

PENGEMBANGAN *APPLICATION SERVICE DATA* MAHASISWA PADA *WEBSITE* JURUSAN ILMU KOMPUTER

¹Mardhi Nopriyansah, ²Dwi Sakethi dan ³Didik Kurniawan

^{1,2,3}Jurusan Ilmu Komputer FMIPA Universitas Lampung,
Jalan Prof Sumantri Brojonegoro No. 1 Bandar Lampung, Lampung 35145
e-mail : ¹mardhi.nopriyansah@gmail.com, ²dwijim@fmipa.unila.ac.id, ³didik.kurniawan@fmipa.unila.ac.id

Abstract — *The information system at the Department of Computer Science, University of Lampung has some data such as data about students, lecturers, courses schedule and final assignments or theses. The information system contained in the Computer Science Department, University of Lampung is still not integrated between one system and another. This makes the existing data inconsistent so that there are some differences in data between systems. To solve this problem, a student data Application Service Interface (API) program with Representational Transfer State (REST) architectural style was created. Framework CodeIgniter (CI) will be used in the development of this application using the waterfall method. This research has succeeded in building thirteen APIs and tested using Postman software. System testing is also carried out using the equivalence partitioning method and system functional testing. Based on testing with the equivalence partitioning method and system functional testing, overall success was obtained at each API access point. The results of the API access will be sent in the Javascript Object Notation (JSON) data format.*

Keywords: *Application Programming Interface (API); Framework CodeIgniter (CI); Representational Transfer State (REST)*

1. PENDAHULUAN

Sistem informasi yang ada pada Jurusan Ilmu Komputer Universitas Lampung memiliki data tentang mahasiswa, data dosen, data jadwal perkuliahan dan data tugas akhir atau biasa disebut skripsi. Data awal yang ada akan bertambah seiring waktu. Data informasi tentang mahasiswa yang awalnya hanya mengenai nama, nomor pokok mahasiswa, alamat, agama, program studi dan lain-lainnya akan bertambah beberapa data baru seperti dosen pembimbing akademik, mata kuliah yang telah diambil, jadwal perkuliahan dan tugas akhir atau skripsi. Tugas akhir atau skripsi juga memiliki beberapa data tersendiri seperti judul, dosen pembimbing, dosen penguji atau dosen pembahas dan data lainnya. Terdapat dua program studi yang ada di Jurusan Ilmu Komputer Universitas Lampung yaitu program studi S1 Ilmu Komputer dan D3 Manajemen Informatika.

Sistem informasi yang terdapat pada Jurusan Ilmu Komputer Universitas Lampung masih berdiri sendiri-sendiri. Sistem informasi yang terdapat pada jurusan belum memiliki integrasi antara sistem satu dengan sistem yang lainnya. Belum adanya integrasi antar sistem membuat pertukaran data antar sistem belum bisa dilakukan. Sistem informasi yang terdapat pada Jurusan Ilmu Komputer Universitas Lampung yaitu *website* Ilmu Komputer dan kuliah *online*, *monitoring* tugas akhir atau skripsi.

Data yang terdapat di setiap sistem informasi Jurusan Ilmu Komputer Universitas Lampung belum tersinkronisasi antar sistem informasi yang ada. Belum adanya sinkronisasi ini membuat data yang ada menjadi inkonsistensi data sehingga menyebabkan terjadinya perbedaan data antara sistem. Perbedaan data tersebut membuat terjadinya kesalahan data dalam proses perkuliahan maupun proses administrasi.

2. LANDASAN TEORI

Representational State Transfer (REST) adalah arsitektur standar web yang menggunakan protokol HTTP dalam komunikasi data. Arsitektur tersebut dikembangkan berdasarkan sumber data di mana masing-masing komponen merupakan sumber data. Sumber data diakses oleh antarmuka yang sama dengan menggunakan metode standar HTTP. Dalam arsitektur REST, server yang mengikuti arsitektur REST menyediakan akses ke sumber data dan klien yang mengambil data. Setiap sumber data diidentifikasi menggunakan link URL. REST menggunakan berbagai format untuk menyajikan data, seperti teks, JSON dan XML. Berikut adalah metode HTTP yang umumnya digunakan dalam arsitektur REST: 1) *GET* untuk menyediakan akses untuk membaca sumber data, 2) *PUT* untuk memperbarui data yang tersedia, 3) *DELETE* untuk menghapus data, 4) *POST* untuk membuat data baru [1].

Java Script Object Notation (JSON) adalah format pertukaran data yang ditemukan oleh Douglas Crockford pada tahun 2006 yang memiliki ukuran data yang lebih kecil serta waktu proses yang lebih cepat dibandingkan dengan XML yang sudah terlebih dulu ada [2].

Hypertext Transfer Protocol (HTTP) merupakan sebuah protokol komunikasi antara *client* dan *server* dengan konsep *request-response*. Sebagai sebuah protokol, HTTP menentukan prosedur komunikasi baik dalam format dan cara komunikasi hingga aksi dan reaksi antara *web server* dan *browser* [3].

Hypertext Preprocessor (PHP) adalah suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browser* menjadi kode HTML. PHP adalah salah satu bahasa pemrograman yang dirancang untuk membangun aplikasi web [4].

Application Programming Interface (API) merupakan ekspresi terfokus keseluruhan fungsional dalam suatu modul perangkat lunak yang dapat diakses oleh orang yang membutuhkan dengan cara yang telah ditentukan layanan. Representasi terfokus dari fungsi yang dideklarasikan dalam API dimaksudkan untuk menyediakan rangkaian layanan yang spesifik untuk target tertentu [5].

CodeIgniter (CI) adalah aplikasi *open source* berupa *framework* dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP. *CodeIgniter* memudahkan pengembang web untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuat dari awal [6].

Postman merupakan sebuah perangkat lunak yang memuat fungsi lengkap pengembangan sistem dalam mengirimkan dan menerima respons *server*. Perangkat lunak ini mendukung pengembangan sistem REST API dengan mengklasifikasi permintaan berdasarkan *request method*, *URL* dan parameter-parameter *request* [7].

3. METODOLOGI PENELITIAN

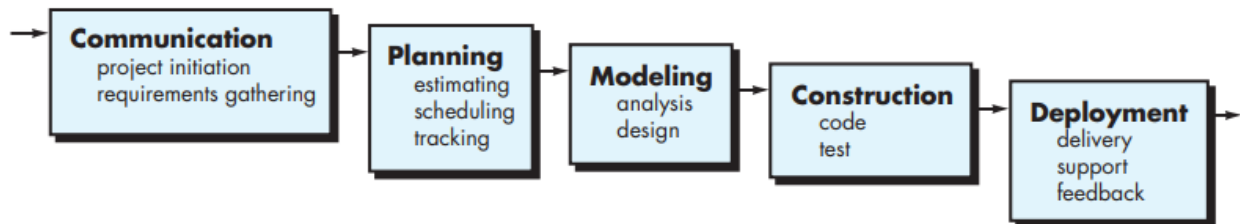
Penelitian ini memiliki tahapan yang harus dilakukan antara lain:

1. Tahapan Pengumpulan Data

Tahapan pengumpulan data dilakukan untuk mendapatkan data yang diperlukan untuk pengembangan sistem. Tahapan pengumpulan data dilakukan untuk pemahaman teori, konsep, materi, dasar ilmu dari teknologi yang digunakan dan mengetahui permasalahan yang dihadapi. Dalam tahap pengumpulan data dilakukan dengan dua metode yaitu dengan observasi dan studi pustaka. Peneliti melakukan observasi data untuk mengumpulkan data dari pihak yang terlibat secara langsung di dalam proses pembuatan sistem. Peneliti melakukan studi pustaka data untuk mengumpulkan berbagai literatur seperti pada buku, jurnal, ataupun dokumen yang berkaitan dengan penelitian ini.

2. Tahapan Pengembangan Sistem

Tahapan pengembangan sistem menggunakan metode *waterfall* karena pada setiap tahapan mampu menerima perubahan dan pengembangan konsep dari tahapan sebelumnya serta memiliki alur yang terstruktur dan sesuai dengan pengembangan sistem REST API. Gambar metode *waterfall* dapat dilihat pada Gambar 1.



Gambar 1. Metode *Waterfall*

Tahapan-tahapan tersebut adalah sebagai berikut:

a. *Communication*

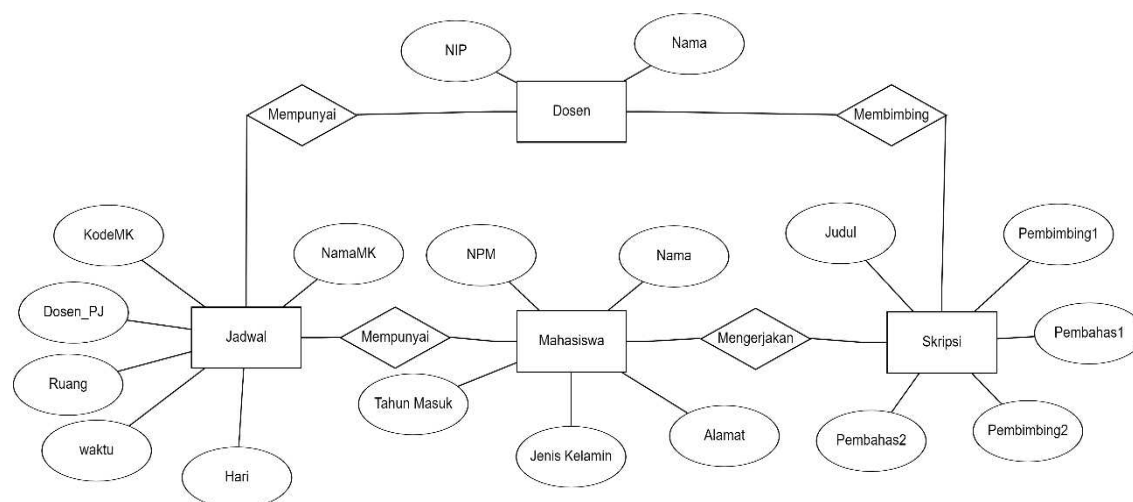
Tahapan komunikasi dilakukan untuk menggambarkan fungsi, fitur dari sistem dan *output* dari sistem yang dibuat agar sesuai dengan keinginan klien. Tahapan komunikasi merupakan tahapan pengumpulan data yang akan digunakan untuk pengembangan sistem.

b. *Planning*

Tahapan perencanaan menentukan prioritas pembuatan sistem berdasarkan bobot dan dibuat secara berkelompok untuk selanjutnya dilakukan proses pemodelan dari sistem yang akan dibuat. Tahapan perencanaan menentukan estimasi waktu yang diperlukan untuk mengembangkan sistem, jadwal dari pembuatan sistem dan penandaan proses pengerjaan sistem.

c. *Modeling*

Tahapan analisis kebutuhan dilakukan dengan melakukan analisa permasalahan dalam bentuk *Entity Relationship Diagram (ERD)*, kandidat *database* dan analisa kandidat API. Diagram relasi entitas dapat dilihat pada Gambar 2.



Gambar 2. Diagram Relasi Entitas

d. Construction

Tahapan konstruksi pada metode *waterfall* adalah tahapan pembuatan program yang dibuat. Bahasa pemrograman yang digunakan adalah bahasa pemrograman PHP dengan *framework CodeIgniter*. Tahapan konstruksi pada penelitian ini akan dilakukan dalam beberapa sub proses, antara lain:

1. Membangun lingkungan pengembangan sistem.
2. Mendesain struktur tabel pada *database*.
3. Mengembangkan akses titik *API*.

Tahapan pengujian akan dilakukan setelah pembuatan program telah selesai. Tahapan pengujian dilakukan untuk memastikan bahwa sistem yang telah dibuat berjalan sesuai dengan kebutuhan dan sesuai dengan yang diinginkan.

e. Deployment

Tahapan penyebaran adalah tahapan final dari sebuah pembuatan sistem. Setelah melakukan tahapan komunikasi, tahapan perencanaan, tahapan pemodelan dan tahapan konstruksi maka sistem yang telah dibuat bisa digunakan oleh pengguna. Tahapan penyebaran juga melakukan pemeliharaan secara berkala terhadap sistem yang telah dibuat sehingga sistem tetap dapat berkembang sesuai dengan fungsinya.

4. HASIL DAN PEMBAHASAN

Pembuatan struktur API telah membuat sebuah daftar API yang menjelaskan hubungan antara *resource* dengan klasifikasinya terhadap *path* dan *query string* serta *request method* yang digunakan. Klasifikasi *resource* mempunyai dua landasan yaitu klasifikasi berdasarkan *path* dan *query string* dan klasifikasi berdasarkan *request method*. Klasifikasi berdasarkan *path* dan *query string* memiliki dua pola, antara lain:

1. Titik akses */resource?query=string* yang digunakan untuk operasi mendapatkan daftar data dari anggota *resource* yang dimaksud.
2. Titik akses */resource/id?query=string* yang digunakan untuk mendapatkan rincian berdasarkan id *resource* yang dimaksud.

Sedangkan untuk klasifikasi berdasarkan *request method* digunakan dua jenis *request method*. Klasifikasi ini akan menjadi klasifikasi penunjang untuk klasifikasi *path* dan *query string* dan membantu menentukan jenis *method* yang akan digunakan kepada *resource*. *Request method* yang digunakan adalah:

1. *Request method GET* yang digunakan untuk mendapatkan daftar dari anggota *resource* dan detail dari anggota *resource*.
2. *Request method POST* yang digunakan untuk melakukan pencarian anggota *resource*.

Daftar API dapat dilihat pada Tabel 1.

Tabel 1. Daftar API

NO	Nama Proses	Request Method	URI	Keterangan
1	List mahasiswa	GET	/mahasiswa/listmahasiswa	Menampilkan daftar seluruh mahasiswa
2	Get mahasiswa	GET	/mahasiswa/detailmahasiswa/NPM	Menampilkan rincian data mahasiswa
3	Get kata kunci pencarian mahasiswa	GET	/mahasiswa/search/<katakunci>	Menampilkan data mahasiswa berdasarkan kata kunci pencarian
4	Post Data mahasiswa	POST	/mahasiswa/filtermahasiswa/tahunmasthuk/jurusan	Menampilkan data mahasiswa berdasarkan jurusan atau tahun masuk mahasiswa
5	List Dosen	GET	/dosen/listdosen	Menampilkan daftar seluruh data dosen
6	Get Dosen	GET	/dosen/detaildosen/NIP	Menampilkan rincian data dosen
7	Get kata kunci pencarian dosen	GET	/dosen/search/<katakunci>	Menampilkan data dosen berdasarkan kata kunci pencarian dosen
8	List jadwal	GET	/jadwal/listjadwal	Menampilkan daftar jadwal perkuliahan
9	Get jadwal	GET	/jadwal/detailjadwal/KodeMK	Menampilkan rincian jadwal perkuliahan
10	Get kata kunci pencarian jadwal	GET	/jadwal/search/<katakunci>	Menampilkan data perkuliahan berdasarkan kata kunci pencarian jadwal
11	List Skripsi	GET	/skripsi/listskripsi	Menampilkan daftar data skripsi
12	Get skripsi	GET	/skripsi/detailskripsi/judul	Menampilkan rincian data skripsi
13	Post data skripsi	POST	/skripsi/seacrh/<katakunci>	Menampilkan data skripsi berdasarkan kata kunci pencarian skripsi

Penetapan parameter berdasarkan kebutuhan *request* API yang diklasifikasikan berdasarkan tabel dalam *database*. Terdapat beberapa keterangan dalam tabel yang telah ditetapkan yaitu:

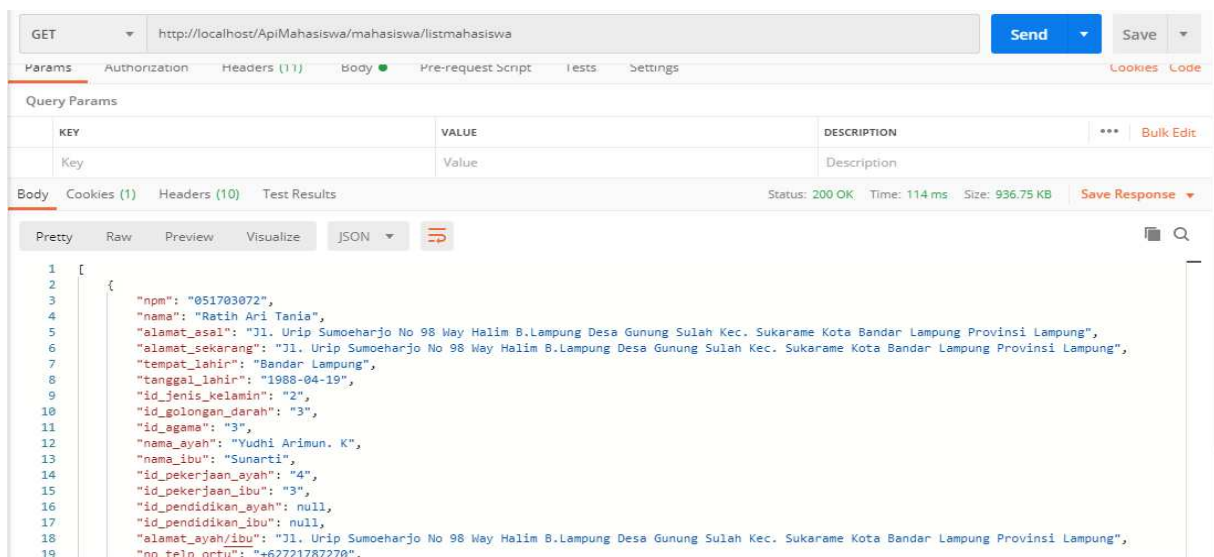
1. Kolom No sebagai indeks parameter
2. Kolom nama parameter sebagai nama dari parameter.

3. Kolom tipe data yang merupakan keterangan tipe data dari parameter.
4. Kolom parameter yang menginformasikan parameter yang bisa atau tidak bisa dikirimkan terhadap request method. Informasi pada kolom ini diklasifikasikan sebagai berikut:
 - a. (x) yang berarti parameter yang bisa dikirimkan
 - b. (y) yang berarti parameter yang tidak bisa dikirimkan
 - c. (o) yang berarti parameter yang dapat dikirimkan (opsional)

Parameter API dalam mengakses *resource* dibagi berdasarkan setiap *resource*. Terdapat 4 *resource* yang digunakan yaitu:

1. Mahasiswa
Dalam *database*, tabel mahasiswa menyimpan informasi berupa data diri mahasiswa.
2. Dosen
Dalam *database*, tabel dosen menyimpan informasi berupa data diri dosen.
3. Penjadwalan perkuliahan
Dalam *database*, tabel jadwal perkuliahan menyimpan informasi berupa data jadwal perkuliahan.
4. Skripsi
Dalam *database*, tabel skripsi menyimpan informasi berupa data skripsi.

Daftar API yang telah dibuat dilakukan uji coba dalam mengakses API dengan menggunakan perangkat lunak postman. Contoh akses API *List* mahasiswa dapat dilihat pada Gambar 3.



Gambar 3. Akses API *List* Mahasiswa

Berdasarkan hasil dari titik akses API maka telah didapat hasil pengujian fungsional dari keseluruhan titik akses API. Pengujian fungsional dimuat dalam format tabel yang dapat dilihat pada Tabel 2.

Tabel 2. Hasil Pengujian Fungsional API

No	Nama Proses	Keterangan	Hasil pengujian
1	List mahasiswa	Menampilkan daftar seluruh mahasiswa	Berhasil
2	Get mahasiswa	Menampilkan rincian data mahasiswa	Berhasil
3	Get kata kunci pencarian mahasiswa	Menampilkan data mahasiswa berdasarkan kata kunci pencarian	Berhasil
4	Post Data mahasiswa	Menampilkan data mahasiswa berdasarkan jurusan atau tahun masuk mahasiswa	Berhasil
5	List Dosen	Menampilkan daftar seluruh data dosen	Berhasil
6	Get Dosen	Menampilkan rincian data dosen	Berhasil
7	Get kata kunci pencarian dosen	Menampilkan data dosen berdasarkan kata kunci pencarian dosen	Berhasil
8	List jadwal	Menampilkan daftar jadwal perkuliahan	Berhasil
9	Get jadwal	Menampilkan detail jadwal perkuliahan	Berhasil
10	Get kata kunci pencarian jadwal	Menampilkan data perkuliahan berdasarkan kata kunci pencarian jadwal	Berhasil
11	List skripsi	Menampilkan daftar data skripsi	Berhasil
12	Get skripsi	Menampilkan detail data skripsi	Berhasil
13	Post data skripsi	Menampilkan data skripsi berdasarkan kata kunci pencarian skripsi	Berhasil

5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan bahwa pengembangan *application service* data mahasiswa pada *website* Jurusan Ilmu Komputer telah berhasil dikembangkan dan telah berhasil membuat tiga belas daftar API.

DAFTAR PUSTAKA

- [1] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," University Of California, Irvine, 2000.
- [2] M. R. Hakim, "Prototype Sistem Informasi Akademik Berbasis Mobile Menggunakan Java Script Object Notation (JSON)," Skripsi STIKOM Surabaya, Surabaya, 2012.
- [3] P. H. and J. K. Kawista, Pemrograman Web, Bandung: Informatika, 2015.

- [4] B. R. I. H. and E. R. K, Modul Pemrograman Web HTML, PHP&MySQL, Bandung: Modula, 2012.
- [5] G. M. Rama and A. K. , "Software Practice and Experience," Wiley Online Library, New Jersey, 2013.
- [6] S. and V. P. , Pemrograman Web dengan menggunakan PHP dan Framework Codeigniter, Yogyakarta: Deepublish, 2016.
- [7] "Postman is the most complete API Development Environment," Postdot Technology, 2019. [Online]. Available: <https://www.getpostman.com/postman>. [Accessed 5 Juni 2020].