



Pemrograman Berbasis Web

Pertemuan 10

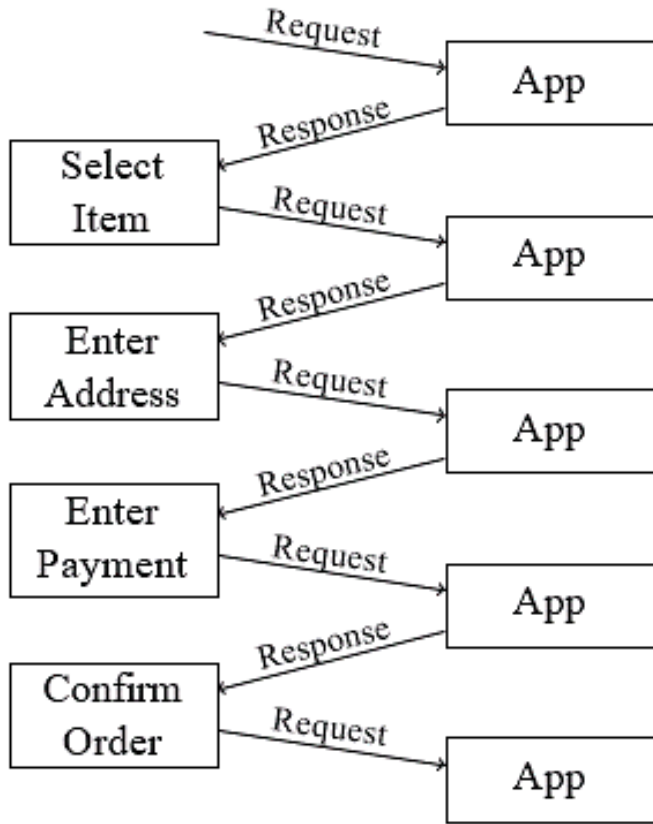
Nori Wilantika
Lya Hulliyyatus Suadaa
Yeni Rimawati

Politeknik Statistika STIS
Prodi DIV Komputasi Statistik



10

SESSION & COOKIES



- Interaksi antara pengguna dan aplikasi web server-side biasanya membutuhkan sequence dari request dan response
- HTTP adalah **stateless protocol**
- Untuk setiap request, aplikasi selalu memulai dari awal (tidak mengingat data apapun antara request sebelum dan setelahnya)
- Tidak dapat mengetahui apakah requests datang dari pengguna yang sama atau berbeda
- Data perlu dikirim dari eksekusi aplikasi yang satu ke eksekusi selanjutnya



Statelessness

- The server does not store any state about the client session on the server-side.
- Each request is independent: cannot access information stored in previous requests.
- Each request from the client to the server must contain all of the necessary information to understand the request.
- The client is responsible for storing and handling the session related information on its own side



form1.php

```
<form action="form2.php" method="post">
  <label>Item: <input type="text" name="item"></label>
</form>
```

form2.php

```
<form action="process.php" method="post">
  <label>Address: <input type="text" name="address"></label>
</form>
```

process.php

```
<?php
    echo $_POST['item'];
    echo $_POST['address'];
?>
```



form1.php

```
<form action="form2.php" method="post">
  <label>Item: <input type="text" name="item"></label>
</form>
```

form2.php

```
<form action="process.php" method="post">
  <label> Address:<input type="text" name="address"> </label>
  <input type="hidden" name="item value="<?php echo $_POST['item']?>">
</form>
```

process.php

```
<?php
  echo $_POST['item'];  echo $_POST['address'];
?>
```

The background is a light cream color, decorated with several abstract elements. In the top left, there is a horizontal red brushstroke and a cluster of small black dots. On the left side, there is a large, textured grey circle with a black starburst pattern above it. In the bottom left, there is a pink brushstroke and a cluster of small black dots. On the right side, there is a large, textured yellow circle, a thin black wavy line, and a small yellow circle. In the bottom right, there is a grey brushstroke and a small yellow circle.

SESSION

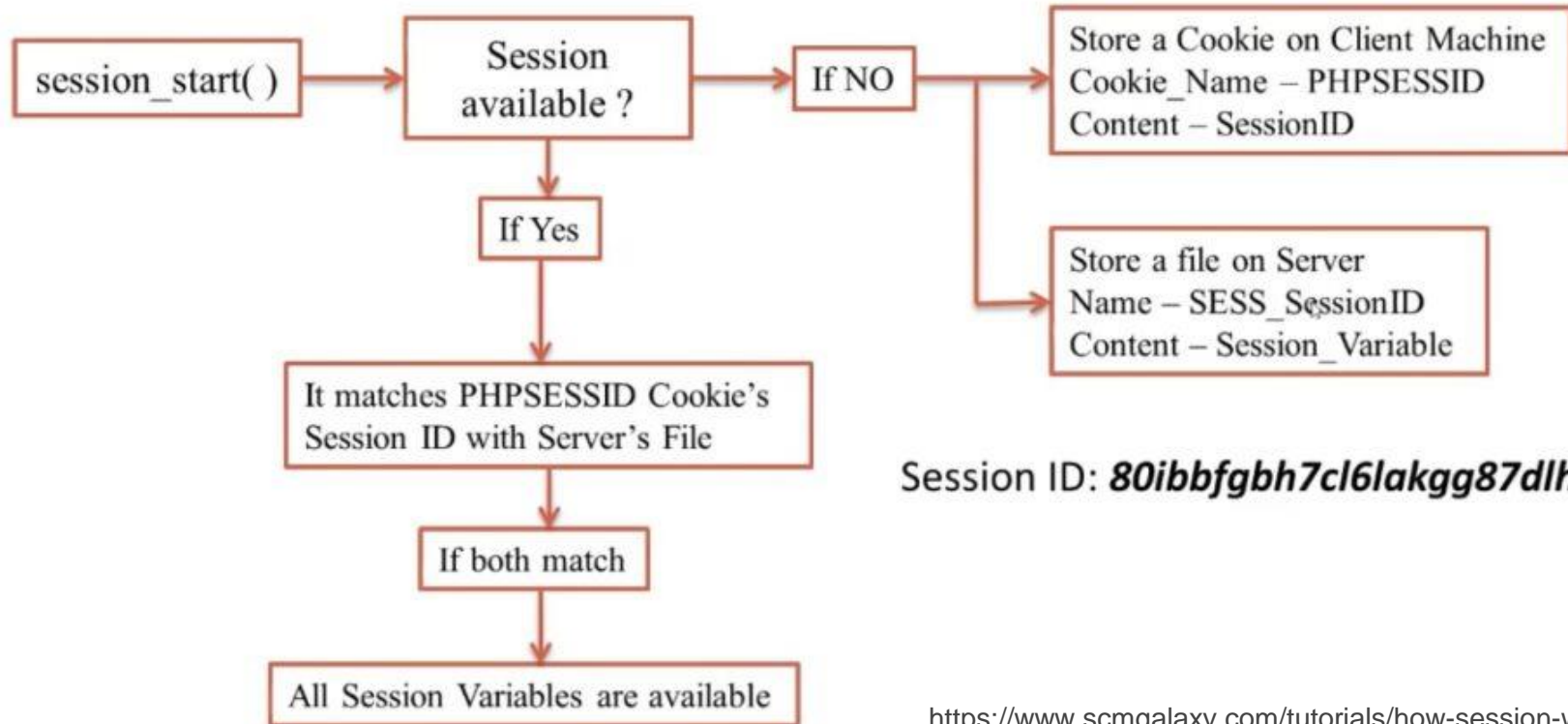


SESSION

- Session adalah cara untuk menyimpan informasi (dalam variabel) untuk digunakan di beberapa halaman (misalnya nama pengguna, id pengguna, warna favorit, dll).
- Variabel session hanya menyimpan informasi sementara. Secara default, variabel session bertahan hingga pengguna menutup browser.
- Server menjaga history session pengguna dengan session identifier, yang:
 - Dihasilkan server ketika session dimulai
 - Diingat oleh browser
 - Dikirim ke browser dengan HTTP request selanjutnya ke server tersebut
 - Dilupakan browser ketika session berakhir atau browser ditutup



How Session Works



Session ID: **80ibbfgbh7cl6lakgg87dlhv7v**



START A PHP SESSION

```
<?php
// Start the session
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// Set session variables
$_SESSION["favcolor"] = "green";
$_SESSION["favanimal"] = "cat";
echo "Session variables are set.";
?>

</body>
</html>
```

- Session dimulai dengan fungsi **session_start()**.
- Ketika fungsi tersebut dijalankan:
 - fungsi tersebut membuat session identifier (session id)
 - menyiapkan array `$_SESSION` yang menyimpan variabel dan data session.
- Fungsi `session_start()` dituliskan di paling awal dokumen, sebelum tag HTML.
- Untuk mengubah nilai dari variabel sesion, cukup timpa dengan nilai yang baru, contoh:
`$_SESSION["favcolor"] = "yellow";`



GET PHP SESSION VARIABLE VALUES

- Variabel session tidak dikirim secara individual ke setiap halaman baru melainkan diambil dari session di awal setiap halaman (session_start()).

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// Echo session variables that were set on previous page
echo "Favorite color is " . $_SESSION["favcolor"] . "<br>";
echo "Favorite animal is " . $_SESSION["fanimal"] . ".";
?>

</body>
</html>
```



END A PHP SESSION

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// remove all session variables
session_unset();

// destroy the session
session_destroy();
?>

</body>
</html>
```

- Untuk menghapus semua variable global session dan menghancurkan session, gunakan **session_unset()** dan **session_destroy()**
- **session_unset()**: membebaskan semua variabel session yang saat ini terdaftar
- **session_destroy()**: menghancurkan semua data yang terkait dengan sesi saat ini

The background is a light cream color with various abstract elements. In the top left, there is a horizontal red brushstroke and a cluster of small black dots. On the left side, there is a large, textured grey circle with a black starburst line above it. In the bottom left, there is a pink brushstroke and a cluster of small black dots. On the right side, there is a large yellow watercolor circle, a thin black wavy line with two small yellow circles attached to it, and a grey brushstroke at the bottom right.

COOKIES



COOKIES

- Cookies dapat bertahan dalam satu session dan mentransfer informasi dari satu session ke session berikutnya
- Cookie adalah file kecil yang disematkan server di komputer pengguna.
- Tidak seperti session, pada cookie, informasi disimpan di komputer pengguna.



CREATE COOKIES

- Cookie dibuat dengan fungsi `setcookie()`.

```
setcookie(name, value, expire, path, domain, secure, httponly);
```

- Hanya parameter **name** dibutuhkan. Parameter lain opsional.
- Fungsi `setcookie()` harus muncul SEBELUM tag `<html>`.
- Untuk mengambil nilai dari cookie, gunakan variable global `$_COOKIE`
- Untuk mengubah cookie, atur (kembali) cookie menggunakan fungsi `setcookie()`



CREATE COOKIES

```
<!DOCTYPE html>
<?php
$cookie_name = "user";
$cookie_value = "John Doe";
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/"); // 86400 = 1 day
?>
<html>
<body>
```




DELETE A COOKIE

- Untuk menghapus cookie, gunakan fungsi setcookie() dengan expiration date pada masa lalu:

```
<?php
// set the expiration date to one hour ago
setcookie("user", "", time() - 3600);
?>

<html>
<body>

<?php
echo "Cookie 'user' is deleted.";
?>

</body>
</html>
```



KEGUNAAN SESSIONS DAN COOKIE

- Shopping cart
- Suggestion dari browsing history
- Fasilitas remember me login
- Otentikasi



SESSIONS AND AUTHENTICATION

- Session biasanya digunakan untuk mengizinkan atau menolak akses ke halaman web berdasarkan pengguna yang telah diautentikasi untuk melindungi halaman dari penggunaan yang tidak sah
- Cara kerjanya:
 - Sebelum diizinkan mengakses halaman php (misalnya content.php), pengguna harus terlebih dahulu melakukan otentikasi dengan memberikan username dan password pada halaman login
 - Sistem mencocokkan nama pengguna dan kata sandi yang dimasukkan oleh pengguna dengan yang disimpan pada database. Jika pemeriksaan berhasil, session variabel diset.
 - Halaman content.php memeriksa apakah session variabel sudah disetel, jika sudah, maka pengguna dapat melihat konten halaman
 - Jika session variabel tidak disetel, pengguna dialihkan ke login.php.
 - Sistem juga menyediakan halaman logout.php untuk memungkinkan pengguna logout

The background is a light cream color, decorated with several abstract elements. In the top left, there is a horizontal red brushstroke and a cluster of small black dots. On the left side, there is a large, textured grey circle with a black starburst pattern above it. In the bottom left, there is a pink brushstroke and a cluster of small black dots. On the right side, there is a large, textured yellow circle, a thin black wavy line with two small yellow circles attached to it, and a grey brushstroke at the bottom right.

CACHE



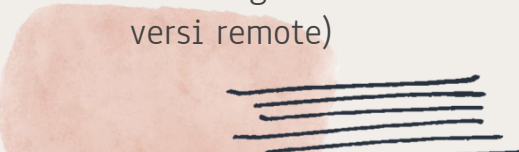

Caching

- ❖ Caching is the ability to store copies of frequently accessed data in several places along the request-response path.
- ❖ Caching can be local cache, proxy cache, or reverse proxy
- ❖ Cache adalah local copy dari informasi yang diperoleh dari sumber yang lain
- ❖ Caching brings performance improvement for the client-side and better scope for scalability for a server because the load has been reduced.
- ❖ Each request is independent of the previous request, so the results can be stored in the cache



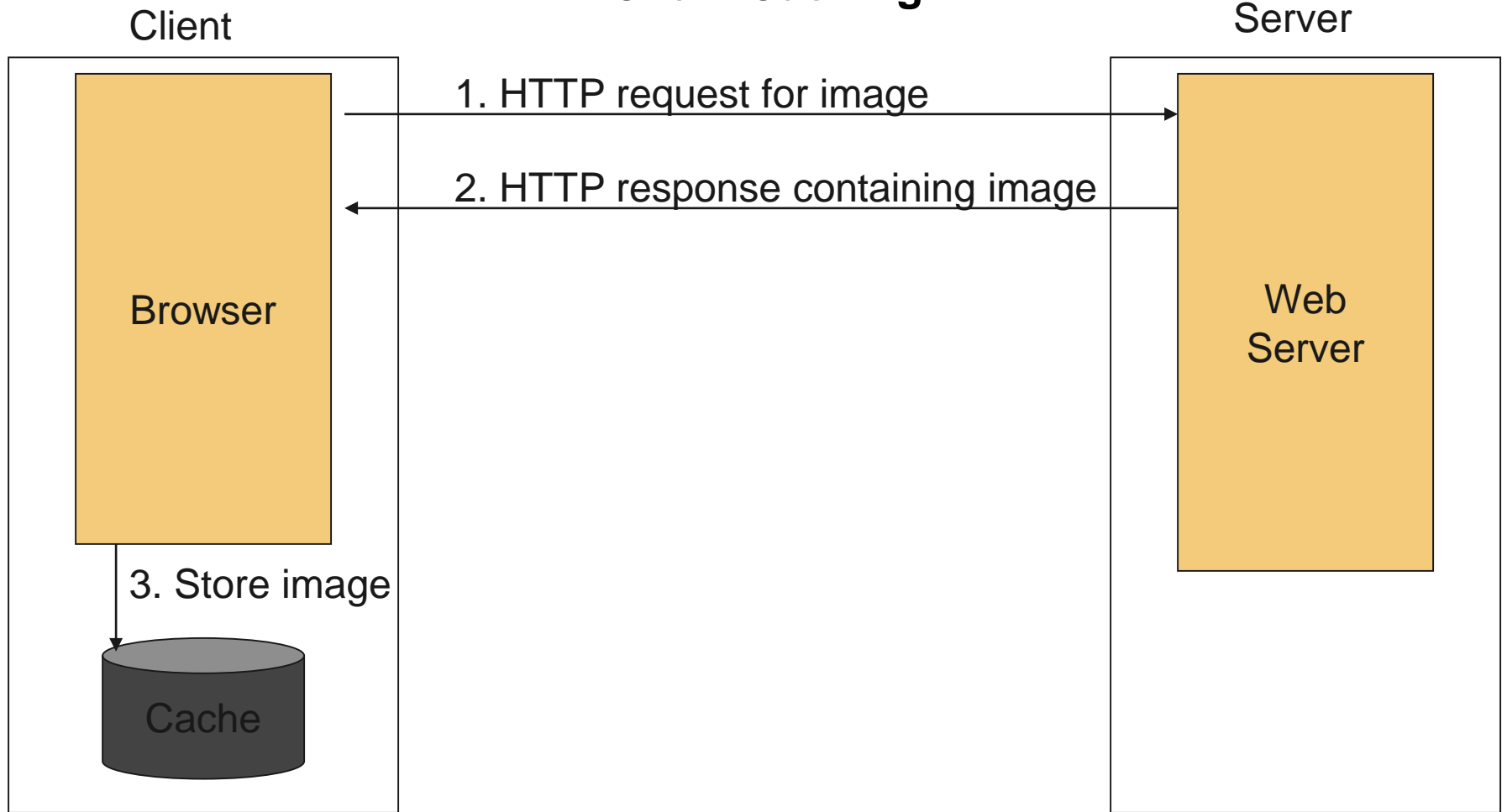


Client Caching

- ❖ Kebanyakan web browsers menggunakan cache untuk menyimpan resources yang diminta sehingga request selanjutnya ke sumber yang sama tidak lagi membutuhkan HTTP request/response
 - Contohnya icon yang tampil berulang kali dalam suatu halaman web
 - ❖ Keuntungan cache:
 - (Jauh) Lebih cepat dari HTTP request/response
 - Less network traffic
 - Less load on server
 - Reduce bandwidth
 - Reduce latency
 - Hide network failures
 - ❖ Kekurangan cache: cache yang tersimpan mungkin invalid (tidak konsisten dengan versi remote)
- 
- 

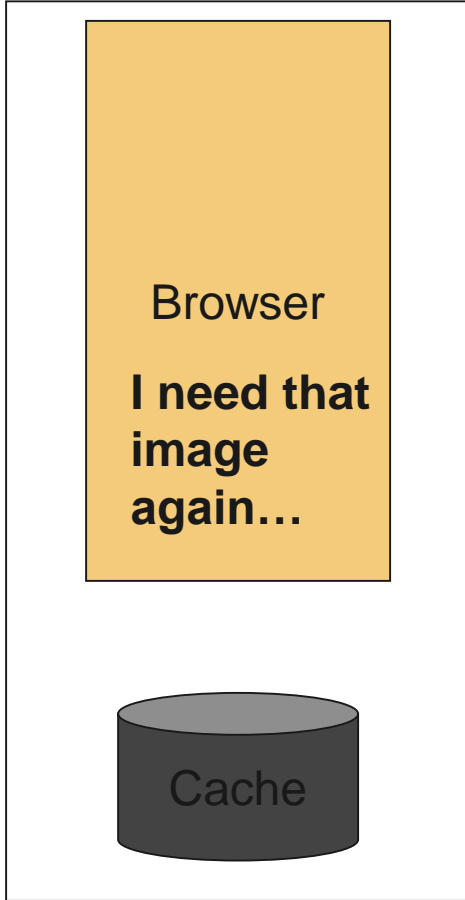


Client Caching

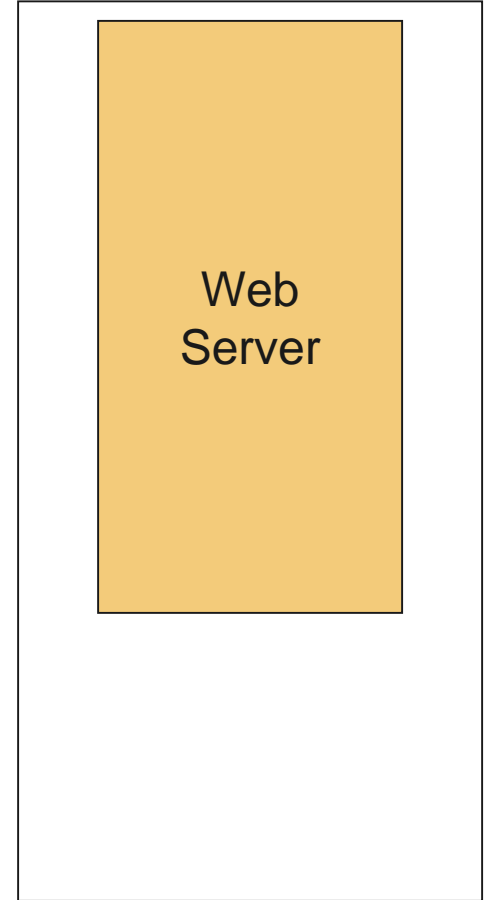


Client Caching

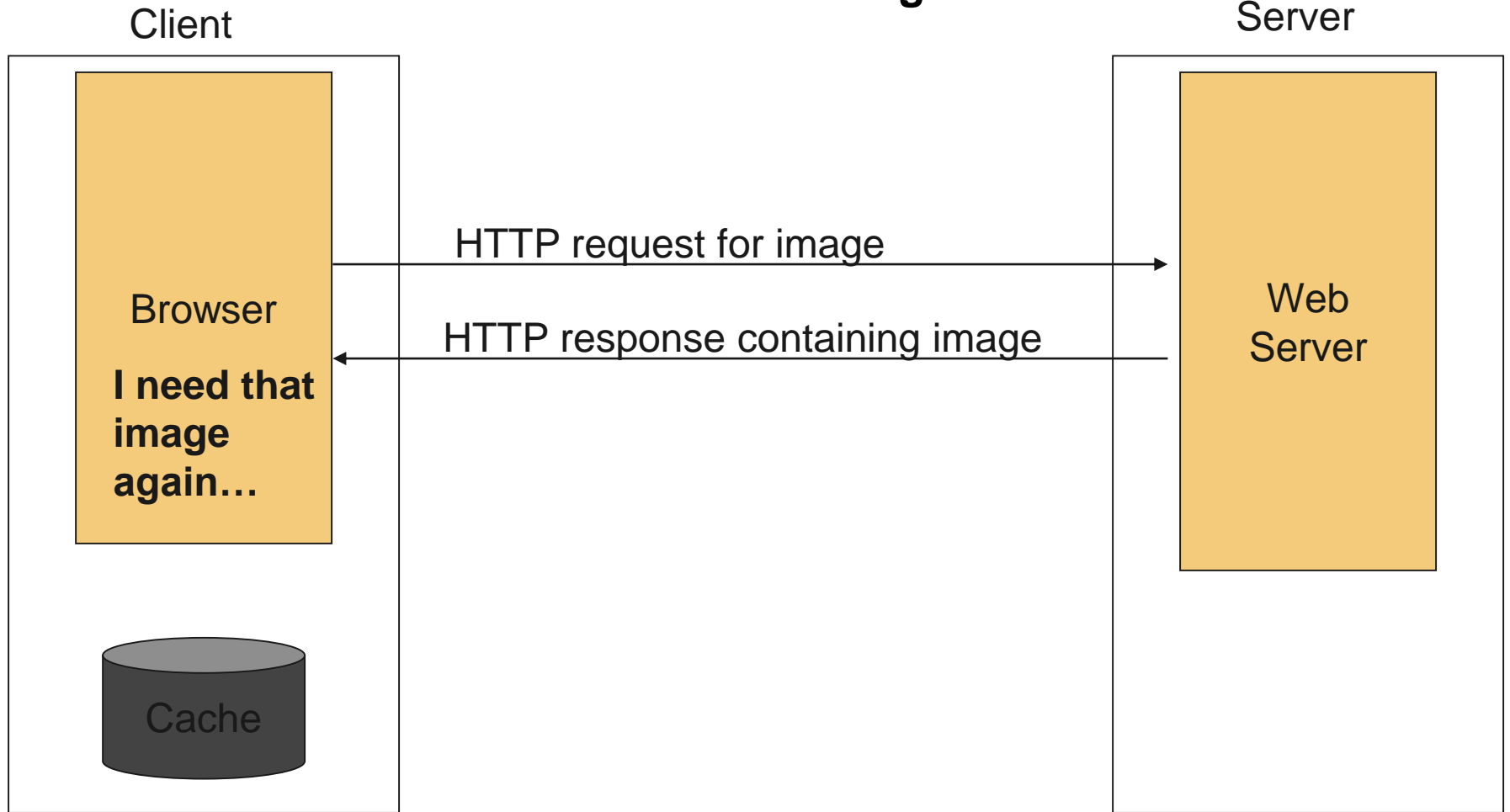
Client



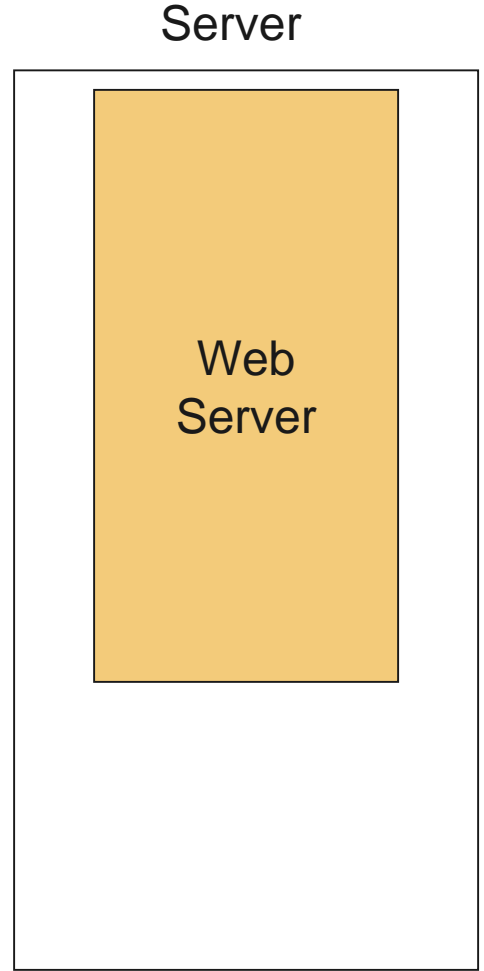
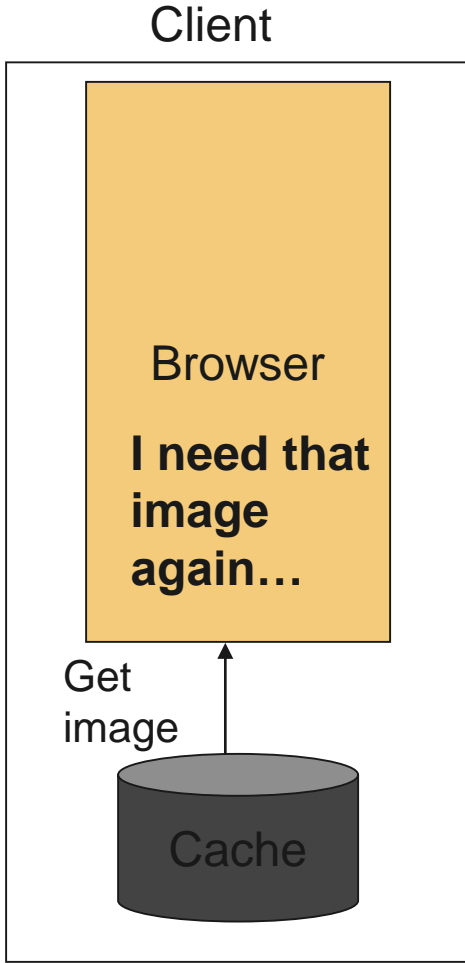
Server



Client Caching



Client Caching





REFERENSI

Materi disadur dari:

1. COMP519 Web Programming (2020-21). Diakses pada Januari 2021, dari <https://cgi.csc.liv.ac.uk/~ullrich/COMP519/>
2. W3Schools Online Web Tutorials. Diakses pada Januari 2021, dari <https://www.w3schools.com/>

Thanks!

Do you have any questions?



wilantika@stis.ac.id

lya@stis.ac.id

yeni.rima@stis.ac.id

CREDITS: This presentation template was created by Slidesgo, including icons by Flaticon, and infographics & images by Freepik