Rails View Components

View whats now?

View whats now?

- Joel Hawksley @ Github

View whats now?

- Joel Hawksley @ Github
- Inspired by React

View whats now?

- Joel Hawksley @ Github
- Inspired by React
- Rails 6.1

Is this just another templating language?

Is this just another templating language?

- **NO**

Is this just another templating language?

- **NO**
- erb

Is this just another templating language?

- **NO**
- erb
- slim

Is this just another templating language?

- **NO**
- erb
- slim
- haml

Scaffold Code

```
rails new view_components_demo

rails generate scaffold Post title:string body:text
```

app/views/posts/_form.html.erb

```
<%= form_with(model: post, local: true) do |form| %>
.
.
.
  <div class="field">
    <%= form.label :body %>
    <%= form.text_area :body %>
  </div>

  <div class="actions">
    <%= form.submit %>
  </div>
<% end %>
```

app/views/posts/_form.html.erb

```
<%= form_with(model: post, local: true) do |form| %>
.
.
.
  <div class="field">
    <%= form.label :body %>
    <%= form.text_area :body %>
  </div>

  <div class="actions">
    <%= form.submit %>
  </div>
<% end %>
```

app/components/form_component.html.erb

```
<%= form_with(model: @post, local: true) do |form| %>
    .
    .
    .
      <div class="field">
        <%= form.label :body %>
        <%= form.text_area :body %>
      </div>

      <div class="actions">
        <%= form.submit %>
      </div>
    <% end %>
```

app/components/form_component.html.erb

```
<%= form_with(model: @post, local: true) do |form| %>
  .
  .
  .
   <div class="field">
    <%= form.label :body %>
    <%= form.text_area :body %>
   </div>

   <div class="actions">
    <%= form.submit %>
   </div>
  <% end %>
```

app/components/form_component.rb

```ruby
class FormComponent < ViewComponent::Base
    def initialize(post:)
        @post = post
    end
end
```

app/components/form_component.rb

```ruby
class FormComponent < ViewComponent::Base
    def initialize(post:)
        @post = post
    end
end
```

app/components/form_component.rb

```ruby
class FormComponent < ViewComponent::Base
    def initialize(post:)
        @post = post
    end
end
```

app/views/posts/new.html.erb

```erb
<h1>New Post</h1>

<%= render 'form', post: @post %>

<%= link_to 'Back', posts_path %>
```

app/views/posts/new.html.erb

```erb
<h1>New Post</h1>

<%= render 'form', post: @post %>

<%= link_to 'Back', posts_path %>
```

app/views/posts/new_with_component.html.erb

```erb
<h1>New Post</h1>

<%= render FormComponent.new(post: @post) %>

<%= link_to 'Back', posts_path %>
```

app/views/posts/new_with_component.html.erb

```erb
<h1>New Post</h1>

<%= render FormComponent.new(post: @post) %>

<%= link_to 'Back', posts_path %>
```

Easy to use, must be difficult to set up.. right?

```
#Gemfile
gem 'view_component'

#config/application.rb
require "view_component/engine"
```

Why View Components?

Why View Components?

- Performance

Why View Components?

- Performance
- Reuse

Why View Components?

- Performance
- Reuse
- Testing

Performance

```ruby
# app/controllers/posts_controller.rb
# GET /posts/new
def new
  @post = Post.new
  render 'new_with_component' if params[:use_component].present?
end
```

Performance

```ruby
# app/controllers/posts_controller.rb
# GET /posts/new
def new
  @post = Post.new
  render 'new_with_component' if params[:use_component].present?
end
```

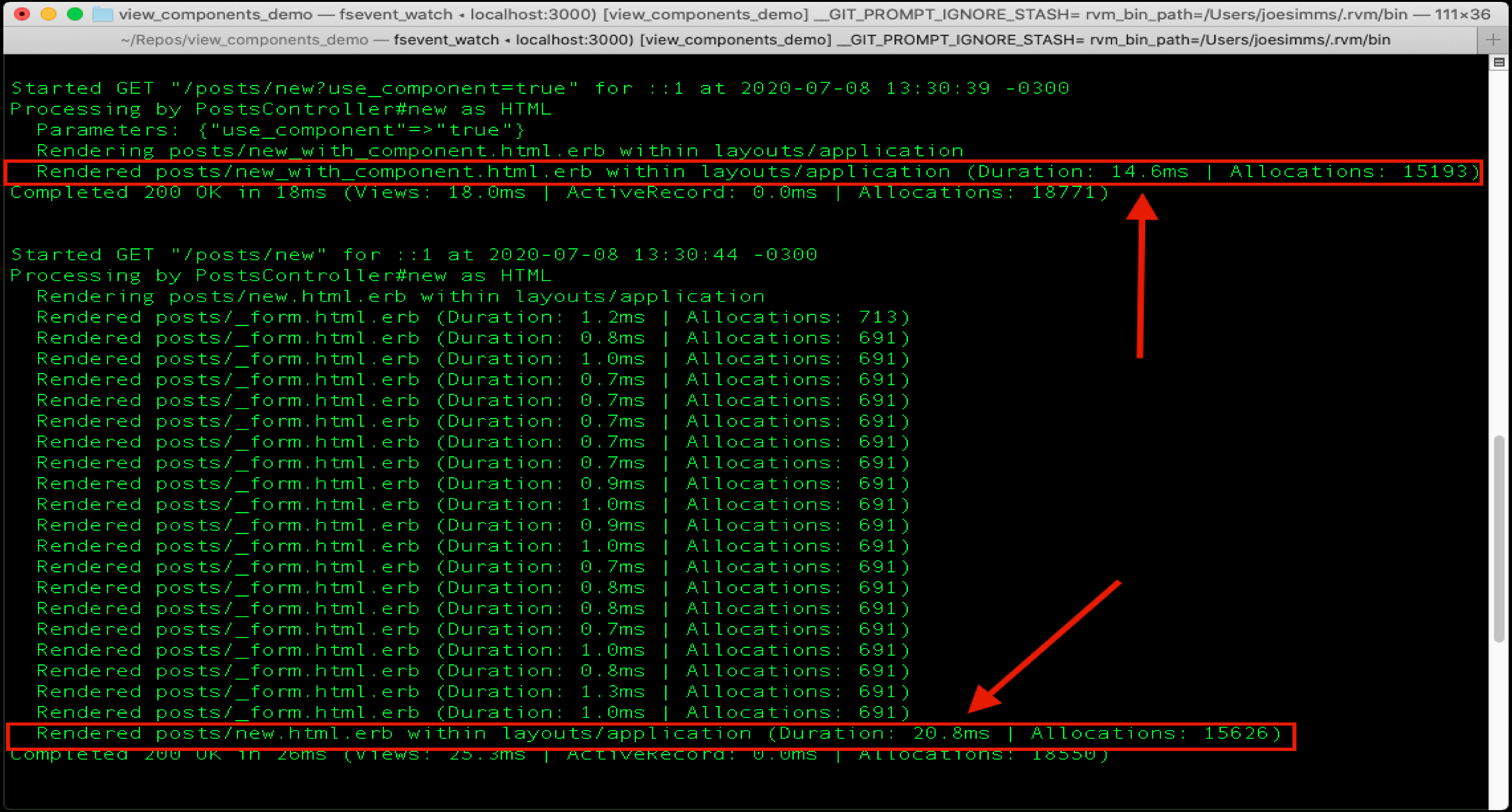
Performance

```
# app/views/posts/new.html.erb
<% 20.times do %>
    <h1>New Post</h1>

    <%= render FormComponent.new(post: @post) %>

    <%= link_to 'Back', posts_path %>
<% end %>
```

Performance


view_components_demo — fsevent_watch ◂ localhost:3000) [view_components_demo] __GIT_PROMPT_IGNORE_STASH= rvm_bin_path=/Users/joesimms/.rvm/bin — 111×36
~/Repos/view_components_demo — fsevent_watch ◂ localhost:3000) [view_components_demo] __GIT_PROMPT_IGNORE_STASH= rvm_bin_path=/Users/joesimms/.rvm/bin

```
Started GET "/posts/new?use_component=true" for ::1 at 2020-07-08 13:30:39 -0300
Processing by PostsController#new as HTML
  Parameters: {"use_component"=>"true"}
  Rendering posts/new_with_component.html.erb within layouts/application
  Rendered posts/new_with_component.html.erb within layouts/application (Duration: 14.6ms | Allocations: 15193)
Completed 200 OK in 18ms (Views: 18.0ms | ActiveRecord: 0.0ms | Allocations: 18771)


Started GET "/posts/new" for ::1 at 2020-07-08 13:30:44 -0300
Processing by PostsController#new as HTML
  Rendering posts/new.html.erb within layouts/application
  Rendered posts/_form.html.erb (Duration: 1.2ms | Allocations: 713)
  Rendered posts/_form.html.erb (Duration: 0.8ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 1.0ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.9ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 1.0ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.9ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 1.0ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.8ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.8ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.7ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 1.0ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 0.8ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 1.3ms | Allocations: 691)
  Rendered posts/_form.html.erb (Duration: 1.0ms | Allocations: 691)
  Rendered posts/new.html.erb within layouts/application (Duration: 20.8ms | Allocations: 15626)
Completed 200 OK in 26ms (Views: 25.3ms | ActiveRecord: 0.0ms | Allocations: 18550)
```

Reuse

Reuse

- Wherever you can use a partial, you can use a component.

Reuse

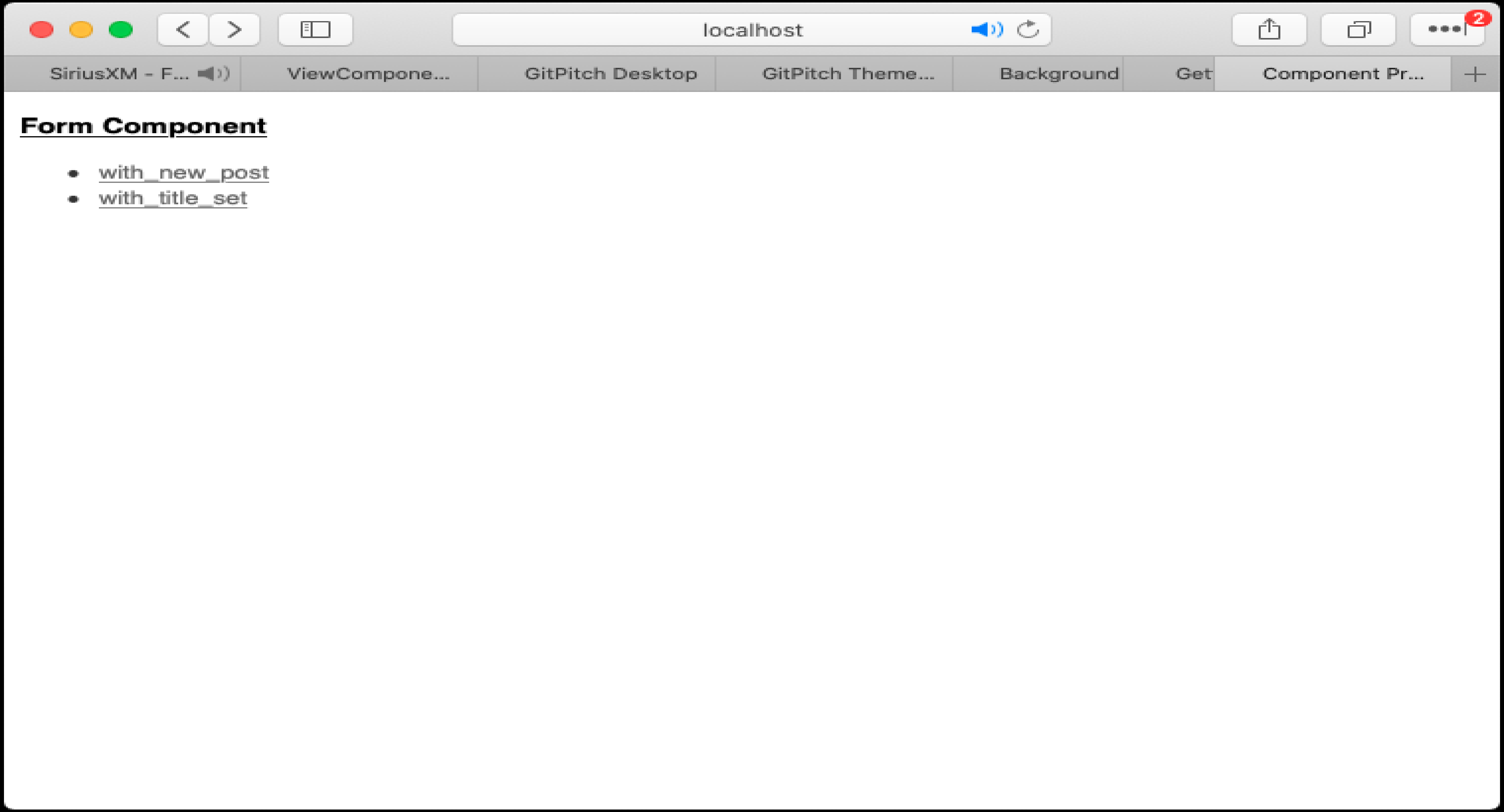
- Wherever you can use a partial, you can use a component.
- Previews make it easy to see what is already built

Previews

```ruby
#test/components/previews/form_component_preview.rb
class FormComponentPreview < ViewComponent::Preview
    def with_new_post
        render(FormComponent.new(post: Post.new))
    end

    def with_title_set
        render(FormComponent.new(post: Post.new(title: 'Title.')))
    end
end
```

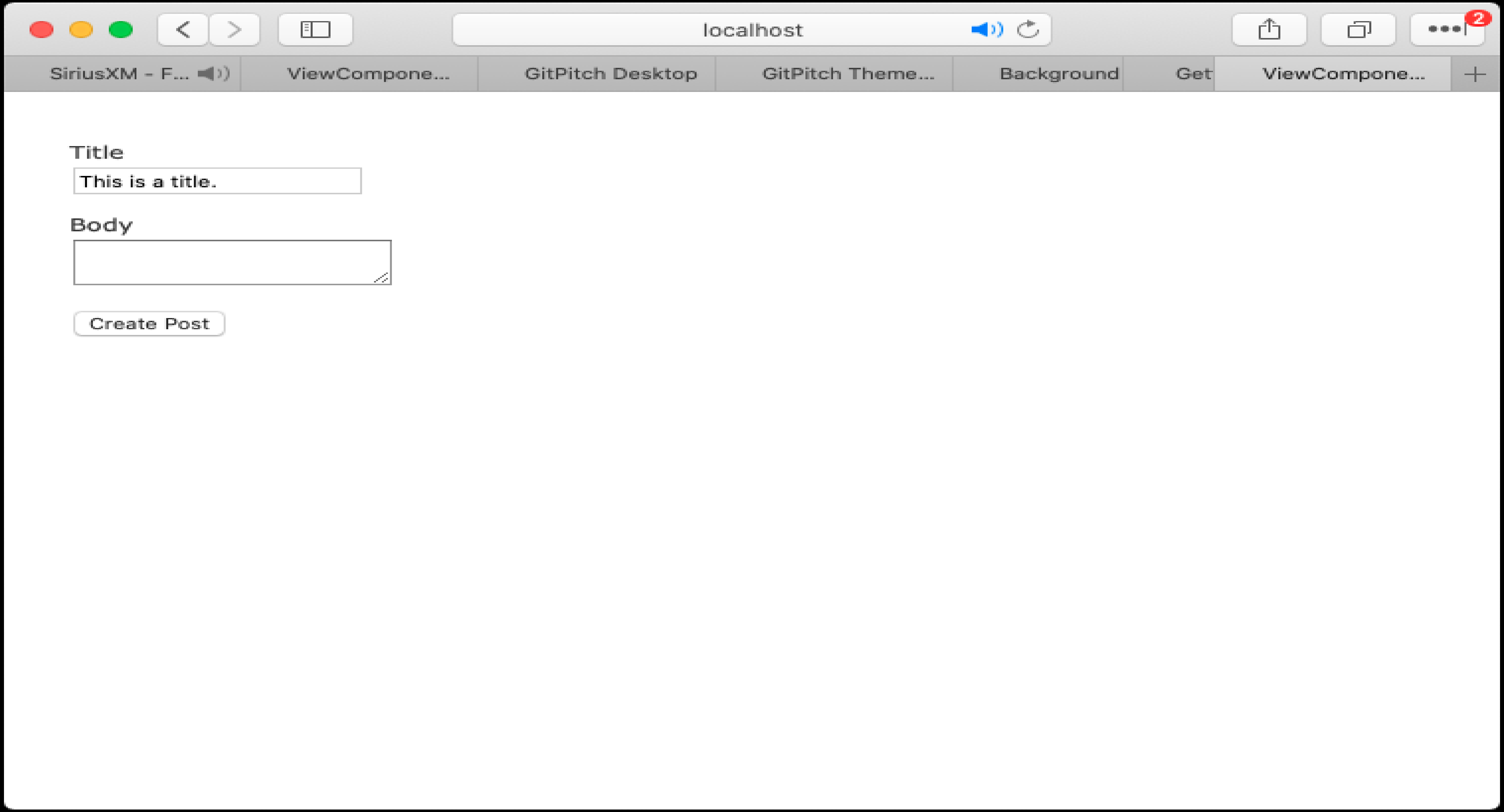
Preview listing



Form Component

- with_new_post
- with_title_set

Form Component Preview



Title

This is a title.

Body

Testing

```ruby
#test/components/form_component_test.rb
require "view_component/test_case"
class FormComponentTest < ViewComponent::TestCase
    test "render component" do
        post = Post.new(title: 'Title.')
        result = render_inline(FormComponent.new(post: post))
        assert_includes result.css("input[type='text']").to_html,
'value="Title."'
    end
end
```

Testing

```ruby
#test/components/form_component_test.rb
require "view_component/test_case"
class FormComponentTest < ViewComponent::TestCase
    test "render component" do
        post = Post.new(title: 'Title.')
        result = render_inline(FormComponent.new(post: post))
        assert_includes result.css("input[type='text']").to_html,
'value="Title."'
    end
end
```

Testing

```
#test/components/form_component_test.rb
require "view_component/test_case"
class FormComponentTest < ViewComponent::TestCase
    test "render component" do
        post = Post.new(title: 'Title.')
        result = render_inline(FormComponent.new(post: post))
        assert_includes result.css("input[type='text']").to_html,
'value="Title."'
    end
end
```

Testing

```ruby
#test/components/form_component_test.rb
require "view_component/test_case"
class FormComponentTest < ViewComponent::TestCase
    test "render component" do
        post = Post.new(title: 'Title.')
        result = render_inline(FormComponent.new(post: post))
        assert_includes result.css("input[type='text']").to_html,
'value="Title."'
    end
end
```

Links

Joel Hawksley

github/view_component

Rails 6.1 Component Documentation

This Code and Presentation

GitPitch

Questions?

joe@joesim.ms