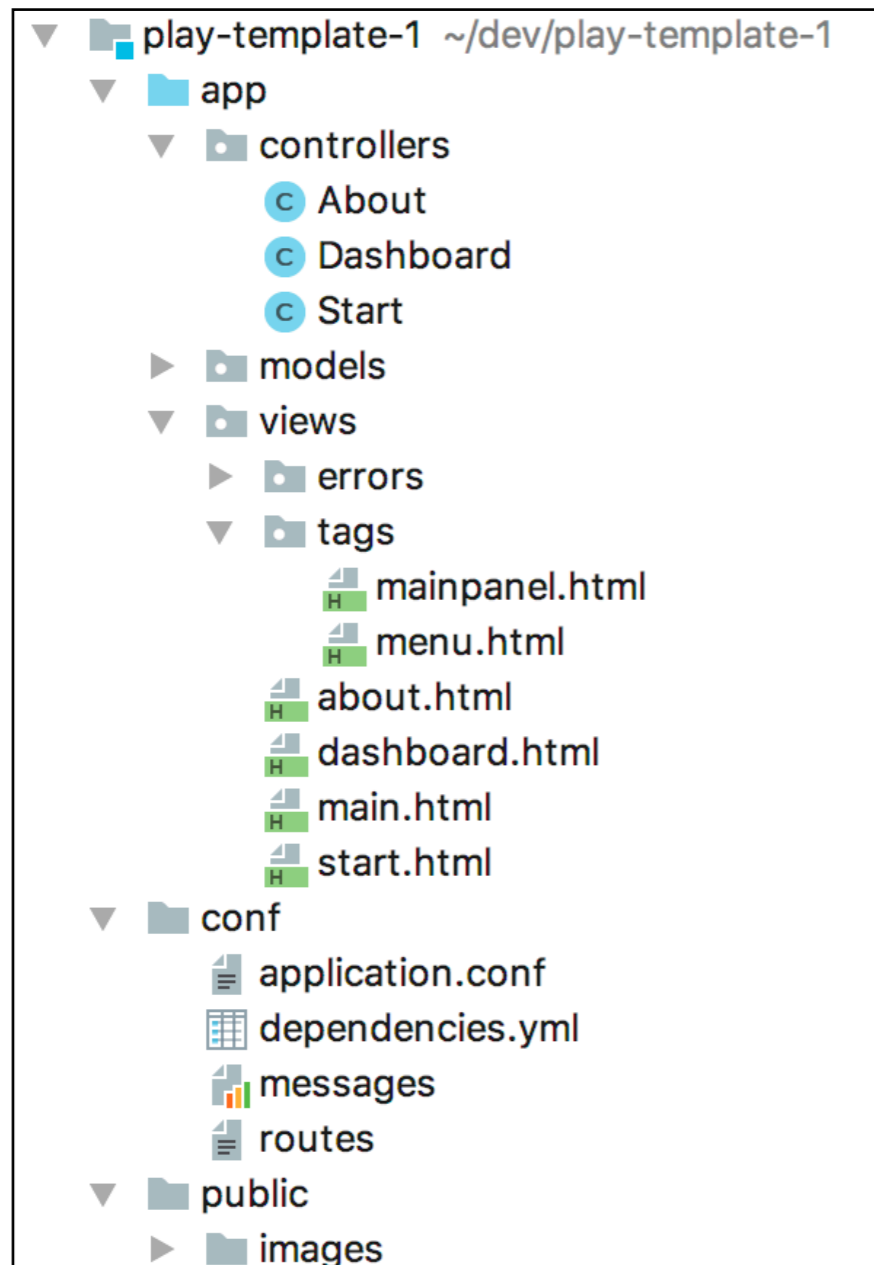# Back-end

Application Structure

- App implements Routes + Model/View/Controller Architecture

- Back-end + Front-end collaborate to support structured, predictable application workflow

Back-end


```
▼ 📁 app
  ▼ 📁 controllers
      © About
      © Dashboard
      © Start
  ▶ 📁 models
  ▼ 📁 conf
      📄 application.conf
      📊 dependencies.yml
      📊 messages
      📄 routes
```
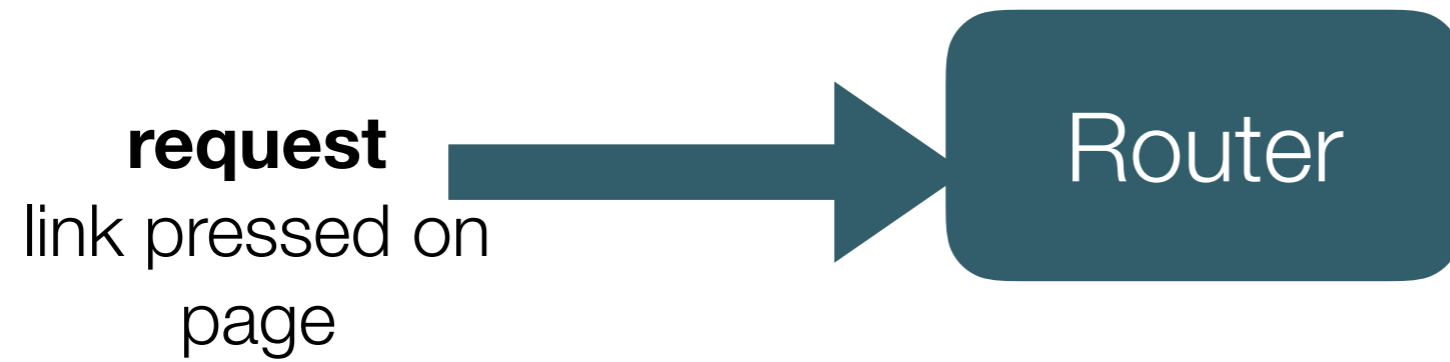
- All written in Java

- Consists of:

  - **Routes** - supported urls

  - **Controllers** - objects to handle the routes

  - **Config** - system wide config of application
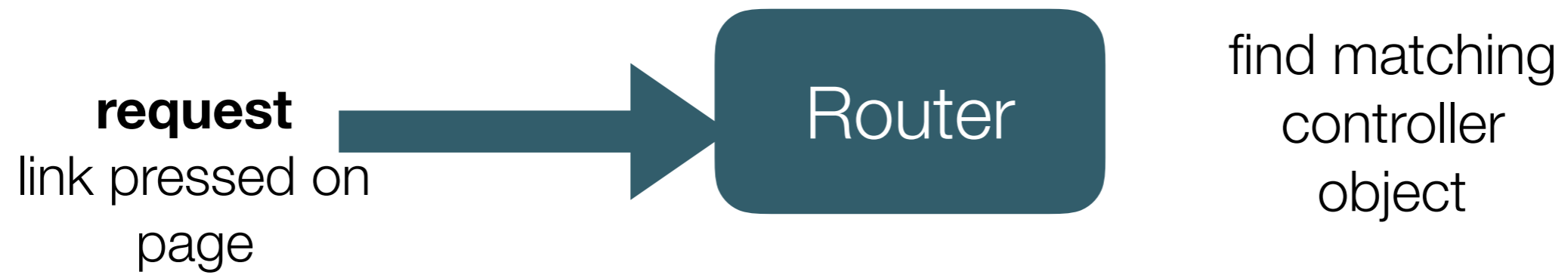
- Will include **Models** later…
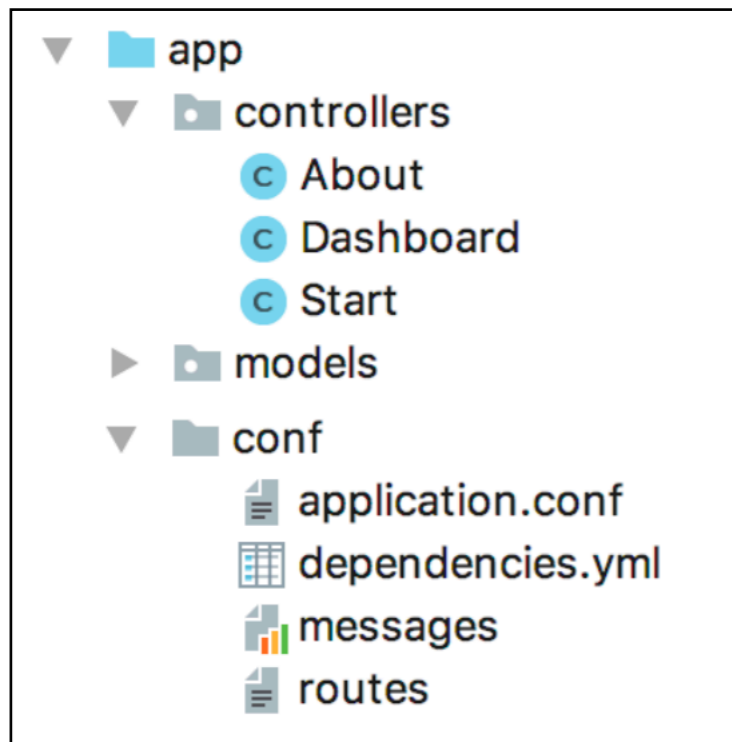
Request/Response Lifecycle

1. **Request** - link pressed on page

2. **Router** - find matching controller object

3. **Controller** - method invoked to handle request

4. **View** - data sent from controller to view to construct response

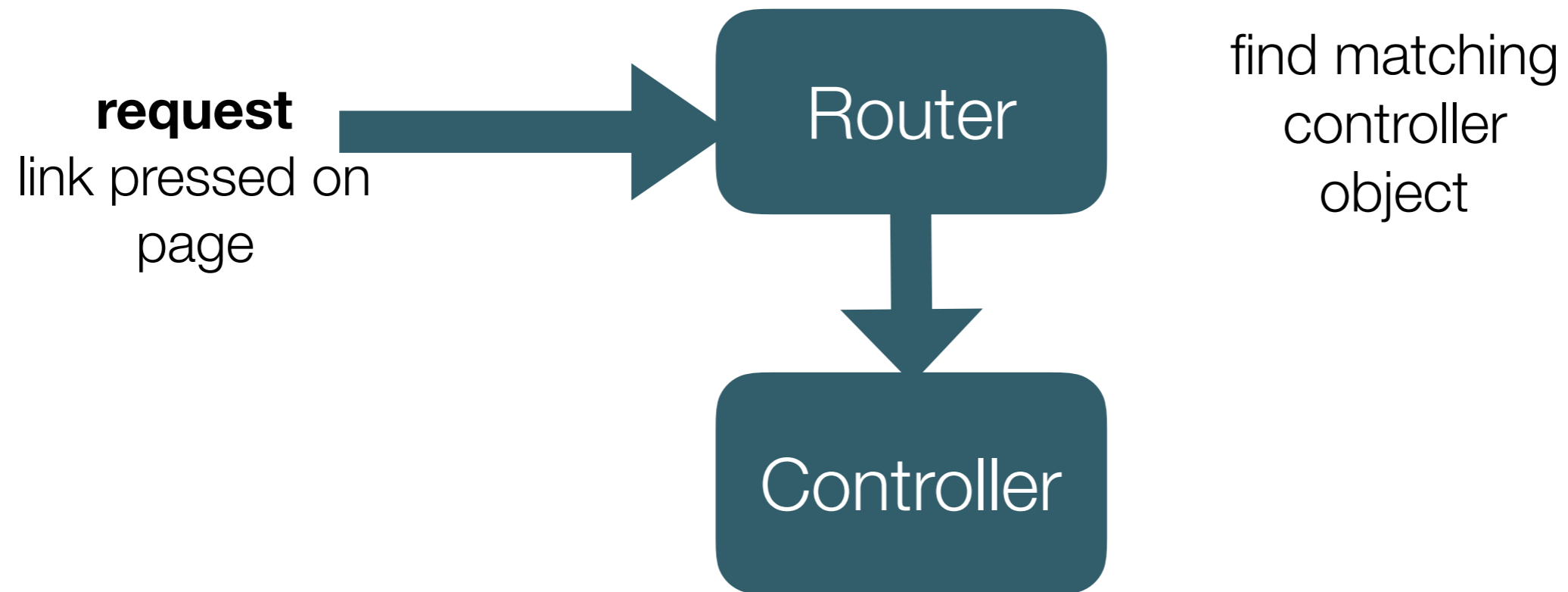5. **Response** - complete page rendered into browser

Router/Controller/View

request
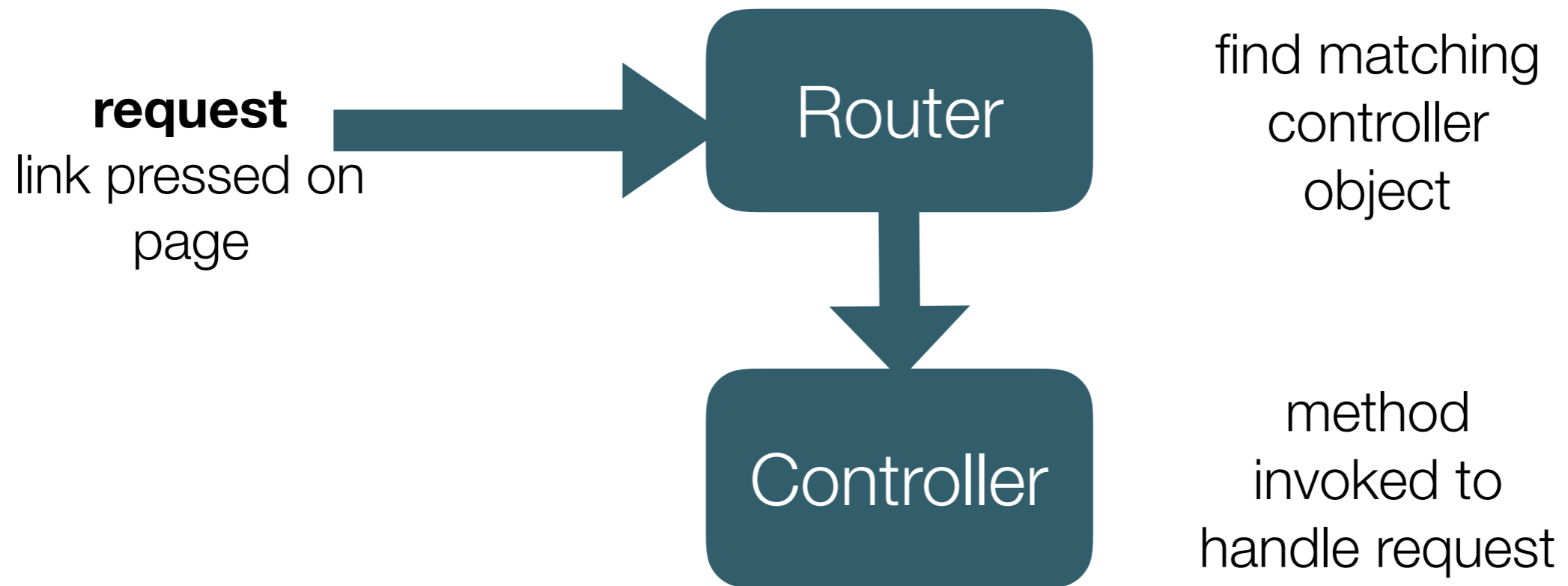link pressed on
page

Router

Router/Controller/View

request
link pressed on
page

Router

find matching
controller
object

Router/Controller/View

request
link pressed on page

Router

find matching controller object

Controller

Router/Controller/View

request
link pressed on page

Router

find matching controller object

Controller

method invoked to handle request

Router/Controller/View

request
link pressed on
page

Router

find matching
controller
object

Controller

method
invoked to
handle request

View

Router/Controller/View

request
link pressed on
page

Router

find matching
controller
object

Controller

method
invoked to
handle request

View

data sent from
controller to view
to construct
response

Router/Controller/View

request
link pressed on
page

Router

find matching
controller
object

Controller

method
invoked to
handle request

View

data sent from
controller to view
to construct
response

response
complete page
rendered into
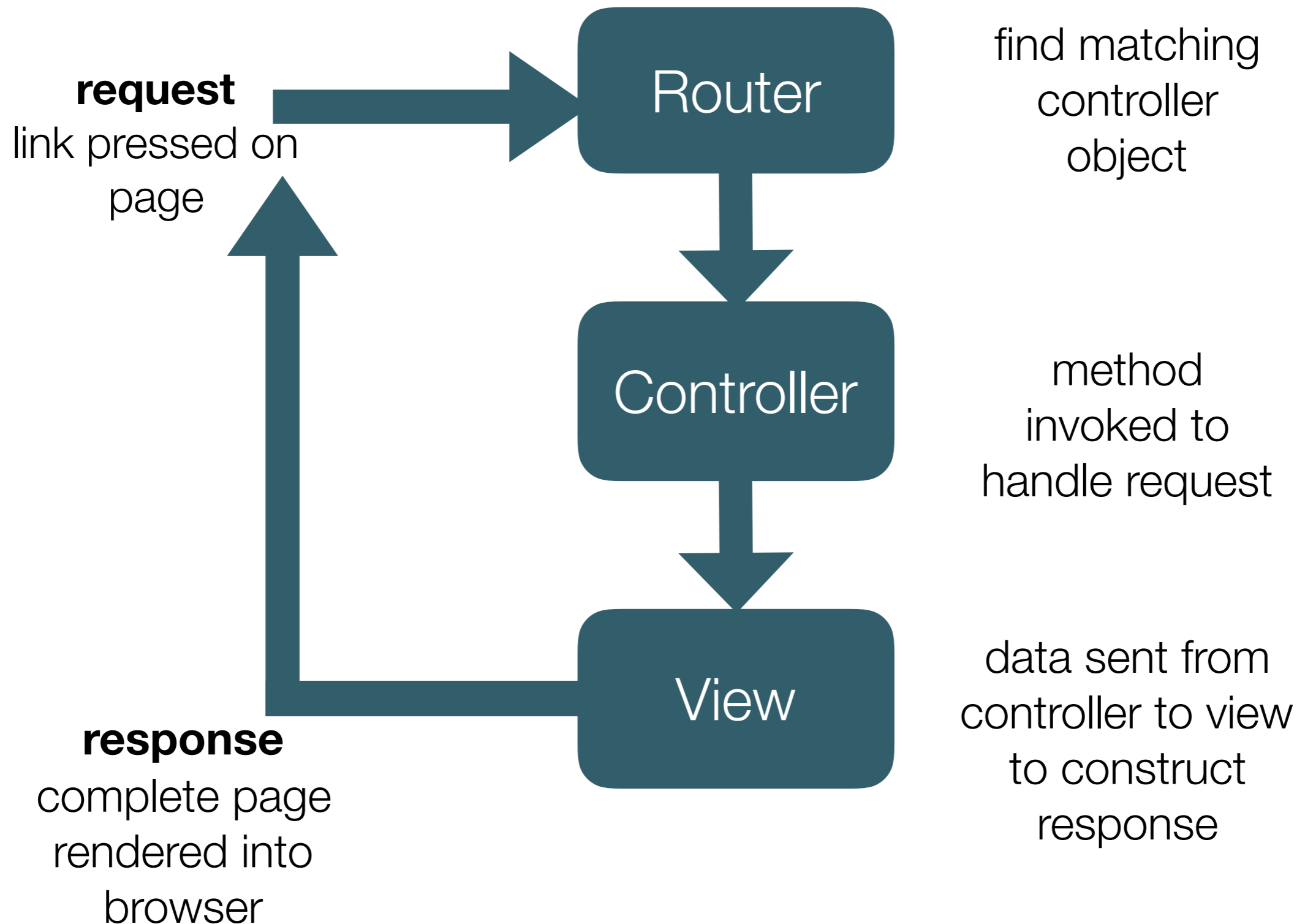browser

Request - link pressed on page
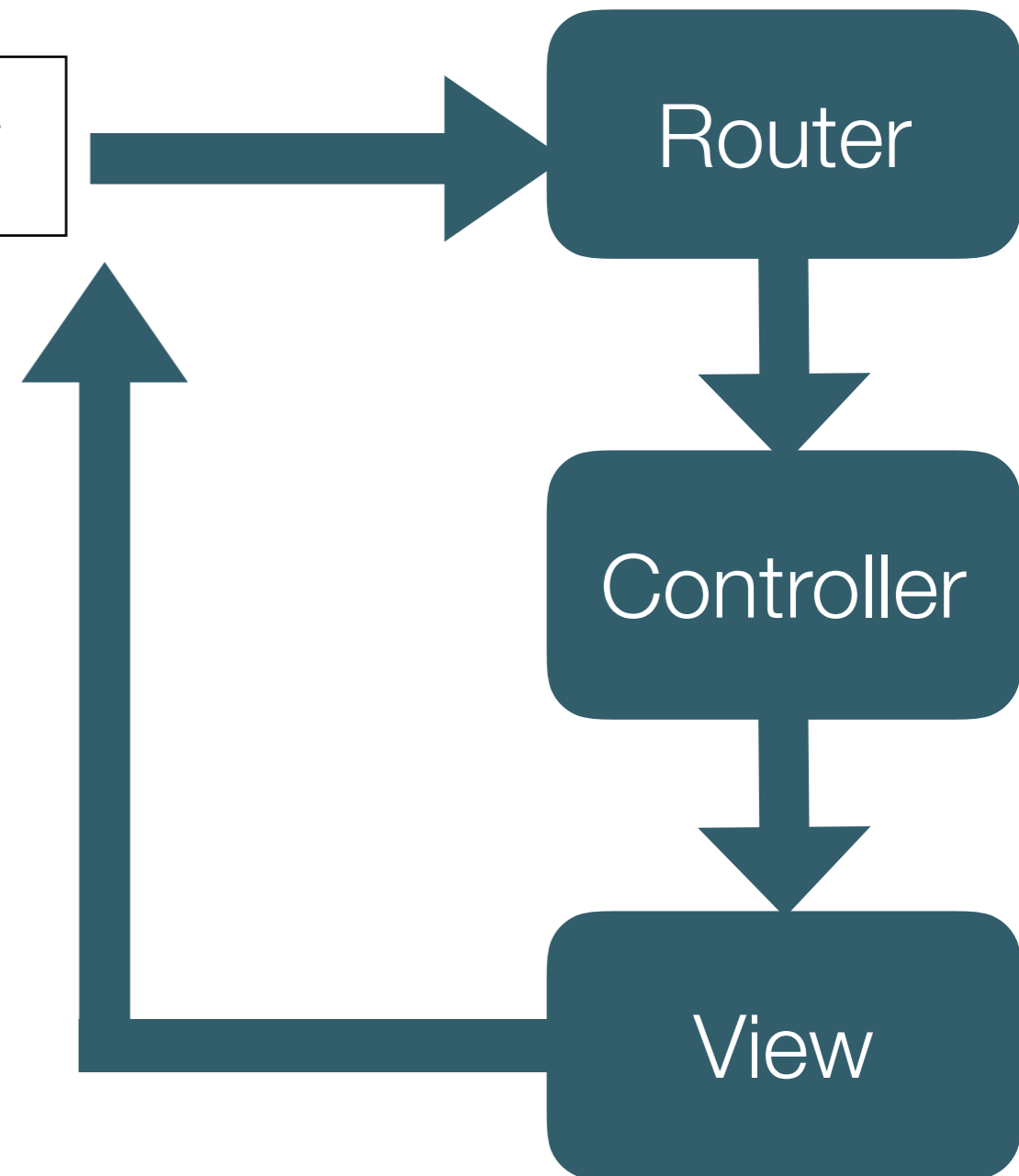
```
...
    <a id="dashboard" class="item" href="/dashboard"> Dashboard  </a>
    <a id="about" class="item" href="/about"> About </a>
...
```
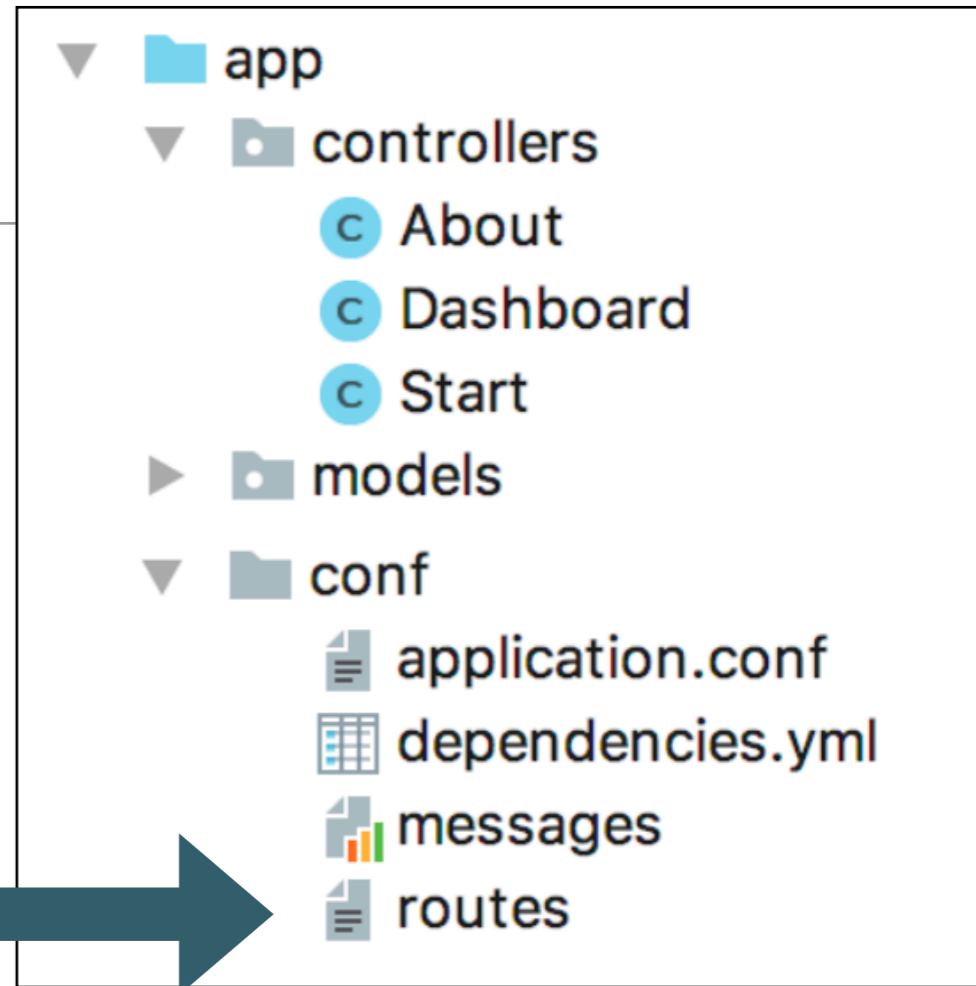
- Requests defined in links in views:

  - href in <a> tags

  - href in Menus

  - href in Buttons

  - action links in forms

Router

Controller

View

Router - find matching controller object

```
▼  📁 app
   ▼  📂 controllers
      ⓒ About
      ⓒ Dashboard
      ⓒ Start
   ▶  📂 models
   ▼  📁 conf
      📄 application.conf
      📊 dependencies.yml
      📊 messages
  ➡️ 📄 routes
```

routes

Mach the link patterns with the controler.method

```
...
GET      /                              Start.index
GET      /dashboard                     Dashboard.index
GET      /about                         About.index
...
```
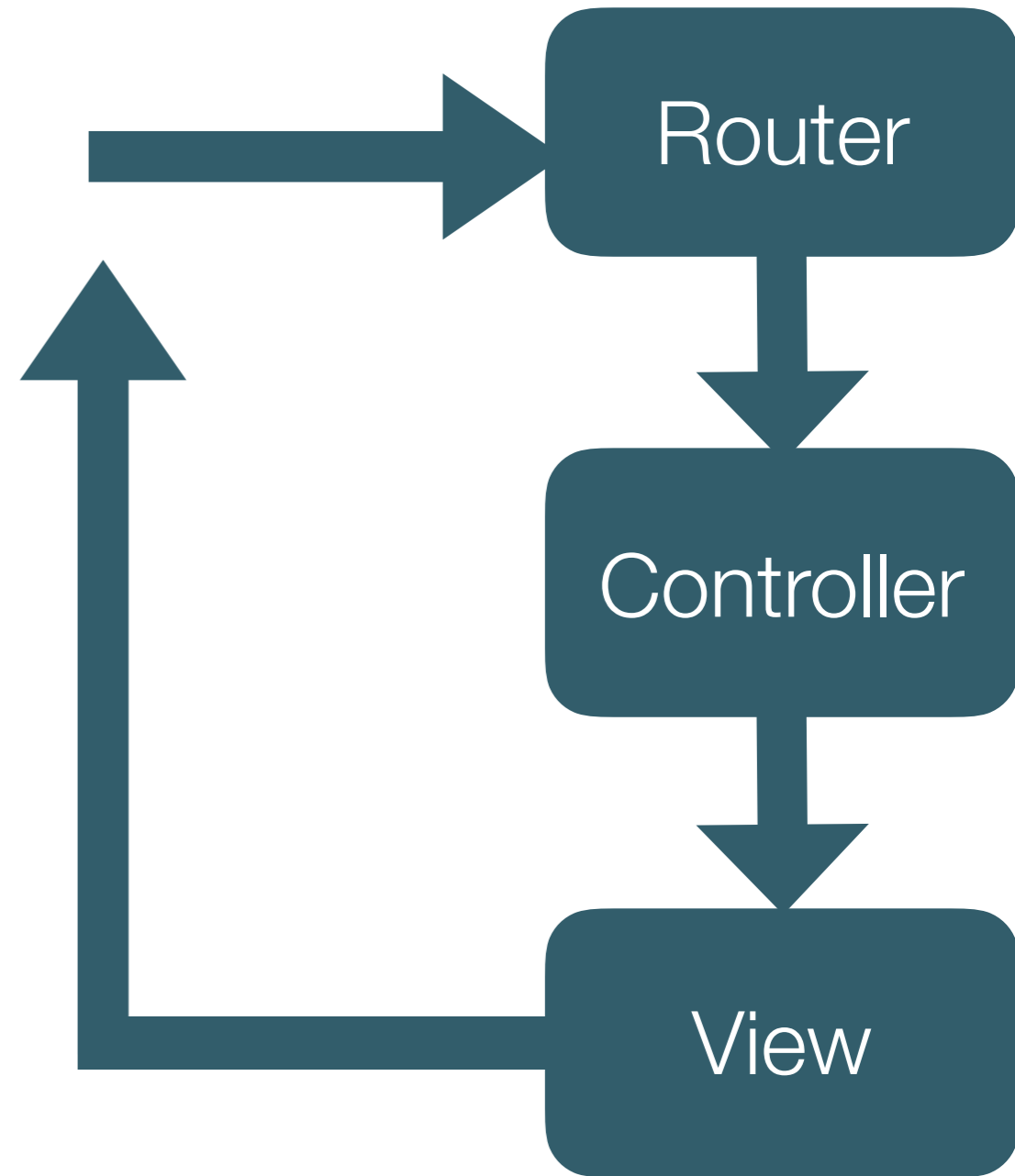
Router Behaviour

If user selects these links…

…then call the corresponding controller methods

`'/'` ⟶ `Start.index`

`/dashboard'` ⟶ `Dashboard.index`

`/about'` ⟶ `About.index`

Router

Controller

View

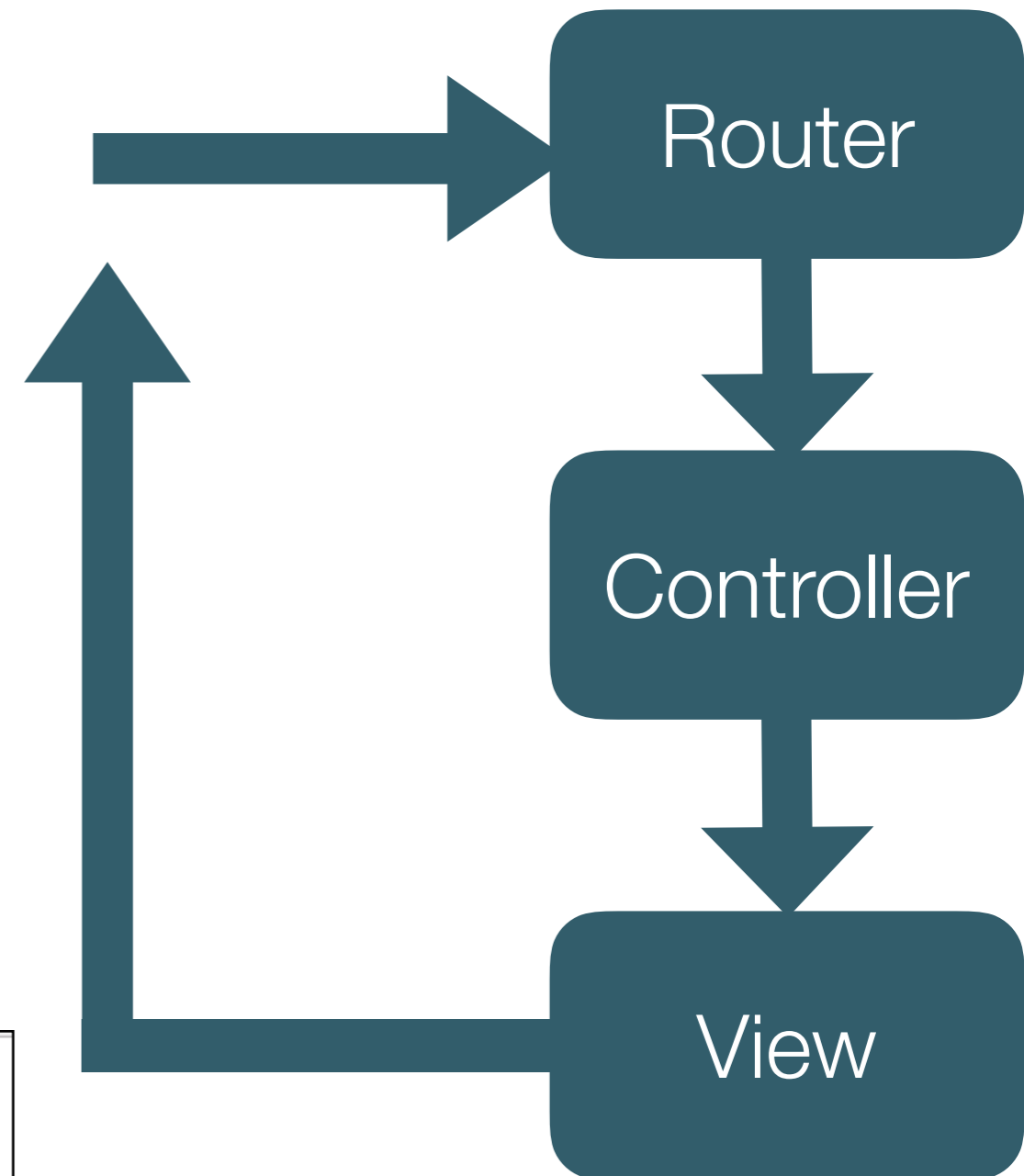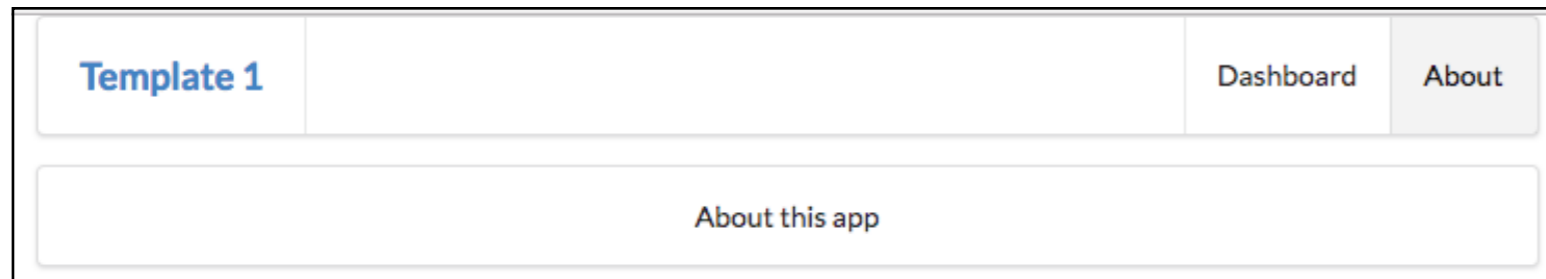# **Controller** method invoked to handle request

About.java

```java
package controllers;

import play.*;
import play.mvc.*;
import java.util.*;
import models.*;

public class About extends Controller
{
  public static void index() {
    Logger.info("Rendering about");
    render ("about.html");
  }
}
```
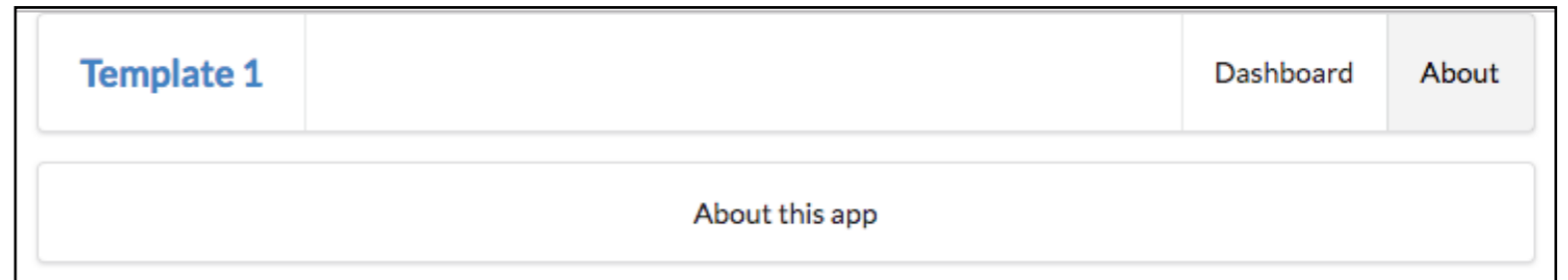
The About controller

Template 1		Dashboard	About
	About this app		

Router

Controller

View

The 'About' controller index *action*

```java
public static void index() {
    Logger.info("Rendering about");
    render ("about.html");
}
```

```
12:28:42,041 INFO  ~ Application  playlist  is no
12:28:42,127 INFO  ~ Rendering Dashboard
12:28:44,618 INFO  ~ Rendering about
12:28:45,193 INFO  ~ Rendering Dashboard
12:28:46,233 INFO  ~ Rendering Start
12:28:47,233 INFO  ~ Rendering Dashboard
12:29:18,611 INFO  ~ Rendering Dashboard
12:29:46,311 INFO  ~ Rendering about
12:29:47,111 INFO  ~ Rendering Dashboard
12:29:48,199 INFO  ~ Rendering Start
```
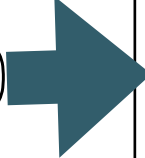
Template 1 | Dashboard | About

About this app

Render the 'about' view

Log a message
to the console

Back-end + Front-End

about.js

```
public static void index() {
    Logger.info("Rendering about");
    render ("about.html");
}
```

about.html

```
#{extends 'main.html' /}
#{set title:'About' /}

#{menu id:"about"/}

<section class="ui center aligned middle aligned segment">
  <p>
    About this app
  </p>
</section>
```

menu.html
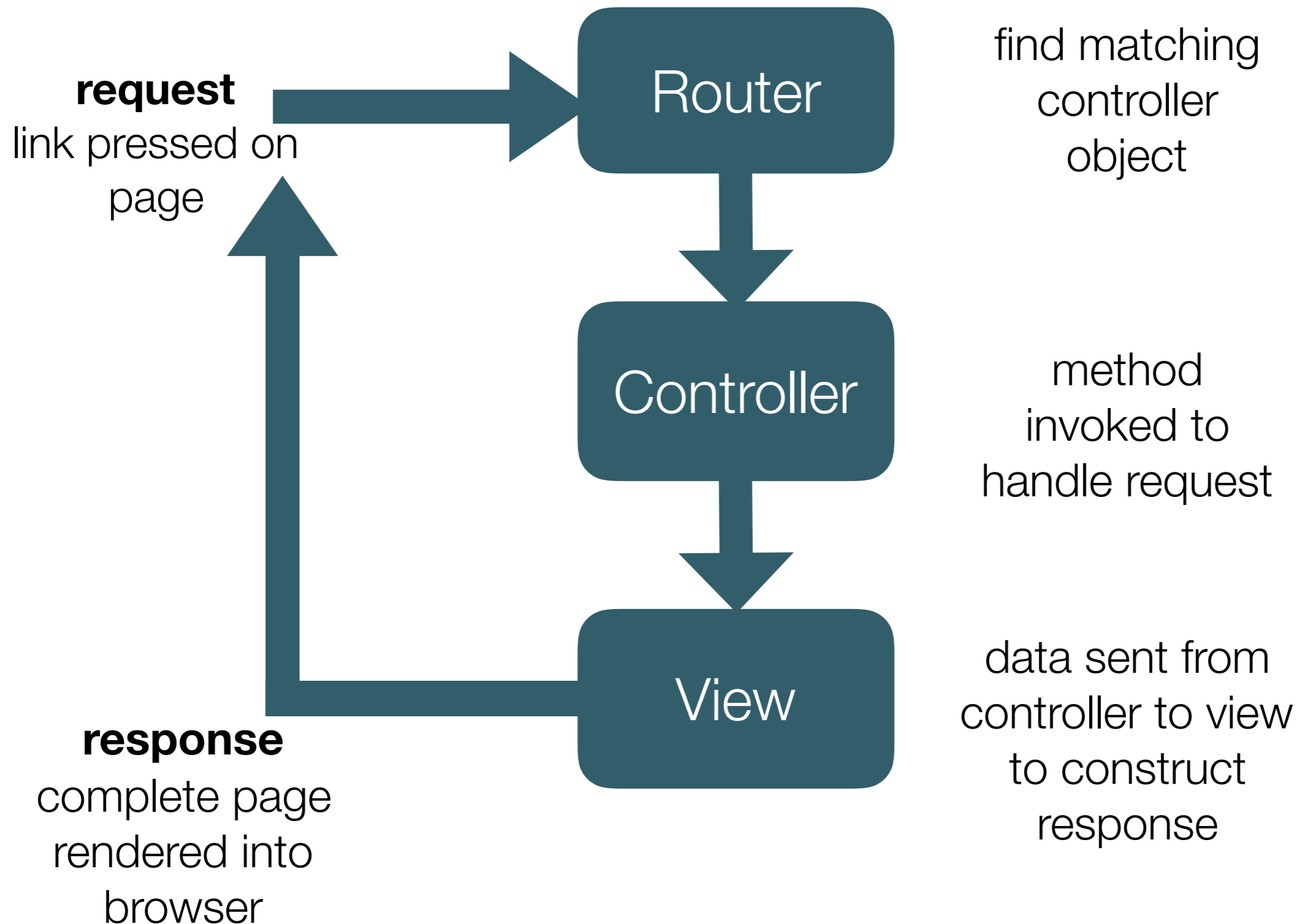
```
<nav class="ui menu">
  <header class="ui header item"> <a href="/"> Template 1 </a></header>
  <div class="right menu">
    <a id="dashboard" class="item" href="/dashboard"> Dashboard  </a>
    <a id="about" class="item" href="/about"> About </a>
  </div>
</nav>

<script>
  $("#${_id}").addClass("active item");
</script>
```

main.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>#{get 'title' /}</title>
    <script type="text/javascript" src=
    <script type="text/javascript" src=
    <link type="text/css"  href="https:
  </head>
  <body>
    <section class="ui container">
      #{doLayout /}
    </section>
  </body>
```

Router/Controller/View

request
link pressed on page

Router

find matching controller object

Controller

method invoked to handle request

View

data sent from controller to view to construct response

response
complete page rendered into browser

Dashboard Controller

```java
package controllers;

import play.Logger;
import play.mvc.Controller;

public class Dashboard extends Controller
{
  public static void index() {
    Logger.info("Rendering Dashboard");
    render ("dashboard.html");
  }
}
```

Start Controller

```java
package controllers;

import play.Logger;
import play.mvc.Controller;

public class Start extends Controller
{
  public static void index() {
    Logger.info("Rendering Start");
    render ("start.html");
  }
}
```
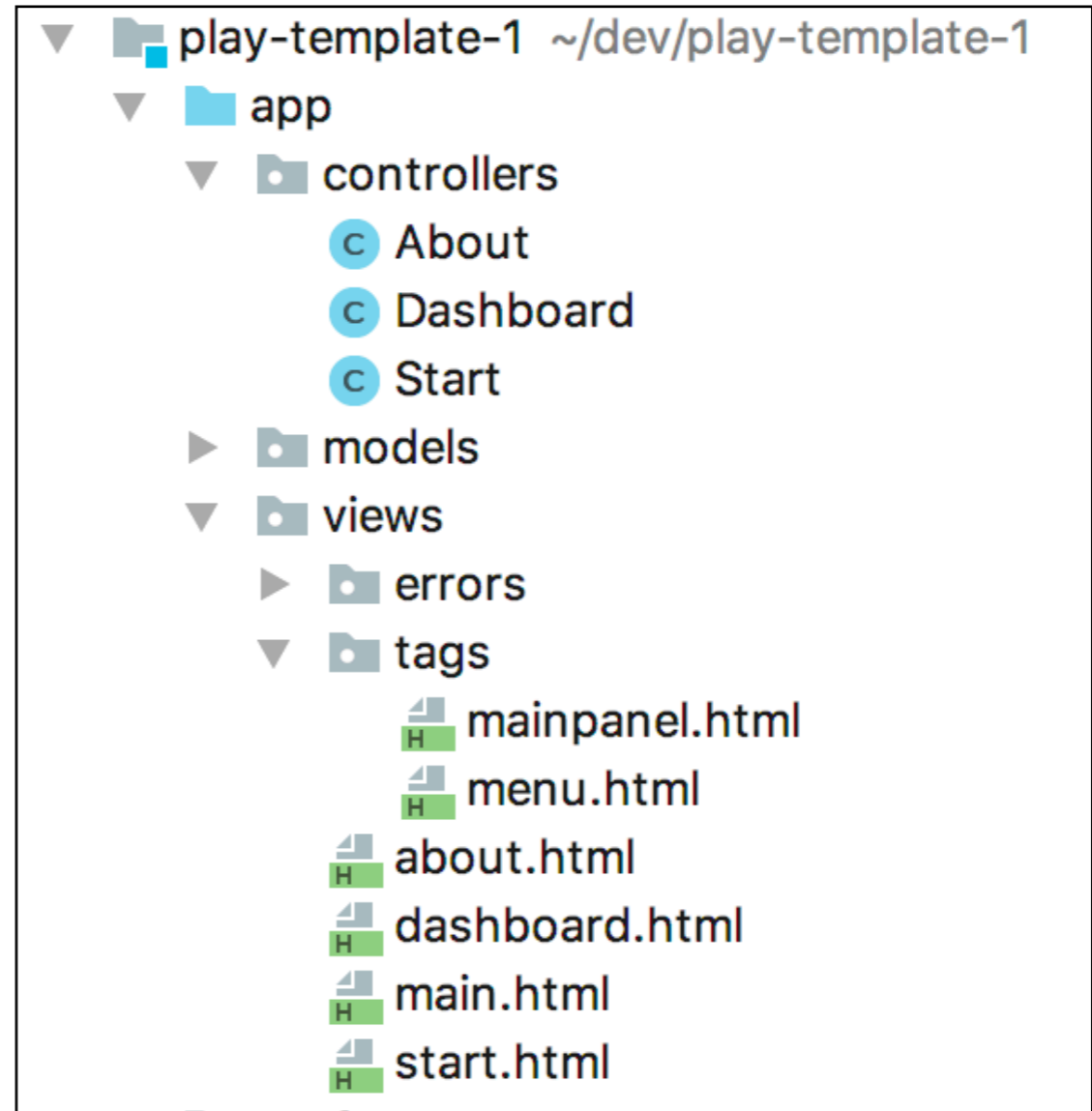
Application Structure

3 Controllers

which will render

3 matching views

routes/controllers

Start.java

```java
package controllers;

…

public class About extends Controller
{
  public static void index() {
    Logger.info("Rendering about");
    render ("about.html");
  }
}
```
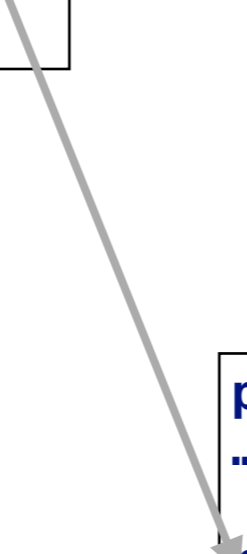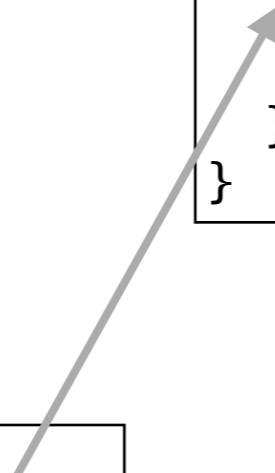
Dashboard.java

```java
package controllers;

…

public class Dashboard extends Controller
{
  public static void index() {
    Logger.info("Rendering Dashboard");
    render ("dashboard.html");
  }
}
```

routes.js

```
...
GET      /                   Start.index
GET      /dashboard          Dashboard.index
GET      /about              About.index
...
```

About.java

```java
package controllers;

…

public class Start extends Controller
{
  public static void index() {
    Logger.info("Rendering Start");
    render ("start.html");
  }
}
```

controllers/views

start.html

```java
package controllers;
…

public class About extends Controller
{
  public static void index() {
    Logger.info("Rendering about");
    render ("about.html");
  }
}
```

```html
#{extends 'main.html' /}
#{set title:'About' /}

#{menu id:"start"/}

<section class="ui center aligned middle aligned segment">
  <h1 class="ui header">
    Welcome to Template 1
  </h1>
  <p>
    To be replaced with content...
  </p>
</section>
```

dashboard.html

```java
package controllers;
…

public class Dashboard extends Controller
{
  public static void index() {
    Logger.info("Rendering Dashboard");
    render ("dashboard.html");
  }
}
```

```html
#{extends 'main.html' /}
#{set title:'Dashboard' /}

#{menu id:"dashboard"/}

<section class="ui segment">
  #{mainpanel/}
</section>
```

about.html

```java
package controllers;
…

public class Start extends Controller
{
  public static void index() {
    Logger.info("Rendering Start");
    render ("start.html");
  }
}
```

```html
#{extends 'main.html' /}
#{set title:'About' /}

#{menu id:"about"/}

<section class="ui center aligned middle aligned segment">
  <p>
    About this app
  </p>
</section>
```