

NASKAH PUBLIKASI
IMPLEMENTASI REPRESENTATIONAL STATE TRANSFER
APLIKASI LAYANAN INFORMASI LOWONGAN
PEKERJAAN

PROYEK TUGAS AKHIR



Disusun oleh :

FAUZAN ANDRIYANTO

51304115151

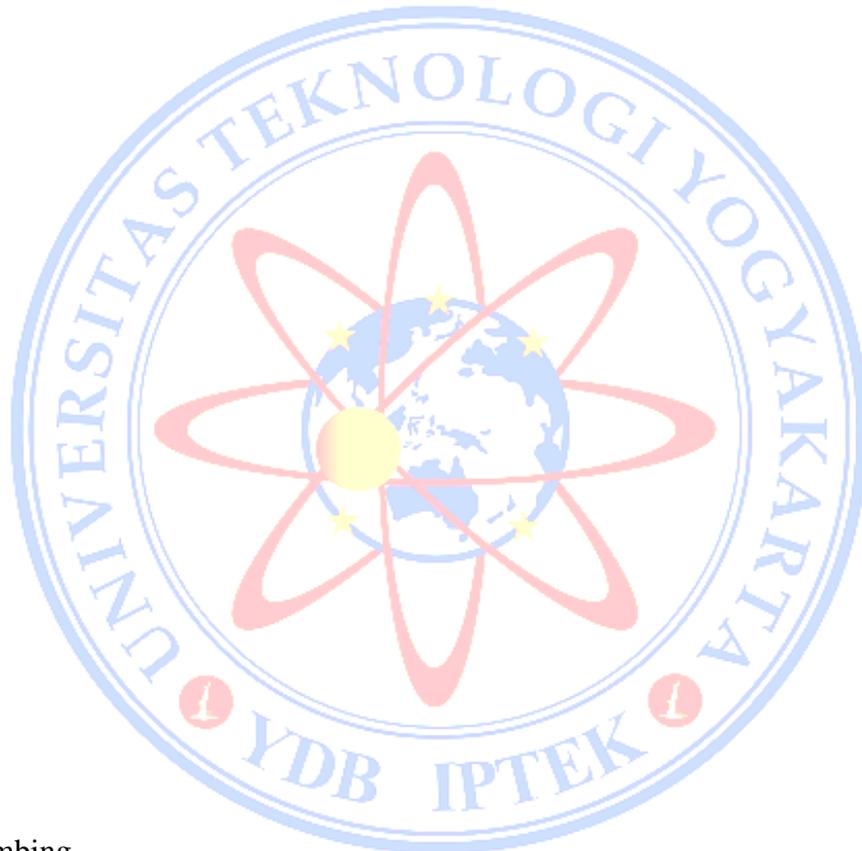
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO
UNIVERSITAS TEKNOLOGI YOGYAKARTA
2019

NASKAH PUBLIKASI
IMPLEMENTASI REPRESENTATIONAL STATE TRANSFER
APLIKASI LAYANAN INFORMASI LOWONGAN
PEKERJAAN

Disusun oleh :

FAUZAN ANDRIYANTO

51304115151



Pembimbing,

Tri Widodo, S.T., M.Kom.

Tanggal :

Implementasi Representational State Transfer Aplikasi Layanan Informasi Lowongan Pekerjaan

Fauzan Andri Yanto

Program Studi Teknik Informatika, Fakultas Teknologi Informasi dan Elektro

Universitas Teknologi Yogyakarta

Jl. Siliwangi (Ringroad Utara), Jombor, Sleman, Yogyakarta

E-mail : fauzan.tugas@gmail.com

ABSTRAK

Sebuah perusahaan harus mampu memanfaatkan kemajuan teknologi dan informasi sebagai salah satu sarana pendukung proses bisnis. Perusahaan membutuhkan otomatisasi bisnis proses dengan melakukan pertukaran dokumen dari aplikasi website. REST dalam arsitektur web service menggunakan protokol HTTP untuk pertukaran data. Pada penelitian ini akan disusun sebuah aplikasi dengan menerapkan teknologi web service untuk mengusulkan layanan informasi pekerjaan dengan dukungan protokol REST, dimana server dan client dapat berinteraksi dengan interface yang seragam, server akan menyediakan sumber data dan client mengelola sumber data yang disediakan server. Secara umum, sumber data yang dihasilkan berupa format JSON dan XML dan dapat bervariasi tergantung pada keinginan. Penelitian ini disusun agar proses bisnis yang terjadi pada saat pengguna (client) melakukan permintaan data seperti data lowongan, perusahaan, pencari kerja, dan lowongan berdasarkan kategori tertentu, kemudian dimanfaatkan client. Client pada aplikasi ini adalah aplikasi web. Setelah menerima data dari client, server kemudian menyebarkan informasi data kepada seluruh pengguna.

Kata Kunci : Lowongan Pekerjaan, Web Service, Representational State Transfer (REST), JSON, XML

ABSTRACT

A company must be able to take advantage of advances in technology and information as a means to support business processes. Companies need business process automation by exchanging documents from web site applications. REST in web service architecture uses HTTP protocol for data exchange. In this research an application will be prepared by applying web service technology to propose work information services with the support of the REST protocol, where the server and client can interact with a uniform interface, the server will provide a data source and the client manages the data source provided by the server. In general, the source of data generated is in the form of JSON and XML and can vary depending on desire. This research is structured so that business processes that occur when users (clients) make requests for data such as vacancy data, companies, job seekers, and vacancies based on certain categories, are then utilized by the client. Clients in this application are web applications. After receiving data from the client, the server then disseminates data information to all users.

Keywords : Lowongan Pekerjaan, Web Service, Representational State Transfer (REST), JSON, XML

1. PENDAHULUAN

REST (*representational state transfer*) merupakan standar arsitektur komunikasi berbasis web yang sering diterapkan dalam pengembangan layanan berbasis web. *server* menyediakan *resources* dan *client* mengakses dan menampilkan *resource* tersebut untuk penggunaan selanjutnya. Salah satu langkah yang di ambil adalah dengan membuat aplikasi lowongan kerja, yang di mana para pencari kerja dan penyedia kerja dapat bertemu dan bertukar informasi secara detail dan tidak di pungut biaya, cukup dengan melakukan pertukaran informasi melalui internet, untuk mengatasi para pencari kerja dalam mencari lowongan kerja yang sesuai kriteria. Lowongan pekerjaan merupakan salah satu layanan web yang sangat berguna bagi penggunanya dengan menyediakan fitur web *service* bagi para penggunanya. Sebagai seorang pengguna aplikasi layanan informasi lowongan pekerjaan, ketika mulai mengakses sebuah aplikasi kita akan mengirim suatu HTTP *request* ke *server*. *Request* tersebut kemudian diarahkan ke pemilik dari aplikasi tersebut. Ada suatu informasi seperti *user ID* dan sejenisnya yang akan di *request* dari *server* pemilik aplikasi ke *server* anggota yang terdaftar sebagai pencari kerja maupun perusahaan atau organisasi yang memiliki lowongan pekerjaan. Selanjutnya *server* akan merespons dan mengirimkan data sesuai dengan yang diminta.

1.1. Rumusan Masalah

Sesuai dengan latar belakang yang sudah disimpulkan masalah yang dirumuskan yaitu:

- Bagaimana membuat sistem aplikasi layanan informasi lowongan pekerjaan menggunakan *Representational State Transfer* ?
- Bagaimana membuat implementasi *Representational State Transfer* aplikasi layanan informasi lowongan pekerjaan yang mempermudah *Client* dalam mengakses informasi yang dibutuhkan ?
- Bagaimana membuat sistem implementasi *Representational State Transfer* sebagai sarana yang memudahkan *Client* dan *Server* ?

1.2. Batasan Masalah

Berdasarkan rumusan masalah yang telah disebutkan, penelitian ini memberikan batasan-batasan dengan tujuan agar peneliti tidak melakukan berbagai penyimpangan yang terlalu jauh dari judul. Adapun batasan-batasan yang disusun adalah sebagai berikut :

- Sistem akan memuat informasi lowongan kerja dalam beberapa kategori.
- Sistem akan memuat informasi kualifikasi dan persyaratan yang ditentukan oleh perusahaan.
- Bahasa pemrograman yang di gunakan sistem adalah PHP dan MySQL sebagai *database*, akan tetapi dalam penelitian ini dibatasi pada pengambilan *function-function* php di *web server* yang satu untuk dipakai di *web service* lain yang mengelola *website*.
- Metode yang akan digunakan adalah *Web Service* dengan metode REST.
- Representational state transfer* (REST) adalah salah satu rancang bangun arsitektur sistem *web service*. Rest dapat dibangun dengan sedikit *tools*, dengan melakukan testing terhadap REST *service* dapat dilakukan secara sederhana pada *web browser* tanpa harus melakukan simulasi *client-server*.
- Sistem REST *Server* menyediakan *resources* (sumber data) dan sistem REST *Client* mengakses dan menampilkan *resource*, direpresentasikan dalam bentuk format XML/JSON.

1.3. Tujuan Penelitian

Berdasarkan pada rumusan masalah, maka tujuan dari penelitian ini yaitu implementasi *representational state transfer* aplikasi layanan informasi lowongan pekerjaan.

- Membantu perusahaan (penyedia kerja) untuk memberikan informasi lowongan pekerjaan di perusahaannya dan mengetahui informasi data calon pelamar yang akan bekerja di perusahaan tersebut.
- Membantu *client* dalam memperoleh informasi sistem lowongan kerja.

- c. Membangun perangkat lunak dalam hal ini REST guna mengintegrasikan aplikasi yang berbeda dengan memanfaatkan *function-function* PHP.
- d. Merancang sistem REST pada aplikasi layanan informasi lowongan pekerjaan.

1.4. Manfaat Penelitian

Adapun Manfaat yang didapat dari hasil penelitian Tugas Akhir ini adalah sebagai berikut :

- a. Memberikan pemahaman konsep implementasi *representational state transfer* aplikasi layanan informasi lowongan pekerjaan.
- b. Memberikan kemudahan dalam penelusuran lowongan kerja, web yang dibangun menggabungkan beberapa *platform* yang memiliki alamat situs tersendiri.
- c. Memberikan kemudahan kepada pelamar kerja dan penyedia lowongan kerja dalam mencapai tujuannya masing-masing.

2. LANDASAN TEORI

2.1. Lowongan Kerja

Beberapa perusahaan besar maupun kecil di Indonesia menyediakan informasi lowongan kerja melalui media cetak seperti koran, ada juga yang mengumumkan lowongan pekerjaan melalui papan pengumuman serta melalui media elektronik termasuk internet. Lowongan kerja yang paling banyak dicari oleh pencari kerja adalah lowongan kerja terbaru dan belum lewat waktu terakhir pendaftaran. Hal ini akan menjadi kesulitan bagi pencari kerja jika mekanisme pelaksanaan pengelolaan informasi lowongan kerja dari perusahaan terkait tidak berjalan dengan baik [3].

2.2. Web Service

Web Service merupakan suatu komponen software yang merupakan *self-containing*, aplikasi modular *self-describing* yang dapat dipublikasikan, dialokasikan, dan dilaksanakan pada *web* [18]. *Web Service* adalah teknologi yang mengubah kemampuan internet dengan menambahkan kemampuan *transactional web*, yaitu

kemampuan *web* untuk saling berkomunikasi dengan pola *program to program* (P2P). Fokus *web* selama ini didominasi oleh komunikasi *program to user* dengan interaksi *business to customer* (B2C), sedangkan *stransactional web* akan didominasi oleh P2P dengan interaksi *business to business* [6].

Web service dapat didefinisikan sebagai sekumpulan fungsionalitas yang dapat diakses melalui *Internet Protocol* (IP) standar. *Web service* banyak digunakan untuk aplikasi sistem yang terdistribusi secara heterogen baik dari sistem operasi sampai model objek. *Web service* merupakan entitas yang dapat diprogram dan menyediakan beberapa macam fungsi seperti informasi, aplikasi logik dan diakses banyak *platform* melalui beberapa media yaitu *Hypertext Transfer Protocol* (HTTP) dan XML.

Kekuatan dari *web service* adalah menggunakan model *messaging* berbasis teks untuk komunikasi yang memungkinkan dapat beroperasi secara efektif pada *platform* yang berbeda. Oleh karena itu, *web service* banyak digunakan dan diakui secara luas. Pada dasarnya teknik dari *web service* adalah membuat sebuah pelayanan (*service*) terhadap segala masukkan dari *client*. Dalam hal ini *web service* dikembangkan dalam area ruang kerja berbasis web.

Konsep teknologi *web service* muncul untuk mendukung sistem terdistribusi yang berjalan pada infrastruktur yang berbeda. Dengan adanya kombinasi dari *eXtensible Markup Language* (XML) dan *Hypertext Transport Protocol* (HTTP), *web service* yang berbasis XML sangat mungkin untuk diimplementasikan. Bahkan teknologi *web service* ini mampu menggantikan fungsi dari beberapa teknologi yang serupa yang telah berkembang yaitu *Common Object Request Broker Architecture* (CORBA), *Dynamic Component Object Model* (DCOM), dan *Java Remote Method Invocation* (RMI).

2.3. Javascript Object Notation (JSON)

Javascript Object Notation (JSON) adalah sebuah general-purpose data encoding format yang populer. Penerapan JSON telah banyak digunakan pada database dan *web service*. Struktur dokumen JSON secara opsional

dapat dibatasi berdasarkan skema yang terdiri atas dua hal yakni map (pemetaan struktur nilai berdasarkan klasifikasi jenisnya) dan list (pengelompokan nilai berdasarkan klasifikasi jenisnya) [12].

2.4. eXtensible Markup Language (XML)

eXtensible Markup Language (XML) merupakan dasar terbentuknya *web service* yang digunakan untuk mendeskripsikan data. Pada level paling detail *web service* secara keseluruhan dibentuk di atas XML. Fungsi utama dari *eXtensible Markup Language* (XML) adalah komunikasi antar aplikasi, integrasi data dan komunikasi aplikasi eksternal dengan partner luaran. Dengan standarisasi XML, aplikasi-aplikasi yang berbeda dapat mudah berkomunikasi antar satu dengan yang lain [18].

Saat ini XML dapat dikatakan telah menggantikan HTML. Masing-masing dikembangkan untuk tujuan yang berbeda. Kalau HTML digunakan untuk menampilkan informasi dan berfokus pada bagaimana informasi terlihat, XML mendeskripsikan susunan informasi dan berfokus pada informasi itu sendiri. XML terutama dibutuhkan untuk menyusun dan menyajikan informasi dengan format yang tidak mengandung format standar layaknya *heading*, *paragraph*, *table* dan lain sebagainya.

2.5. Representational State Transfer (REST)

REST adalah salah satu jenis *web service* yang menerapkan konsep perpindahan antar *state*. *State* disini dapat digambarkan seperti jika *browser* meminta suatu halaman *web*, maka *server* akan mengirimkan *state* halaman *web* yang sekarang ke *browser* [17]. Bernavigasi melalui *link-link* yang disediakan sama halnya dengan mengganti *state* dari halaman *web*. Begitu pula REST bekerja, dengan bernavigasi melalui *link-link* HTTP untuk melakukan aktivitas tertentu, seakan-akan terjadi perpindahan *state* satu sama lain. Perintah HTTP yang bisa digunakan adalah fungsi GET, POST, PUT atau DELETE. Balasan yang dikirimkan adalah dalam bentuk XML sederhana tanpa

ada protokol pemaketan data, sehingga informasi yang diterima lebih mudah dibaca dan diparsing disisi *client*.

REST merupakan salah satu teknologi *web service* yang terbilang cukup populer di masa sekarang ini. Teknologi ini bekerja berdasarkan *resource* untuk membuat sistem terdistribusi. REST adalah perangkat lunak yang didesain dengan penekanan pada kesederhanaan, skalabilitas, serta kegunaan.

2.6. Keunggulan REST

Adapun beberapa keunggulan yang dimiliki oleh REST adalah sebagai berikut [4]:

- a. REST menyediakan infrastruktur yang bagus dalam proses *caching* melalui metode HTTP GET. Hal ini dapat meningkatkan performa jika informasi tidak diubah dan tidak dinamis.
- b. REST memisahkan perspektif *server* dan *client* melalui interaksi yang menggunakan HTTP.
- c. REST dapat mengembalikan *response* dalam format yang beragam dan sesuai dengan permintaan *client*.
- d. REST dapat dikembangkan menggunakan bahasa pemrograman manapun selama bahasa tersebut dapat membuat permintaan berbasis web melalui HTTP.
- e. REST cocok digunakan pada aplikasi perangkat bergerak.

3. METODOLOGI PENELITIAN

3.1. Metodologi Penelitian

Metode yang digunakan untuk penelitian adalah sebagai berikut :

- a. Studi Literatur, Peneliti memanfaatkan referensi yang ada tentang penelitian sebelumnya dengan pengembangan sistem *web service* dan gaya arsitektur REST. Hal ini dilakukan untuk menghindari kesalahan yang mungkin dilakukan dalam penelitian sebelumnya.
- b. Pembuatan sistem representational state transfer aplikasi layanan informasi lowongan pekerjaan, yang meliputi tahapan sebagai berikut :

1. Perancangan perangkat lunak berupa backend server dan frontend client yang akan menjadi interface untuk menampilkan hasil dari sistem layanan informasi lowongan pekerjaan dengan menggunakan REST services.
2. Implementasi dan uji coba.
3. Analisis sistem dengan melakukan uji coba.

3.2. Pengembangan Sistem

Pada tahap analisis dan perancangan adalah tahap yang menjelaskan bagaimana sistem dapat memenuhi kebutuhan informasi. Untuk dapat memenuhi kebutuhan pengguna, sistem memerlukan beberapa tahap desain seperti desain input, desain output, desain basis data, desain proses dan desain interface, selain pada desain sistem akan diberikan gambaran secara detail tentang DAD dan ERD pada sistem. Berikut rincian tentang desain input, desain output, desain basis data, desain proses dan desain interface yang akan dibuat adalah sebagai berikut :

- a. Desain Input
Desain Input berfungsi untuk memasukkan data dan memproses ke dalam format yang sesuai.
- b. Desain Proses
Desain proses merupakan tahap untuk membuat sketsa yang akan terjadi pada setiap modul yang dimiliki sistem. Sketsa tersebut dijadikan acuan dalam membuat website. Pada tahap ini, desain prosesnya adalah informasi seperti *user ID* dan sejenisnya yang akan di *request* dari *server* pemilik aplikasi ke server anggota yang terdaftar sebagai pencari kerja maupun perusahaan atau organisasi yang memiliki lowongan pekerjaan. Selanjutnya server akan merespon dan mengirimkan data sesuai dengan yang diminta.
- c. Desain Output
Desain output merupakan format informasi yang diperlukan, serta menentukan unsur yang dibutuhkan untuk membuat informasi. Data output yang diharapkan dari sistem meliputi data teks.

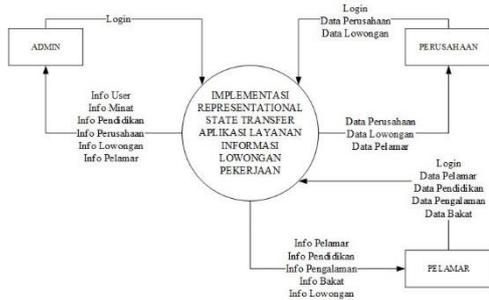
- d. Desain Basis Data
Desain basis data adalah Pengembangan basis data yang akan dilakukan pada sistem dengan menggunakan Database MySQL.
- e. Desain Antar Muka
Desain perancangan antar muka dilakukan sederhana mungkin tetapi tidak menghilangkan unsur-unsur penting dalam menyampaikan informasi, desain akan dibuat sederhana tetapi tidak menghilangkan kebutuhan dari sistem, hal ini dimaksudkan agar pengguna dapat mudah memahami pengoperasian sistem tersebut.

4. HASIL DAN PEMBAHASAN

4.1. Perancangan Sistem

Dalam proses perancangan sistem memiliki tujuan membuat rancangan sistem baru yang dapat menyelesaikan masalah-masalah dengan cepat dan tepat, Dalam rancangan suatu sistem dapat menggunakan *Entity Relationship Diagram* (ERD) dan *Data Flow Diagram* (DFD) yang bertujuan untuk mendesain sistem yang akan dihasilkan. pada tahap perancangan ini memberikan gambaran tentang sistem yang akan dibuat, yaitu mengenai alur data, proses dari sistem mulai dari masukan serta keluaran yang akan dibuat.

- a. Context Diagram
Context Diagram ini akan menerangkan tentang keterlibatan entitas dalam sistem. Entitas yang terlibat pada sistem di antaranya admin, perusahaan dan pelamar. Setiap entitas memiliki peran dan fungsi yang berbeda-beda sesuai dengan kapasitas dan hak aksesnya ke dalam sistem aplikasi layanan informasi lowongan pekerjaan. Adapun rancangan *Context Diagram* sistem ini dapat dilihat pada Gambar 1.

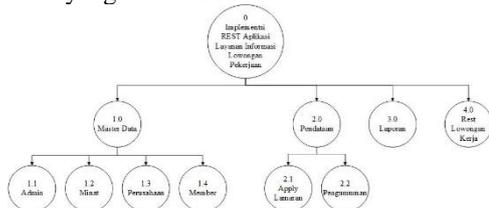


Gambar 1 Context Diagram

Pada *Context Diagram* Gambar 1 menunjukkan proses aliran data yang terjadi di dalam sistem terdapat beberapa entitas yang terlibat. Pada masing-masing entitas memiliki hak akses atas sistem yang berbeda-beda.

b. Hierarchy Diagram

Hierarchy Diagram bertujuan mempermudah penggambaran *Data Flow Diagram* (DFD) ke level yang lebih bawah lagi. Untuk memudahkan dalam merancang suatu sistem yang menggambarkan semua proses yang ada dalam sebuah sistem. Penulis merancang *Hierarchy Diagram* setelah melakukan pengidentifikasian masalah yang dianalisis terlebih dahulu.



Gambar 2 Hierarchy Diagram

Dari gambar di atas dapat dilihat fitur-fitur yang ada dalam sistem aplikasi layanan informasi lowongan pekerjaan terdapat beberapa proses di antaranya, *Master Data*, *Pendaftaran*, *laporan*, *Rest Lowongan Kerja*. Pada menu *master data* terdapat dua submenu yaitu data admin, data minat, data perusahaan dan data member. menu *pendaftaran* terdapat dua submenu yaitu data *apply* lamaran dan pengumuman. Pada menu *laporan* menampilkan berkas pelamar yang nantinya digunakan untuk perusahaan dalam penerimaan karyawan baru. Sedangkan *Rest Lowongan Kerja* berfungsi sebagai penghubung antar aplikasi sistem itu sendiri.

c. Perancangan ERD

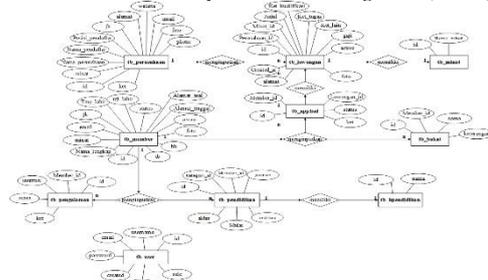
Salah satu pemodelan yang sering digunakan untuk merancang basis data adalah *Entity Relationship Diagram* (ERD). Dua elemen fundamental dalam ER adalah entitas dan relationship.

1. Identifikasi Entitas

Entitas pada sistem aplikasi layanan informasi lowongan pekerjaan adalah sebagai berikut:

- a) tb_user
- b) tb_bakat
- c) tb_lowongan
- d) tb_member
- e) tb_minat
- f) tb_pendidikan
- g) tb_pengalaman
- h) tb_perusahaan
- i) tb_kpendidikan
- j) tb_applied

2. Entity Relation Diagram (ERD)



Gambar 3 Entity Relation Diagram (ERD)

3. Relasi Tabel

Relasi Antar Tabel yang mempresentasikan hubungan antar objek di dunia nyata. Relasi merupakan hubungan yang terjadi pada suatu tabel dengan lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur operasi suatu *database*. Relasi antar tabel di bawah ini merupakan skema relasi pada *desainer database* yang digunakan dimana satu tabel dengan tabel lainnya berelasi antara *primary key* dan *foreign key*. *Tiap file database* yang tersusun, masing-masing dihubungkan atau direlasikan berdasarkan kunci *field* penghubung pada masing-masing *database*. Relasi masing-masing

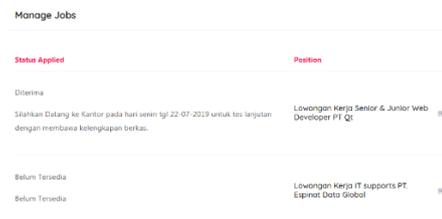


Gambar 8 Tampilan Halaman Detail Lowongan

Pada Gambar 8 dapat dilihat Pada halaman detail lowongan terdapat informasi tentang kualifikasi perusahaan, keterangan tugas, keterangan lainnya, kategori minat, gaji dan apply job untuk menambah lamaran kerja di perusahaan.

5. Halaman Manage Apply job

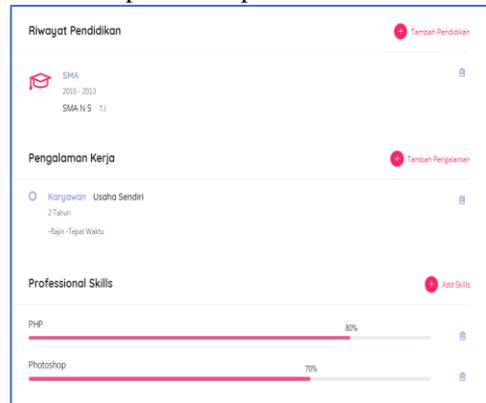
Halaman Manage Apply job untuk menampilkan informasi perusahaan yang kita lamar. Halaman Manage Apply job dapat dilihat pada Gambar 9.



Gambar 9 Tampilan Halaman Manage Apply Job

6. Halaman Menu Resume

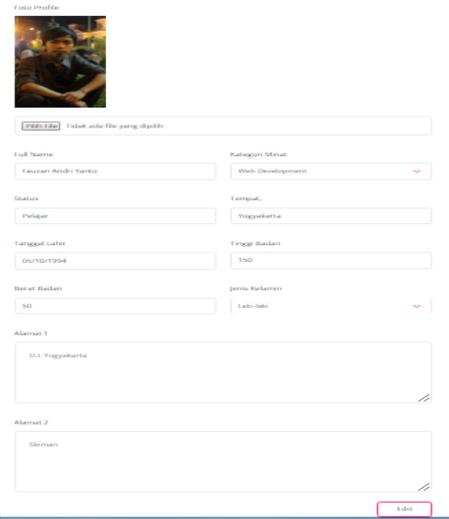
Halaman sub menu resume untuk menampilkan informasi data menambah data riwayat pendidikan, pengalaman kerja dan profesional skills. Halaman Sub Menu Resume dapat dilihat pada Gambar 4.10.



Gambar 10 Tampilan Halaman Sub Menu Resume

7. Halaman My Profil Member

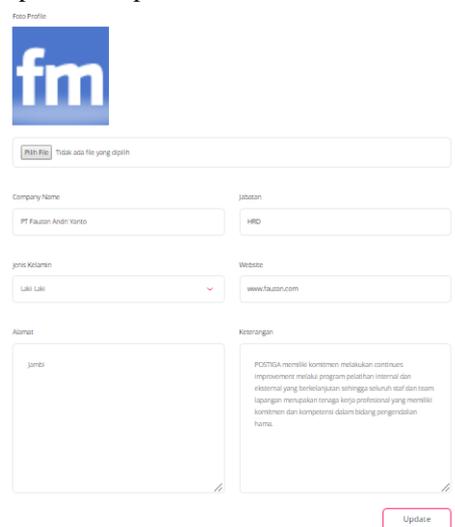
Halaman My Profil untuk menampilkan informasi data member dan update data profil member meliputi foto profil, fullname, kategori minat, status, tempat lahir, tanggal lahir, tinggi badan, berat badan, jenis kelamin, alamat 1, alamat 2. Halaman Menu Profil dapat dilihat pada Gambar 11.



Gambar 11 Tampilan Halaman My Profil Member

8. Halaman Profil Perusahaan

Halaman Profil untuk menampilkan informasi data perusahaan dan update profil perusahaan seperti foto profil, Company name, Jabatan, Jenis kelamin, website, alamat, keterangan . Halaman Menu Profil dapat dilihat pada Gambar 12.



Gambar 12 Tampilan Halaman Profil Perusahaan

9. Halaman Post New Job

Halaman Post New Job untuk menambahkan lowongan kerja pada form judul job, ket. kualifikasi, ket. tugas, ket. berkas, gaji, kategori minat dan masa aktif lowongan. Halaman Post New Job dapat dilihat pada Gambar 4.13.

Gambar 13 Tampilan Halaman Post New Job

10. Halaman Lowongan

Halaman Lowongan untuk menampilkan informasi data lowongan perusahaan berupa tabel seperti title, masa aktif, status, dan action. Pada kolom action terdapat icon mata yang memiliki fungsi melihat daftar member yang melamar kerja dan memberikan notifikasi kepada member diterima/ditolak. Halaman Menu Profil dapat dilihat pada Gambar 14.



Gambar 14 Tampilan Halaman Lowongan

11. Halaman Notifikasi/Konfirmasi

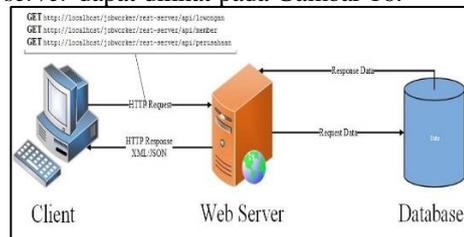
Halaman konfirmasi untuk pemberitahuan ke member atas lowongan yang telah di lamar untuk di tindak lanjuti oleh member jika lamarannya di setuju oleh perusahaan. Perusahaan mengisi form yang telah disediakan seperti form status

(diterima/ditolak) dan menambah keterangan tertulis. Halaman notifikasi dapat dilihat pada Gambar 15.

Gambar 15 Tampilan Halaman Notifikasi/Konfirmasi

4.3. Implementasi Rest Web Server dan Client

Rest diimplementasikan pada *web server* sebagai antar muka penghubung *client*. Aplikasi *client* berkomunikasi dengan *web server* melalui protokol HTTP. Dimana *client* mengirimkan *request* ke *server* melalui HTTP untuk melakukan aktivitas tertentu, perintah HTTP yang bisa digunakan adalah fungsi GET, POST, PUT, atau DELETE. mengembalikan data *response* melalui HTTP *response* ke pengguna dalam format XML/JSON sederhana disisi *client*. Untuk gambaran lebih jelas tentang proses komunikasi dan pertukaran data antara *client-server* dapat dilihat pada Gambar 16.

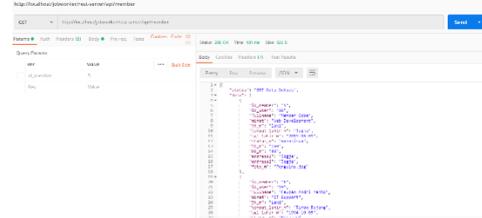


Gambar 16 Skema proses komunikasi dan pertukaran data

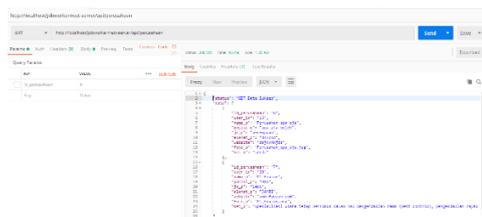
4.4. Implementasi Aplikasi Rest Server

Pada aplikasi *Rest Server* yang menyediakan layanan sumber data diimplementasikan menggunakan bahasa pemrograman PHP dengan *Apache web server* serta database MySQL. Untuk protokol komunikasinya sendiri antara *client* dan *server* menggunakan protokol http. Aplikasi yang digunakan pada

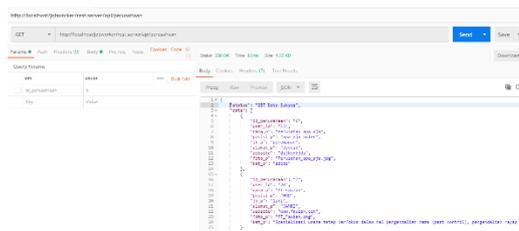
server adalah REST dalam format JSON. Aplikasi *server* ini tidak dibuatkan *interface* untuk menangani *database*, karena sebagai fungsi utamanya yaitu menyediakan layanan sumber data *Client*.



Gambar 17 Hasil Pengujian Data Member dalam format JSON.



Gambar 18 Hasil Pengujian Data Perusahaan dalam format JSON



Gambar 19 Hasil Pengujian Data Lowongan dalam format JSON

5. PENUTUP

5.1. Kesimpulan

Berdasarkan penelitian yang dilakukan oleh penulis, didapatkan kesimpulan sebagai berikut :

- Aplikasi *web server* dapat di akses oleh *client* dalam mendapatkan informasi seperti, data member, data perusahaan, dan data lowongan.
- Penerapan REST *service* dengan format JSON sebagai *backend* pada sistem ini sangat cocok digunakan karena dengan format dokumen JSON yang lebih terstruktur di bandingkan dengan format dokumen XML.
- REST *server* yang dirancang dan dibuat dapat di uji dengan baik menggunakan *tools* pengujian *postman*.

- Pengujian sistem dalam penelitian ini memperoleh keberhasilan pada lingkungan uji dengan *backend server* satu atau lebih pada metode REST dengan fungsi GET, POST, PUT, dan DELETE.

5.2. Saran

Berdasarkan batasan-batasan penelitian yang dilakukan oleh penulis, maka untuk kepentingan penelitian yang lebih lanjut penulis memberikan saran sebagai berikut :

- Supaya mampu dimanfaatkan oleh awam, *backend* dari sistem informasi berbasis REST harus memiliki *frontend* sebagai antarmukanya. Untuk itu dibutuhkan pengembangan lebih lanjut terutama untuk antarmuka yang menjadi media interaksi antara pengguna dengan *backend server*.
- Penambahan fitur keamanan pada REST *server* seperti token akses dan dokumentasi untuk menggunakan sumber data yang dibutuhkan *client*.

DAFTAR PUSTAKA

- Amri, A.A. dan Astuti, E.S., (2018) *Evaluasi Implementasi Sistem Informasi Lowongan Kerja Berbasis Web*. Fakultas Ilmu Administrasi, Universitas Brawijaya. Malang.
- Connolly, T.M., dan Begg, C.E., (2005) *Database System A Practical Approach to Design, Implementation and Management* Fourt Edition : Addison Wesley
- Darmastuti, D., (2013) *Implementasi Metode Simple Additive Weighting (Saw) Dalam Sistem Informasi Lowongan Kerja Berbasis Web Untuk Rekomendasi Pencari Kerja Terbaik*, Program Studi Teknik Informatika, Jurusan Teknik Elektro Fakultas Teknik Universitas Tanjung pura.
- Dhingra, S., (2013) *REST vs. SOAP: Choosing the best web service*.
- Fathansyah, (2015) *Basis Data*. Bandung : INFORMATIKA
- Gottschalk, K, (2002) *Introduction to Web Services Architecture*. IBM System Journal, Vol 41, No 2.
- Habibie, F.H., (2014) *Pembangunan sistem informasi penerimaan calon tenaga kerja secara online berbasis web pada bursa kerja khusus smk ganesh*

- tama boyolali. Program Studi Teknik Informatika, Universitas Surakarta.
- [8] Hutahaean, J., (2014) *Konsep Sistem Informasi*. Yogyakarta: Deepublish
 - [9] Kadir, A., (2008) *Dasar Pemrograman Web Dinamis Menggunakan PHP*. Yogyakarta: Andi
 - [10] Kadir, A., (2011) *From Zero To A Pro : CSS - Tip Dan Trik Untuk Menyertakan Cascading Style Sheet Dalam Halaman Web*. Yogyakarta: Andi
 - [11] Kadir, A., (2013) *From Zero to A Pro: JavaScript & jQuery Rahasia untuk Menguasai Pembuatan Aplikasi Web yang Interaktif dan Dinamis*. Yogyakarta: Andi
 - [12] Martin, K. dan Beresford, A.R., (2017) *A Conflict-Free Replicated JSON Datatype*. University of Cambridge Computer Laboratory, Cambridge.
 - [13] Nurwaldi, T., (2014) *Simulasi Web Service Lowongan Kerja Dan Perekrutan Karyawan Se-Kota Solo*. Fakultas Komunikasi dan Informatika, Universitas Muhammadiyah Surakarta.
 - [14] Paramartha, A.A.G.Y., Suryaningsih, G.K., dan Aryanto, K.Y.E., (2016) *Implementasi Web Service Pada Sistem Pengindeksan Dan Pencarian Dokumen Tugas Akhir, Skripsi, Dan Praktik Kerja Lapangan*. Jurusan Management Informatika, Fakultas Teknik & Kejuruan Universitas Pendidikan Ganesha, Singaraja, Indonesia.
 - [15] Rachmanto, R., (2017) *9 Langkah Praktis Membuat Website Gratis*. Jakarta: PT. Elex Media Komputindo.
 - [16] Rosa, A.S. dan Shalahuddin, M. (2013) *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Informatika. Bandung.
 - [17] Tidwell, D., (2001) *Web Services: The Web's next Revolution*.
 - [18] Wahli, U. Burroughs, O. Cline, O. dan Tung, L., (2006) *Service Handbook for WebSphere Application Server 6.1*. IBM Redbooks
 - [19] Yakub. (2012) *Konsep Dasar Sistem Informasi*. Yogyakarta. Andi
 - [20] Yazdi, M. (2012) *Implementasi Web-Service Pada Sistem Pelayanan Perijinan Terpadu Satu Atap Di Pemerintah Kota Palu*. Fakultas MIPA, Universitas Tadulako. Palu-Sulawesi Tengah.