

Bài giảng

PHÁT TRIỂN ỨNG DỤNG WEB

Lê Đình Thanh

Khoa Công nghệ Thông tin

Trường Đại học Công nghệ, ĐHQGHN

E-mail: thanhld@vnu.edu.vn Mobile: 0987.257.504

Chương 9

Viết lại và định tuyến URL

Nội dung

- Viết lại URL
- Định tuyến URL
- REST

W3C standard, non-semantic URL

- URL hay Định vị tài nguyên đồng nhất dùng để xác định vị trí (toàn cầu) của một tài nguyên trên Internet
`scheme://host:port/path?query_string#bookmark`
- Ví dụ `http://uet.vnu.edu.vn:8080/daotao/monhoc.py`



Semantic URL

- Tạo liên kết
 - gọn gàng, dễ nhớ, thân thiện với người dùng
 - thân thiện với các search engines
 - Che được chuỗi truy vấn đối với người dùng
 - Cho site vẫn sử dụng được URL khi thay đổi công nghệ phía dưới

TRY IT YOURSELF

<http://www.tienphong.vn/xa-hoi/phong-su/602827/Vuot-thoat-mot-Con-Dao-chinh-minh.html>

<http://www.tienphong.vn/xa-hoi/phong-su/602827/misc.html>

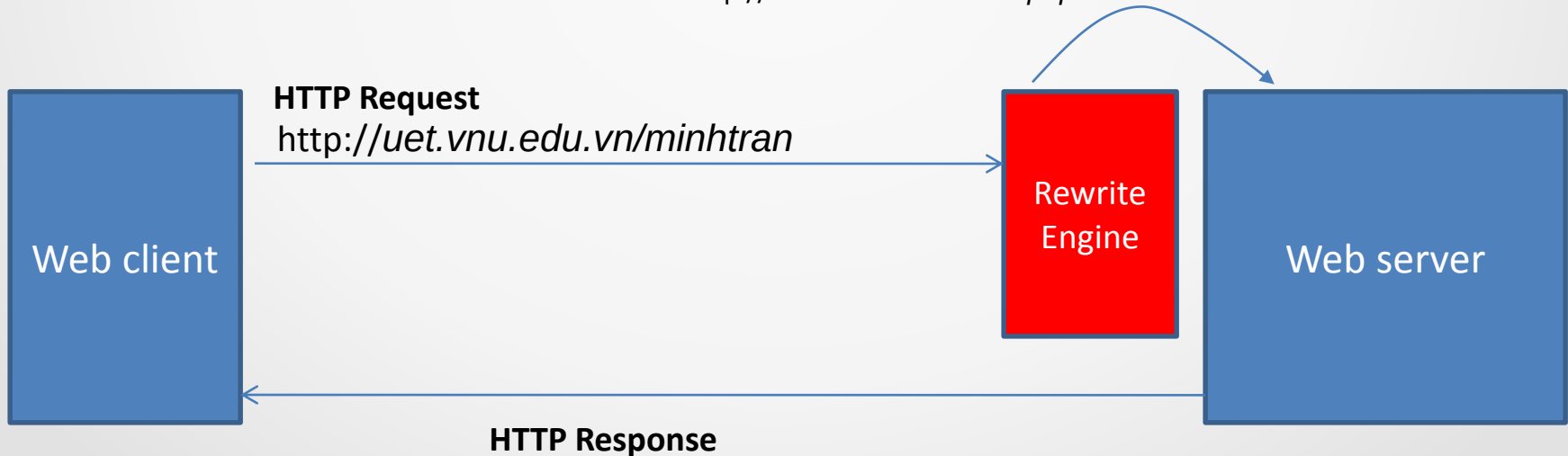
<http://www.tienphong.vn/xa-hoi/phong-su/602827/does-not-matter.html>

<http://www.tienphong.vn/xa-hoi/phong-su/602827/whatever-you-like.html>

Viết lại URL là gì?

- Viết lại URL (URL Rewrite) là sửa đổi URL do web client yêu cầu trước khi web server phục vụ
- Rewrite Engine thực hiện viết lại URL
 - Apache: mod_rewrite

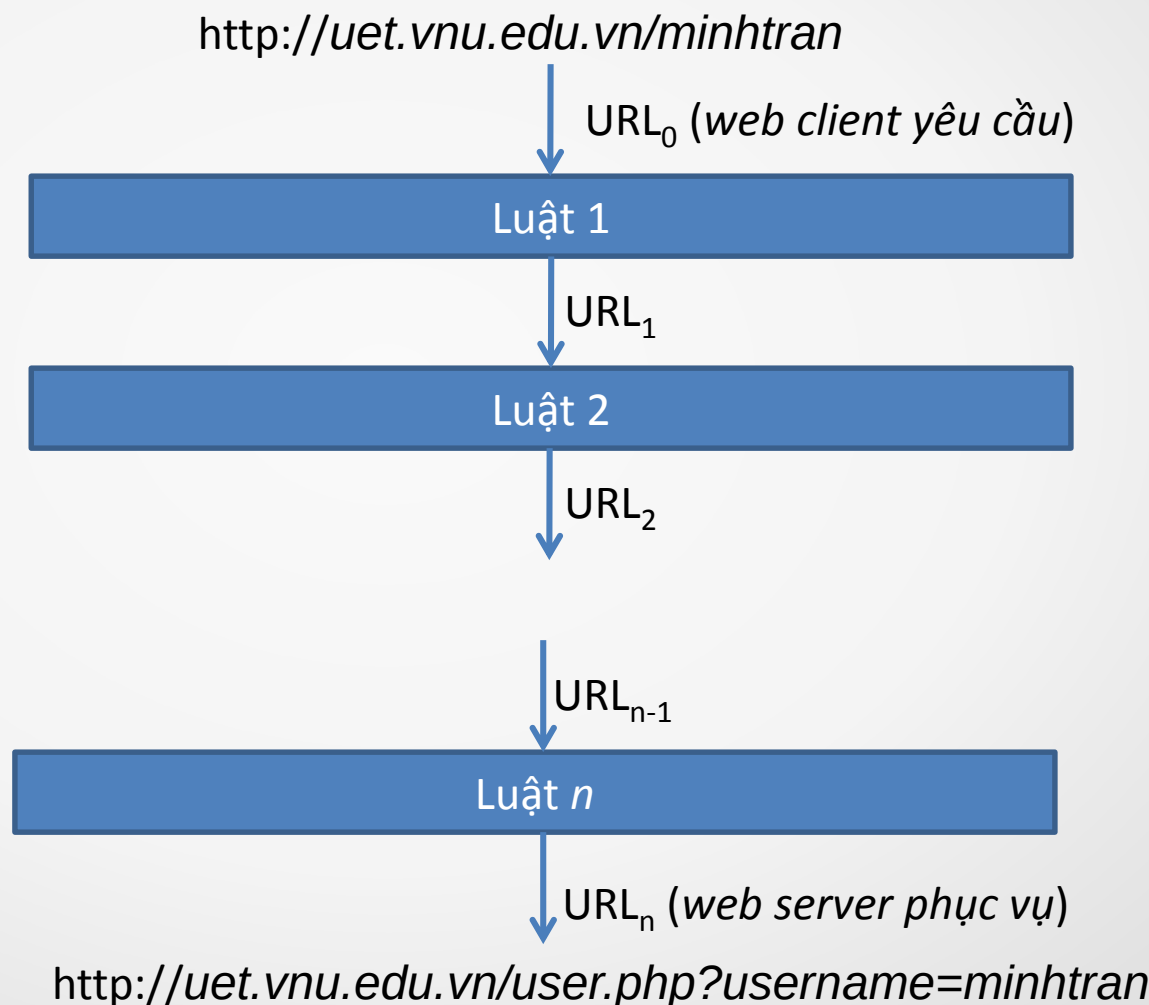
http://uet.vnu.edu.vn/user.php?username=minhtran



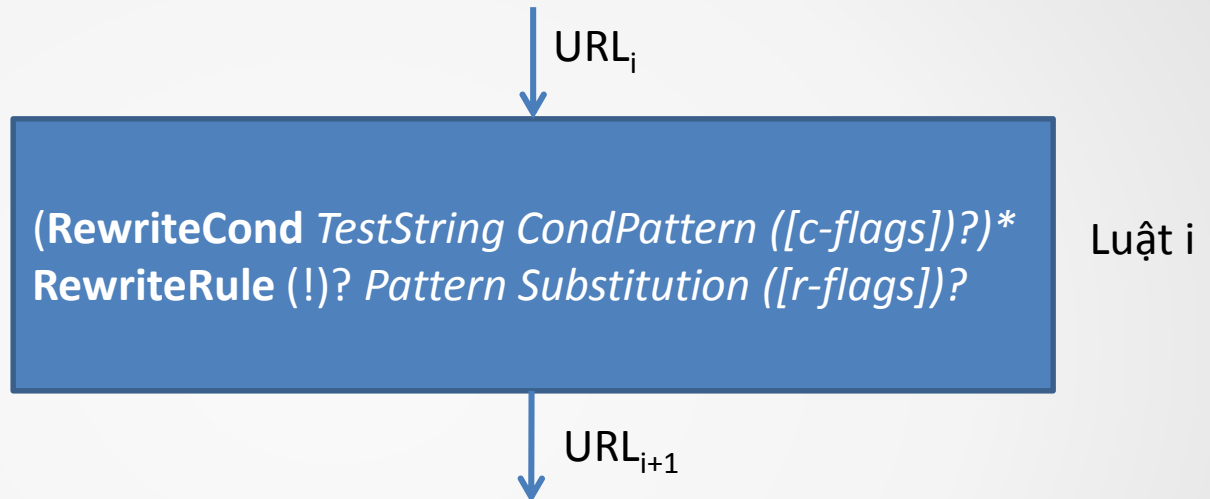
Viết lại URL như thế nào?

Rewrite Engine
thực hiện viết
lại URL dựa trên
tập luật được
cấu hình trước

Luật cho biết
cách sửa đổi
URL, được viết
theo biểu thức
chính quy



Luật viết lại URL



URL_{i+1} = URL_i
[Nếu mọi *TestString* khớp *CondPattern*]
Nếu URL_i (không) khớp *Pattern*
URL_{i+1} = *Substitution*

Luật viết lại URL (tiếp)

- **Pattern, CondPattern** là xâu ký tự được viết theo biểu thức chính quy

- Cú pháp biểu thức chính quy:

.	Bất kỳ ký tự nào
[string]	Bất kỳ ký tự nào thuộc “string”
[^string]	Bất kỳ ký tự nào khác các ký tự thuộc “string”
(text)	Nhóm văn bản
text1 text2	text1 hoặc text2
?	0 hoặc 1 lần xuất hiện của văn bản phía trước
*	0 hoặc N lần xuất hiện của văn bản phía trước (N > 0)
+	1 hoặc N lần xuất hiện của văn bản phía trước (N > 1)
^	Bắt đầu liên kết
\$	Kết thúc liên kết
\char	ký tự đặc biệt char

Ví dụ Pattern

- `^/wiki/.*$`
 - Khớp
 - `/wiki/Semantic_URL`
 - Không khớp
 - `/wiki-Semantic_URL`
- `^/su-kien/.*-[0-9]+\\.htm$`
 - Khớp
 - `/su-kien/mien-bac-mien-trung-mua-lon-trong-ngay-trung-thu-20171004080130699.htm`
 - `/su-kien/bao-dan-tri-tang-qua-trung-thu-cho-cac-em-co-hoan-can-h-kho-khan-20171004065430012.htm`
 - Không khớp
 - `/su-kien/987-error.htm.`

Luật viết lại URL (tiếp)

- ***TestString, Substitution*** là chuỗi ký tự có thể chứa các cấu trúc mở rộng sau:
 - $\$N$** ($0 \leq N \leq 9$) Tham chiếu ngược đến nhóm thứ N trong *Pattern*
 - $\%N$** ($0 \leq N \leq 9$) Tham chiếu ngược đến nhóm thứ N trong *CondPattern* liền trước
 - $\{\text{mapname:key|default}\}$** Giá trị ánh xạ
 - $\{\text{NAME_OF_VARIABLE}\}$** Biến server

Ví dụ tham chiếu ngược

/std/Tran-Van-Binh_012345.html



```
RewriteRule ^/std/[A-Za-z\-\_]+_(.*)\.html$ /std/viewStd.php?std_id=$1
```



/std/viewStd.php?std_id=012345

Biến server

- HTTP_USER_AGENT
HTTP_REFERER
HTTP_COOKIE
HTTP_FORWARDED
HTTP_HOST
HTTP_PROXY_CONNECTION
HTTP_ACCEPT
- DOCUMENT_ROOT
SERVER_ADMIN
SERVER_NAME
SERVER_ADDR
SERVER_PORT
SERVER_PROTOCOL
SERVER_SOFTWARE
- API_VERSION
THE_REQUEST
REQUEST_URI
REQUEST_FILENAME
IS_SUBREQ
HTTPS
- REMOTE_ADDR
REMOTE_HOST
REMOTE_PORT
REMOTE_USER
REMOTE_IDENT
REQUEST_METHOD
SCRIPT_FILENAME
PATH_INFO
QUERY_STRING
AUTH_TYPE
- TIME_YEAR
TIME_MON
TIME_DAY
TIME_HOUR
TIME_MIN
TIME_SEC
TIME_WDAY
TIME

Ví dụ biến server

uet.vnu.edu.vn



```
RewriteCond %{HTTP_HOST} ^uet\.vnu\.edu\.vn$  
RewriteRule (.*) http://www.uet.vnu.edu.vn/$1 [R=301,L]
```



www.uet.vnu.edu.vn

Ánh xạ

- **RewriteMap** *MapName MapType:MapSource*

- Ví dụ

RewriteMap product2id txt:productmap.txt

RewriteRule ^/product/(.)\$ /prods.php?id=\${product2id:\$1|0} [PT]*

#productmap.txt

##

Product to ID map

##

television 993

stereo 198

fishingrod 043

basketball 418

telephone 328

URL_i

/product/television

/product/basketball

/product/mobilefone

URL_{i+1}

/prods.php?id=993

/prods.php?id=418

/prods.php?id=0

Luật viết lại URL (tiếp)

- **c-flags:**

nocase | NC

Không phân biệt hoa thường khi so sánh TestString với CondPattern

ornext | OR

Kết hợp theo logic OR với điều kiện liền sau.

""

Kết hợp logic AND với điều kiện liền sau

Luật viết lại URL (tiếp)

- **r-flags:**

nocase | **NC**

chain | **C**

Không phân biệt hoa thường khi so sánh
Tạo chuỗi luật với luật tiếp theo. Nếu
một luật không khớp, các luật phía sau trong
chuỗi sẽ bị bỏ qua.

last | **L**

Dừng quá trình viết lại, không áp dụng thêm
các luật khác

next | **N**

Quay về luật thứ nhất

forbidden | **F**

Trả về response có mã 403 FORBIDDEN

gone | **G**

Trả về response có mã 410 GONE

redirect | **R[=*code*]**

Chuyển hướng

skip | **S=num**

Bỏ qua num luật tiếp sau nếu luật hiện
tại khớp

Một số chỉ dẫn khác

- **RewriteBase** *URL-path*

Tiền tố của URL được dùng trong chỉ thị RewriteRule thay thế đường dẫn tương đối

- **RewriteEngine** **on** | **off**

Bật hoặc tắt viết lại URL

Ví dụ

```
<IfModule mod_rewrite.c>
```

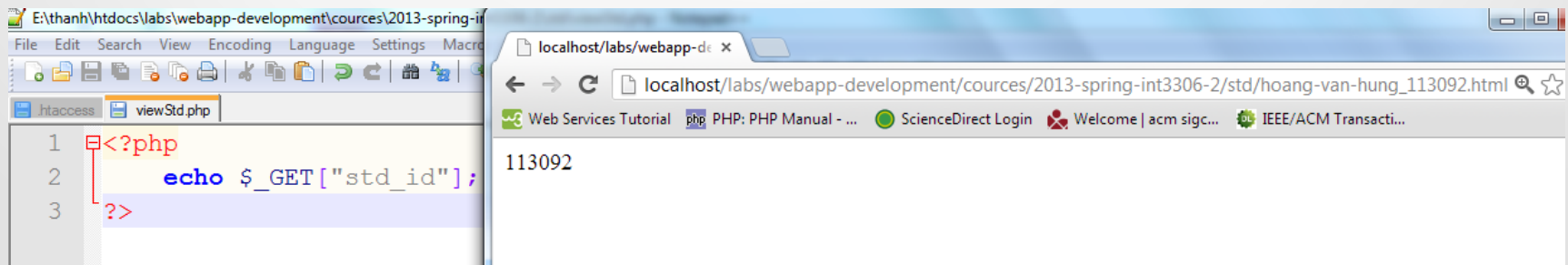
```
Options +FollowSymLinks
```

```
RewriteEngine on
```

```
RewriteBase /labs/webapp-development/courses/2013-  
spring-int3306-2/
```

```
RewriteRule ^std/[A-Za-z\-]+\_(.*)\.html$  
std/viewStd.php?std_id=$1
```

```
</IfModule>
```



Triển khai viết lại

- Bật modul viết lại *mod_rewrite* trong Apache
sudo a2enmod rewrite
- Cấu hình thư mục gốc của ứng dụng:
<Directory /path/to/home/folder/>
Options Indexes FollowSymLinks
AllowOverride All
</Directory>
- Luật viết lại được đặt trong các tệp *.htaccess* thuộc các thư mục của ứng dụng.

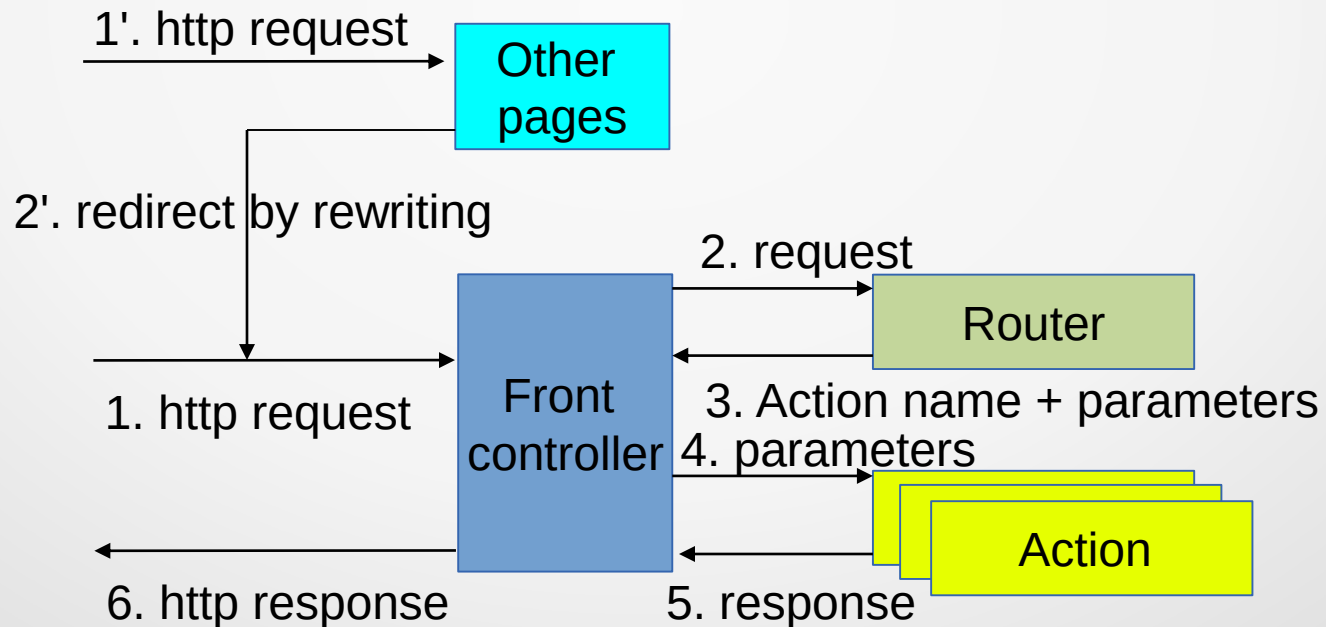
Định tuyến URL

Định tuyến URL là gì?

- Định tuyến URL (URL Routing) là phân tích url để quyết định xử lý ở server
 - If HTTP method matches verb_1 and URL matches pattern_1 then action_1
 - Else if HTTP method matches verb_2 and URL matches pattern_2 then action_2
 - ...

Định tuyến URL như thế nào?

- Ứng dụng web thực hiện định tuyến URL
 - Một trang đặc biệt được gọi là **bộ điều khiển mặt trước** (front controller) tiếp nhận mọi request
 - Bộ điều khiển mặt trước giống như cửa duy nhất vào ứng dụng
 - Cần hỗ trợ của viết lại URL để chuyển hướng mọi request đến trang khác về bộ điều khiển mặt trước
 - Bộ điều khiển mặt trước chuyển URL cho **bộ định tuyến** (router)
 - Bộ định tuyến phân tích url để xác định xử lý sẽ phục vụ
 - Xử lý được gọi và thực hiện.



Chuyển hướng tất cả các trang về bộ điều khiển mặt trước

- Tập .htaccess ở thư mục gốc

Options +FollowSymLinks

IndexIgnore */*

RewriteEngine On

RewriteCond %{REQUEST_FILENAME} !-f

RewriteCond %{REQUEST_FILENAME} !-d

RewriteRule .* index.php

REST

- Phong cách REST (Representational State Transfer)
 - Mỗi tài nguyên chỉ là một khái niệm, ánh xạ đến các thực thể.
 - Đặt tên cho tài nguyên và ánh xạ tài nguyên đến URL
 - RESTful hay REST-style URL

REST

- Tài nguyên có thể ở dạng sưu tập hoặc cá thể:

/forums

/forums/{forumId}

REST

- Tài nguyên có thể chứa các tài nguyên con:
/forums/{forumId}/posts
/forums/{forumId}/posts/{postId}
/forums/{forumId}/meta

REST

- Tài nguyên điều khiển (controller resource) được sử dụng để mô hình hóa các hàm chức năng:

/forums/{forumId}/close

REST

- Kết hợp URL với phương thức HTTP để mô hình hóa các hàm CRUD:

GET /forums/{forumId}/posts

GET /forums/{forumId}/posts/{postId}

POST /forums/{forumId}/posts

PUT /forums/{forumId}/posts/{postId}

DELETE /forums/{forumId}/posts/{postId}

REST

- Có thể sử dụng truy vấn để thực hiện các tính năng chung như sắp xếp, lọc, phân trang trên tài nguyên sưu tập:

GET

/forums/{forumId}/posts?from=01/01/2018&sort=date&order=desc

REST

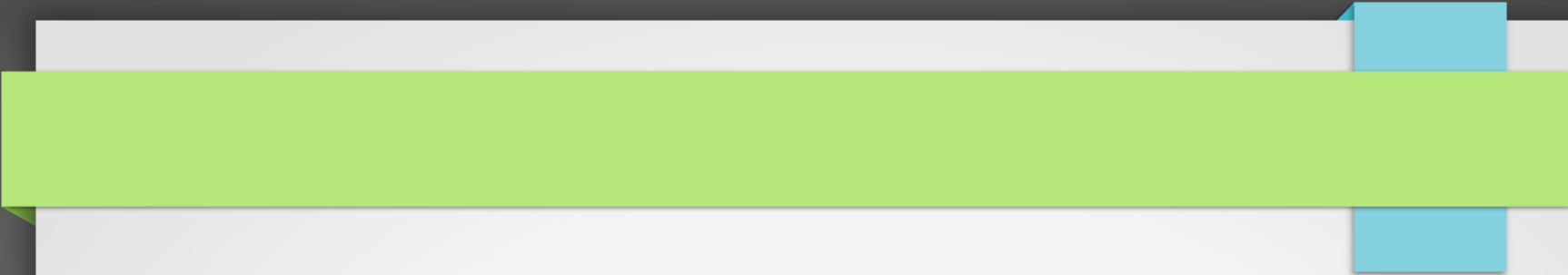
- Khác:
 - sử dụng chữ viết thường,
 - sử dụng dấu gạch ngang (-)
 - không sử dụng dấu gạch dưới (_)
 - không sử dụng tên mở rộng của tệp
 - không sử dụng dấu chéo trái ở cuối URL

Định tuyến cho RESTful URL

- Tên bó, tên bộ điều khiển và tên hành động là tên của những thực thể không cần xuất hiện trong URL.
- Bộ định tuyến cần ánh xạ tên tài nguyên sang các thực thể bó, bộ điều khiển và hành động
 - Kiểm tra phương thức HTTP và đối sánh URL với biểu thức chính quy. Nếu cả phương thức HTTP và biểu thức chính quy được thỏa mãn, bộ định tuyến sẽ trả về tên bó, tên bộ điều khiển, tên hành động cùng các tham số tương ứng.

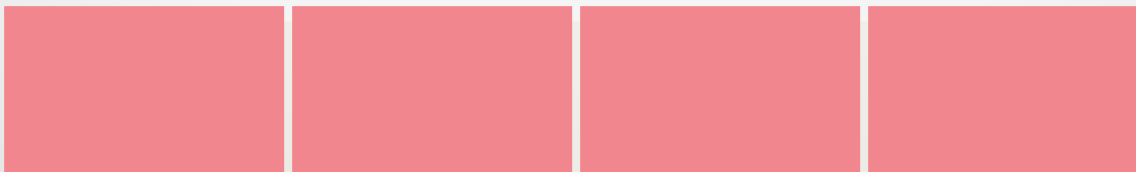
Định tuyến cho RESTful URL

```
22. // Phân tích URL để xác định lớp điều khiển, hành động cùng tham số
23. if ($_SERVER['REQUEST_METHOD'] == 'GET' &&
    preg_match("/^\/forums$/", $uri) == 1) {
24.     // GET /forums
25.     $moduleName = "forum-manager";
26.     $controllerName = "forum";
27.     $actionName = "listAllForums";
28. } elseif ($_SERVER['REQUEST_METHOD'] == 'GET' &&
    preg_match("/^\/forums\/[[:digit:]]+$/", $uri) == 1) {
29.     // GET /forums/{forumId}
30.     $moduleName = "forum-manager";
31.     $controllerName = "forum";
32.     $actionName = "viewForumById";
33.     $parameters["forumId"] = $commandArray[1];
34. } elseif ($_SERVER['REQUEST_METHOD'] == 'PUT' &&
    preg_match("/^\/forums\/[[:digit:]]+$/", $uri) == 1) {
35.     // PUT /forums/{forumId}
36.     $moduleName = "forum-manager";
37.     $controllerName = "forum";
38.     $actionName = "updateForumById";
39.     $parameters["forumId"] = $commandArray[1];
40.     parse_str(file_get_contents('php://input'), $input);
41.     $parameters["forumNewName"] = $input["forumNewName"];
42. } elseif ($_SERVER['REQUEST_METHOD'] == 'GET' &&
```



Laravel

<https://laravel.com>



Ánh xạ URL đến hành động của lớp điều khiển

```
<?php
```

```
// Tập routes/web.php
```

```
...
```

```
Route::get('/books/{bookId}', 'BookController@show');
```

Ánh xạ URL đến hành động của lớp điều khiển

```
Route::resource('/books', 'BookController');
```

Phương thức HTTP	URI	Hành động
<i>GET</i>	<i>/books</i>	<i>index</i>
<i>GET</i>	<i>/books/create</i>	<i>create</i>
<i>POST</i>	<i>/books</i>	<i>store</i>
<i>GET</i>	<i>/books/{bookId}</i>	<i>show</i>
<i>GET</i>	<i>/books/{bookId}/edit</i>	<i>edit</i>
<i>PUT/PATCH</i>	<i>/books/{bookId}</i>	<i>update</i>
<i>DELETE</i>	<i>/books/{bookId}</i>	<i>destroy</i>

Tiếp theo
Thao tác CSDL

