AoR-AntiPatterns

Richard Schneeman

Rails Developer for Gowalla

Rails 3.1 Contributor :)

CabForward & EngineYard

http://github.com/schneems

About this Presentation

Condensed 16 hour advanced class

1 hour beginner Antipatterns

What is an Antipattern?

Model View Controller

What is a view?

.html.erb

display and format information

What isn't a view?

NOT where we put business logic

```ruby
# Not where we query the database
User.where(:username => "Schneems"
<!--  app/view/definition/show.htı

<% if @definition.present? %>
    <% @top_user = @definition.use
                    order("rank DI
    <%= link_to "Top User", @top_u
<% else %>
  # ...
```

Antipattern

SQL queries in your views

Solution

Move them to your controllers

```ruby
# /app/controllers/definitions_co
def show
  @definition = Definition.where(
                           first
  @top_user   = @definition.users
                    order("rank D
  respond_with  @definition
end
```

View is much better

```
<!--  app/view/definition/show.htr

<% if @top_user.present? %>
    <%= link_to "Top User", @top_u
<% else %>
  # ...
```

Controllers

What is a controller?

Put your business logic

map resources to CRUD (Create, Read, Update, Delete)

What is a controller?

Index

Show

New

Create

Edit

What isn't a controller?

add_definition

change_first_language

save_definitions

editi_exisiting_definitions

update_all_languages

show_child_add

Antipattern

One controller with many many methods

Solution

Many controllers with about 7 methods

Why ~ 7 Methods?

Sometimes you don't need all 7 actions (Index, Show, New, create, Edit, Update, Destroy)

Sometimes you need a few

First rule of controllers ...

Never have more than 13 methods in a controller

Second rule of controllers ...

NEVER EVER have more than 13 methods in a controller

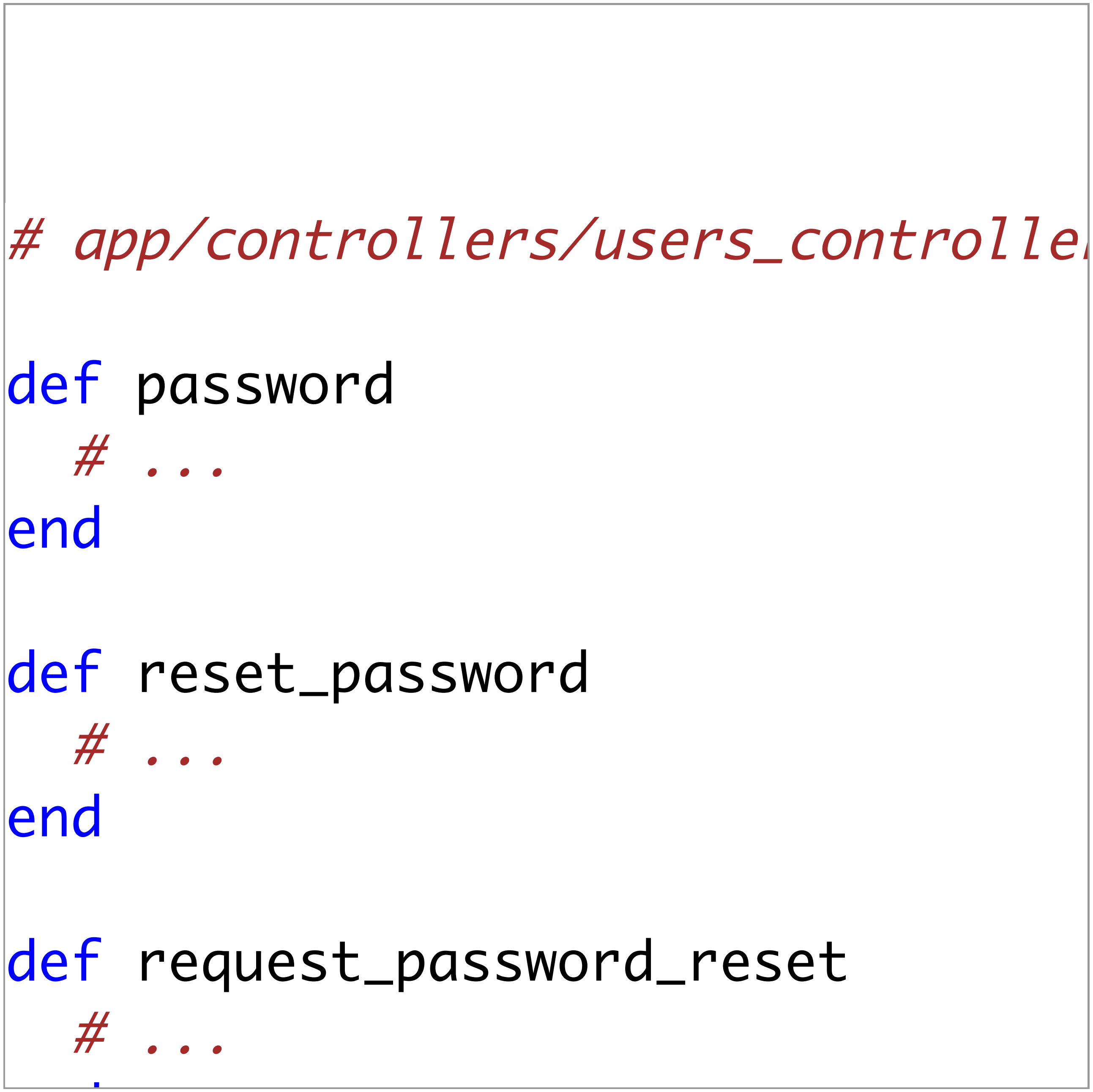
OR DHH will hunt you down

Third rule of controllers ...

Typically don't have more than 7 methods in a controller

Common Example

Reset Password



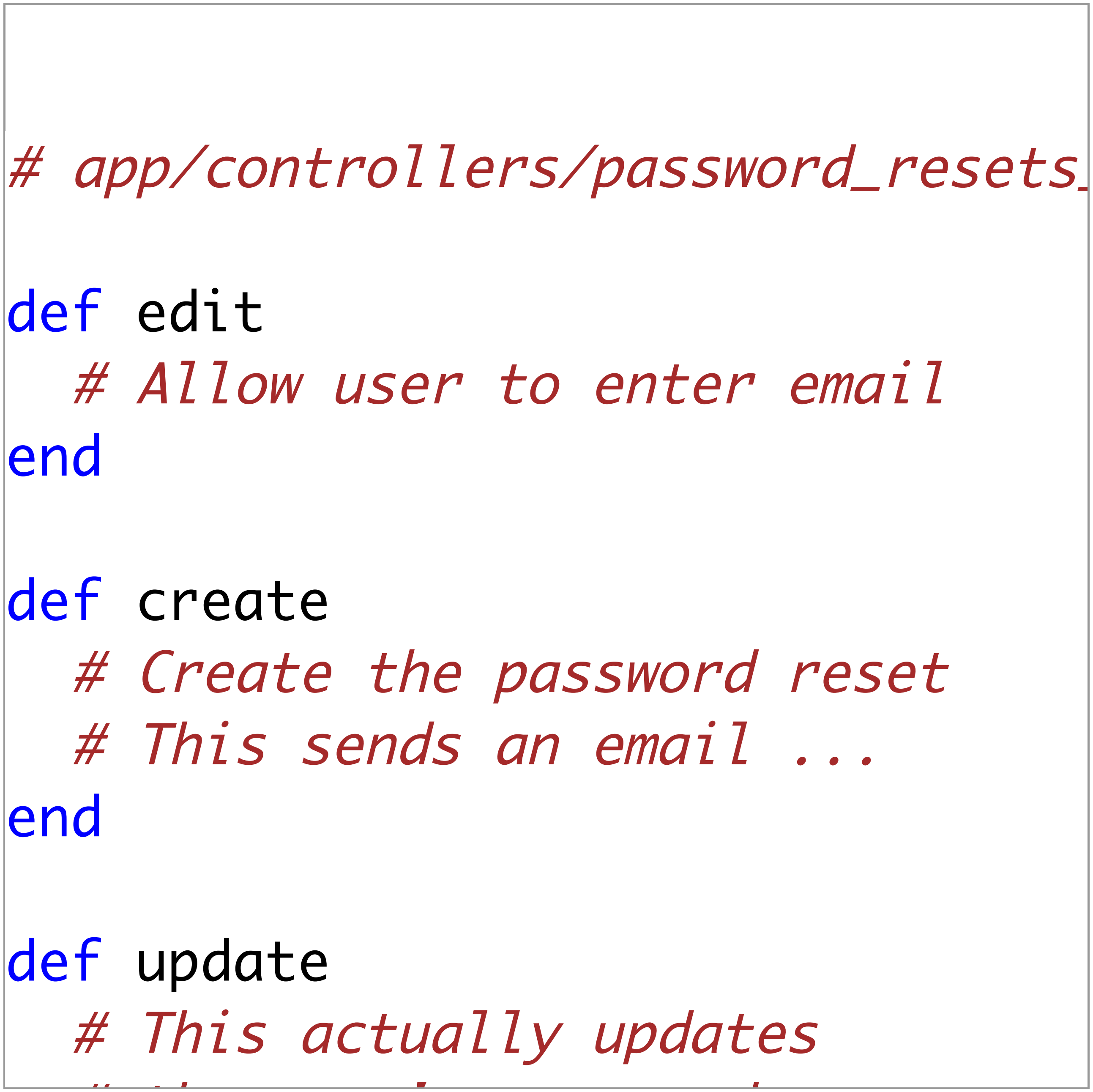
```ruby
# app/controllers/users_controlle

def password
  # ...
end

def reset_password
  # ...
end

def request_password_reset
  # ...
```

Move to Password Resets Controller



```
# app/controllers/password_resets_

def edit
  # Allow user to enter email
end

def create
  # Create the password reset
  # This sends an email ...
end

def update
  # This actually updates
```

SMAC your controller

Stop (see how non-standard actions you have)

Ask (Do they act on another resource?)

Move (Move methods to

Models

Skinny Controllers?

Fat Models ?

```
# /app/controllers/definitions_co
def show
  @definition = Definition.where(
                          first
  @top_user   = @definition.users
                    order("rank D
  respond_with @definition
end
```

```ruby
# app/models/definitions.rb

def top_user
  self.users.order("rank DESC").f
end
```

```ruby
# /app/controllers/definitions_co
def show
  @definition = Definition.where(
                          first
  @top_user   = @definition.top_us
  respond_with  @definition
end
```

Remember

No SQL in the view

No more than 13 actions on controllers

(Shoot for ~ 7)

Move logic

Remember M.V.C

Models-hold-logic-not **V**iews or **C**ontrollers

Questions ?

http://github.com/schneems/john

http://github.com/schneems/keyt

Antipatterns through
CabForward

The Biggest Beginner Antipattern of All?

Antipattern

Not writing & shipping code

Solution

Pick a project

stick with it

ship it

Solution

Get Involved

Come to AustinOnRails

Come to AustinRB

Take classes

Ask questions on StackOverflow

One Last Thing

Rails Mentors

Questions

SRSLY this time