

Nama : Nur Hikmah
NIM : 2310817120010
Mata Kuliah : Pemrograman Mobile

Tugas 1:

Bagaimana penggunaan HTTP Request dalam pemrograman mobile? Lakukan riset mengenai berbagai metode HTTP Request dan bagaimana penerapannya dalam pengembangan aplikasi mobile. sertakan artikel yang dapat mendukung opini kamu

HTTP Request (*HyperText Transfer Protocol Request*) adalah permintaan yang dikirim oleh klien (misalnya, browser web atau aplikasi mobile) ke server untuk mendapatkan akses sumber daya di internet melalui protokol HTTP (Rohmah, 2024). Dalam pengembangan aplikasi mobile, HTTP Request digunakan untuk menghubungkan aplikasi dengan server guna mengirim dan menerima data (Li, Lyu, Gui, & J. Halfond, 2016). HTTP Request dalam pemrograman mobile digunakan untuk komunikasi antara aplikasi dan server, memungkinkan aplikasi untuk mengambil atau mengirim data melalui internet. Ini penting dalam aplikasi yang memerlukan API (*Application Programming Interface*), seperti aplikasi berita, media sosial, atau e-commerce.

A. Struktur HTTP Request

HTTP Request terdiri dari beberapa bagian utama, yaitu:

1. Request Line

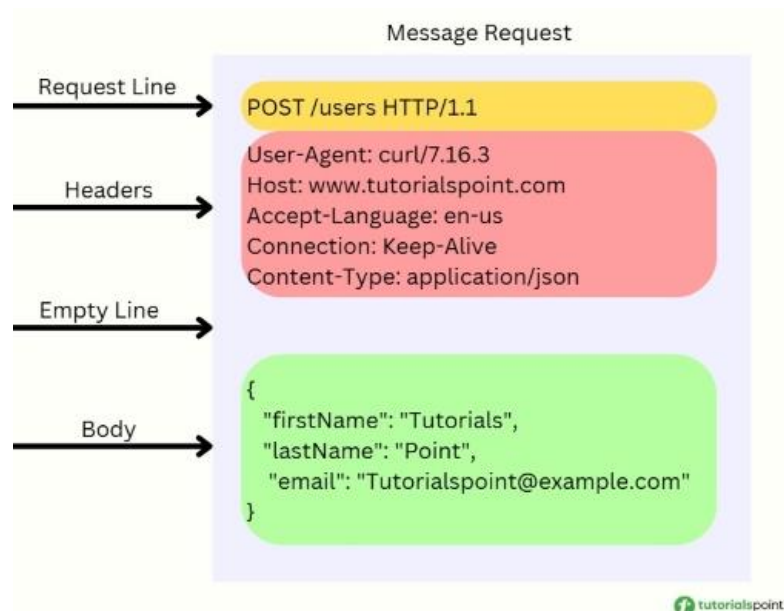
Menentukan metode HTTP (GET, POST, PUT, dll.), URL, dan versi protokol.

2. Headers

Berisi informasi tambahan seperti tipe konten (Content-Type), autentikasi (Authorization), dan agen pengguna (User-Agent).

3. Body (Opsional)

Digunakan dalam metode seperti POST atau PUT untuk mengirim data ke server.



Gambar 1. Struktur HTTP Request

Sumber: https://www.tutorialspoint.com/http/http_requests.htm

B. Metode yang digunakan dalam HTTP Request ialah:

- GET

Digunakan untuk mengambil data dari sumber daya tertentu. Ini ialah permintaan hanya baca, yang berarti tidak boleh mengubah data apa pun di server. Misalnya, aplikasi seluler dapat menggunakan permintaan GET untuk mengambil informasi profil pengguna.

Contoh sintaksisnya:

```
GET /test HTTP/1.1
Host: abe.com
User-Agent: Mozilla/5.0
Accept: application/json
Connection: keep-alive
```

- POST

Digunakan untuk mengirimkan data ke server untuk diproses. Misalnya, ketika aplikasi seluler meminta pengguna untuk membuat akun baru, aplikasi mengirimkan permintaan POST ke server dengan informasi pengguna.

Contoh sintaksisnya:

```
POST /test HTTP/1.1
```

```
Host: abe.com
Content-Type: application/x-www-form-urlencoded
Content-Length: 27
```

```
field1=value1&field2=value2
```

- **PUT**

Digunakan untuk memperbarui data yang sudah ada di server. Data yang akan diperbarui termasuk dalam badan permintaan. Misalnya, aplikasi seluler dapat menggunakan permintaan PUT untuk memperbarui informasi profil pengguna.

Contoh sintaksisnya:

```
PUT /test HTTP/1.1
Host: abe.com
Content-Type: text/html
Content-Length: 29
```

```
field1=newvalue1&field2=newvalue2
```

- **DELETE**

Menghapus data dari server. Misalnya, aplikasi seluler dapat menggunakan permintaan DELETE untuk memungkinkan pengguna menghapus akunnya.

Contoh sintaksisnya:

```
DELETE /test HTTP/1.1
Host: abe.com
Connection: keep-alive
```

- **HEAD**

Metode ini digunakan untuk mengambil metadata, tetapi tanpa isi body respons.

Contoh sintaksisnya:

```
HEAD /test HTTP/1.1
Host: abe.com
User-Agent: Mozilla/5.0
Accept: application/json
Connection: keep-alive
```

- PATCH

Metode ini digunakan untuk memperbarui sebagian data pada server.

Contoh sintaksisnya:

```
PATCH /test HTTP/1.1
```

```
Host: abe.com
```

```
Content-Type: application/json
```

```
Content-Length: 17
```

```
field1=patchedValue
```

- OPTIONS

Digunakan untuk mengetahui metode HTTP yang didukung server atau menanyakan opsi yang tersedia.

Contoh sintaksisnya:

```
OPTIONS /test HTTP/1.1
```

```
Host: abe.com
```

```
Connection: keep-alive
```

- CONNECT

Metode ini jarang digunakan dalam aplikasi web biasa. Biasanya digunakan untuk membuat terowongan atau mengatur koneksi aman melalui proxy.

Contoh sintaksisnya:

```
CONNECT abe.com:443 HTTP/1.1
```

```
Host: abe.com:443
```

```
Proxy-Connection: keep-alive
```

- TRACE

Mengembalikan permintaan asli yang diterima server, sering digunakan untuk debugging.

Contoh sintaksisnya:

```
TRACE /test HTTP/1.1
```

```
Host: abe.com
```

```
User-Agent: Mozilla/5.0
```

Connection: keep-alive

Sumber:

https://www.w3schools.com/tags/ref_httpmethods.asp

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods/GET>

C. Penerapan dalam Pengembangan Aplikasi Mobile

Di Android, terdapat beberapa cara untuk melakukan HTTP Request, diantaranya yaitu:

- HttpURLConnection

Kelas bawaan yang mendukung komunikasi dasar melalui HTTP/HTTPS.

- Retrofit

Library populer dari Square yang menyediakan API yang lebih deklaratif dan mendukung deserialisasi otomatis untuk berbagai format data seperti JSON.

- Volley

Library dari Google yang cocok untuk permintaan jaringan yang cepat dan efisien.

- Cronet

Library berbasis Chromium yang dioptimalkan untuk performa tinggi dan mendukung HTTP/3.

Sumber: <https://developer.android.com/develop/connectivity/cronet/start?hl=id>

Di iOS, HTTP Request dilakukan menggunakan:

- NSURLSession

API utama untuk menangani komunikasi jaringan, mendukung berbagai jenis request termasuk yang berjalan di background.

- Alamofire

Library berbasis Swift yang mempermudah penggunaan HTTP Request dengan fitur seperti serialization otomatis dan request chaining.

- Dukungan HTTP/3

iOS kini mendukung HTTP/3 yang lebih cepat dan lebih aman dibanding HTTP/2.

Sumber:

<https://developer.apple.com/library/archive/documentation/NetworkingInternetWeb/Conceptual/NetworkingOverview/WorkingWithHTTPAndHTTPSRequests/WorkingWithHTTPAndHTTPSRequests.html>

Referensi

- Althubiti, S., Yuan, X., & Esterline, A. (2017). Analyzing HTTP Requests for Web Intrusion Detection. *KSU Proceedings on Cybersecurity Education, Research and Practice*.
- Li, D., Lyu, Y., Gui, J., & J. Halfond, W. (2016). Automated Energy Optimization of HTTP Requests for Mobile Applications. *IEEE International Conference on Software Engineering*, 249-260.
- Rohmah, M. (2024). *Memahami HTTP Request Beserta Kegunaanya*. Retrieved from <https://dibimbing.id/blog/detail/memahami-http-request-beserta-kegunaanya>