

RESTful Web service Untuk Integrasi Antar Sistem Informasi

Teguh Andriyanto¹⁾, Rini Indriati²⁾

¹⁾²⁾ Prodi Sistem Informasi, Fakultas Teknik,
Universitas Nusantara PGRI Kediri

Jl. KH. Ahmad Dahlan 76 Kediri

¹⁾ teguh@unpkediri.ac.id

²⁾ rini.indriati@unpkediri.ac.id

Abstrak – Di Universitas Nusantara PGRI (UN PGRI) Kediri terdapat beberapa sistem informasi yang masih belum saling terintegrasi. Akibatnya terjadi redundansi data antar sistem informasi tersebut, Contohnya adalah data mahasiswa. Seringkali terdapat perbedaan data yang seharusnya sama misalnya nama mahasiswa akibat input data yang sama dilakukan pada setiap aplikasi sistem informasi. Untuk mengatasi permasalahan tersebut maka perlu dilakukan integrasi data antar sistem informasi. Pada penelitian telah dihasilkan sistem integrasi data antara sistem informasi akademik dan sistem informasi ujian skripsi. Sistem informasi tersebut merupakan aplikasi berbasis web yang dibangun menggunakan bahasa pemrograman php dan database MySQL. Data yang akan diintegrasikan adalah data mahasiswa. Integrasi data dibangun menggunakan RESTful web service, bahasa pemrograman php dan database MySQL. Data request dan data respon webservice menggunakan format JSON.

Kata Kunci: Integrasi, RESTful Web service, Sistem Informasi

1. PENDAHULUAN

Saat ini terdapat beberapa aplikasi Sistem Informasi di UN PGRI Kediri. Beberapa diantaranya adalah Sistem Informasi Akademik, Sistem Informasi Karya Ilmiah dan Sistem Informasi KKN. Aplikasi-aplikasi tersebut diatas berjalan sendiri-sendiri, menggunakan database terpisah. Setiap aplikasi memiliki data dasar (*master*) sendiri-sendiri seperti data mahasiswa, dosen, mata kuliah. Akibat dari belum terintegrasinya aplikasi-aplikasi tersebut maka input data yang sama akan dilakukan di setiap aplikasi. Hal ini tentunya rawan terjadi salah input data. Misalnya nama untuk mahasiswa yang sama berbeda antar sistem informasi. Mahasiswa maupun dosen juga harus memiliki *login* yang berbeda untuk setiap aplikasi tersebut. Hal ini tentunya membuat aplikasi berjalan tidak efektif dan kemungkinan terjadi ketidaksinkronan data.

Salah satu solusi untuk mengatasi permasalahan tersebut yaitu dengan melakukan integrasi data. Dalam penelitian ini dilakukan integrasi data antara Sistem Informasi Ujian Skripsi dengan Sistem Informasi Akademik. Terdapat banyak cara untuk melakukan integrasi data. Dalam penelitian ini digunakan RESTful web service untuk proses integrasi data. Data yang dipilih adalah data mahasiswa dalam aplikasi Sistem Informasi Ujian Skripsi yang diintegrasikan dengan data mahasiswa dari Sistem Informasi Akademik.

2. TINJAUAN PUSTAKA

Metode *Representational State Transfer* (REST) dikembangkan oleh Fielding (2000). REST adalah gaya arsitektur perangkat lunak yang berisi panduan dan praktik terbaik untuk membuat *scalable web service*. REST merupakan sebuah batasan set terkoordinasi yang diterapkan pada desain komponen

dalam sistem *hypermedia* terdistribusi yang dapat membuat arsitektur lebih mudah dirawat. REST secara efisien menggunakan HTTP (Kumari, 2015).

REST didasari oleh empat prinsip utama teknologi, yaitu:

1. *Resource identifier through Uniform Resource Identifier* (URI)
2. *Uniform interface* (sumberdaya CRUD menggunakan operasi PUT, GET, POST, dan DELETE)
3. *Self-descriptive messages* (sumberdaya tidak terikat sehingga dapat mengakses konten HTML, XML, PDF, JPEG, plain text, meta data, dll)
4. *Stateful interactions through hyperlinks* (bersifat *stateless*) (Pautasso, 2008).

REST memiliki keuntungan yaitu (Kumari, 2015):

1. Support komunikasi *stateless*
2. Mudah untuk dipelajari dan diterapkan
3. Penggunaan HTTP secara efisien
4. Membutuhkan bandwidth yang kecil pada saat mengirim pesan dalam format JSON (Javascript Object Notation) dan juga dapat digunakan format lain

5. Untuk keamanan menggunakan standart HTTP
 6. REST dapat dikonsumsi oleh beberapa *client*
 7. REST membuat data sebagai sumber daya
- Sedangkan kelemahan REST adalah:

1. Tidak cocok untuk data besar
2. Dibandingkan SOAP, REST tidak meliputi semua jenis standar *web service* seperti keamanan, transaksi dll.
3. REST tidak handal

Penelitian mengenai *web service* telah banyak dilakukan dengan berbagai studi kasus. Santoso (2011) melakukan penelitian dengan membuat perangkat lunak untuk mengintegrasikan data dari beberapa situs penjualan (Amazon, Commission

Junction, dan Ebay). Perangkat lunak tersebut menangani penambahan, perubahan dan penghapusan data produk yang tersimpan dalam *database*. Sementara itu Susanto (2013) menggunakan REST *web service* untuk menghubungkan sistem pembayaran mahasiswa Udinus dengan sistem perbankan. Sistem yang dibangun memanfaatkan SMS pemberitahuan dari bank yang diterima oleh *server SMS Gateway*. Data SMS tersebut selanjutnya diproses untuk mengupdate status pembayaran mahasiswa. Sementara *web service* digunakan untuk menampilkan daftar mahasiswa yang sudah membayar dan yang belum membayar spp. *Web service* tersebut dibangun dengan menggunakan PHP *Framework Codeigniter* dan data yang diolah berformat JSON.

Melalui makalah penelitiannya, Sutanta (2012) mengungkapkan kondisi sistem informasi yang ada di Pemkab Bantul DIY. Pemkab Bantul telah mengembangkan 33 aplikasi sistem informasi yang sebagian besar dapat diakses dari portal web Pemkab Bantul. Akan tetapi aplikasi-aplikasi tersebut belum terintegrasi. Sedangkan Sutanta melihat adanya kesamaan obyek data (entitas) dan kedekatan hubungan antar obyek data yang diolah dalam sistem informasi. Sehingga perlu dirancang layanan *web service* antar aplikasi e-Gov Pemkab Bantul dengan menggunakan model REST. Sedangkan Adi (2013) mengimplementasikan SOAP *web service* dalam bidang kesehatan yaitu untuk mengintegrasikan data farmasi di RSUD Banyumas. Data yang diintegrasikan adalah data dalam *server* Farmasi dan *server* SIMRS. Aplikasi *desktop* di *client* dapat memanfaatkan *method* yang ada dalam *web service* tersebut. *Output* dari *method* tersebut berupa ADO.NET Dataset. Penelitian serupa dan pada tahun yang sama juga dilakukan oleh Hidayat (2013) yang menerapkan teknologi *web service* untuk mengintegrasikan layanan puskesmas dan rumah sakit. Sistem dibangun berbasis *web*. Aplikasi untuk rumah sakit menggunakan ASP.NET sedangkan pada Asri Medical Center (AMC) menggunakan PHP. Sehingga *web service provider* pada rumah sakit dan AMC dibangun sesuai dengan bahasa pemrograman masing-masing. Hasil integrasi berupa sebuah sistem yang merupakan hasil gabungan data dari dua rumah sakit tersebut.

3. METODE PENELITIAN

Berikut ini adalah langkah-langkah yang dilakukan dalam penelitian ini:

3.1 Pengumpulan Data

Pengumpulan data yang digunakan dalam penelitian ini dilakukan dengan beberapa cara yaitu sebagai berikut:

1. Studi Pustaka

Studi pustaka berisi kegiatan mempelajari literatur-literatur yang berhubungan dengan topik penelitian yaitu mengenai integrasi data menggunakan *web service*. Literatur diperoleh dari berbagai sumber antara lain dari jurnal ilmiah, laporan hasil penelitian, majalah dan buku.

2. Wawancara

Wawancara dilakukan dengan cara tanya jawab kepada orang-orang yang memiliki kapasitas informasi dan tanggung jawab yang dibutuhkan dalam penelitian ini. Diantaranya adalah kepala Biro Sistem Informasi, ketua program studi, administrasi prodi, mahasiswa dan dosen pembimbing skripsi.

3. Pengambilan Data

Pengambilan data berupa berbagai formulir *hardcopy* maupun *softcopy* persyaratan dan hasil sidang proposal skripsi guna proses analisis, perancangan, implementasi dan pengujian aplikasi yang akan dibangun.

3.2 Analisis dan Perancangan

Tahap analisis dan perancangan meliputi beberapa tahap sebagai berikut:

3.2.1 Analisis Kebutuhan Sistem

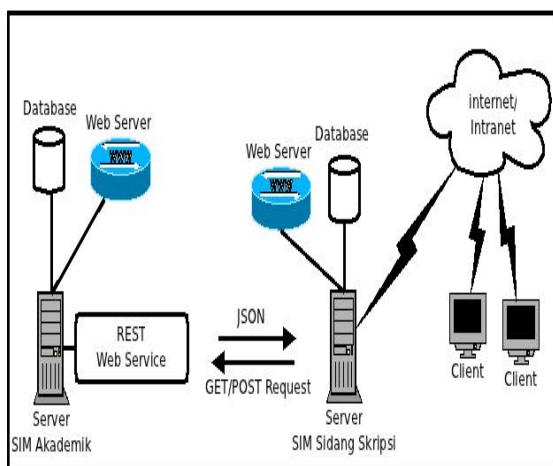
Kebutuhan sistem yang diperlukan dalam penelitian ini dapat dikelompokkan menjadi 3 yaitu kebutuhan data, *hardware* dan *software*.

Data yang dibutuhkan dalam penelitian ini adalah sebagai berikut:

- a) Data Mahasiswa
 - b) Data Program Studi dan Fakultas
- Kebutuhan *hardware* dan *software* adalah sebagai berikut:
- a) Komputer *Server*
Komputer *server* digunakan untuk menjalankan aplikasi SIM Sidang Proposal Skripsi. *Software* yang diperlukan dalam komputer *server* adalah *web server Apache*, *database* MySQL dan PHP.
 - b) *Web service*
Web service digunakan untuk integrasi data antara *server* aplikasi SIM Sidang Skripsi dan *server* SIM Akademik Universitas nusantara PGRI Kediri.
 - c) Komputer *Client*
Komputer *client* digunakan untuk mengakses aplikasi SIM Sidang Skripsi.
 - d) Jaringan Intranet/Internet
Jaringan intranet/internet digunakan untuk komunikasi data antara komputer *server* dan *client*.

3.2.2 Desain Arsitektur

Desain arsitektur integrasi data dapat digambarkan dalam Gambar 1.



Gambar 1. Desain arsitektur integrasi sistem

Pada Gambar 1 terdapat dua server yaitu Server Sistem Informasi Akademik (Siakad) dan Server Sistem Informasi Sidang Skripsi (Siski). Masing-masing server menggunakan web server apache dan database MySQL. Server Siakad menggunakan sistem operasi Linux Debian, sedangkan server Siski menggunakan sistem operasi Linux Ubuntu. Pada server siakad akan dibangun web service server untuk melayani permintaan data mahasiswa. Web service tersebut akan terhubung dengan database akademik. Web service dibangun dengan menggunakan bahasa pemrograman PHP. Output dari web service tersebut berupa string JSON (Javascript Object Notation). Sedangkan pada server Siski akan dibangun service client untuk melakukan permintaan data dan mengambil data output dari web service server yang akan digunakan lebih lanjut oleh aplikasi Siski guna pemrosesan data. Aplikasi Siski dan web service client dibangun menggunakan bahasa pemrograman PHP.

3.2.3 Desain Database

Dalam tahap ini akan dirancang tabel database untuk mengelola keamanan akses web service.

3.3 Development

Pada tahap implementasi akan dilakukan proses coding web service menggunakan PHP sebagai backend programming dan database MySQL. Proses coding terbagi menjadi dua bagian yaitu coding web service provider dan requestor. Web service provider bertujuan untuk menyediakan layanan data untuk diakses server SIM Sidang Skripsi. Web service menggunakan REST dibuat dengan menggunakan bahasa pemrograman PHP. Data Request berisi nomor induk mahasiswa, data respons berisi biodata mahasiswa. Data request dan respons web service memiliki format JSON.

3.4 Pengujian

Pengujian web service dilakukan dengan melakukan akses langsung ke alamat web service. Pengujian terbagi menjadi dua bagian yaitu pengujian

keamanan akses web service dan pengujian request data.

4. PEMBAHASAN

4.1 Spesifikasi Web service

Web service yang telah dibuat berfungsi untuk melayani permintaan data mahasiswa. Data Mahasiswa didapatkan melalui akses ke server ke database siakad.

Spesifikasi Web service:

1. Url: http://service.teguh.dosen.unpkediri.ac.id/service_mhs
2. Deskripsi: melakukan permintaan data mahasiswa
3. Method submit: POST
4. Request format: JSON
5. Response format: JSON
6. Param request: npm, username, password

4.2 Database

Database digunakan untuk mengelola keamanan akses web service. Data username dan password akses disimpan dalam tabel login. Tabel login terdapat dalam tabel akses seperti dalam Gambar 2. Sedangkan data mahasiswa disimpan dalam tabel mhs_baa seperti dalam Gambar 3.

#	Column	Type
1	npm	varchar(30)
2	nama	varchar(100)
3	jenis_kelamin	varchar(20)
4	alamat	varchar(255)
5	tahun	year(4)
6	jenjang	varchar(10)
7	fakultas	varchar(50)
8	prodi	varchar(50)

Gambar 2. Tabel mahasiswa

#	Column	Type
1	username	varchar(30)
2	passwd	varchar(255)
3	ket	varchar(255)

Gambar 3. Tabel login akses web service

4.3 Web service Development

4.3.1 Web service Provider

Web service provider dibangun menggunakan bahasa pemrograman php yang berfungsi untuk melayani permintaan data mahasiswa. Data request dikirimkan oleh client dalam format JSON.

Script web service provider terdapat dalam Gambar 4.

```

$json_str =
file_get_contents('php://input');
$json_obj = json_decode($json_str);
$npm = $json_obj->"npm";
$username = $json_obj->"username";
$password = $json_obj->"password";
$sql = "SELECT * FROM akses WHERE
username='$username' AND
password=PASSWORD('$password')";
if($db->jumrec($sql)>0){
    $sql = "SELECT * FROM mhs_baa WHERE npm
LIKE '$npm'";
    $data = $db->fetchdata($sql);
    $arr_output = array();
    if($db->jumrec($sql)==0){
        $arr_output['status'] = 0;
        $arr_output['keterangan'] = 'Data Not
Found !!!!';
    }
    else{
        $arr_output['status'] = 1;
        $arr_output['keterangan'] = 'Data
Found';
        $i = 0;
        foreach($data as $dat){
            $i++;
            $arr_output['data'][$i]['npm'] =
$dat['npm'];
            $arr_output['data'][$i]['nama'] =
$dat['nama'];

            $arr_output['data'][$i]['jenis_kelamin'] =
$dat['jenis_kelamin'];
            $arr_output['data'][$i]['alamat'] =
$dat['alamat'];
            $arr_output['data'][$i]['tahun'] =
$dat['tahun'];
            $arr_output['data'][$i]['jenjang']
= $dat['jenjang'];

            $arr_output['data'][$i]['fakultas'] =
$dat['fakultas'];
            $arr_output['data'][$i]['prodi'] =
$dat['prodi'];
        }
    }
}
else{
    $arr_output['status'] = 0;
    $arr_output['keterangan'] = 'Access
Denied !!!!';
}
header('Content-type: application/json');
$output = json_encode($arr_output);
echo $output;

```

Gambar 4. Script web service provider

Web service akan melakukan dua proses. Proses pertama yaitu pengecekan akses melalui *username* dan *password*. Pengecekan ini berfungsi untuk membatasi akses pengguna terhadap *web service*. Hanya pengguna yang memiliki *username* dan *password* yang dapat mengakses *web service*. Jika *username* dan *password* tidak sesuai maka akan menghasilkan *output* status 0 dan keterangan *Access Denied*. Jika sesuai maka akan dilanjutkan dengan proses kedua yaitu akses *database*. Jika data ditemukan dalam *database* maka akan menghasilkan *output* status 1, keterangan *Data Found* dan biodata mahasiswa. Jika data tidak ditemukan maka akan menghasilkan *output* status 0 dan keterangan *Data Not Found*.

4.3.2 Akses Web service

Akses *web service* dilakukan dengan mengirimkan request data kepada *web service*. Script akses *web service* terdapat dalam Gambar 5.

```

$npm = $_POST['npm'];
$data = array(
    'username' => 'tguh',
    'password' => 'tguh',
    'npm' => $npm
);
$data_string = json_encode($data);
$ch =
curl_init($service_url.'/mhs_info.php');
curl_setopt($ch, CURLOPT_CUSTOMREQUEST,
"POST");
curl_setopt($ch, CURLOPT_POSTFIELDS,
$data_string);
curl_setopt($ch, CURLOPT_RETURNTRANSFER,
true);
curl_setopt($ch, CURLOPT_HTTPHEADER, array(
'Content-Type: application/json',
'Content-Length: ' . strlen($data_string)
));
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER,
false);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST,
false);
$result = curl_exec($ch);
$http_status = curl_getinfo($ch,
CURLINFO_HTTP_CODE);
$curl_error = curl_error($ch);
curl_close($ch);
$data_res = json_decode($result,true);
if($data_res['status']=='1'){
    foreach($data_res['data'] as $dat)
        echo
$data_res['status'].'||'.$data_res['keterangan']
.'||'.$dat['npm'].'||'.

    $dat['nama'].'||'.$dat['alamat'].'||'.$dat['
prodi'];
    }
    else{
        echo
$data_res['status'].'||'.$data_res['keterangan']
;
    }
}

```

Gambar 5. Script akses web service

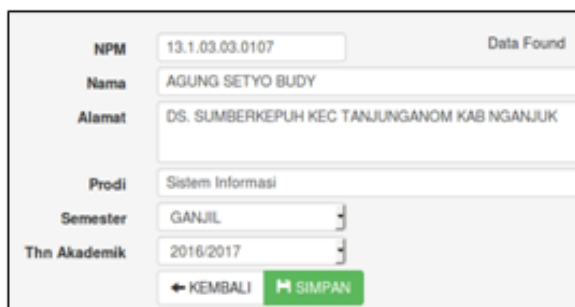
Akses *web service request* digunakan oleh aplikasi sistem informasi manajemen ujian skripsi untuk melakukan pencarian dan sinkronisasi data mahasiswa. Data yang dikirimkan memiliki format JSON, terdiri dari parameter *username*, *password* dan *npm*. Proses request dan pengambilan data dilakukan menggunakan perintah CURL. Setelah data berhasil diambil selanjutnya dilakukan proses pengecekan status hasil pengambilan data tersebut. Jika status bernilai 0 berarti bahwa data tidak ditemukan sehingga akan muncul *output* "0||Data Not found" atau "0||Access Denied". Jika *output* pertama muncul berarti bahwa data yang diminta tidak dapat ditemukan. Jika *output* kedua yang muncul berarti terdapat kesalahan *username* dan *password* saat mengakses *web service*. Sedangkan jika status bernilai 1 maka data yang diminta berhasil ditemukan, kemudian dilanjutkan dengan proses pengambilan

data detil yang diminta yaitu data nama, npm, alamat dan prodi.

4.4 Pengujian

4.4.1 Pengujian Keamanan Akses

Pengujian keamanan dilakukan dengan dua cara yaitu dengan mengirimkan *username* dan *password* yang sesuai dan tidak sesuai pada script dalam Gambar 5. Pengiriman *username* dan *password* yang sesuai menghasilkan tampilan dalam Gambar 6. Sedangkan pengiriman *password* yang tidak sesuai menghasilkan tampilan dalam Gambar 7.



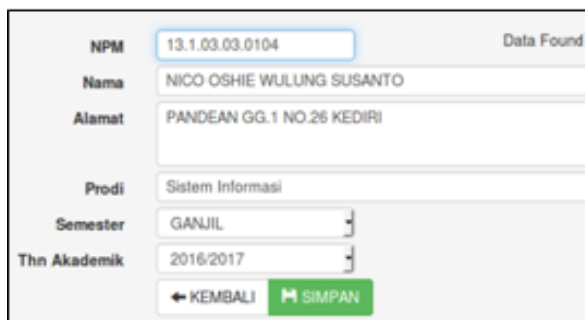
Gambar 6. Akses *web service* diterima



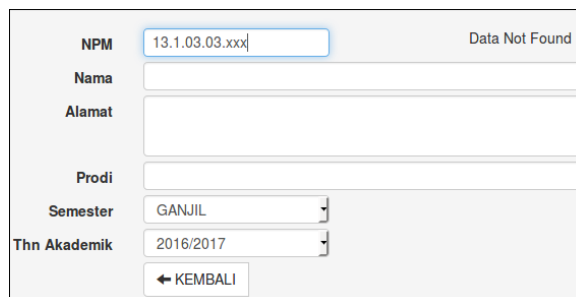
Gambar 7. Akses *web service* ditolak

4.4.2 Pengujian Request Data

Pengujian request data dilakukan dengan mengirimkan data npm yang sesuai yaitu 13.1.03.03.0107 dan data npm yang tidak sesuai yaitu 13.1.03.03.xxx. Pengiriman data npm yang sesuai menghasilkan tampilan dalam Gambar 8. Sedangkan pengiriman data yang tidak sesuai menghasilkan tampilan dalam Gambar 9.



Gambar 8 Permintaan data ditemukan



Gambar 9 Permintaan data tidak ditemukan

5. KESIMPULAN

Penelitian ini telah berhasil membangun sistem integrasi data antar sistem informasi yaitu sistem informasi ujian skripsi dan sistem informasi akademik di Universitas Nusantara PGRI Kediri. Data yang diintegrasikan adalah data mahasiswa yang bersumber dari sistem informasi akademik. Integrasi data dibangun menggunakan RESTful *web service*, bahasa pemrograman php dan *database* mysql. Akses *web service* dilakukan dengan mengirimkan data dalam format JSON yaitu data npm, *username* dan *password*. Data respon *web service* memiliki format JSON. Untuk meningkatkan keamanan data *web service* maka pada penelitian selanjutnya dapat dikembangkan penggunaan sistem enkripsi data untuk mengirimkan data request dan data respon.

DAFTAR PUSTAKA

- Fielding R.T., 2000, *Architectural Style & Design of Network-Based Software Architectures*, Ph.D. Thesis, Department of Information & Computer Science, UCLA, Irvine.
- Hidayat, R., Ashari, A., 2013, Penerapan Teknologi *Web service* Untuk Integrasi layanan Puskesmas dan Rumah Sakit, *Berkala MIPA*, (23)1, Januari 2013
- Kumari, V., 2015, *Web services Protocol: SOAP vs REST*, *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 4 Issue 5*
- Pautasso, C., 2008, *REST vs SOAP Making the Right Architectural Decision*, *SOA Symposium, Amsterdam*.
- Santosa, B., Prasetyo, D.B., Pungki, Y., 2011, Integrasi Toko Online Menggunakan Teknologi *Web service*, *TELEMATIKA Vol. 8, No. 1, JULI 2011 : 33 – 42*
- Susanto, A., Rahmayanti, Y., 2014, Rancang Bangun Prototype REST *Web service* Untuk Menghubungkan Sistem Pembayaran UDINUS Dengan Sistem Perbankan, *Tugas Akhir*
- Sutanta, E., Mustofa, K., 2012, Kebutuhan *Web services* Untuk Sinkronisasi Data Antar Sistem Informasi Dalam E-Gov di Pemkab Bantul Yogyakarta, *JURTIK - STMIK BANDUNG (edisi Mei 2012)*

Biodata Penulis

Teguh Andriyanto, S.T., menyelesaikan Sarjana Teknik pada jurusan Teknik Elektro Institut Teknologi Sepuluh November pada tahun 2004. Memperoleh gelar Magister of Computer Science dari Universitas Gadjah Mada pada tahun 2015. Saat ini aktif sebagai dosen di Prodi Sistem Informasi Universitas Nusantara PGRI Kediri.

Rini Indriati, M.Kom, memperoleh gelar Sarjana Komputer dari STMIK STIKOM Surabaya pada tahun 2005. Menyelesaikan Magister Teknologi Informasi di STTS pada tahun 2012. Saat ini menjadi dosen di Prodi Sistem Informasi Universitas Nusantara PGRI Kediri.