



Developing Amazing Web Apps with Ruby on Rails

Gabriel Guimaraes '17

Stephen Turban '17

Zack Chauvin '17

Model View Controller

MVC as a Store



The Cast: Vince



Beautiful, but stupid.



The Cast: Carl



Controlling, But Incapable.



The Cast: Mitt



Capable, But Submissive



The Process:

- Let's go to the store.
- Volunteers needed.



What is RoR?

Framework





```
unsigned int DJBHash(const char* str)
{
    unsigned int hash = 5381;

    for(int i = 0; i < strlen(str); i++)
    {
        hash = ((hash << 5) + hash) + str[i];
    }

    return hash;
}
```

```
dictionary = Hash.new  
dictionary["foo"] = true
```

```
if dictionary["foo"] == true  
  return true
```


Gems

Plenty of help online

FUN!

Just like C...

- puts "Hello World"
- x = 1
- while x < 50
 - if x > 25
 - puts "Big!"
 - else
 - puts "Small..."
 - end
- x += 1
- end

But more powerful!

- `x = 5`
- `puts "I am #{x}" if x > 3`
- `list = ["John", 4, 2]`
- `list << 3`
- `List.each do |elem|`
- `puts elem`
- `end`

Ruby + HTML

- `<body>`
- `<% if user.logged_in? %>`
- `<div>Welcome <%= user.name %></div>`
- `<% end %>`
- `</body>`

Resources

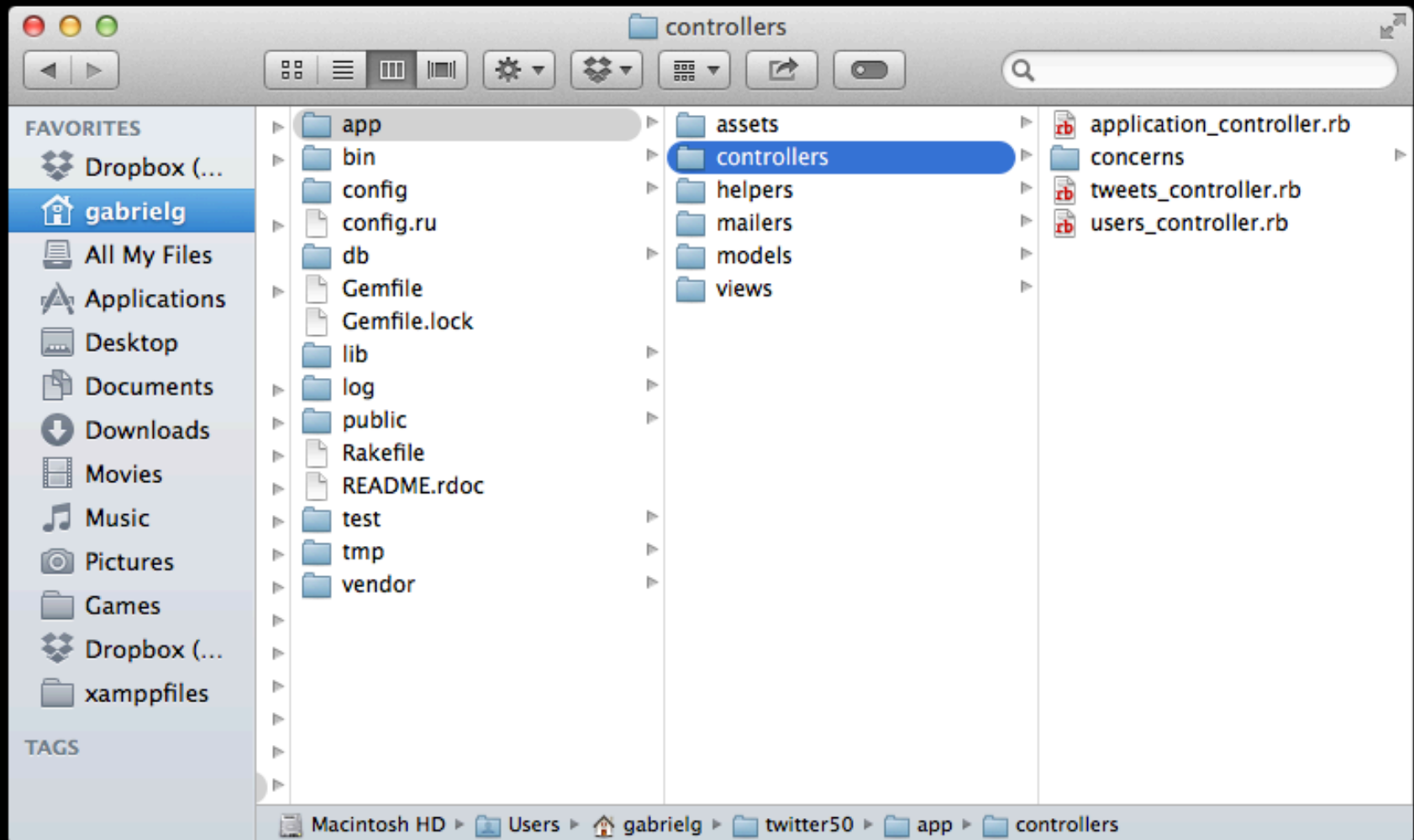
- <http://ruby-lang.org>
- <http://tryruby.org>
- <http://codecademy.com>



RAILS is LOVE

Why Rails?

- MVC makes everything simpler
- Less time spent on low-level problems
- Very popular right now! Rails 4.0!



Welcome to Twitter50, Harvard's own Twitter!

You are signed in as gablg [Sign out!](#)

To view, edit or delete your profile, click [here](#).

What's on your mind?

Tweet!

Enter your query here

Here are some tweetz:

CS50 is the bestest!

created at 3:04AM by @chauvin

Cats!

created at 3:06AM by @chauvin

Hi there [Delete](#)

created at 3:31PM by gablg

Model

- Each table in the Database has a corresponding model
- rails generate model User email:string password:string
- Database relations - has_many/belongs_to

* tweet.rb

* user.rb

```
class Tweet < ActiveRecord::Base
  belongs_to :user

  validates :body, presence: true, length: {maximum: 140}
  validates :user, presence: true

  def self.filter(q)
    if q
      where("body LIKE '%#{q}%'")
    else
      all
    end
  end
end
```

View



HTML & CSS

twitter50

- app
 - assets
 - controllers
 - helpers
 - mailers
 - models
 - concerns
 - tweet.rb
 - user.rb
 - views
 - devise
 - layouts
 - application.html.e
 - tweets
 - _tweets.html.erb
 - delete.html.erb
 - index.html.erb
 - index.js.erb
 - new.html.erb
 - users
 - edit.html.erb
 - new.html.erb
 - profile.html.erb
 - update.html.erb
- bin
- config
- config.ru
- db
- Gemfile
- Gemfile.lock
- lib
- log
- public
- Rakefile
- README.rdoc
- test
- tmp
- vendor

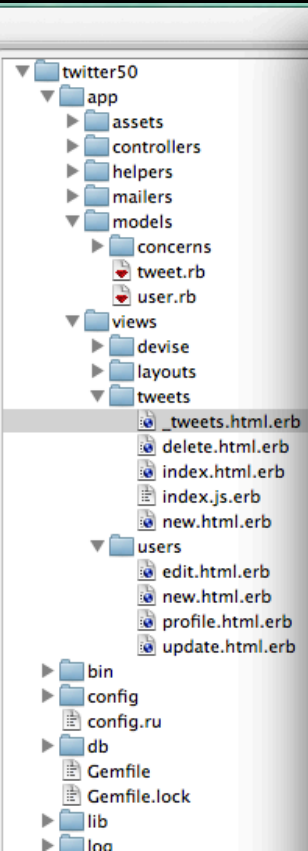
× tweet.rb × profile.html.erb × new.html.erb × index.html.erb × new.html.erb × index.js.erb × _tweets.html.erb × application.html.erb × delete.html.erb × user.rb

```
<!DOCTYPE html>
<html>
<head>
  <title>Twitter50</title>
  <link rel="stylesheet" href="//netdna.bootstrapcdn.com/bootstrap/3.1.1/css/bootstrap.min.css">
  <%= stylesheet_link_tag 'application' %>
  <%= javascript_include_tag 'application' %>
  <%= csrf_meta_tags %>
</head>
<body>

  <div class="container">
    <div class="box">
      <% if flash[:success] %>
        <div class="flash success"><%= flash[:success] %></div>
      <% end %>

      <% if flash[:error] %>
        <div class="flash error"><%= flash[:error] %></div>
      <% end %>

      <%= yield %>
    </div>
  </div>
</body>
</html>
```



```
twitter50
├── app
│   ├── assets
│   ├── controllers
│   ├── helpers
│   ├── mailers
│   ├── models
│   │   ├── concerns
│   │   ├── tweet.rb
│   │   └── user.rb
│   ├── views
│   │   ├── devise
│   │   ├── layouts
│   │   └── tweets
│   │       ├── _tweets.html.erb
│   │       ├── delete.html.erb
│   │       ├── index.html.erb
│   │       ├── index.js.erb
│   │       ├── new.html.erb
│   │       └── users
│   │           ├── edit.html.erb
│   │           ├── new.html.erb
│   │           ├── profile.html.erb
│   │           └── update.html.erb
├── bin
├── config
├── config.ru
├── db
├── Gemfile
├── Gemfile.lock
├── lib
└── log
```

twitter50 — _tweets.html.erb

✕ tweet.rb ✕ profile.html.erb ✕ new.html.erb ✕ index.html.erb ✕ new.html.erb ✕ index.js.erb ✕ _tweets.html.erb ✕ delete.html.erb ✕ user.rb

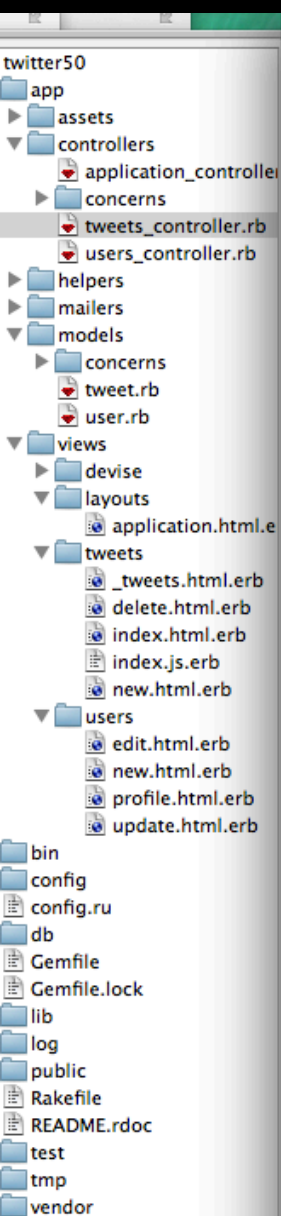
```
<% if @tweets.any? %>
<div class='bold'>Here are some tweets:</div>
<% else %>
<div class='bold'>No tweets to display!</div>
<% end %>

<div class='tweets'>
  <% @tweets.each do |tweet| %>
    <div class='tweet'>
      <%= tweet.body %>
      <% if tweet.user == current_user %>
        <%= link_to 'Delete', "/tweets/#{tweet.id}", method: :delete, class: 'delete' %>
      <% end %>
      <br>
      <span class="italic small">created at <%= tweet.created_at.strftime("%l:%M%p") %> by <%= tweet.user.handle %></span>

    </div>
  <% end %>
  <%= will_paginate @posts %>
</div>
```

Controller

- Interface between Model and View
- Loads Information for the view
- Calls methods from Model



* tweet.rb * profile.html.erb * new.html.erb * index.html.erb * new.html.erb * index.js.erb * _tweets.html.erb * application.html.erb * application_controller.rb

```
class TweetsController < ApplicationController
  def index
    @tweets = Tweet.filter(params[:q]).paginate(:page => params[:page], :per_page => 3)
    @new_tweet = Tweet.new
    @user = current_user
  end

  def create
    @tweet = Tweet.new(tweet_params)
    @tweet.user = current_user

    flash[:error] = @tweet.errors.full_messages.join('\n') if !@tweet.save

    redirect_to '/'
  end

  def new
  end

  def delete
    @tweet = Tweet.find(params[:id])
    if @tweet.user == current_user
      flash[:success] = "Tweet with body: \"#{@tweet.body}\" destroyed!"
      @tweet.destroy
    else
      flash[:error] = "That's not yours to delete!"
    end
    redirect_to '/'
  end

  private

  def tweet_params
    params.require(:tweet).permit(:body)
  end
end
```


Installing Rails

- <http://rubyonrails.org/download>
- http://guides.rubyonrails.org/getting_started.html
- Appliance
- Windows
- Mac

Questions?

