
Ruby on Rails

— Sessions Controller & View —

login/new.html.erb

```
<div class="bs-docs-section">
  <div class="row">
    <div class="col-lg-12">
      <div class="page-header">
        <h1 id="forms">Login</h1>
      </div>
    </div>
  </div>
```

Es para obtener un reviewer, y se hace referencia a los fields de reviewer

Se hace submit a la url de session

```
  <div class="row">
    <div class="col-lg-6">
      <div class="well bs-component">
        <%= form_for :reviewer, url: sessions_path, class: "form-horizontal" do |f| %>
          <div class="form-group">
            <div><%= f.label :name , class: "col-lg-2 control-label" %> <br/> <%= f.text_field :
              name , class: "form-control"%></div>
            <div><%= f.label :password %> <br/> <%= f.password_field :password , class: "form-
              control"%></div>
          </div>
          <div class="actions"><%= f.submit "Login" , class: "btn btn-primary"%></div>
        <% end %>
      </div>
    </div>
  </div>
</div>
```

reviewer[name] / reviewer[password]

Login

Name

div | 238.68 x 71.67

Password

.....

Login

```
27cc0f1v05Dfmg-->
▼<div class="form-group">
  ▼<div>
    <label class="col-lg-2 control-label" for=
      "reviewer_name">Name</label>
    <br>
    ... <input class="form-control" type="text" name="reviewer[name]
      id="reviewer_name"> == $0
    </div>
  ▶<div>...</div>
</div>
```

Session Controller

sessions_controller.rb

```
1 class SessionsController < ApplicationController
2
3   def new
4     # Login Page - new.html.erb
5   end
6
7   def create
8     reviewer = Reviewer.find_by(name: params[:reviewer][:name])
9     password = params[:reviewer][:password]
10
11     if reviewer && reviewer.authenticate(password)
12       session[:reviewer_id] = reviewer.id
13       redirect_to root_path, notice: "Logged in successfully"
14     else
15       redirect_to login_path, alert: "Invalid username/password combination"
16     end
17   end
18
19   def destroy
20     reset_session # wipe out session and everything in it
21     redirect_to login_path, notice: "You have been logged out"
22   end
23 end
```

localhost:3000

Logged in successfully

Logged in successfully

Listing Books

| Name | Author |
|-------------------------------|--|
| Eloquent Ruby | Russ Olsen ShowEditDestroy |
| Beginning Ruby | Peter Cooper ShowEditDestroy |
| Metaprogramming Ruby 2 | Paolo Perrotta ShowEditDestroy |
| Design Patterns in Ruby | Russ Olsen ShowEditDestroy |
| The Ruby Programming Language | David Flanagan ShowEditDestroy |

New Book

Application

Manifest

Service Workers

Clear storage

Storage

Local Storage

http://localhost:3000

Session Storage

http://localhost:3000

IndexedDB

Web SQL

Cookies

| Name | Value |
|-----------------------|-------------------------------|
| _utma | 111872281.2013253281.1459624. |
| _blog_web_app_session | VFRWcy9MZDI2MGtOU3BBYy8zVj |
| _bodega_session | VWpWV0FiRUg5UVBwTTlhOWtjS. |
| _i_reviewed_session | eitYMEJScm0vZ1czeU1OMFovTXR |
| _practica_session | Mnl2L21jME4zb0syTnVKaVp3RUV |

Llaveando nuestra aplicación

Ya tenemos el login!

- Pero siempre podemos acceder via url a los demas recursos.
- Vamos a agregar un **before_action** en el **ApplicationController** (del cual todos los demas controllers heredan) que enviará de vuelta al login si todavía no estamos logeados.
- Pero si todo está bloqueado... el **SessionController** también hereda de Application controller..
- Los Controllers pueden sobre escribir el **before_action** con **skip_before_action**.

application_controller.rb x

```
1 class ApplicationController < ActionController::Base
2   # Prevent CSRF attacks by raising an exception.
3   # For APIs, you may want to use :null_session instead.
4   protect_from_forgery with: :exception
5   before_action :ensure_login
6
7   protected
8   def ensure_login
9     # Always go to login page unless session contains
10    # reviewer_id
11    redirect_to login_path unless session[:reviewer_id]
12  end
13 end
14
```

sessions_controller.rb ●

```
1 class SessionsController < ApplicationController
2   skip_before_action :ensure_login, only: [:new, :create]
3
4   def new
5     # Login Page - new.html.erb
6   end

```

Invalid username/password combination

Login

Name

Joe

Password

•••••

Login

Entonces...

La página de login corresponde a un action new en el SessionController, pero utiliza atributos de reviewer.

Para “llavear” nuestra aplicación utilizamos el before_action en el ApplicationController.