

**QUERY VISUAL PADA WORKFLOW REPOSITORY  
BERBASIS GRAPH DATABASE MENGGUNAKAN  
BUSINESS PROCESS MODEL AND NOTATION  
QUERY (BPMN-Q)**

**SKRIPSI**

**Oleh:  
GALANG LUHUR PEKERTI  
NIM. 14650060**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2021**

**QUERY VISUAL PADA WORKFLOW REPOSITORY  
BERBASIS GRAPH DATABASE MENGGUNAKAN  
BUSINESS PROCESS MODEL AND NOTATION  
QUERY (BPMN-Q)**

**SKRIPSI**

**Diajukan kepada:**

**Fakultas Sains dan Teknologi  
Universitas Islam Negeri (UIN)  
Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:**

**GALANG LUHUR PEKERTI  
NIM. 14650060**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2021**

**HALAMAN PERSETUJUAN**

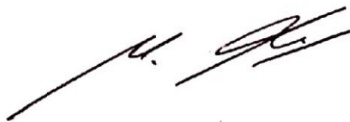
**QUERY VISUAL PADA WORKFLOW REPOSITORY  
BERBASIS GRAPH DATABASE MENGGUNAKAN  
BUSINESS PROCESS MODEL AND NOTATION  
QUERY (BPMN-Q)**

**SKRIPSI**

**OLEH:  
GALANG LUHUR PEKERTI  
NIM. 14650060**

Telah Diperiksa dan Disetujui untuk Diuji  
Tanggal : Juni 2021

Pembimbing I



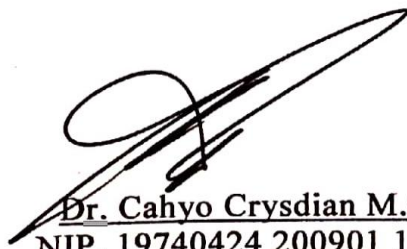
M. Ainul Yaqin, M.Kom  
NIP. 19761013 200604 1 004

Pembimbing II



Prof. Dr. Suhartono, S.Si M.Kom  
NIP. 19680519 200312 1 001

Mengetahui,  
Ketua Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Cahyo Crysdian M.CS  
NIP. 19740424 200901 1 008

**HALAMAN PENGESAHAN**

**QUERY VISUAL PADA WORKFLOW REPOSITORY  
BERBASIS GRAPH DATABASE MENGGUNAKAN  
BUSINESS PROCESS MODEL AND NOTATION  
QUERY (BPMN-Q)**

**SKRIPSI**

**OLEH:  
GALANG LUHUR PEKERTI  
NIM. 14650060**

Telah Dipertahankan di Depan Dewan Penguji  
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Pada Tanggal Juni 2021

**Susunan Dewan Penguji**


**Penguji Utama** : Fatchurrochman, M.Kom  
NIP. 19700731 200501 1 002

**Ketua Penguji** : Syahiduz Zaman, M.Kom  
NIP. 19700502 200501 1 005

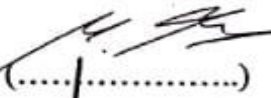
**Sekretaris Penguji** : M. Ainul Yaqin, M.Kom  
NIP. 19761013 200604 1 004

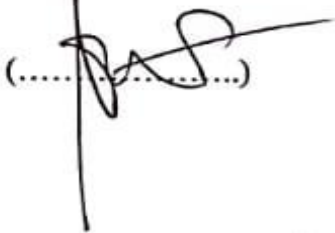
**Anggota Penguji** : Prof. Dr. Suhartono, S.Si M.Kom  
NIP. 19680519 200312 1 001

**Tanda Tangan**

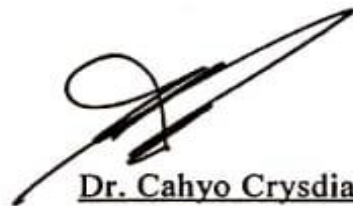
  
(.....)

  
(.....)

  
(.....)

  
(.....)

Mengetahui,  
Ketua Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang

  
Dr. Cahyo Crysdian M.CS  
NIP. 19740424 200901 1 008

## HALAMAN PERSEMBAHAN

الحمد لله رب العالمين

Puji syukur kehadiran Allah SWT, shalawat serta salam kepada Rasulullah SAW. Penulis persembahkan skripsi ini kepada:

Kedua orang tua penulis tercinta, Bapak Ir. Yusdianto dan Ibu Drs. Liyna Haliya serta keluarga di rumah yang selalu membimbing penulis, memberikan do'a, dukungan, serta motivasi yang tidak terhingga. Dan terimakasih kepada saudara-saudara saya yang senantiasa mengingatkan saya ketika melakukan kesalahan, membantu saya dalam segala hal, dan mendukung saya dalam menyelesaikan tugas akhir ini.

Dosen pembimbing penulis Bapak M. Ainul Yaqin, M.Kom dan Bapak Dr. Suhartono, M.kom yang telah dengan sabar membimbing penelitian skripsi ini dan selalu memberikan masukan dan semangat untuk menjalani setiap tahapan skripsi.

Seluruh dosen Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang telah membimbing dan memberikan ilmunya yang bermanfaat.

Seluruh Keluarga Teknik Informatika, terutama teman-teman Teknik Informatika angkatan 2014 yang telah memberikan semangat dan motivasi. Serta teman-teman penulis yang terus mendorong penulis untuk melewati tahapan skripsi.

## HALAMAN PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Galang Luhur Pekerti  
NIM : 14650060  
Fakultas/Jurusan : Sains dan Teknologi / Teknik Informatika  
Judul Skripsi : *Query Visual pada Workflow Repository Berbasis Graph Database Menggunakan Business Process Model and Notation Query (BPMN-Q)*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 15 Juni 2021

Yang Membuat Pernyataan,



Galang Luhur Pekerti

NIM. 14650060

## KATA PENGANTAR

Puji syukur bagi Allah SWT yang telah mencurahkan nikmat serta kekuatan sehingga penulis dapat menyelesaikan skripsi dengan baik. Tidak lupa juga shalawat serta salam kepada nabi Muhammad S A W yang telah membimbing umatnya ke jalan yang benar.

Dalam mengerjakan skripsi ini, tidak luput dari bantuan, bimbingan, serta dukungan berbagai pihak baik secara moril maupun materil. Oleh karenanya, izinkan penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Abdul Haris, M.Ag, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Ibu Dr. Sri Harini, M.Si, selaku Dekan Fakultas Sains dan Teknologi.
3. Bapak Dr. Cahyo Crysdiyan, selaku Ketua Jurusan Teknik Informatika.
4. Bapak M. Aiunul Yaqin, M.Kom selaku dosen pembimbing I dan dosen wali yang telah meluangkan waktu untuk membimbing serta memberi masukan kepada penulis dalam mengerjakan skripsi ini.
5. Bapak Dr. Suhartono, M.kom selaku dosen pembimbing II yang juga senantiasa memberikan masukan dan petunjuk tentang penulisan skripsi ini.
6. Seluruh dosen Teknik Informatika yang telah mencurahkan ilmunya kepada penulis selama kuliah.
7. Bapak, Ibu, serta keluarga besar penulis yang selalu meberikan doa serta nasihat kepada penulis selama kuliah.
8. Wening Pramesti yang selalu membantu dalam hal penulisan skripsi dan sebagai kakak yang selalu membantu dalam kesulitan.
9. Teman-teman Teknik Informatika 2014, khususnya Muhammad Burhanudin, Ulung Muhammad Bestari, Dinda Ockta N, Nindy Agustina dan Lia Andini, yang telah membantu penulis dalam menyelesaikan skripsi.

Dalam Penulisan skripsi ini, masih terdapat kekurangan. Namun penulis berharap, skripsi ini dapat memberikan manfaat kepada pembaca dan kepada penulis. *Aamiin* .

Malang, 15 Juni 2021

Penulis

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERSEMBAHAN .....	iv
HALAMAN PERNYATAAN KEASLIAN TULISAN .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR .....	x
DAFTAR KODE SUMBER .....	xi
ABSTRAK .....	xii
ABSTRACT.....	xiii
المخلص .....	xiv
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang Masalah.....	1
1.2 Pernyataan Masalah .....	2
1.3 Tujuan Penelitian .....	2
1.4 Manfaat Penelitian .....	3
1.5 Batasan Masalah.....	3
BAB 2 STUDI PUSTAKA .....	4
2.1 <i>Workflow</i> .....	4
2.2 <i>Workflow Management System (WfMS)</i> .....	4
2.3 <i>Workflow Repository</i> .....	5
2.4 BPMN .....	5
2.5 <i>XML Process Definition Language (XPDL)</i> .....	11
2.6 <i>Graph Database</i> .....	13
2.7 BPMN-Q .....	14
2.8 Penelitian Terkait .....	16
BAB 3 DESAIN DAN IMPLEMENTASI .....	19
3.1 Desain Penelitian.....	19
3.1.1 Tipe Penelitian .....	19
3.1.2 Prosedur Penelitian.....	19
3.1.3 Identifikasi Masalah .....	21
3.1.4 Perancangan Sistem .....	21
3.1.5 Implementasi BPMN-Q .....	30
3.1.6 Pengujian Akurasi .....	34
3.2 Perangkat yang digunakan .....	35
3.2.1 Perangkat keras .....	35
3.2.2 Perangkat lunak.....	35
3.3 Interface Aplikasi .....	36
3.4 Langkah Uji Coba .....	40
BAB 4 UJI COBA DAN PEMBAHASAN .....	65
4.1 Hasil Uji Coba.....	65
4.1.1 <i>Sequence</i> .....	68
4.1.2 <i>Branch</i> .....	71
4.1.3 Siklus.....	74



4.1.4	Campuran .....	76
4.2	Pembahasan.....	79
4.3	Integrasi Islam.....	83
BAB 5 KESIMPULAN DAN SARAN .....		85
5.1	Kesimpulan .....	85
5.2	Saran.....	85
DAFTAR PUSTAKA .....		86

## DAFTAR TABEL

Tabel 2.1 <i>Query</i> BPMN-Q .....	15
Tabel 3.1 Skema Notasi BPMN dalam <i>Graph Database</i> (NEO4J) .....	26
Tabel 3.2 Notasi BPMN-Q ke <i>Cypher</i> .....	28
Tabel 3.3 Spesifikasi Perangkat Keras .....	35
Tabel 3.4 Spesifikasi Perangkat lunak untuk Pembuatan <i>Web</i> .....	36
Tabel 3.5 Perangkat Lunak untuk Pengujian .....	36
Tabel 3.6 Data BPMN <i>Sequence</i> .....	40
Tabel 3.7 Data BPMN <i>Branch</i> .....	42
Tabel 3.8 Data BPMN Siklus .....	56
Tabel 3.9 BPMN-Q <i>sequence</i> .....	60
Tabel 3.10 BPMN-Q <i>Branch</i> .....	61
Tabel 3.11 BPMN-Q Siklus .....	62
Tabel 3.12 BPMN-Q <i>Negative Path</i> .....	62
Tabel 4.1 BPMN-Q ke <i>Cypher</i> .....	65
Tabel 4.2 Keterangan TP, FP, TN dan FN .....	68
Tabel 4.3 Hasil uji coba pada data <i>sequence</i> .....	69
Tabel 4.4 Nilai <i>precision</i> , <i>recall</i> dan <i>accuracy</i> pada data <i>sequence</i> .....	70
Tabel 4.5 Nilai rata-rata dan persentase pada data <i>sequence</i> .....	70
Tabel 4.6 Hasil uji coba pada data <i>branch</i> .....	72
Tabel 4.7 Nilai <i>precision</i> , <i>recall</i> dan <i>accuracy</i> pada data <i>branch</i> .....	73
Tabel 4.8 Nilai rata-rata dan persentase pada data <i>branch</i> .....	73
Tabel 4.9 Hasil uji coba pada data siklus .....	74
Tabel 4.10 Nilai <i>precision</i> , <i>recall</i> dan <i>accuracy</i> pada data siklus .....	75
Tabel 4.11 Nilai rata-rata dan persentase pada data siklus .....	76
Tabel 4.12 Hasil uji coba pada data campuran .....	76
Tabel 4.13 Nilai <i>precision</i> , <i>recall</i> dan <i>accuracy</i> pada data campuran .....	78
Tabel 4.14 Nilai rata-rata dan persentase pada data campuran .....	79

## DAFTAR GAMBAR

Gambar 2.1 Notasi <i>Event</i> .....	7
Gambar 2.2 Notasi <i>Activity</i> .....	8
Gambar 2.3 Notasi <i>Gateway</i> .....	8
Gambar 2.4 Notasi <i>Sequence Flow</i> .....	9
Gambar 2.5 Notasi <i>Message Flow</i> .....	9
Gambar 2.6 Notasi <i>Association</i> .....	9
Gambar 2.7 Notasi <i>Data</i> .....	10
Gambar 2.8 Notasi <i>Artifak</i> .....	10
Gambar 2.9 Notasi <i>Swimmer</i> .....	11
Gambar 2.10 Pemodelan BPMN .....	12
Gambar 3.1 Prosedur Penelitian .....	20
Gambar 3.2 Alur Kerja Sistem .....	21
Gambar 3.3 BPMN Kesantrian .....	31
Gambar 3.4 BPMN pada NEO4J .....	31
Gambar 3.5 <i>Query</i> BPMN-Q .....	32
Gambar 3.6 Hasil <i>Query</i> A .....	33
Gambar 3.7 Hasil <i>Query</i> B .....	33
Gambar 3.8 Hasil <i>Query</i> C .....	34
Gambar 3.9 Halaman <i>Login</i> .....	37
Gambar 3.10 Halaman <i>Dashboard</i> .....	37
Gambar 3.11 Halaman <i>Guru</i> .....	38
Gambar 3.12 Halaman <i>siswa</i> .....	38
Gambar 3.13 Halaman <i>User</i> .....	39

## DAFTAR KODE SUMBER

Kode Sumber 2.1 Kode XPDL dari <i>header</i> BPMN .....	13
Kode Sumber 3.1 XPDL .....	22
Kode Sumber 3.2 Dekripsi .....	23
Kode Sumber 3.3 <i>Activity</i> .....	23
Kode Sumber 3.4 Data <i>Object</i> .....	24
Kode Sumber 3.5 <i>Flow</i> .....	24
Kode Sumber 3.6 <i>Flow Data</i> .....	24
Kode Sumber 3.7 Data <i>Store</i> .....	25
Kode Sumber 3.8 <i>Message Flow</i> .....	25
Kode Sumber 3.9 Association .....	25
Kode Sumber 3.10 <i>Shape</i> .....	27
Kode Sumber 3.11 <i>Connect</i> .....	27
Kode Sumber 3.12 <i>Master</i> .....	28

## ABSTRAK

Pekerti, Galang Luhur. 2021. *Query Visual pada Workflow Repository Berbasis Graph Database Menggunakan Business Process Model and Notation (BPMN-Q)*. Skripsi. Jurusan Teknik Informatika, Fakultas Sains dan Teknologi. Universitas Islam Negeri Maulana Malik Ibrahim Malang.  
Pembimbing: (I) M. Ainul Yaqin, M.Kom.,  
(II) Dr. Suhartono M.Kom.

---

**Kata Kunci:** *Workflow, Workflow Repository, BPMN-Q, BPMN, XPDL, Graph Database, NEO4J, Cypher.*

Jumlah *workflow* yang digunakan perusahaan cukup besar. Oleh karena itu, *workflow repository* diperlukan untuk menyimpan alur kerja. Pada umumnya *workflow repository* menggunakan RDBMS (*Relational Database Management System*). Selain itu pencarian pada *workflow repository* sebagian besar menggunakan teks. Sedangkan *workflow* berbentuk *graph*. Penelitian ini menggunakan BPMN-Q untuk *query visual* pada *workflow repository* berbasis *graph database* (NEO4J). Pertama BPMN disimpan dalam format XPDL supaya dapat diparsing. Kemudian XPDL diparsing. Selanjutnya data dari XPDL disimpan dalam *graph database* (NEO4J) sebagai DBMS. Untuk pencarian menggunakan BPMN-Q yang dibuat dengan *microsoft visio 2007*. *Query* BPMN-Q akan disimpan dalam format VDX. Lalu diparsing dan diterjemahkan ke *cypher*. *Cypher* memproses dan menghasilkan BPMN yang sesuai dalam bentuk daftar. Penelitian ini juga memberikan skema *database* untuk penyimpanan *workflow* dalam bentuk grafik. Performa penggunaan BPMN-Q pada *workflow repository* berbasis *graph database* adalah sangat baik. Performa didapat dari persentase *precision* yang bernilai 98.75%.

## ABSTRACT

Pekerti, Galang Luhur. 2021. **Visual Query on Graph Database-Based Repository Workflow Using Business Process Model and Notation (BPMN-Q)**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University of Malang.

Pembimbing: (I) M. Ainul Yaqin, M.Kom.,  
(II) Dr. Suhartono M.Kom.

---

**Kata Kunci:** *Workflow, Workflow Repository, BPMN-Q, BPMN, XPD, Graph Database, NEO4J, Cypher.*

The number of workflows used by companies is quite large. Therefore, a workflow repository is required to store the workflow. In general, repository workflows using RDBMS (Relational Database Management System). In addition, searches on the repository workflow as large as using text. While the workflow is graph-shaped. This study used BPMN-Q to query visuals on graph database-based repository workflows (NEO4J). First BPMN is saved in XPD format so that it can be parsed. Then XPD parsed. Furthermore, the data from the XPD stored in a graph database (NEO4J) as a DBMS. To search by using BPMN-Q that was created by microsoft visio 2007. BPMN-Q queries will be saved in VDX format. Then parsed and translated to cypher. Cypher processes and generates the BPMN that matches the list form. This research also provided a database schema for workflow storage in the form of graphs. The performance of BPMN-Q usage in graph database-based repository workflows is excellent. Performance is obtained from precision percentage is 98.75%.

## المخلص

بكرت، جلانج لوهر. 2021 الاستعلام المرني على سير العمل مستودع القائم على قاعدة بيانات الرسم البياني باستخدام نموذج عملية الأعمال والتدوين (BPMN-Q). البحث الجامعي. قسم التقنية المعلوماتية كلية العلوم الكلمات الرئيسية: مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف (I): محمد أنل يقين الماجستير . (II) سهرتنو الماجستير

الكلمات الرئيسية : سير العمل، مستودع سير العمل، BPMN-Q، BPMN، XPD L، قاعدة بيانات الرسم البياني، NEO4J، سايفر.

عدد مهام سير العمل المستخدمة من قبل الشركات كبير جدا. لذلك، مطلوب مستودع سير عمل لتخزين سير العمل. بشكل عام، سير عمل المستودع باستخدام ( RDBMS Relational Database Management System). بالإضافة إلى ذلك، البحث في سير عمل المستودع كبير مثل استخدام النص. بينما يكون سير العمل على شكل رسم بياني. استخدمت هذه الدراسة BPMN-Q للاستعلام عن المرئيات على سير عمل المستودع المستند إلى قاعدة بيانات الرسم البياني (NEO4J). يتم حفظ BPMN الأولى في تنسيق XPD L بحيث يمكن تحليلها. ثم تحليل XPD L. وعلاوة على ذلك، فإن البيانات من XPD L المخزنة في قاعدة بيانات الرسم البياني (NEO4J) ك DBMS. للبحث باستخدام BPMN-Q التي تم إنشاؤها بواسطة Microsoft visio 2007. سيتم حفظ استعلامات BPMN-Q بتنسيق VDX. ثم تحليلها وترجمتها إلى سايفر. يعالج Cypher ويقوم بإنشاء BPMN الذي يطابق نموذج القائمة. كما قدم هذا البحث مخطط قاعدة بيانات لتخزين سير العمل في شكل رسوم بيانية. أداء استخدام BPMN-Q في سير عمل المستودع المستند إلى قاعدة بيانات الرسم البياني ممتاز. يتم الحصول على الأداء من الدقة النسبية المئوية %98,75.

## BAB 1

### PENDAHULUAN

#### 1.1 Latar Belakang Masalah

*Workflow* adalah urutan tindakan atau langkah yang digunakan dalam suatu proses, yang biasanya dijalankan oleh lebih dari satu pihak dan menggunakan banyak sumber daya yang berbeda yang biasanya dibangkitkan dari serangkaian aturan pengerjaan (Palmer, 2009). Menurut Atmagi & Sarno (2014), alur kerja merupakan hal yang menarik dalam perkembangan teknologi SOA. Melalui alur kerja ini, pengguna dapat mengubah alur kerja sistem. Namun, jumlah alur kerja yang digunakan perusahaan cukup besar. Dengan cara ini, setiap modul mungkin memiliki alur kerja yang serupa / berbeda dengan alur kerja lainnya. Oleh karena itu, repositori alur kerja diperlukan untuk menyimpan alur kerja.

Pada umumnya *workflow repository* menggunakan RDBMS (*Relational Database Management System*). RDBMS menyimpan data yang sangat terstruktur dalam tabel dengan kolom yang telah ditentukan sebelumnya dari jenis tertentu dan banyak baris dari jenis informasi yang sama. Relasi antar tabel pada RDBMS dihubungkan dengan *primary key* dan *foreign key*. *Query JOIN* digunakan untuk menggabungkan dua *records* dari 2 tabel atau lebih. Namun *sql query* tidak dapat mengambil nama relasi. Selain itu RDBMS memiliki *schema* yang kaku. Hal ini akan menghambat proses pencocokan struktural. Karena *workflow* itu sendiri memiliki *schema* yang *flexible*.



Sehingga pada penelitian ini menggunakan *Business Process Models Query language* yang disebut BPMN-Q. BPMN-Q dapat mengekspresikan *query* struktural dan menentukan apakah model proses yang diberikan (grafik) secara struktural mirip dengan grafik permintaan. *Query* pada BPMN-Q berupa notasi-notasi yang diambil dari notasi pada BPMN itu sendiri. Maka BPMN-Q akan cocok digunakan untuk pencarian *workflow* melalui pendekatan *structural*. Selain itu *graph database* (NEO4J) akan digunakan sebagai media penyimpanan *workflow*.

Pada *graph database* mempunyai *Query* sendiri yang disebut *Cypher*. *Cypher* mampu memanggil nama relasi bahkan atribut dari relasi itu sendiri. Sehingga akan memudahkan pada saat pencarian. Selain itu *graph database* memiliki skema yang fleksibel.

## 1.2 Pernyataan Masalah

Berdasarkan latar belakang yang sudah dijelaskan pada bab sebelumnya, pertanyaan penelitian yang dapat diambil, sebagai berikut :

1. Bagaimana membuat *workflow repository* dengan *Graph Database*?
2. Bagaimana BPMN-Q diterjemahkan ke *Cypher* pada *Graph Database* (NEO4J)?
3. Seberapa akurat *BPMN-Q* dalam melakukan pencarian *workflow* ketika *Graph Database* diimplementasikan pada *workflow repository*?

## 1.3 Tujuan Penelitian

Tujuan penyusunan penelitian sebagai berikut:

1. Membuat *workflow repository* dengan *graph database* (NEO4J).

2. Menerjemahkan BPMN-Q ke *Cypher* pada *Graph Database* (NEO4J)
3. Mengukur akurasi BPMN-Q dalam melakukan pencarian *workflow* ketika *Graph Database* diimplementasikan pada *workflow repository*.

#### **1.4 Manfaat Penelitian**

Manfaat yang dapat diambil adalah :

1. Membantu penyusunan skema penyimpanan *workflow* pada *graph database* (NEO4J).
2. Membantu pencarian *workflow* pada *workflow repository*.

#### **1.5 Batasan Masalah**

Batasan masalah dibuat oleh peneliti untuk memfokuskan penelitian pada masalah yang ada, Berikut batasan masalah pada penelitian ini:

1. Data *workflow* merupakan data *dummy*.
2. Format file *workflow* berupa XPDL.
3. Penyimpanan *workflow* dengan *Graph database* (NEO4J).
4. Metode *BPMN-Q* digunakan dalam *workflow discovery*.
5. Menggunakan pendekatan *structural similarity* pada proses pencarian.

## **BAB 2**

### **STUDI PUSTAKA**

#### **2.1 *Workflow***

*Workflow* adalah urutan tindakan atau langkah yang digunakan dalam suatu proses, yang biasanya dijalankan oleh lebih dari satu pihak dan menggunakan banyak sumber daya berbeda yang dibangkitkan dari serangkaian aturan pengerjaan (Palmer, 2009).

#### **2.2 *Workflow Management System (WfMS)***

*Workflow management coalition* (WFMC) adalah sebuah organisasi *non-profit* internasional untuk *workflow*. WFMC mendeskripsikan *workflow* sebagai otomatisasi keseluruhan atau sebagian dari proses bisnis. Dalam proses ini dokumen, informasi atau tugas telah diteruskan dari satu partisipan ke partisipan yang berbeda melalui operasi yang sudah ditentukan sesuai dengan seperangkat aturan prosedural. Saat digunakan dalam berbagai skenario alur kerja, definisi ini masih sangat umum (Palmer, 2009).

WfMC mendeskripsikan *Workflow Management System* (WFMS) sebagai sistem yang menginterpretasi, membuat, dan mengatur eksekusi alur kerja dengan menggunakan perangkat lunak yang berjalan pada satu atau beberapa mesin alur kerja. Perangkat lunak ini dapat menguraikan deskripsi proses dan berkorelasi dengan tugas yang berbeda saat dibutuhkan. Chang (2016) menambahkan bahwa definisi alur kerja dan WfMS terlihat menyerupai dengan pengertian BPM dan BPMS. Faktanya, kegunaan WfMS sama dengan sebagian elemen BPMS. Sebagaimana BPMS, WfMS

menyediakan fasilitas untuk menggabungkan pengguna dan aplikasi sehingga bisa berperan serta pada sistem tersebut

### **2.3 *Workflow Repository***

*Workflow repository* adalah sistem yang menyimpan dan menjelaskan alur kerja rangkaian langkah bisnis di repositori. Menurut Chengfei Liu et al. (2003), fungsi repositori adalah untuk mendistribusikan data pada database yang menyimpan alur kerja yang dibuat dan digunakan oleh perusahaan. Manajer repositori menerapkan lapisan layanan kontrol untuk pemodelan, pengambilan dan pengelolaan objek dalam repositori di DBMS. Selain penyimpanan, alur kerja yang ada di repositori juga bisa diakses lagi untuk membuat alur kerja baru. Pencarian pada *workflow repository* harus berguna dan efektif.

Pada penelitian Yan et al. (2012), semangat diperoleh melalui pengembangan sistem basis data informasi BPM. Pertama, basis data informasi menyediakan ruang penyimpanan dan menyederhanakan pencarian proses bisnis. Repositori ini kemudian dapat berfungsi untuk membentuk ulang model proses khusus industri. Penelitian mereka juga berfokus pada prosedur penggunaan kembali, seperti preset dalam pilihan komposisi dan penggunaan ontologi untuk menyesuaikan antar proses.

### **2.4 BPMN**

*Business Process Management Initiative* (BPMI) mengembangkan metode mutakhir yang disebut *Business Process Management Notation* (BPMN). BPMN merupakan tolak ukur baru dalam penyusunan prosedur bisnis dan berfungsi sebagai alat desain sistem yang canggih (White, 2004).

Sasaran utama penyusunan BPMN ini yaitu untuk memberikan notasi yang gampang dipahami, yang dapat dipahami oleh seluruh pemakai bisnis mulai dari analis bisnis yang membuat rencana awal proses hingga *developer* teknis yang berkewajiban untuk mengimplementasikan teknologi tersebut. Bahkan untuk para pebisnis yang akan mengatur dan memantau proses tersebut.

Aturan penotasian proses bisnis sebagai berikut:

1. Kejadian-kejadian dimodelkan dari awal proses hingga akhir arus proses.
2. Keputusan bisnis atau arus cabang dinotasikan oleh *gateway*. *Gateway* ini sama dengan simbol keputusan di diagram alur.
3. Proses dalam alur kerja dapat berisi sub-proses, yang dapat direpresentasikan secara ilustratif oleh diagram proses bisnis (BPD) lain, yang dihubungkan ke simbol proses oleh *hyperlink*.
4. Jika langkah tidak ditugaskan ke *sub*-proses, maka merupakan tingkat proses yang paling rendah.
5. Simbol "+" pada ikon proses menandakan bahwa langkah telah diuraikan, apabila tidak terdapat simbol "+", langkah tersebut artinya tugas.

BPMN juga mendeskripsikan *Business Process Diagram* (BPD) berdasarkan teknologi diagram alur, dan kemudian menyesuaikannya untuk mengimplementasikan pola grafis mekanisme proses bisnis. Kemudian, pola proses bisnis adalah hubungan entitas grafis yang mewakili aktivitas atau mendefinisikan kegiatan dan pengendalian proses dari pesan aktivitas. Menurut Sari & Asniar (2015) definisi empat notasi tersebut adalah:

1. Notasi *Event*

*Event* diwakili oleh lingkaran dan merupakan kejadian dalam langkah bisnis. *Event* berpengaruh pada langkah dan umumnya mempunyai sebab (penyulut) atau efek (akibat). Perlakuan berbeda antara pemicu atau hasil yang berlainan pada penanda internal menggunakan Event lingkaran inti terbuka. Menurut waktu dampak *event* pada aliran, *event* dibagi menjadi tiga jenis: mulai, tengah, dan akhir. Simbol notasi *event* ada pada Gambar 2.1 berikut :

Types	Start			Intermediate			End	
	Top-Level	Event Sub-Process interrupting	Event Sub-Process Non-interrupting	Catching	Boundary Interrupting	Boundary Non-Interrupting		Throwing
None								
Message								
Timer								
Error								
Escalation								
Cancel								
Compensation								
Conditional								
Link								
Signal								
Terminate								
Multiple								
Parallel Multiple								

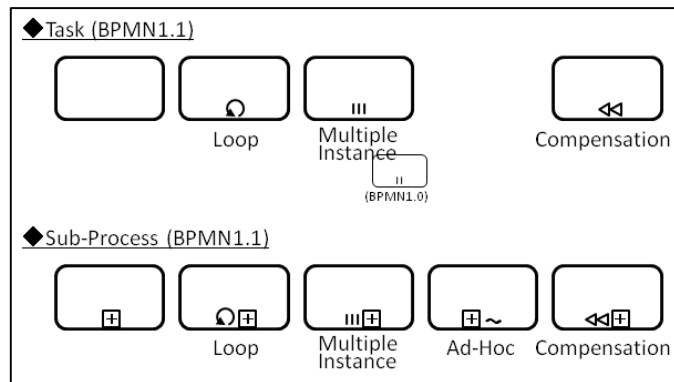
Gambar 2.1 Notasi *Event*

## 2. Notasi *Activity*

*Activity*(tugas) tersebut dinotasikan oleh persegi panjang bersudut lengkung, yang melambangkan sebutan bagi kegiatan yang dilakukan oleh perusahaan/organisasi. *Activity* bisa berbentuk atom atau non-atom (kombinasi). Macam-macam *Activity* yaitu: tugas dan *sub* proses. *Sub-*

proses dibedakan dengan simbol *plus* kecil pada tengah bawah simbol.

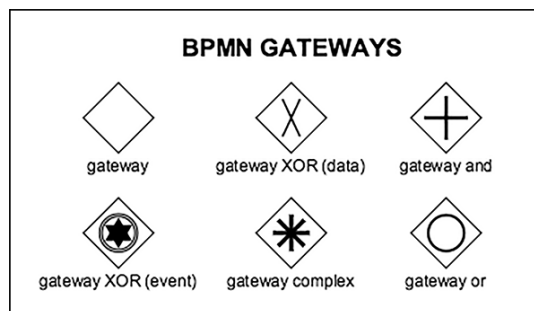
Gambar mengenai notasi *activity* ada pada Gambar 2.2.



Gambar 2.2 Notasi Activity

### 3. Notasi Gateway

Gateway dinotasikan oleh *diamond* yang sudah dikenal dan berfungsi untuk mengontrol divergensi dan konvergensi aliran urutan. Oleh karena itu, ini akan memastikan pengambilan keputusan tradisional, percabangan, penggabungan, dan jalur penggabungan. *Internal marker* akan menampilkan bentuk kontrol perilaku seperti pada Gambar 2.3

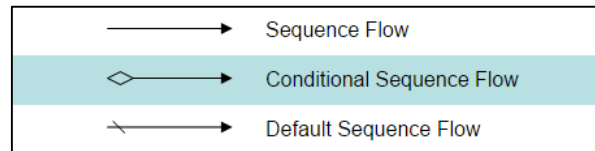


Gambar 2.3 Notasi Gateway

### 4. Penghubung

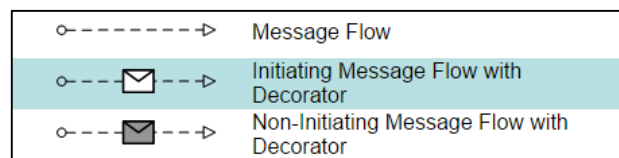
Notasi penghubung terdiri dari tiga jenis yaitu:

- a. *Sequence flow*, menentukan *activity* selanjutnya adalah kelanjutan *activity* sebelumnya. Gambar 2.4 menampilkan rupa notasi *sequence flow*.



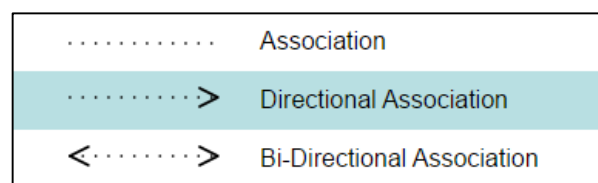
Gambar 2.4 Notasi *Sequence Flow*

- b. *Message flow*, menentukan arus pesan dimana proses yang dituju bukan kesinambungan langkah sebelumnya. Gambar 2.5 menampilkan notasi *message flow*.



Gambar 2.5 Notasi *Message Flow*

- c. *Association*, menentukan kaitan antara dua kegiatan. Gambar 2.6 menampilkan notasi *association*.

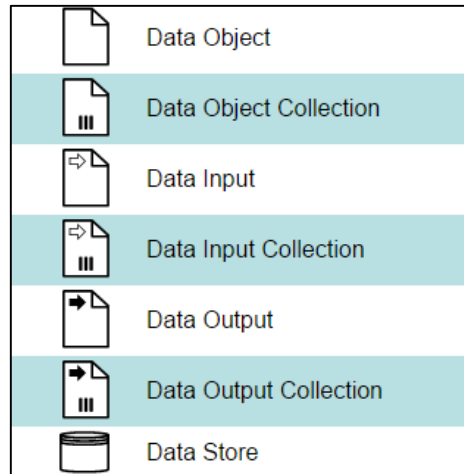


Gambar 2.6 Notasi *Association*

## 5. Data

Notasi data adalah untuk mengidentifikasi komponen grafik tertentu secara visual. Menjelaskan operasi yang telah selesai, menjelaskan simbolnya, dan tipe yang digunakan. Notasi data juga menunjukkan kaitan antar komponen grafik. Rupa notasi data ditampilkan pada Gambar 2.7



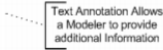




Gambar 2.7 Notasi Data

## 6. Artifak

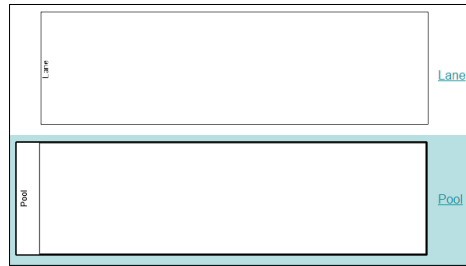
BPMN dipersiapkan untuk mendukung alat pemodelan dan pola yang luwes untuk memberikan varian simbol dasar dan menyediakan kemampuan untuk menggambarkan kondisi pemodelan tertentu. Sejumlah artefak bisa diimbuhkan ke bagan agar sesuai dengan konteks langkah bisnis yang akan dibuat. Notasi artefak pada Gambar 2.8

Notasi	Keterangan
	<b>Data Object</b> Data objek adalah mekanisme untuk menunjukkan bagaimana data yang diperlukan atau dihasilkan oleh aktivitas. Dihubungkan menggunakan asosiasi
	<b>Group</b> group ini dapat digunakan untuk keperluan dokumentasi atau analisis, tapi tidak mempengaruhi sequential flow.
	<b>Annotation</b> Annotation adalah mekanisme untuk modeler untuk memberikan informasi teks tambahan untuk pembaca

Gambar 2.8 Notasi Artifak

## 7. Swimlanes

*Swimlanes* berfungsi untuk menggambarkan kemampuan fungsional yang berlainan. Merujuk pada Gambar 2.9



Gambar 2.9 Notasi *Swimmer*

## 2.5 XML Process Definition Language (XPDL)

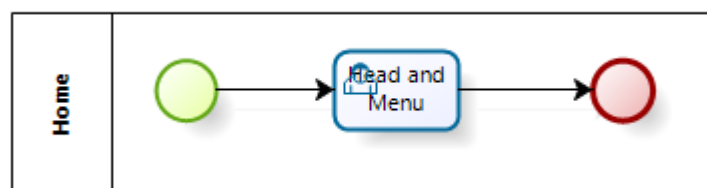
XML *Process Definition Language* adalah pola dasar yang disusun oleh WfMC untuk bertukar uraian langkah bisnis antara alur kerja yang berlainan (yaitu, antara alat pembuat model dan rangkaian manajemen yang berlainan). XPDL mendeskripsikan kerangka kerja melalui XML untuk memastikan bagian deklaratif dari alur kerja bisnis.

XPDL ditujukan untuk memudahkan bertukar bagan BPMN, termasuk grafik dan semantik alur kerja langkah bisnis. XPDL ini adalah format utama pada pertukaran bagan BPMN dan dipersiapkan tersendiri untuk menyimpan seluruh komponen BPMN. Selain itu XPDL bisa disebut serialisasi XML dari BPMN. XPDL berisi elemen yang digunakan untuk menyimpan keterangan grafis, seperti lokasi node X dan Y, dan bagian yang akan digunakan untuk mengoperasikan langkah. Menurut Handayani et al. (2009) terdapat 6 komponen pada XPDL, yakni:

1. Komponen *<Package>* adalah paket yang mendeskripsikan proses bisnis. Konsepsi paket berfungsi untuk membatasi definisi yang berlebihan yang digunakan dalam banyak proses.
2. Komponen *<WorkflowProcess>* adalah komponen yang berfungsi untuk menentukan *sub*-proses dari langkah tertentu.

3. Komponen *<ActivitySet>* menampilkan gabungan beragam aktivitas dan peralihan terautnya.
4. Fungsi Komponen *<BlockActivity>* menjalankan komponen *<ActivitySet>*.
5. Komponen *<Participant>* digunakan untuk menentukan peserta dalam alur kerja, seperti objek yang dapat melakukan aktivitas.
6. Komponen *<DataType>* dan *<DataField>* berfungsi untuk menentukan data yang terkait dengan alur kerja. Data ini digunakan untuk pengambilan keputusan atau data referensi di luar alur kerja dan ditransmisikan ke seluruh aktivitas dan *sub*-alur.

Salah satu kelebihan menggunakan XPDL yaitu proses uraian XML yang terdapat pada XPDL dapat diubah menjadi bentuk grafik, sehingga memudahkan dan mempercepat pengguna dalam memahami proses bisnis organisasi. Bentuk BPMN akan dijelaskan pada Gambar 2.10 dan kode xpdL pada header BPMN Kode Sumber 2.1.



Gambar 2.10 Pemodelan BPMN

```

<?xml version="1.0" encoding="utf-8"?>
<Package      xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
Id="75825e66-ba88-4360-a071-83658958bcf4" Name="Diagram 1"
xmlns="http://www.wfmc.org/2009/XPDL2.2">
<PackageHeader>
<XPDLVersion>2.2</XPDLVersion>
<Vendor>Bizagi Process Modeler.</Vendor>
<Created>2016-11-05T00:39:01.8247066+07:00</Created>
      <ModificationDate>2017-03-
      02T19:12:05.0499293+07:00</ModificationDate>
<Description>Diagram 1</Description>
<Documentation />
</PackageHeader>
<RedefinableHeader>
<Author>navIras</Author>
<Version>1.0</Version>
<Countrykey>CO</Countrykey>
</RedefinableHeader>
<ExternalPackages />
<DataStores>

```

Kode Sumber 2.1 Kode XPDL dari *header* BPMN

## 2.6 *Graph Database*

*Graph Database Management System* (selanjutnya disebut basis data grafik) adalah sistem manajemen basis data online dengan pola penyimpanan data berupa grafik. Basis data grafik biasanya digunakan dengan sistem transaksional (OLTP). Dengan cara ini, lazimnya dimaksimalkan untuk kemampuan transaksi dan memiliki desain untuk transaksi dan integritas operasional (Robinson et al., 2014). *Graph Database* adalah teknologi basis data yang metode pencadangan datanya mirip dengan grafik yang terbuat dari gabungan *edge* dan *vertices* yang bisa langsung diakses memakai aplikasi dan mendukung OLTP (Robinson et al., 2014).

*Graph Database* menggunakan penyimpanan *database native* yang dirancang khusus untuk skalabilitas dan kinerja, tetapi masih terbatas karena kurangnya pemahaman tentang konsep dan aplikasinya. Ini berbeda dari *database non-native* dan *database non-Grafik* seperti sistem manajemen database relasional (RDBMS), yang mempunyai fitur bahasa kueri yang dapat dipahami dengan gampang oleh sebagian besar *developer*. Dalam bahasa lain, *database grafik* memiliki kinerja yang lebih baik, tetapi lebih sulit untuk memahami kueri.

NEO4J merupakan aplikasi database yang menggunakan graph database. NEO4J ini bekerja dengan cara menyimpan data ke *node* yang terhubung dengan *relationship* (Dharmawan & Sarno, 2017).


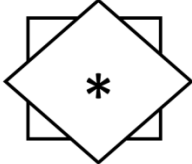
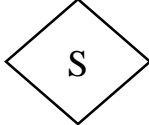
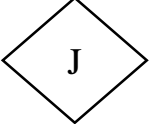
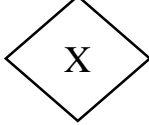
## 2.7 BPMN-Q

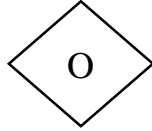
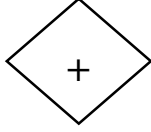
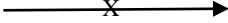
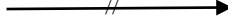

Ada tiga macam informasi spesifik yang dapat diperoleh dari *business process models*, yaitu Struktur grafis, Semantik perilaku, dan Semantik operasi. Jadi berdasarkan informasi spesifik diatas, *Business process models* dapat dianggap sebagai sejenis data grafik dengan semantik perilaku dan operasi. Berdasarkan asumsi ini, Wang et al. (2014) membagi *query* model proses bisnis menjadi lima jenis yaitu, *query* tepat berdasarkan struktur grafik, *query* kesamaan berdasarkan struktur grafik, *query* persis berdasarkan semantik perilaku, *query* kesamaan berdasarkan perilaku semantik, *query* berdasarkan semantik operasi. Dalam penelitian ini, jenis *business process models* yang dipakai adalah *query* kesamaan berdasarkan struktur grafik

BPMN-Q adalah *query* visual yang didasarkan pada notasi BPMN. *Query* BPMN-Q dirancang untuk membantu pemodelan proses bisnis pada

permintaan *repository* model proses bisnis berbasis grafik. *Query* ini bergantung pada notasi BPMN sebagai sintaks (Awad, 2007). Notasi BPMN-Q dan penjelasannya terdapat pada tabel 2.1.

Tabel 2.1 *Query* BPMN-Q

NAMA	KETERANGAN	NOTASI
Variabel <i>Node</i>	Ini digunakan untuk menunjukan <i>activity</i> yang tidak diketahui dalam <i>query</i> . Ini menyerupai <i>Activity</i> hanya dibedakan dengan “@” di awal label	
<i>Generic Node</i>	Ini menunjukkan <i>node</i> yang tidak dikenal dalam suatu proses. Bisa mengevaluasi ke tipe <i>node</i> apa pun.	
<i>Generic Split</i>	<i>split gateway</i>	
<i>Generic Join</i>	<i>Join gateway</i>	
<i>XOR</i>	<i>XOR gateway</i>	

NAMA	KETERANGAN	NOTASI
<i>OR</i>	<i>OR gateway</i>	
<i>AND</i>	<i>AND gateway</i>	
<i>Negative Sequence Flow</i>	Ini menyatakan bahwa dua <i>node</i> A dan B tidak terkait langsung dengan <i>sequence flow</i>	
<i>Path</i>	Ini menyatakan bahwa harus ada jalur dari A ke B.	
<i>Negative Path</i>	Ini menyatakan bahwa tidak ada jalan antara dua <i>node</i> A dan B.	

## 2.8 Penelitian Terkait

Awad (2007) memperkenalkan bahasa *query* visual baru untuk *business* proses yang bernama BPMN-Q. Dalam penelitian ini menggunakan RDBMS yakni SQL. Ter Hofstede et al. (2017) menggunakan BPMN-Q dengan *eTVSM ontology*. Namun tidak bisa memproses *set query* yang besar.

Dominguez-Sal et al. (2010) mengevaluasi kinerja dari empat proyek *graph database* asli yang paling skalabel (NEO4J, Jena, HypergraphDB dan DEX). Mereka menerapkan tolak ukur Analisis Grafik HPC Lengkap, dan menguji kinerja setiap database untuk operasi grafik dan ukuran grafik yang berbeda-beda, menunjukkan bahwa dalam status pengembangan (NEO4J,

Jena, HypergraphDB dan DEX) saat ini, DEX dan NEO4J adalah basis data grafik yang paling tepat guna.

Pada penelitian yang dikerjakan oleh Ardani et al. (2019) implementasi *graph database* dengan metode *Weighted Directed Acyclic Graph* (WDAG) digunakan dalam penentuan rute transportasi umum. Data lintasan trayek angkot di Kota Malang dan Kota Batu disimpan dalam *Graph Database*. Titik pemberhentian yang dilalui disimbolkan menjadi *node*. Sedangkan jalur atau arah tujuan angkot melaju disimbolkan menjadi relasi atau *edge*.

Dalam penelitian yang dilakukan oleh Abriani & Yaqin (2019) metode *semantic similarity* digunakan untuk mengukur kemiripan makna antar kalimat. Pada tahap pertama, dilakukan penghitungan *word similarity* menggunakan pendekatan *wu palmer*, *lin*, dan *path* menggunakan WS4J dengan melibatkan tiga kriteria tingkat kemiripan yaitu mirip, sedang, dan tidak mirip. Setelah didapat nilai *word similarity*, kemudian dilakukan perhitungan *sentence similarity* dengan dua jenis bobot nilai yaitu (0,75, 0,25) dan (0,25, 0,75)

Penelitian yang dilakukan oleh Wahyudi & Akbar (2019) yaitu tentang menyalurkan dan menyimpan basis informasi ke dalam kumpulan data bagan. Basis data graf NEO4J dipilih karena menggunakan pendekatan dan pemrosesan graf yang *native*. Basis informasi Yago dipilih dengan alasan mempunyai akurasi Yago sangat baik, yaitu diatas 95%. Ekstraksi basis pengetahuan ke dalam kumpulan data grafik dilakukan menggunakan perhitungan GPE karena data yang dimasukkan banyak. Untuk membuang pengulangan data dan membuat keterkaitan data lebih mudah didapat.



basisdata graf yang dibuat setelah dicoba dengan beberapa *query* memberikan hasil sesuai harapan. Pada penelitian yang dilaksanakan menawarkan penggunaan metode BPMN-Q pada *graph database*. Hal ini diharapkan mempercepat proses *query* pada *workflow* dan menghasilkan akurasi yang presisi.

## **BAB 3**

### **DESAIN DAN IMPLEMENTASI**

#### **3.1 Desain Penelitian**

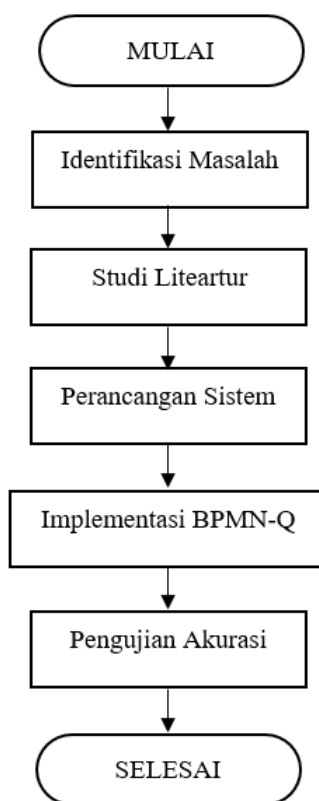
Rancangan penelitian menggambarkan proses dan metode penelitian yang akan dilakukan. Meliputi jenis penelitian, mekanisme penelitian dan metode pengerjaan datanya. Berikut pembedaan dari masing-masing mekanisme penelitian.

##### **3.1.1 Tipe Penelitian**

Penelitian ini memakai metode kuantitatif berdasarkan jumlah atau tingkat akurasi yang diimplementasikan pada sistem repositori *workflow* menggunakan BPMN-Q untuk pengambilan data.

##### **3.1.2 Prosedur Penelitian**

Prosedur penelitian menjelaskan tentang langkah-langkah yang akan digunakan dalam pengumpulan data hingga tahapan penelitian. Prosedur penelitian akan direpresentasikan dalam Gambar 3.1 berikut



Gambar 3.1 Prosedur Penelitian

Menurut Gambar 3.1 suatu proses harus diselesaikan untuk menyelesaikan studi ini. Penelitian dimulai dengan studi literatur tentang alur kerja. Selain itu perlu juga dilakukan penelitian literatur tentang konsep BPMN-Q. Tahapan selanjutnya adalah menentukan masalah dengan mengidentifikasi masalah penelitian, kemudian menyusun desain sistem untuk menentukan tahapan sistem produksi. Langkah selanjutnya adalah melakukan pengujian sistem dan analisis *usability* berdasarkan hasil pengujian.

### 3.1.3 Identifikasi Masalah

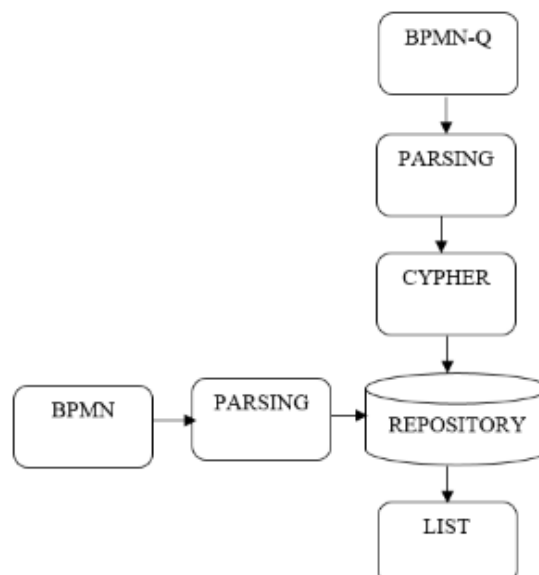
Pembeberan terkait dengan rumusan masalah sebelumnya sudah dibahas di BAB I.

### 3.1.4 Perancangan Sistem

Pada perancangan sistem ini menjelaskan alur kerja sistem yang akan digunakan dalam penelitian.

Alur kerja sistem dalam penelitian ini dimulai dari BPMN. BPMN berupa file XPDL. Kemudian diparsing untuk mengambil datanya. Data yang sudah diambil akan disimpan pada *database* NEO4J (*Graph Database*).

Selanjutnya penggunaan BPMN-Q untuk pencarian BPMN yang telah tersimpan. Dimulai dari pembuatan BPMN-Q menggunakan *microsoft visio 2007* dan disimpan dengan berformat *.VDX*. Setelah itu *file .VDX* tadi diparsing untuk diambil datanya. Kemudian menerjemahkan data yang telah diambil ke *cypher*. *Cypher* akan memproses pencarian BPMN pada NEO4J. Hasil dari proses pencarian akan ditampilkan berupa list BPMN yang sesuai. Alur ada pada gambar 3.2.



Gambar 3.2 Alur Kerja Sistem

### 3.1.4.1 BPMN

BPMN yang akan disimpan dalam *repository* berformat \*.xpdL. *file* BPMN \*.xpdL akan diparsing untuk diambil datanya. Data yang diambil akan dijelaskan pada proses *parsing*. Contoh *file* XPDL pada kode sumber 3.1.

```
<?xml version="1.0" encoding="utf-8"?>
<Package      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"      Id="75825e66-ba88-
4360-a071-83658958bcf4"      Name="Diagram      1"
xmlns="http://www.wfmc.org/2009/XPDL2.2">
<PackageHeader>
<XPDLVersion>2.2</XPDLVersion>
<Vendor>Bizagi Process Modeler.</Vendor>
<Created>2016-11-05T00:39:01.8247066+07:00</Created>
<ModificationDate>2017-03-
02T19:12:05.0499293+07:00</ModificationDate>
<Description>Diagram 1</Description>
```

Kode Sumber 3.1 XPDL

### 3.1.4.2 Parsing (BPMN)

*Parsing* (BPMN) adalah proses pengambilan data dari *file* \*.xpdL. Data yang akan diambil sebagai berikut :

### a. Deskripsi

Data pada deskripsi yang akan diambil yaitu *author*, *version*, *created* dan *modificationdate*. Kode xml deskripsi ada pada kode sumber 3.2.

```
<?xml version="1.0" encoding="utf-8"?>
<Package xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
Id="8e0c3517-6ac1-4325-aabb-6dfe68aa9fae"
Name="Diagram 1" xmlns="http://www.wfmc.org/2009/XPDL2.2">
  <PackageHeader>
    <XPDLVersion>2.2</XPDLVersion>
    <Vendor>Bizagi Process Modeler.</Vendor>
    <Created>2016-11-14T17:51:36.1406269+07:00</Created>
    <ModificationDate>2016-11-16T04:49:05.9378539+07:00</ModificationDate>
    <Description>Diagram 1</Description>
    <Documentation />
  </PackageHeader>
  <RedefinableHeader>
    <Author>MOY.CastleClash</Author>
    <Version>1.0</Version>
    <Countrykey>C0</Countrykey>
  </RedefinableHeader>
  <ExternalPackages />
```

Kode Sumber 3.2 Deskripsi

### b. Activity

Data pada *activity* yang akan diambil yaitu *Id* dan *Name*. Kode xml *activity* ada pada kode sumber 3.3.

```
<Activities>
  <Activity Id="f4599dc2-e4d2-4fe9-90ff-83830dd9261e" Name="">
    <Description />
    <Event>
      <StartEvent Trigger="None" />
    </Event>
    <Documentation />
    <NodeGraphicsInfos>
      <NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="30" Width="30"
        BorderColor="-10311914" FillColor="-1638505">
        <Coordinates XCoordinate="119" YCoordinate="65" />
      </NodeGraphicsInfo>
    </NodeGraphicsInfos>
    <ExtendedAttributes>
      <ExtendedAttribute Name="RuntimeProperties" value="{}" />
    </ExtendedAttributes>
  </Activity>
```

Kode Sumber 3.3 Activity

### c. Data Object

Data pada data *object* yang akan diambil yaitu Id dan *Name*.

Kode xml data *object* ada pada kode sumber 3.4.

```
<DataObjects>
  <DataObject Id="db5db842-34c0-41ff-a8ce-78986d7dce44" Name="Data Kesantrian">
    <object>
      <Documentation />
    </object>
    <NodeGraphicsInfos>
      <NodeGraphicsInfo ToolId="BizAgi_Process_Modeler" Height="50" Width="40"
        BorderColor="-10066330" FillColor="-986896">
        <Coordinates XCoordinate="153" YCoordinate="844" />
      </NodeGraphicsInfo>
    </NodeGraphicsInfos>
    <DataField />
    <ExtendedAttributes />
  </DataObject>
</DataObjects>
```

Kode Sumber 3.4 Data *Object*

#### d. *Flow*

Data pada *flow* yang akan diambil yaitu Id, *From* dan *To*.

Kode xml *flow* ada pada kode sumber 3.5

```
<Transitions>
  <Transition Id="25e102a1-b126-4c37-af56-bdeb6c6ff246" From="f4599dc2-e4d2-4fe9-90ff-83830dd9261e"
    To="9183ad4c-a125-4e1c-a665-c79d5202a694">
    <condition />
    <Description />
    <ConnectorGraphicsInfos>
      <ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="-16777216">
        <Coordinates XCoordinate="149" YCoordinate="80" />
        <Coordinates XCoordinate="195" YCoordinate="80" />
      </ConnectorGraphicsInfo>
    </ConnectorGraphicsInfos>
    <ExtendedAttributes />
  </Transition>
</Transitions>
```

Kode Sumber 3.5 *Flow*

#### e. *Flow Data*

Data pada *flow* data yang akan diambil yaitu Id, *From* dan *To*.

Kode xml *flow* data ada pada kode sumber 3.6.

```
<DataAssociations>
  <DataAssociation Id="e21d1f8b-c043-44cb-9360-9c6de7fa03a2" From="2f0bd9f4-44b2-42dc-9764-832ed5111fe8"
    To="9f80d5df-aa68-4aa7-a968-9b9986aacb41">
    <Description />
    <ConnectorGraphicsInfos>
      <ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="-16777216" />
    </ConnectorGraphicsInfos>
    <ExtendedAttributes />
  </DataAssociation>
</DataAssociations>
```

Kode Sumber 3.6 *Flow Data*

#### f. Data Store

Data pada *data store* yang akan diambil yaitu Id, *Source* dan

*Target*. Kode xml *data store* ada pada kode sumber 3.7.

```
<DataStores>
  <DataStore Id="62b3e5cd-d8d5-400c-b6b6-6999caabdac6" Name="" IsUnlimited="true">
    <Object>
      <Documentation />
    </Object>
  </DataStore>
```

Kode Sumber 3.7 Data Store

#### g. Message Flow

Data pada *message flow* yang akan diambil yaitu Id dan *Name*.

Kode xml *message flow* ada pada kode sumber 3.8.

```
<MessageFlows>
  <MessageFlow Id="93192374-8283-414d-a40f-033d621fbb13"
    Source="60c3f02f-58b3-47fc-832e-195ce0bb0c33" Target="abc dce9-d078-4f02-b0d9-2d04c840d79d">
    <ConnectorGraphicsInfos>
      <ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="-16777216">
        <Coordinates XCoordinate="650" YCoordinate="264" />
        <Coordinates XCoordinate="769" YCoordinate="264" />
        <Coordinates XCoordinate="769" YCoordinate="1064" />
      </ConnectorGraphicsInfo>
    </ConnectorGraphicsInfos>
    <ExtendedAttributes />
  </MessageFlow>
```

Kode Sumber 3.8 Message Flow

#### h. Association

Data pada *association* yang akan diambil yaitu Id, *Source* dan

*Target*. Kode xml *association* ada pada kode sumber 3.9.

```
<Associations>
  <Association Id="580e314c-c808-4da4-b5fb-79153059b394"
    Source="431eb629-eae3-4f7c-83e9-2353e5e1db4d" Target="db5db842-34c0-41ff-a8ce-78986d7dce44" Name="">
    <ConnectorGraphicsInfos>
      <ConnectorGraphicsInfo ToolId="BizAgi_Process_Modeler" BorderColor="-16777216">
        <Coordinates XCoordinate="171" YCoordinate="943" />
        <Coordinates XCoordinate="171" YCoordinate="926" />
        <Coordinates XCoordinate="173" YCoordinate="926" />
        <Coordinates XCoordinate="173" YCoordinate="894" />
      </ConnectorGraphicsInfo>
    </ConnectorGraphicsInfos>
    <ExtendedAttributes />
  </Association>
```

Kode Sumber 3.9 Association

### 3.1.4.3 Workflow repository

*Workflow Repository* adalah tempat penyimpanan *workflow*.

*Repository* menggunakan *graph database* (NEO4J). Skema penyimpanan

*workflow* pada *graph database* akan dijelaskan sebagai berikut:



Tabel 3.1 Skema Notasi BPMN dalam *Graph Database* (NEO4J)

<b>Node/Edge</b>	<b>Label</b>	<b>Properti</b>	<b>Notasi BPMN</b>
<i>Node</i>	:Deskripsi	- Penulis - Versi	-
<i>Edge</i>	:DIBUAT	- tgl_pembuatan - tgl_modifikasi	-
<i>Node</i>	<i>Event</i>	- tipe	Notasi <i>Event</i>
<i>Node</i>	<i>Activity</i>	- nama - jenis	Notasi <i>Activity</i>
<i>Node</i>	<i>Gateway</i>	- jenis	Notasi <i>gateway</i>
<i>Edge</i>	Sesuai dengan jenis relasinya. Misal: - <i>sequence flow</i> = <i>SEQUENCE</i>	-	Notasi <i>flow</i> (relasi)
<i>Node</i>	Data	- nama - tipe	Notasi data

Pada Tabel 3.1 dijelaskan bagaimana bentuk skema *workflow* yang disimpan dalam graph database. *Node* pertama akan dibuat selalu berlabel deskripsi. Ini bertujuan untuk menyimpan deskripsi dari sebuah *workflow*. Kemudian dilanjutkan dengan *edge* yang diberi label: DIBUAT. Dan memiliki *property* : tgl\_pembuatan, tgl\_modifikasi. Maka deskripsi dengan *workflow* akan tetap terhubung, supaya saat pencarian dapat dipanggil juga. Selanjutnya *node* dan *edge* akan mengikuti notasi-notasi pada *workflow*.

### 3.1.4.4 BPMN-Q

Proses ini adalah menyusun query untuk pencarian *workflow*.  
*Query* ini nantinya diterjemahkan ke Bahasa *Cypher*. Penyusunan *query* menggunakan *tools* Microsoft Visio.

### 3.1.4.5 Parsing (BPMN-Q)

*Parsing* (BPMN-Q) adalah proses pengambilan data dari *file* BPMN-Q yang berupa \*.vdx. Data yang akan diambil sebagai berikut :

#### a) Shape

Data pada *shape* yang akan diambil yaitu Id dan *Master*.

Kode xml *shape* ada pada kode sumber 3.10.

```
<Shapes>
  <Shape ID='1' NameU='Start' Type='Shape' Master='0'>
    ...
  </Shape>
</Shapes>
```

Kode Sumber 3.10 *Shape*

#### b) Connect

Data pada *connect* yang akan diambil yaitu *FronSheet* dan

*ToSheet*. Kode xml *connect* ada pada kode sumber 3.11

```
<Connects>
  <Connect FromSheet='2' FromCell='BeginX' FromPart='9' ToSheet='1' ToCell='Connections.X2' ToPart='101' />
  <Connect FromSheet='2' FromCell='EndX' FromPart='12' ToSheet='4' ToCell='Connections.X4' ToPart='103' />
  <Connect FromSheet='8' FromCell='BeginX' FromPart='9' ToSheet='4' ToCell='Connections.X1' ToPart='100' />
  <Connect FromSheet='8' FromCell='EndX' FromPart='12' ToSheet='10' ToCell='Connections.X2' ToPart='101' />
  <Connect FromSheet='9' FromCell='BeginX' FromPart='9' ToSheet='4' ToCell='Connections.X2' ToPart='101' />
  <Connect FromSheet='9' FromCell='EndX' FromPart='12' ToSheet='11' ToCell='Connections.X2' ToPart='101' />
</Connects>
```

Kode Sumber 3.11 *Connect*

#### c) Master

Data pada *master* yang akan diambil yaitu *Name*. Kode xml

*master* ada pada kode sumber 3.12.

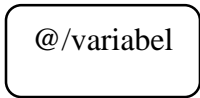
```
<Masters>
  <Master ID='2' NameU='Path' Name='Path' Prompt="" IconSize='1' AlignName='2' MatchByName='0'
  IconUpdate='1' UniqueID='{315D6DA4-0003-0000-8E40-00608CF305B2}' BaseID='{C32DD7B7-C894-4085-
  8988-C28AB0F69116}' PatternFlags='0' Hidden='0'>
    <PageSheet LineStyle='0' FillStyle='0' TextStyle='0'>...
  </PageSheet>
  <Shapes>
    <Shape ID='5' NameU='Path.35' Type='Shape' LineStyle='6' FillStyle='6'
  TextStyle='6'>...
  </Shape>
  </Shapes>
  <Icon>...
  </Icon>
  </Master>
</Masters>
```

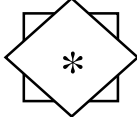
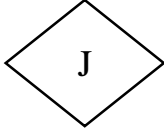
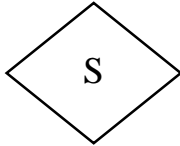
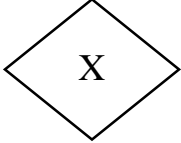
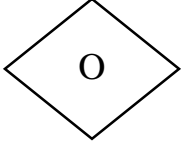
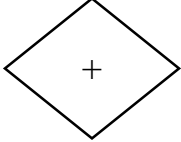
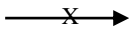
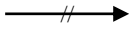
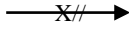
Kode Sumber 3.12 *Master*

#### 3.1.4.6 *Cypher*

*Query* BPMN-Q diparsing terlebih dahulu sebelum diterjemahkan ke *cypher*. Setelah *parsing* selesai BPMN-Q diterjemahkan ke bahasa *cypher*. Notasi BPMN-Q yang diterjemahkan ke *cypher* berdasarkan tabel 3.2.

Tabel 3.2 Notasi BPMN-Q ke *Cypher*

Notasi BPMN-Q	<i>Cypher</i>
	(a:activity{name:"variabel"}) = variabel (a:activity) = @

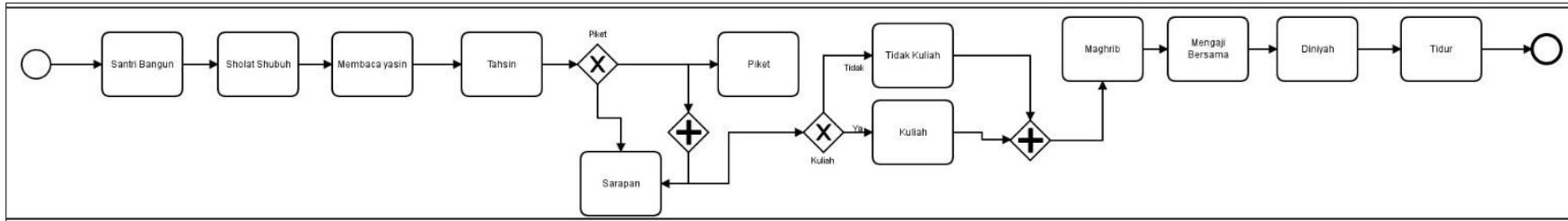
Notasi BPMN-Q	<i>Cypher</i>
	(e:event)
	(e:event{tipe:"join"})
	(e:event{tipe:"split"})
	(e:event{tipe:"xor"})
	(e:event{tipe:"or"})
	(e:event{tipe:"and"})
	Where not [r:SEQUENCE]
	(a1)-[r*2..]->(a2) where a1.nama="..." or a2.nama="..."
	Where not [r]

#### **3.1.4.7 List BPMN**

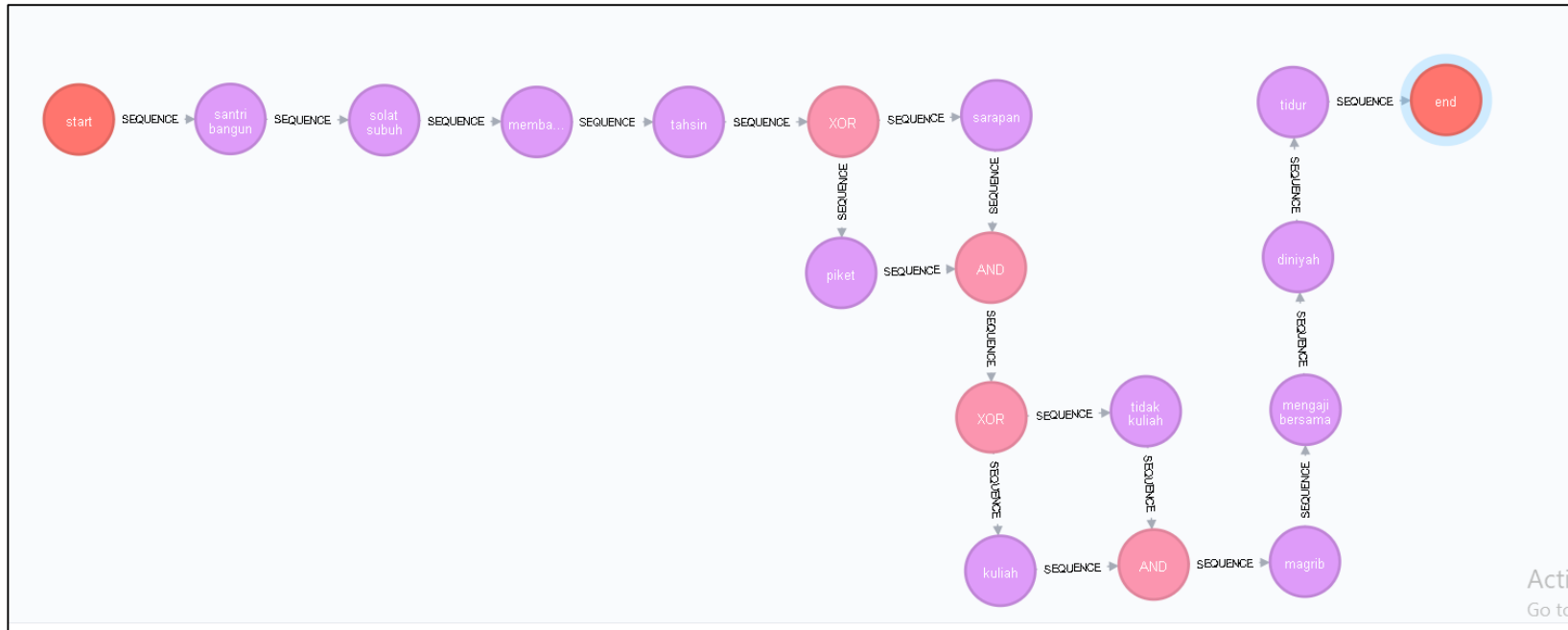
Hasil dari pencarian dimunculkan dalam bentuk *list*. Disinilah kegunaan *node* “deskripsi” dan *edge* “DIBUAT”. *Property* dari keduanya ditampilkan dalam tabel BPMN hasil pencarian.

#### **3.1.5 Implementasi BPMN-Q**

Dalam sistem yang dibangun, BPMN-Q diimplementasikan pada *cypher*. Contoh menggunakan *workflow* kesantrian pondok pesantren pada gambar 3.3. Dan hasil penyimpanan *workflow* pada NEO4J ada pada gambar 3.4.

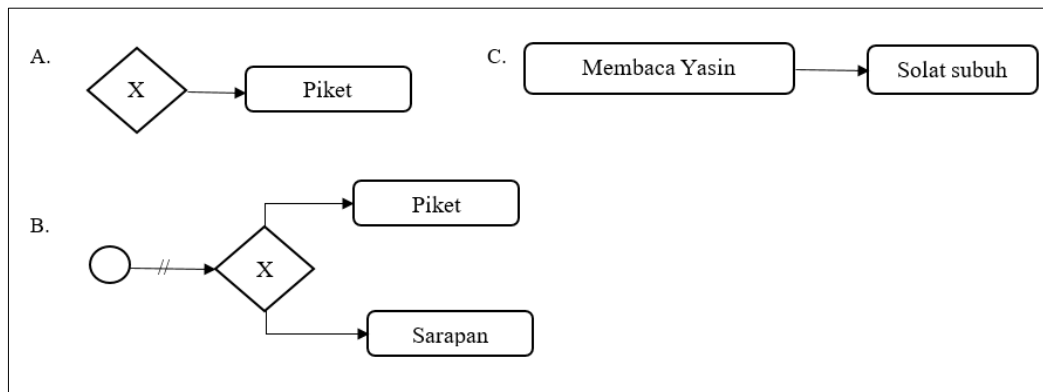


Gambar 3.3 BPMN Kesantrian



Gambar 3.4 BPMN pada NEO4J

Kemudian contoh *query* yang digunakan ada pada gambar 3.5.



Gambar 3.5 *Query* BPMN-Q

*Query* pada gambar 3.5 diparsing untuk diterjemahkan ke dalam

*Cypher*. *Cypher* dari *query* pada gambar 3.5 yaitu:

```
A. MATCH j=(g:gateway {nama:"xor"})-[r:SEQUENCE]->(a:activity {nama:"piket"}) RETURN j
```

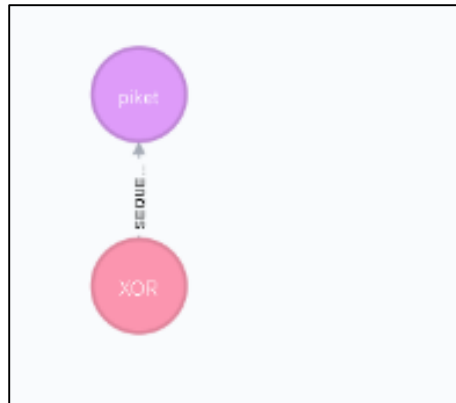
```
B. MATCH j=(e:event)-[r*2..]->(g:gateway)-->(a:activity) where e.nama="start" or g.nama="xor" or a.nama="piket" or a.nama="sarapan" return j
```

```
C. MATCH j=(a:activity {nama:"membaca yasin"})-[r:SEQUENCE]->(a:activity {nama:"solat subuh"}) RETURN j
```

Hasil:

A. *Query A*

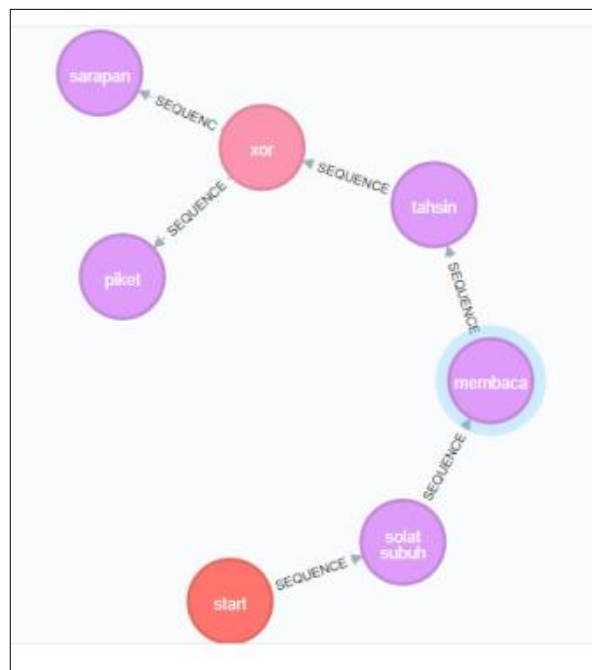
*Query A* menghasilkan *node* serta relasi yang sesuai. Hasil ada pada gambar 3.6.



Gambar 3.6 Hasil *Query A*

B. *Query B*

*Query B* menghasilkan *node* serta relasi yang sesuai. Hasil ada pada gambar 3.7.

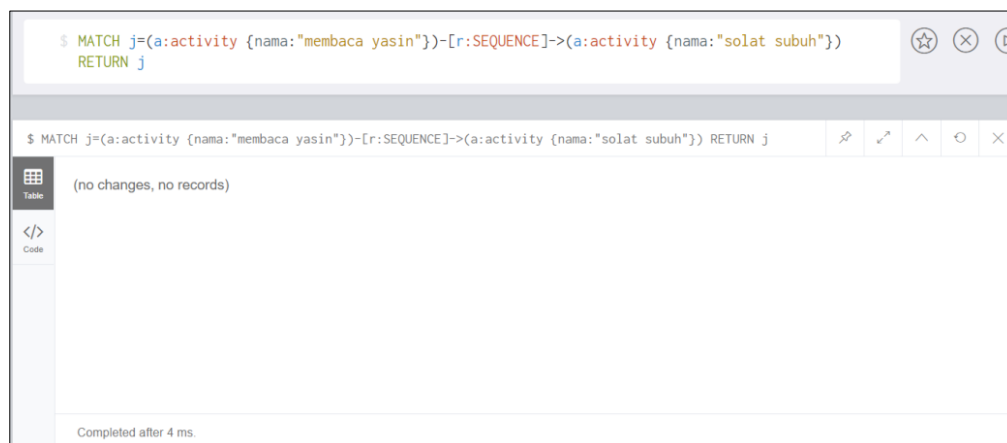


Gambar 3.7 Hasil *Query B*

C. *Query C*



*Query C* tidak mendapatkan hasil yang sesuai karena *activity* urutan yang salah. Seharusnya *activity* “solat subuh” ke *activity* “membaca yasin”. Hal ini dinamakan *negative edge*. Ada pada gambar 3.8.



Gambar 3.8 Hasil *Query C*

### 3.1.6 Pengujian Akurasi

Uji akurasi dilakukan untuk mengetahui seberapa besar tingkat akurasi yang diperoleh dari pencarian BPMN dari langkah-langkah implementasi database grafik menggunakan BPMN-Q di penyimpanan alur kerja. Perhitungan presisi menggunakan Presisi pada Rumus 1, maka tingkat *recall* pada Rumus 2 dan Akurasi pada Rumus 3. Efektivitas sistem pencarian bisa diukur dengan menghitung nilai perolehan (*recall*), nilai ketepatan (*accuracy*) dan jatuh semu (*fallout*) (Tague-Sutcliffe & Blustein, 1994). Namun, di antara metode ini, perhitungan akurasi adalah metode yang paling umum digunakan. Akurasi mewakili tingkat akurasi sistem yang menemukan informasi yang relevan kepada *user*. *Recall rate* adalah nilai yang menunjukkan tingkat penemuan dari hasil yang ditemukan oleh sistem.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

### 3.2 Perangkat yang digunakan

Perangkat yang digunakan dalam penelitian ini, yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*). Berikut penjelasan perangkat keras dan perangkat lunak yang digunakan:

#### 3.2.1 Perangkat keras

Perangkat keras yang digunakan dalam pembuatan dan pengujian dijelaskan pada Tabel 3.3.

Tabel 3.3 Spesifikasi Perangkat Keras

NO	Perangkat keras	Spesifikasi
1.	<i>Processor</i>	<i>Intel Core i7 6700HQ (Base 2,6 GHz/Turbo Boost 3,5 GHz)</i>
2.	<i>Graphics</i>	<i>NVIDIA GeForce GTX Graphics</i>
3.	RAM	4GB
4.	HDD	1000GB

#### 3.2.2 Perangkat lunak

Perangkat lunak yang digunakan dalam pembuatan dijelaskan pada Tabel 3.4.

Tabel 3.4 Spesifikasi Perangkat lunak untuk Pembuatan *Web*

NO	Perangkat Lunak	Spesifikasi
1.	Xampp	V3.2.2 (64bit)
2.	Codeigniter	3.1.5
3.	Sublime Text	V3
4.	NEO4J	V3.2.3 (64bit)
5.	Bizagi Modeler	V3.1.0.011 (64bit)
6.	Everyman	<i>Library neo4j</i> untuk php

Perangkat lunak yang digunakan dalam pengujian dijelaskan pada tabel 3.5.

Tabel 3.5 Perangkat Lunak untuk Pengujian

NO	Perangkat Lunak	Spesifikasi
1.	Xampp	V3.2.2 (64bit)
2.	NEO4J	V3.2.3 (64bit)
3.	Microsoft visio	2007
4.	BPMN-Q Shapes	V4_SBPMN-Q

### 3.3 *Interface Aplikasi*

Antarmuka (*interface*) pada penelitian ini menggunakan web *apps* yang dibangun menggunakan *framework* Codeigniter 3.1.5. Selain itu untuk

menyimpan semua informasi digunakan basis data graph yaitu NEO4J.

Berikut *interface* aplikasi :



### a. Halaman *Login*

Pada halaman *login* terdapat *input username* dan *password*. Berikut tampilan halaman *login* pada gambar 3.9.

Gambar 3.9 Halaman *Login*

### b. Halaman *Dashboard*

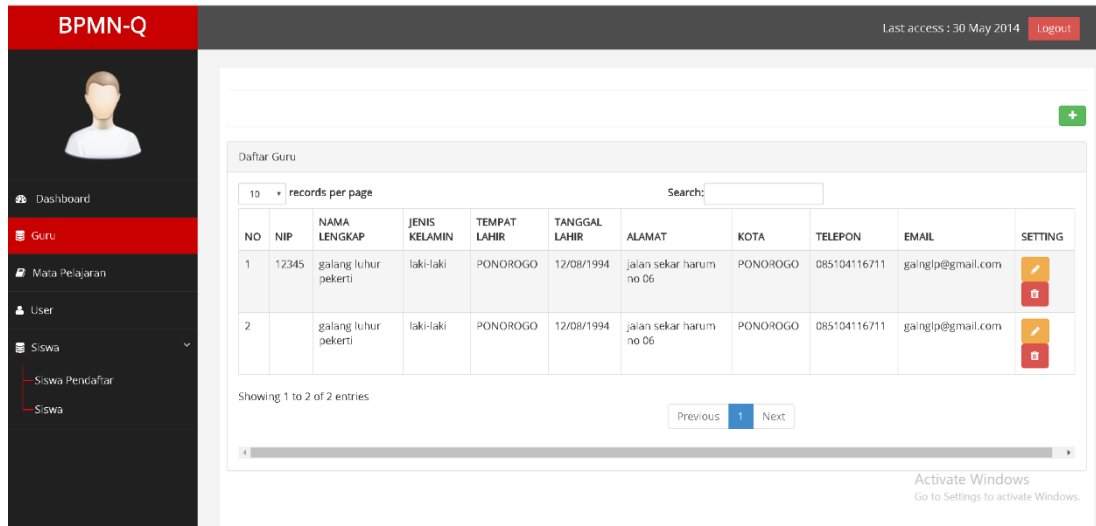
Halaman *dashboard* digunakan untuk pengelolaan *workflow*. Mulai dari *add*, *edit*, *list* dan *delete* workflow. Pada halaman ini juga digunakan untuk pencarian *workflow* menggunakan *bpmn-q*. Berikut tampilan halaman dashboard pada gambar 3.10.

NO	JUDUL	PENULIS	TANGGAL DIBUAT	TANGGAL MODIFIKASI	SETTING
1	penerimaan siswa baru MA al karimi	Administrator	2019-01-22T00:45:32.1919576+07:00	2019-01-22T00:58:34.3532886+07:00	 

Gambar 3.10 Halaman *Dashboard*

### c. Halaman Guru

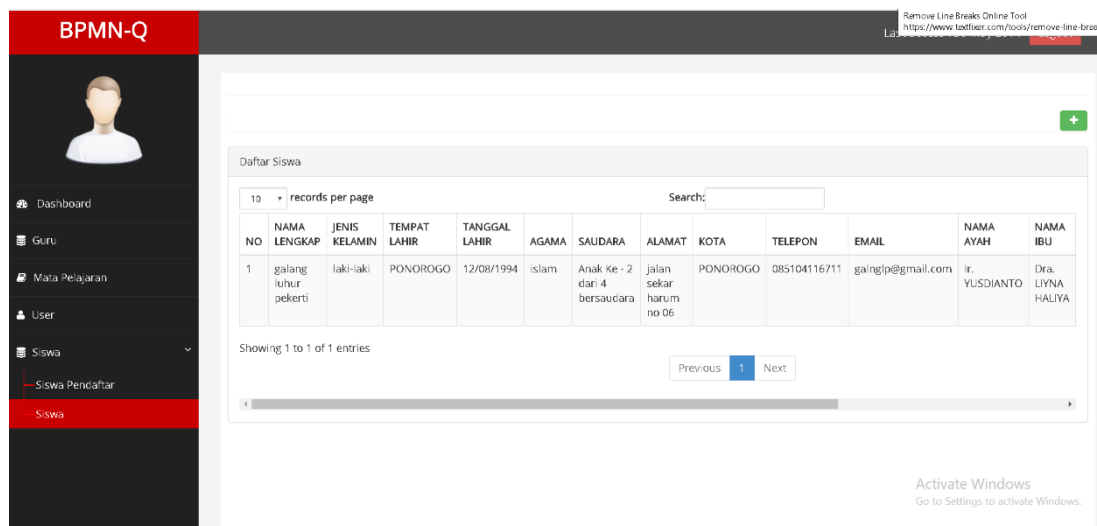
Halaman guru digunakan untuk menyimpan profil guru. Berikut tampilan halaman guru pada gambar 3.11.



Gambar 3.11 Halaman Guru

### d. Halaman Siswa

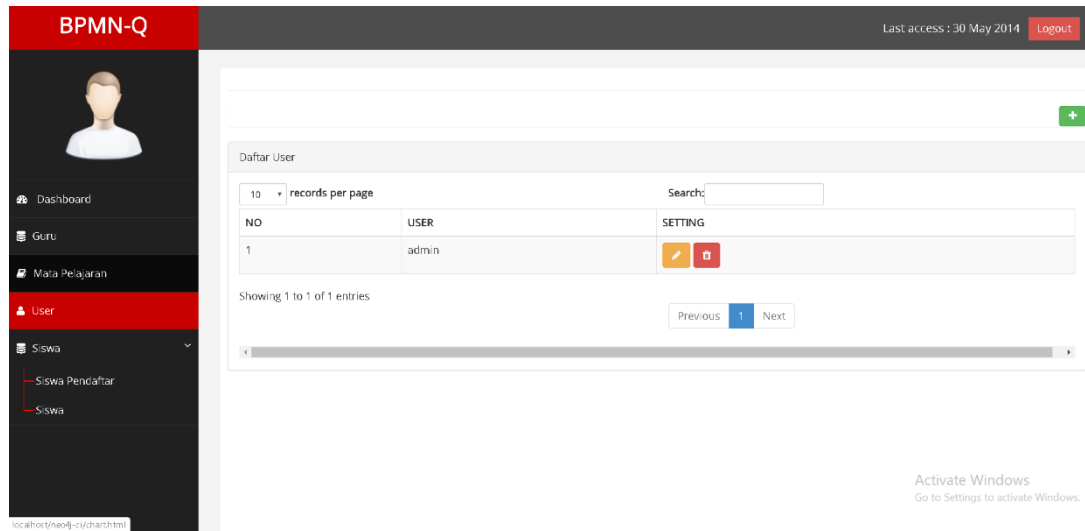
Halaman siswa digunakan untuk menyimpan profil siswa. Berikut tampilan halaman siswa pada gambar 3.12.



Gambar 3.12 Halaman siswa

### e. Halaman *User*

Halaman *user* digunakan untuk menyimpan *user*. Berikut tampilan halaman *user* pada gambar 3.13.



Gambar 3.13 Halaman *User*

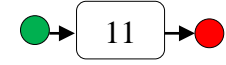
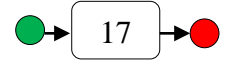
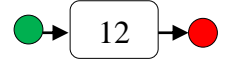
### 3.4 Langkah Uji Coba

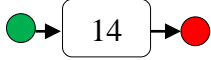
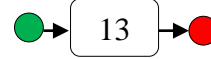
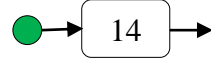
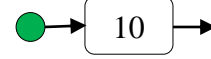
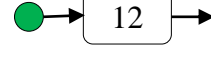
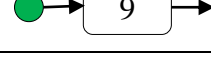
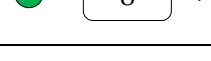
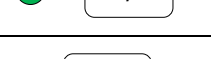
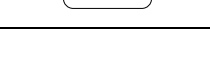
Langkah uji coba sistem adalah sebagai berikut :

#### 1) Pengumpulan data *workflow*

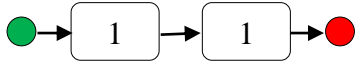
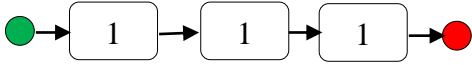
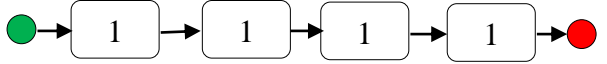
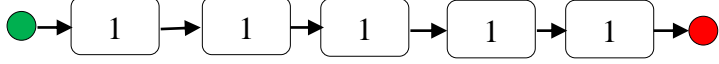
Data *workflow* diambil dari prosedur penerimaan santri baru berbagai pondok pesantren dan berbagai prosedur lain. Kemudian diubah menjadi BPMN menggunakan Bizagi Modeler. Penelitian ini menggunakan 50 BPMN. Data BPMN dibagi menjadi 3 tipe. Tipe sequence diberi kode WS, tipe cabang diberi kode WB dan tipe siklus diberi kode WC. Data BPMN sequence ada pada tabel 3.6.

Tabel 3.6 Data BPMN *Sequence*

NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
1	WS01	PSB Podok Pesantren cokro kertopati	
2	WS02	PSB Pondok Pesantren miftahu nurul huda	
3	WS03	PSB Pondok Pesantren miftahul huda	

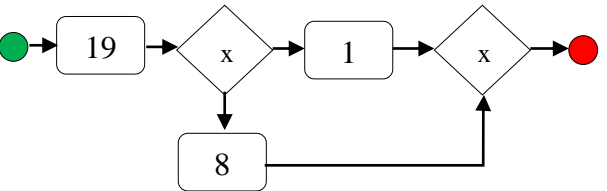
<b>NO</b>	<b>KODE</b>	<b>NAMA WORKFLOW</b>	<b>STRUKTUR</b>
4	WS04	PSB Pondok Pesantren NU tuluangung	
5	WS05	PSB Pondok Pesantren Tahfid Quran Oemah Quran	
6	WS06	PSB Pondok modern Al Hikam	
7	WS07	PSB SDN Sidorejo 1	
8	WS08	PSB MIN Pehnongko	
9	WS09	PSB MIN turi	
10	WS10	PSB MI psm jetak	
11	WS11	PSB SDN simo 1	
12	WS12	WS12	



NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
13	WS13	WS13	
14	WS14	WS14	
15	WS15	WS15	
16	WS16	WS16	

BPMN tipe cabang ada pada tabel 3.7.

Tabel 3.7 Data BPMN *Branch*

NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
1	WB01	PSB Pondok Pesantren Tambak Beras Jombang	

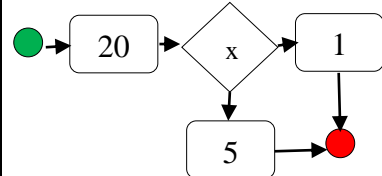
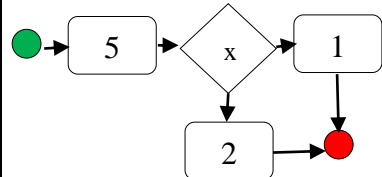
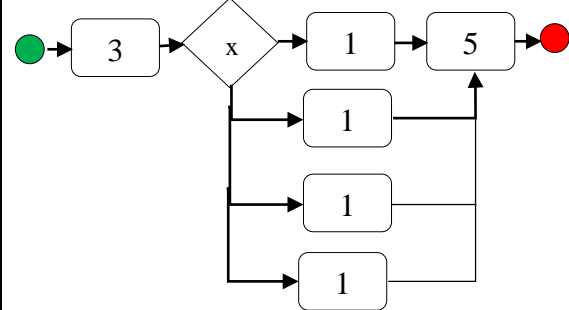
NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
2	WB02	PSB Pondok Pesantren al hikmah al fathimiyyah	<pre> graph LR   Start(( )) --&gt; N18[18]   N18 --&gt; D1{x}   D1 --&gt; N18_2[18]   D1 --&gt; N18_3[18]   N18_2 --&gt; End(( ))   N18_3 --&gt; End   </pre>
3	WB03	PSB Pondok Pesantren darul rohmah	<pre> graph LR   Start(( )) --&gt; N14[14]   N14 --&gt; D1{x}   D1 --&gt; N4[4]   D1 --&gt; N1[1]   N4 --&gt; End(( ))   N1 --&gt; End   </pre>
4	WB04	PSB Pondok Pesantren sabilurrosyad	<pre> graph LR   Start(( )) --&gt; N13[13]   N13 --&gt; D1{x}   D1 --&gt; N1_1[1]   D1 --&gt; N6[6]   N6 --&gt; N1_2[1]   N1_1 --&gt; End(( ))   N1_2 --&gt; End   </pre>
5	WB05	PSB Pondok Pesantren terpadu amanatul ummah	<pre> graph LR   Start(( )) --&gt; N17[17]   N17 --&gt; D1{x}   D1 --&gt; N1_1[1]   D1 --&gt; N9[9]   N9 --&gt; D2{x}   D2 --&gt; N1_2[1]   D2 --&gt; End(( ))   </pre>

NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
6	WB06	PSB AI Amanah AI Gontory	<pre> graph TD     Start((Green)) --&gt; N5[5]     N5 --&gt; D1{x}     D1 --&gt; N2_1[2]     D1 --&gt; D2{x}     N2_1 --&gt; D2     D2 --&gt; N4[4]     D2 --&gt; D3{x}     D3 --&gt; N2_2[2]     D3 --&gt; D4{x}     N2_2 --&gt; D4     D4 --&gt; N5_2[5]     D4 --&gt; N1[1]     N5_2 --&gt; A[+]     A --&gt; N7[7]     N7 --&gt; D5{x}     D5 --&gt; N2_3[2]     D5 --&gt; D6{x}     N2_3 --&gt; D6     D6 --&gt; N2_4[2]     D6 --&gt; N6[6]     N2_4 --&gt; D6     N6 --&gt; End((Red))     N1 --&gt; End   </pre>

NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
7	WB07	PSB Pesantren modern mahasiswa ar Rahman	<pre> graph TD     Start(( )) --&gt; N7_1[7]     N7_1 --&gt; D1{x}     D1 --&gt; N4[4]     D1 --&gt; N3_1[3]     N4 --&gt; D2{x}     D2 --&gt; N1_1[1]     N3_1 --&gt; D3{x}     D3 --&gt; N2_1[2]     N2_1 --&gt; D4{x}     D4 --&gt; N2_2[2]     N2_2 --&gt; N1_1     N1_1 --&gt; D5{x}     D5 --&gt; N7_2[7]     N7_2 --&gt; D1     D5 --&gt; N3_2[3]     N3_2 --&gt; D6{x}     D6 --&gt; N1_2[1]     N1_2 --&gt; D7{x}     D7 --&gt; N2_3[2]     N2_3 --&gt; D8{x}     D8 --&gt; N2_4[2]     N2_4 --&gt; D9{x}     D9 --&gt; N2_5[2]     N2_5 --&gt; D10{x}     D10 --&gt; N6[6]     N6 --&gt; End(( ))   </pre>



NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
11	WB11	PSB Pondok Tahfidz Al Barokah	
12	WB12	PSB Pondok Pesantren Tahfidz Nurul Furqon	
13	WB13	PSB Pondok Tahfidz Mahasiswa Madiun	

NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
14	WB14	PSB MA al karimi gresik	 <pre> graph LR   Start(( )) --&gt; 20[20]   20 --&gt; x{x}   x --&gt; 1[1]   x --&gt; 5[5]   1 --&gt; End(( ))   5 --&gt; End </pre>
15	WB15	PSB SMP maarif simo	 <pre> graph LR   Start(( )) --&gt; 5[5]   5 --&gt; x{x}   x --&gt; 1[1]   x --&gt; 2[2]   1 --&gt; End(( ))   2 --&gt; End </pre>
16	WB16	PKM ITS	 <pre> graph LR   Start(( )) --&gt; 3[3]   3 --&gt; x{x}   x --&gt; 1a[1]   x --&gt; 1b[1]   x --&gt; 1c[1]   x --&gt; 1d[1]   1a --&gt; 5[5]   1b --&gt; 5   1c --&gt; 5   1d --&gt; 5   5 --&gt; End(( )) </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
17	WB17	PMB mandiri tulis uin 2017	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{x}     D1 --&gt; T1_1[1]     D1 --&gt; T1_2[1]     D1 --&gt; T1_3[1]     T1_1 --&gt; T2[2]     T1_2 --&gt; T2     T1_3 --&gt; T2     T2 --&gt; D2{o}     D2 --&gt; T3[1]     D2 --&gt; T4[1]     T3 --&gt; D3{o}     D3 --&gt; T5[1]     D3 --&gt; T6[1]     D3 --&gt; T7[1]     T5 --&gt; D4{o}     T6 --&gt; D4     T7 --&gt; D4     D4 --&gt; T8[2]     T8 --&gt; End(( ))   </pre>



NO	KODE	NAMA <i>WORKFLOW</i>	STRUKTUR
18	WB18	Alur permintaan pembelian	<pre> graph LR     Start(( )) --&gt; T2[2]     T2 --&gt; D1{x}     D1 --&gt; T1_1[1]     D1 --&gt; T1_2[1]     D1 --&gt; T1_3[1]     T1_1 --&gt; End1(( ))     T1_2 --&gt; End2(( ))     T1_3 --&gt; T2_1[2]     T2_1 --&gt; End3(( )) </pre>
19	WB19	Proses Pengiriman Pengecer Perangkat Keras	<pre> graph LR     Start(( )) --&gt; D1{+}     D1 --&gt; T1_1[1]     D1 --&gt; T1_2[1]     T1_1 --&gt; D2{+}     D2 --&gt; T1_3[1]     T1_3 --&gt; End1(( ))     T1_2 --&gt; D3{x}     D3 --&gt; T1_4[1]     D3 --&gt; T1_5[2]     T1_4 --&gt; D4{o}     D4 --&gt; T1_6[1]     T1_6 --&gt; D5{o}     D5 --&gt; D3     T1_5 --&gt; D6{x}     D6 --&gt; D2     D6 --&gt; T1_7[1]     T1_7 --&gt; End2(( )) </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
20	WB20	Pendaftaran CPNS 2018	<pre> graph LR     Start((Green)) --&gt; T3_1[3]     T3_1 --&gt; T3_2[3]     T3_2 --&gt; D1{x}     D1 --&gt; T2[2]     D1 --&gt; T1_1[1]     T2 --&gt; T4[4]     T4 --&gt; D2{x}     D2 --&gt; T1_2[1]     D2 --&gt; End((Red))     T1_1 --&gt; End     T1_2 --&gt; End     T1_1 --&gt; T1_2          subgraph SubDiagram         T1_3[1] -.-&gt; T3_1         T1_4[1] -.-&gt; T3_2         T1_3 -.-&gt; T1_4     end </pre>
21	WB21	WB21	<pre> graph LR     Start((Green)) --&gt; T1_1[1]     T1_1 --&gt; D1{x}     D1 --&gt; T1_2[1]     D1 --&gt; T1_3[1]     D1 --&gt; T1_4[1]     T1_2 --&gt; D2{x}     T1_3 --&gt; D3{x}     T1_4 --&gt; D4{x}     D3 --&gt; T1_5[1]     D4 --&gt; T1_6[1]     T1_5 --&gt; D2     T1_6 --&gt; D2     D2 --&gt; T2[2]     T2 --&gt; End((Red)) </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
22	WB22	WB22	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{x}     D1 --&gt; T2[1]     D1 --&gt; T3[1]     D1 --&gt; T4[1]     T2 --&gt; T5[1]     T3 --&gt; D2{x}     T4 --&gt; D2     D2 --&gt; T6[1]     D2 --&gt; T7[1]     T6 --&gt; D3{x}     T7 --&gt; D3     D3 --&gt; D4{x}     D4 --&gt; T8[1]     T8 --&gt; End(( )) </pre>
23	WB23	WB23	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{x}     D1 --&gt; T2[1]     D1 --&gt; T3[1]     D1 --&gt; T4[1]     D1 --&gt; T5[1]     T2 --&gt; D2{x}     T3 --&gt; D2     T4 --&gt; D2     T5 --&gt; D2     D2 --&gt; T6[3]     T6 --&gt; End(( )) </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
24	WB24	WB24	<pre> graph LR   Start(( )) --&gt; N1[1]   N1 --&gt; D1{x}   D1 --&gt; N2[1]   N2 --&gt; D2{x}   D2 --&gt; N3[2]   N3 --&gt; End(( ))   D1 --&gt; B1[ ]   B1 --&gt; N4[1]   B1 --&gt; N5[1]   B1 --&gt; N6[1]   B1 --&gt; N7[1]   N4 --&gt; D2   N5 --&gt; D2   N6 --&gt; D2   N7 --&gt; D2   </pre>
25	WB25	WB25	<pre> graph LR   Start(( )) --&gt; N1[1]   N1 --&gt; D1{+}   D1 --&gt; N2[1]   N2 --&gt; D2{+}   D1 --&gt; N3[1]   N3 --&gt; D3{x}   D3 --&gt; N4[1]   N4 --&gt; D4{x}   D4 --&gt; D2   D1 --&gt; N5[1]   N5 --&gt; D4   D4 --&gt; N6[1]   N6 --&gt; D4   D2 --&gt; N7[1]   N7 --&gt; End(( ))   </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
26	WB26	WB26	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{x}     D1 --&gt; T2[1]     D1 --&gt; T3[1]     D1 --&gt; T4[1]     T2 --&gt; D2{x}     T3 --&gt; D2     T4 --&gt; D2     D2 --&gt; T5[1]     T5 --&gt; D3{+}     D3 --&gt; T6[1]     D3 --&gt; T7[1]     T6 --&gt; D4{+}     T7 --&gt; D4     D4 --&gt; T8[1]     T8 --&gt; D5{x}     D5 --&gt; T9[1]     T9 --&gt; End(( )) </pre>
27	WB27	WB27	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{o}     D1 --&gt; T2[1]     D1 --&gt; T3[1]     D1 --&gt; T4[1]     T2 --&gt; D2{o}     T3 --&gt; D2     T4 --&gt; D2     D2 --&gt; T5[1]     T5 --&gt; End(( )) </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
28	WB28	WB28	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{+}     D1 --&gt; T2[1]     D1 --&gt; T3[1]     D1 --&gt; T4[1]     T2 --&gt; D2{+}     T3 --&gt; D2     T4 --&gt; D2     D2 --&gt; T5[1]     T5 --&gt; D3{+}     D3 --&gt; T6[1]     D3 --&gt; T7[1]     T6 --&gt; D4{+}     T7 --&gt; D4     D4 --&gt; T8[1]     T8 --&gt; End((( ))) </pre>
29	WB29	WB29	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; D1{+}     D1 --&gt; T2[1]     D1 --&gt; T3[1]     D1 --&gt; T4[1]     T2 --&gt; T5[1]     T5 --&gt; D2{+}     T3 --&gt; D2     T4 --&gt; D2     D2 --&gt; T6[1]     T6 --&gt; D3{+}     D3 --&gt; T7[1]     D3 --&gt; T8[1]     T7 --&gt; D4{+}     T8 --&gt; D4     D4 --&gt; T9[1]     T9 --&gt; End((( ))) </pre>

BPMN tipe siklus ada pada tabel 3.8.

Tabel 3.8 Data BPMN Siklus

NO	KODE	NAMA WORKFLOW	STRUKTUR
1	WC01	PSB Pondok Pesantren mahasiswa al iqbal	<pre> graph LR     Start(( )) --&gt; T1[1]     T1 --&gt; T2[1]     T2 --&gt; T3[7]     T3 --&gt; D1{x}     D1 --&gt; T4[5]     T4 --&gt; End(( ))     D1 --&gt; T5[5]     T5 --&gt; T2     </pre>
2	WC02	PSB mts al hidayah ginuk	<pre> graph LR     Start(( )) --&gt; T1[5]     T1 --&gt; D1{x}     D1 --&gt; Start     D1 --&gt; T2[5]     T2 --&gt; D2{x}     D2 --&gt; T3[1]     T3 --&gt; D3{+}     D3 --&gt; End(( ))     </pre>
3	WC03	PSB SMP 1 maarif ponorogo	<pre> graph LR     Start(( )) --&gt; T1[7]     T1 --&gt; T2[3]     T2 --&gt; D1{x}     D1 --&gt; T3[1]     T3 --&gt; End(( ))     D1 --&gt; T4[3]     T4 --&gt; T2     </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
4	WC04	PSB MTSN unggulan (Asrama)	
5	WC05	WC05	
6	WC06	WC06	



NO	KODE	NAMA WORKFLOW	STRUKTUR
7	WC07	WC07	<pre> graph LR   Start((Green)) --&gt; N1[1]   N1 --&gt; D1{x}   D1 --&gt; N2[1]   N2 --&gt; D2{x}   D2 --&gt; N3[2]   N3 --&gt; End((Red))   D1 --&gt; N4[1]   N4 --&gt; D1   D2 --&gt; N5[1]   N5 --&gt; D2 </pre>
8	WC08	WC08	<pre> graph LR   Start((Green)) --&gt; N1[1]   N1 --&gt; N2[1]   N2 --&gt; D1{x}   D1 --&gt; N3[1]   N3 --&gt; D2{x}   D2 --&gt; N4[1]   N4 --&gt; End((Red))   D1 --&gt; N5[1]   N5 --&gt; D1   D2 --&gt; N6[1]   N6 --&gt; D2 </pre>
9	WC09	WC09	<pre> graph LR   Start((Green)) --&gt; N1[1]   N1 --&gt; N2[1]   N2 --&gt; D1{x}   D1 --&gt; N3[1]   N3 --&gt; D2{x}   D2 --&gt; N4[1]   N4 --&gt; D3{x}   D3 --&gt; N5[1]   N5 --&gt; End((Red))   D1 --&gt; N6[1]   N6 --&gt; D1   D2 --&gt; N7[1]   N7 --&gt; D2   D3 --&gt; N8[1]   N8 --&gt; D3 </pre>

NO	KODE	NAMA WORKFLOW	STRUKTUR
10	WC10	WC10	

## 2) BPMN parsing

BPMN dikonversi menjadi *file XPD*L terlebih dahulu. Sebelum disimpan ke dalam *workflow repository*. *Parsing* BPMN dilakukan dengan menyimpan BPMN ke dalam *workflow repository*.

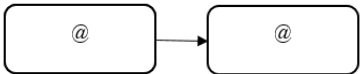
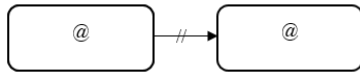

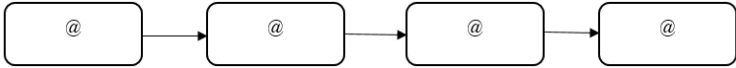
## 3) Query dengan BPMN-Q

Pembuatan *query* BPMN-Q menggunakan *microsoft visio 2007* dan BPMN-Q *shapes*. *Query* yang digunakan terbagi menjadi 4 variasi.

a. *Query* dengan struktur *sequence*. *Query* ada pada tabel

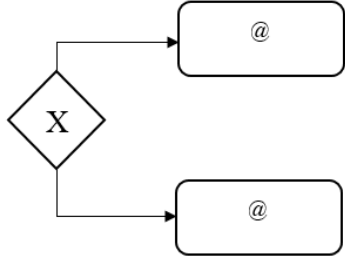
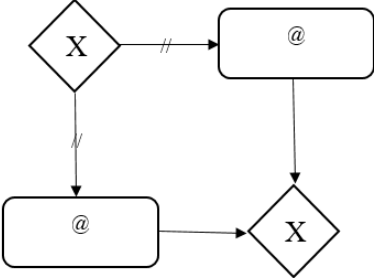
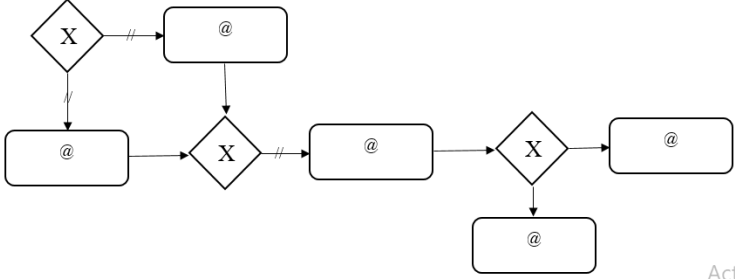
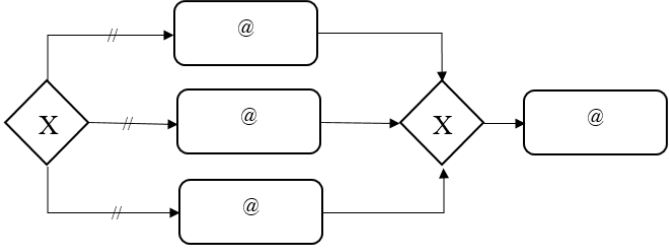
3.9 :

Tabel 3.9 BPMN-Q *sequence*

NO	KODE	BPMN-Q
1.	SQ01	
2.	SQ02	
3.	SQ03	
4.	SQ04	

b. *Query* dengan cabang. *Query* ada pada tabel 3.10 :

Tabel 3.10 BPMN-Q *Branch*

NO	KODE	BPMN-Q
1.	BR01	
2.	BR02	
3.	BR03	
4.	BR04	

c. *Query dengan siklus. Query ada pada tabel 3.11 :*

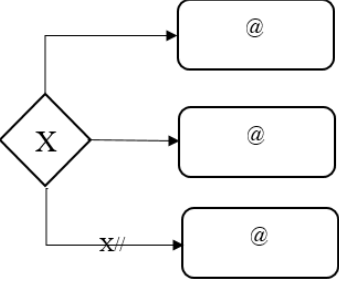
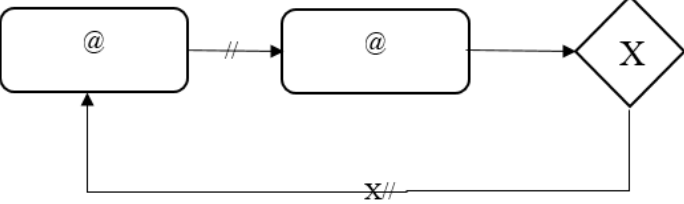
Tabel 3.11 BPMN-Q Siklus

NO	KODE	BPMN-Q
1.	CC01	
2.	CC02	
3.	CC03	
4.	CC04	

d. *Query dengan negative path. Query ada pada tabel 3.12:*

Tabel 3.12 BPMN-Q *Negative Path*

NO	KODE	BPMN-Q
1.	NP01	

NO	KODE	BPMN-Q
2.	NP02	
3.	NP03	

#### 4) Masukkan query

Dalam langkah ini *query* yang telah dibuat akan dikonversi menjadi *file XPD*L. Kemudian diinputkan untuk mencari BPMN yang diinginkan. *Query* diunggah pada halaman *dashboard*. Kemudian klik *search* maka sistem akan memproses *query* yang diterima.

#### 5) Hasil Pecarian

Setelah sistem memproses *query*. Sistem akan menampilkan beberapa BPMN yang sesuai dengan *query*. Hasil pencarian akan disajikan dalam bentuk *list*. Pada *list* terdapat tombol untuk menampilkan bentuk BPMN yang dipilih.

#### 6) Evaluasi Hasil

Langkah terakhir adalah mengevaluasi hasil dari pengolahan sistem. Efektivitas sistem temu kembali dapat diukur dengan menghitung nilai

perolehan (*recall*), nilai ketepatan (*accuracy*) dan jatuh semu (*fallout*) (Tague-Sutcliffe & Blustein, 1994).

## BAB 4

### UJI COBA DAN PEMBAHASAN

Pada bab ini membahas tentang hasil uji coba, perhitungan akurasi dan pembahasan.

#### 4.1 Hasil Uji Coba

Pengujian menggunakan *query* yang telah disusun berdasarkan BPMN-Q. *Query* dikelompokkan berdasarkan 4 tipe seperti pada subbab langkah uji coba. *Query* diujikan pada dataset yang sudah dikelompokkan. Dataset dikelompokkan menjadi 4 yaitu dataset *sequence*, *branch*, siklus dan campuran. Hasil penerjemahan *query* BPMN-Q ke *cypher* ada pada tabel 4.1.

Tabel 4.1 BPMN-Q ke *Cypher*

NO	KODE BPMN-Q	CYPHER
1	SQ01	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)- [:SEQUENCE]->(a2:activity)
2	SQ02	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)-[*..]- >(a2:activity)
3	SQ03	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)- [:SEQUENCE]->(a2:activity)-[:SEQUENCE]- >(a3:activity)
4	SQ04	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)- [:SEQUENCE]->(a2:activity)-[:SEQUENCE]- >(a3:activity)-[:SEQUENCE]->(a4:activity)
5	BR01	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]- >(g1:gateway{nama:"xor"})-[:SEQUENCE]- >(a1:activity),(g1)-[:SEQUENCE]->(a2:activity)
6	BR02	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]- >(g1:gateway{nama:"xor"})-[*..]->(a1:activity),(g1)-[*..]- >(a2:activity),(a1)-[:SEQUENCE]- >(g2:gateway{nama:"xor"}),(a2)-[:SEQUENCE]->(g2)



NO	KODE BPMN-Q	<i>CYPHER</i>
7	BR03	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]- >(g1:gateway{nama:"xor"})-[*..]->(a1:activity),(g1)-[*..]- >(a2:activity),(a1)-[:SEQUENCE]- >(g2:gateway{nama:"xor"}),(a2)-[:SEQUENCE]->(g2)- [*..]->(a3:activity)-[:SEQUENCE]- >(g3:gateway{nama:"xor"})-[:SEQUENCE]- >(a4:activity),(g3)-[:SEQUENCE]->(a5:activity)
8	BR04	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]- >(g1:gateway{nama:"xor"})-[*..]->(a1:activity),(g1)-[*..]- >(a2:activity),(g1)-[*..]->(a3:activity),(a2)- [:SEQUENCE]->(g2:gateway{nama:"xor"}),(a1)- [:SEQUENCE]->(g2),(a3)-[:SEQUENCE]->(g2)- [:SEQUENCE]->(a4:activity)
9	CC01	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)-[*..]- >(a2:activity)-[:SEQUENCE]- >(g1:gateway{nama:"xor"})-[*..]->(a3:activity)- [:SEQUENCE]->(a1)
10	CC02	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)- [:SEQUENCE]->(g1:gateway{nama:"xor"})- [:SEQUENCE]->(a2:activity)-[:SEQUENCE]->(a1)
11	CC03	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)-[*..]- >(g1:gateway{nama:"xor"})-[:SEQUENCE]->(a1),(g1)- [:SEQUENCE]->(a2:activity)
12	CC04	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)- [:SEQUENCE]->(a2:activity)-[:SEQUENCE]- >(g1:gateway{nama:"xor"})-[:SEQUENCE]- >(a3:activity)-[:SEQUENCE]- >(g2:gateway{nama:"xor"})-[:SEQUENCE]- >(a4:activity),(g2)-[:SEQUENCE]->(a2),(a4)- [:SEQUENCE]->(a1)
13	NP01	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)- [:SEQUENCE]->(a2:activity)-[:SEQUENCE]- >(a3:activity)-[:SEQUENCE]->(a4:activity)- [:SEQUENCE]->(a5:activity)-[*..]->(a6:activity)
14	NP02	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]- >(g1:gateway{nama:"xor"})-[:SEQUENCE]- >(a1:activity),(g1)-[:SEQUENCE]->(a2:activity),(g1)- [*..]->(a3:activity)

15	NP03	MATCH (d:deskripsi)-[r:DIBUAT]- (e1:event{nama:"start"})-[*..]->(a1:activity)-[*..]- >(a2:activity)-[:SEQUENCE]- >(g1:gateway{nama:"xor"})-[*..]->(a1)
----	------	---

Kemudian perhitungan akurasi digunakan untuk mengetahui seberapa nilai akurasi yang akan diperoleh dari implementasi *Graph Database* dalam pencarian BPMN menggunakan *query* visual pada *workflow repository*. Pada penelitian ini menghitung akurasi menggunakan presisi pada Rumus 1, maka tingkat *recall* pada Rumus 2 dan *accuracy* pada Rumus 3. Efektivitas sistem temu kembali dapat diukur dengan menghitung nilai perolehan (*recall*), nilai ketepatan (*accuracy*) dan jatuh semu (*fallout*) (Tague-Sutcliffe & Blustein, 1994). Namun, di antara metode ini, perhitungan *accuracy* adalah metode yang paling umum digunakan. *Accuracy* mewakili tingkat akurasi sistem yang menemukan informasi yang relevan kepada *user*. *Recall rate* adalah nilai yang menunjukkan tingkat penemuan dari hasil yang ditemukan oleh sistem. *Precision* menunjukkan ketepatan sistem dalam menemukan BPMN yang sesuai dengan *query* BPMN-Q

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

TP (*True Positive*) adalah jumlah BPMN yang sesuai dengan *query* BPMN-Q dan muncul pada hasil pencarian. FP (*False Positif*) jumlah BPMN yang tidak sesuai dengan *query* BPMN-Q dan muncul pada hasil pencarian. TN (*True Negative*) jumlah BPMN yang tidak sesuai dengan *query* BPMN-Q dan tidak muncul pada hasil pencarian. FN (*False Negative*) adalah jumlah

BPMN yang sesuai dengan *query* BPMN-Q dan tidak muncul pada hasil pencarian. Ketentuan dari TP, FP, TN, dan FN terdapat pada Tabel 4.17 dibawah ini.

Tabel 4.2 Keterangan TP, FP, TN dan FN

NO	VARIABEL	KETERANGAN
1	TP ( <i>True Positive</i> )	Jumlah BPMN yang sesuai dengan <i>query</i> BPMN-Q dan muncul pada hasil pencarian
2	FP ( <i>False Positive</i> )	Jumlah BPMN yang tidak sesuai dengan <i>query</i> BPMN-Q dan muncul pada hasil pencarian
3	TN ( <i>True Negative</i> )	Jumlah BPMN yang tidak sesuai dengan <i>query</i> BPMN-Q dan tidak muncul pada hasil pencarian
4	FN ( <i>False Negative</i> )	Jumlah BPMN yang sesuai dengan <i>query</i> BPMN-Q dan tidak muncul pada hasil pencarian

#### 4.1.1 Sequence

Pengujian pertama dilakukan pada dataset *sequence*. Data ada pada tabel 3.6. Pengujian menggunakan 4 tipe *query* yang telah dijelaskan pada langkah uji coba. Hasil uji coba pada data *sequence* ada pada tabel 4.3.

Tabel 4.3 Hasil uji coba pada data *sequence*

<b>NO</b>	<b>KODE QUERY</b>	<b>JUMLAH WORKFLOW YANG SESUAI</b>	<b>KODE WORKFLOW YANG SESUAI</b>	<b>HASIL</b>	<b>TP</b>	<b>FP</b>	<b>TN</b>	<b>FN</b>
1	SQ01	15	WS01-WS11,WS13-WS16	WS01-WS11,WS13-WS16	15	0	1	0
2	SQ02	15	WS01-WS11,WS13-WS16	WS01-WS11,WS13-WS16	15	0	1	0
3	SQ03	14	WS01-WS11.WS14,WS15,WS16	WS01-WS11,WS14-WS16	14	0	2	0
4	SQ04	13	WS01-WS11,WS15,WS16	WS01-WS11,WS15,WS16	13	0	3	0
5	BR01	0	-	-	0	0	16	0
6	BR02	0	-	-	0	0	16	0
7	BR03	0	-	-	0	0	16	0
8	BR04	0	-	-	0	0	16	0
9	CC01	0	-	-	0	0	16	0
10	CC02	0	-	-	0	0	16	0
11	CC03	0	-	-	0	0	16	0
12	CC04	0	-	-	0	0	16	0
13	NP01	12	WS01-WS11,WS16	WS01-WS11	11	0	4	1
14	NP02	0	-	-	0	0	16	0
15	NP03	0	-	-	0	0	16	0

Setelah mendapatkan hasil uji coba dilakukan perhitungan nilai *precision*, *recall* dan *accuracy*. Dari 15 query yang diujikan menghasilkan nilai *precision* 1. Nilai *recall* dan *accuracy* bernilai 1 pada 14 *query* yang diujikan. Pada *query negative path* berkode “NP01” nilai *recall* dan *accuracy* berturut-turut yaitu 0,9167 dan 0,9375. Hasil perhitungan ada pada tabel 4.4.

Tabel 4.4 Nilai *precision*, *recall* dan *accuracy* pada data *sequence*

<b>NO</b>	<b>KODE QUERY</b>	<b>PRECISION</b>	<b>RECALL</b>	<b>ACCURACY</b>
1	SQ01	1	1	1
2	SQ02	1	1	1
3	SQ03	1	1	1
4	SQ04	1	1	1
5	BR01	1	1	1
6	BR02	1	1	1
7	BR03	1	1	1
8	BR04	1	1	1
9	CC01	1	1	1
10	CC02	1	1	1
11	CC03	1	1	1
12	CC04	1	1	1
13	NP01	1	0,916666667	0,9375
14	NP02	1	1	1
15	NP03	1	1	1

Kemudian dihitung nilai rata-rata dan persentase dari *precision*, *recall* dan *accuracy*. Persentase *precision*, *recall* dan *accuracy* dari data *sequence* berturut-turut yaitu 100%, 99,44% dan 99,58%. Nilai rata-rata dan persentase ada pada tabel 4.5.

Tabel 4.5 Nilai rata-rata dan persentase pada data *sequence*

<b>NO</b>	<b>VARIABEL</b>	<b>RATA-RATA</b>	<b>PERSENTASE</b>
1	Precision	1	100%
2	Recall	0,994444444	99,44%
3	Accuracy	0,995833333	99,58%

#### 4.1.2 Branch

Pengujian kedua dilakukan pada dataset *branch*. Data ada pada tabel 3.7. Pengujian menggunakan 4 tipe *query* yang telah dijelaskan pada langkah uji coba. Hasil uji coba pada data *branch* ada pada tabel 4.6.

Tabel 4.6 Hasil uji coba pada data *branch*

<b>NO</b>	<b>KODE QUERY</b>	<b>JUMLAH WORKFLOW YANG SESUAI</b>	<b>KODE WORKFLOW YANG SESUAI</b>	<b>HASIL</b>	<b>TP</b>	<b>FP</b>	<b>TN</b>	<b>FN</b>
1	SQ01	25	WB01-WB24,WB29	WB01-WB24,WB29	25	0	4	0
2	SQ02	29	WB01-WB29	WB01-WB29	29	0	0	0
3	SQ03	20	WB01-WB18,WB20,WB23	WB01-WB18,WB20,WB23	20	0	9	0
4	SQ04	17	WB01-WB16,WB20	WB01-WB16,WB20	17	0	12	0
5	BR01	26	WB01-WB26	WB01-WB26	26	0	3	0
6	BR02	16	WB01,WB05-WB13,WB21-WB26	WB01,WB05-WB13,WB21-WB26	16	0	13	0
7	BR03	6	WB06-WB09,WB12,WB13	WB06-WB09,WB12,WB13	6	0	23	0
8	BR04	3	WB13,WB23,WB24	WB13,WB23,WB24	3	0	26	0
9	CC01	0	-	-	0	0	29	0
10	CC02	0	-	-	0	0	29	0
11	CC03	0	-	-	0	0	29	0
12	CC04	0	-	-	0	0	29	0
13	NP01	17	WB01-WB16,WB20	WB01-WB16,WB20	17	0	12	0
14	NP02	26	WB01-WB26	WB13,WB16,WB17,WB18,WB21,WB22,WB23,WB24,WB26	9	0	3	17
15	NP03	21	WB01-WB16,WB18-WB20,WB22,WB29	-	0	0	8	21

Setelah mendapatkan hasil uji coba dilakukan perhitungan nilai *precision*, *recall* dan *accuracy*. Dari 13 *query* yang diujikan menghasilkan nilai *precision*, *recall* dan *accuracy* 1. Sedangkan pada *query* berkode “NP02” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 1, 0,346154 dan 0,413793103. Pada *query* berkode “NP03” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0, 0 dan 0,275862069. Hasil perhitungan ada pada tabel 4.7.

Tabel 4.7 Nilai *precision*, *recall* dan *accuracy* pada data *branch*

<b>NO</b>	<b>KODE QUERY</b>	<b>PRECISION</b>	<b>RECALL</b>	<b>ACCURACY</b>
1	SQ01	1	1	1
2	SQ02	1	1	1
3	SQ03	1	1	1
4	SQ04	1	1	1
5	BR01	1	1	1
6	BR02	1	1	1
7	BR03	1	1	1
8	BR04	1	1	1
9	CC01	1	1	1
10	CC02	1	1	1
11	CC03	1	1	1
12	CC04	1	1	1
13	NP01	1	1	1
14	NP02	1	0,346154	0,413793103
15	NP03	0	0	0,275862069

Kemudian dihitung nilai rata-rata dan persentase dari *precision*, *recall* dan *accuracy*. Persentase *precision*, *recall* dan *accuracy* dari data *branch* berturut-turut yaitu 93,33%, 88,97% dan 91,26%. Nilai rata-rata dan persentase ada pada tabel 4.8.

Tabel 4.8 Nilai rata-rata dan persentase pada data *branch*

<b>NO</b>	<b>VARIABEL</b>	<b>RATA-RATA</b>	<b>PERSENTASE</b>
1	Precision	0,933333333	93,33%
2	Recall	0,88974359	88,97%
3	Accuracy	0,912643678	91,26%



### 4.1.3 Siklus

Pengujian kedua dilakukan pada dataset siklus. Data ada pada tabel 3.8. Pengujian menggunakan 4 tipe *query* yang telah dijelaskan pada langkah uji coba. Hasil uji coba pada data siklus ada pada tabel 4.9.

Tabel 4.9 Hasil uji coba pada data siklus

<b>NO</b>	<b>KODE QUERY</b>	<b>JUMLAH WORKFLOW YANG SESUAI</b>	<b>KODE WORKFLOW YANG SESUAI</b>	<b>HASIL</b>	<b>TP</b>	<b>FP</b>	<b>TN</b>	<b>FN</b>
1	SQ01	9	WC01-WC09	WC01-WC09	9	0	1	0
2	SQ02	10	WC01-WC10	WC01-WC10	10	0	0	0
3	SQ03	5	WC01-WC04,WC06	WC01-WC04,WC06	5	0	5	0
4	SQ04	4	WC01-WC04	WC01-WC04	4	0	6	0
5	BR01	9	WC01-WC09	WC01-WC10	9	1	0	0
6	BR02	0	-	WC10	0	1	9	0
7	BR03	0	-	-	0	0	10	0
8	BR04	0	-	-	0	0	10	0
9	CC01	6	WC01,WC03,WC04,WC06,WC08,WC09	WC01,WC03,WC04,WC06,WC08,WC09	6	0	4	0
10	CC02	0	-	-	0	0	10	0
11	CC03	2	WC02,WC06	WC02,WC06	2	0	8	0
12	CC04	1	WC06	WC06	1	0	9	0
13	NP01	4	WC01-WC04	WC01-WC04	4	0	6	0
14	NP02	8	WC01-WC04,WC06-WC09	WC10	0	1	1	8
15	NP03	9	WC01-WC09	WC01-WC04,WC06,WC08,WC09	7	0	1	2

Setelah mendapatkan hasil uji coba dilakukan perhitungan nilai *precision*, *recall* dan *accuracy*. Pada *query* berkode “SQ01”, “SQ02”, “SQ03”, “SQ04”, “BR03”, “BR04”, “CC01”, “CC02”, “CC03”, “CC04” dan “NP01” menghasilkan nilai *precision*, *recall* dan *accuracy* 1. Pada *query* berkode “BR01” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0,9, 1 dan 0,9. Pada *query* berkode “BR02” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0, 0 dan 0,9. Pada *query* berkode “NP02” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0, 0 dan 0,1. Pada *query* berkode “NP03” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 1, 0,777778 dan 0,8. Hasil perhitungan ada pada tabel 4.10.

Tabel 4.10 Nilai *precision*, *recall* dan *accuracy* pada data siklus

<b>NO</b>	<b>KODE QUERY</b>	<b>PRECISION</b>	<b>RECALL</b>	<b>ACCURACY</b>
1	SQ01	1	1	1
2	SQ02	1	1	1
3	SQ03	1	1	1
4	SQ04	1	1	1
5	BR01	0,9	1	0,9
6	BR02	0	0	0,9
7	BR03	1	1	1
8	BR04	1	1	1
9	CC01	1	1	1
10	CC02	1	1	1
11	CC03	1	1	1
12	CC04	1	1	1
13	NP01	1	1	1
14	NP02	0	0	0,1
15	NP03	1	0,777778	0,8

Kemudian dihitung nilai rata-rata dan persentase dari *precision*, *recall* dan *accuracy*. Persentase *precision*, *recall* dan *accuracy* dari data siklus berturut-turut yaitu 86%, 85,19% dan 91,33%. Nilai rata-rata dan persentase ada pada tabel 4.11.

Tabel 4.11 Nilai rata-rata dan persentase pada data siklus

NO	VARIABEL	RATA-RATA	PERSENTASE
1	Precision	0,86	86%
2	Recall	0,851851852	85,19%
3	Accuracy	0,913333333	91,33%

#### 4.1.4 Campuran

Pengujian kedua dilakukan pada dataset campuran. Pengujian menggunakan 4 tipe *query* yang telah dijelaskan pada langkah uji coba. Hasil uji coba pada data siklus ada pada tabel 4.12.

Tabel 4.12 Hasil uji coba pada data campuran

NO	KODE QUERY	JUMLAH WORKFLOW YANG SESUAI	KODE WORKFLOW YANG SESUAI	HASIL	TP	FP	TN	FN
1	SQ01	49	WS01-WS11,WS13-WS16,WB01-WB24,WB29,WC01-WC09	WS01-WS11,WS13-WS16,WB01-WB24,WB29,WC01-WC09	49	0	1	0
2	SQ02	54	WS01-WS11,WS13-WS16,WB01-WB29,WC01-WC10	WS01-WS11,WS13-WS16,WB01-WB29,WC01-WC10	54	0	0	0
3	SQ03	39	WS01-WS11,WS14-WS16,WB01-WB18,WB20,WB23,WC01-WC04,WC06	WS01-WS11,WS14-WS16,WB01-WB18,WB20,WB23,WC01-WC04,WC06	39	0	5	0

<b>NO</b>	<b>KODE QUERY</b>	<b>JUMLAH WORKFLOW YANG SESUAI</b>	<b>KODE WORKFLOW YANG SESUAI</b>	<b>HASIL</b>	<b>TP</b>	<b>FP</b>	<b>TN</b>	<b>FN</b>
4	SQ04	34	WS01-WS11,WS15,WS16,WB01-WB16,WB20,WC01-WC04	WS01-WS11,WS15,WS16,WB01-WB16,WB20,WC01-WC04	34	0	6	0
5	BR01	35	WB01-WB26,WC01-WC09	WB01-WB26,WC01-WC10	35	1	19	0
6	BR02	16	WB01,WB05-WB13,WB21-WB26	WB01,WB05-WB13,WB21-WB26,WC10	16	1	38	0
7	BR03	6	WB06-WB09,WB12,WB13	WB06-WB09,WB12,WB13,	6	0	49	0
8	BR04	3	WB13,WB23,WB24	WB13,WB23,WB24	3	0	52	0
9	CC01	6	WC01,WC03,WC04,WC06,WC08,WC09	WC01,WC03,WC04,WC06,WC08,WC09	6	0	49	0
10	CC02	0	-	-	0	0	55	0
11	CC03	2	WC02,WC06	WC02,WC06	2	0	53	0
12	CC04	1	WC06	WC06	1	0	54	0
13	NP01	33	WS01-WS11,WS16,WB01-WB16,WB20,WC01-WC04	WS01-WS11,WB01-WB16,WB20,WC01-WC04	32	0	22	1
14	NP02	34	WB01-WB26,WC01-WC04,WC06-WC09	WB13,WB16-WB18,WB21-WB24,WB26,WC10	9	1	28	17
15	NP03	30	WB01-WB16,WB18-WB20,WB22,WB29,WC01-WC09	WC01-WC04,WC06,WC08,WC09	7	0	25	23

Setelah mendapatkan hasil uji coba dilakukan perhitungan nilai *precision*, *recall* dan *accuracy*. Pada *query* berkode “SQ01”, “SQ02”, “SQ03”, “SQ04”, “BR03”, “BR04”, “CC01”, “CC02”, “CC03” dan “CC04” menghasilkan nilai *precision*, *recall* dan *accuracy* 1. Pada *query* berkode “BR01” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0,972222222, 1 dan 0,981818182. Pada *query* berkode “BR02” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0,941176471, 1 dan 0,981818182. Pada *query* berkode “NP01” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 1, 0,96969697 dan 0,981818182. Pada *query* berkode “NP02” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 0,9, 0,346153846 dan 0,672727273. Pada *query* berkode “NP03” menghasilkan nilai *precision*, *recall* dan *accuracy* berturut-turut yaitu 1, 0,233333333 dan 0,581818182. Hasil perhitungan ada pada tabel 4.13.

Tabel 4.13 Nilai *precision*, *recall* dan *accuracy* pada data campuran

<b>NO</b>	<b>KODE QUERY</b>	<b>PRECISION</b>	<b>RECALL</b>	<b>ACCURACY</b>
<b>1</b>	SQ01	1	1	1
<b>2</b>	SQ02	1	1	1
<b>3</b>	SQ03	1	1	1
<b>4</b>	SQ04	1	1	1
<b>5</b>	BR01	0,972222222	1	0,981818182
<b>6</b>	BR02	0,941176471	1	0,981818182
<b>7</b>	BR03	1	1	1
<b>8</b>	BR04	1	1	1
<b>9</b>	CC01	1	1	1
<b>10</b>	CC02	1	1	1
<b>11</b>	CC03	1	1	1
<b>12</b>	CC04	1	1	1
<b>13</b>	NP01	1	0,96969697	0,981818182
<b>14</b>	NP02	0,9	0,346153846	0,672727273
<b>15</b>	NP03	1	0,233333333	0,581818182

Kemudian dihitung nilai rata-rata dan persentase dari *precision*, *recall* dan *accuracy*. Persentase *precision*, *recall* dan *accuracy* dari data campuran berturut-turut yaitu 98,75%, 90,33% dan 94,67%. Nilai rata-rata dan persentase ada pada tabel 4.14.

Tabel 4.14 Nilai rata-rata dan persentase pada data campuran

NO	VARIABEL	RATA-RATA	PERSENTASE
1	Precision	0,987559913	98,75%
2	Recall	0,903278943	90,33%
3	Accuracy	0,946666667	94,67%

## 4.2 Pembahasan

BPMN-Q adalah *query* visual yang didasarkan pada notasi BPMN [14]. *Query* BPMN-Q dirancang untuk membantu pemodelan proses bisnis pada permintaan *repository* model proses bisnis berbasis grafik. *Query* ini bergantung pada notasi BPMN sebagai sintaks. *Query* uji coba dibagi menjadi 4 tipe yaitu : *sequence*, cabang, siklus dan negative path. *Query* diujikan pada dataset yang telah dikelompokkan. Dataset dikelompokkan menjadi 4 yaitu dataset *sequence*, *branch*, siklus dan campuran.

Pada dataset *sequence* semua *query* berhasil menemukan BPMN yang sesuai. Maka dihasilkan persentase *precision* 100%. Hal ini dikarenakan dataset *sequence* mempunyai struktur yang sederhana. Bentuk strukturnya lurus tanpa adanya percabangan.

Persentase *recall* pada dataset *sequence* didapatkan nilai 99,44%. Kesalahan hasil pencarian terjadi pada *query* “NP01”. “NP01” dapat menemukan 11 BPMN yang sesuai, akan tetapi terdapat satu BPMN yang sesuai tidak ditemukan. BPMN yang tidak dapat ditemukan berkode “WS16”. “WS16” tidak

dapat ditemukan karena pada *query* “NP01” sistem menemukan BPMN yang memiliki 6 *activity* dengan struktur *sequence*. Sedangkan yang dimaksud dari *query* “NP01” mencari BPMN dengan 5 *activity* dan pada *activity* kelima menuju notasi berikutnya tidak boleh ada notasi *activity*.

Kemudian persentase *accuracy* pada dataset *sequence* didapatkan nilai 99,58%. *Accuracy* yang sangat baik. Hal ini dikarenakan dataset *sequence* mempunyai struktur yang sederhana. Bentuk strukturnya lurus tanpa adanya percabangan.

Pada dataset *branch* persentase *precision* didapatkan nilai 93,33%. Karena pada *query* “NP03” tidak dapat menemukan BPMN yang sesuai. Harusnya “NP03” menemukan 21 BPMN yang sesuai. 21 BPMN yang tidak dapat ditemukan karena pada *query* “NP03” sistem mencari BPMN yang memiliki 2 *activity* dan 1 *gateway* dengan adanya siklus. Sedangkan yang dimaksud dari *query* “NP03” mencari BPMN dengan 2 *activity* dan 1 *gateway* tanpa adanya siklus.

Persentase *recall* pada dataset *branch* didapatkan nilai 88,97%. Kesalahan hasil pencarian terjadi pada *query* “NP02” dan “NP03”. “NP02” dapat menemukan 9 BPMN yang sesuai, akan tetapi terdapat 17 BPMN yang sesuai tidak ditemukan. 17 BPMN yang tidak dapat ditemukan karena sistem menemukan BPMN yang memiliki 3 percabangan. Sedangkan yang dimaksud dari *query* “NP02” mencari BPMN dengan 2 percabangan dan tidak memiliki percabangan ketiga. “NP03” tidak dapat menemukan satupun yang sesuai, akan tetapi terdapat 21 BPMN yang sesuai. 21 BPMN yang tidak dapat ditemukan karena sistem mencari BPMN yang memiliki 2 *activity* dan 1 *gateway* dengan adanya siklus.

Sedangkan yang dimaksud dari *query* “NP03” mencari BPMN dengan 2 *activity* dan 1 *gateway* tanpa adanya siklus.

Kemudian persentase *accuracy* pada dataset *branch* didapatkan nilai 91,26%. Persentase *accuracy* pada dataset *branch* sedikit menurun dibanding dataset *sequence*. Karena BPMN tipe *branch* memiliki struktur lebih kompleks.

Pada dataset siklus persentase *precision* didapatkan nilai 86%. Kesalahan hasil pencarian terjadi pada *query* “BR01”, “BR02” dan “NP02”. “BR01” dapat menemukan 9 BPMN yang sesuai, akan tetapi terdapat 1 BPMN yang tidak sesuai ikut terpanggil. 1 BPMN yang tidak sesuai ditemukan karena sistem menemukan BPMN yang memiliki *gateway* XOR. Sedangkan 1 BPMN yang tidak sesuai adalah *gateway* AND. “BR02” dapat menemukan 1 BPMN, akan tetapi BPMN yang tidak sesuai. 1 BPMN yang tidak sesuai ditemukan karena sistem menemukan BPMN yang memiliki *gateway* XOR. Sedangkan 1 BPMN yang tidak sesuai adalah *gateway* AND. “NP02” dapat menemukan 1 BPMN, akan tetapi BPMN yang tidak sesuai. 1 BPMN yang tidak sesuai ditemukan karena sistem menemukan BPMN yang memiliki *gateway* XOR. Sedangkan 1 BPMN yang tidak sesuai adalah *gateway* AND. Selain itu 8 BPMN yang sesuai tidak dapat ditemukan. 8 BPMN yang tidak dapat ditemukan karena sistem menemukan BPMN yang memiliki 3 percabangan. Sedangkan yang dimaksud dari *query* “NP02” mencari BPMN dengan 2 percabangan dan tidak memiliki percabangan ketiga.

Persentase *recall* pada dataset siklus didapatkan nilai 85,19%. Kesalahan hasil pencarian terjadi pada *query* “BR02”, “NP02” dan “NP03”. “BR02” dapat menemukan 1 BPMN, akan tetapi BPMN yang tidak sesuai. 1 BPMN yang tidak sesuai ditemukan karena sistem menemukan BPMN yang memiliki *gateway* XOR.



Sedangkan 1 BPMN yang tidak sesuai adalah *gateway* AND. “NP02” dapat menemukan 1 BPMN, akan tetapi BPMN yang tidak sesuai. 1 BPMN yang tidak sesuai ditemukan karena sistem menemukan BPMN yang memiliki *gateway* XOR. Sedangkan 1 BPMN yang tidak sesuai adalah *gateway* AND. Selain itu 8 BPMN yang sesuai tidak dapat ditemukan. 8 BPMN yang tidak dapat ditemukan karena sistem menemukan BPMN yang memiliki 3 percabangan. Sedangkan yang dimaksud dari *query* “NP02” mencari BPMN dengan 2 percabangan dan tidak memiliki percabangan ketiga. “NP03” dapat menemukan 7 BPMN yang sesuai, akan tetapi terdapat 2 BPMN yang sesuai tidak ditemukan. 2 BPMN yang tidak dapat ditemukan karena sistem menemukan BPMN yang memiliki 2 *activity* dan 1 *gateway* dengan adanya siklus. Sedangkan yang dimaksud dari *query* “NP03” mencari BPMN dengan 2 *activity* dan 1 *gateway* tanpa adanya siklus.

Kemudian persentase *accuracy* pada dataset siklus didapatkan nilai 91,33%. Persentase *accuracy* pada dataset siklus sedikit menurun dibanding dataset *sequence*. Karena BPMN tipe *branch* memiliki struktur lebih kompleks.

Selanjutnya semua dataset digabung menjadi dataset campuran. Nilai persentase *precision*, *recall* dan *accuracy* yang diperoleh berturut-turut yaitu 98,75%, 90,33% dan 94,67%. Nilai *precision* dan *accuracy* memiliki nilai diatas 90%. Artinya ketepatan sistem dalam menemukan BPMN yang sesuai sangat baik dan akurat.

Untuk penentuan performa BPMN-Q menggunakan nilai *precision* yang bernilai 98,75%. Karena pada kasus pencarian workflow dibutuhkan terjadinya TRUE POSITIF. Maka performa penggunaan BPMN-Q pada workflow repository berbasis graph database adalah sangat baik.

### 4.3 Integrasi Islam

Aplikasi ini membantu orang dalam menyimpan dan mencari *workflow*. *Workflow* sangat diperlukan bagi sebuah perusahaan/organisasi agar terorganisir. Tidak terjadi saling tumpang tindih dalam tugas antar anggota. Hal ini tertuang pada al-quran surat as-shaff ayat 4 yang berbunyi :

إِنَّ اللَّهَ يُحِبُّ الَّذِينَ يُقَاتِلُونَ فِي سَبِيلِهِ صَفًّا كَأَنَّهُمْ بُنْيَانٌ مَّرصُومٌ

Artinya: Sesungguhnya Allah menyukai orang yang berperang di jalan-Nya dalam barisan yang teratur seakan-akan mereka seperti suatu bangunan yang tersusun kokoh.

Dalam ayat ini, Allah memuji mereka yang berjuang dengan barisan yang teratur dan kesatuan yang kuat. Allah mencintai Muslim seperti itu. Tidak ada retakan kecil pun, seperti dinding kokoh yang ditumpuk rapat dengan batu semen. Ayat ini mengimbau kepada umat Islam bahwa mereka harus tetap kuat dan bersatu, memiliki jiwa yang luhur, dan suka berjuang dan berkorban.

Menjalin dan memelihara persatuan dan kesatuan sesama umat Islam berarti menyingkirkan segala sesuatu yang dapat menyebabkan perpecahan, seperti kurangnya toleransi, keegoisan, kesombongan terhadap ras dan keturunan, ketidakmanusiawian, dll. Oleh karena itu dalam hal mengedepankan persatuan dan kesatuan Islam, Allah SWT memerintahkan umat Islam untuk menjaga, menata dengan rapi berjalan dengan selamat dalam sholat (berbaris), dan merapatkan barisan dalam sholat agar tidak ada tempat yang kosong. Karena tempat kosong merupakan tempat setan, dan Setan merupakan musuh umat manusia. Tidaklah baik bagi seseorang untuk berdoa sendirian di belakang kantung udara kecuali jika orang

yang berada di dalam kantung udara itu ditarik ke belakang di depannya. Menyusun tim sholat merupakan kegiatan mengatur tim jihadis di jalan Allah.

## BAB 5

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Dalam *workflow repository* terdapat fitur pencarian. Pencarian ini digunakan untuk menemukan kembali BPMN yang telah disimpan. BPMN-Q digunakan pengguna untuk membuat *query* yang sesuai dengan BPMN yang dicari. Tidak hanya mencari menggunakan *query* melalui nama BPMN tetapi juga bisa membuat *query* sesuai dengan struktur BPMN.

Penggunaan *graph database* (NEO4J) pada *workflow repository* memberikan skema yang fleksibel. Selain itu NEO4J menyimpan data dalam bentuk *graph*. Hal ini mirip dengan BPMN yang mempunyai bentuk *graph*.

Hasil presentase *precision*, *recall*, *accuracy* yang dihasilkan secara berurutan adalah 98,75%, 90,33% dan 94,67%. Untuk penentuan performa BPMN-Q menggunakan nilai *precision* yang bernilai 98,75%. Karena pada kasus pencarian workflow dibutuhkan terjadinya TRUE POSITIF. Maka performa penggunaan BPMN-Q pada workflow repository berbasis *graph database* adalah sangat baik.

#### 5.2 Saran

Penelitian ini mempunyai beberapa kekurangan yang semoga dapat diselesaikan pada penelitian berikutnya. Pengujian *query* yang belum begitu rumit dan pada pengujian *query* dengan *negative path* terdapat hasil yang tidak sesuai. Belum adanya notasi yang lebih kompleks seperti notasi artifak, dokumen atau pool. Selain itu penelitian dapat dikembangkan pada pengecekan kepatuhan BPMN.

## DAFTAR PUSTAKA

- Abriani, G. U., & Yaqin, M. A. (2019). Analisis Implementasi Metode Semantic Similarity untuk Pengukuran Kemiripan Makna Antar Kalimat. *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, 1(2), 47–57. <https://doi.org/10.28926/ilkomnika.v1i2.15>
- Ardani, M., Yaqin, M. A., & Suhartono. (2019). Implementasi Graph Database untuk Menentukan Rute Perjalanan Transportasi Umum. *Seminar Nasional Inovasi Dan Aplikasi Teknologi Di Industri*, 36–42.
- Atmagi, R. W., & Sarno, R. (2014). Penggunaan Workflow Repository dan Kombinasi Metode Temu Kembali Berprioritas untuk Meningkatkan Performa Configurable-Process Di ERP. *Jurnal Ilmiah Mikrotek*, 1(2), 39–46.
- Awad, A. (2007). BPMN-Q: A language to query business processes. In M. Reichert, S. Strecker, & K. Turowski (Eds.), *Proceedings of the 2nd International Workshop on Enterprise Modelling and Information Systems Architectures - Concepts and Applications, EMISA 2007* (pp. 115–128). Gesellschaft für Informatik e.V. (GI). <https://dl.gi.de/handle/20.500.12116/22179>
- Chang, J. F. (2016). Business Process Management Systems. In *Business Process Management Systems: Strategy and Implementation* (1st ed.). Auerbach Publications. <https://doi.org/10.1201/9781420031362>
- Chengfei Liu, Xuemin Lin, Xiaofang Zhou, & Orłowska, M. (2003). Building a repository for workflow systems. *Proceedings Technology of Object-Oriented Languages and Systems (Cat. No.PR00393)*, 348–357. <https://doi.org/10.1109/tools.1999.796505>
- Dharmawan, I. N. P. W., & Sarno, R. (2017). Book recommendation using Neo4j graph database in BibTeX book metadata. *Proceeding - 2017 3rd International Conference on Science in Information Technology: Theory and Application of IT for Education, Industry and Society in Big Data Era, ICSITech 2017*, 47–52. <https://doi.org/10.1109/ICSITech.2017.8257084>
- Dominguez-Sal, D., Urbón-Bayes, P., Giménez-Vañó, A., Gómez-Villamor, S., Martínez-Bazán, N., & Larriba-Pey, J. L. (2010). Survey of Graph Database Performance on the HPC Scalable Graph Analysis Benchmark BT - Web-Age Information Management. In H. T. Shen, J. Pei, M. T. Özsu, L. Zou, J. Lu, T.-W. Ling, G. Yu, Y. Zhuang, & J. Shao (Eds.), *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 6185 LNCS* (pp. 37–48). Springer Berlin Heidelberg. [https://doi.org/10.1007/978-3-642-16720-1\\_4](https://doi.org/10.1007/978-3-642-16720-1_4)
- Handayani, P. W., Bressan, S., Han, D. K., & Boucelma, dan O. (2009). Implementasi Sistem Pengelolaan Proses Bisnis Mengacu pada MIT Process Handbook. *Jurnal Sistem Informasi*, 5(2), 106–113. <https://doi.org/10.21609/jsi.v5i2.270>
- Palmer, N. (2009). *Workflow Management Coalition BT - Encyclopedia of*

- Database Systems* (L. LIU & M. T. ÖZSU (eds.); p. 3550). Springer US. [https://doi.org/10.1007/978-0-387-39940-9\\_472](https://doi.org/10.1007/978-0-387-39940-9_472)
- Robinson, I., Webber, J., & Eifrem, E. (2014). Graph Databases. In M. Beaugureau (Ed.), *O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472* (2nd ed., pp. 27–46). O'Reilly Media, Inc.
- Sari, S. K., & Asniar, A. (2015). Analisis dan Pemodelan Proses Bisnis Prosedur Pelaksanaan Proyek Akhir sebagai Alat Bantu Identifikasi Kebutuhan Sistem. *Jurnal Informatika, Telekomunikasi Dan Elektronika*, 7(2), 143–152.
- Tague-Sutcliffe, J., & Blustein, J. (1994). A Statistical Analysis of the TREC-3 Data. *Proceedings of the Third Text Retrieval Conference (TREC-3)*, 385–389.
- Ter Hofstede, A. H. M., Ouyang, C., La Rosa, M., Song, L., Wang, J., & Polyvyanyy, A. (2017). APQL: A process-model query language. In M. Song, M. T. Wynn, & J. Liu (Eds.), *Lecture Notes in Business Information Processing* (Vol. 159, pp. 23–38). Springer International Publishing. [https://doi.org/10.1007/978-3-319-02922-1\\_2](https://doi.org/10.1007/978-3-319-02922-1_2)
- Wahyudi, & Akbar, F. (2019). Ekstraksi Basis Pengetahuan ke Dalam Basisdata Graf Menggunakan Graf Property. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 5(1), 41–48. <https://doi.org/10.25077/TEKNOSI.v5i1.2019.41-48>
- Wang, J., Jin, T., Wong, R. K., & Wen, L. (2014). Querying business process model repositories. *World Wide Web*, 17(3), 427–454. <https://doi.org/10.1007/s11280-013-0210-z>
- White, S. A. (2004). *Introduction to BPMN*. [Www.BPTrends.Com](http://www.bptrends.com). <https://www.bptrends.com/introduction-to-bpmn/>
- Yan, Z., Dijkman, R., & Grefen, P. (2012). Business process model repositories - Framework and survey. *Information and Software Technology*, 54(4), 380–395. <https://doi.org/10.1016/j.infsof.2011.11.005>