



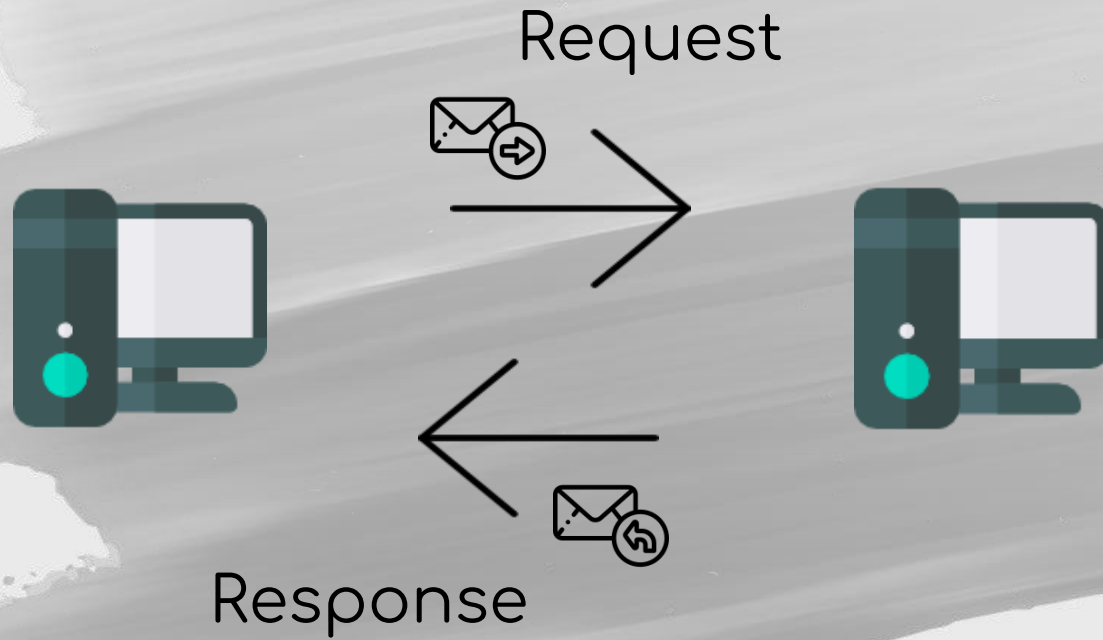
HTTP Protocol

By: Andre Gunawan, S.Kom.



1. Concept

How It Works?





Request

GET /docs/index.html HTTP/1.1

Host: **www.nowhere123.com**

Accept: image/gif, image/jpeg, */*

Accept-Language: en-us

Accept-Encoding: gzip, deflate

User-Agent: Mozilla/4.0 (compatible;
MSIE 6.0; Windows NT 5.1)

(Blank Line)



Response

HTTP/1.1 200 OK

Date: Sun, 18 Oct 2009 08:56:53 GMT

Server: Apache/2.2.14 (Win32)

Last-Modified: Sat, 20 Nov 2004 07:16:26
GMT

Accept-Ranges: bytes

Content-Length: 44

Content-Type: text/html

<html>

<body><h1>It works!</h1></body>

</html>



2. Sample Code

This Code Will Show a
PDF File on Browser!

```
<?php  
  
$file = "BUKU_PANDUAN.pdf";  
  
header('Content-type: application/pdf');  
header('Content-Disposition: inline; filename="' . $file . '"');  
header('Content-Transfer-Encoding: binary');  
header('Accept-Ranges: bytes');  
header('Content-Length: ' . filesize($file));  
  
echo file_get_contents($file);
```

Filename

What Kind of Content is Sent

How Content Encoded

Support Partial Request

Content Length in Byte

Body of Response

What Happen IF No *Content-Length* Header?

```
<?php
```

```
$file = "BUKU_PANDUAN.pdf";
```

```
header('Content-type: application/pdf');
```

```
header('Content-Disposition: inline; filename="' . $file . '');
```

```
header('Content-Transfer-Encoding: binary');
```

```
header('Accept-Ranges: bytes');
```

```
//header('Content-Length: ' . filesize($file));
```

 Commented

```
echo file_get_contents($file);
```

No Loading Bar!

What Happen IF *Content-Disposition* is *Attachment*?

```
<?php
```

```
$file = "BUKU_PANDUAN.pdf";
```

```
header('Content-type: application/pdf');
```

```
header('Content-Disposition: attachment; filename="' . $file . '');
```

```
header('Content-Transfer-Encoding: binary');
```

```
header('Accept-Ranges: bytes');
```

```
//header('Content-Length: ' . filesize($file));
```

Commented

```
echo file_get_contents($file);
```

It Download the PDF File



“

*Header Control How Content is
Presented!*

Another Example, What Happen?

```
<?php  
  
$txt = "Meow!";  
?  
  
<html>  
  
<head>  
    <title>Hello World</title>  
</head>  
  
<body>  
    <h1><?php echo $txt; ?></h1>  
</body>  
</html>
```

It Show a Website

IF you code like this, What Happen?

```
<?php
header('Content-type: text/plain');
$txt = "Meow!";
?>
<html>
<head>
    <title>Hello World</title>
</head>
<body>
    <h1><?php echo $txt; ?></h1>
</body>
</html>
```

It Show a Text

“

PHP is not to make a Website!
*PHP is to write a **Response or***
Request!

As you can make a response
you can make a website



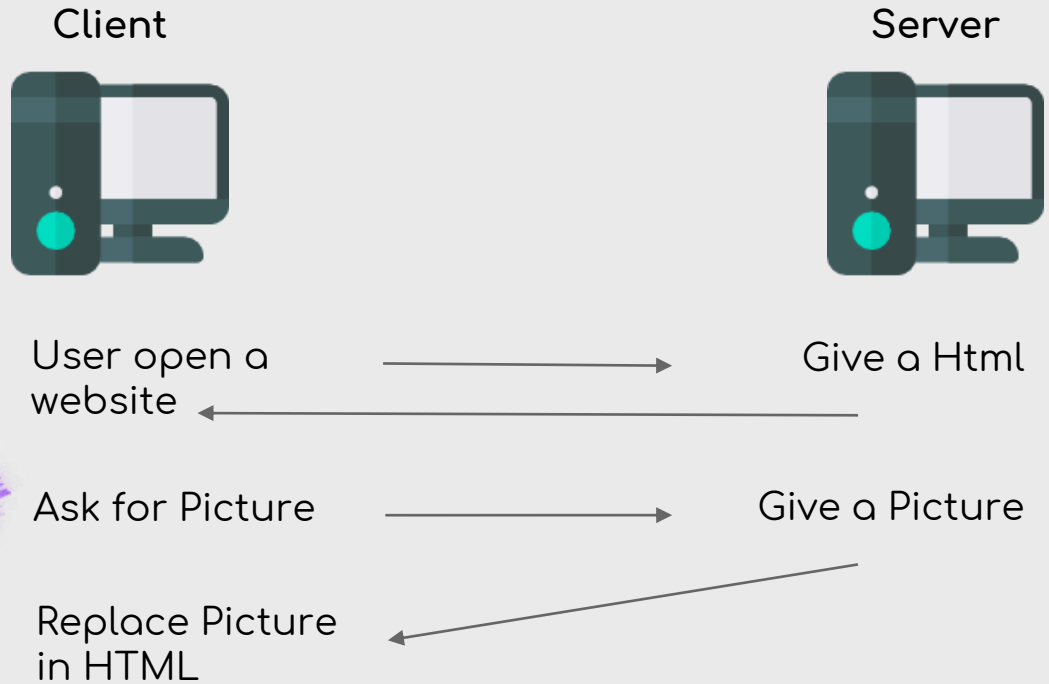
How to Make a Service?

A green brushstroke graphic with a white rectangular frame. The brushstroke is a vibrant green color and has a textured, painterly appearance. The white frame is a simple rectangle that encloses the word "Services".

Services

Service is in form of many things. Web, Images, Stream, Document, and more. This Bonus Chapter give a view what services looks like?

Example



Example: a simple service that give JSON data.

```
<?php

header('Content-type: text/plain');

$data = array(
    "name" => "Andre Gunawan",
    "nrp" => "26414004"
);

echo json_encode($data);
```

Include the JQuery:

```
<script  
  src="https://code.jquery.com/jquery-3.3.1.js"  
  integrity="sha256-2Kok7MbOyxpgUVvAk/HJ2jigOSYS2auK4Pfzbm7uH60="  
  crossorigin="anonymous"></script>
```

Not Working ..?

Html in Body:

```
<div>  
  <p id="name"></p>  
  <p id="nrp"></p>  
</div>
```

Script:

```
<script>  
  $(document).ready(function() {  
    $.get("data.php", {}, function(data) {  
      $("#name").html(data['name']);  
      $("#nrp").html(data['nrp']);  
    });  
  });  
</script>
```



Remember!

When You Send Response With This Header:

```
header('Content-type: text/plain');
```

Javascript Interpret it as **STRING** not **ARRAY**!

Solution:

```
header('Content-type: application/json');
```

Update Update Update

```
<?php

header('Content-type: application/json');

$data = array(
    "name" => "Andre Gunawan",
    "nrp" => "26414004"
);

echo json_encode($data);
```

img.php:

Example: Returning Image as Service

```
<?php

$file = 'cute-kitten.jpg';
$type = 'image/jpeg';
header('Content-Type:'.$type);
header('Content-Length: ' . filesize($file));

echo file_get_contents($file);
```

On HTML:

```

```

The background of the slide features two large, horizontal, overlapping brushstrokes. The top stroke is a deep blue, and the bottom stroke is a vibrant teal. Both strokes have a textured, painterly appearance with visible brush marks and irregular edges. The text "Another Example ..." is centered over the teal stroke in a white, sans-serif font.

Another Example ...

img.php:

Example: Returning Image as Service

```
<?php

header('Content-type: application/json');

$file = 'cute-kitten.jpg';

$str = file_get_contents($file);
$base64 = base64_encode($str);

$data = array(
    "name" => $file,
    "img" => $base64
);

echo json_encode($data);
```

This is on Javascript

```
$.get("img.php", {}, function(data) {  
    $("#img").attr('src', 'data:image/jpeg;base64,' + data['img']);  
});
```

Base64 is a group of similar binary-to-text encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation. The term Base64 originates from a specific MIME content transfer encoding. Each Base64 digit represents exactly 6 bits of data. Three 8-bit bytes (i.e., a total of 24 bits) can therefore be represented by four 6-bit Base64 digits

- *Wikipedia*



THANKS!
Any questions?