







TP - PHP

You have 7 hours to realize the following case.

You will be evaluated on your ability to meet the following  functional goals.

PHP and Web Services

The job is about to repeat endpoint creation and provide routing :





-  You need to handle the client request : [@see w3.org request](#)
-  You need to follow REST principles : [@see wiki rest](#)
-  You need to send a response : [@see w3.org response](#)



Login endpoint



An uri with parameters for a method have to provide an user :

```
path: /login?email=USER_EMAIL&password=USER_PASSWORD
method: GET
controller: Wog\Controler\Api>LoginController
action: login
```

-  The api must provide an user for the route described
-  The api must repond with a 404 status code for non existing user
-  The api must repond with a 200 status code for existing user
-  The api must repond with a json body of the retrieved user

Password

After you provide the user for his email and his password, take care about password storage.


-  At creation :
 - The api must hash password and do not store clear value
-  At login :
 - The api must find user for a hashed password

[@see password-hash](#)

[@see password-verify](#)

Token

After you provide hash the user password, take care about token storage.

-  At creation :
 - The api must hash token with the password value

[@see md5](#)



Routing

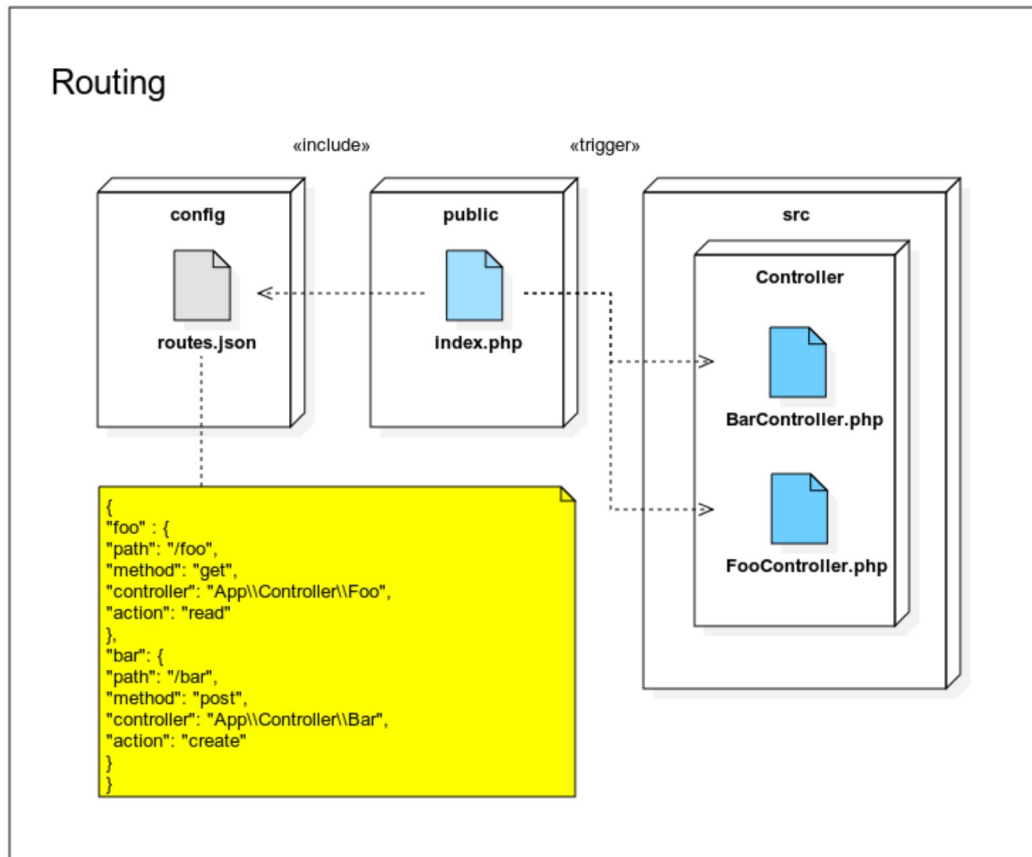
At the moment when you add an endpoint you must add code in your entry point

You need to follow the open/closed principle for routing.

[@see open/closed](#)

- An endpoint can be described by a route
- Routes can be stored in a configuration file
- Project can use this configuration file for trigger an controller action

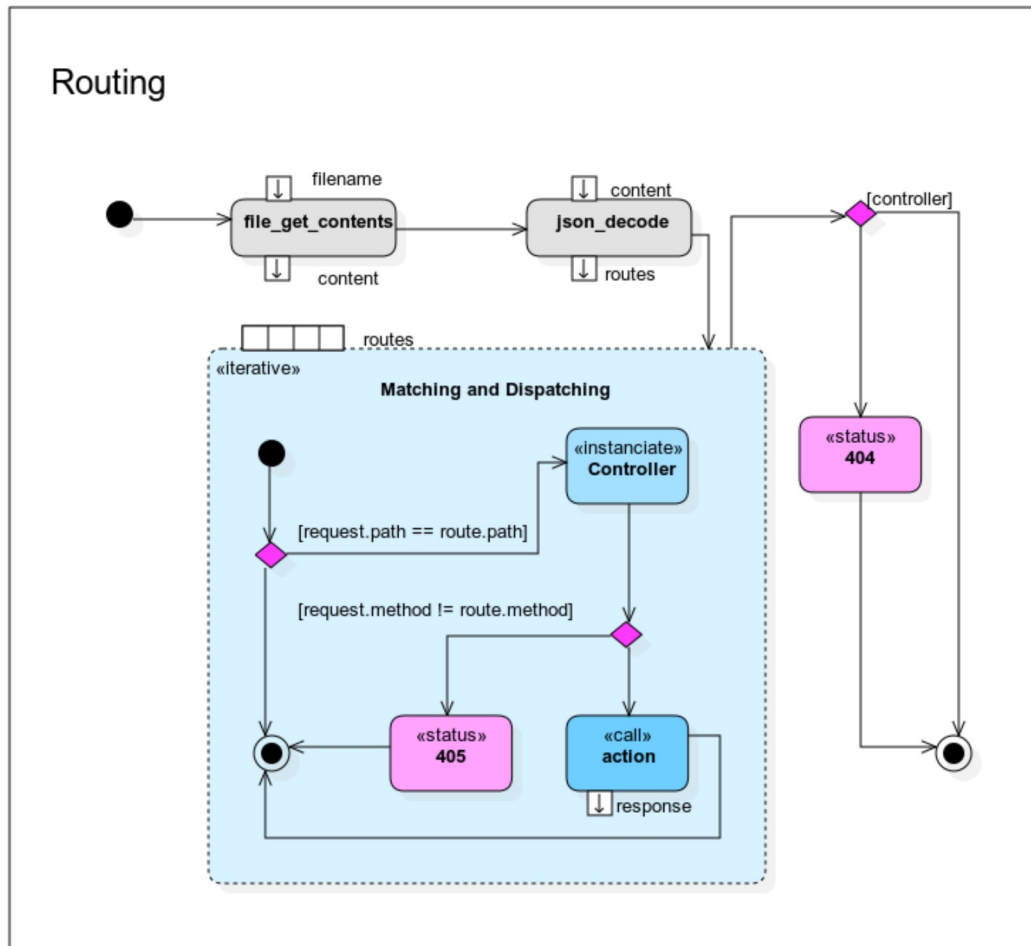
This principle can be described by the following deployment diagram:



[@see file_get_contents](#)

[@see json_decode](#)

This principle can be described by the following activity diagram:



- The api must use a configuration file for routing
- The configuration file must store a collection of route
- A route have a least a path, method, controller and action

Tips

Class identifier can be dynamic

```
$className = "Foo";  
$object = new $className();
```

Method identifier can be dynamic

```
$methodName = "read";  
$object->$methodName();
```

[@see dynamic language features](#)

Manage your time

You have one thematic, you need to create your ow api, focus and try to find solutions for improve your skill.

Let's focus