

## GNU/Linux Administration - Support #710

### Building a Simple Messenger Web Application with Ruby on Rails

12/09/2015 10:54 AM - Daniel Curtis

<b>Status:</b>	Closed	<b>Start date:</b>	12/09/2015
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Daniel Curtis	<b>% Done:</b>	100%
<b>Category:</b>	Ruby	<b>Estimated time:</b>	3.00 hours
<b>Target version:</b>	*nix	<b>Spent time:</b>	7.00 hours

#### Description

**NOTE:** This guide was adapted from the Ruby on Rails tutorial on [Codecademy](#)

- Create the new Ruby on Rails web application:

```
rails new message_app
```

- Change into the message\_app directory:

```
cd message_app
```

- Start the rails development web server locally:

```
bundle exec rails s
```

- **NOTE:** The development web server bind the IP address to the localhost by default, so accessing it over the network will not work. To allow network access to the development web server:

```
bundle exec rails s -b 0.0.0.0
```

## The Main Page

### Main Page Controller

- Create a controller for the Page class:

```
rails generate controller Page
```

- Edit the routes file:

```
vi config/routes.rb
```

- And add the route for the default landing page:

```
Rails.application.routes.draw do
  get '/' => "page#home"
end
```

- Edit the Page controller:

```
vi app/controllers/page_controller.rb
```

- And create the home method:

```
class PageController < ApplicationController
  def home
  end
end
```

## Main Page View

- Edit the home page view:

```
vi app/views/page/home.html.erb
```

- And add the following bare html page:

```
<div class="main">
  <div class="container">
    <h1>Hello World!</h1>
  </div>
</div>
```

## The Message Page

### Create the Message Model

- Start by creating a Message model:

```
rails generate model Message
```

- Edit the database migrate file:

```
vi db/migrate/20151209164020_create_messages.rb
```

- And add the `*t.text :content*` to create the content row to the messages table:

```
class CreateMessages < ActiveRecord::Migration
  def change
    create_table :messages do |t|
      t.text :content
      t.timestamps
    end
  end
end
```

- Run the database migration:

```
rake db:migrate
```

### Create the Message Controller

- Create the Messages controller:

```
rails generate controller Messages
```

## Index Method

- Next edit the routes file:

```
vi config/routes.rb
```

- Then map the URL /messages to the Messages controller index action:

```
Rails.application.routes.draw do
  get '/' => "page#home"
  get '/messages' => "messages#index"
end
```

- Edit the Messages controller:

```
vi app/controllers/messages_controller.rb
```

- And define an index method with a class array named @messages that contains all the messages in the database table:

```
class MessagesController < ApplicationController
  def index
    @messages = Message.all
  end
end
```

## Create the Message View

- Edit the messages index Ruby HTML view:

```
vi app/views/messages/index.html.erb
```

- Then create a for loop to loop through the contents of the @messages array:

```
<div class="header">
  <div class="container">
    <h1>Messenger</h1>
  </div>
</div>

<div class="messages">
  <div class="container">
    <% @messages.each do |message| %>
      <div class="message">
        <p class="content"><%= message.content %></p>
        <p class="time"><%= message.created_at %></p>
      </div>
    <% end %>
  </div>
</div>
```

## Create the New Message Function

## New Method

- Edit the routes file

```
vi config/routes.rb
```

- Then create a route that maps requests to messages/new to the Message controller's new action:

```
Rails.application.routes.draw do
  get '/' => "page#home"
  get '/messages' => 'messages#index'
  get 'messages/new' => 'messages#new'
end
```

- Edit the Messages controller:

```
vi app/controllers/messages_controller.rb
```

- And define the new method that adds new messages to the database table:

```
class MessagesController < ApplicationController
  def index
    @messages = Message.all
  end

  def new
    @message = Message.new
  end
end
```

## Create Method

- Edit the routes file again

```
vi config/routes.rb
```

- And map requests to the Message controller's create action:

```
Rails.application.routes.draw do
  get '/' => "page#home"
  get '/messages' => 'messages#index'
  get 'messages/new' => 'messages#new'
  post 'messages' => 'messages#create'
end
```

- Edit the Messages controller again:

```
vi app/controllers/messages_controller.rb
```

- And create the create method and the private message\_params method below the new method:

```
class MessagesController < ApplicationController
  def index
    @messages = Message.all
  end
```

```

def new
  @message = Message.new
end

def create
  @message = Message.new(message_params)
  if @message.save
    redirect_to messages_path
  else
    render 'new'
  end
end

private
def message_params
  params.require(:message).permit(:content)
end
end

```

## Create the New Message View

- Edit the new message Ruby HTML view:

```
vi app/views/messages/new.html.erb
```

- And add the following:

```

<div class="header">
  <div class="container">
    <h1>Messenger</h1>
  </div>
</div>

<div class="create">
  <div class="container">
    <%= form_for(@message) do |f| %>
      <div class="field">
        <%= f.label :message %><br>
        <%= f.text_area :content %>
      </div>
      <div class="actions">
        <%= f.submit "Create" %>
      </div>
    <% end %>
  </div>
</div>

```

- Edit the messages index Ruby HTML view:

```
vi app/views/messages/index.html.erb
```

- And add the `link_to` function:

```

<div class="header">
  <div class="container">
    <h1>Messenger</h1>
  </div>
</div>

```

```

<div class="messages">
  <div class="container">
    <% @messages.each do |message| %>
      <%= link_to 'New Message', messages_new_path %>
      <div class="message">
        <p class="content"><%= message.content %></p>
        <p class="time"><%= message.created_at %></p>
      </div>
    <% end %>
  </div>
</div>

```

## Create the Show Message Function

- Edit the routes file again

```
vi config/routes.rb
```

- And map requests to the Message controller's create action:

```

Rails.application.routes.draw do
  get '/' => "page#home"
  get '/messages' => 'messages#index'
  get 'messages/new' => 'messages#new'
  post 'messages' => 'messages#create'
  get '/messages/:id', to: 'messages#show', as: 'message'
end

```

## Show Method

- Edit the Messages controller again:

```
vi app/controllers/messages_controller.rb
```

- And create the create method and the private message\_params method below the new method:

```

class MessagesController < ApplicationController
  def index
    @messages = Message.all
  end

  def new
    @message = Message.new
  end

  def create
    @message = Message.new(message_params)
    if @message.save
      redirect_to messages_path
    else
      render 'new'
    end
  end

  private
  def message_params
    params.require(:message).permit(:content)
  end
end

```

```
def show
  @message = Message.find(params[:id])
end
end
```

## Create the Show Message View

- Edit the show message Ruby HTML view:

```
vi app/views/messages/show.html.erb
```

- And add the following:

```
<div class="header">
  <div class="container">
    <h1>Messenger</h1>
  </div>
</div>

<div class="show">
  <div class="container">
    <div class="message">
      <%= @message.content %>
    </div>
  </div>
</div>
```

## Resources

- <https://www.codecademy.com/en/courses/learn-rails>
- [http://guides.rubyonrails.org/getting\\_started.html](http://guides.rubyonrails.org/getting_started.html)
- <http://guides.rubyonrails.org/routing.html>
- [http://guides.rubyonrails.org/layouts\\_and\\_rendering.html](http://guides.rubyonrails.org/layouts_and_rendering.html)

## History

---

### #1 - 12/09/2015 11:01 AM - Daniel Curtis

- Description updated
- Status changed from New to In Progress
- % Done changed from 0 to 30

### #2 - 12/09/2015 11:19 AM - Daniel Curtis

- Description updated

### #3 - 03/12/2016 07:51 PM - Daniel Curtis

- Subject changed from Building a Simple Messenger Web Application in Ruby to Building a Simple Messenger Web Application with Ruby on Rails
- Description updated

### #4 - 03/13/2016 09:04 PM - Daniel Curtis

- Description updated
- % Done changed from 30 to 50

### #5 - 05/08/2016 05:13 PM - Daniel Curtis

- % Done changed from 50 to 60
- Description updated

### #6 - 05/12/2016 02:57 PM - Daniel Curtis

- Status changed from In Progress to Resolved

- % Done changed from 60 to 100

**#7 - 05/13/2016 09:42 PM - Daniel Curtis**

- Status changed from Resolved to Closed

**#8 - 07/08/2016 10:29 PM - Daniel Curtis**

- Description updated

**#9 - 07/09/2016 11:29 AM - Daniel Curtis**

- Description updated

**#10 - 07/09/2016 11:30 AM - Daniel Curtis**

- Description updated

**#11 - 07/15/2016 10:13 PM - Daniel Curtis**

- Description updated

**#12 - 07/17/2016 12:00 PM - Daniel Curtis**

- Description updated

**#13 - 07/17/2016 12:49 PM - Daniel Curtis**

- Description updated