Ruby on Rails
web application dev

lesson 4

about me   **thomas** mak

thomas@cpttm.org.mo

agenda

rails console

lab – adding category

uploading files

rails console

rails console

querying your data storage and playing with the active_record commands.

rails console

```
$ rails console
```

rails console

> Comment

```
irb(main):002:0> Comment
=> Comment(id: integer, name: string,
content: text, created_at: datetime,
updated_at: datetime, job_id: integer)
```

rails console

Creating new comment object.

```
irb(main):003:0> c = Comment.new
=> #<Comment id: nil, name: nil,
content: nil, created_at: nil,
updated_at: nil, job_id: nil>

irb(main):004:0> c.name = "Thomas"
=> "Thomas"

irb(main):005:0> c.content = "test"
=> "test"
```

rails console

adding job to comment.

```
irb(main):006:0> c.job = Job.first
  Job Load (0.2ms)  SELECT "jobs".* FROM
"jobs" LIMIT 1
=> #<Job id: 1, title: "111111", content:
"11111", created_at: "2013-09-13 07:03:30"
, updated_at: "2013-09-13 07:27:17",
category: "ruby", salary: 10000,
valid_till: "2023-09-13">


irb(main):007:0> c
=> #<Comment id: nil, name: "Thomas",
content: "test", created_at: nil,
updated_at: nil, job_id: 1>
```

rails console

saving the comment object.

```
irb(main):008:0> c.save
   (0.1ms)  begin transaction
  SQL (12.0ms)  INSERT INTO "comments"
("content", "created_at", "job_id",
"name", "updated_at") VALUES (?, ?, ?,
?, ?)  [["content", "test"],
["created_at", Wed, 18 Sep 2013 15:36:11
UTC +00:00], ["job_id", 1], ["name",
"Thomas"], ["updated_at", Wed, 18 Sep 2013
 15:36:11 UTC +00:00]]
   (0.9ms)  commit transaction
=> true

irb(main):009:0> c
=> #<Comment id: 1, name: "Thomas",
content: "test", created_at: "2013-09-18
 15:36:11", updated_at: "2013-09-18
15:36:11", job_id: 1>
```

rails console

selecting all comments for a job object.

```
irb(main):010:0> j = Job.first
  Job Load (0.4ms)  SELECT "jobs".* FROM
"jobs" LIMIT 1
=> #<Job id: 1, title: "111111", content:
"11111", created_at: "2013-09-13 07:03:30",
updated_at: "2013-09-13 07:27:17", category:
"ruby", salary: 10000, valid_till:
"2023-09-13">

irb(main):011:0> j.comments
  Comment Load (0.2ms)  SELECT "comments".*
FROM "comments" WHERE "comments"."job_id" = 1
=> [#<Comment id: 1, name: "Thomas", content:
 "test", created_at: "2013-09-18 15:36:11",
updated_at: "2013-09-18 15:36:11", job_id: 1>]
```

rails console

counting the comments.

```
irb(main):018:0> j.comments.count
    (0.3ms)  SELECT COUNT(*) FROM
"comments" WHERE "comments"."job_id" = 1
=> 1
```

lab – adding category

adding category

try to allow assigning one and only one category to each job.

adding category

steps:
1. category model
2. category controller
3. form for category
4. associate with job
5. set category in job

selecting the category as a list instead of inputing the number.

```
f.collection_select :category_id, Category.all, :id, :title, prompt: 'Choose a category'
```

uploading files

uploading files

two popular gems:
— paperclip
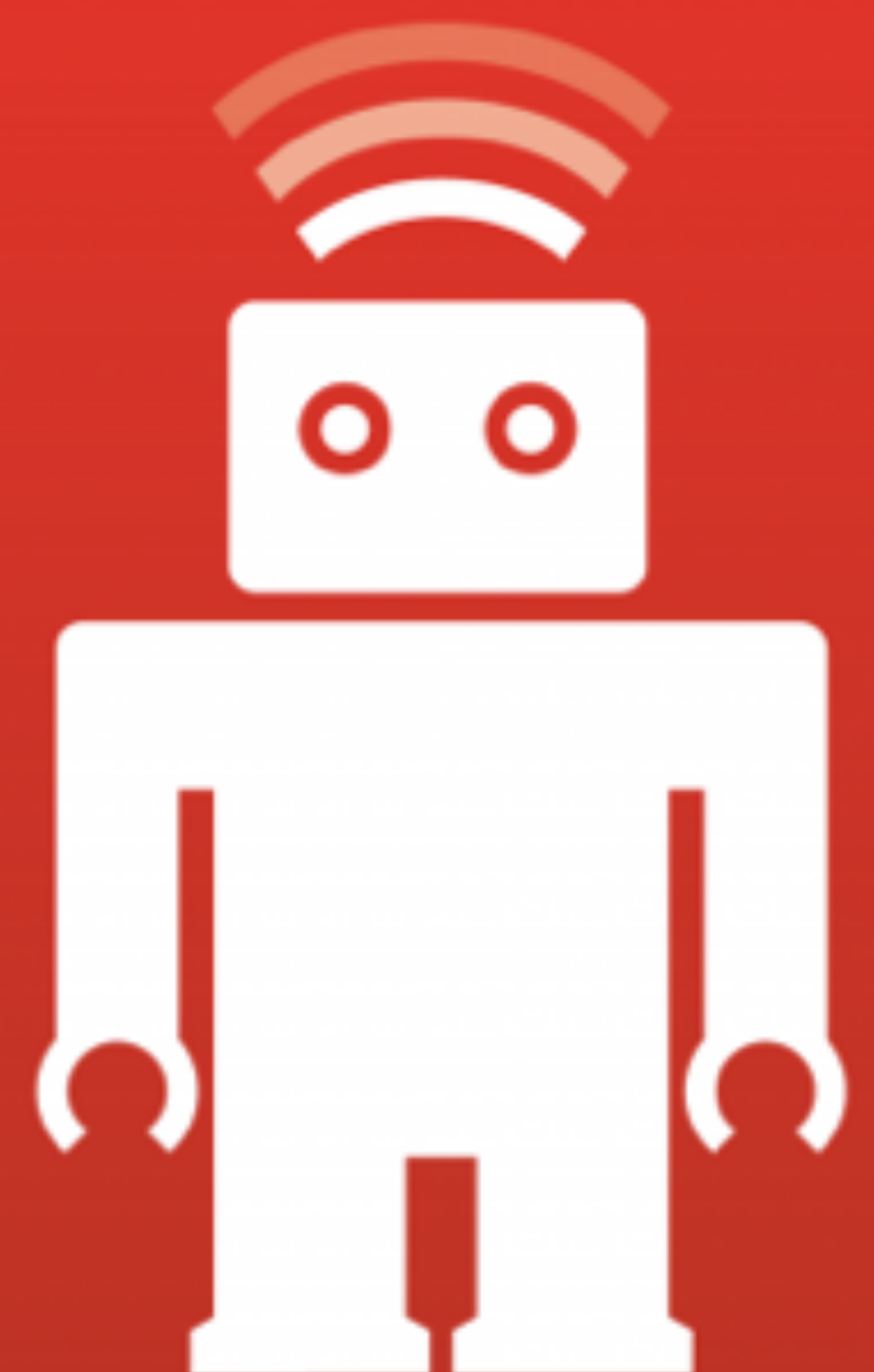**— carrierwave**

uploading
files

paperclip:
must bound to model

carrierwave:
indepentent to model

uploading
files

paperclip gem

https://github.com/thoughtbot/paperclip

uploading files

we need to install imagemagick for image resize.

```
$ brew install imagemagick
```

uploading files

add the gem to Gemfile. Then call bundle install.

```
gem "paperclip"
```

uploading files

add a poster image to job model.
add the following to **job.rb** file.

```ruby
has_attached_file :poster,
:styles => { :medium => "300x300>",
:thumb => "100x100>" },
:default_url => "/images/:style/missing.png"
```

uploading files

how about the database changes?

`$ rails generate paperclip job poster`

then we add the new poster to the job form.

```erb
<%= form_for @job :html => { :multipart => true } do |f| %>
  <%= f.file_field :poster %>
<% end %>
```

uploading files

how about displaying the poster image?

```erb
<%= image_tag @job.poster.url %>
<%= image_tag @job.poster.url(:medium) %>
<%= image_tag @job.poster.url(:thumb) %>
```

recources

github.com/thoughtbot

blog.railsthemes.com/
2013/05/14/my-favorite-
gem-paperclip/

bcjordan.github.io/posts/
rails-image-uploading-
framework-comparison/