
identical app/helpers/application_helper.rb
identical app/jobs/application_job.rb
identical app/mailers/application_mailer.rb
identical app/models/application_record.rb
identical app/views/layouts/application.html.erb
identical app/views/layouts/mailer.html.erb
identical app/views/layouts/mailer.text.erb
exist     app/assets/images
identical app/assets/images/.keep
identical app/controllers/concerns/.keep
identical app/models/concerns/.keep
exist     bin
identical bin/rails
identical bin/rake
identical bin/setup
exist     config
identical config/routes.rb
identical config/application.rb
identical config/environment.rb
identical config/cable.yml
identical config/puma.rb
identical config/storage.yml
```

```

Import Turbo
  append app/javascript/application.js
Pin Turbo
  append config/importmap.rb
Run turbo:install:redis to switch on Redis and use it in development for t
urbo streams
Create controllers directory
  create app/javascript/controllers
  create app/javascript/controllers/index.js
  create app/javascript/controllers/application.js
  create app/javascript/controllers/hello_controller.js
Import Stimulus controllers
  append app/javascript/application.js
Pin Stimulus
Appending: pin "@hotwired/stimulus", to: "stimulus.min.js", preload: true"
  append config/importmap.rb
Appending: pin "@hotwired/stimulus-loading", to: "stimulus-loading.js", pr
eload: true
  append config/importmap.rb
Pin all controllers
Appending: pin_all_from "app/javascript/controllers", under: "controllers"
  append config/importmap.rb
PS C:\Users\calag>

```

Exploramos un momento el contenido de nuestro nuevo directorio de aplicaciones:

```

PS C:\Users\calag\rottenpotatoes> ls

Directorio: C:\Users\calag\rottenpotatoes

Mode                LastWriteTime         Length Name
----                -
d-----          11/10/2023      11:17             app
d-----          11/10/2023      11:17             bin
d-----          11/10/2023      11:17            config
d-----          11/10/2023      11:16              db
d-----          11/10/2023      11:16             lib
d-----          11/10/2023      11:17             log
d-----          11/10/2023      11:16            public
d-----          11/10/2023      11:16            storage
d-----          11/10/2023      11:16             test
d-----          11/10/2023      11:16             tmp
d-----          11/10/2023      11:17            vendor
-a-----          11/10/2023      11:16           703 .dockerignore
-a-----          11/10/2023      11:16           350 .gitattributes
-a-----          11/10/2023      11:16           815 .gitignore
-a-----          11/10/2023      11:16            11 .ruby-version
-a-----          11/10/2023      11:16           160 config.ru
-a-----          11/10/2023      11:16          2014 Dockerfile
-a-----          11/10/2023      11:16          2165 Gemfile
-a-----          11/10/2023      11:17          6011 Gemfile.lock
-a-----          11/10/2023      11:16           227 Rakefile
-a-----          11/10/2023      11:16           374 README.md

PS C:\Users\calag\rottenpotatoes>

```

Abrimos el archivo llamado Gemfile. En él ya encontraremos nombres de gemas sin la necesidad de que hayamos tenido que escribir algún código de aplicación:

```
Gemfile X
C:\Users\calag\rottenpotatoes> Gemfile
1 source "https://rubygems.org"
2
3 ruby "3.0.6"
4
5 # Bundle edge Rails instead: gem "rails", github: "rails/rails", branch: "main"
6 gem "rails", "~> 7.1.0"
7
8 # The original asset pipeline for Rails [https://github.com/rails/sprockets-rails]
9 gem "sprockets-rails"
10
11 # Use sqlite3 as the database for Active Record
12 gem "sqlite3", "~> 1.4"
13
14 # Use the Puma web server [https://github.com/puma/puma]
15 gem "puma", ">= 5.0"
16
17 # Use JavaScript with ESM import maps [https://github.com/rails/importmap-rails]
18 gem "importmap-rails"
19
20 # Hotwire's SPA-like page accelerator [https://turbo.hotwired.dev]
21 gem "turbo-rails"
22
23 # Hotwire's modest JavaScript framework [https://stimulus.hotwired.dev]
24 gem "stimulus-rails"
25
26 # Build JSON APIs with ease [https://github.com/rails/jbuilder]
27 gem "jbuilder"
28
29 # Use Redis adapter to run Action Cable in production
30 # gem "redis", ">= 4.0.1"
31
32 # Use Kredis to get higher-level data types in Redis [https://github.com/rails/kredis]
33 # gem "kredis"
34
35 # Use Active Model has_secure_password [https://guides.rubyonrails.org/active_model_basics.html#securepassword]
36 # gem "bcrypt" "~> 3.1.7"
```

Ya no hace falta agregar al Gemfile: **ruby '3.2.2'** debajo de la fuente de 'https://rubygems.org' porque ya se encuentra en la línea 4 nuestra versión de Ruby de manera automática.

Ahora, para que SQLite3 pueda funcionar de manera correcta con Rails, se nos sugiere realizar un cambio en la gema que se encuentra dentro de Gemfile, así:

```
# Use sqlite3 as the database for Active Record
gem "sqlite3", "~> 1.3"
```

Pero, lo dejaré en la versión que ya mencioné me sale que es la 1.4.

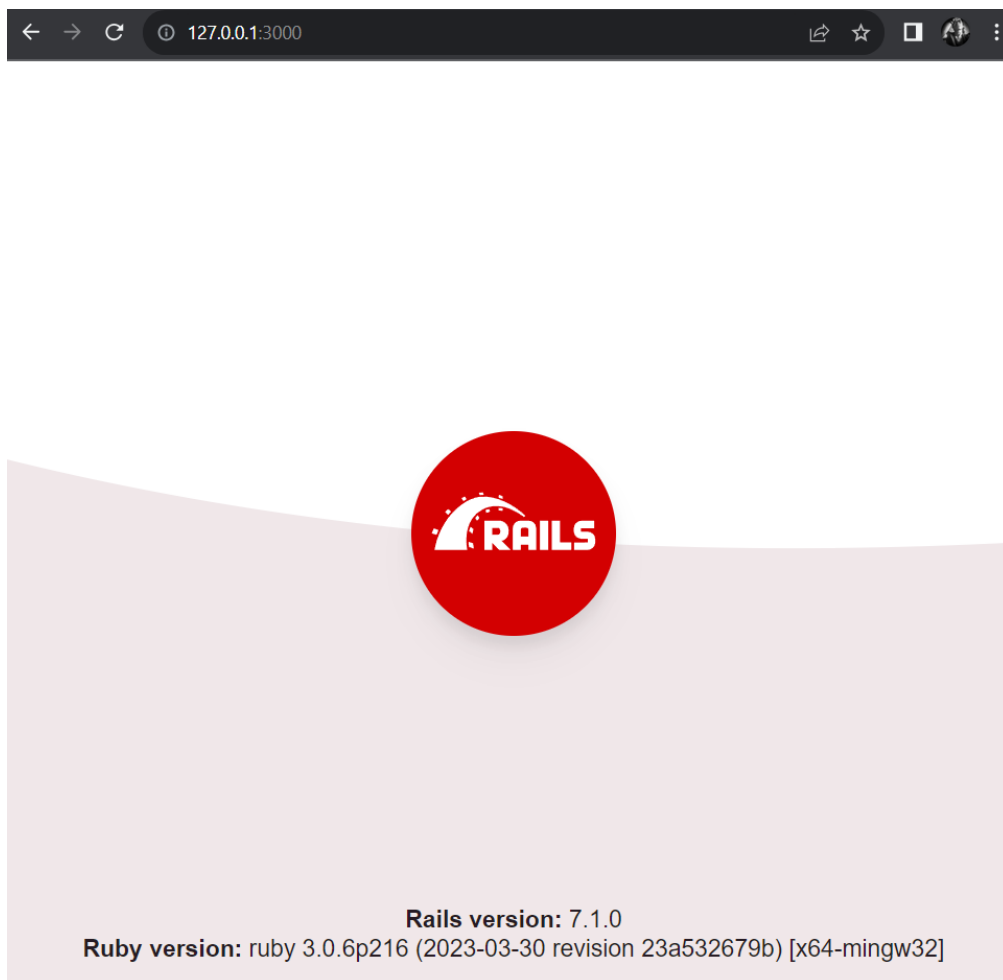
Ahora, de haber realizado el cambio, sólo tendría que ejecutar la actualización del paquete y verificar que la salida contenga "Fetching sqlite3 1.3.x" y "Installing sqlite3 1.3.x", donde x es cualquier versión menor. Pero dado que no necesitamos hacer el cambio, lo que nos aparece es lo siguiente:

```
PS C:\Users\calag\rottenpotatoes> bundle update
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies...
Bundle updated!
PS C:\Users\calag\rottenpotatoes>
```

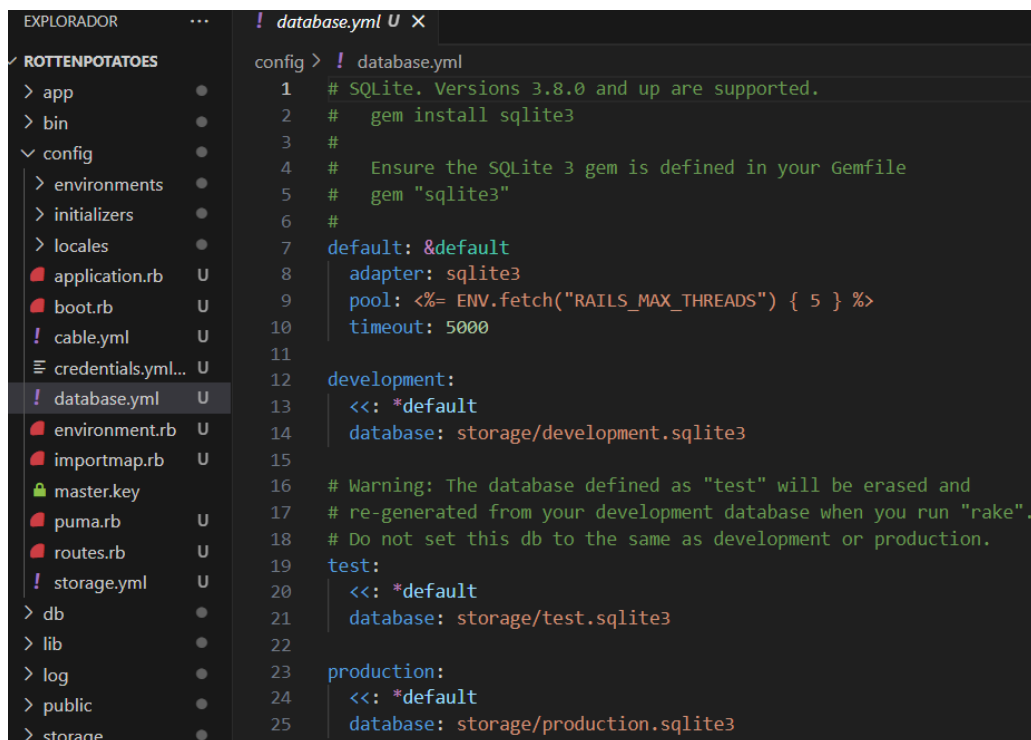
Ahora, para asegurarme de que todo funcione, ejecutaré la aplicación localmente utilizando el comando **rails server**, y me aparece lo siguiente:

```
PS C:\Users\calag\rottenpotatoes> rails server
=> Booting Puma
=> Rails 7.1.0 application starting in development
=> Run 'bin/rails server --help' for more startup options
*** SIGUSR2 not implemented, signal based restart unavailable!
*** SIGUSR1 not implemented, signal based restart unavailable!
*** SIGHUP not implemented, signal based logs reopening unavailable!
Puma starting in single mode...
* Puma version: 6.4.0 (ruby 3.0.6-p216) ("The Eagle of Durango")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 27412
* Listening on http://[::1]:3000
* Listening on http://127.0.0.1:3000
Use Ctrl-C to stop
```

Y al colocarlo en nuestro navegador, podemos ver que se despliega lo siguiente:



Rails almacena los entornos y los medios para conectarse a la base de datos asociada con cada uno de ellos en config/database.yml:



```
EXPLORADOR  ...  ! database.yml U X
✓ ROTTENPOTATOES
  > app
  > bin
  > config
    > environments
    > initializers
    > locales
    application.rb U
    boot.rb U
    ! cable.yml U
    credentials.yml... U
    ! database.yml U
    environment.rb U
    importmap.rb U
    master.key
    puma.rb U
    routes.rb U
    ! storage.yml U
  > db
  > lib
  > log
  > public
  > storage

config > ! database.yml
1 # SQLite. Versions 3.8.0 and up are supported.
2 #   gem install sqlite3
3 #
4 # Ensure the SQLite 3 gem is defined in your Gemfile
5 #   gem "sqlite3"
6 #
7 default: &default
8   adapter: sqlite3
9   pool: <%= ENV.fetch("RAILS_MAX_THREADS") { 5 } %>
10  timeout: 5000
11
12 development:
13   <<: *default
14   database: storage/development.sqlite3
15
16 # Warning: The database defined as "test" will be erased and
17 # re-generated from your development database when you run "rake".
18 # Do not set this db to the same as development or production.
19 test:
20   <<: *default
21   database: storage/test.sqlite3
22
23 production:
24   <<: *default
25   database: storage/production.sqlite3
```

Crear la base de datos

Utilizamos el comando rails **generate migration create_movies**:

```
PS C:\Users\calag\rottenpotatoes> rails generate migration create_movies
      invoke  active_record
      create  db/migrate/20231011165707_create_movies.rb
PS C:\Users\calag\rottenpotatoes>
```

Esto generará un archivo de migración en el directorio db/migrate con un nombre como 20231011234567_create_movies.rb. El nombre de la migración contiene una marca de tiempo para garantizar que las migraciones se apliquen en el orden correcto.

Editar la migración

En la dirección que se nos brinda editamos la migración, así:

```
20231011165707_create_movies.rb U X
db > migrate > 20231011165707_create_movies.rb
1  class CreateMovies < ActiveRecord::Migration[7.1]
2    def change
3      create_table :movies do |t|
4
5        t.string :title
6        t.string :rating
7        t.text :description
8        t.datetime :release_date
9
10       t.timestamps
11     end
12   end
13 end
14
```

Una vez que guardé este archivo, puedo aplicar la migración con el comando “rails db:migrate” para así crear la tabla movies en la base de datos de desarrollo y, posteriormente, puedo actualizar la base de datos de prueba con “rails db:test:prepare”:

```
PS C:\Users\calag\rottenpotatoes> rails db:migrate
== 20231011165707 CreateMovies: migrating =====
-- create_table(:movies)
   => 0.0035s
== 20231011165707 CreateMovies: migrated (0.0039s) =====
PS C:\Users\calag\rottenpotatoes>
```

Actualizamos la base de datos de prueba con los cambios de la migración ejecutando “rails db:test:prepare”:


```
PS C:\Users\calag\rottenpotatoes> rails db:test:prepare
PS C:\Users\calag\rottenpotatoes>
```

Crear el modelo inicial e inicializar la base de datos

Como se nos indica, Rails no tiene idea de que existe una base de datos que contiene una tabla de películas. Entonces el siguiente paso es crear el modelo ActiveRecord que utiliza esta tabla. Para ellos nos dirigimos a **app/models/movie.rb**


```
PS C:\Users\calag\rottenpotatoes> cd app
PS C:\Users\calag\rottenpotatoes\app> cd models
PS C:\Users\calag\rottenpotatoes\app\models> code movie.rb
```


Y dentro llenamos el archivo .rb de la siguiente manera:

```
app > models >  movie.rb
1  class Movie < ActiveRecord::Base
2  end
```

Sembrar la base de datos

Para hacer esto, crearé un archivo llamado “**seeds.rb**” en el directorio “**db**” y pegaré el código que se nos ha proporcionado:

```
db >  seeds.rb
1  # This file should ensure the existence of records required to run the application in every environment (production,
2  # development, test). The code here should be idempotent so that it can be executed at any point in every environment.
3  # The data can then be loaded with the bin/rails db:seed command (or created alongside the database with db:setup).
4  #
5  # Example:
6  #
7  # ["Action", "Comedy", "Drama", "Horror"].each do |genre_name|
8  #   MovieGenre.find_or_create_by!(name: genre_name)
9  # end
10 # Seed the RottenPotatoes DB with some movies.
11
12 ##DEJARÉ LOS COMENTARIOS DE ARRIBA INTENCIONALMENTE
13 more_movies = [
14   { title: 'Ganibal', rating: 'G', release_date: '25-Nov-1992' },
15   { title: 'Fuerza bruta', rating: 'R', release_date: '21-Jul-1989' },
16   { title: 'The Ring', rating: 'PG-13', release_date: '10-Aug-2011' },
17   { title: 'Alien: The Return', rating: 'PG', release_date: '12-Jun-1981' }
18 ]
19
20 more_movies.each do |movie|
21   Movie.create!(movie)
22 end
```

Luego ejecutaré “rake db:seed” en la terminal para sembrar la base de datos con estos datos iniciales:

```
PS C:\Users\calag\rottenpotatoes\db> rake db:seed
(in C:/Users/calag/rottenpotatoes)
PS C:\Users\calag\rottenpotatoes\db>
```

Y abriendo la consola de rails (usando ‘rails console’) podemos verificar que haya películas en la base de datos:

```
PS C:\Users\calag\rottenpotatoes\db> rails console
Loading development environment (Rails 7.1.0)
irb(main):001> Movie.first
  Movie Load (0.1ms)  SELECT "movies".* FROM "movies" ORDER BY "movies"."id" AS
C LIMIT ?  [["LIMIT", 1]]
=>
#<Movie:0x00000271c6096750
 id: 1,
 title: "Ganibal",
 rating: "G",
 description: nil,
 release_date: Wed, 25 Nov 1992 00:00:00.000000000 UTC +00:00,
 created_at: Wed, 11 Oct 2023 18:33:36.709688000 UTC +00:00,
 updated_at: Wed, 11 Oct 2023 18:33:36.709688000 UTC +00:00>
irb(main):002>
```

Creación de rutas, acciones y vistas CRUD para películas

Ejecutaré la aplicación nuevamente, pero esta vez, en lugar de visitar la página de inicio, intentaré visitar /movies, así:

```
PS C:\Users\calag\rottenpotatoes> rails server
=> Booting Puma
=> Rails 7.1.0 application starting in development
=> Run 'bin/rails server --help' for more startup options
*** SIGUSR2 not implemented, signal based restart unavailable!
*** SIGUSR1 not implemented, signal based restart unavailable!
*** SIGHUP not implemented, signal based logs reopening unavailable!
Puma starting in single mode...
* Puma version: 6.4.0 (ruby 3.0.6-p216) ("The Eagle of Durango")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 2808
* Listening on http://[::1]:3000
* Listening on http://127.0.0.1:3000
Use Ctrl-C to stop
```

← → ↻ ⓘ 127.0.0.1:3000/movies

Routing Error

No route matches [GET] "/movies"

Rails.root: C:/Users/calag/rottenpotatoes

[Application Trace](#) | [Framework Trace](#) | [Full Trace](#)

Routes

Routes match in priority from top to bottom

Helper (Path / Url)	HTTP Verb	Path	Controller#Action	Source Location
rails_health_check_path	GET	/up(.:format)	rails/health#show	config/routes.rb:6
turbo_recede_historical_location_path	GET	/recede_historical_location(.:format)	turbo/native/navigation#recede	
turbo_resume_historical_location_path	GET	/resume_historical_location(.:format)	turbo/native/navigation#resume	

¡Nunca había sentido tanta felicidad de que me salga un error y es que, que aparezca esto está bien!, lo que ocurre es que Rails reconoce que mi URI no coincide con ninguna ruta, porque no he especificado ninguna ruta que asigne URI a métodos de aplicación.

Si utilizo un editor para abrir el archivo “log/development.log” puedo observar que el mensaje de error se registra allí:

```
development.log X
log > development.log
44  [1] 36mTRANSACTION (0.1ms) [1] 36mcommit transaction[0m
45  [1] 36mTRANSACTION (0.0ms) [1] 36mbegin transaction[0m
46  [1] 36mMovie Create (0.5ms) [1] 32mINSERT INTO "movies" ("title", "rating", "description", "release_date", "created_at", "updated_at"
47  [1] 36mTRANSACTION (0.0ms) [1] 36mcommit transaction[0m
48  [1] 36mTRANSACTION (0.0ms) [1] 36mbegin transaction[0m
49  [1] 36mMovie Create (0.5ms) [1] 32mINSERT INTO "movies" ("title", "rating", "description", "release_date", "created_at", "updated_at"
50  [1] 36mTRANSACTION (0.1ms) [1] 36mcommit transaction[0m
51  [1] 36mMovie Load (0.1ms) [1] 34mSELECT "movies".* FROM "movies" ORDER BY "movies"."id" ASC LIMIT ? [1] 1]]
52  Started GET "/" for 127.0.0.1 at 2023-10-11 13:47:34 -0500
53  [1] 36mActiveRecord::SchemaMigration Load (0.1ms) [1] 34mSELECT "schema_migrations"."version" FROM "schema_migrations" ORDER BY "sche
54  Processing by Rails::WelcomeController#index as HTML
55  Rendering C:/Ruby30-x64/lib/ruby/gems/3.0.0/gems/railties-7.1.0/lib/rails/templates/rails/welcome/index.html.erb
56  Rendered C:/Ruby30-x64/lib/ruby/gems/3.0.0/gems/railties-7.1.0/lib/rails/templates/rails/welcome/index.html.erb (Duration: 0.8ms | Allocations: 57
57  Completed 200 OK in 169ms (Views: 36.4ms | ActiveRecord: 0.0ms | Allocations: 7370)
58
59
60  Started GET "/movies" for 127.0.0.1 at 2023-10-11 13:47:41 -0500
61
62  ActionController::RoutingError (No route matches [GET] "/movies"):
```

Crear rutas CRUD

Debo modificar el archivo “config/routes.rb” para definir las rutas CRUD para mi modelo ‘Movie’. Entonces, debo agregar las líneas que se me brindan al archivo de la siguiente manera:

```
development.log routes.rb U X
config > routes.rb
1  Rails.application.routes.draw do
2    resources :movies
3    root to: redirect('/movies')
4  end
```

Estas rutas CRUD son generadas automáticamente por Rails y asocian las acciones de crear, leer, actualizar y eliminar (CRUD) con el modelo ‘Movie’. Además, se redirige la ruta raíz (‘/’) a la lista de películas en ‘/movies’.

Nota:

Luego de guardar el archivo routes.rb debemos ejecutar “**rails route**” (‘rake’ es utilizada para versiones antiguas de rails) y observaremos que debido a mi cambio en routes.rb, la primera línea de salida dice que el URI GET/movies intentará llamar a la acción index de MoviesController:

```
PS C:\Users\calag\rottenpotatoes> rails routes
Prefix Verb URI Pattern Controller
#Action
movies GET /movies(.:format) movies#index
ex POST /movies(.:format) movies#create
reate
new_movie GET /movies/new(.:format) movies#new
edit_movie GET /movies/:id/edit(.:format) movies#edit
t
movie GET /movies/:id(.:format) movies#show
PATCH /movies/:id(.:format) movies#update
PUT /movies/:id(.:format) movies#update
DELETE /movies/:id(.:format) movies#destroy
root GET / redirect(301, /movies)
turbo_recede_historical_location GET /recede_historical_location(.:format) turbo/navigate
ve/navigation#recede
turbo_resume_historical_location GET /resume_historical_location(.:format) turbo/navigate
ve/navigation#resume
```

```

conductor/action_mailbox/inbound_emails#show
new_rails_conductor_inbound_email_source GET /rails/conductor/action_mailbox/inbound_emails/sources/new(.:format) rails/conductor
uctor/action_mailbox/inbound_emails/sources#new
rails_conductor_inbound_email_sources POST /rails/conductor/action_mailbox/inbound_emails/sources(.:format) rails/conductor
uctor/action_mailbox/inbound_emails/sources#create
rails_conductor_inbound_email_reroute POST /rails/conductor/action_mailbox/:inbound_email_id/reroute(.:format) rails/conductor
uctor/action_mailbox/reroutes#create
rails_conductor_inbound_email_incinerate POST /rails/conductor/action_mailbox/:inbound_email_id/incinerate(.:format) rails/conductor
uctor/action_mailbox/incinerates#create
rails_service_blob GET /rails/active_storage/blobs/redirect/:signed_id/*filename(.:format) active_storage
rage/blobs/redirect#show
rails_service_blob_proxy GET /rails/active_storage/blobs/proxy/:signed_id/*filename(.:format) active_storage
rage/blobs/proxy#show
GET /rails/active_storage/blobs/:signed_id/*filename(.:format) active_storage
rage/blobs/redirect#show
rails_blob_representation GET /rails/active_storage/representations/redirect/:signed_blob_id/:variation_key/*filename(.:format) active_storage
rage/representations/redirect#show
rails_blob_representation_proxy GET /rails/active_storage/representations/proxy/:signed_blob_id/:variation_key/*filename(.:format) active_storage
rage/representations/proxy#show
GET /rails/active_storage/representations/:signed_blob_id/:variation_key/*filename(.:format) active_storage
rage/representations/redirect#show
rails_disk_service GET /rails/active_storage/disk/:encoded_key/*filename(.:format) active_storage
rage/disk#show
update_rails_disk_service PUT /rails/active_storage/disk/:encoded_token(.:format) active_storage
rage/disk#update
rails_direct_uploads POST /rails/active_storage/direct_uploads(.:format) active_storage
rage/direct_uploads#create
PS C:\Users\calag\rottenpotatoes>

```

Y si recargo la página en mi navegador puedo notar un error diferente: MoviesController constante no inicializada.

127.0.0.1:3000/movies

Routing Error

uninitialized constant MoviesController

Rails.root: C:/Users/calag/rottenpotatoes

[Application Trace](#) | [Framework Trace](#) | [Full Trace](#)

Routes

Routes match in priority from top to bottom

Helper (Path / Url)	HTTP Verb	Path	Controller#Action	Source Location
Search				
movies_path	GET	/movies(.:format)	movies#index	config/routes.rb:2
	POST	/movies(.:format)	movies#create	config/routes.rb:2
new_movie_path	GET	/movies/new(.:format)	movies#new	config/routes.rb:2
edit_movie_path	GET	/movies/:id/edit(.:format)	movies#edit	config/routes.rb:2

Y sí, también este error es bueno. Rail no puede encontrar la clase MoviesController, pero el hecho de que incluso esté buscando esa clase nos dice que nuestra ruta está funcionando correctamente.

Crear acciones y vistas

Para crear tanto el archivo del controlador como las vistas asociadas, ejecutamos el siguiente comando:

“rails g scaffold_controller Movie title rating description release_date --skip-test”

```
PS C:\Users\calag\rottenpotatoes> rails g scaffold_controller Movie title rating
description release_date --skip-test
create    app/controllers/movies_controller.rb
invoke    erb
create    app/views/movies
create    app/views/movies/index.html.erb
create    app/views/movies/edit.html.erb
create    app/views/movies/show.html.erb
create    app/views/movies/new.html.erb
create    app/views/movies/_form.html.erb
create    app/views/movies/_movie.html.erb
invoke    resource_route
route     resources :movies
invoke    test_unit
create    test/controllers/movies_controller_test.rb
create    test/system/movies_test.rb
invoke    helper
create    app/helpers/movies_helper.rb
invoke    test_unit
invoke    jbuilder
create    app/views/movies/index.json.jbuilder
create    app/views/movies/show.json.jbuilder
create    app/views/movies/_movie.json.jbuilder
PS C:\Users\calag\rottenpotatoes>
```

Este comando generará el controlador 'MoviesController' y las vistas CRUD (crear, leer, actualizar, eliminar) para el modelo 'Movie'.

Cambiar la base de datos para producción

Como dije en un inicio, no usaré Heroku, en su lugar, usaré Render. Para eso debo asegurarme de que el proyecto esté versionado con Git y alojado en mi repositorio en GitHub.

Render admite bases de datos PostgreSQL, así que debo estar seguro de que la aplicación esté configurada para usar PostgreSQL en lugar de SQLite en el archivo 'config/database.yml'.

```
72  group :production do
73    |   gem 'pg', '~> 0.21' # Heroku deployment
74    |   gem 'rails_12factor'
75  end
```

```
11  group :development, :test do
12    |   # Use sqlite3 as the database for Active Record
13    |   gem "sqlite3", "~> 1.4"
14  end
```

```

config > ! database.yml
1  # SQLite. Versions 3.8.0 and up are supported.
2  #   gem install sqlite3
3  #
4  #   Ensure the SQLite 3 gem is defined in your Gemfile
5  #   gem "sqlite3"
6  #
7  default: &default
8    adapter: postgresql
9    pool: <%= ENV.fetch("RAILS_MAX_THREADS") { 5 } %>
10   timeout: 5000

```

Finalmente, como hemos modificado mi Gemfile, vuelvo a ejecutar bundle install.

```

PS C:\Users\calag\rottenpotatoes> bundle install
Fetching gem metadata from https://rubygems.org/.....
Resolving dependencies...
Installing pg 0.21.0 with native extensions
Bundle complete! 16 Gemfile dependencies, 73 gems now installed.
Gems in the groups 'development' and 'test' were not installed.
Use 'bundle info [gemname]' to see where a bundled gem is installed.
PS C:\Users\calag\rottenpotatoes>

```

← → ↺ 127.0.0.1:3000/movies

Movies

Title: Ganibal

Rating: G

Description:

Release date: 1992-11-25 00:00:00 UTC

[Show this movie](#)

Title: Fuerza bruta

Rating: R

Description:

Release date: 1989-07-21 00:00:00 UTC

[Show this movie](#)

Title: The Ring

Rating: PG-13

Description:

Release date: 2011-08-10 00:00:00 UTC

[Show this movie](#)

Title: Alien: The Return

Rating: PG

Description:

Release date: 1981-06-12 00:00:00 UTC

Finalmente, el despliegue del servidor se hará por Render y el link lo colocaré en un único archivo de texto.