Nested Routes
(Project / TimeEntry Index)

Imagine you want to display the time entries of a project

Time tracking tool
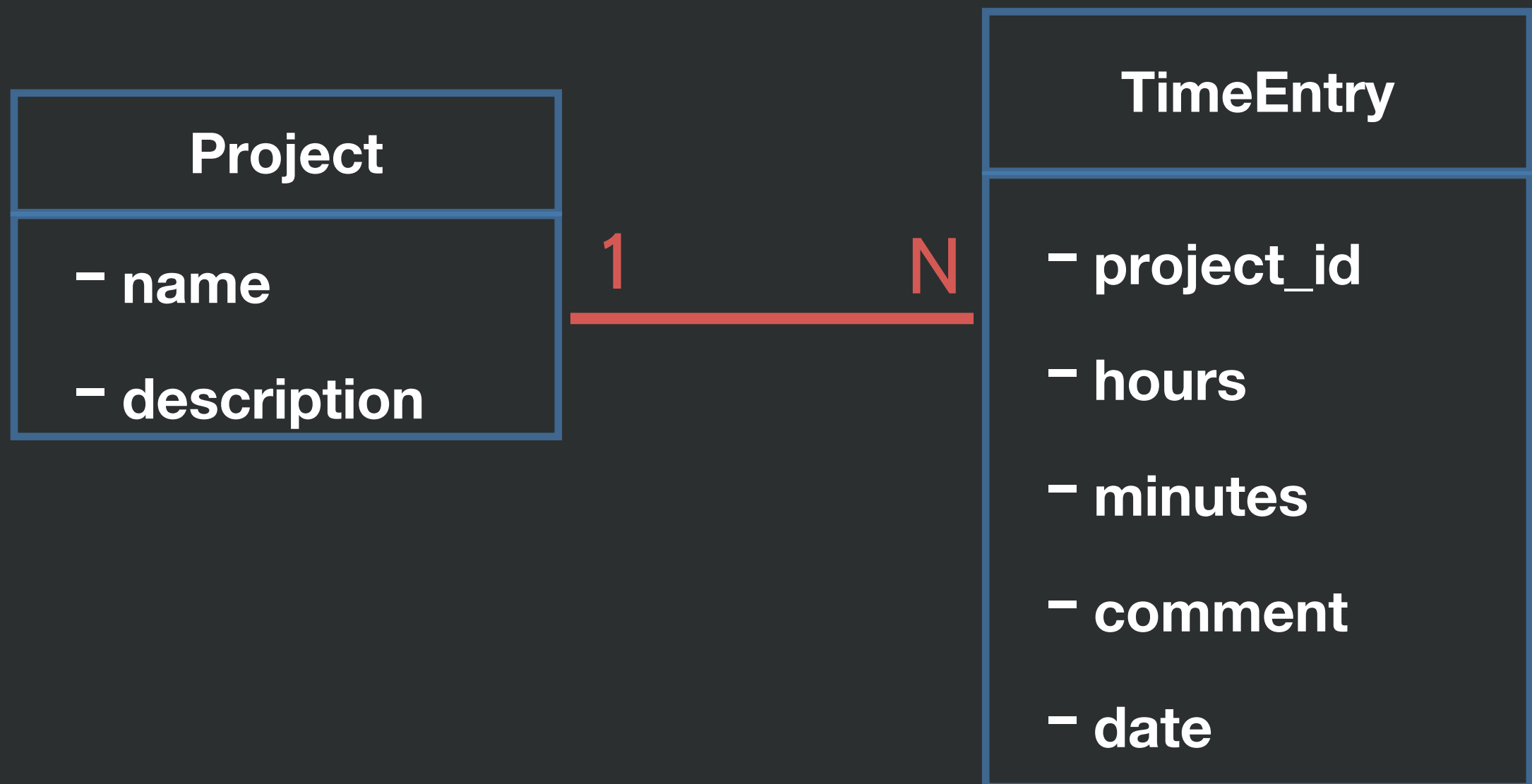
—

Ironhack

- 0 h 34 m - 2014-02-16
- 1 h 14 m - 2014-02-15
- 0 h 24 m - 2014-02-14

[Back to Projects](#) · [New entry](#)

Ironhack - Rails introduction
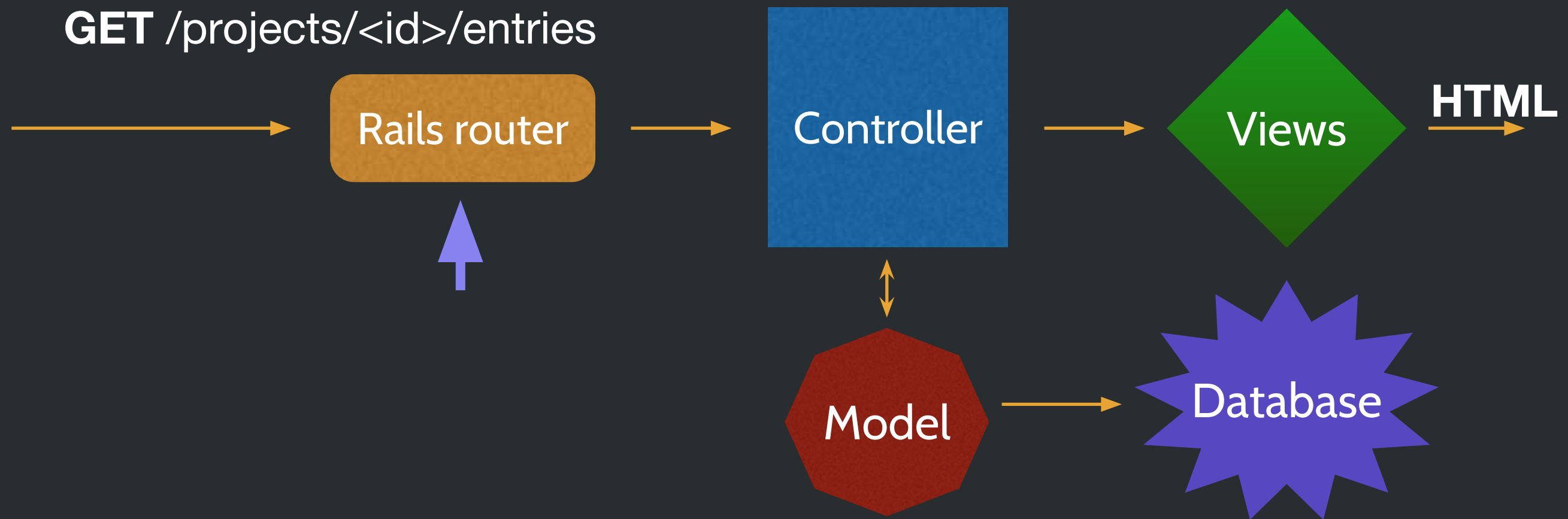
How do you identify it in an URL?

Model: TimeEntry

Project

- name

- description

1 —————— N

TimeEntry

- project_id

- hours

- minutes

- comment

- date

/projects/**\<id\>**

Parameter!

/projects/**<id>**/**time_entries**

Project
identifier!

GET /projects/<id>/entries

Rails router → Controller → Views → **HTML**

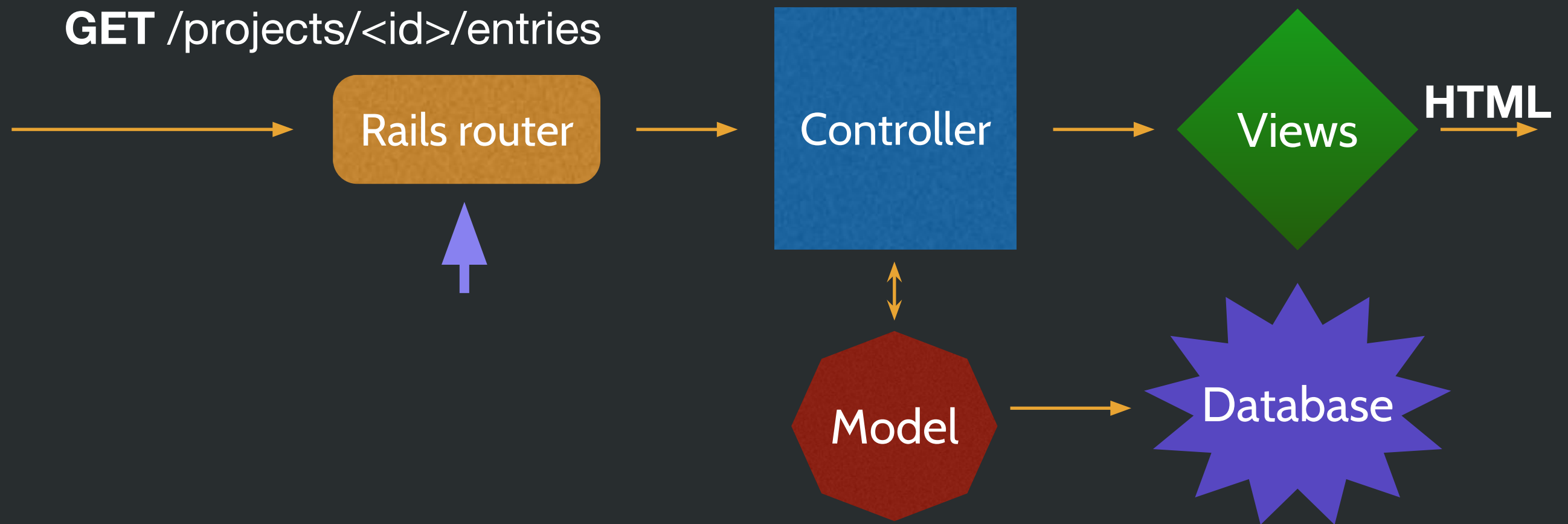Controller ↕ Model → Database

config/routes.rb

```ruby
Timetracking::Application.routes.draw do

  get '/', to: 'site#home'

  get '/projects', to: 'projects#index'

  get '/projects/:id', to: 'projects#show'

  get ??

end
```
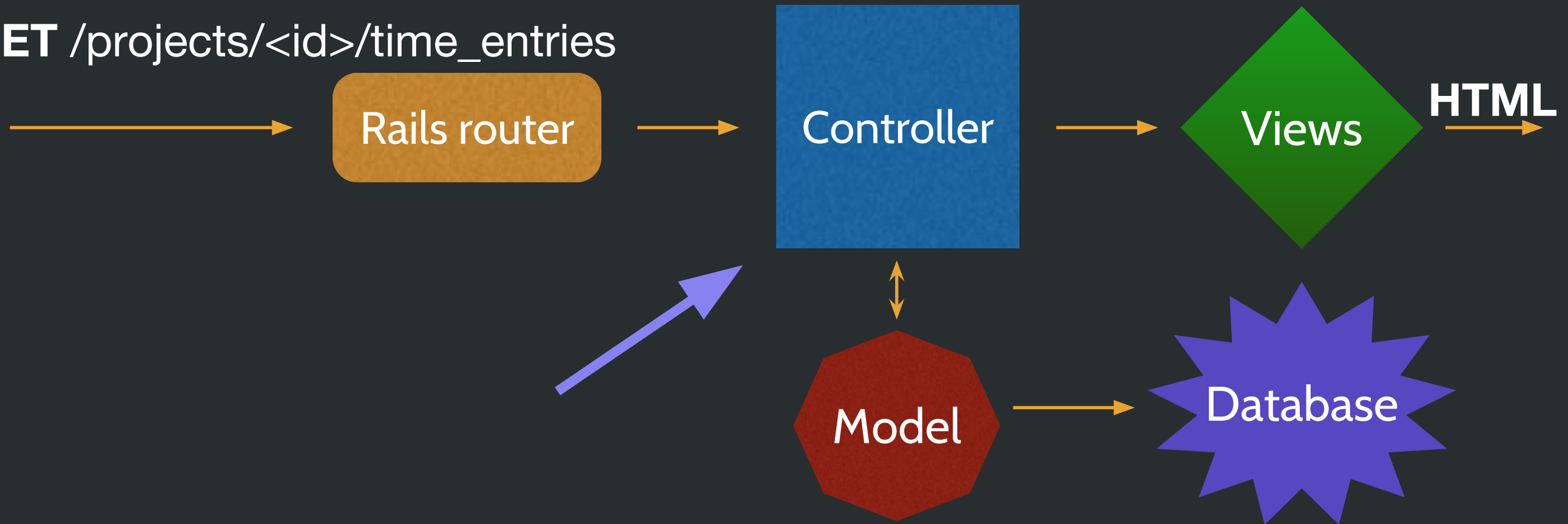
GET /projects/<id>/entries

Rails router → Controller → Views → **HTML**

Controller ↕ Model → Database

config/routes.rb

```ruby
Timetracking::Application.routes.draw do

  get '/', to: 'site#home'

  get '/projects', to: 'projects#index'

  get '/projects/:id', to: 'projects#show'

  get '/projects/:id/time_entries', to: 'time_entries#index'
end
```

Why a new controller?

- It's a good practice to leave controllers the CRUD responsibility of only one resource

- Although the path of request is in the projects resource, it's representing a nested relation

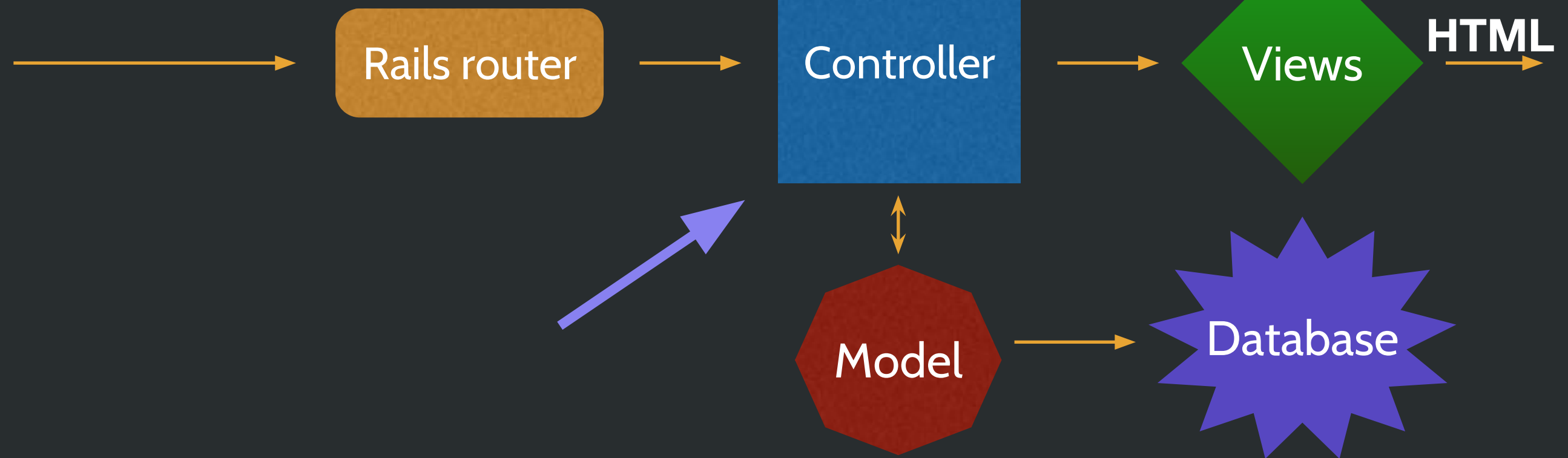IRON
HACK

GET /projects/<id>/time_entries

Rails router → Controller → Views → **HTML**

Controller ↕ Model → Database

$ rails g controller time_entries

app/controllers/time_entries_controller.rb

IRON HACK

GET /projects/<id>/time_entries

```
Rails router  →  Controller  →  Views  →  HTML
```

Model → Database

app/controllers/time_entries_controller.rb

```ruby
class TimeEntriesController < ApplicationController
  def index
  end
end
```

IRON
HACK

GET /projects/<id>/time_entries

Rails router → Controller → Views → **HTML**

Controller ↕ Model → Database

app/controllers/time_entries_controller.rb

```ruby
class TimeEntriesController < ApplicationController

  def index

    ??

  end

end
```

RON HACK
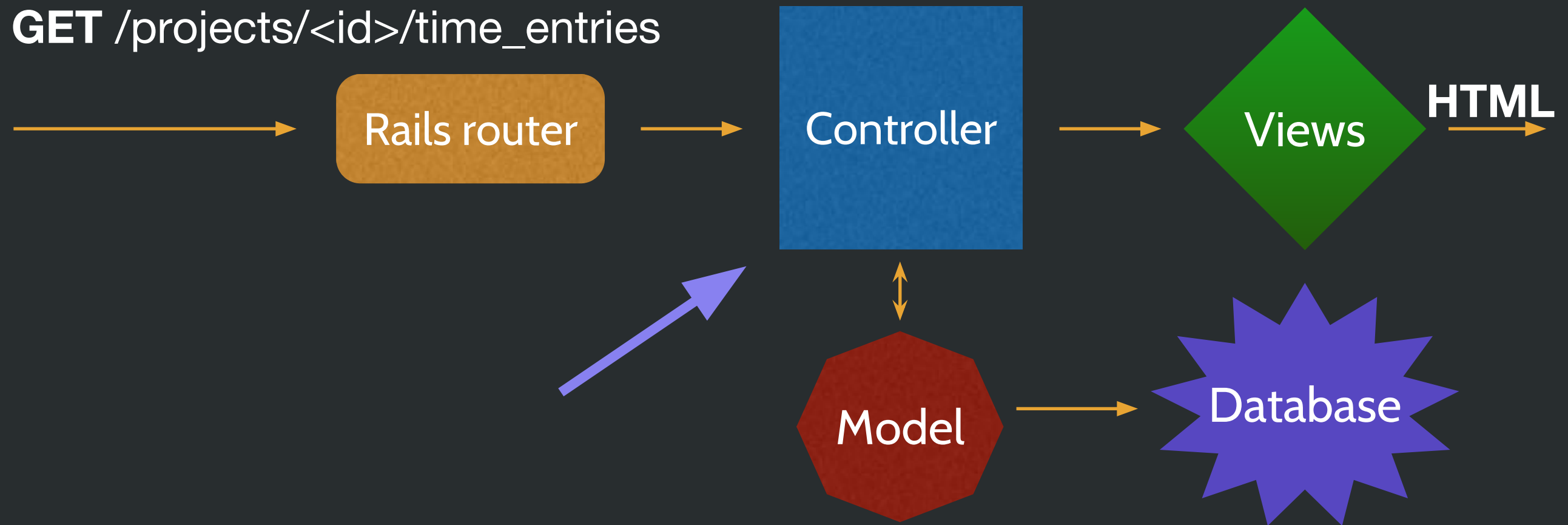
/projects/**<id>**/**time_entries**

```ruby
class TimeEntriesController < ApplicationController

  def index

    @my_project = Project.find(params[:id])

    @my_entries = @my_project.time_entries

  end

end
```

app/views/entries/index.html.erb

```erb
<h1><%= @my_project.name %></h1>
<ul>

  <% @my_entries.each do |entry| %>

    <li>

      <%= entry.hours %> h

      <%= entry.minutes %> m -

      <%= entry.date %>

    </li>

  <% end %>

</ul>
```

Time tracking tool

—

Ironhack

- 0 h 34 m - 2014-02-16
- 1 h 14 m - 2014-02-15
- 0 h 24 m - 2014-02-14

Back to Projects · New entry

Ironhack - Rails introduction

Exercise

Update the routes and the controllers to
rename the parameter of the new URL
to `:project_id`

config/routes.rb

```ruby
Rails.application.routes.draw do

  get '/', to: 'site#home'

  get '/projects', to: 'projects#index'

  get '/projects/:id', to: 'projects#show'

  get '/projects/:project_id/time_entries', to: 'time_entries#index'

end
```

app/controllers/time_entries_controller.rb

```ruby
class TimeEntriesController < ApplicationController

  def index

    @my_project = Project.find(params[:project_id])

    @my_entries = @my_project.time_entries

  end

end
```

IRON
HACK

Exercise

Update the entries page to display only the entries of this month.

Remember to commit your changes in git