

# MODUL I

## Dynamic Web Project

### A. Tujuan

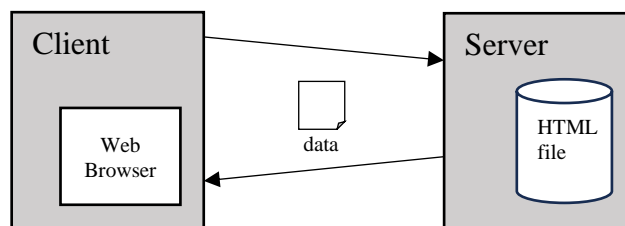
Pada modul Membuat Dynamic Web Project, mahasiswa diharapkan:

1. Mampu mengenal tool yang dibutuhkan untuk Java web application.
2. Mampu memahami perbedaan static web page dengan dynamic web page.
3. Mampu menguasai mekanisme kerja JavaServer Page dan servlet dalam membangun Java web application.

### B. Teori Dasar

#### 1. Web Server

Data yang ditampilkan di website tersimpan di suatu komputer yang disebut dengan komputer server. Data yang tersimpan di server dapat berbentuk file, misalnya file HTML. Komputer yang mengakses data dari komputer server, disebut dengan komputer client, dapat mengakses melalui internet. Komputer server bertugas mengatur pertukaran data dengan komputer client.



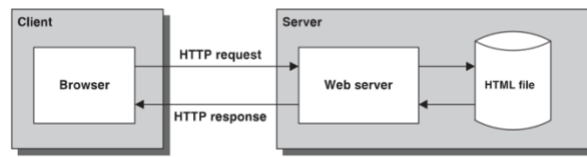
Gambar 1 Komputer server dan client

Web browser yang terpasang komputer client digunakan oleh user untuk mengakses file di web server. Perangkat lunak di dalam web server yang mengendalikan aksesibilitas user ini ialah HTTP server.

#### 2. Static vs Dynamic Web Page

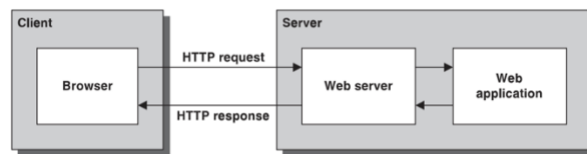
Data di suatu website dapat bersifat tetap ataupun berubah sesuai input dari user. Oleh karena itu, web page di suatu website terbagi menjadi dua tipe, yakni static dan dynamic web page. Static web page merupakan web page yang tampil dan tidak akan berubah saat diberi input oleh user.

User dapat mengirimkan permintaan/request untuk mengakses web page menggunakan web browser. Web browser menggunakan HTTP request yang mengikuti Hypertext Transfer Protocol (HTTP), yakni suatu aturan dalam pengiriman request ke web server. Web server yang menerima HTTP request, akan memberikan file ke web browser dalam bentuk HTTP response. Web browser menerjemahkan HTTP response dalam bentuk web page.



Gambar 2 Mekanisme static web page

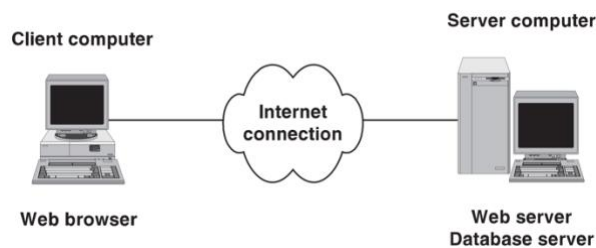
Sedangkan dynamic web page merupakan webpage yang memiliki tampilan yang dapat berubah berdasarkan input user. User menginputkan data ke web page, kemudian web browser mengirim HTTP request ke web server. Request ini berisi alamat webpage beserta data yang diinputkan. Web server menerima HTTP request ini dan mengirimkannya ke web application. Web application memproses data yang diinput user kemudian membuat dokumen HTML. Dokumen ini dikirim ke web server untuk diberikan kembali ke web browser dalam bentuk HTTP response. Web browser menampilkannya dalam bentuk webpage.



Gambar 3 Mekanisme dynamic web page

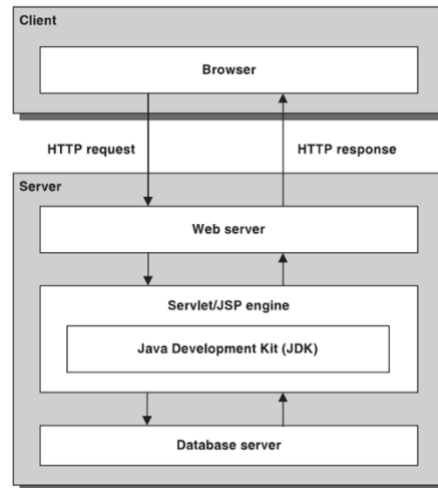
### 3. Web Application

Web application atau web app adalah sekumpulan web page yang dibuat dari response berdasarkan request yang diminta oleh user. User dapat mengakses web app menggunakan web browser. Web app tersimpan di web server yang telah dipasang HTTP server. HTTP server yang populer untuk web app di Java ialah Apache HTTP Server. Web app dapat mengelola data melalui database management system (DBMS) yang terpasang di web server. DBMS yang populer untuk Java ialah Oracle.



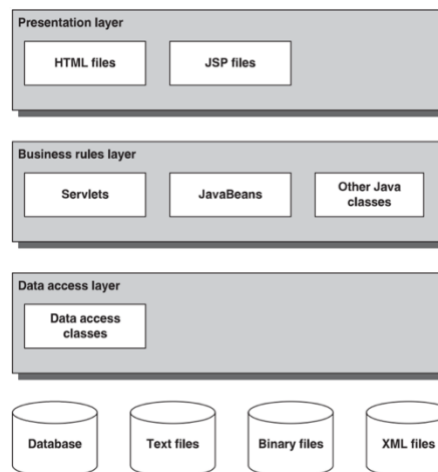
Gambar 4 Komponen web application

Java web application dibangun menggunakan JavaServer Pages (JSP) dan servlet. Web server dapat menjalankan aplikasi Java melalui servlet dan melalui suatu software, yakni JSP engine. Salah satu servlet/JSP engine yang populer untuk Java ialah Apache Tomcat. Servlet/JSP engine membutuhkan akses terhadap Java Development Kit (JDK) yang berisi Java compiler dan class Java. JDK juga berisi Java Runtime Environment (JRE) untuk menjalankan class Java yang telah dikompilasi.



Gambar 5 Mekanisme Java web application

Java web application memiliki arsitektur yang terbagi menjadi tiga layer, yakni presentation layer, business rules layer, dan data access layer. Ketiga layer saling terhubung satu sama lain.



Gambar 6 Arsitektur Java web application

Presentation layer terdiri dari file HTML dan JSP. Sedangkan business rules layer berisi servlet yang mengendalikan alur aplikasi. Servlet memanggil Java class untuk menyimpan dan mengakses data dari database, serta memberikan hasilnya ke JSP atau servlet lain. Data layer mengelola data yang tersimpan di web server. Data ini dapat disimpan di file Extensible Markup Language (XML), file biner, ataupun file teks.

#### 4. Java Servlet

Java Servlet atau Jakarta Servlet merupakan program Java yang berjalan di web server. Servlet menangani request dari web server, memprosesnya, membuat response, lalu mengirimkannya kembali ke web server. Java Servlet digunakan untuk membuat halaman web yang dinamis. Servlet berbentuk class dalam bahasa

pemrograman Java yang dapat memperluas kemampuan server dalam model request-response. Contohnya, Servlet dapat menerima input user dari form HTML.

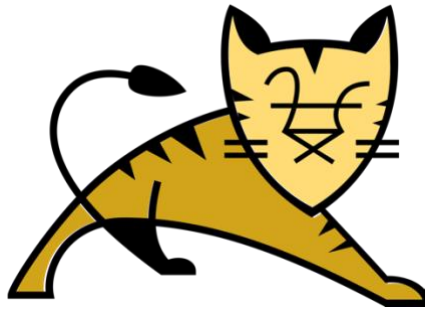
### 5. Java Server Page

Java Server Page atau JSP merupakan kode Java yang disisipkan di dalam kode HTML melalui tag JSP. Ketika JSP diproses oleh server, JSP engine akan mengkonversikan kode JSP menjadi servlet lalu mengkompilasinya. Kemudian JSP engine akan memuat servlet ke servlet engine untuk dijalankan. Pengembangan Java web app menggunakan kombinasi servlet dan JSP. Berikut kode JSP yang disisipkan ke dalam HTML.

```
<!DOCTYPE html>
<html>
<head>
...
</head>
<body>
  <% String fullName = request.getParameter("fullName"); %>
</body>
</html>
```

### 6. Apache Tomcat

Apache Tomcat merupakan HTTP server yang menjadikan user dapat menjalankan Servlet. Dengan kata lain, Tomcat adalah server untuk aplikasi web Java. Apache Tomcat dikembangkan oleh Apache Foundation sebagai web server untuk aplikasi yang ditulis dalam bahasa pemrograman Java. Tomcat menyediakan HTTP web server environment yang dapat menjalankan program Java untuk pengembangan web.



Gambar 7 Logo Apache Tomcat