

Metode Penyimpanan Data pada Joget Workflow (Studi Kasus: Website Electronic Human Resource)

by John Doe

Submission date: 24-Nov-2020 08:49AM (UTC+0700)

Submission ID: 1340650195

File name: publikasi-ta-joget.pdf (756.21K)

Word count: 2886

Character count: 18110

Metode Penyimpanan Data pada Joget Workflow (Studi Kasus: Website Electronic Human Resource)

Abstrak— Electronic Human Resource atau biasa disingkat dengan EHR merupakan sebuah *website* yang digunakan untuk membantu pengolahan data sumber daya manusia (SDM) terutama data pegawai dalam suatu perusahaan. EHR akan mengelola berbagai data pegawai mulai dari data pribadi sampai data pekerjaannya. Seiring berjalannya waktu, data pegawai tersebut akan mengalami perubahan. Setiap proses perubahan data akan melalui proses bisnis yang berlaku dalam perusahaan tersebut. Sistem akan mengelola banyak data dengan berbagai macam proses perubahan data sehingga akan membuat proses penyimpanan data sulit untuk ditangani. Maka dari itu, pengembangan proyek ini memerlukan mekanisme penyimpanan data yang efektif, solusinya adalah penyimpanan data pada *platform* bernama Joget Workflow. Proses penyimpanan data yang banyak sesuai proses bisnis dapat ditangani oleh *platform* ini. Joget Workflow mempunyai fitur pengembangan sistem berbasis proses bisnis dengan berbagai macam metode penyimpanan data. Setiap metode tersebut dapat digunakan sesuai dengan skema yang ada sehingga akan mempermudah proses penyimpanan data. Setelah dilakukan pengembangan dapat disimpulkan bahwa penerapan metode penyimpanan data pada Joget Workflow cocok untuk digunakan dalam pengembangan sistem dengan skema penyimpanan data yang bervariasi.

Keywords— *Electronic Human Resource; penyimpanan data; Joget Workflow.*

I. PENDAHULUAN

Sebuah organisasi terdiri dari berbagai sumber daya yang salah satunya adalah sumber daya manusia (SDM). Sumber daya manusia (SDM) sangat penting dalam sebuah organisasi karena pegawai merupakan orang yang menggunakan dan mengelola sumber daya lainnya untuk menjalankan perusahaan [1]. Kumpulan pegawai tersebut akan membentuk sebuah struktur organisasi yang terdiri dari berbagai macam unit kerja. Unit kerja pegawai dibagi berdasarkan fungsi dan pembagian kerja. Banyaknya jumlah informasi yang harus dikelola berhubungan dengan SDM sehingga memunculkan permasalahan dengan tingkat kompleksitas tinggi [2]. Apabila ada informasi yang salah, maka akan mengganggu jalannya organisasi. Seiring berjalannya waktu, informasi data pegawai tentu akan mengalami perubahan. Perubahan data yang tidak ditangani dengan baik akan riskan terjadi kesalahan informasi. Untuk itu perlu dibangun sebuah sistem informasi sumber daya manusia (HRIS). HRIS yang dibangun bernama Electronic Human Resource atau disingkat EHR. Sistem ini akan digunakan untuk mengatur data pegawai sesuai dengan proses bisnis yang ada pada sebuah perusahaan korporasi.. Dengan adanya sistem EHR maka dapat diketahui data informasi terkini dari tiap individu pegawai.

Pengolahan data akan menjadi fitur utama sistem. Salah satu di dalamnya adalah penyimpanan data sesuai proses bisnis. Data yang disimpan mempunyai banyak bentuk dengan penyimpanan data yang berbeda. Untuk itu, sistem EHR dikembangkan menggunakan *platform* Joget Workflow sebagai *tools* utama. Berbagai macam metode penyimpanan data telah disediakan oleh Joget Workflow. Setiap metode

mempunyai fungsinya masing-masing sesuai dengan penyimpanan data yang ditemui. Selain menggunakan Joget Workflow, untuk menampung data yang menggunakan DBMS MySQL. Kedua *tools* tersebut akan saling terintegrasi satu sama lain sehingga metode penyimpanan data yang digunakan dapat berfungsi dengan baik. Makalah ini diharapkan dapat memberi pengetahuan bagi praktisi untuk memanfaatkan metode penyimpanan data pada Joget Workflow sesuai dengan kebutuhan.

II. KAJIAN PUSTAKA

Sistem informasi sumber daya manusia (HRIS) dapat didefinisikan sebagai suatu sistem terintegrasi yang digunakan untuk mengumpulkan, menyimpan dan menganalisa informasi tentang sumber daya manusia dalam sebuah organisasi yang terdiri dari database, komputer aplikasi, perangkat keras dan perangkat lunak yang diperlukan untuk mengumpulkan, merekam, menyimpan, mengelola, memberikan, menyajikan dan memanipulasi data untuk fungsi – fungsi sumber daya manusia [3].

Pengembangan sistem Electronic Human Resource ini menggunakan *platform* Joget Workflow. Joget Workflow merupakan *platform* pengembangan sistem *open-source* dan *no-code* yang terkini untuk membangun sebuah sistem web *enterprise* dan automasi proses bisnis yang dapat digunakan di *cloud* [4]. Penyimpanan data pada Joget Workflow menggunakan beberapa metode antara lain:

- Bean Shell Form Binder adalah metode penyimpanan yang menggunakan implementasi dari Bean Shell. Bean Shell adalah kumpulan fungsi Java yang dapat dimasukkan ke dalam fitur penulisan bahasa dan ditulis ke dalam bahasa pemrograman Java [5].
- JDBC Binder adalah metode yang digunakan untuk menyimpan atau memuat data dari Joget maupun *database* khusus melalui *query* SQL [6]. Dalam penyimpanan data hanya perlu menulis *query* SQL untuk *select*, *insert*, *update*, dan *delete*.
- Workflow Form Binder adalah *form* untuk proses penyimpanan maupun pemuatan data yang ditangani oleh Form Builder milik Joget. *Form* ini akan menyimpan data secara otomatis tanpa perlu ada tabelnya terlebih dahulu. Tabel dan kolom akan dibuat oleh Joget dengan penamaan yang disesuaikan dengan pengaturan *form* [7].

Selain penyimpanan data, fitur lain Joget Workflow adalah memfalsifikasi pembuatan sistem dengan proses bisnis yang ada. Hal ini membantu dalam menjalankan kegiatan operasional UPT Layanan Internasional Universitas Andalas. Penggunaan Joget Workflow dapat memudahkan dan meringankan proses bisnis serta meningkatkan efisiensi kinerja [8].

III. METODOLOGI

17 Pengembangan sistem EHR dikerjakan bersama dengan tim yang terdiri dari analis sistem, programmer dan proyek manager. Proses pengerjaan dimulai dengan pengumpulan data dengan berdiskusi dengan klien. Terpilihnya Joget Workflow sebagai platform pengembangan sistem ini menjadi salah satu hasil diskusi tersebut. Kemudian analisis sistem menganalisis data yang telah diperoleh. Hasil analisis menjadi kumpulan task yang diberikan kepada programmer. Pembuatan form menjadi task utama dalam pengembangan sistem EHR. Berbagai macam form yang berbeda akan dibuat.

Pada tahap selanjutnya yaitu implementasi yang dilakukan oleh programmer. Di sini, programmer perlu mengetahui detail task secara rinci. Hal ini untuk mempermudah programmer dalam memahami data yang akan diolah dan ditampilkan oleh sistem. Selanjutnya programmer akan melakukan pemilihan metode penyimpanan data pada form karena Joget Workflow menyediakan berbagai macam metode penyimpanan data. Pemilihan metode akan digunakan sesuai dengan kasus yang ada. Misalnya penyimpanan data yang sederhana tanpa skema khusus menggunakan Workflow Form Binder, penyimpanan data dengan query SQL menggunakan JDBC Binder dan penyimpanan data dengan bantuan query Java menggunakan Bean Shell Form Binder. Tahap terakhir yaitu pengujian data yang tersimpan, apakah telah berhasil masuk atau tidak. Pengujian data dapat dilakukan dengan melakukan debugging. Debugging ini menggunakan log Tomcat.

16

IV. HASIL DAN PEMBAHASAN

A. Analisis Kebutuhan Sistem

Sistem EHR mempunyai beberapa fitur form di antaranya yaitu Profil Saya, Pengurusan Staf Baru, Pengurusan Organisasi, dan Pengurusan Pegawai. Fitur Profil Saya digunakan untuk menampilkan data profil pegawai yang telah masuk. Pada halaman ini pegawai dapat melakukan perubahan data profilnya. Profil diri yang diubah perlu disetujui dari pihak pengelola agar data profil dapat berubah. Fitur kedua adalah Pengurusan Staf Baru dimana data profil pegawai baru akan ditampilkan. Pengelola akan melakukan konfirmasi akhir kepada data tersebut agar data pegawai baru ini terdaftar pada susunan pegawai yang ada. Fitur ketiga adalah Pengurusan Organisasi, fitur ini digunakan untuk mengolah data divisi yang ada. Fitur Pengurusan Pegawai dimana fitur ini mempunyai beberapa form berbeda untuk mengubah data pegawai yang berkaitan dengan informasi divisi kerjanya. Form-form ini antara lain: Kenaikan Pangkat, Memangku, dan Penukaran Penempatan secara Administrasi. Perubahan data pada fitur Pengurusan Pegawai disimpan ke dalam tabel sementara terlebih dahulu. Penyimpanan lanjut ke data pegawai akan dilakukan secara otomatis oleh cron Laravel.

Setiap form tadi menampilkan data yang berbeda-beda sehingga skema penyimpanan pada tiap form akan berbeda. Terlebih lagi ada fitur form yang di dalamnya masih terdapat sub form. Selain single row, pada form tertentu dapat berbentuk multiple rows. Tabel 1. memperlihatkan pembagian fitur form dan jenis row data secara terperinci.

TABLE I. ANALISIS KEBUTUHAN FORM SISTEM

Fitur	Nama Form	Jenis Row Data	Skema Penyimpanan Data
Profil Saya	Informasi Pribadi	Single	Terdapat kondisi pengecekan dengan <i>while</i> dan <i>if</i>
	Informasi Pendidikan	Multiple	Terdapat kondisi pengecekan dengan <i>while</i> dan <i>if</i>
Pengurusan Staf Baru	Tukar Penempatan	Single	Terdapat kondisi pertukaran nilai data lama dan baru
	Pengesahan Laporan Diri	Multiple	Terdapat kondisi pengecekan dengan <i>while</i> dan <i>if</i>
Pengurusan Organisasi	Tambah Jabatan	Single	Terdapat kondisi perubahan nilai yang disimpan dengan <i>case when</i>
	Ubah Jabatan	Single	Terdapat kondisi perubahan nilai yang disimpan dengan <i>case when</i>
Pengurusan Pegawai	Kenaikan Pangkat	Single	Penyimpanan data sementara ke dalam satu tabel
	Memangku	Single	Penyimpanan data sementara ke dalam satu tabel
	Penukaran Penempatan secara Administrasi	Single	Penyimpanan data sementara ke dalam satu tabel

B. Penentuan Metode Penyimpanan Data

Setelah mengetahui skema penyimpanan yang digunakan dan jenis row-nya, maka tahap berikutnya adalah melakukan penentuan metode penyimpanan data yang akan digunakan di tiap form. Penentuan metode dapat dilihat pada Tabel II.

TABLE II. METODE PENYIMPANAN DATA

No	Nama Form	Metode
1	Informasi Pribadi	Bean Shell Form Binder
2	Informasi Pendidikan	Bean Shell Form Binder
3	Tukar Penempatan	Bean Shell Form Binder
4	Pengesahan Laporan Diri	Bean Shell Form Binder
5	Tambah Jabatan	JDBC Binder
6	Ubah Jabatan	JDBC Binder
7	Kenaikan Pangkat	Workflow Form Binder
8	Memangku	Workflow Form Binder
9	Penukaran Penempatan secara Administrasi	Workflow Form Binder

Tabel II memperlihatkan empat form dengan metode Bean Shell Form Binder. Hal ini karena form-form tersebut memiliki skema penyimpanan data yang rumit. Skema dapat dikatakan rumit apabila skema tersebut mempunyai kondisi percabangan *if* dan perulangan *while* sedangkan skema yang sederhana hanya membutuhkan query dasar SQL seperti *select*, *insert*, *update*, dan *delete*. Bean Shell Form Binder juga dapat pengolahan data dengan jenis data single row maupun multiple rows. Selanjutnya metode JDBC Binder digunakan pada dua form dengan jenis data single row dan skema yang digunakan membutuhkan fungsi *case when*. Fungsi ini dapat

untuk melakukan penyimpanan data adalah insert dan update. Sebelum itu data dari *form* perlu dipilih terlebih dahulu. Pemilihan disesuaikan dengan *field form* yang ada. Gambar 5. memperlihatkan cara untuk memilih data dari *field form*.

Gambar 5. Pemilihan *field* pada *form* Informasi Pribadi

Selanjutnya dibuat *query* untuk *insert* data. Karena penyimpanan ini mempunyai kondisi untuk membandingkan dengan data yang telah ada maka perlu membuat *query select* data pembandingan. *Query insert* dan *select* dapat dilihat pada Gambar 6.

```
String insertQuery = "insert into temp_update_dir_staff_profile(
    userId, columnName,
    new_value, categoryId,
    maklumat, old_value, permohonanId)
+" select ?,?,?, 'Pribadi', ?, ?,
concat('PROFIL',DATE_FORMAT(NOW(), '%d%m%y'),'-',
lpad(a.jumlah+1,7,0)) from (select count(1) as jumlah
from profile_main) a";

PreparedStatement istmt = con.prepareStatement(insertQuery);

String selectQuery = "select b.tel_m, b.tel_ext"
+" from dir_user a"
+" left join dir_staff_profile b on a.id = b.fk_userId"
+" where a.id = ?";

PreparedStatement sstmt = con.prepareStatement(selectQuery);
```

Gambar 6. *Query insert* dan *select* pada Informasi Pribadi

Setelah *query insert* dan *select* telah siap maka dibuat *query update* untuk memasukkan data ke tabel utama. Pengecekan kondisi akan dilakukan pada setiap *row* yang telah dipilih. Gambar 7 memperlihatkan *query* secara lebih rinci.

```
String user = "#currentUser.id#";
sstmt.setString(1, user);
ResultSet rs = sstmt.executeQuery();
while (rs.next()) {
    if (!row.getProperty("new_tel_m")
        .equalsIgnoreCase(rs.getString("tel_m"))) {
        istmt.setString(1, user);
        istmt.setString(2, "tel_m");
        istmt.setString(3, row.getProperty("new_tel_m"));
        istmt.setString(4, "No Telefon Bimbit");
        istmt.setString(5, rs.getString("tel_m"));
        istmt.executeUpdate();
    }
    if (!row.getProperty("new_tel_ext")
        .equalsIgnoreCase(rs.getString("tel_ext"))) {
        istmt.setString(1, user);
        istmt.setString(2, "tel_ext");
        istmt.setString(3, row.getProperty("new_tel_ext"));
        istmt.setString(4, "No Sambungan");
        istmt.setString(5, rs.getString("tel_ext"));
        istmt.executeUpdate();
    }
}
```

Gambar 7. *Query update* dan pengecekan data pada Informasi Pribadi

Penyimpanan data Bean Shell Form Binder ini dilakukan pada *form* Informasi Pribadi. Tampilan dari *form* tersebut dapat dilihat pada Gambar 8.

Gambar 8. Halaman *form* Informasi Pribadi

2) JDBC Binder

Metode yang kedua adalah JDBC Binder. Langkah yang untuk membuat *query* penyimpanan disini hampir sama dengan *query* di *SQL*. Perbedaannya hanya pada nilai data yang diambil karena *platform* Joget Workflow menyediakan fungsi untuk menyimpan data *realtime* bernama *hash variable*.

Penyimpanan data ini memerlukan *query* dengan fungsi *insert* dan *update*. *Query insert* akan memilih tabel dan kolom yang digunakan untuk menyimpan data. Setelah itu terdapat *query select* untuk memilih data yang ingin disimpan. Jumlah kolom data yang dipilih pada *query insert* dan *select* harus sama. *Query* tersebut dapat dilihat pada Gambar 9.

```
INSERT dir_department(
    id, name, description,
    hod, organizationId,
    parentId, createdBy,
    createdByName, dateCreated,
    org_order)
select
    case when (a.id is null or id = '')
        then 'J001' else id end as id,
    {name}, {description} as description,
    case when {hod} = '' then null
        else {hod} end,
    'ppj_cartaOrganisasi', {parentId},
    '#currentUser.id#' as createdBy,
    '#currentUser.firstName#' as createdByName,
    now() as dateCreated, {org_order} as org_order
from (
    select (
        select concat(
            '', lpad(substr(dir_department.id,3,3) + 1,3,0)
        ) as id from dir_department
        where id like 'J%' and organizationId = 'ppj_cartaOrganisasi'
        order by dir_department.id desc limit 1) id) a;
```

Gambar 9. *Query* penyimpanan data pada Tambah Jabatan

Metode JDBC Binder ini digunakan pada *form* Tambah Jabatan. Gambar 10 memperlihatkan tampilan *form* Tambah Jabatan.

Gambar 10. Halaman *form* Tambah Jabatan

3) Workflow Form Binder

Metode Workflow Form Binder termasuk metode dengan cara penggunaan yang paling mudah diantara metode lain di platform Joget Workflow. *Form* yang sudah dibuat hanya perlu diubah cara penyimpanan data -nya menjadi Workflow Form Binder. Proses penyimpanan akan ditangani oleh Joget. Data yang disimpan akan mempunyai nama tabel dan kolom yang sesuai dengan *form*-nya. Akan tetapi penamaan akan diubah sedikit pada bagian awalnya. Tabel III akan memperlihatkan contoh perbandingan antara nama di *form* dengan nama di *database*.

TABLE III. PERBANDINGAN NAMA TABEL DAN KOLON

Jenis	Nama di Form	Nama di Database
Tabel	temp_pendtabdiran	app_fd_ temp_pendtabdiran
Kolom	positionId	c_positionId
	employeeCode	c_employeeCode
	from_departmentId	c_from_departmentId
	userId	c_userId
	employeeId	c_employeeId
	jobId	c_jobId
	firstName	c_firstName
	jobName	c_jobName
	to_departmentId	c_to_departmentId
	hod	c_hod
	groupId	c_groupId
	startDate	c_startDate
	status_update_sap	c_status_update_sap
status_update_main	c_status_update_main	

Metode Workflow Form Binder ini digunakan pada *form* Penukaran secara Administrasi. Gambar 11 memperlihatkan tampilan *form* Penukaran secara Administrasi.

Gambar 11. Halaman *form* Penukaran secara Administrasi

E. Pengujian

Penyimpanan data yang telah dibuat akan melalui proses pengujian terlebih dahulu. Hal ini untuk mengetahui data yang disimpan telah berhasil atau tidak. Pengujian dilakukan melalui *debugging* data. *Debugging* Joget Workflow menggunakan Tomcat *log* dengan menjalankan perintah berikut di terminal.

```
tail -f apache-tomcat-8.5.41/logs/catalina.out
```

Terminal akan menampilkan *log* yang berjalan pada sistem EHR termasuk proses penyimpanan data. Halaman *debugging* ini akan ditampilkan pada Gambar 12. Setelah itu hasil pengujian dapat dilihat pada Tabel. IV.

```
File Edit View Search Terminal Help
brayad@IP: ~/Downloads/joget-enterprise-linux-7.0.105 tail -f apache-tomcat-8.5.41/logs/catalina.out
at org.apache.catalina.core.StandardThreadExecutor.run(ThreadPoolExecutor.java:64)
at org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
at java.lang.Thread.run(Thread.java:748)
Caused by: java.sql.SQLException: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '(order by a.datecreated asc) as no, a.id, a.apply no, b.firstName, c.name, a.site' at line 1
at org.marlab.jdbc.internal.protocol.AbstractQueryProtocol.readFromPacket(AbstractQueryProtocol.java:159)
at org.marlab.jdbc.internal.protocol.AbstractQueryProtocol.readPacket(AbstractQueryProtocol.java:161)
at org.marlab.jdbc.internal.protocol.AbstractQueryProtocol.getResult(AbstractQueryProtocol.java:140)
at org.marlab.jdbc.internal.protocol.AbstractQueryProtocol.executeQuery(AbstractQueryProtocol.java:249)
at org.marlab.jdbc.ClientSidePreparedStatement.executeInternal_arroundBody(ClientSidePreparedStatement.java:216)
...
ERROR 16 Nov 2020 15:13:05 org.joget.plugin.enterprise.jdbcatalina.loader - java.sql.SQLException: (conn=388) Table 'app_fd_temp_pendtabdiran' doesn't exist
java.sql.SQLException: (conn=388) Table 'app_fd_temp_pendtabdiran' doesn't exist
at org.marlab.jdbc.internal.util.exceptions.ExceptionMapper.get(ExceptionMapper.java:24)
at org.marlab.jdbc.internal.util.exceptions.ExceptionMapper.get(ExceptionMapper.java:164)
at org.marlab.jdbc.MarLabStatement.executeExceptionLog_arroundBody(MarLabStatement.java:204)
at org.marlab.jdbc.MarLabStatement.executeExceptionLog_arroundBody(MarLabStatement.java:127)
at org.marlab.jdbc.MarLabStatement.executeExceptionLog_arroundBody(MarLabStatement.java:11)
at org.marlab.jdbc.ClientSidePreparedStatement.executeInternal_arroundBody(ClientSidePreparedStatement.java:221)
at org.marlab.jdbc.ClientSidePreparedStatement.executeInternal_arroundBody(ClientSidePreparedStatement.java:127)
```

Gambar 12. Halaman *debugging* Tomcat *log*

TABLE IV. HASIL PENGUJIAN

Nama Form	Berhasil/Gagal
Informasi Peribadi	Berhasil
Informasi Pendidikan	Berhasil
Tukar Penempatan	Berhasil
Pengesahan Laporan Diri	Berhasil
Tambah Jabatan	Berhasil
Ubah Jabatan	Berhasil
Kenaikan Pangkat	Berhasil
Memangku	Berhasil
Penukaran Penempatan secara Administrasi	Berhasil

F. Pembahasan

Berdasarkan pemaparan sebelumnya, dapat diketahui bahwa Joget Workflow menyediakan berbagai macam metode untuk melakukan penyimpanan data. Hal yang perlu diperhatikan dalam pemilihan metode ini adalah seberapa

rumit skema yang akan digunakan saat proses penyimpanan data. Skema yang rumit terdapat kondisi *if* atau perulangan *while*. Skema tersebut akan menggunakan metode Bean Shell Form Binder, dan JDBC Form Binder digunakan untuk skema yang sederhana seperti pemanfaatan *query* dasar SQL. Sementara penyimpanan data tanpa adanya skema akan menggunakan Workflow Form Binder. Pemanfaatan penyimpanan data pada Joget Workflow ini mempermudah proses penyimpanan data sesuai proses bisnis terutama dalam hal pengolahan data karena metode yang digunakan sesuai skema yang ada. Hal ini sesuai dengan permintaan klien untuk membuat sistem dengan pengolahan data yang berjalan sesuai proses bisnis yang ada.

V. KESIMPULAN DAN SARAN

Penyimpanan data dalam Electronics Human Resources (EHR) dapat dikembangkan menggunakan metode yang disediakan oleh *platform* Joget Workflow. Berbagai skema penyimpanan dapat ditangani dengan metode yang berbeda. Skema yang kompleks seperti penggunaan kondisi percabangan *if* atau perulangan *while* dapat menggunakan metode Bean Shell Form Binder. Skema dengan *query* SQL dapat menggunakan JDBC Form Binder, sedangkan untuk skema sederhana yang hanya butuh penyimpanan data sementara pada satu tabel dapat menggunakan Workflow Form Binder.

Berdasarkan hasil makalah ini, pengembangan aplikasi EHR difokuskan pada *form* penyimpanan data, dapat disarankan agar pengembangan selanjutnya untuk lebih memaksimalkan pembuatan *form* dengan fitur *workflow* pada Joget Workflow. Hal itu perlu, karena fitur ini digunakan untuk membuat sekumpulan *form* dalam *workflow* yang sama sehingga alur pengiriman data pada proses bisnis akan dapat diketahui dengan baik. Selain itu dalam melakukan *query* khususnya SQL dapat memanfaatkan *stored procedure*. Pemanfaatan *stored procedure* dalam menampung *query* akan mengurangi *traffic* data antara aplikasi dan *database*.

VI. REFERENCES

- [1] Franky and J. Devitra, "Analisis dan Perancangan Sistem Informasi," *Jurnal Manajemen Sistem Informasi*, vol. 1, no. 2, pp. 185-190, 2016.
- [2] H. F. Muhammad and P. S. Niki, "Pengembangan Human Resource Information System (HRIS)," *JUPITER (Jurnal Pendidikan Teknik Elektro)*, vol. 03, no. 02, pp. 1-12, 2018.
- [3] S. N. Ambo and M. Ghufron, "Rancang Bangun Aplikasi Human Resource Information System (HRIS) Menggunakan Metode Model View Controller (MVC)," *ISSN : 2407 - 1846*, 2015.
- [4] Joget Workflow, "Joget Workflow - JOGET: OPEN SOURCE WORKFLOW AND LOW CODE PLATFORM FOR DIGITAL TRANSFORMATION," 2019. [Online]. Available: <https://www.joget.com/joget-workflow.html>. [Accessed 23 April 2020].
- [5] Administrator and Justin, "Bean Shell Form Binder - Knowledge Base for v6 - Joget | COMMUNITY," 15 Januari 2018. [Online]. Available: <https://dev.joget.org/community/display/KBv6/Bean+Shell+Form+Binder>. [Accessed 24 April 2020].
- [6] Justin and Andrew, "JDBC Form Binder - Knowledge Base for v6 - Joget | COMMUNITY," 29 Oktober 2019. [Online]. Available: <https://dev.joget.org/community/display/KBv6/JDBC+Form+Binder>. [Accessed 24 April 2020].
- [7] Administrator and Justin, "Workflow Form Binder - Knowledge Base for v6 - Joget | COMMUNITY," 23 Agustus 2017. [Online]. Available: <https://dev.joget.org/community/display/KBv6/Workflow+Form+Binder>. [Accessed 24 April 2020].
- [8] K.-I. O. David, "Implementasi Joget Workflow Pada Proses Bisnis Unit Pelaksana Teknis (UPT) Layanan Internasional Universitas Andalas," pp. 128-129, 2017.

Metode Penyimpanan Data pada Joget Workflow (Studi Kasus: Website Electronic Human Resource)

ORIGINALITY REPORT

9%

SIMILARITY INDEX

9%

INTERNET SOURCES

0%

PUBLICATIONS

0%

STUDENT PAPERS

PRIMARY SOURCES

1	jurnal.umj.ac.id Internet Source	2%
2	dev.joget.org Internet Source	1%
3	jurnal.umk.ac.id Internet Source	1%
4	repo.unand.ac.id Internet Source	1%
5	ejurnal.stmik-budidarma.ac.id Internet Source	1%
6	scholar.unand.ac.id Internet Source	<1%
7	www.joget.com Internet Source	<1%
8	conference.binadarma.ac.id Internet Source	<1%
9	docobook.com	

Internet Source

<1%

10

jurnalmsi.stikom-db.ac.id

Internet Source

<1%

11

workflowsistem.blogspot.com

Internet Source

<1%

12

fr.scribd.com

Internet Source

<1%

13

doku.pub

Internet Source

<1%

14

ojs.ummetro.ac.id

Internet Source

<1%

15

eprints.ums.ac.id

Internet Source

<1%

16

research.kalbis.ac.id

Internet Source

<1%

17

insearch.unibi.ac.id

Internet Source

<1%

18

dosenit.com

Internet Source

<1%

19

repository.unpas.ac.id

Internet Source

<1%

Exclude quotes Off

Exclude matches Off

Exclude bibliography On