

Bridging Data Sistem Informasi Rumah Sakit (SIMRS) Dan Laboratory Information System (LIS)

Dhea Kalingga Lintang
Program Studi Teknologi Informasi
Universitas Muhammadiyah Purworejo
Purworejo, Indonesia
dheakalinggalintang@gmail.com

Krisna Widatama
Program Studi Teknologi Informasi
Universitas Muhammadiyah Purworejo
Purworejo, Indonesia
krisnawidatama@umpwr.ac.id

Ike Yunia Pasa
Program Studi Teknologi Informasi
Universitas Muhammadiyah Purworejo
Purworejo, Indonesia
ikeypasa@umpwr.ac.id

Abstract - The Laboratory Information system (LIS) is an integrated system between Hospital Laboratory Unit and LIS server. From LIS server, it send the results of examination to Hospital Laboratory System. Nowadays, LIS and the Hospital Management Information System (SIMRS) are not connected yet. This is because the code from the LIS is not in accordance with SIMRS and the mechanism for sending and withdrawing data between SIMRS and LIS has not been developed.

As a result, officers work inefficiently because they do manual records which can result in treating errors in affixing results. Errors in recording results can be a fatal result, it is because the result of examination is related with other units. It will malpractice and errors in recording payment of inspection bills.

The method used in this research is mapping the data between the audit data in the LIS database and the SIMRS database. This is because the collation code in the two databases is different from each other. In addition, the data transmission method used JSON (JavaScript Object Notation) data. JSON is a programming language that allows each server to exchange data with each other.

This paper expected laboratory officers can work efficient to recording the results of laboratory examinations. In addition, the security of patient data who perform examinations is guaranteed by the system.

Keywords: bridging, integration, SIMRS, JSON.

I. PENDAHULUAN

Perkembangan teknologi informasi saat ini membantu dan mempermudah pengguna di seluruh bidang, salah satunya adanya Sistem Informasi Manajemen Rumah Sakit di bidang kesehatan.[1] Sistem Informasi Manajemen Rumah Sakit (SIMRS) merupakan alat bagi pihak manajemen rumah sakit dalam bersaing bidang kompetensi yang semakin ketat, terutama di bidang jasa pelayanan kesehatan [2]. Informasi yang tersedia di dalam SIMRS menjadi gambaran dan tolak ukur terkait kegiatan operasional rumah sakit, sehingga dari data yang didapatkan dapat dilakukan perencanaan, pengendalian dan pengembangan operasional maupun kebijakan pimpinan. [3]

Rumah Sakit bekerja sama dengan Sysmex, sebuah vendor dari peralatan dan juga sistem informasi kesehatan, khususnya di unit laboratorium dan menanam Laboratory Information System (LIS) di server rumah sakit tersebut. [4] Sistem ini dapat menangkap hasil pemeriksaan laboratorium dari alat, kemudian dikirim melalui server dan ditampilkan di komputer petugas. Masalah yang terjadi adalah pemeriksaan yang diorder dari SIMRS tidak dapat dikirim ke LIS dan hasil yang diterima tidak dapat dikirim ke SIMRS, sehingga petugas laboratorium harus memasukan data tersebut ke masing-masing sistem secara manual. Maka dari itu perlu dilakukan bridging system antara SIMRS dan LIS.

Bridging System adalah sebuah servis yang dapat menghubungkan aplikasi berbasis *web* pada sistem pelayanan kesehatan menjadi satu agar mampu meningkatkan pelayanan yang optimal dan efisien di rumah sakit. [5] [6] Petugas di Unit Laboratorium Rumah Sakit cukup mengisikan data dari salah satu dari dua sistem tersebut, kemudian data yang

diisikan dapat dikirim dan diterima antara dua sistem tersebut. Sehingga, kesalahan dalam memasukkan hasil dapat ditekan seminimal mungkin.

Metode yang digunakan dalam penelitian ini menggunakan format *JavaScript Object Notation* (JSON) dalam pengiriman dan penerimaan data. *Javascript Object Notation* (JSON) adalah format pertukaran data yang memiliki ukuran data yang lebih kecil serta waktu proses yang lebih cepat dibandingkan format lain. [7] [8]. Selain itu, alasan dari penggunaan JSON karena JSON sangat populer dikalangan peneliti basis data karena terstruktur dan ringan [9], [10][11]. Selain JSON, metode yang digunakan yakni *mapping*, sebuah cara untuk memetakan kode jenis pemeriksaan antara SIMRS dan LIS yang berbeda satu sama lain. Sehingga kedua sistem tersebut dapat bertukar data.

Harapan yang ingin dicapai yakni LIS dapat terhubung dengan SIMRS dan begitu juga sebaliknya, sehingga memberikan data yang dimasukan akurat, faktual, dan aman.

II. METODE PENELITIAN

2.1. Metode Pengembangan Sistem

Metode pengembangan sistem yang dilakukan dalam penelitian ini yakni menggunakan *waterfall*. *Waterfall* adalah strategi pengembangan perangkat lunak pertama yang diterapkan, menyerupai desain yang digunakan di industri lain. [12] Strategi ini memungkinkan proyek untuk dipecah menjadi beberapa fase tetap, dengan setiap fase membutuhkan analisis dan pekerjaan dari fase sebelumnya Adapun tahapan-tahapan dalam pengembangan sistem dalam metode *Waterfall* adalah sebagai berikut

a. Identifikasi

Peneliti mengidentifikasi masalah-masalah yang ada pada Unit Laboratorium Rumah Sakit . Salah satu masalah yang ditemukan yakni antara Sistem Informasi Manajemen Rumah Sakit (SIMRS) dan Laborat Information System (LIS) belum terintegrasi.

b. Analisis dan pengumpulan data

Pengumpulan data dan informasi di lakukan dengan cara interview kepada narasumber yaitu pihak Laboratorium dan mendapatkan data yang berupa *database* Laboratorium Rumah Sakit . Selain itu, interview juga dilakukan dengan pihak Sysmex, yakni perusahaan yang menjadi vendor dari Rumah Sakit . Interview ini menghasilkan bagaimana cara kerja dari LIS dan *database* nya.

c. Perancangan sistem

Perancangan sistem ini merupakan tahap pengembangan setelah analisis sistem dilakukan, beberapa proses yang dilakukan dalam perancangan sistem yaitu: *Use case diagram*, *Activity diagram*, struktur tabel dan desain mockup untuk dibuat sistem selanjutnya.

d. Pengkodean dan Implementasi sistem

Sebelum sistem dijalankan maka harus membuat sistem informasi rumah sakit dengan menggunakan perintah bahasa pemrograman, pada tahap ini penulis menggunakan bahasa pemrograman PHP, sehingga menghasilkan sebuah Sistem Informasi Manajemen Rumah Sakit berbasis *web* yang telah terintegrasi dengan LIS.

e. Pengujian sistem

Setelah tahap pengkodean dan implelementasi selesai maka akan dilakukan pengujian terhadap sistem mulai dari fungsi *input* maupun *output*, guna untuk menghindari terjadi *error* dan hasilnya harus sesuai dengan yang sudah di rancang. Pengujian ini dilakukan oleh Unit Laboratorium Rumah Sakit dan dapat dioperasikan hingga hari ini.

f. Deployment

Deployment adalah kegiatan yang bertujuan untuk menyebarkan aplikasi yang telah dikerjakan oleh para pengembang. Pada tahap ini, SIMRS yang telah berjalan diberikan hak akses nya kepada petugas Unit Laboratorium.

g. Pemeliharaan

Tahap pemeliharaan sistem mencakup seluruh proses yang diperlukan untuk menjamin kelangsungan, kelancaran, dan penyempurnaan sistem yang telah dioperasikan. Pemeliharaan juga mencakup perbaikan *bug* dan *error* yang ada pada sistem.

2.2. Metode Pengumpulan Data.

Metode ini digunakan untuk mengumpulkan data pasien, pemeriksaan dan basis data terkait SIMRS dan LIS. Data yang ada dalam metode ini terdiri dari data fisik dan data digital [13].

a. Observasi

Observasi adalah Observasi adalah pengamatan langsung dengan memakai format tertentu sesuai dengan kebutuhan peneliti [14]. Pada tahap ini, peneliti melakukan observasi langsung di Unit Laboratorium Rumah Sakit .

b. Wawancara

Wawancara adalah percakapan dengan maksud tertentu dengan melibatkan setidaknya dua orang, satu sebagai pewawancara dan satu orang sebagai narasumber [15]. Percakapan itu dilakukan oleh dua pihak, yaitu pewawancara yang mengajukan pertanyaan dan narasumber memberikan jawaban atas pertanyaan tersebut. Penulis melakukan wawancara terhadap pihak-pihak yang terkait dengan penelitian ini. Pada kasus ini, penulis mewawancarai Kepala Ruang Unit Laboratorium Rumah Sakit dan salah satu teknisi dari LIS selaku narasumber.

Pada wawancara tersebut diambil kesimpulan bahwa seharusnya, untuk mempermudah kinerja dari petugas laboratorium serta mempergunakan mesin dari LIS dengan optimal, maka *bridging* data harus segera dilakukan. Jika tidak, maka petugas harus menuliskan hasil sebanyak 2 kali, yakni penulisan hasil di LIS dan hasil di SIMRS.

c. Studi Pustaka

Studi kepustakaan adalah teknik pengumpulan data dengan penelaahan terhadap buku-buku, literatur-literatur, catatan-catatan, dan laporan-laporan yang ada hubungannya dengan masalah yang dipecahkan. Penulis mencari sumber-sumber untuk memperkuat penelitian ini melalui buku, jurnal dan sumber lainnya yang berhubungan dengan masalah yang diteliti.

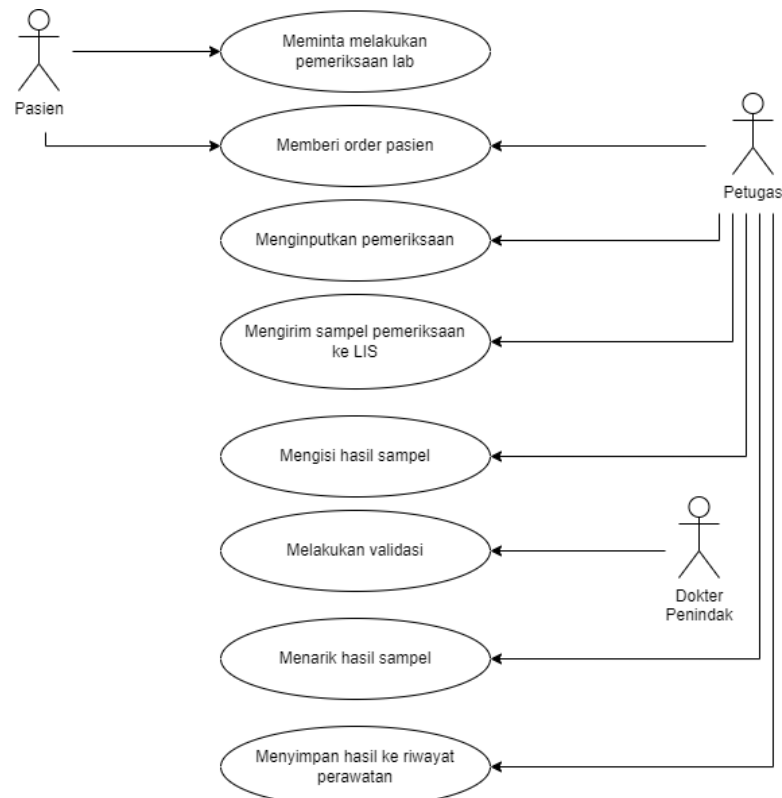
d. Kuisioner

Kuisioner diberikan kepada *person in charge* (PIC) unit laboratorium Rumah Sakit . Kuisioner ini menjawab beberapa pertanyaan mengenai sistem dan fungsionalitas di tingkat petugas.

III. HASIL DAN ANALISIS

3.1. Usecase Diagram

Use case diagram untuk menggambarkan interaksi yang terjadi antara sistem dengan pengguna [16]. Berdasarkan sistem yang akan dibuat, terdapat beberapa pengguna yang akan berinteraksi terhadap sistem. Pengguna yang memiliki akses terhadap sistem yakni petugas dan dokter penindak. Berikut merupakan penjelasan pada gambar 3.1. sebagai berikut :



Gambar 3.1 Usecase diagram

Pada proses ini pasien meminta untuk melakukan pemeriksaan ke laboratorium, selanjutnya petugas memberi nomer order pasien sekaligus menginputkan pemeriksaan, setelah petugas melakukan penginputan pemeriksaan lalu sampel pemeriksaan tersebut dikirim ke LIS Sysmex, yang kemudian sampel pemeriksaan itu diolah oleh petugas, selanjutnya hasil sampel tersebut di validasi oleh dokter penindak. Kembali ke data permintaan laboratorium, petugas memilih pasien yang sudah divalidasi, kemudian petugas dapat menarik data yang telah disimpan di LIS, lalu disimpan di simrs. Di Simrs, hasil tarikan data nantinya tertera di bagian riwayat perawatan pasien yang berisikan pemeriksaan, hasil yang didapat, satuan, petugas pekaksana, dokter penindak dan harga pemeriksaan.

3.2. Implementasi Sistem

Pasien yang berada di unit laboratorium merupakan pasien dari rawat jalan dan rawat inap. Dua status tersebut memiliki perbedaan dalam segi administrasi maupun dari sisi sistem. Pada penelitian ini, data sampel yang diberikan yakni pasien dengan status rawat inap dengan pemeriksaan yang dituju yakni Unit Laboratorium. Pasien yang telah diberikan status ranap dan melakukan permintaan pemeriksaan laboratorium akan muncul di menu laboratorium. Berikut merupakan tampilan dari menu Laboratorium sebagai awal mula *bridging* sistem dengan LIS :

Permintaan Laboratorium

Home / Laboratorium / Permintaan Laboratorium

Data Permintaan Laboratorium

Rawat Jalan Rawat Inap

Data Permintaan Item Permintaan

Action	No. Permintaan	No. Rawat	Pasien	Permintaan	Jam	Sampel	Hasil	Jam	Dokter Perujuk	Poli Registrasi	Informasi Tambahan	Diagnosis Klinis	Jenis Bayar
	RJ201		Catuk Permintaan Lab Rutap	2022-07-10	18:32:46	103101	103101	103101	dr. Katherine Goveary Sugarto	IGD	-		UMUM
	RJ201		Kirim Permintaan ke LIS Sysmex	2022-07-10	13:58:12	104105	104105	104105	dr. Meda Mitasari	IGD	-		BPJS
	RJ201		Ambil Hasil ke LIS Sysmex	2022-07-10	12:10:55	102107	102107	102107	dr. Meda Mitasari	IGD	-		UMUM
	RJ201	022/07/10/00004	RAF SAF	2022-07-10	11:29:19	103105	103105	103105	dr. Meda Mitasari	IGD	-		BPJS
	RJ201	022/07/10/00003	FATI NAA	2022-07-10	10:33:56	103105	103105	103105	dr. Meda Mitasari	IGD	-		BPJS
	RJ201	022/07/10/00003	PING NIO	2022-07-10	09:43:46	004101	004101	004101	dr. Meda Mitasari	IGD	-		BPJS
	RJ202207100001	022/07/10/00001		2022-07-10	08:34:44	003102	003102	003102	dr. Meda Mitasari	IGD	-		BPJS

Gambar 3.2 Pemeriksaan laboratorium

Gambar di atas menunjukkan cara pasien untuk melakukan tindakan pemeriksaan laboratorium. klik kanan pada pasien yang ingin dilakukan tindakan tersebut, kemudian pilih kirim permintaan ke LIS Sysmex. Pada tombol tersebut, terdapat *script* JSON yang dimaksud. Data pemeriksaan pasien yang telah dipilih petugas melalui SIMRS, nantinya dapat dikirim menggunakan JSON ke LIS. Pada komputer LIS, data-data ini akan diolah sedemikian rupa sehingga akan memunculkan nilai hasil. Berikut potongan *script* yang dimaksud:

```

148 $postdata = http_build_query(
149     array(
150         "message_dt" => $tgl . $jam,
151         "order_control" => "NW",
152         "pid" => $data_pasien->no_rkm_medis,
153         "pname" => $data_pasien->nm_pasien,
154         "address1" => $data_pasien->alamat,
155         "address2" => $data_pasien->nm_kel,
156         "address3" => $data_pasien->nm_kec,
157         "address4" => $data_pasien->nm_kab,
158         "ptype" => "IN",
159         "birth_dt" => $data_pasien->tgl_lahir,
160         "sex" => $jk,
161         "ono" => $_GET['noorder'],
162         "request_dt" => $data_detail->tgl_permintaan . " " . $data_detail->jam_permintaan,
163         "source" => $source,
164         "clinician" => $data_detail->dokter_perujuk . "^" . $data_detail->nm_dokter,
165         "room_no" => $room,
166         "priority" => "R",
167         "comment" => $data_detail->diagnosa_klinis,
168         "visitno" => $data_pasien->no_rawat,
169         "order_testid" => $abc
170     )
171 );
172 $opts = array(
173     'http' =>
174     array(
175         'method' => 'POST',
176         'header' => 'Content-Type: application/x-www-form-urlencoded',
177         'content' => $postdata
178     )
179 );
180 $context = stream_context_create($opts);
181 $result = file_get_contents('http://172.17.0.1/bungkarno/insert.php', false, $context);
182 echo $result;
183 ?>

```

Gambar 3.3 Potongan script JSON

JSON tersebut akan mengirimkan data dalam *array* ke dalam *file* tujuan. *File* tujuan tersebut dapat dilihat pada baris ke-181. Kata yang disensor merupakan alamat IP *local* dari Rumah Sakit. Kemudian, *bungkarno* pada kalimat berikutnya merupakan *folder* tempat penyimpanan *file* tersebut. Sedangkan *insert.php* merupakan *file service* yang ditanam untuk menerima data *array* dari SIMRS yang telah dibuat. Berikut merupakan isi dari *file insert.php* :

```

<?php
$josndata = file_get_contents('php://input');
$data = json_decode($josndata, true);
$message_dt = $_POST['message_dt'];
$order_control = $_POST['order_control'];
$pid = $_POST['pid'];
$pname = $_POST['pname'];
$address1 = $_POST['address1'];
$address2 = $_POST['address2'];
$address3 = $_POST['address3'];
$address4 = $_POST['address4'];
$ptype = $_POST['ptype'];
$birth_dt = $_POST['birth_dt'];
$sex = $_POST['sex'];
$sono = $_POST['ono'];
$request_dt = $_POST['request_dt'];
$source = $_POST['source'];
$clinician = $_POST['clinician'];
$room_no = $_POST['room_no'];
$priority = $_POST['priority'];
$comment = $_POST['comment'];
$visitno = $_POST['visitno'];
$order_testid = $_POST['order_testid'];
$conn = new mysqli(
);

$sql = "INSERT INTO lis_order
(message_dt,order_control,pid,pname,address1,address2,address3,ad
dress4,ptype,birth_dt,sex,ono,request_dt,source,clinician,room_no,
priority,comment,visitno,order_testid,Flag)
VALUES ($message_dt','$order_control','$pid','$pname',
'$address1','$address2', '$address3','$address4', '$ptype','$birth_dt',
'$sex','$sono','$request_dt', '$source','$clinician',
'$room_no','$priority','$comment', '$visitno','$order_testid','0')";
$q = mysqli_multi_query($conn, $sql);
if ($q)
{
    $sql = "INSERT INTO logtrx (nomor,loge,jsone)
VALUES('0','Sukses','$josndata')";
    $q = mysqli_multi_query($conn, $sql);
}
else
{
    $sql = "INSERT INTO logtrx (nomor,loge,jsone)
VALUES('0','Gagal','$message_dt')";
    $q = mysqli_multi_query($conn, $sql);
}
?>

```

Potongan script di atas berfungsi untuk menangkap data yang berupa *array* dari SIMRS lalu menyimpannya di *database* LIS agar data pasien yang dimaksudkan dapat diolah di sistem LIS. *Array* dalam bentuk JSON tersebut harus di-*decode* lalu akan dikirimkan ke *database* yang dimaksud [17]. Beberapa data seperti nama

database, *username* dan kata sandi telah disensor. *Array* tersebut kemudian dikirimkan ke *database* LIS. Berikut penjelasan mengenai data tersebut :

Tabel 3.1 Struktur tabel LIS

No.	Object	Keterangan
	<i>Message_dt</i>	<i>Message_dt</i> kepanjangan dari <i>message date</i> yang berarti tanggal dan jam pemeriksaan pasien tersebut dikirimkan dari SIMRS ke LIS
	<i>Order_control</i>	<i>Order_control</i> disini menunjukkan permintaan pemeriksaan yang dikirimkan. NW adalah kepanjangan dari <i>new</i> atau baru
	<i>PID</i>	<i>PID</i> merupakan singkatan dari <i>patient ID</i> yang berarti nomor rekam medis. Setiap pasien memiliki nomor rekam medis sendiri di Rumah Sakit . Data ini berasal dari tabel pasien yang berelasi dengan <i>reg_periksa</i> dan <i>permintaan_lab</i> di <i>database</i> SIMRS
	<i>PNAME</i>	<i>PNAME</i> merupakan singkatan dari <i>patient name</i> yang berarti nama pasien yang melakukan pemeriksaan tersebut. Data ini berasal dari tabel pasien yang berelasi dengan <i>reg_periksa</i> dan <i>permintaan_lab</i> di <i>database</i> SIMRS
	<i>Address1</i>	<i>Address1</i> merupakan alamat dari pasien yang melakukan pemeriksaan. Data ini berasal dari tabel pasien yang berelasi dengan <i>reg_periksa</i> dan <i>permintaan_lab</i> di <i>database</i> SIMRS
	<i>Address2</i>	<i>Address2</i> merupakan nama kelurahan dari pasien yang melakukan pemeriksaan. Data ini berasal dari tabel pasien yang berelasi dengan <i>reg_periksa</i> dan <i>permintaan_lab</i> di <i>database</i> SIMRS
	<i>Address3</i>	<i>Address3</i> merupakan nama kecamatan dari pasien yang melakukan pemeriksaan. Data ini berasal dari tabel pasien yang berelasi dengan <i>reg_periksa</i> dan <i>permintaan_lab</i> di <i>database</i> SIMRS
	<i>Address4</i>	<i>Address4</i> merupakan nama kabupaten dari pasien yang melakukan pemeriksaan. Data ini berasal dari tabel pasien yang berelasi dengan <i>reg_periksa</i> dan <i>permintaan_lab</i> di <i>database</i> SIMRS
	<i>Ptype</i>	<i>Ptype</i> merupakan kependekan dari <i>patient type</i> . Pada kasus ini, tipe yang dimaksudkan adalah masukkan dari SIMRS ke LIS yang berarti in (masuk)
	<i>Birth_dt</i>	<i>Birth_dt</i> merupakan kependekan dari <i>birth date</i> . <i>Birth date</i> adalah tanggal lahir pasien yang melakukan pemeriksaan. Tujuan dari dimasukkannya tanggal lahir pasien untuk mengecek umur dari pasien tersebut. Umur tersebut nantinya akan menjadi pembeda dari detail pemeriksaan yang dilakukan di LIS.

No.	Object	Keterangan
	<i>Sex</i>	<i>Sex</i> merupakan jenis kelamin dari pasien yang melakukan pemeriksaan. Seperti umur pasien, jenis kelamin juga menentukan detail pemeriksaan yang akan dilakukan. Pada sistem ini, jenis kelamin laki-laki ditandai dengan angka 1 (satu) sedangkan perempuan ditandai dengan angka 2 (dua)
	<i>Ono</i>	<i>Ono</i> merupakan nomor order (permintaan) dari pasien yang akan melakukan pemeriksaan. Contoh nomor order yakni RJ2022030100040.
	<i>Request_dt</i>	<i>Request dt</i> merupakan kependekan dari <i>request date</i> yang berarti tanggal permintaan. Berbeda dengan <i>message_dt</i> yang berarti tanggal pengiriman pemeriksaan dari SIMRS ke LIS, <i>request_dt</i> merupakan tanggal dan jam pasien melakukan registrasi permintaan pemeriksaan di bagian laboratorium. Data ini tersimpan di tabel <i>permintaan_lab</i> .
	<i>Source</i>	<i>Source</i> merupakan tempat poliklinik pasien mendaftarkan diri. Rumah Sakit memiliki 19 poliklinik. Pada data ini, sistem mengirimkan kode poli dan nama poli yang dimaksud. Nama dan kode poli ini tersimpan di kolom poliklinik di SIMRS.
	<i>Clinican</i>	<i>Clinican</i> merupakan dokter perujuk dalam pemeriksaan. Dokter perujuk diambil dari tabel dokter yang berelasi dengan tabel <i>reg_periksa</i> dan <i>permintaan_lab</i> pada SIMRS. Pada <i>clinican</i> tersebut, data yang dikirim merupakan kode dokter dan nama dokter
	<i>Room_no</i>	<i>Room_no</i> berupa data kode poli dari poliklinik tempat pasien melakukan pemeriksaan. Kode poli ini disimpan di tabel Poliklinik di basis data SIMRS.
	<i>Comment</i>	<i>Comment</i> merupakan informasi tambahan pasien ketika melakukan pemeriksaan di Unit Laboratorium Rumah Sakit . Informasi tambahan ini bisa berupa gejala pasien atau penyakit yang di derita. Di dalam basis data SIMRS, <i>comment</i> disimpan dalam kolom <i>informasi_tambahan</i>
	<i>VISITNO</i>	<i>Visitno</i> merupakan nomor rawat dari pasien yang melakukan pemeriksaan di Rumah Sakit .
	<i>Order_testid</i>	<i>Ordet_testid</i> merupakan kode jenis pemeriksaan dari setiap pemeriksaan yang dilakukan oleh pasien

3.2.1. Format Data dan Alur *Bridging*

Bridging (integrasi) pada SIMRS Unit Laboratorium dapat menjembatani antara dua sistem, yakni sistem SIMRS dan LIS. Kedua sistem ini dapat saling berkomunikasi dan bertukar data. Namun, ada beberapa data utama dengan

format yang harus diperhatikan dalam *bridging system* tersebut. Berikut merupakan penjelasannya :

a. ONO

Pada keterangan sebelumnya, ONO merupakan nomor order yang dimiliki pasien. Pasien akan menerima nomor order berbeda beda sesuai nomor urutan pasien di hari tersebut. Contoh ONO / nomor order yakni RJ202203010040. Berikut merupakan penjelasan format pada nomor tersebut :

1. RJ merupakan status dari pasien tersebut. Kepanjangan dari RJ yakni Rawat Jalan. Pada kasus pasien rawat inap, maka kode yang akan diberikan adalah RI
2. 2022 merupakan tahun dari pasien yang melakukan pemeriksaan.
3. 0301 merupakan bulan dan tanggal pemeriksaan. 03 berarti merupakan bulan Maret sedangkan 01 adalah tanggalnya.
4. 0040 merupakan urutan pasien yang melakukan pemeriksaan. Nomor urut ini berawal dari 0001. Setiap hari, nomor urut ini akan otomatis *reset* kembali ke 0001 lagi.

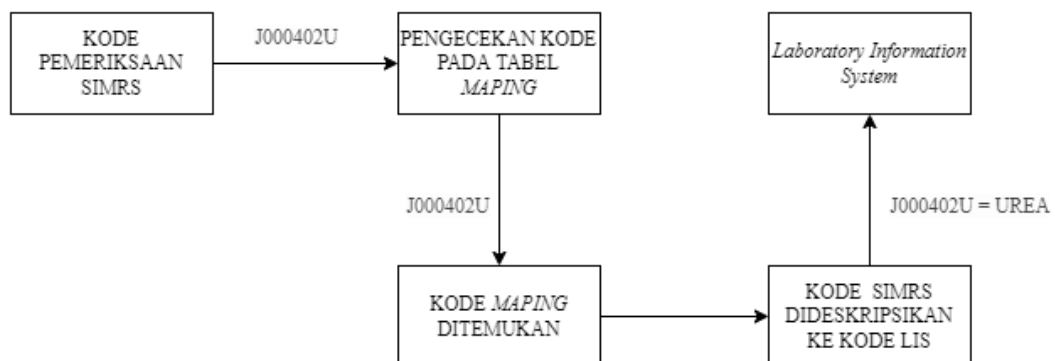
b. VISITNO

VISITNO merupakan nomor rawat pasien. Nomor rawat ini akan diberikan sistem kepada pasien saat pasien melakukan pendaftaran di rumah sakit. Setiap pasien akan memiliki VISITNO / nomor rawat yang berbeda. Contoh VISITNO / no rawat adalah 2022/03/01/0040. Berikut merupakan format dari VISITNO / nomor rawat yang dimaksud :

1. 2022 merupakan tahun ketika pasien diperiksa
2. 03 merupakan bulan pasien melakukan pemeriksaan.
3. 01 merupakan tanggal pemeriksaan pasien
4. 00040 merupakan nomor urut pasien mendaftarkan diri di Rumah Sakit . Nomor urut ini di-*reset* kembali ke 00001 setiap hari.

c. Order_testid

Pada penjelasan sebelumnya, *order_testid* merupakan kode jenis pemeriksaan laboratorium. Pada sistem ini, kode dari SIMRS dan LIS berbeda, oleh sebab itu perlu dilakukan *mapping* (pemetaan data). Pemetaan data ini bertujuan agar kode jenis pemeriksaan dari SIMRS dan LIS saling terintegrasi satu sama lain. Hasil *mapping* tersebut disimpan di dalam tabel *mapping* di database SIMRS maupun LIS. Berikut merupakan penjelasan mengenai *mapping* :



Gambar 3.4 Alur *mapping*

Gambar diatas menjelaskan bagaimana *mapping* data bekerja. Misalnya pasien bernama A periksa Ureum di Unit Laboratorium Rumah Sakit dengan kode jenis pemeriksaan nya J000402U. Ketika petugas mengirimkan pemeriksaan pasien dengan metode *bridging* ke LIS, sistem akan mengecek kode tersebut apakah telah di dipetakan atau belum. Jika sudah, kode dari SIMRS akan diubah menjadi kode LIS. Bermula dari J000402U yang merupakan kode jenis pemeriksaan di SIMRS, kemudian kode tersebut berubah menjadi UREA dan dikirimkan ke LIS. Berikut merupakan tabel *mapping* yang berada di basis data SIMRS maupun LIS :

Field	Type	Constraint
LIS	Varchar(8)	
HIS	Varchar(50)	
NAMA	Varchar(50)	

Tabel 3.2 Tabel *mapping*

Tabel di atas merupakan tabel *mapping* yang berada di SIMRS dan LIS. Tabel ini menjadi tabel yang menjadi pen jembatan antara 2 kode jenis pemeriksaan dari LIS dan SIMRS yang berbeda format satu sama lain. Berikut merupakan beberapa contoh hasil *mapping* dari LIS ke SIMRS maupun sebaliknya :

LDLD	J000400B	LDL Kolesterol
LDLD	J000401B	LDL Kolesterol
UREA	J000402U	Ureum
UREA	J000403U	Ureum

Gambar 3.5 Contoh data *mapping*

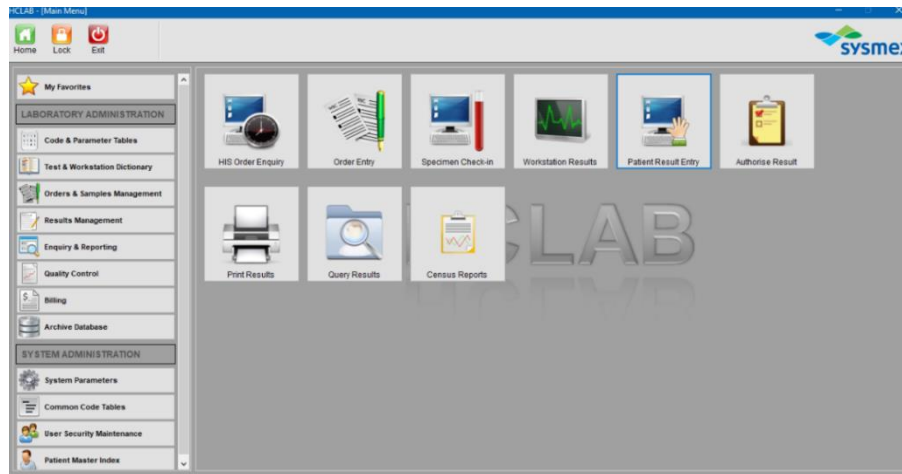
Gambar di atas merupakan contoh hasil *mapping* 4 pemeriksaan. LDL Kolesterol dengan kode J000400B dan J000401B merupakan pemeriksaan yang sama. Alasan kode pemeriksaan tersebut dibuat berbd a satu sama lain karena kebijakan dari rumah sakit. Isi dari kebijakan tersebut adalah setiap pemeriksaan setidaknya memiliki minimal 2 kode pemeriksaan yang berbeda. Hal ini menjadikan kode tersebut sebagai *backup* dari kode yang satunya.

Setelah kode-kode tersebut dapat tersedia di tabel kan, maka sistem dapat mengirim data pemeriksaan pasien tersebut ke LIS. Berikut merupakan hasil dari *var_dump* data dari proses pengiriman JSON ke LIS :

```
message_dt=20220802103533&order_control=NW&pid=015973&pname=FITRA
;1=CARIKAN+R
&address3=JUWIRING&address4=KAB.+KLATEN&ptype=IN&birth_dt=1991-08-
12&sex=1&ono=RJ202208020036&request_dt=2022-08-
02+10%3A27%3A09&source=RJ%5ERawat+Jalan&clinician=1
2100
2%5Edr.+Agus+Legowo&room_no=RJ&priority=R&comment=MCU&visitno=2022
%2F08%2
&order_testid=J032603
```

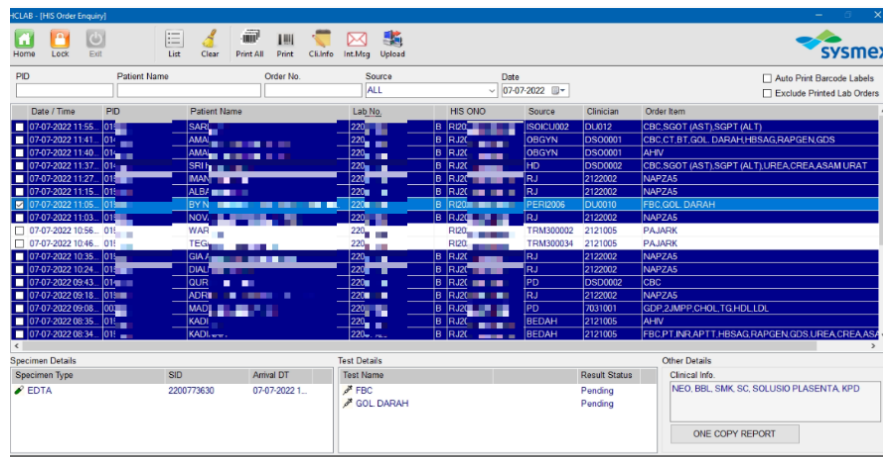
Gambar 3.6 Contoh format JSON saat kirim data

Gambar di atas merupakan hasil *var_dump* dari beberapa data yang dikirimkan dari SIMRS ke LIS. Data tersebut menyangkut identitas pasien dan pemeriksaan laboratorium yang akan dijalani.



Gambar 3.7 Sistem LIS

Gambar diatas menunjukkan menu yang terdapat didalam LIS Syntax. Petugas memilih menu *Patient Result Entry* untuk menunjukkan data pasien yang masuk ke pemeriksaan laboratorium baik untuk pasien baru maupun lama yang digunakan untuk tindak lanjut dari pemeriksaan pasien.



Gambar 3.8 Tampilan orderan pasien laboratorium

Gambar diatas menunjukkan data pasien yang melakukan tindakan pemeriksaan laboratorium. warna biru menandakan jika pasien sudah dilakukan pemeriksaan lab, warna putih menandakan belum dilakukan tindakan pemeriksaan, dan warna kuning belum dilakukan validasi. Pilih pada pasien yang belum dilakukan pemeriksaan, maka akan masuk ke menu selanjutnya.

Test Name	Cur Result	Prev Result	Prev Date	AF	RP	R1	R2	R3	Comment	Unit	Reference Range
Darah Lengkap											
Hemoglobin	H 17.8			N		17.8				g/dL	13.5 - 17.5
Leukosit	H			N		27.78				10 ³ /mm ³	4.4 - 11.3
Eritrosit				N		4.59				juta/mm ³	4.1 - 5.1
Trombosit				N		361				10 ³ /µL	150 - 450
Hitung Jenis Leukosit				N		47				%	35 - 45
Basofil				N		-				%	RNF
Eosinofil				N		0.7				%	1 - 5
Limfosit				N		12.6				%	25 - 40
Monosit				N		9.0				%	1 - 11
Neutrofil Segmen				N		50				%	50 - 70
MCV				N		103.1				fL	80.0 - 106.0
MCH				N		38.8				pg	28 - 33
MCHC				N		37.6				%	33 - 36
Rasio NfL				N							< 3.13
LED 1 JAM				N						mm/jam	< 15
Golongan Darah + Rhesus											
Golongan Darah	B										
Rhesus	Positif			N							

Gambar 3.9 Detail hasil pasien

Gambar diatas menunjukkan data detail pasien dan keluhan yang dimiliki pasien tersebut. pada kategori terdapat pilihan IMP (belum diperiksa) dan All (semua pasien). Pada pasien tersebut ditunjukkan secara detail kesehatan pasien. Setelah itu, pilih save dan release. Setelah itu, pada gambar 32. kembali ke menu awal dari LIS Sysmex, kemudian petugas pilih ke menu Authorise Sysmex

Test Name	Cur Result	Prev Result	Prev Date	AF	Unit	Reference Range	Comment
Darah Lengkap							
Hemoglobin	H 17.8	8.5	05-07-2022	A	g/dL	13.5 - 17.5	
Leukosit	H	24.72	05-07-2022	A	10 ³ /mm ³	4.4 - 11.3	
Eritrosit		3.07	05-07-2022	A	juta/mm ³	4.5 - 5.9	
Trombosit		223	05-07-2022	A	10 ³ /µL	150 - 450	
Hitung Jenis Leukosit		24	05-07-2022	A	%	35 - 45	
Basofil			04-07-2022	A	mmol/L	135 - 147	
Eosinofil			04-07-2022	A	mmol/L	3.5 - 5.0	
Limfosit			04-07-2022	A	mmol/L	95 - 105	
Monosit			25-06-2022	A	mg/dL	60 - 140	
Neutrofil Segmen			04-07-2022	A	mg/dL	10 - 50	
MCV		9.7	04-07-2022	A	mg/dL	0.9 - 1.2	

Gambar 3.10 Pengisian hasil detail pasien

.Gambar diatas menunjukkan tampilan dari menu Authorise Syntax. Pilih pada pasien yang diperiksa, kemudian pilih Authorise. Setelah dipilih authorise maka menandakan bahwa pasien telah dilakukan *inputan* hasil dan selanjutnya hasil yang ada dapat diberikan kepada pasien. Namun, hasil tersebut belum terekap di menu riwayat perawatan pasien yang ada di SIMRS. Berikut merupakan langkah - langkah untuk menginputkan hasil dari LIS ke SIMRS menggunakan sistem *bridging* sistem memakai JSON.

Permintaan Laboratorium

Home / Laboratorium / Permintaan Laboratorium

Data Permintaan Laboratorium

Rawat Jalan Rawat Inap

Data Permintaan Item Permintaan

Search:

Action	No. Permintaan	No. Rawat	Pasien	Permintaan	Jam	Sampel	Jam	Hasil	Jam	Dokter Perujuk	Poli Registrasi	Informasi Tambahan	Diagnosis Klinis	Jenis Bayar
	RJ20	002	Cetak Permintaan Lab Rekap	2022-07-10	18:32:46	2022-07-10	18:32:46	18:32:46	18:32:46	dr. Katherine Gowyary Sugianto	IGD	-		UMUM
	RJ20	002	Kirim Permintaan ke LIS Sysmex	2022-07-10	13:58:12	2022-07-10	13:58:12	13:58:12	13:58:12	dr. Meda Mitasari	IGD	-		BPIS
	RJ20	002	Ambil Hasil ke LIS Sysmex	2022-07-10	12:10:55	2022-07-10	12:10:55	12:10:55	12:10:55	dr. Meda Mitasari	IGD	-		UMUM
	RJ20	022/07/10/00004	RAF SAR	2022-07-10	11:29:19	2022-07-10	11:29:19	11:29:19	11:29:19	dr. Meda Mitasari	IGD	-		BPIS
	RJ20	022/07/10/00002	FAT1 NAA	2022-07-10	10:33:56	2022-07-10	10:33:56	10:33:56	10:33:56	dr. Meda Mitasari	IGD	-		BPIS
	RJ20	022/07/10/00003		2022-07-10	09:43:46	2022-07-10	09:43:46	09:43:46	09:43:46	dr. Meda Mitasari	IGD	-		BPIS
	RJ20	022/07/10/00002		2022-07-10	08:34:44	2022-07-10	08:34:44	08:34:44	08:34:44	dr. Meda Mitasari	IGD	-		BPIS

Gambar 3.11 Data permintaan laboratorium

Gambar diatas akan menampilkan detail pasien yang sudah dilakukan pemeriksaan melalui laboraturium dengan memilih “ambil hasil ke LIS Sysmex”. Berikut tampilannya :

Input Data Periksa Laboratorium

Home / Periksa Laboratorium / Input Data Periksa Laboratorium

Input Data Periksa Laboratorium

No. Rawat: 2022/07/05/00099

Dokter RJ: dr. Galuh Candrasari, Sp. N

Kode Perujuk: DS

Nama Perujuk: dr. Galuh Candrasari, Sp. N

Tgl. registrasi: 2022-07-07

Jam. registrasi: 09:35:25

Indikasi Klinis: DM 2

Pemeriksaan

Search:

P	Kode Periksa	Nama Pemeriksaan	Detail	ID Detail	Hasil	Satuan	Nilai Rujukan	Data Type	Flag	Drop Seq
☒	GLUK	Glukosa Puasa	Glukosa Puasa	GLUK	142	mg/dL	70 - 110	Num	H	120003000000
☒	GPST	Glukosa 2 jam PP	Glukosa 2 jam PP	GPST	432	mg/dL	80 - 130	Num	H	120003000000

Simpan

Gambar 3.12 Hasil tarikan data dari LIS

Gambar diatas akan menunjukan data detail dari hasil pemeriksaan pasien setelah dilakukan tindakan pemeriksaan melalui laboratorium. Data data yang tercantum pada tabel dibawah merupakan data yang diambil dari database LIS Sysmex menggunakan *bridging*. Hasil yang disimpan pada LIS setelah melalui beberapa pemrosesan, lalu diauthorise oleh petugas, maka data tersebut dapat diambil oleh petugas melalui SIMRS sehingga petugas hanya tinggal memilih nama petugas yang menginputkan, lalu klik simpan. Secara otomatis, data yang ada akan tersimpan di menu riwayat perawatan pasien. Berikut penjelasan mengenai tarik data pada menu tersebut :

Get data adalah penerimaan data menggunakan template JSON. Data pemeriksaan pasien yang telah diproses oleh petugas di komputer LIS dapat diambil oleh SIMRS menggunakan JSON. Pada SIMRS, data-data ini akan disimpan di tabel `periksa_lab` dan `detail_periksa_lab` agar data

tersebut muncul dibagian riwayat perawatan pasien. Berikut merupakan template JSON yang dimaksud :

```

104 <?php
105 $post_fields = $_GET['noorder'];
106 $ch = curl_init("http://192.168.1.100/bungkarno/select.php?ONO=" . $post_fields);
107 curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
108 curl_setopt($ch, CURLOPT_HTTPGET, 1);
109 curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
110 $result = curl_exec($ch);
111 $bridge = json_decode($result, true);
112 ?>
113

```

Gambar 3. 13 Potongan script JSON

JSON tersebut akan mengambil data dalam *array* ke dalam file tujuan. File tujuan tersebut dapat dilihat pada baris ke-106. Kata yang disensor merupakan alamat IP *local* dari RUMAH SAKIT . Kemudian, kalimat selanjutnya merupakan folder diletakkannya file tersebut, sedangkan *select.php* merupakan file service yang telah di tanam di komputer LIS. Pada kalimat berikutnya, terdapat parameter yang diambil dari variabel *\$post_fields*. Parameter yang dimaksud berupa nomor order dari pasien yang melakukan pemeriksaan. Jadi, baris kode tersebut akan mencari data dari basis data LIS yang memiliki ONO sesuai tangkapan dari variabel tersebut.

IV. KESIMPULAN

Salah satu cara memetakan kode pemeriksaan antara SIMRS dan LIS yakni dengan cara membuat tabel *mapping* yang didalamnya memuat kode pemeriksaan dari SIMRS dan LIS Sehingga sebelum sistem berkomunikasi satu sama lain, keduanya harus melewati pengecekan di tabel *mapping* tersebut. *Bridging* sistem SIMRS dan LIS dengan cara memetakan data kode jenis pemeriksaan dari kedua sistem tersebut. Setelah itu, membuat *service* yang dibuat menggunakan *JSON*. *Service* ini yang nantinya menjadi penjembaran antara SIMRS dan LIS.

REFERENSI

- [1] W. Krisna, H. J. Muhammad, and P. Danti, "Penggunaan Digital Signature Untuk Absensi Pada Universitas Muhammadiyah," *J. Sist. Cerdas*, vol. 05, no. 01, pp. 36–45, 2022, [Online]. Available: <https://apic.id/jurnal/index.php/jsc/article/download/188/98>.
- [2] Y. Wahyudin, S. Suhada, T. Hidayatulloh, and D. A. Firmansyah, "Rancang Bangun Bridging System Aplikasi Simrs Dan Aplikasi Virtual Claim Di Rumah Sakit Islam Assyifa Sukabumi," *Swabumi*, vol. 7, no. 1, pp. 84–89, 2019, doi: 10.31294/swabumi.v7i1.5926.
- [3] R. C. Buwono, "Web Services Menggunakan Format JSON," *J. Teknol. Inf.*, vol. XIV, pp. 1–10, 2019.
- [4] M. F. Fahlevi and I. G. Anugrah, "Implementasi Integrasi Sistem Informasi Manajemen Rumah Sakit Dengan Sistem Informasi Laboratorium Di Rumah Sakit PKU Muhammadiyah Sekapuk," *Bina Insa. Ict J.*, vol. 8, no. 1, p. 33, 2021, doi: 10.51211/biict.v8i1.1512.
- [5] A. Kurniawan, D. Tamtomo, and B. Murti, "Evaluation of Community Health Center Management Information System (SIMPUS), Primary Care (P Care), and Bridging Data System in Sukoharjo District," *J. Heal. Policy Manag.*, vol. 02, no.

- 02, pp. 157–164, 2017, doi: 10.26911/thejhpm.2017.02.02.07.
- [6] A. Triawan and M. A. Prasetyo, “Penerapan Web Service (XML dan JSON) Untuk Meningkatkan Performance Pada Informasi Pembayaran Uang Kuliah,” *Teknois J. Ilm. Teknol. Inf. dan Sains*, vol. 8, no. 1, pp. 78–93, 2019, doi: 10.36350/jbs.v8i1.22.
- [7] A. B. Warsito, A. Ananda, and D. Triyanjaya, “Penerapan Data JSON Untuk Mendukung Pengembangan Aplikasi Pada Perguruan Tinggi Dengan Teknik Restfull Dan Web Service,” *Technomedia J.*, vol. 2, no. 1, pp. 26–36, 2017, doi: 10.33050/tmj.v2i1.313.
- [8] C. O. Truică, E. S. Apostol, J. Darmont, and T. B. Pedersen, “The Forgotten Document-Oriented Database Management Systems: An Overview and Benchmark of Native XML DODBMSes in Comparison with JSON DODBMSes,” *Big Data Res.*, vol. 25, 2021, doi: 10.1016/j.bdr.2021.100205.
- [9] P. Bourhis, J. L. Reutter, F. Suárez, and D. Vrgoč, “JSON: Data model, Query languages and Schema specification,” *Proc. ACM SIGACT-SIGMOD-SIGART Symp. Princ. Database Syst.*, vol. Part F127745, pp. 123–135, 2017, doi: 10.1145/3034786.3056120.
- [10] F. Pezoa, J. L. Reutter, F. Suarez, M. Ugarte, and D. Vrgoč, “Foundations of JSON schema,” *25th Int. World Wide Web Conf. WWW 2016*, pp. 263–273, 2016, doi: 10.1145/2872427.2883029.
- [11] T. Lv, P. Yan, and W. He, “Survey on JSON Data Modelling,” *J. Phys. Conf. Ser.*, vol. 1069, no. 1, 2018, doi: 10.1088/1742-6596/1069/1/012101.
- [12] B.-A. Andrei, A.-C. Casu-pop, S.-C. Gheorghe, and C.-A. Boiangiu, “a Study on Using Waterfall and Agile Methods in Software Project Management,” *J. Inf. Syst. Oper. Manag.*, vol. 13, no. 1, pp. 125–135, 2019.
- [13] K. Widatama, “Sistem Monitoring Bukti Digital Untuk Meningkatkan Kontrol Terhadap Kasus Cybercrime Di Indonesia,” *INTEK J. Inform. dan Teknol. Inf.*, vol. 2, no. 1, pp. 39–46, 2019, doi: 10.37729/intek.v2i1.88.
- [14] G. M. M.Si, “Peningkatan Hasil Belajar Menyusun Teks Laporan Hasil Observasi Pada Peserta Didik Kelas X Sman 7 Malang Dengan Model Pembelajaran Integratif,” *JINoP (Jurnal Inov. Pembelajaran)*, vol. 5, no. 1, p. 39, 2019, doi: 10.22219/jinop.v5i1.7244.
- [15] S. Hansen, “Investigasi Teknik Wawancara dalam Penelitian Kualitatif Manajemen Konstruksi,” *J. Tek. Sipil*, vol. 27, no. 3, p. 283, 2020, doi: 10.5614/jts.2020.27.3.10.
- [16] K. Widatama and B. Suranto, “Sistem Informasi Manajemen Rancangan Anggaran Dan Pendapatan Sekolah Pada Dinas Pendidikan Kota Magelang,” ... *Inform. dan Teknol. Inf.*, vol. 1, no. November, pp. 82–89, 2018, [Online]. Available: <http://ejournal.umpwr.ac.id/index.php/intek/article/view/5348>.
- [17] B. Destian Wijaya, F. E.M.A, and A. Fiade, “Implementasi JSON Parsing Pada Aplikasi Mobile E-commerce Studi Kasus: CV V3 Tekno Indonesia,” *Pseudocode*, vol. 2, no. 1, pp. 1–9, 2015, doi: 10.33369/pseudocode.2.1.1-9.