

BAB II. KAJIAN PUSTAKA

A. Rapor

Rapor berasal dari kata dasar *report* yang berarti laporan. Rapor merupakan laporan hasil dari suatu kegiatan yang disusun secara benar. Materi yang dilaporkan dalam hal ini adalah hasil ulangan harian, tugas harian, ujian tengah semester, ujian akhir semester, kepribadian, ekstrakurikuler beserta data yang diperlukan yang berkaitan dengan rapor (Pratiwi 2009).

B. Konsep Web Service

1. Pengertian *Web Service*

Web service adalah sebuah entitas komputasi yang dapat diakses melalui jaringan internet maupun intranet dengan standar protokol tertentu dalam *platform* dan antar muka bahasa pemrograman yang independen. Tujuan pengembangannya adalah untuk menjembatani komunikasi antar program sehingga aplikasi yang satu dan aplikasi yang lain yang terdapat pada suatu jaringan yang sama atau pada jaringan berbeda dapat saling berkomunikasi asalkan menggunakan standar protokol yang ditetapkan oleh *web service*. Hal ini bisa terjadi, karena standar protokol itu tidak terikat pada suatu *platform* atau bahasa pemrograman. Protokol itu sendiri dibangun oleh *Extensible Markup Language* (XML) yang memang kenyataannya telah didukung oleh banyak *platform*, bahasa pemrograman dan oleh *developer* di seluruh dunia (Siregar dan Purba, 2012).

Web service ternyata sangat berbeda dengan *website*. Perbedaan yang paling terlihat adalah *website* dibuat untuk memiliki tampilan atau *user interface*

yang bagus sedangkan *web service* tidak memiliki tampilan, karena *web service* tidak dibuat untuk berinteraksi langsung dengan *user*. Sesuai dengan kata *service* yang ada pada namanya, *web service* hanya menyediakan layanan. Layanan tersebutlah yang kemudian akan digunakan atau dipanggil oleh aplikasi lainnya. Dengan demikian yang akan menjadi *interface* adalah aplikasi yang memanggilnya bukan *web service* itu sendiri (Lucky, 2008).

Dari kedua penjelasan diatas dapat disimpulkan bahwa *web service* adalah komponen yang didesain untuk menjembatani pertukaran data antar aplikasi dengan *database* yang biasanya diserialisasi dengan menggunakan XML (*Extensible Markup Language*) dan dapat diakses melalui protokol terbuka yaitu SOAP (*Simple Object Access Protocol*) dengan bahasa WSDL (*Web Service Description Language*) dan terdaftar dalam UDDI (*Universal Discovery Description and Integration*).

2. Komponen Pendukung *Web Service*

a. XML (*Extensible Markup Language*)

Deviana (2011) menyatakan bahwa XML merupakan dasar terbentuknya *web service* yang digunakan untuk mendeskripsikan data. Pada level paling detail *web service* secara keseluruhan dibentuk diatas XML. Fungsi utama dari XML adalah komunikasi antar aplikasi, integrasi data, dan komunikasi aplikasi eksternal dengan partner luaran. Dengan standarisasi XML, aplikasi-aplikasi yang berbeda dapat dengan mudah berkomunikasi antar satu dengan yang lain.

Feature-feature yang ditawarkan XML yaitu (Purnamasari, 2008):

- 1) XML dapat menyimpan dan mengorganisir semua jenis informasi dalam bentuk yang kita sukai (dapat disesuaikan dengan kebutuhan).
- 2) Sebuah *open standard*, XML tidak terikat dengan perusahaan atau perangkat lunak manapun.
- 3) Dengan *Unicode* sebagai karakter set standar, XML mendukung berbagai macam sistem penulisan (*scripts*) dan simbol. dari karakter *Skandinavia* sampai *ideograf* bangsa *China Han*.
- 4) XML menawarkan berbagai cara untuk memeriksa kualitas sebuah dokumen dengan aturan *syntax*, *internal link checking*, perbandingan dengan modul dokumen, dan *datatyping*.
- 5) *Syntaks* XML sederhana dan tidak mempunyai struktur yang *ambigu*. Sehingga mudah dibaca oleh manusia maupun program.
- 6) XML mudah untuk dikombinasikan dengan *stylesheet* untuk membuat format dokumen sesuai dengan *style* yang kita inginkan.

b. SOAP (*Simple Object Access Protocol*)

SOAP merupakan suatu format standar dokumen berbentuk XML yang digunakan untuk melakukan proses *request* dan *response* antara *web service* dengan aplikasi yang memanggilmnya. Dokumen SOAP yang digunakan untuk melakukan *request* disebut dengan SOAP *request* sedangkan dokumen SOAP yang diperoleh dari *web service* disebut dengan SOAP *response* (Lucky, 2008).

c. WSDL (*Web Service Description Language*)

Web services description language (WSDL) merupakan protokol berbasis XML yang digunakan untuk mendeskripsikan suatu *web service* dan

memfasilitasi *application-to-application communication*. Suatu *file* WSDL mendeskripsikan apa yang dilakukan oleh *service* dan bagaimana untuk memanggil (*invoke*) operasi dari *service* dimana *service* tersebut dapat ditemukan dalam jaringan/Internet, dan *interface-interface*-nya (Wellem, 2009).

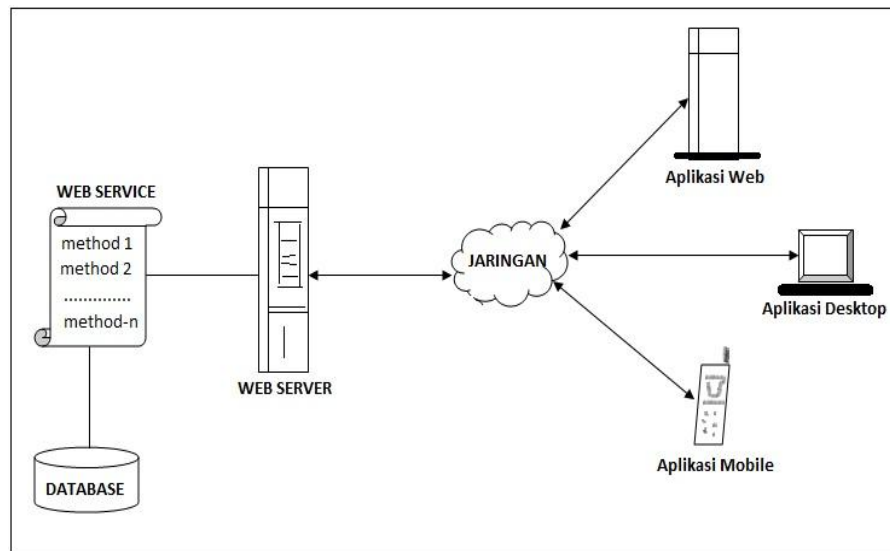
d. UDDI (*Universal Discovery Description and Integration*)

Lucky (2008) menyatakan bahwa UDDI merupakan suatu *directory service* yang digunakan untuk meregistrasikan dan mencari *web srvice*.

3. Penerapan *Web Service*

Web service itu digunakan saat kita akan mentransformasi sebuah bisnis *logik* / sebuah *class* dan *object* yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dan *security* dapat di tangani dengan baik selain itu, *web service* juga lebih mudah dalam proses *deployment*, karena tidak memerlukan registrasi khusus ke dalam sistem operasi dan *web service* cukup di-*upload* ke *web server* dan siap diakses oleh pihak-pihak yang telah diberikan otorisasi. *Web service* berjalan di *port* 80 yang merupakan protokol standar HTTP, dengan demikian mengurangi resiko terblokir oleh *firewall*. Kendala arsitektur COM/DCOM adalah memerlukan konfigurasi khusus di sisi *firewall* dan ini tidak perlu dilakukan untuk mengakses *web service* (Daniel, 2003).

Penerapan *web service* tentunya memudahkan perangkat-perangkat yang menggunakan aplikasi yang berbeda-beda dapat saling betukar data seperti pada gambar berikut (Gambar 1).



Gambar 1. Penerapan *Web Service* Untuk Pertukaran Data Antara Aplikasi Yang Berbeda-Beda

Pada Gambar 1, dapat disimpulkan bahwa dalam meenerapkan *web service* terdapat tiga lapisan yaitu:

a. Lapisan *Data Provider*

Lapisan ini bertanggung jawab menyimpan data dan menyediakan data yang akan diberikan ke lapisan *user interface*.

b. Lapisan *Business Service*

Lapisan ini mengendalikan semua data yang diakses dan meng-*update* data yang ada dalam *basis data*. lapisan ini biasanya dapat digunakan oleh modul-modul lain dalam aplikasi.

c. Lapisan *User Interface*

Lapisan *user interface* (tampilan antar muka) merupakan bagian aplikasi yang berinteraksi langsung dengan *user*.

Tujuan pemisahan kedalam lapisan-lapisan tersebut adalah apabila suatu saat nanti terdapat perubahan pada aplikasi yang telah dibuat itu tidak perlu

membongkar keseluruhan dari aplikasi tersebut tetapi cukup dirubah pada salah satu lapisan (*layer*) yang diperlukan saja.

4. Keunggulan Dan *Fleksibilitas Web Service*

Penggunaan *web service* juga menawarkan banyak kelebihan dan *fleksibilitas*. Beberapa diantaranya adalah sebagai berikut (Lucky, 2008):

a. Lintas *Platform*

Penggunaan *web service* memungkinkan komputer-komputer yang berbeda sistem operasi dapat saling bertukar data.

b. *Language Independent*

Sebuah *web service* dapat diakses menggunakan bahasa pemrograman apa saja.

c. Jembatan Penghubung Dengan *Database*

Web service dapat dijadikan sebagai jembatan penghubung antara aplikasi dengan *database* tanpa memerlukan *driver database* dan tidak perlu tahu apa jenis *DBMS* yang digunakan pihak sistem informasi lain. Aplikasi tersebut cukup mengetahui fungsi apa saja yang disediakan *web service* untuk memanfaatkan fasilitasnya.

d. Mempermudah Proses Pertukaran Data

Penggunaan *web service* dapat mempermudah dan mempercepat pertukaran data, tidak harus menyesuaikan aplikasi, *database* atau *platform* yang digunakan.

e. Penggunaan Kembali Komponen Aplikasi

Dengan adanya *web service* beberapa aplikasi yang berbeda bisa saja memerlukan sebuah fungsi yang sama.

C. Tampilan Antar Muka Pemakai (*user interface*)

User interface merupakan mekanisme komunikasi antara *user* dengan sistem. Antarmuka pemakai (*user interface*) dapat menerima informasi dari *user* dan memberikan informasi kepada *user*. *User interface* sangat berperan penting dalam dunia komputer karena dengan adanya *user interface* maka kemudahan *user* dalam mengoperasikan suatu perangkat komputer menjadi lebih mudah.

1. *Windows Form Application*

Windows forms atau juga dikenal sebagai *winforms* merupakan nama yang diberikan pada API (*application programming interface*) untuk aplikasi antar muka grafis GUI (*graphical user interface*) yang termuat dalam *framework .NET* yang diluncurkan oleh *microsoft corp.* *winforms* merupakan blok-blok bangunan dasar untuk pengembangan antar muka (Nugroho, 2010).

2. *Web Application*

Aplikasi *web* merupakan halaman dinamis yang mengizinkan interaksi dengan *user* (*user* melakukan sesuatu). Interaksi *user* dengan aplikasi *web* misalnya *user* mengklik sebuah tombol dan warna latar belakang *web* berubah. Aplikasi *web* biasanya dibuat dengan menggunakan *JavaScript*, aplikasi *flash*, atau *applet* pada *java* (Shalahuddin dan Rosa, 2008).

3. *Mobile Application*

Aplikasi *mobile* adalah aplikasi yang berjalan di dalam sebuah *Mobile device* (*PDA* ataupun *Smart Phone*). *Android* adalah sistem operasi untuk telepon seluler yang berbasis *linux*. *Android* menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. Awalnya *Google Inc* membeli *android Inc*, pendatang baru yang membuat

piranti lunak untuk ponsel, kemudian untuk mengembangkan *android* dibentuk *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, piranti lunak dan telekomunikasi termasuk *Google, HTC, Intel, Motorola, Qualcomm, T-Mobile* dan *Nvidia* (Kurniawan 2011).

D. Hasil Penelitian Sejenis

Beberapa penelitian tentang pengembangan *web service* yang telah dilakukan:

1. Hakim dan Wahid (2005) telah melakukan penelitian tentang Penerapan Arsitektur *Three-Tier* Dengan COM+ yang membahas konsep *three-tier* pada kasus Portal Jurnal. Beberapa teori yang digunakan sebagai dasar sangat terpengaruh dari sumber referensi *Microsoft*, yaitu MSDN (*MicroSoft Developer Network*). Kemudian *middleware* yang digunakan adalah COM+.
2. Priawan (2013) mengembangkan *web service* yang di implementasikan pada aplikasi kartu hasil studi berbasis *android*. Dalam penerapannya menggunakan *web service* jenis REST (*representational state transfer*) yang merupakan penyederhanaan dari HTTP (*hypertext transfer protocol*).