

Rancang Bangun Modul Portofolio Pegawai pada Aplikasi MyITS Human Capital Management Dengan Arsitektur Event Driven

Rida Adila, Rizky Januar Akbar, dan Hudan Studiawan

Departemen Teknik Informatika, Institut Teknologi Sepuluh Nopember (ITS)

e-mail: rja@its.ac.id

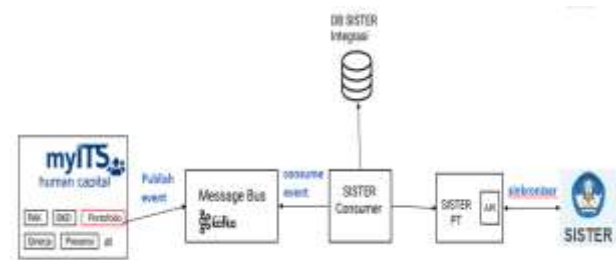
Abstrak—MyITS Human Capital Management (MyITS HCM) adalah aplikasi yang sedang dikembangkan dan berfungsi untuk menyediakan berbagai keperluan terkait dengan kepegawaian ITS. Pada MyITS HCM, terdapat modul portofolio yang berfungsi untuk melakukan pengelolaan data portofolio pegawai ITS. Modul portofolio berfungsi sebagai sumber data yang banyak dimanfaatkan aplikasi lain. Salah satu aplikasi tersebut ialah SISTER ITS, yaitu aplikasi yang dimiliki oleh DIKTI, dimana dosen dapat melakukan manajemen data portofolio yang dimiliki. Untuk meringankan tugas dari dosen ITS, diperlukan adanya integrasi antara modul portofolio MyITS HCM dengan SISTER ITS. Integrasi dilakukan dengan menerapkan arsitektur Event Driven dengan Apache Kafka. Pengujian integrasi dilakukan pada tahap pengembangan dikarenakan dari DIKTI masih belum membuka akses ke production. Hasil yang didapatkan dari pengujian integrasi ialah semua portofolio berhasil terintegrasi namun ada 1 fitur portofolio yang gagal, dikarenakan endpoint API dari pihak SISTER masih mengalami error.

Kata Kunci—Event Driven, Integrasi, MyITS Human Capital Management, Portofolio, SISTER ITS.

I. PENDAHULUAN

DI ITS penggunaan aplikasi berbasis web sudah menjadi rutinitas dalam kegiatan sehari-hari. Salah satu aplikasi berbasis web ITS yang digunakan untuk menyediakan berbagai keperluan yang berhubungan dengan keperluan untuk kepegawaian ITS ialah myITS Human Capital Management. Aplikasi myITS Human Capital Management ialah aplikasi yang berfungsi untuk melakukan pengelolaan dan pengembangan yang terkait dengan SDM kepegawaian ITS. Pada aplikasi myITS Human Capital Management akan berisi modul-modul aplikasi lain yang berfungsi untuk menangani segala kebutuhan terkait kepegawaian ITS. Contohnya seperti kinerja, presensi pegawai, PAK (Penilaian Angka Kredit), karir tendik, rekrutmen, layanan administrasi kepegawaian, dan portofolio pegawai. Aplikasi myITS Human Capital Management sendiri masih tetap memerlukan pengembangan sistem. Salah satunya ialah dalam melakukan pengelolaan dan pengaksesan data portofolio kepegawaian ITS.

Dimana data mengenai portofolio kepegawaian ini nantinya akan berisi informasi lengkap mengenai data diri pegawai, riwayat hidup pegawai, karya ilmiah yang pernah dibuat oleh pegawai, serta segala riwayat kegiatan yang pernah dilakukan oleh pegawai. Semua hal tersebut akan dikelola pada suatu aplikasi modul yakni modul portofolio pegawai. Selain itu, modul portofolio pegawai ini juga akan



Gambar 1. Skema integrasi dengan event driven.

berfungsi sebagai sumber data kepegawaian yang banyak dimanfaatkan oleh aplikasi lain, baik untuk aplikasi internal ITS maupun eksternal. Salah satu aplikasi yang akan memanfaatkan data dari modul portofolio pegawai ialah SISTER ITS.

SISTER ITS merupakan sebuah sistem informasi sumberdaya terintegrasi, yang mana sistem informasi ini dimiliki oleh Direktorat Jenderal Pendidikan Tinggi. Namun pada aplikasi ini juga membutuhkan data-data terkait kepegawaian ITS yang dimiliki oleh modul portofolio pegawai myITS Human Capital Management. Sehingga jika pegawai ITS ingin melakukan penambahan, penyuntingan, maupun penghapusan data pada aplikasi SISTER ITS yang mana sebenarnya data tersebut juga ada di modul portofolio myITS Human Capital Management dan dilakukan proses operasi data yang sama pula pada aplikasi tersebut, pegawai tersebut harus melakukan operasi-operasi (tambah, sunting, hapus) itu lagi pada aplikasi SISTER ITS.

Maka dari itu, untuk memudahkan serta meminimalkan tugas dari pegawai ITS dalam melakukan proses pengelolaan data di aplikasi SISTER, diperlukan adanya integrasi antara modul portofolio pegawai myITS Human Capital Management dan aplikasi SISTER. Yang mana nantinya aplikasi modul portofolio pegawai ini akan mengirimkan data-data *event* ke *message broker* (Apache Kafka), kemudian aplikasi SISTER akan mengonsumsi data-data *event* yang ada di *message broker* tersebut untuk kemudian dilakukan proses penambahan, penyuntingan, dan penghapusan data sesuai yang terjadi pada modul portofolio di MyITS Human Capital Management.

Tujuan dari penelitian ini adalah dapat mengintegrasikan aplikasi modul portofolio MyITS Human Capital Management dengan SISTER ITS.

II. TINJAUAN PUSTAKA

A. Event Driven Architecture

Event Driven Architecture merupakan sebuah arsitektur

Tabel 1.
Jenis portofolio dan daftar fitur

Kode Portofolio	Jenis Portofolio	Daftar Fitur yang Ada
P01	Anggota Profesi	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P02	Bahan Ajar	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P03	Detasering	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P04	Pengembangan Diri	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P05	HKI (Hak Kekayaan Intelektual)	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P06	Pembicara	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P07	Penelitian	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P08	Pengabdian	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P09	Pengelola Jurnal	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P10	Kepanitiaan	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P11	Karya Ilmiah	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P12	Penghargaan	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P13	Riwayat Pekerjaan	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P14	Sertifikasi	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen
P15	Tes	Menambah data, Menyunting data, Menghapus data, Mengupload dokumen, Menghapus dokumen, Menyunting detail dokumen

terdistribusi asinkronus yang digunakan pada aplikasi yang membutuhkan skalabilitas dan performa yang tinggi. Dimana pada arsitektur ini sebuah sistem akan melakukan eksekusi setelah menerima satu atau beberapa *event* dari sistem lain. Pada *event driven architecture*, terdapat 2 jenis topologi utama, yakni topologi *mediator* dan topologi *broker* [1].

Topologi *mediator* merupakan suatu topologi pada arsitektur *event driven* dimana akan ada sebuah mediator yang bertugas untuk melakukan kontrol *event-event* kepada setiap sistem yang akan menerima *event* dari mediator tersebut [1]. Sedangkan topologi *broker* merupakan suatu topologi pada arsitektur *event driven* dimana akan terdapat suatu *broker* sebagai perantara penengah antara pihak *producer* dan *consumer*. Dimana *producer* ialah sistem yang akan melakukan pengiriman *event* kepada *broker* tersebut dan pihak *consumer* akan melakukan konsumsi data dari *event* yang ada di *broker* tersebut [1].

B. Apache Kafka

Apache Kafka merupakan salah satu *message broker* yang berfungsi sebagai sebuah perantara dimana antar aplikasi dapat melakukan pertukaran data. Diantara aplikasi-aplikasi yang akan melakukan pertukaran data, akan ada aplikasi yang berperan sebagai *producer* dan *consumer*. Dimana aplikasi yang berperan sebagai *producer* akan bertugas mengirimkan data *event* ke *message broker*, dan aplikasi yang berperan sebagai *consumer* akan mendapatkan data dengan melakukan konsumsi data yang diinginkan dari *message broker*.

C. PHP

PHP adalah bahasa pemrograman yang biasanya untuk digunakan untuk melakukan pengembangan aplikasi web. Bahasa pemrograman PHP mempunyai *library* (pustaka) yang dapat digunakan untuk mempermudah pengembangan web.

D. Black Box Testing

Black Box Testing adalah salah satu metode ujicoba dalam perangkat lunak terhadap fungsionalitas atau kegunaan aplikasi, biasanya dilakukan berdasarkan kasus penggunaan atau spesifikasi kebutuhan.

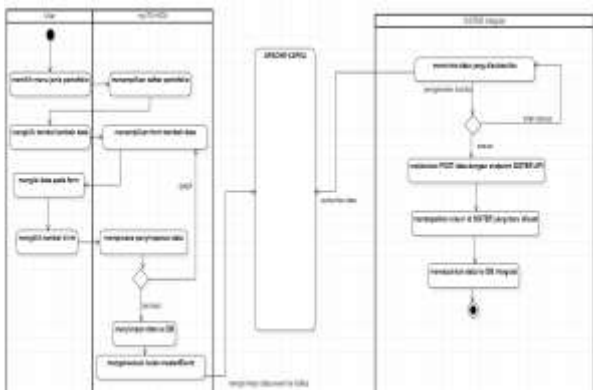
Pada *Black Box Testing* ini dilakukan pengujian yang didasarkan pada detail aplikasi seperti tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi, dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh customer. *Black-box testing* bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya hanya terfokus pada informasi domain.

III. ANALISIS DAN PERANCANGAN SISTEM

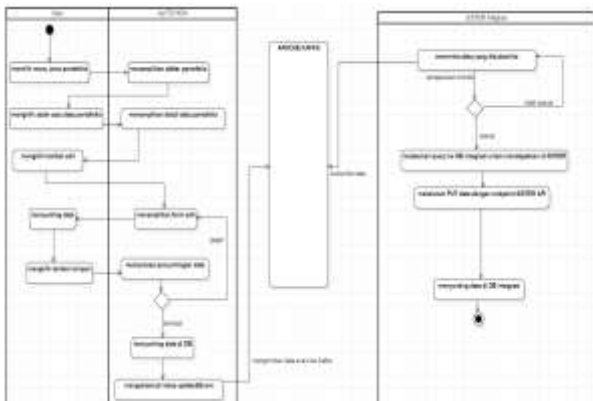
A. Analisis

Permasalahan utama yang diangkat dalam penelitian ini adalah bagaimana dapat mengintegrasikan modul portofolio pegawai pada aplikasi myITS Human Capital Management ITS dengan SISTER ITS.

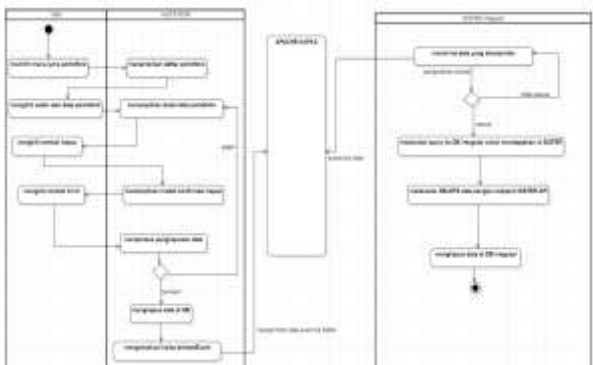
Modul portofolio pegawai sendiri dapat berfungsi sebagai master data, yang mana data-data yang ada pada modul tersebut juga akan banyak dibutuhkan oleh aplikasi-aplikasi lain, salah satunya ialah SISTER ITS. SISTER ITS merupakan suatu aplikasi yang dimiliki oleh Kemenristekdikti, dimana para dosen ITS dapat melakukan akses pengelolaan data-data portofolio yang dimiliki. Pada bagian ini akan dibahas terkait daftar fitur yang ada dari pihak aplikasi yang akan mengirimkan data, yakni modul portofolio MyITS Human Capital Management. Sedangkan untuk daftar fitur pada aplikasi *consumer* yaitu SISTER ITS, menyediakan *endpoint* API yang dapat diakses oleh administrator perguruan tinggi, yang mana daftar *endpoint* tersebut bisa dilihat pada daftar pustaka poin [2]. Jenis portofolio dapat



Gambar 2. Rancangan algoritma created event.



Gambar 3. Rancangan algoritma updated event.



Gambar 4. Rancangan algoritma deleted event.

dilihat pada Tabel 1.

B. Perancangan Arsitektur

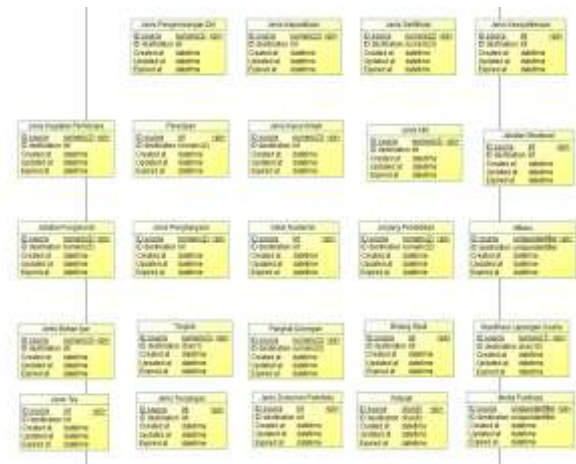
Pada bagian ini akan dijelaskan mengenai arsitektur yang digunakan untuk mengintegrasikan modul portofolio myITS Human Capital Management dengan SISTER ITS. Dalam menerapkan arsitektur *event driven* pada penelitian ini, akan menggunakan sebuah *message broker* yaitu Apache Kafka. Untuk skema integrasinya sendiri tertera pada Gambar 1.

Modul portofolio yang ada pada myITS Human Capital Management pada kasus ini akan berperan sebagai *producer* atau pihak yang akan mengirimkan data *event* ke Apache Kafka. Sedangkan SISTER ITS akan berperan sebagai *consumer* atau pihak yang akan mengkonsumsi data *event* yang dikirimkan oleh modul portofolio myITS Human Capital Management.

Dimana data-data yang ada pada modul portofolio myITS Human Capital Management akan terkirim ke topik-topik yang ada pada Apache Kafka setelah user pada aplikasi tersebut melakukan proses tambah, sunting, dan hapus data.



Gambar 5. Rancangan basis data integrasi bagian 1.



Gambar 6. Rancangan basis data integrasi bagian 2.

Kemudian dari pihak SISTER ITS, akan terdapat sebuah file *consumer.php* yang berfungsi untuk melakukan proses konsumsi data dari topik-topik yang telah di subscribe untuk menerima data-data event.

Kemudian data yang telah diterima akan diproses dengan SISTER API untuk dilakukan operasi tambah, sunting, dan hapus data ke aplikasi SISTER sesuai apa yang dilakukan oleh user pada aplikasi myITS Human Capital Management di modul portofolio. Selain itu juga terdapat suatu database integrasi yang berfungsi untuk menyimpan hasil pemetaan data-data referensi antara myITS Human Capital Management dengan SISTER ITS dan untuk menyimpan pemetaan id antara data portofolio yang baru dibuat di myITS Human Capital Management dan data portofolio yang baru dibuat di SISTER ITS.

C. Perancangan Algoritma

Pada bagian ini akan menjelaskan mengenai rancangan algoritma *event driven* yang akan digunakan untuk proses *publish* data dan *consume* data. Rancangan algoritma terdiri atas 3 bagian, yaitu algoritma *created event*, *updated event*, dan *deleted event*.

Pada setiap gambar algoritma, terdapat 4 pihak, yaitu *user*, MyITS HCM, Apache Kafka dan SISTER Integrasi. Rancangan algoritma diilustrasikan dalam bentuk diagram aktivitas yang bisa dilihat pada gambar 2 untuk algoritma *created event*, gambar 3 untuk algoritma *updated event*, dan gambar 4 untuk algoritma *deleted event*.

```

class PublisherConfig
{
public function getPublisherConfig()
{
    $conf = new \RdKafka\Conf();
$conf-
>set('bootstrap.servers',config('kafka.bootstrap_
servers'));
    $conf->set('request.required.acks',
'all');
    $conf->set('enable.idempotence', true);

    $producer = new \RdKafka\Producer($conf);

    return $producer;
}

// ...
// definisi daftar topik
// ...
}
    
```

Gambar 7. Kode sumber 1 implementasi konfigurasi producer.

Tabel 2.
Tabel lingkungan uji coba

Spesifikasi	Deskripsi
Jenis Perangkat	Laptop
CPU	AMD Ryzen 5 4500U
RAM	8GB
Sistem Operasi	Windows 10

D. Perancangan Basis Data

Pada bagian ini akan menjelaskan mengenai rancangan basis data yang akan digunakan. Gambar 5 menunjukkan rancangan basis data integrasi untuk tabel transaksi yang berfungsi untuk menyimpan data-data utama dan menyimpan hasil pemetaan *id* portofolio, sedangkan pada gambar 6 menunjukkan dari rancangan basis data integrasi untuk tabel tabel referensi yang berfungsi untuk menyimpan hasil pemetaan *id* referensi MyITS HCM dan SISTER ITS.

IV. IMPLEMENTASI

A. Implementasi Pada Sisi Aplikasi Producer

Modul portofolio pada aplikasi MyITS Human Capital Management pada penelitian ini akan bertindak sebagai pihak aplikasi *producer*, yaitu pihak aplikasi yang akan mengirimkan data *event* ke Kafka. Untuk implementasi integrasi pada sisi *producer* menggunakan *library* dari *rdkafka* PHP.

Kode sumber 1 yang tertera pada Gambar 7 merupakan sebuah *PublisherConfig.php* dimana file ini berfungsi untuk mengatur konfigurasi *producer* yang diinginkan. Dan dibawah ini ialah salah satu implementasi *event producer* dari portofolio penelitian. Kode sumber 2 tertera pada Gambar 8.

```

class PenelitianCreatedEvent extends
PublisherConfig
{
    private array $data;

    public function __construct($field)
    {
        $this->data['id_penelitian_hcm'] =
$field->id_litabmas;
        $this->data['judul'] = $field->judul;
        $this->data['nama_sdm'] = $field-
>nama_sdm;
        $this->data['nidn_sdm'] = $field-
>nidn_sdm;
        $this->data['lokasi'] = $field-
>lokasi_kegiatan;
        $this->data['nama_event'] =
'penelitian_created';
        // ...
        // daftar data yang akan dikirim
        // ...

        try {
            $producer = $this-
>getPublisherConfig();
            $topic_array = $this-
>getArrayOfTopic();
            $litabmas = $producer-
>newTopic($topic_array['penelitian']);
            $litabmas->produce(RD_KAFKA_PARTITION_UA, 0,
                json_encode($this->data));
            $producer->poll(0);
            $producer->flush(1000);
        }
        catch(\Exception $e)
        {
            Log::channel('producer')->error($e-
>getMessage());
        }
    }
    // ...
    // ...
    // ...
}
    
```

Gambar 8. Kode sumber 2 implementasi penelitian created event.

B. Implementasi Pada Sisi Aplikasi Consumer

SISTER ITS pada kasus integrasi kali ini akan bertindak sebagai pihak *consumer*, yakni pihak yang akan menerima data atau melakukan *consume* data dari Kafka. Untuk dapat melakukan hal tersebut, diperlukan adanya suatu file *consumer.php*, yang mana dalam implementasinya

Tabel 3.
Hasil pengujian integrasi

Kode	Kasus Penggunaan	Terpenuhi	
		Ya	Tidak
UJ-001	Menambah data anggota profesi	✓	
UJ-002	Menyunting data anggota profesi	✓	
UJ-003	Menghapus data anggota profesi	✓	
UJ-004	Menambah data bahan ajar	✓	
UJ-005	Menyunting data bahan ajar	✓	
UJ-006	Menghapus data bahan ajar	✓	
UJ-007	Menambah data detasering	✓	
UJ-008	Menyunting data detasering	✓	
UJ-009	Menghapus data detasering	✓	
UJ-010	Menambah data pengembangan diri	✓	
UJ-011	Menyunting data pengembangan diri	✓	
UJ-012	Menghapus data pengembangan diri	✓	
UJ-013	Menambah data HKI (Hak Kekayaan Intelektual)	✓	
UJ-014	Menyunting data HKI (Hak Kekayaan Intelektual)	✓	
UJ-015	Menghapus data HKI (Hak Kekayaan Intelektual)	✓	
UJ-016	Menambah data orasi ilmiah	✓	
UJ-017	Meyunting data orasi ilmiah	✓	
UJ-018	Menghapus data orasi ilmiah	✓	
UJ-019	Menambah data pembicara	✓	
UJ-020	Menyunting data pembicara	✓	
UJ-021	Menghapus data pembicara	✓	
UJ-022	Menambah data penelitian	✓	
UJ-023	Menyunting data penelitian	✓	
UJ-024	Menghapus data penelitian	✓	
UJ-025	Menambah data pengabdian	✓	
UJ-026	Menyunting data pengabdian	✓	
UJ-027	Menghapus data pengabdian	✓	
UJ-028	Menambah data pengelola jurnal	✓	
UJ-029	Menyunting data pengelola jurnal	✓	
UJ-030	Menghapus data pengelola jurnal	✓	
UJ-031	Menambah data penghargaan	✓	
UJ-032	Menyunting data penghargaan	✓	
UJ-033	Menghapus data penghargaan	✓	
UJ-034	Menambah data kepanitiaan	✓	
UJ-035	Menyunting data kepanitiaan		✓
UJ-036	Menghapus data kepanitiaan	✓	
UJ-037	Menambah data karya ilmiah	✓	
UJ-038	Menyunting data karya ilmiah	✓	
UJ-039	Menghapus data karya ilmiah	✓	
UJ-040	Menambah data riwayat pekerjaan	✓	
UJ-041	Menyunting data riwayat pekerjaan	✓	
UJ-042	Menghapus data riwayat pekerjaan	✓	
UJ-043	Menambah data sertifikasi profesi	✓	
UJ-044	Menyunting data sertifikasi profesi	✓	
UJ-045	Menghapus data sertifikasi profesi	✓	
UJ-046	Menambah data tes	✓	
UJ-047	Menyunting data tes	✓	
UJ-048	Menghapus data tes	✓	
UJ-049	Menambah data visiting scientist	✓	
UJ-050	Menyunting data visiting scientist	✓	
UJ-051	Menghapus data visiting scientist	✓	

menggunakan *library rdafka*.

Consumer akan terus menerus melakukan *consume* data dari daftar topik yang telah di *subscribe*. Ketika *consumer* telah menerima data, maka data tersebut perlu dicek kembali apakah merupakan data dengan nilai *id* jenis sdm sama dengan dosen atau bukan, dikarenakan pada *SISTER* sendiri yang tersedia ialah data-data dosen. Jika data tersebut merupakan data dosen, maka data akan diproses menggunakan fungsi *endpoint API* sesuai dengan nama *event* data nya.

Gambar 9 adalah file *consumer.php* yang digunakan pada *SISTER ITS*, dimana akan dibagi menjadi beberapa bagian, dikarenakan kode sumber secara utuh tidak terlalu jelas dan sulit untuk dibaca.

V. UJI COBA DAN EVALUASI

Dibutuhkan suatu lingkungan uji coba sebagai standar

untuk melakukan uji coba, lingkungan ini akan digunakan untuk pengujian skenario setiap kasus penggunaan dan pengujian kualitas perangkat lunak. Spesifikasi lingkungan uji coba yang digunakan saat pengujian perangkat lunak yang dikembangkan dalam penelitian ini tertera pada Tabel 2.

Ringkasan hasil pengujian integrasi dilakukan berdasarkan skenario ujicoba integrasi. Hasil pengujian tertera pada Tabel 3.

VI. KESIMPULAN DAN SARAN

Berdasarkan penjabaran pada subbab-subbab sebelumnya, dari hasil uji coba yang telah dilakukan, dapat disimpulkan desain dari modul portofolio *MyITS Human Capital Management* yang dapat berfungsi sebagai master data dibuat generik yakni agar *MyITS Human Capital Management* dapat diintegrasikan dengan banyak aplikasi lain, tidak hanya *SISTER ITS* saja.

```

class Consumer
{
public function execute()
{
    $event_type = new EventType();
    $conf = new \RdKafka\Conf();
    $conf->set('group.id', 'test-12');
    $conf->set('bootstrap.servers',
'10.199.16.93:9092');
    $conf->set('auto.offset.reset',
'smallest');
    $consumer = new
\RdKafka\KafkaConsumer($conf);
    // ...
    // looping consume data
    // ...
}
// ...
// definisi daftar topik
// ...
}

```

Gambar 9. Kode sumber 3 implementasi consumer.

Arsitektur event *driven* berhasil diterapkan pada modul

portofolio MyITS Human Capital Management dengan menyediakan *producer* pada sisi aplikasi MyITS Human Capital Management dan menyediakan *consumer* pada sisi aplikasi SISTER ITS.

Pada tahap *development* sudah dicoba integrasi antara modul portofolio MyITS Human Capital Management dengan SISTER ITS dan integrasi berhasil dilakukan, Namun ada 1 skenario yang tidak berhasil dilakukan karena *endpoint* terkait dari SISTER masih tidak bisa. Sedangkan pada tahap *production* belum dilakukan integrasi karena dari pihak DIKTI masih belum membuka akses *production*.

Beberapa saran yang dapat diberikan kepada penulis terkait penelitian ini adalah meliputi sebagai berikut: pada aplikasi MyITS Human Capital juga terdapat beberapa modul profil dan kepegawaian yang dapat diintegrasikan dengan SISTER ITS. Sehingga integrasi perlu diimplementasikan juga.

Perlu diimplementasikan semua integrasi pada MyITS Human Capital Management terhadap aplikasi-aplikasi lain yang membutuhkan data dari My ITS Human Capital Management.

DAFTAR PUSTAKA

- [1] M. Richards and N. Ford, *Fundamentals of Software Architecture an Engineering Approach*. California: O'Reilly Media, 2020.
- [2] S. Zhelev and A. Rozeva, "Using Microservices and Event Driven Architecture for Big Data Stream Processing," in *Proceedings of the 45th International Conference on Application of Mathematics in Engineering and Economics*. 2019.