

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

FACULTAD DE CIENCIAS Y TECNOLOGÍAS.

INGENIERIA EN TELEMATICA.



COMPONENTE: SAAS.

Grupo 2 Telemática

INTEGRANTES:

- Hosting Stewart Lopez Martinez.

DOCENTE: Ervin Montes

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

1. Crear un nuevo proyecto en Rails.

1.1 Crear un proyecto y ubicarse en su interior haciendo uso del terminal.

```
Hosting@debian 2024-09-10 13:43:53 Proyectos_RoR $ rails new App_Scaffold
create
create README.md
create Rakefile
create .ruby-version
create config.ru
create .gitignore
create .gitattributes
create Gemfile
```

1.2 Generar un scaffold, con una tabla llamada Buy con los campos:

```
Restrict your gemfile to Ruby 3.1 or 3.2 to stop seeing this message
Hosting@debian 2024-09-10 17:56:10 App_Scaffold $ rails generate scaffold Buy category:string description:text amount:decimal
Running via Spring preloader in process 5790
invoke active_record
create db/migrate/20240910235710_create_buys.rb
create app/models/buy.rb
invoke test_unit
create test/models/buy_test.rb
create test/fixtures/buys.yml
invoke resource_route
route resources :buys
invoke scaffold_controller
create app/controllers/buys_controller.rb
invoke erb
create app/views/buys
create app/views/buys/index.html.erb
create app/views/buys/edit.html.erb
create app/views/buys/show.html.erb
create app/views/buys/new.html.erb
create app/views/buys/_form.html.erb
invoke resource_route
invoke test_unit
create test/controllers/buys_controller_test.rb
create test/system/buys_test.rb
invoke helper
```

2. Migrar los datos.

Las migraciones son el modo más conveniente de cambiar el esquema de la base de datos a través del tiempo de una manera consistente y fácil, utilizan un lenguaje de definición de esquemas (DSL) en Ruby, por lo que no tiene que escribir SQL.

```
Hosting@debian 2024-09-10 17:57:28 App_Scaffold $ rake db:migrate
== 20240910235710 CreateBuys: migrating =====
-- create_table(:buys)
   -> 0.0724s
== 20240910235710 CreateBuys: migrated (0.0781s) =====
```

3. Rutas

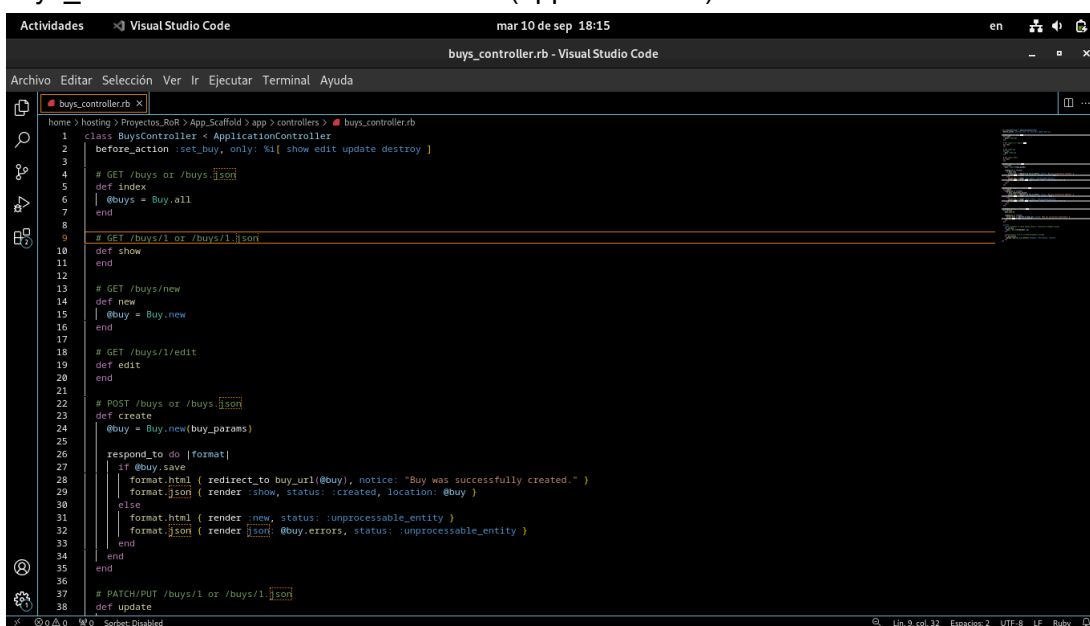
3.1 Escribir el siguiente comando en el terminal y observar las rutas generadas.

NOTA: El comando rake routes ya no se usa, ahora tiene que ser rails routes.

```
Hosting@debian 2024-09-10 18:02:00 App_Scaffold $ rails routes
           Prefix Verb   URI Pattern
buys GET    /buys(.:format)
buys POST   /buys(.:format)
buys#new GET    /buys/new(.:format)
buys#edit GET    /buys/:id/edit(.:format)
```

4. Verificar el controlador.

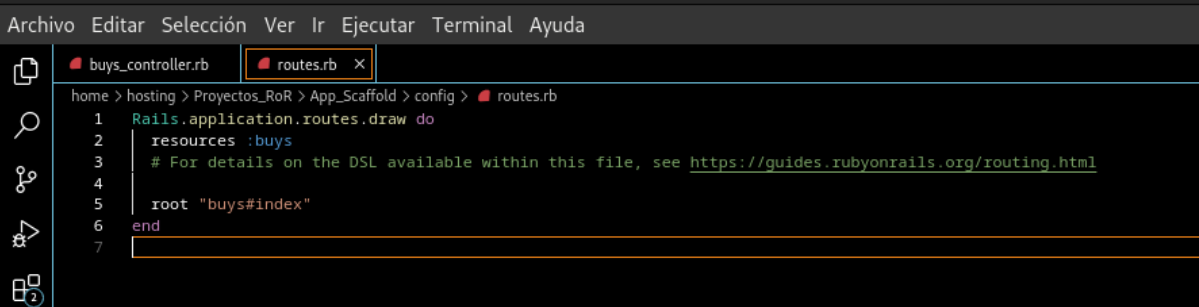
Observar que se han creado los distintos métodos del controlador en el archivo `buys_controller.rb` dentro del directorio `(app/controller/)`.



```
Actividades Visual Studio Code mar 10 de sep 18:15
buys_controller.rb - Visual Studio Code
Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda
buys_controller.rb
home > hosting > Proyectos_RoR > App_Scaffold > app > controllers > buys_controller.rb
1 class BuysController < ApplicationController
2   before_action :set_buy, only: [:show, :edit, :update, :destroy]
3
4   # GET /buys or /buys.json
5   def index
6     @buys = Buy.all
7   end
8
9   # GET /buys/1 or /buys/1.json
10  def show
11  end
12
13  # GET /buys/new
14  def new
15    @buy = Buy.new
16  end
17
18  # GET /buys/1/edit
19  def edit
20  end
21
22  # POST /buys or /buys.json
23  def create
24    @buy = Buy.new(buy_params)
25
26    respond_to do |format|
27      if @buy.save
28        format.html { redirect_to buy_url(@buy), notice: "Buy was successfully created." }
29        format.json { render :show, status: :created, location: @buy }
30      else
31        format.html { render :new, status: :unprocessable_entity }
32        format.json { render :json, @buy.errors, status: :unprocessable_entity }
33      end
34    end
35  end
36
37  # PATCH/PUT /buys/1 or /buys/1.json
38  def update
```

5. Configurar el index generado por el scaffold como la página de inicio de la aplicación.

5.1 Escribir la siguiente línea de código en el archivo config/routes.rb.



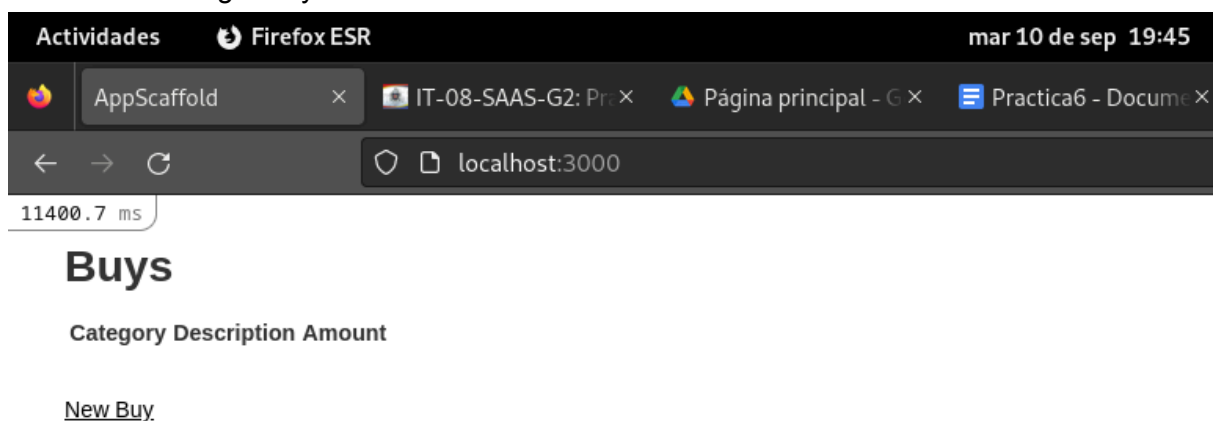
```
home > hosting > Proyectos_RoR > App_Scaffold > config > routes.rb
1  Rails.application.routes.draw do
2    resources :buys
3    # For details on the DSL available within this file, see https://guides.rubyonrails.org/routing.html
4
5    root "buys#index"
6  end
7
```

5.2 Iniciar el servidor.



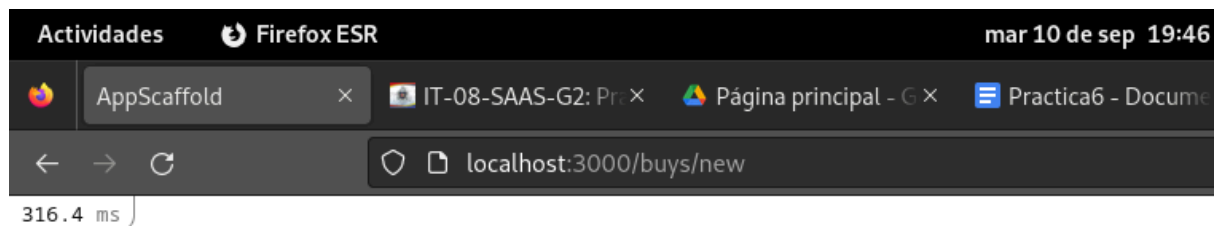
```
Hosting@debian 2024-09-10 18:26:44 Proyectos_RoR $ cd App_Scaffold
Hosting@debian 2024-09-10 18:26:58 App_Scaffold $ rails s
=> Booting Puma
=> Rails 6.1.7.8 application starting in development
=> Run `bin/rails server --help` for more startup options
Puma starting in single mode...
* Puma version: 5.6.8 (ruby 3.1.2-p20) ("Birdie's Version")
* Min threads: 5
* Max threads: 5
* Environment: development
* PID: 7353
* Listening on http://127.0.0.1:3000
* Listening on http://[::1]:3000
Use Ctrl-C to stop
```

5.3 Abrir el navegador y acceder a la dirección.

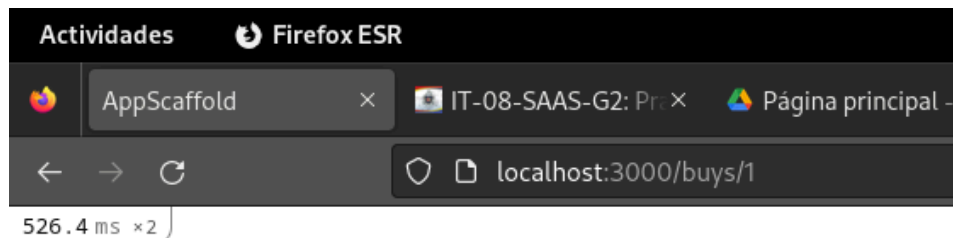


6. Verificar el funcionamiento de las vistas.

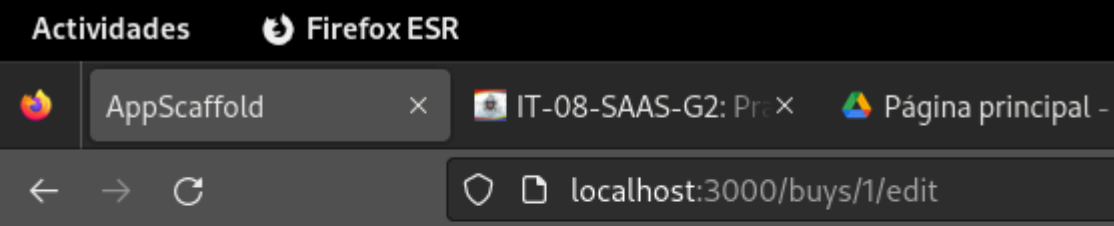
6.1 En la vista mostrada anteriormente, presiona new buy, se mostrará el formulario para poder agregar datos a la aplicación.



6.2 Ingresar una nueva compra llenando el formulario, al presionar el botón create buy, el dato se ha creado y almacenado correctamente.



1.1 Click en Edit, podrá observar que permite editar el dato anteriormente almacenado.



148.1 ms

Editing Buy

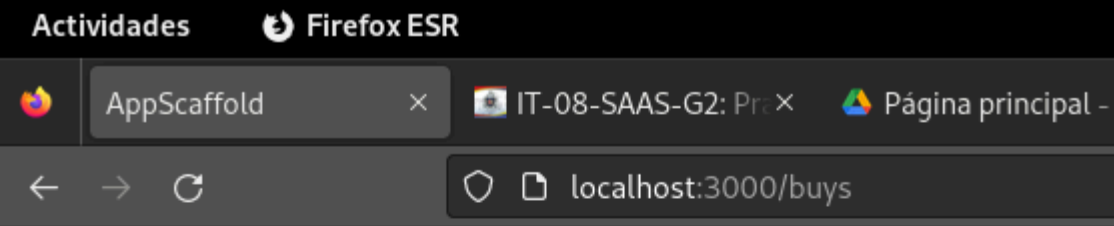
Category

Description

Amount

[Show](#) | [Back](#)

1.2 Verificar que se muestran los datos almacenados dentro del index de la aplicación.



122.3 ms

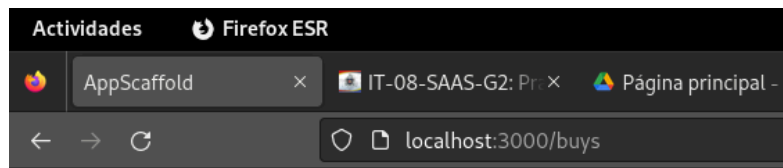
Buys

Category	Description	Amount	
Zapatos	Para ir a clase	500.0	Show Edit Destroy

[New Buy](#)

2. Validaciones al modelo buy

2.1 Abrir el archivo /model/buy.rb y agregar el siguiente código.



New Buy

3 errors prohibited this buy from being saved:

- Category can't be blank
- Description can't be blank
- Amount can't be blank

Category

Description

Amount

Create Buy

[Back](#)

Ejercicios propuestos para ser entregados al docente

2. Crear un proyecto nuevo, utilice scaffold para generar el código y crear una tabla de nombre Estudiante con los campos nombres, apellidos, carrera, carnet, fecha de nacimiento, edad, celular; deberá configurar el archivo routes.rb para que la página principal de la aplicación sea el index generado por el scaffold, validar el campo celular para que solo admita número y que no permita campos vacíos. La aplicación deberá mostrar un formulario parecido al de la figura 69, se puede observar en la figura como el framework crea automáticamente las cajas de texto, dependiendo del tipo de dato que se le especifica al generar el scaffold, como el campo fecha de nacimiento donde crea un input tipo date_select en el formulario para seleccionar la fecha de nacimiento del estudiante.

Actividades Firefox ESR

AppScaffold × IT-08-SAAS-G2: Pr... × Página principal

← → ↻ localhost:3000/estudiantes/new

196.3 ms

New Estudiante

Nombres

Apellidos

Carrera

Carnet

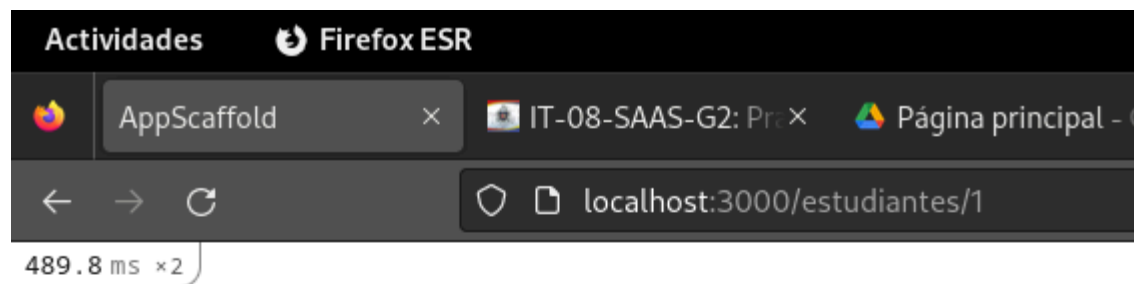
Fecha de nacimiento
2024 ▼ September ▼ 11 ▼

Edad

Celular

Create Estudiante

[Back](#)



Estudiante was successfully created.

Nombres: Hosting

Apellidos: Lopez

Carrera: Telematica

Carnet: 21-02153-0

Fecha de nacimiento: 2019-04-17

Edad: 21

Celular: 76087881

[Edit](#) | [Back](#)

Actividades Firefox ESR

AppScaffold × IT-08-SAAS-G2: Pr × Página principal -

← → ↻ localhost:3000/estudiantes/1/edit

326.5 ms

Editing Estudiante

Nombres

Apellidos

Carrera

Carnet

Fecha de nacimiento

Edad

Celular

Update Estudiante

[Show](#) | [Back](#)

Actividades Firefox ESR mar 10 de sep 20:22

AppScaffold × IT-08-SAAS-G2: Pr × Página principal - G × Practica6 - Docume ×

← → ↻ localhost:3000/estudiantes

186.4 ms

Estudiantes

Nombres	Apellidos	Carrera	Carnet	Fecha de nacimiento	Edad	Celular	
Hosting	Lopez	Telematica	21-02153-0	2019-04-17	21	76087881	Show Edit Destroy

[New Estudiante](#)