

Rancang Bangun *Web Service* Untuk Aplikasi Protika Berbasis Android Dengan Metode Rest

Kiran Apriyana, Safaruddin Hidayat Al Ikhsan, Fitrah Satrya Fajar K
Program Studi Teknik Informatika, Fakultas Teknik, Universitas Ibn Khaldun Bogor
apriyanakiran@gmail.com

Abstrak

Web Service adalah suatu perangkat lunak yang dirancang untuk menyediakan komunikasi dan interaksi pada dua aplikasi atau lebih dengan sifat *program-to-program*. Dengan adanya *web service*, maka satu aplikasi dapat diakses dan berhubungan tanpa perlu mengetahui detail program secara mendalam dengan aplikasi lain walaupun aplikasi tersebut berbeda platform, bahasa pemrograman, dan sistem informasi. Penggunaan teknologi *REST web service* dapat meningkatkan kinerja dengan mengurangi format pada paket, sehingga mengurangi juga penggunaan memori dan komputasi pada device. Dengan layanan *REST web service*, memudahkan dalam fungsi request yang memanfaatkan query request provider melalui *HTTP* dengan menggunakan *Uniform Resource Identifier*. Pada penelitian ini dilakukan pembuatan *web service* menggunakan *Slim Framework* yang diterapkan pada Aplikasi Profil Program Studi Teknik Informatika (*PROTIKA*) berbasis android. Prinsip Kerja *Slim Framework* yaitu hanya menerima *HTTP request*, memanggil fungsi yang diinginkan, dan mengembalikan *HTTP response*. Aplikasi Profil Program Studi Teknik Informatika (*PROTIKA*) adalah aplikasi berbasis android yang bertujuan untuk memberikan informasi seputar civitas Teknik Informatika. Aplikasi yang dihasilkan dapat menyediakan layanan *web service* dalam format *JSON* yang memberikan informasi atau data sesuai dengan kebutuhan aplikasi. Sehingga komunikasi data dapat berjalan dengan baik dan terintegrasi dengan sistem pada Aplikasi Profil Program Studi Teknik Informatika.

Kata kunci: *Web Service, REST, Slim, Android, PROTIKA*

PENDAHULUAN

Latar Belakang

Program Studi S1 Teknik Informatika merupakan salah satu dari 4 (empat) program studi di bawah Fakultas Teknik dan Sains Universitas Ibn Khaldun Bogor (UIKA Bogor). Program studi ini dibentuk untuk mempersiapkan tenaga akademik dan/atau profesional yang mampu mengembangkan ilmu pengetahuan dan keterampilan di bidang Teknik Informatika. Program Studi S1 Pendidikan Teknik Informatika saat ini baru memiliki empat konsentrasi/peminatan/ lab keilmuan yaitu: Rekayasa Perangkat Lunak (RPL), *Net Centric Computing* (NCC), Sistem Informasi (SI), dan Geo Informasi (GI) dan juga memiliki satu lab Riset *Artificial Intelligence* (AI). Program Studi ini ditunjang oleh fasilitas laboratorium praktikum dan laboratorium penelitian untuk menunjang proses belajar mengajar serta didukung oleh berbagai kerjasama bidang pendidikan dan industri.

Dalam penyebaran dan publikasi informasi mengenai Program Studi Teknik Informatika masih sulit didapatkan karna belum adanya suatu

wadah informasi tersebut, juga belum terintegrasi dengan *web service* dalam transaksi sistem datanya. Oleh karna itu pengembangan wadah informasi yang berkaitan dengan akademik dan civitas kemahasiswaan di Teknik Informatika ini akan dibuat sebuah aplikasi berbasis Android *mobile* bernama Profil TI UIKA. Aplikasi tersebut akan menjadi sumber informasi bagi jajarannya Prodi Teknik Informatika, maupun untuk seluruh mahasiswa-mahasiswanya dalam mencari informasi terbaru, teraktual, dan terintegrasi dengan sistem. Untuk mendukung proses transaksi data pada aplikasi Profil TI UIKA maka diperlukan teknologi *web service*.

Web Service adalah suatu software yang dirancang untuk menyediakan komunikasi dan interaksi pada dua aplikasi atau lebih dengan sifat *program-to-program* [1]. Dengan adanya *Web service*, maka satu aplikasi dapat diakses dan berhubungan tanpa perlu mengetahui detail program secara mendalam dengan aplikasi lain walaupun aplikasi tersebut berbeda platform, bahasa pemrograman, dan sistem informasi [2]. *Web service* memiliki dua metode yang

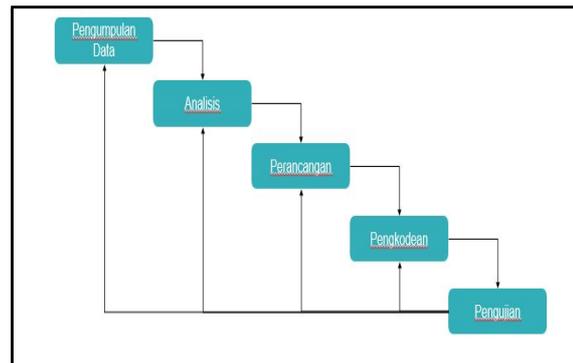
berorientasi pada sumber daya informasi dan layanan informasi yaitu Simple Object Access Protocol (SOAP) dan Representational State Transfer (REST). Sudah banyak perusahaan yang menggunakan metode SOAP, tetapi saat ini metode REST adalah yang paling dominan digunakan untuk pengembangan *web service* dikarenakan lebih efektif dalam pengembangan dan penggunaan. Format response yang dimiliki SOAP hanya berupa XML, berbeda dengan REST yang dapat memberikan response berupa XML, JavaScript Object Notation (JSON) ataupun format text lainnya [3].

REST adalah sebuah metode dalam menyampaikan *resource* melalui media web. Sedangkan *resource* sendiri didefinisikan sebagai segala sesuatu yang dapat disimpan didalam sebuah komputer dan ditampilkan sebagai urutan bit, misalnya sebuah dokumen, tabel dalam sistem basis data, atau hasil dari sebuah perhitungan [4]. Penggunaan teknologi REST *web service* dapat meningkatkan kinerja dengan mengurangi format pada paket, sehingga mengurangi juga penggunaan memori dan komputasi pada *device*. Dengan layanan REST *web service*, memudahkan dalam fungsi request yang memanfaatkan *query request provider* melalui HTTP dengan menggunakan *Uniform Resource Identifier* [5]. Dalam penggunaannya, REST API terbukti lebih cepat dalam transfer data daripada metode lain yang serupa, dalam hal ini SOAP (*Symple Object Access Protocol*) [6]. Tidak seperti SOAP, tidak ada notasi standar resmi untuk RESTful API. Hal ini dikarenakan REST merupakan arsitektur, sedangkan SOAP adalah protokol.

Berdasarkan uraian latar belakang diatas, maka pada penelitian ini akan dibangun suatu “Rancang Bangun *Web service* Untuk Aplikasi PROTIKA Berbasis Android Dengan Metode REST” yang diharapkan dapat mengoptimalkan proses transaksi data pada aplikasi Profil TI UIKA lebih cepat dan aman.

METODE PENELITIAN

Metode penelitian yang digunakan mengadopsi metode *waterfall* menurut Pressman [7]. Meliputi Pengumpulan Data, Analisis, Perancangan, Pengkodean, Pengujian. Metode penelitian ditunjukkan pada Gambar 1.



Gambar 1. Model Waterfall Menurut Pressman (2015:42)

a. Pengumpulan Data

Pengumpulan data pada penelitian ini menggunakan beberapa metode yang digunakan yaitu sebagai berikut:

1. Observasi

Metode observasi merupakan metode yang dilakukan melalui pengamatan secara langsung pada suatu kegiatan yang sedang dilakukan. Metode observasi pada penelitian ini dilakukan dengan cara turun ke lapangan dan melakukan riset untuk mendapatkan data-data yang berhubungan dengan civitas Prodi Teknik Informatika sehingga dapat digunakan untuk penyempurnaan database Prodi TI pada Aplikasi PROTIKA.

2. Studi Literatur

Studi literature merupakan kegiatan yang berkenaan dengan metode pengumpulan data pustaka, pemahaman terhadap teori-teori sumber literature, seperti jurnal dan buku terkait dengan penelitian yang akan dilakukan. Teori yang berhubungan dengan *web service* dan REST *web service* digunakan sebagai bahan penelitian ini.

b. Analisis

Pada tahap ini dilakukan proses analisis kebutuhan sistem. Analisis yang dilakukan ialah mengetahui analisis kebutuhan fungsional, analisis kebutuhan non-fungsional, dan analisis arsitektur sistem dari sistem PROTIKA yang akan berjalan nantinya. Analisis kebutuhan fungsional yang diperlukan meliputi apa saja fitur-fitur dari layanan *web service* yang harus disediakan pada Aplikasi PROTIKA.

c. Perancangan

Pada tahap ini dilakukan proses perancangan *database* yang dibutuhkan sistem PROTIKA, *database* ini yang akan digunakan sebagai wadah

data dan informasi untuk kebutuhan aplikasi PROTIKA. *Database* digambarkan melalui perancangan *database schema* yang ditunjukkan pada Gambar 2.

d. Pengkodean

Pada tahap ini dilakukan proses implementasi ke dalam kode program menggunakan bahasa pemrograman *PHP* menggunakan *Slim Framework*. Proses ini merupakan implementasi desain ke dalam bahasa yang dikenali oleh komputer. Lalu setelah pengekodean selesai, maka dilakukan proses *testing* terhadap sistem yang telah dibuat.

e. Pengujian

Pada tahap ini dilakukan proses pengujian sistem dengan menggunakan Metode *Black Box* untuk pengujian fungsi dari setiap *web service* yang dibuat.

HASIL DAN PEMBAHASAN

Pengimplementasian *web service* sudah banyak dikembangkan dimana-mana oleh para *developer* aplikasi. Dengan hadirnya teknologi *web service* ini pertukaran informasi atau data dapat dilakukan walaupun berbeda *platform*. Pada penelitian ini *web service* yang dibangun berperan untuk melakukan transaksi data pada aplikasi PROTIKA berbasis *android*, yang nantinya dapat membuat aplikasi pada *smartphone* menjadi ringan karena data yang dibutuhkan terdapat pada *server*. Dalam prosesnya, aplikasi hanya melakukan *request* sesuai dengan kebutuhan yang kemudian akan di *response* oleh *web service*.

Slim Framework adalah salah satu mikro framework *PHP* untuk membuat aplikasi web. *Slim* biasanya digunakan untuk membuat aplikasi-aplikasi kecil dan lebih banyak dipakai untuk membuat RESTful API atau *web service*.

Prinsip kerja *Slim Framework* yaitu menerima *HTTP Request*, lalu menjalankan sebuah fungsi dan mengembalikan sebuah *HTTP Response*. Yang biasanya *response* berbentuk *JSON* karna dapat dikonsumsi oleh aplikasi lainnya.

Berikut ini adalah hasil dan pembahasan dari setiap tahapan yang telah dilakukan selama penelitian ini berlangsung.

a. Data

Data yang digunakan dalam penelitian ini

adalah informasi berupa tabel yang berisi *atribut* dan tipe data dari *database* PROTIKA. *Database* PROTIKA dibentuk dari hasil observasi langsung ke lapangan. Berinteraksi dengan pengguna sistem, mengetahui data-data apa saja yg dibutuhkan untuk penyajian informasi pada Aplikasi PROTIKA berbasis *mobile*.

b. Analisis Kebutuhan Fungsional

Kebutuhan fungsional pada sistem merupakan layanan dalam *web service* yang harus disediakan untuk Aplikasi PROTIKA berbasis *Android*. Berdasarkan kebutuhan Aplikasi PROTIKA, maka modul-modul yang akan disediakan oleh *web service* diantaranya adalah Modul Profil Sejarah TI, Modul Visi Misi TI, Modul Struktural Prodi TI, Modul Dosen Prodi TI, Modul Staff Prodi TI, Modul Akademik TI, Modul Kurikulum TI, Modul Lab Riset TI, Modul Jurnal Publikasi TI, Modul Kerjasama TI, Modul Fasilitas TI, Modul Kemahasiswaan TI, Modul Alumni TI, Modul Berita TI, Modul Forum Diskusi TI.

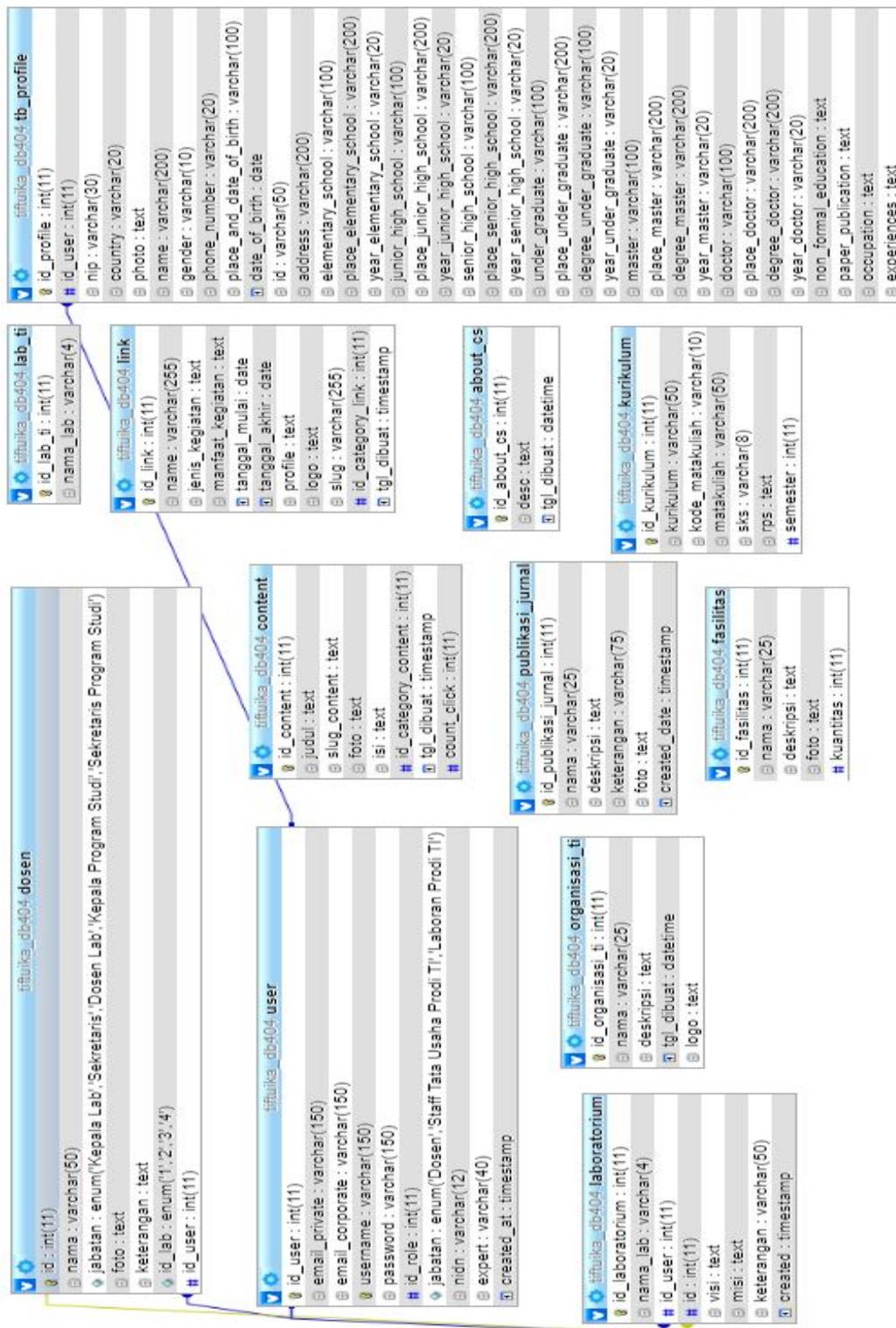
c. Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional merupakan analisis yang digunakan untuk mengetahui bagian-bagian yang berinteraksi dengan sistem yang sedang berjalan. Kebutuhan non-fungsional dari sistem ini yaitu:

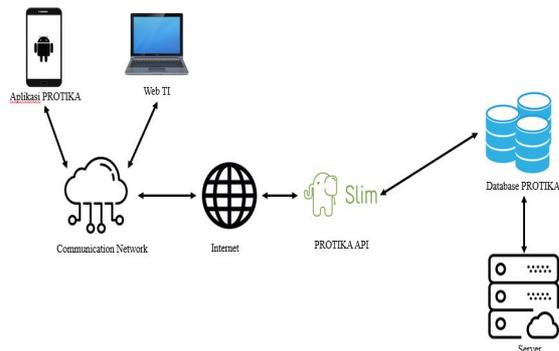
1. *Portability Web Service* dapat berjalan dalam waktu 24 jam/hari. *Web service* diakses melalui jaringan internet.
2. *Supportability Web Service* dapat digunakan melalui *platform mobile* *Android*.

d. Arsitektur Sistem

Arsitektur sistem *web service* yang akan dibangun berjalan pada *platform* *Android* yaitu Aplikasi PROTIKA berbasis *mobile*. Komunikasi yang terjadi dalam *web service* ini adalah *request* dan *response*. Pada *web service* komunikasi ini tidak dilakukan secara langsung melainkan dijembatani oleh *file web service*. Dalam penelitian ini, *format web service* yang akan digunakan adalah *JSON*. Sehingga saat Aplikasi PROTIKA akan mengakses *database* tidak akan ditangani langsung oleh *server*, melainkan melalui perantara yang disebut *web service*. Arsitektur sistem digambarkan sebagai alur kerja dari semua *device* yang terlibat sistem yang dibuat. Arsitektur Sistem ditunjukkan pada Gambar 3.



Gambar 2 Perancangan Database



Gambar 3 Arsitektur Sistem PROTIKA

e. Coding

Pada tahap *coding* ini dilakukan implementasi kode program menggunakan bahasa pemrograman PHP, modul-modul yang berisi dari *request link* dibuat, lalu di *testing* pada POSTMAN untuk menguji apakah berhasil memanggil data atau tidak. *request-request link* ini disimpan dengan nama *routes.php* yang nantinya akan dibaca oleh *middleware.php* dari setiap kali pemanggilan *request link* yang dijalankan. *middleware.php* ini berisi perintah untuk memvalidasi *API Key* yang berada pada *table key* dalam *database*. *API Key* ini harus di *input*-kan setiap kali *testing request link*, dan *API Key* harus sesuai agar saat di *testing* berhasil menarik data yang di *request*.

Berikut ini *coding* pada modul Dosen TI, modul yang digunakan oleh pengguna untuk melihat seluruh daftar dosen yang saat ini aktif di Program Studi Teknik Informatika. nantinya *coding request link* ini akan di *testing* dengan cara seperti ini.

http://ti.ft.UIKA-bogor.ac.id/prodi_ti/api-prodi/public/jabatan/dosen/?key=202cb962ac59075b964b07152d234b70

```
$app->get("/jabatan/dosen/", function (Request $request, Response $response, $args) {

    $sql = "SELECT * FROM user
    WHERE jabatan='Dosen' ORDER BY id_user
    ASC";

    $stmt = $this->db->prepare($sql);
    $stmt->execute();
    $result = $stmt->fetchALL();
    $row = $stmt->rowCount();

    if($row > 0){
        return $response->withJson(["status" => true , "data" => $result], 200);
    }
});
```

```
}else{
    return $response->withJson(["status" => false, "data" => null], 200);
}
});
```

f. Pengujian

Setelah tahap pengkodean selesai maka dilakukan tahap pengujian dilakukan menggunakan *software* POSTMAN.

Berikut hasil dari pengujian Modul Dosen Program Studi Teknik Informatika. Modul ini digunakan oleh pengguna untuk melihat daftar dosen yang saat ini aktif di Program Studi Teknik Informatika.

Tabel 1. Hasil Pengujian 1
 Modul Dosen TI

| | |
|--------|---|
| URI | http://ti.ft.UIKA-bogor.ac.id/prodi_ti/api-prodi/public/jabatan/dosen/?key=202cb962ac59075b964b07152d234b70 |
| Method | GET |

```
var options = {
    'method': 'GET',
    'hostname': 'ti.ft.UIKA-bogor.ac.id',
    'path': '/prodi_ti/api-prodi/public/jabatan/dosen/?key=202cb962ac59075b964b07152d234b70',
    'headers': {
    }
};
var req =
http.request(options,
function (res) {
    var chunks = [];

    res.on("data", function (chunk) {
        chunks.push(chunk);
    });

    res.on("end", function (chunk) {
        var body =
Buffer.concat(chunks);

        console.log(body.toString());
    });

    res.on("error", function (error) {
        console.error(error);
    });
});
```

Success Response

```
res.on("data", function (chunk) {
    chunks.push(chunk);
});

res.on("end", function (chunk) {
    var body =
Buffer.concat(chunks);

    console.log(body.toString());
});

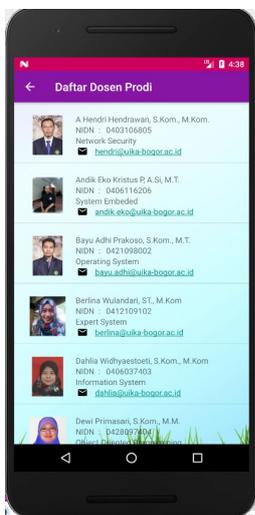
res.on("error", function (error) {
    console.error(error);
});
});
```

```
req.end();
```

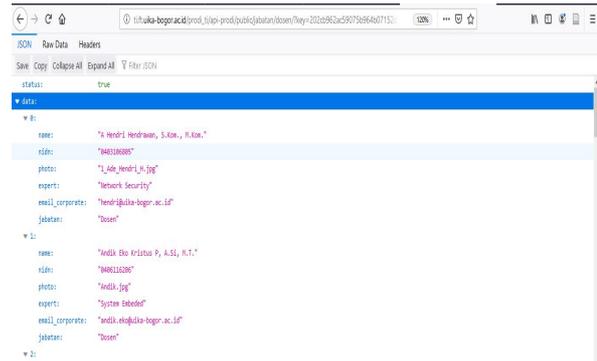
| | |
|-----------------------|--|
| Hasil Pengujian | <pre>1 { 2 "status": true, 3 "data": [4 {</pre> |
| Status Pengujian | Berhasil |
| Response Time | 177ms 107ms 199ms |
| Average Response Time | 161ms |

Berdasarkan tabel hasil pengujian 1 berikut *RESTful API* dari modul Dosen Program Studi Teknik Informatika berhasil dibuat, dengan *response time* yang berubah-ubah sesuai dengan jaringan internet saat melakukan *request* terhadap *API Protika*, jumlah *response* dari *API Protika*, maupun spesifikasi dari *device* yang melakukan request terhadap *API Protika*.

Berikut hasil *API* modul dosen yang pada Aplikasi Profil Program Studi Teknik Informatika dan Pengujian pada *web browser* yang dapat dilihat pada Gambar 4 dan Gambar 5.



Gambar 4 Hasil pada Aplikasi PROTIKA



Gambar 5 Hasil dari pengujian *Via Web Browser*

Adapun pengujian yang dilakukan dengan *API Key* berbeda dan juga pengujian dengan tidak menggunakan *API Key* pada modul yang sama saat diuji dengan menggunakan *Software POSTMAN*. Berikut hasil dari pengujianya.

Tabel 2. Hasil Pengujian 2

| | |
|------------------|---|
| Kondisi 1 | |
| URI | http://ti.ft.UIKA-bogor.ac.id/prodi_ti/api-prodi/public/jabatan/dosen/?key=202cb962ac59075b964b07152d234b71 |
| Method | GET |
| Hasil Pengujian | <pre>1 { 2 "status": "Unauthorized" 3 }</pre> |
| Status Pengujian | "Unauthorized" |
| Kondisi 2 | |
| URI | http://ti.ft.UIKA-bogor.ac.id/prodi_ti/api-prodi/public/jabatan/dosen/ |
| Method | GET |
| Hasil Pengujian | <pre>1 { 2 "status": "API Key required" 3 }</pre> |
| Status Pengujian | "API Key Required" |

Berdasarkan tabel hasil pengujian 2 berikut *RESTful API* dari modul yang sama yaitu modul Dosen Program Studi Teknik Informatika tidak

berhasil menarik data yang telah di *request* yang menampilkan 2 kondisi yang berbeda. Kondisi yang pertama dikarenakan tidak sesuainya *API Key* yang di *input*, lalu kondisi yang kedua dikarenakan tidak di ada *API Key* yang di *input*.

KESIMPULAN

Kesimpulan dari hasil penelitian yang sudah dilakukan, dengan menggunakan metode REST berhasil diterapkan pada *web service* Aplikasi PROTIKA berbasis Android menggunakan *Slim Framework* sebagai teknologi untuk mengembangkan *web service*. Sehingga komunikasi data pada Aplikasi PROTIKA berbasis *mobile* berhasil dibuat dan dapat berjalan dengan baik sesuai dengan kebutuhan dan hasil pengujian yang sudah dilakukan.

DAFTAR PUSTAKA

- [1] Gottschalk, K, Graham, S., Kreger, H., Snell, J. 2002. Introduction to Web Services Architecture. IBM Systems Journal, Vol 41 No 2, pp. 170-177
- [2] Suryadi, G., Hiryanto, L., Tumbelaka, B. 2013. Implementasi Web Service Untuk Mobile Commerce. Jurnal Ilmu Komputer dan Sistem Informasi, Vol 1 No 1, pp 73-78
- [3] Sutanta, E., & Mustofa, K.s. 2012 Identifying The Needs of Web Service to Data Synchronization Between Information Systems as E-Government Ecosystem at Bantul-Yogyakarta. *Jurnal Article*. Bandung – Teknik Informatika STMIK Bandung
- [4] Sandoval, Jose. 2009. *RESTful Java Web Services*. Packt Publishing. Birmingham.
- [5] Arsana, I Nyoman A. 2014. RANCANG BANGUN APLIKASI E-COMMERCE UNTUK BOOKING ONLINE HOTEL MENGGUNAKAN LAYANAN WEB SERVICE. *Jurnal Article*. 16-18. Yogyakarta
- [6] Dudhe, A., dan Sharekar, S. S. 2014. Performance Analysis of SOAP and RESTful Mobile Web Services in Cloud Environment. Second National Conference on Recent Trends in Information Security. Nagpur, India, Januari 2014. 1-4.
- [7] Pressman, R.S. 2015. Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku I. Yogyakarta: Andi.