

## **IMPLEMENTASI WEB SERVICE BERBASIS WEB TERINTEGRASI SERVICE WISMA MM UGM AND CONVENTION**

<sup>1</sup>Arwanto (10018083), <sup>2</sup>Imam Riadi(0510088001)

<sup>1,2</sup>Program Studi Teknik Informatika

Universitas Ahmad Dahlan

Prof. DR. Soepomo, SH, Janturan, Umbulharjo, Yogyakarta 55516

<sup>1</sup>Email : arwanboy@gmail.com

<sup>2</sup>Email : imam.riadi@is.uad.ac.id

### **Abstrak**

*Pesatnya perkembangan teknologi komputer di negara maju, membuat para peneliti semakin ingin mencoba tenaga komputasi yang baru. Organisasi atau lembaga banyak yang menggunakan layanan web service. Penggunaan layanan web service diantaranya beberapa lembaga pendidikan, perkantoran dan masih banyak lagi yang lainnya. Integrasi sistem karena kendala seperti keterbatasan atau pemborosan akses data yang menyebabkan keterlambatan beberapa kegiatan pengkomputasian. Permasalahan tersebut menyebabkan akses data dari database server membutuhkan waktu yang cukup lama.*

*Tahapan penelitian terdiri dari perancangan sistem, perancangan jaringan, implementasi sistem dan pengujian sistem. Langkah pengembangan sistem yang meliputi perancangan jaringan dan perancangan sistem. Integrasi sistem dibangun dengan framework Code Igniter untuk memudahkan developer dalam pengembangan selanjutnya. Pengujian sistem menggunakan uji kelayakan pada keadaan sebelum dan sesudah diterapkannya layanan web service.*

*Berdasarkan hasil dari pengujian dengan metode uji kelayakan diperoleh hasil 67% responden menyatakan sangat setuju 8% menyatakan setuju dan 25% menyatakan tidak setuju bahwa implementasi web service yang diintegrasikan pada aplikasi pendataan di Wisma MM UGM and Convention. Berdasarkan hasil pengujian dapat diambil kesimpulan bahwa aplikasi yang dikembangkan mampu menghubungkan dua database sehingga proses pembuatan laporan per periode lebih cepat dan efisien.*

**Kata Kunci :** Integrasi, Sistem, Web, Service dan Maintenance

### **1. PENDAHULUAN**

Informasi termasuk bagian penting dan sangat berharga di dalam dunia usaha dan kerja. Informasi yang akurat dan tepat waktu akan membantu dalam

mengambil keputusan dan menentukan langkah-langkah yang harus dilakukan untuk mempertahankan dan mengembangkan organisasi.

Wisma MM UGM Hotel and Convention termasuk dalam perusahaan yang bergerak dalam bidang perhotelan yang ada di Indonesia. Berbagai pendataan transaksi dan *reservasi* kamar tersedia pada Wisma MM UGM Hotel and Convention. Banyaknya data dan informasi yang perlu ditangani dari berbagai tempat yang letaknya jauh, maka perlu suatu akses *database public*, sehingga semua pembaruan data yang berhubungan dengan transaksi dan pemesanan kamar hotel bisa diketahui oleh semua pihak.

Sistem *database* yang ada pada Wisma MM UGM Hotel and Convention sampai saat ini masih bersifat *local*, sehingga dari pihak kantor yang lain belum dapat melihat hasil transaksi yang baru. Proses melihat data transaksi yang baru seorang pegawai harus mendatangi kantor yang berada di Wisma MM UGM and Convention berulang kali, misalnya perubahan data baru hasil transaksi perhari, maka akan banyak membutuhkan waktu lama. Berdasarkan masalah tersebut, maka dibuatlah suatu penelitian untuk mengembangkan aplikasi dengan mengintegrasikan layanan *web service* yang mampu untuk menyelesaikan masalah pengaksesan *database*.

## 2. KAJIAN PUSTAKA

Penelitian yang dilakukan oleh Fahmi Rizal, 2012, Teknik Informatika, Universitas Ahmad Dahlan Yogyakarta. Menjelaskan *web service* memberikan kemudahan dan lebih menghemat waktu bagi siapa saja untuk mengetahui letak buku yang sedang dicari dengan judul : implementasi *web service* pada database terdistribusi.[1]

Penelitian yang dilakukan Ery Dianawati, 2013, Teknik Informatika, Universitas Ahmad Dahlan Yogyakarta. Menjelaskan penerapan *web service* dengan arsitektur REST sebagai sarana pertukaran data antara web afiliiasi dan web toko *online* dengan judul : implementasi *distributed computing* untuk pengembangan *web store* dengan *affiliate sistem*. [2]

Penelitian yang dilakukan Fajril Asri, 2010, Teknik Informatika, Universitas Ahmad Dahlan Yogyakarta. Menjelaskan cara pertukaran data antar aplikasi dapat dilakukan dengan *web service* yang memberikan keamanan data dalam melakukan proses pertukaran, pengiriman dan penerimaan data dengan judul : implementasi *security* pada *web service nusoap* dengan menggunakan algoritma RIJNDAEL/AES.[3]

### 2.1. Web Service

*Web Service* memungkinkan kita dapat mengakses informasi dari suatu aplikasi melalui aplikasi yang lain. Secara umum *web service* dapat diidentifikasi dengan menggunakan URL seperti hanya web pada umumnya. Perbedaan *web service* dengan web yang interaksi dilihat dari URL web, URL *web service* hanya mengandung kumpulan

informasi, perintah, konfigurasi atau *sintaks* yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi. *Web service* dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan sebuah *database* ditanamkan, dibuat dalam bahasa pemrograman yang mengkonsumsi data, dan di *platform* sebuah data itu dikonsumsi. *Web service* mampu menunjang interoperabilitas. *Web service* mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada.[4]

## 2.2. Data Flow Diagram (DFD)

*Data Flow Diagram* (DFD) awalnya dikembangkan oleh Chris Gane dan Trish Sarson pada tahun 1979. *Data Flow Diagram* digunakan untuk menggambarkan asal dan tujuan data yang keluar dari sistem, tempat data disimpan, proses yang menghasilkan data tersebut dan interaksi data yang tersimpan dan proses. DFD difokuskan pada entitas dan data proses.[5]

## 2.3. Entity Relationship Diagram (ERD)

Pemodelan awal basis data yang paling banyak digunakan untuk *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD termasuk model konseptual yang mendeskripsikan hubungan antara penyimpan (dalam DFD). ERD digunakan untuk memodelkan struktur data dan hubungan antar data. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data.[6]

## 2.4. Framework CodeIgniter

*CodeIgniter* dirilis pertama kali pada 28 Februari 2006, versi stabil terakhir 2.1.0 yang dirilis pada 28 November 2011. *CodeIgniter* termasuk sebuah kerangka atau *framework* pembangunan aplikasi atau mudahnya disebut *toolkit*, untuk *developer* yang akan membuat aplikasi web dengan PHP. Tujuan *CodeIgniter* supaya pembangunan aplikasi lebih cepat dibanding menulis *source code* dari awal, karena CI telah menyediakan banyak *library* untuk proses-proses yang sering digunakan pada suatu aplikasi, dan juga dengan kemudahan dalam menggunakan *library* tersebut serta kesederhaan penggunaannya.[7]

# 3. METODE PENELITIAN

## 3.1. Metode Pengumpulan Data

### 3.1.1. Studi Observasi

Metode pengumpulan data dilakukan dengan pengamatan secara langsung pada Wisma MM UGM and Convention mengenai sistem yang digunakan untuk melayani *customer*. Hasil sistem yang dikembangkan nantinya diharapkan dapat mengatasi segala permasalahan yang diakibatkan oleh pusat, seperti *update* data

terbaru, kecepatan transfer data jauh, dan waktu yang diperlukan dalam transfer data yang lama.

#### 3.1.2. Metode Wawancara

Metode pengumpulan data dengan cara melakukan wawancara langsung kepada narasumber tentang materi yang terkait dengan sistem hotel dan transaksi kerja beserta berbagai permasalahannya. Hasil penelitian diharapkan dapat membenahi permasalahan sesuai fakta yang ada.

#### 3.1.3. Studi Literatur

Metode pengumpulan data yang dilakukan dengan mencari, membaca dan mengumpulkan dokumen sebagai referensi seperti buku, artikel dan literatur. Studi tersebut berguna dalam penentuan topik dan dasar teori penelitian. Berdasarkan literatur-literatur yang dibaca tersebut, dapat diketahui perkembangan teknologi, permasalahan penelitian yang telah dilakukan dan pengimplementasian *web service*.

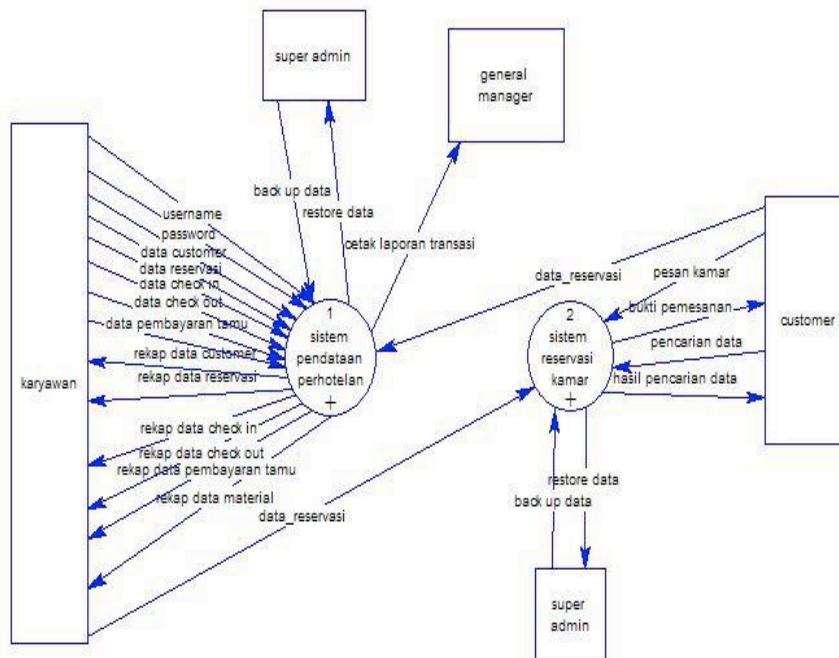
### 4. HASIL DAN PEMBAHASAN

#### 4.1. Perancangan Sistem

##### 4.1.1. DFD (Data Flow Diagram)

Langkah-langkah yang digunakan dalam sistem ini dengan membuat perancangan diagram konteks dan diagram aliran data untuk mempermudah dalam membuat sistem. Diagram alir data difokuskan pada entitas dan data proses. Entitas digunakan untuk menerapkan integritas. Proses data untuk menggambarkan alur aktifitas yang ada di sistem dari mulai input sampai output. Proses yang dilakukan dalam sistem ini digambarkan menggunakan DFD (Data Flow Diagram).

DFD menjelaskan terdapat dua proses yang ada di dalam sistem tersebut diantaranya proses pendataan transaksi hotel dari aplikasi *server* dan proses pemesanan kamar dari aplikasi *client*. Dapat terlihat pada Gambar 1.

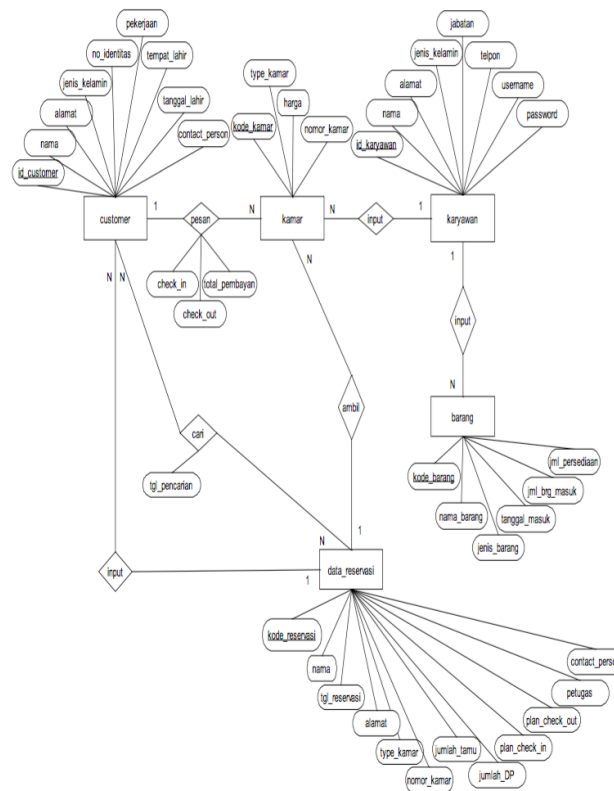


Gambar 1. Diagram Konteks

#### 4.1.2. ERD (*Entity Relationship Diagram*)

*Entity Relationship Diagram* (ERD) termasuk diagram yang memperlihatkan entitas-entitas yang terkait dengan sistem dan hubungan-hubungan antara entitas tersebut, sehingga perancangan basis data akan lebih mudah.

*Entity Relationship Diagram* (ERD) dimulai dari Entitas *customer* memesan kamar, *check in*, *check out* dan pembayaran kamar kemudian karyawan menginputkan data *customer* ke dalam sistem. *Customer* juga dapat melakukan pemesanan kamar melalui web klien, aplikasi yang khusus digunakan untuk pemesanan kamar. Web klien ini juga memberikan fasilitas fungsi melakukan pencarian lokasi data kamar. Fungsi ini ditujukan untuk mempermudah *customer* untuk melakukan pencarian lokasi kamar yang sedang dicari. Perancangan *Entity Relationship Diagram* (ERD) dapat dilihat pada Gambar 2.



Gambar 2. Perancangan ERD

## 4.2. Implementasi

### 4.2.1. Form Login

Tampilan *form* ini digunakan *login* admin untuk dapat mengakses aplikasi tersebut, admin harus memasukkan *username* dan *password* yang sudah didaftarkan sebelumnya. Admin memasukan *username* dan *password* dengan benar maka admin dapat mengakses aplikasi ini, apabila *username* dan *password* yang dimasukan salah maka gagal *login* dan memeriksa kembali apakah *username* sudah pernah dibuat atau belum. Tampilan *form login* dapat dilihat pada Gambar 3.



Gambar 3. Tampilan form Login

| No | Script  |
|----|---|
| 1  | <code>if (count(\$row)){</code>   |
| 2  | <code>\$_SESSION['userid'] = \$row['id'];</code>                              |
| 3  | <code>redirect('admin/dashboard','refresh');</code>                           |
| 4  | <code>}else{</code>   |
| 5  | <code>\$this-&gt;session-&gt;set_flashdata('result','maap</code>              |
| 6  | <code>username atau password Anda salah!');</code>                            |
| 7  | <code>redirect('template/login','refresh');}</code>                           |
| 8  | <code>}else{</code>   |
| 9  | <code>\$this-&gt;session-&gt;set_flashdata('result','Isi</code>               |
| 10 | <code>Username dan password dulu!');</code>                                   |
| 11 | <code>redirect('template/login','refresh');}</code>                           |
| 12 | <code>&lt;form name='login' action='&lt;?php echo</code>                      |
| 13 | <code>base_url();?&gt;index.php/template/ceklogin'</code>                     |
| 14 | <code>method='post' &gt;</code>   |
| 15 | <code>&lt;table align="center"&gt;</code>                                     |
| 16 | <code>&lt;center&gt;&lt;img border="0" src='&lt;?php echo</code>              |
| 17 | <code>base_url();?&gt;assets/images/a.jpg' width="100"</code>                 |
| 18 | <code>height="100"&gt;&lt;/center&gt;&lt;br&gt;</code>                        |
| 19 | <code>&lt;tr&gt;&lt;td&gt;Username&lt;/td&gt;&lt;td&gt;&lt;input</code>       |
| 20 | <code>placeholder='username' type='text'</code>                               |
| 21 | <code>name='username'&gt;</code>  |
| 22 | <code>&lt;tr&gt;&lt;td&gt;Password&lt;/td&gt;&lt;td&gt;&lt;input</code>       |
| 23 | <code>placeholder='password' type='password'</code>                           |
| 24 | <code>name='password'&gt;</code>  |
| 25 | <code>&lt;tr&gt;&lt;td&gt;&lt;/td&gt;&lt;td&gt;&lt;input type='submit'</code> |
| 26 | <code>value='Login'&gt;&lt;/td&gt;&lt;/tr&gt;</code>                          |

Listing 1 : *source code Login* ke dalam Sistem

Listing 1 menjelaskan pengecekan data yang dimasukkan untuk *login* sudah terisi semua dan yang dimasukkan benar atau salah, jika benar maka akan masuk ke dalam sistem, jika salah akan keluar peringatan atau instruksi untuk melakukan *login* ulang.

#### 4.2.2. Form Data Booking Terintegrasi Web Service

Tampilan data *booking* yang terintegrasi *Web Service* menampilkan halaman data *customer* yang melakukan pemesanan kamar dan lokasi data kamar yang diambil dari *database server* dan *client* terlihat pada Gambar 4 dengan dilihat pada listing 3.



Gambar 4. Data *booking* terintegrasi web service

| No | Script   |
|----|--|
| 1  | <code>\$json =</code>  |
|    | <code>file_get_contents('http://192.168.10.28/super/index.php/parsing_json/data_customer');</code>   |
| 2  | <code>\$json = json_decode(\$json, true);</code>   |
| 3  | <code>\$data_json = \$json['customer'];</code>   |
| 4  | <code>&lt;input type="text" class="span3" name="kode_reservasi" size="30" placeholder="Masukkan Kode Reservasi. . ." value=&lt;?php echo \$search['kode_reservasi'];?&gt;</code>                                       |
| 5  | <code>&lt;button type="submit" class="btn btn-primary" name="save" value="cari"&gt;&lt;i class="icon-search icon-white"&gt;&lt;/i&gt; Cari&lt;/button&gt;</code>   |
| 6  | <code>&lt;button type="submit" class="btn btn-primary" name="save" value="reset"&gt;&lt;i class="icon-book icon-white"&gt;&lt;/i&gt; Reset&lt;/button&gt;</code>   |
| 7  | <code>&lt;?php echo form_close(); ?&gt;</code>   |
| 8  | <code>&lt;ul class="nav"&gt;</code>  |
| 9  | <code>&lt;/div&gt;&lt;/ul&gt;</code>   |
| 10 | <code>&lt;?php</code>  |
| 11 | <code>if (\$this-&gt;session-&gt;flashdata('message')){</code>   |
|    | <code>echo "&lt;div class='message'&gt;". \$this-&gt;session-&gt;flashdata('message'). "&lt;/div&gt;";</code>  |
| 12 | <code>\$no = 1;</code>   |
| 13 | <code>echo "&lt;table id='table' class='table table-hover table-condensed'&gt;\n";</code>  |
| 14 | <code>echo "&lt;tr valign='top'&gt;\n";</code>   |
|    | <code>echo "&lt;th&gt;no&lt;/th&gt;\n&lt;th&gt;kode</code>   |
| 15 | <code>reservasi&lt;/th&gt;&lt;th&gt;nama&lt;/th&gt;&lt;th&gt;tanggal reservasi&lt;/th&gt;&lt;th&gt;jenis kamar&lt;/th&gt;&lt;th&gt;tgl in &amp; out&lt;/th&gt;&lt;th&gt;nomor kamar&lt;/th&gt;&lt;th&gt;contact</code> |



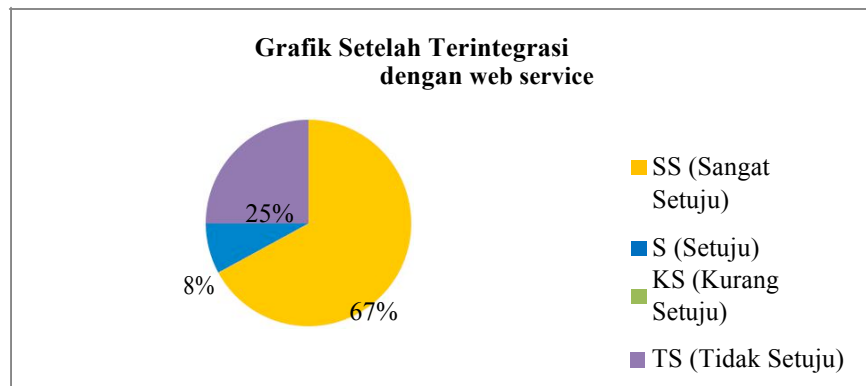
```
person</th>\n";  
echo "</tr>\n";  
16 if (count($customer)){  
    foreach ($customer as $key => $list){  
        echo "<tr valign='top'>\n";  
17        echo "<td> $no."; echo "</td>";  
18        echo "<td  
19        align='center'>". $list['kode_reservasi']. "</td  
        >\n";  
20        echo "<td  
21        align='center'>". $list['nama']. "</td>\n";  
        echo "<td  
        align='center'>". $list['tanggal_reservasi']. "<  
22        /td>\n";  
        echo "<td  
23        align='center'>". $list['jenis_kamar']. "</td>\n  
        ";  
        echo "<td align='center'>in:  
24        ". $list['plan_check_in']. " <br>out:  
        ". $list['plan_chek_out']. "</td>\n";  
25        echo "<td  
        align='center'>". $list['nomer_kamar']. "</td>\n  
26        ";  
27        echo "<td  
        align='center'>". $list['Contact_Person']. "</td  
28        >\n";  
29        echo "</tr>\n";  
30        $no++; }  
}
```

Listing 3 : *source code* data *booking* terintegrasi web service  
Listing 3 menjelaskan untuk menampilkan data pemesanan kamar yang terintegrasi dengan web service, sehingga dapat mengambil data dari *database server* dan *client*. Penggabungan data dari dua basis data ini agar pihak fakultas dapat mengaksesnya secara cepat dan akurat.

#### 4.2.3. Pengujian

Pengujian program dengan cara mengamati keluaran (*output*) program tanpa harus mengetahui segala yang dilakukan oleh sistem dengan berbagai macam masukan (*input*). Pengujian sistem ini dilakukan dengan metode pengujian sistem uji kelayakan. Skenario uji kelayakan pada penelitian ini dibagi menjadi dua tahap pengujian sebelum dan sesudah diintegrasikannya *web service*. Hasil evaluasi yang dilakukan oleh responden terhadap sistem, dapat diperoleh presentasi penilaian pengujian sangat setuju = 67%, setuju = 8% dan tidak setuju 25%.

Berdasarkan hasil pengujian yang diberikan terhadap 10 responden pengujian maka diperoleh hasil uji kelayakan yang dapat dilihat pada Gambar 5.



Gambar 5. Hasil pengujian sistem

## 5. KESIMPULAN

Sistem implementasi web service dikembangkan dengan metode *sekuensial linier (waterfall)* dengan langkah secara berurutan dari tahap pengumpulan data, analisis sistem, perancangan *interface*, pembuatan diagram aliran data, implementasi dan pengujian sistem. Berdasarkan pengujian sistem menggunakan uji kelayakan diperoleh hasil bahwa sangat setuju 67%, setuju 8% dan tidak setuju 25%. Berdasarkan hasil pengujian dapat diambil kesimpulan bahwa aplikasi yang dikembangkan mampu menghubungkan dua database sehingga proses pembuatan laporan per periode lebih cepat dan efisien.

## 6. DAFTAR PUSTAKA

- Rizal, Fahmi. 2012. "Implementasi *Web Service* pada Database Terdistribusi". Yogyakarta: Skripsi Informatika UAD.
- Dianawati, Ery. 2013. "Implementasi *Distributed Computing* untuk Pengembangan *Web Store*". Yogyakarta: Skripsi Informatika UAD.
- Asri, Fajril. 2010. "Implementasi *Security* pada *Web Service nusoap* dengan Menggunakan Algoritma *RIJNDAEL/AES*". Yogyakarta: Skripsi Informatika UAD.
- Siswo. 2004. "Pengenalan *Web Service*". Yogyakarta: Penerbit Erlangga.
- Firmansyah, Ikhsan Wahyudi. 2010. "Pengenalan Pemodelan Sistem". Jawa timur: Universitas Muhammadiyah Malang.
- Mardiyanto, Dwi Aji. 2004. "Pemodelan Data dalam Rekayasa Perangkat Lunak". Yogyakarta : Penerbit Gava Media.
- Heru. 2012. "Pengenalan *Code Igniter*". Yogyakarta : Penerbit Erlangga.