

1. Apa yang dimaksud dengan socket, bagaimana penerapan pada komunikasi antar mesin/pc, serta implementasi pada bahasa pemrograman java, kelas dan method apa saja yang digunakan? Bagaimana cara kerja socket untuk bagian client dan server?

Jawab

Socket adalah mekanisme komunikasi untuk pertukaran data antar aplikasi yang terdapat di dalam sebuah mesin maupun beda mesin dan pertukaran ini terjadi pada sebuah jaringan komputer. Socket juga bisa diartikan sebagai titik komunikasi dari lalu lintas komunikasi antar proses di dalam sebuah jaringan komputer.

Penerapan

Dalam berkomunikasi socket membutuhkan protocol, protocol yang dapat digunakan ada dua jenis yaitu TCP/IP dan UDP. Protocol TCP/IP diperuntukkan untuk komunikasi dua arah antara client dan server dengan mengesampingkan kecepatan, misalkan aplikasi chatting. Ketika client mengirimkan pesan ke server, maka client harus menunggu balasan dari server. Sedangkan Protocol UDP adalah jenis protocol untuk kebutuhan komunikasi satu arah dengan kecepatan tinggi, biasanya diterapkan pada aplikasi broadcast.

Contoh komunikasi sederhana adalah komunikasi antara komputer A dan komputer B. Baik komputer A maupun komputer B harus memiliki identitas unik, yang direpresentasikan oleh IP ising-masing. Komunikasi yang terjadi melalui port, sehingga baik komputer A maupun komputer B harus memiliki port yang dapat diakses satu sama lain.

Implementasi

<https://www.sinaungoding.com/java-socket-programming/>

kelas:

method:

ketika client mengirimkan pesan “hello server” akan dibalas oleh server dengan pesan “hello client”. Secara teknis bahwa ketika class GreetServer sudah diinstance dan method start() sudah dipanggil, ketika itu juga server akan siap menerima request dari client.

2. di dalam java.rmi terdapat kelas Naming jelaskan penggunaan method bind() dan rebind(), lookup()

3.

4. Apa SOAP, RESTful web services?, Apa keuntungan dari SOAP web service, dan RESTful web services?

Apa perbedaan antara SOAP dan RESTful Web Service ? (25%)

SOAP, mulanya didefinisikan sebagai Simple Object Access Protocol, adalah sebuah spesifikasi protokol untuk pertukaran pesan/informasi terstruktur dalam implementasi web servis di jaringan komputer. SOAP menggunakan Extensible Markup Language (XML) sebagai format pesannya, dan biasanya bergantung pada protokol layer aplikasi lainnya, terutama Hypertext Transfer Protocol (HTTP) dan Simple Mail Transfer Protocol (SMTP), untuk transmisi dan negosiasi pesan.

Web servis RESTful (disebut juga web API RESTful) adalah sebuah web servis yang diimplementasikan menggunakan HTTP dan prinsip-prinsip REST. Web servis RESTful merupakan kumpulan resource yang aspek-aspek sebagai berikut:

Kelebihan SOAP

- bahasa, platform, dan transport agnostic
- dirancang untuk menangani lingkungan komputasi terdistribusi
- merupakan standar yang berlaku untuk web servis, sehingga mempunyai dukungan yang lebih baik dari standar yang lain (WSDL, WS-*) dan tools dari berbagai vendor
- built-in error handling (faults)
- extensibility

Kelemahan SOAP

- secara konseptual lebih sulit, lebih "heavy-weight" dibanding REST
- lebih "verbose" (membutuhkan lebih banyak pernyataan/kode program)
- sulit untuk dikembangkan, mebutuhkan tools

Keuntungan REST

- bahasa dan platform agnostic
- lebih sederhana/simpel untuk dikembangkan ketimbang SOAP
- mudah dipelajari, tidak bergantung pada tools
- ringkas, tidak membutuhkan layer pertukaran pesan (messaging) tambahan
- secara desain dan filosofi lebih dekat dengan web

29/10/2017

Perbedaan SOAP dan REST

Adapun perbedaan yang terdapat pada SOAP dan REST dapat dijelaskan pada tabel berikut:

Urutan	SOAP	REST
(1)	(2)	(3)
Protokol komunikasi	HTTP, HTTPS, SMTP, FTP	HTTP, HTTPS
Penggunaan bandwidth	Dalam jumlah request yang banyak, relatif boros bandwidth. Hal ini karena banyaknya markup dalam penulisan format XML	Relatif hemat bandwidth, karena markup-markup ekstra seperti pada XML tidak dipakai
Tipe penggunaan	Banyak mulai beralih ke REST, meski masih tetap ada yang mempertahankan, misalnya untuk integrasi aplikasi ke sistem legacy pada sebuah perusahaan	Mulai populer, banyak dipakai oleh penyedia web servis terkemuka, seperti twitter, yahoo!, flickr, bloglines, technorati, google, amazon, ebay, dan
Aturan penulisan	Ketat, mengikuti spesifikasi XML (SOAP v1.2)	Tidak ada spesifikasi khusus
Format respon	XML dengan spesifikasi SOAP. Agak sulit bagi kita untuk membaca langsung dan memahaminya	XML, JSON, atau format plain teks lainnya. Hal ini memudahkan penerima respon membaca dan memahaminya
Attachment file	Bisa (karena dapat mengembalikan respon dalam format binary)	Tidak bisa
Sifat web servis pada umumnya	Tertutup, lebih ditujukan untuk vendor atau perusahaan tertentu	Terbuka, bisa diakses siapa saja
Caching web	Relatif sulit	Modah, karena menggunakan URI
Penggunaan standar	Standar lama (XML, HTTP) dan baru (SOAP) digunakan bersamaan	Standar yang sudah ada, seperti XML dan HTTP
Tool pengembangan	Banyak, baik komersial maupun open source	Beberapa, karena tidak begitu dibutuhkan
Tool manajemen	Perlu, bahkan kadang harganya mahal	Menggunakan tool yang sudah ada pada sistem jaringan
Ektensiibel	Bisa, banyak ekstensi termasuk standar WS-*	Relatif tidak ektensiibel
Kemudahan implementasi	Mudah jika kita sudah memiliki lingkungan berbasis SOAP	Modah

Informasi lebih lanjut mengenai SOAP dan REST bisa dilihat pada website masing-masing

- SOAP (<http://www.w3.org/TR/soap>)
- REST (http://www.jcu.ac.au/~fielding/talks/webarch_9802)

Pendidik BPS : <http://pusdiklat.bps.go.id>

versi Online : <http://pusdiklat.bps.go.id/index.php?n=artikel&view&id=106>

<https://www.mediainformasi.my.id/2018/08/perbedaan-antara-soap-dan-rest-sebagai.html>