

ANALISIS DEVIASI MODEL PROSES BISNIS STANDAR DAN AKTUAL BERDASARKAN *DISTANCE MEASURE*: PERUSAHAAN PRODUSEN SEMEN

Ifrina Nuritha¹⁾, Mahendrawathi ER²⁾

¹⁾Program Studi Magister Sistem Informasi, Institut Teknologi Sepuluh Nopember

²⁾Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember

Jl. Jalan Raya ITS, Sukolilo, Surabaya, 60111

Telp : (031) 5999944, Fax : (031) 5964965

E-mail : ifrinanuritha@gmail.com¹⁾

Abstract

ERP implementation is typically accompanied by the company bussiness process change in order to comply with the standard best-practice of ERP package. However, companies tend to follow the previously run bussiness process which can lead to mismatches between standard and actual bussiness process. Companies need to evaluate ERP implementation to identify deviation between standard bussiness process model and actual bussiness process model which are executed in ERP system. This paper shows an evaluation case of standard and actual bussiness process of spareparts procurement bussiness process in cement manufacturing company. The evaluation is performed by measuring deviation value between standard and actual bussiness process based on distance measure calculation. This calculation uses dependency graph to represent bussiness process model. A dependency graph consists of nodes that represent activities in the bussiness process and edge points that represent depedency among activities. According to calculation result, deviation value that indicate deviations between dependency graph of standard bussiness process model and actual bussiness process model are around 13 edges.

Abstrak

Implementasi Enterprise Resource Planning (ERP) sering diikuti dengan perubahan proses bisnis perusahaan agar memenuhi standar best-practice paket ERP. Namun, perusahaan cenderung mengikuti proses bisnis yang telah berjalan sebelumnya, sehingga menyebabkan ketidaksesuaian antara proses bisnis standar dan aktual. Perusahaan membutuhkan evaluasi dalam implementasi ERP untuk mengetahui deviasi antara model proses bisnis standar dengan model proses bisnis aktual yang telah dieksekusi sistem ERP. Paper ini menampilkan sebuah kasus evaluasi proses bisnis standar dan aktual pada proses bisnis pengadaan spareparts di perusahaan Produsen Semen. Evaluasi dilakukan dengan mengukur nilai deviasi antara model proses bisnis standar dan aktual berdasarkan penghitungan distance measure. Penghitungan ini menggunakan dependency graph untuk merepresentasikan model proses bisnis. Dependency graph terdiri dari node yang menunjukkan aktivitas dalam proses bisnis dan edge yang menunjukkan ketergantungan antar aktivitas. Berdasarkan hasil penghitungan nilai deviasi, ditemukan 13 ketidaksesuaian hubungan antar aktivitas antara model proses bisnis standar dengan model proses bisnis aktual.

Kata kunci: *dependency graph, distance measure, ERP*

1. PENDAHULUAN

Enterprise Resource Planning (ERP) dipandang sebagai salah satu alternatif pilihan di bidang teknologi untuk meningkatkan proses bisnis. Sistem ERP menyediakan keuntungan bagi perusahaan untuk meningkatkan kinerja dengan memudahkan interaksi antar semua fungsi bisnis, sehingga setiap departemen dalam perusahaan dapat menyimpan dan mengambil informasi secara langsung melalui satu pusat data yang terintegrasi. Namun, kompleksitas

sistem ERP belum tentu menjamin keberhasilan pelaksanaan proses bisnis tersebut.

Implementasi ERP sering diikuti dengan perubahan proses bisnis pada perusahaan. Perusahaan diharapkan menjalankan proses bisnis yang terstandardisasi berdasarkan paket ERP untuk meningkatkan efektivitas produksi dan operasional. Perusahaan cenderung mengikuti proses bisnis yang telah berjalan sebelumnya, sehingga ada beberapa langkah yang ditambahkan atau bahkan tidak dijalankan dikarenakan tidak sesuai dengan proses bisnis awal perusahaan. Hal tersebut dapat

mengakibatkan implementasi ERP menjadi kurang maksimal, sehingga terjadi ketidaksesuaian antara proses bisnis yang distandardisasikan dengan yang telah direalisasikan.

Perusahaan membutuhkan evaluasi dalam implementasi ERP untuk mengidentifikasi perbedaan antara model proses bisnis standar yang sesuai paket ERP dengan model proses bisnis aktual yang telah dieksekusi sistem ERP berdasarkan kondisi nyata di operasional perusahaan. Keberhasilan ERP perlu dievaluasi dalam rangka menunjukkan bagian proses yang mengalami perbedaan terbesar dan memfasilitasi upaya-upaya perbaikan proses bisnis dalam perusahaan. [1] memperkirakan bahwa untuk paket ERP yang terbaik hanya memenuhi 80% dari kebutuhan fungsional perusahaan, sisanya 20% dari kebutuhan ini memerlukan adanya *Business Process Reengineering* (BPR) pada perusahaan sehingga gap yang ada dapat diminimalkan serta tidak berpengaruh pada operasional perusahaan. Evaluasi dalam implementasi ERP dapat dilakukan dengan mengidentifikasi perbedaan antara model proses bisnis standar dengan model proses bisnis aktual. Model proses bisnis standar menggambarkan prosedur standar perusahaan dalam menjalankan proses bisnis, sedangkan model proses bisnis aktual menggambarkan aktivitas bisnis yang telah dieksekusi perusahaan dengan sistem ERP. Salah satu pendekatan kuantitatif yang dapat digunakan untuk menangkap perbedaan diantara dua model proses adalah *distance measure* [2]. Paper ini menampilkan sebuah studi kasus pada proses bisnis pengadaan spareparts di perusahaan Produsen Semen, untuk menunjukkan analisis deviasi antara model proses standar dan aktual berdasarkan penghitungan *distance measure*.

Model proses bisnis standar diperoleh dengan menggambarkan prosedur standar dalam menjalankan proses dengan *Business Process Modelling Notation* (BPMN) yang kemudian ditransformasi menjadi *Petri net* berlabel. Sedangkan model proses bisnis aktual dalam *Petri net* berlabel dihasilkan dengan teknik *process mining* dengan mengekstrak *event log* yang direkam oleh sistem ERP menggunakan algoritma *process mining*. Ada banyak algoritma *process mining*, namun masing-masing algoritma tersebut memiliki karakteristik, properti, spesialisasi dan kinerja yang berbeda untuk proses bisnis yang berbeda [3].

Penelitian ini menggunakan algoritma Inductive Miner, karena algoritma ini bagus ketika berurusan dengan log yang berukuran besar dan mengandung *behavior* yang unik [4].

Karakteristik *event log* proses bisnis pengadaan spareparts, terdiri dari 2279 *case* di mana terdapat beberapa kasus dengan urutan aktivitas (*variant*) yang unik. Berdasarkan aplikasi Disco, *variant* yang terdapat pada *event log* proses bisnis pengadaan sparepart perusahaan Produsen Semen ditunjukkan pada Tabel 1. Selain itu, menurut [5], semua model yang dihasilkan oleh algoritma Inductive Miner memenuhi kriteria *sound* dan menghasilkan *work flow net* yang kompleks dan terstruktur. Model yang dihasilkan oleh algoritma ini juga selalu *fit*, yaitu, model dapat menghasilkan *trace* berdasarkan *event log*.

Tabel 1. *Variant* dalam *Event Log* Proses Bisnis Pengadaan Sparepart Perusahaan Semen

<i>Variant</i> Ke-	Urutan Aktivitas
1	Create Purchase Requisition → Approve Purchase Requisition → Create Request for Quotation → Create Purchase Order → Recieve Spareparts → Check GR into Blocked Stock (GR 103) → Release GR from Blocked Stock (GR 105) → End.
2	Create Purchase Requisition → Approve Purchase Requisition → Create Request for Quotation → Create Purchase Order → End.
3	Create Purchase Requisition → Approve Purchase Requisition → Create Request for Quotation → Create Purchase Order → Recieve Spareparts → Check GR into Blocked Stock (GR 103) → End.
4	Create Purchase Requisition → Approve Purchase Requisition → Create Request for Quotation → Create Purchase Order → Recieve Spareparts → Check GR into Blocked Stock (GR 103) → GR to Blocked Rev. (GR 104) → End.
5	Create Purchase Requisition → Approve Purchase Requisition → Create Purchase Order → Create Request for Quotation → Recieve Spareparts → Check GR into Blocked Stock (GR 103) → Release GR from Blocked Stock (GR 105) → End.
6	Create Purchase Requisition → Approve Purchase Requisition → Create Request for Quotation → Create Purchase Order → Recieve Spareparts → Check GR into Blocked Stock (GR 103) → Release GR from Blocked Stock (GR 105) → GR from Blocked Stock Rev. (GR 106) → GR from Blocked Stock Rev. (GR 106) → End.
7	Create Purchase Requisition → Approve Purchase Requisition → Create Request for Quotation → Create Purchase Order → Recieve Spareparts → Release GR from Blocked Stock (GR 105) → Check GR into Blocked Stock (GR 103) → End.

Penghitungan *distance measure* menggunakan *dependency graph* yang diekstrak dari *Petri net* berlabel model proses bisnis standar dan aktual. *Dependency graph* terdiri dari *node* yang menunjukkan aktivitas dalam proses bisnis dan *edge* yang menunjukkan ketergantungan antar aktivitas. *Distance measure* mensyaratkan adanya pengukuran *comparable* dari kedua *dependency graph* untuk menentukan apakah dua *dependency graph* layak atau dapat diperbandingkan. Analisis deviasi mengacu pada nilai hasil penghitungan *distance measure* yang menunjukkan jumlah ketidaksesuaian hubungan antar aktivitas diantara *dependency graph* model proses bisnis standar dan aktual.

2. KAJIAN LITERATUR

Bagian ini membahas mengenai kajian literatur yang mendukung dalam pengerjaan penelitian.

2.1 Pemodelan Proses Bisnis

Business Process Modelling (BPM) atau pemodelan proses bisnis, adalah kegiatan yang merepresentasikan proses dari suatu perusahaan, sehingga memungkinkan proses saat ini dapat dianalisis dan diperbaiki di masa depan [6]. *Business Process Modelling Notation* (BPMN) merupakan sebuah standar yang diterima secara luas untuk pemodelan proses [7].

2.2 Process Mining

Process mining bertujuan untuk menemukan model proses dari catatan kejadian penggunaan sistem informasi (*event log*) yang disimpan dalam basis data sistem informasi [8]. *Process mining* terdiri dari kumpulan teknik untuk menganalisis informasi yang tersimpan dalam *event log*, di mana analisis berfokus pada penemuan (*discovery*), pemantauan (*monitoring*) dan perbaikan (*enhancement*) proses [9].

2.3 Distance Measure

Distance measure diukur berdasarkan ekstraksi graf dari *Petri net*. Graf tersebut disebut *dependency graph* dalam Definisi 1 yang menyajikan ketergantungan antar aktivitas [2]. *Dependency graph* harus merupakan graf terhubung, dimana *self-edge* tidak diijinkan karena *edge* dimaksudkan untuk menunjukkan ketergantungan antar aktivitas yang berbeda.

Definisi 1. (*Dependency Graph*, DG)

Sebuah *dependency graph* DG didefinisikan dengan sepasang tuple $\langle DN, DE \rangle$, dimana:

- $DN = \{nd_1, nd_2, \dots, nd_n\}$ adalah himpunan berhingga dari node aktivitas dimana $n \geq 1$.

- $DE = \{e_1, e_2, \dots, e_m\}$ adalah himpunan *edge*, $m \geq 0$. Setiap *edge* membentuk $nd_i \rightarrow nd_j$

Dua *dependency graph* dikatakan identik jika dua graf memiliki himpunan *node* yang sama dan himpunan *edge* yang sama. Pengukuran *comparable* dari kedua *dependency graph* untuk menentukan apakah dua *dependency graph* layak untuk dibandingkan adalah dengan menghitung δ – *comparability* berdasarkan Definisi 2.

Definisi 2. (δ -Comparability dari DG)

Misalkan $DG_1 = (DN_1, DE_1)$ dan $DG_2 = (DN_2, DE_2)$ adalah dua *dependency graph*, dan δ adalah kontrol threshold yang ditetapkan pengguna. DG_1 dan DG_2 merupakan δ – *comparable* jika memenuhi $\frac{|DN_1 \cap DN_2|}{|DN_1 \cup DN_2|} \geq \delta$, dimana $0 < \delta \leq 1$.

Salah satu cara untuk membandingkan *dependency graph* adalah dengan mengubah setiap *dependency graph* menjadi representasi numerik. Matrik proses pada Definisi 3 digunakan sebagai representasi numerik dari *dependency graph*.

Definisi 3. (Matrik Proses, M)

Misalkan $g = (DN, DE)$ adalah sebuah *dependency graph* dengan $|DN| = n$ nodes. Sebuah matrik proses M dari g merupakan matrik n-dari-n dengan n baris dan n kolom, dan setiap baris dinamai nama node. Misalkan $M_g(i, j)$ merupakan nilai dari baris i^{th} dan kolom j^{th} dalam M, $1 \leq i, j \leq n$. Maka $M_g(i, j)$ didefinisikan sebagai berikut :

$$M_g(i, j) = \begin{cases} 1 & \text{jika terdapat } nd_i, nd_j \in DN \\ & \text{sedemikian hingga } (nd_i, nd_j) \in DE \\ 0 & \text{lainnya} \end{cases}$$

Matrik proses tersebut selanjutnya dinormalisasi berdasarkan Definisi 4 untuk menyamakan jumlah ordo pada kedua matrik proses. Matrik proses yang telah dinormalisasi akan menghasilkan matrik normalisasi.

Definisi 4. (Matrik Normalisasi, NM)

Misalkan $DG_1 = (DN_1, DE_1)$ dan $DG_2 = (DN_2, DE_2)$ adalah dua *dependency graph*. Misalkan NM_1 dan NM_2 merupakan matrik normalisasi untuk DG_1 dan DG_2 secara berurutan. Hasil NM_1 dan NM_2 dari DG_1 dan DG_2 adalah sebagai berikut:

- Jumlah baris dan kolom dihitung dengan $m = |DN_1 \cup DN_2|$
- Misalkan $DN_1 \cup DN_2 = \{a_1, a_2, \dots, a_m\}$. Nama baris dan kolom NM_1 dan NM_2 dinormalisasi kedalam nama node yang sama a_1, a_1, \dots, a_m berdasarkan

elemen pada himpunan gabungan DN_1 dan DN_2 .

- iii. Misalkan $NM_1(i, j)$ merupakan nilai dari baris i^{th} dan kolom j^{th} di NM_1 dan $NM_2(i, j)$ merupakan nilai dari baris i^{th} dan kolom j^{th} di NM_2

$$NM_1(i, j) = \begin{cases} 1 & \text{jika } (a_i, a_j) \in DE_1 \\ 0 & \text{sebaliknya} \end{cases},$$

$$NM_2(i, j) = \begin{cases} 1 & \text{jika } (a_i, a_j) \in DE_2 \\ 0 & \text{sebaliknya} \end{cases}.$$

Untuk mengukur perbedaan *dependency graph* antara dua model proses, dapat dilakukan penghitungan *Dependency Difference Metric* (d) sesuai dengan Definisi 5.

Definisi 5. (*Depedency Difference Metric, d*)
Misalkan $DG_1 = (DN_1, DE_1)$ dan $DG_2 = (DN_2, DE_2)$ adalah dua *dependency graph*. Misalkan NM_1 dan NM_2 adalah matriks normalisasi dari DG_1 dan DG_2 secara berurutan. Matriks perbedaan simetris pada graf DG_1 dan DG_2 ditetapkan dengan menghitung trace dari perbedaan matriks NM_1 dan NM_2 sebagai berikut :
 $d(DG_1, DG_2) = \text{tr}[(NM_1 - NM_2) \times (NM_1 - NM_2)^T]$
Dimana $\text{tr}[]$ merupakan trace dari sebuah matriks, yaitu penjumlahan elemen-elemen diagonal utama.

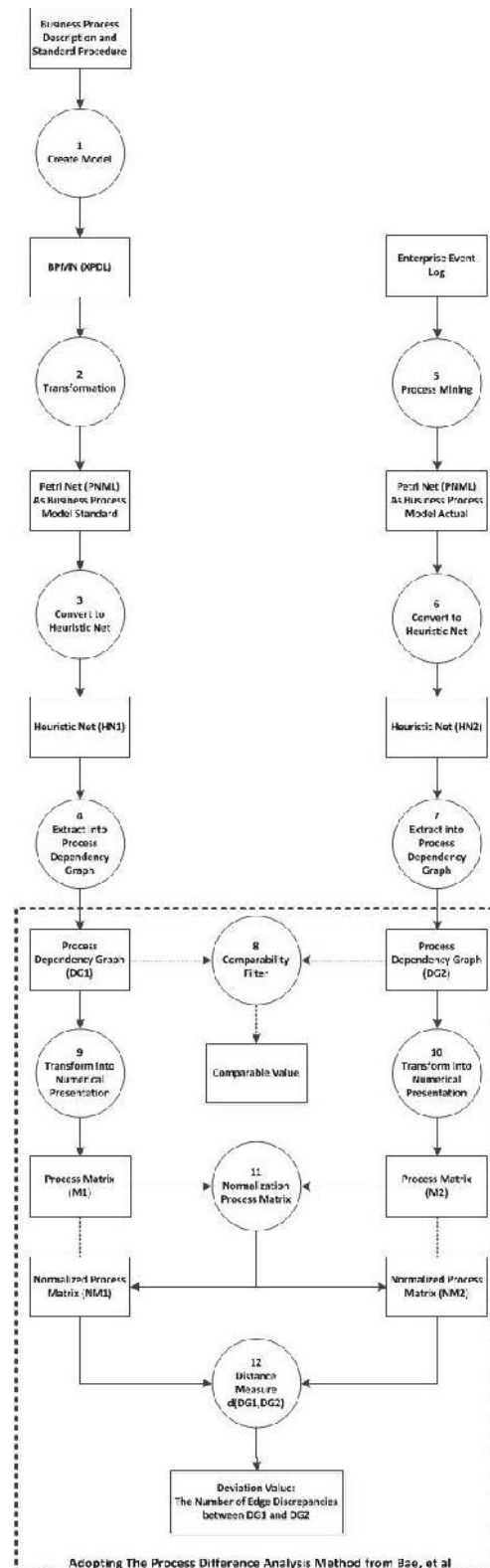
3. METODOLOGI

Penelitian ini mengadopsi metode analisis perbedaan model proses untuk mengevaluasi jarak antara dua model proses yang dikembangkan oleh [2] dan menambah beberapa langkah-langkah pengerjaan seperti pada Gambar 1 dengan maksud menyesuaikan kondisi perusahaan yang umumnya belum memiliki model proses.

Berdasarkan Gambar 1, metodologi dalam penelitian ini terdiri dari 12 langkah pengerjaan, diantaranya adalah:

1. *Create Model*, yaitu memodelkan prosedur standar proses bisnis dengan BPMN.
2. *Transformation*, yaitu mengubah BPMN ke dalam *Petri net* yang merepresentasikan model proses bisnis standar.
3. *Convert to Heuristic net*, yaitu mengkonversi *Petri net* model proses bisnis standar ke dalam *Heuristic net* (HN_1).
4. *Extract into Process Dependency Graph*, yaitu menggambarkan *dependency graph* (DG_1) sesuai dengan Definisi 1 berdasarkan *Heuristic net* dari model proses bisnis standar (HN_1).
5. *Process Mining*, yaitu memodelkan *event log* perusahaan dengan teknik *process mining* menggunakan algoritma Inductive

Miner untuk menghasilkan *Petri net* yang merepresentasikan model proses bisnis aktual.



Gambar 1. Metodologi Penelitian

6. *Convert to Heuristic net*, yaitu mengkonversi *Petri net* model proses bisnis aktual ke dalam *Heuristic net* (HN₂).
7. *Extract into Process Dependency Graph*, yaitu menggambarkan *dependency graph* (DG₂) sesuai dengan Definisi 1 berdasarkan *Heuristic net* dari model proses bisnis aktual (HN₁).
8. *Comparability Filter*, yaitu menentukan apakah dua *dependency graph* (DG₁ dan DG₂) layak untuk dibandingkan dengan menghitung δ -*Comparability* berdasarkan Definisi 2.
9. *Transform into Numerical Presentation*, yaitu merepresentasikan *dependency graph* model proses bisnis standar (DG₁) ke dalam matrik proses (M₁) berdasarkan Definisi 3.
10. *Transform into Numerical Presentation*, yaitu merepresentasikan *dependency graph* model proses bisnis aktual (DG₂) ke dalam matrik proses (M₂) berdasarkan Definisi 3.
11. *Normalization Process Matrix*, yaitu menyamakan ordo pada kedua matrik proses (M₁ dan M₂) untuk menghasilkan matrik normalisasi (NM₁ dan NM₂) berdasarkan Definisi 4.
12. *Distance Measure*, yaitu mengukur jarak antara kedua matrik normalisasi (NM₁ dan NM₂) dengan penghitungan *Dependency Difference Metric* (d) dalam Definisi 5.

4. HASIL dan ANALISIS

Bagian ini menjelaskan mengenai hasil yang diperoleh berdasarkan implementasi metodologi penelitian dan analisis deviasi dari hasil yang didapatkan.

4.1 Hasil

Implementasi metodologi telah dilakukan dan memberikan berbagai hasil. Hasil penelitian dipaparkan pada masing-masing sub bagian berikut ini sesuai dengan langkah-langkah dalam metodologi.

4.1.1 Pemodelan Proses Bisnis Standar dengan BPMN

Perusahaan Produsen Semen menjalankan proses bisnis pengadaan spareparts dengan didukung software ERP yaitu SAP. Prosedur standar proses bisnis pengadaan spareparts pada perusahaan Produsen Semen berfokus pada aktivitas *Purchase Requisition*, *Request for Quotation*, *Purchase Order*, dan *Good Receipt*. Berikut adalah deskripsi dari masing-masing aktivitas tersebut.

1. *Purchase Requisition*

Purchase Requisition (PR) merupakan dokumen internal yang berisi permintaan material dalam jumlah tertentu atau service dalam waktu

tertentu. Setiap PR yang dibuat akan diproses melalui tiga cara yakni PR dapat disetujui (*release*), dikembalikan atau ditolak. Persetujuan PR ini memiliki dua cara, yaitu *individual release* dimana setiap PR yang dibuat akan disetujui satu persatu dan *collective release* yaitu beberapa PR yang ada akan digabungkan menjadi satu persetujuan. Pihak yang melakukan persetujuan adalah kepala seksi, kepala biro, dan kepala departemen untuk material non rutin sedangkan untuk material rutin hanya kepala seksi.

2. *Request for Quotation*

Request for Quotation (RFQ) adalah undangan kepada vendor melalui *Purchasing Organization* untuk mengajukan penawaran pemasokan material atau service. Vendor akan menanggapi undangan tersebut dengan dokumen yang dinamakan *quotation*. Kedua dokumen RFQ dan *quotation* akan diidentifikasi dengan satu nomor dokumen pada SAP.

Setiap pembelian diatur oleh *Purchasing Organization*. *Purchasing Organization* merupakan unit organisasi yang bertanggung jawab terhadap negosiasi kondisi pembelian untuk sejumlah plant dan lokasi. *Purchasing Organization* untuk pembelian material diatur oleh Unit Produksi. Setelah mengatur kondisi pembelian di sejumlah plant dan lokasi, terdapat bagian *Purchasing Group* yang bertanggung jawab terhadap setiap transaksi yang dilakukan ke vendor terhadap pengadaan satu atau sekelompok material. Material bahan masuk kedalam *Purchasing Group G01*, untuk spareparts dan bahan ATK masuk kedalam *Purchasing Group G02* dan *G03*, untuk jasa rutin dan non rutin masuk kedalam *Purchasing Group G04* dan *G05*, dan untuk investasi masuk kedalam *Purchasing Group G06*.

RFQ pada perusahaan Produsen Semen ini dilakukan melalui *e-procurement* termasuk proses lelang. Perusahaan akan menawarkan spesifikasi material yang dibutuhkan. Pemilihan vendor dilakukan secara manual oleh pihak perusahaan berdasarkan kesesuaian, harga dan kualitas material. Perusahaan akan melakukan kontrak perjanjian pembelian material setelah vendor yang terpilih telah ditentukan.

3. *Purchase Order*

Purchase Order (PO) merupakan suatu permintaan atau instruksi dari *Purchasing Organization* kepada vendor (*external supplier*) atau plant untuk mengirimkan sejumlah material atau service pada waktu tertentu. Didalam PO terdapat identifikasi vendor, konfirmasi produk dan jasa yang dipesan, jumlah yang dibutuhkan, dan harga yang disetujui. PO dibuat pada saat pemesanan telah dikonfirmasi.

4. *Quality Inspection*

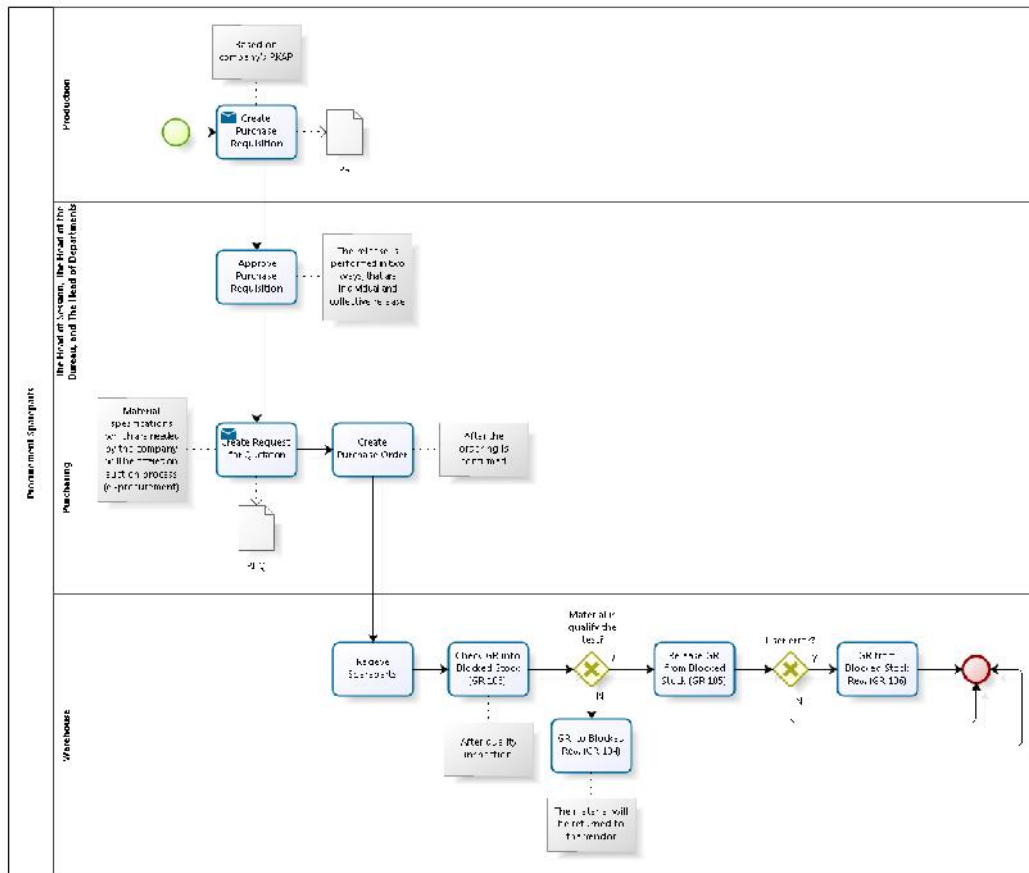
Proses *Quality Inspection* atau pemeriksaan kualitas digunakan untuk memastikan bahwa material yang diterima di dalam gudang adalah material yang telah memenuhi standar kualitas yang telah ditetapkan. *Quality Inspection* terhadap material spareparts dilakukan secara manual oleh pihak tertentu dengan melihat kondisi fisik dari material apakah telah sesuai dengan yang telah dipesan.

5. Good Receipt

Good Receipt adalah proses penerimaan material spareparts yang dilakukan oleh gudang. Ketika melakukan *Quality Inspection*, material akan dicek secara manual dan dapat dicatat di *Good Receipt* dengan *movement type 103* yaitu *check GR into blocked stock*, apabila material tidak lolos uji maka akan dikembalikan ke vendor dan dicatat di *Good Receipt* dengan *movement type 104* (*GR into blocked reversal*). Material yang telah lolos uji atau diterima akan dicatat di *Good Receipt* dengan *movement type 105* (*release GR from blocked stock*), namun apabila material batal diterima karena kesalahan user pada saat melakukan *entry data*, maka

proses *Good Receipt from blocked stock* dibatalkan dan akan dicatat di *GR 106* (*GR from blocked stock reversal*).

Model proses bisnis standar perusahaan ditentukan berdasarkan prosedur standar proses bisnis perusahaan dari *best-practice* paket ERP. Model proses bisnis standar perusahaan terlebih dahulu digambarkan dengan BPMN. Tujuan utama dari penggunaan BPMN adalah untuk memberikan notasi yang mudah dipahami oleh semua pengguna bisnis dalam mengidentifikasi dan memverifikasi urutan aktivitas yang terjadi dalam suatu proses bisnis. Penelitian ini menggunakan aplikasi Bizagi Modeler untuk memodelkan proses bisnis standar perusahaan dengan BPMN. Gambar 2 menggambarkan BPMN proses bisnis pengadaan spareparts perusahaan Produsen Semen yang terdiri dari 1 Start Event, 14 Sequence Flow, 7 Task, 2 Send Task, 2 Exclusive Gateway dan 1 End Event. Proses bisnis tersebut dieksekusi oleh Kepala Seksi, Kepala Biro, Kepala Departemen, departemen Production, Purchasing, dan Warehouse.



Gambar 2. BPMN Proses Bisnis Standar Pengadaan Spareparts di Perusahaan Produsen Semen

Send Task pada Gambar 2, terdiri dari *Create Purchase Requisition* dan *Create Request for Quotation*. *Create Purchase Requisition* merupakan aktivitas mengirim dokumen secara internal, sedangkan *Create Request for Quotation* merupakan aktivitas mengirim dokumen ke pihak eksternal. *Create Purchase Requisition* merupakan aktivitas dimana pihak produksi mengirimkan *Purchase Requisition* (PR) dan daftar vendor yang sesuai kepada pihak pengadaan. *Create Request for Quotation* merupakan aktivitas pengiriman *Request for Quotation* (RFQ) yaitu undangan untuk mengajukan penawaran pemasokan bahan yang ditujukan kepada vendor.

Pada Gambar 2 terdapat dua *Exclusive Gateway as Divergence* yang berperan untuk membuat dua jalur alternatif dalam proses, tetapi hanya satu yang dipilih. Jalur alternatif pertama menunjukkan kondisi bahwa apabila spareparts memenuhi uji kualitas, maka aktivitas *Release GR from Blocked Stock (GR 105)* akan dieksekusi. Namun, apabila tidak memenuhi uji kualitas, aktivitas *GR to Blocked Rev. (GR 104)* akan dieksekusi untuk membatalkan proses *Check GR into Blocked Stock (GR 103)*. Jalur alternatif kedua menunjukkan kondisi bahwa apabila terjadi kesalahan user seperti kesalahan saat melakukan *entry* data, maka aktivitas *GR from Blocked Stock Rev. (GR 106)* akan dieksekusi untuk membatalkan proses *Release GR from Blocked Stock (GR 105)*. Namun, apabila tidak terjadi kesalahan user, maka proses pengadaan sparepart pada perusahaan tersebut dinyatakan telah selesai.

4.1.2 Transformasi BPMN ke *Petri net* Berlabel

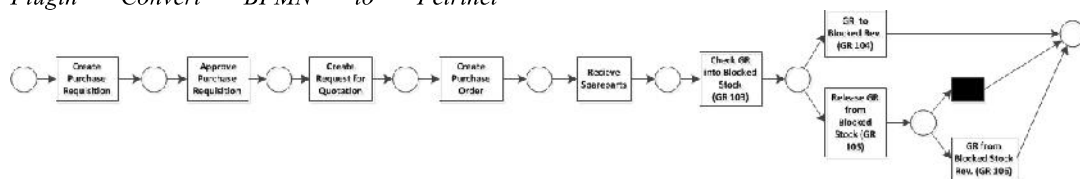
Transformasi untuk mengubah BPMN dalam format XPDN ke dalam bentuk *Petri net* dengan format PNML dilakukan dengan menggunakan *plugin Convert BPMN to Petrinet* pada ProM 6.5.1. *Plugin* tersebut tersedia dalam *package BPMN Miner* yang dikembangkan oleh [10]. *Plugin Convert BPMN to Petrinet*

mengimplementasikan algoritma pemetaan *task*, *event*, dan *gateway* dari BPMN ke *Petri net* yang dibangun oleh [11]

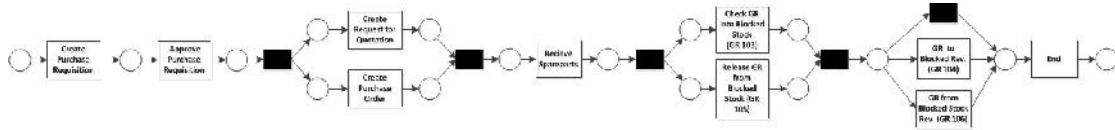
Gambar 3 adalah hasil transformasi model proses bisnis standar pengadaan spareparts dari BPMN pada Gambar 2 ke dalam bentuk *Petri net*. Berdasarkan Gambar 3, dapat diketahui bahwa *Start Event* pada BPMN setelah ditransformasi berubah menjadi *Start Place* pada *Petri net*, *Task* pada BPMN setelah ditransformasi berubah menjadi *Transition* pada *Petri net*, *Exclusive Gateway As Divergence* pada BPMN setelah ditransformasi berubah menjadi *XOR Split* pada *Petri net*, sedangkan *End Event* pada BPMN setelah ditransformasi berubah menjadi *End Place* pada *Petri net*. *Petri net* model proses bisnis pengadaan spareparts perusahaan Produsen Semen pada Gambar 3 terdiri dari 9 *Places*, 20 *Arcs*, dan 10 *Transitions* yang salah satunya merupakan *invisible task* yang ditandai dengan *Transitions* berwarna hitam. Hasil transformasi tersebut diekspor dalam format *Extended Petri net Markup Language* (EPNML) dan *Petri net Markup Language* (PNML).

4.1.3 Pemodelan Proses Bisnis Aktual dengan Teknik *Process Mining*

Pemodelan proses bisnis dari *event log* perusahaan digunakan untuk menghasilkan model proses bisnis aktual. Pemodelan ini dilakukan dengan teknik *process mining* menggunakan algoritma Inductive Miner. Pembentukan model proses bisnis aktual dari *event log* pengadaan sparepart menggunakan algoritma Inductive Miner dilakukan dengan menggunakan *plugin Mine Petri net with Inductive Miner* pada ProM 6.5.1. Penelitian ini menggunakan pengaturan *default* yang telah ditetapkan oleh *plugin* tersebut. Hasil pemodelan proses bisnis aktual dengan menggunakan algoritma Inductive Miner berupa *Petri net* berlabel yang ditunjukkan pada Gambar 4.



Gambar 3. *Petri net* Berlabel Proses Bisnis Standar Pengadaan Spareparts di Perusahaan Produsen Semen

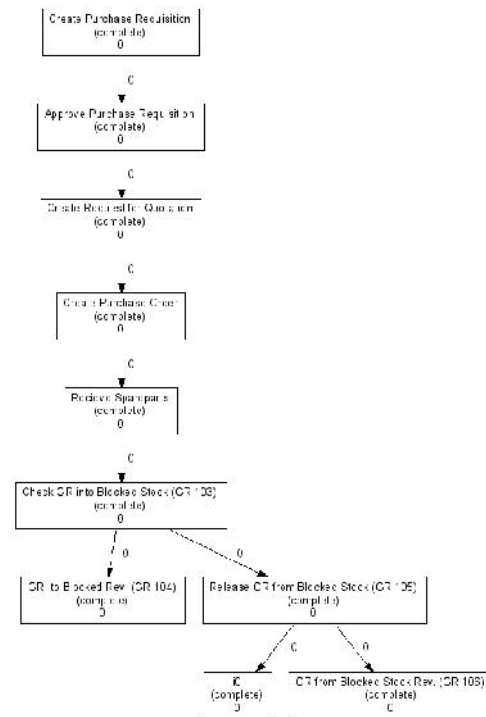


Gambar 4. Petri net Berlabel Proses Bisnis Aktual Pengadaan Spareparts di Perusahaan Produsen Semen

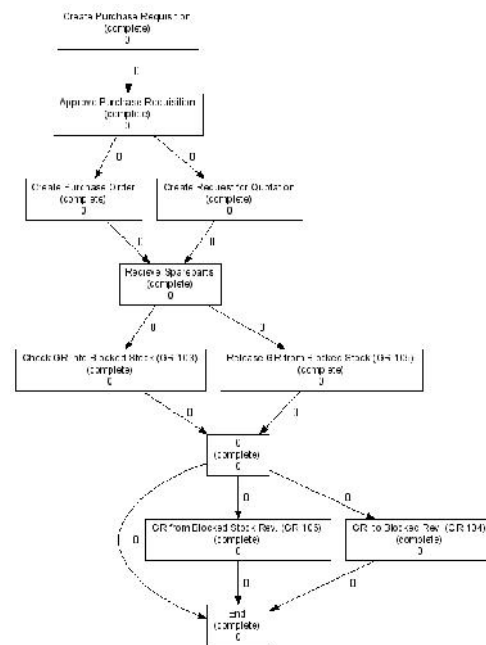
Petri net model proses bisnis pengadaan spareparts perusahaan Produsen Semen pada Gambar 4 terdiri dari 16 *Places*, 34 *Arcs*, dan 15 *Transitions* yang diantaranya terdiri dari 10 *visible task* dan 5 *invisible task* yang ditandai dengan *Transitions* berwarna hitam. Hasil pemodelan tersebut juga dapat diekspor dalam format EPNML dan PNML.

4.1.4 Konversi *Petri net* Berlabel ke *Heuristic net*

Konversi untuk mengubah *Petri net* berlabel dalam format EPNML ke dalam bentuk *Heuristic net* dengan format HN dilakukan dengan menggunakan *plugin Conversion: Petri net to Heuristic net* pada ProM 5.2. *Plugin* tersebut mengasumsikan bahwa: (a) *Petri net* yang akan dikonversi hanya boleh memiliki transisi yang tak terlihat (*invisible task*) dengan tidak lebih dari satu busur *input/output*. Transisi *invisible* yang melanggar ketentuan ini secara otomatis akan dibuat menjadi terlihat (*visible*); (b) *Petri net* yang akan dikonversi hanya boleh memiliki transisi terlihat (*visible*) sebagai *start* atau *end*. Dengan demikian, jika ada transisi *invisible* yang terhubung dengan *start place* atau *end place*, maka transisi yang tidak terlihat (*invisible*) tersebut secara otomatis berubah menjadi salah satu yang terlihat (*visible*).



Gambar 5. Heuristic net Proses Bisnis Standar Pengadaan Spareparts di Perusahaan Produsen Semen (HN₁)



Gambar 6. Heuristic net Proses Bisnis Aktual Pengadaan Spareparts di Perusahaan Produsen Semen (HN₂)

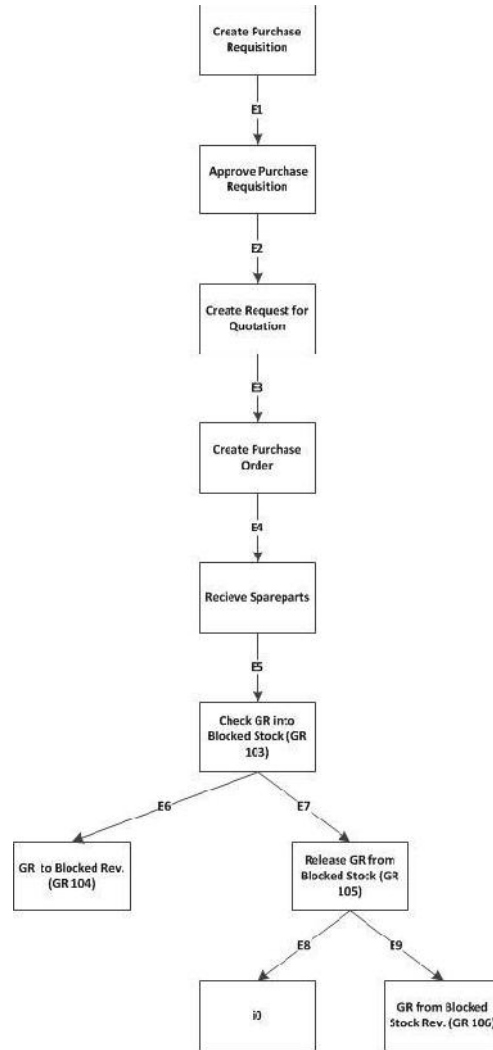
Penelitian ini memberikan batasan bahwa apabila terdapat transisi *invisible* pada *Petri net* yang akan dikoversi ke *Heuristic net* dan transisi tersebut menjadi *visible* karena tidak memenuhi asumsi diatas, maka transisi tersebut diberi label sebagai *i0*. Gambar 5 dan 6 berturut-turut adalah hasil konversi *Petri net* berlabel ke dalam bentuk *Heuristic net* yang menggambarkan model proses bisnis standar dan aktual.

4.15 Ekstraksi *Heuristic net* ke dalam *Dependency Graph*

Distance measure diukur berdasarkan ekstraksi graf dari model proses bisnis aktual dan standar. Graf tersebut disebut *dependency graph* dalam Definisi 1 yang menyajikan ketergantungan antar aktivitas. Menurut [12] *Heuristic net* dari suatu *Petri net* merupakan *dependency graph* dari *Petri net* tersebut. Oleh karena itu, ekstraksi *dependency graph* pada penelitian ini berdasarkan pada hasil konversi *Petri net* model proses bisnis standar dan aktual ke dalam *Heuristic net* pada Gambar 5 dan 6.

Menurut [2] pada perumusan *dependency graph* ada beberapa hal yang harus dipenuhi, diantaranya *self-edge* tidak diijinkan karena *edge* dimaksudkan untuk menunjukkan *data flow dependencies* antara *activity (node)* yang berbeda. Selain itu, *dependency graph* harus merupakan graf terhubung. *Node* yang tidak terhubung dan kelompok *node* yang terisolasi tidak diijinkan dalam graph, karena *node* atau kelompok *node* yang terisolasi dianggap sebagai proses service yang terpisah dalam model.

Dependency graph pada Gambar 7 dan 8 berturut-turut merupakan hasil ekstraksi dari *Heuristic net* yang menggambarkan model proses bisnis standar pada Gambar 5 dan model proses bisnis aktual pada Gambar 6.

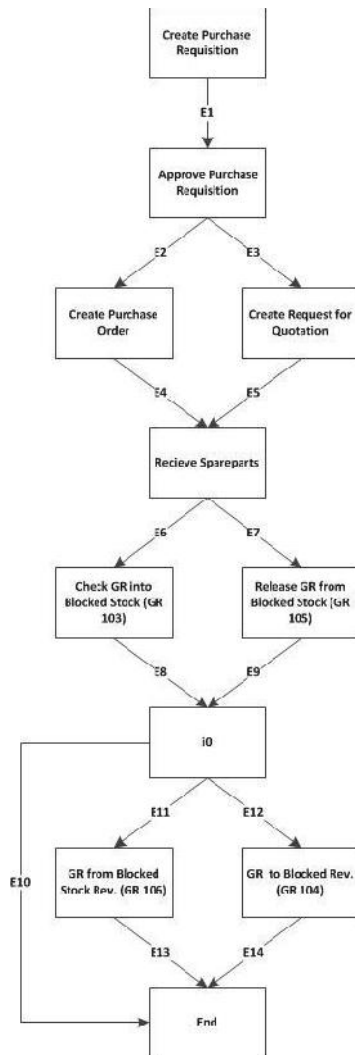


Gambar 7. *Dependency Graph* Proses Bisnis Standar Pengadaan Spareparts di Perusahaan Produsen Semen (DG_1)

Berdasarkan Definisi 1 maka *dependency graph* tersebut terdiri dari:

- $DN_{DG1} = \{Create\ Purchase\ Requisition, Approve\ Purchase\ Requisition, Create\ Request\ for\ Quotation, Create\ Purchase\ Order, Receive\ Spareparts, Check\ GR\ into\ Blocked\ Stock\ (GR\ 103), GR\ to\ Blocked\ Rev.\ (GR\ 104), Release\ GR\ from\ Blocked\ Stock\ (GR\ 105), i0, GR\ from\ Blocked\ Stock\ Rev.\ (GR\ 106)\}$
- $DE_{DG1} = \{E1, E2, E3, E4, E5, E6, E7, E8, E9\}$
- $DN_{DG2} = \{Create\ Purchase\ Requisition, Approve\ Purchase\ Requisition, Create\ Purchase\ Order, Create\ Request\ for\ Quotation, Receive\ Spareparts, Check\ GR\ into\ Blocked\ Stock\ (GR\ 103), Release\ GR\ from\ Blocked\ Stock\ (GR\ 105), i0, End, GR\ from\ Blocked\ Stock\ Rev.\ (GR\ 106), GR\ to\ Blocked\ Rev.\ (GR\ 104)\}$

- $DE_{DG2} = \{E1, E2, E3, E4, E5, E6, E7, E8, E9, E10, E11, E12, E13, 14\}$



Gambar 8. Dependency Graph Proses Bisnis Aktual Pengadaan Spareparts di Perusahaan Produsen Semen (DG_2)

4.1.6 Pengukuran Comparable dari Kedua Dependency Graph

Pengukuran *comparable* dari kedua *dependency graph* untuk menentukan apakah dua *dependency graph* layak untuk dibandingkan adalah dengan menghitung δ -Comparability berdasarkan Definisi 2. Kedua *dependency graph* pada Gambar 7 dan 8, masing-masing memiliki 10 aktivitas (*node*) yang sama, namun terdapat 1 aktivitas (*node*) dan *edge* yang berbeda. Berdasarkan Definisi 2 dan nilai δ (*threshold*) yang ditetapkan pada penelitian ini sebesar 0.5, maka δ -comparability pada kedua *graph* tersebut menunjukkan nilai *comparable*. Keduanya memiliki jumlah *node*

yang sama atau jumlah elemen dalam himpunan irisan sebanyak 10 dan jumlah total *node* atau jumlah elemen dalam himbunan gabungan

sebanyak 11, sehingga $\frac{|DN_1 \cap DN_2|}{|DN_1 \cup DN_2|} = \frac{10}{11} = 0.909$.

Dengan demikian, dua *dependency graph* tersebut memenuhi δ -comparability, karena $0.909 \geq \delta$, dan $\delta = 0.5$.

4.1.7 Transformasi Dependency Graph ke dalam Matrik Proses

Salah satu cara untuk membandingkan dan memeringkat *dependency graph* adalah dengan mengubah setiap *dependency graph* menjadi representasi numerik. Matrik pada Definisi 3 digunakan sebagai representasi numerik dari *dependency graph*. Berdasarkan Definisi 3, Gambar 9 dan 10 pada Lampiran menggambarkan transformasi *dependency graph* pada Gambar 7 dan 8 ke dalam matrik proses. M_1 memiliki ordo 10×10 , sedangkan M_2 memiliki ordo 11×11 . DG_1 dan DG_2 pada Gambar 7 dan 8, memiliki perbedaan *node*. DG_1 tidak memiliki *node End*, sedangkan DG_2 memiliki *node End*. Dengan demikian, diperlukan normalisasi pada setiap matrik proses tersebut.

4.1.8 Normalisasi pada Kedua Matrik Proses

Matrik proses M_1 dan M_2 yang memiliki ordo berbeda, akan dinormalisasi berdasarkan Definisi 4 menjadi matrik normalisasi untuk menyamakan jumlah ordo pada kedua matrik proses. Matrik normalisasi memiliki dimensi ordo yang sama dengan jumlah elemen pada himpunan gabungan dari himpunan *node* pada masing-masing *dependency graph*. Setelah dinormalisasi, kedua matrik proses memiliki jumlah baris dan kolom yang sama. Kedua matrik tersebut, NM_1 dan NM_2 kini memiliki ordo yang sama, yaitu 11×11 seperti yang ditunjukkan pada Gambar 11 dan 12 dalam Lampiran. Setiap *node* yang telah tersedia di dalam matrik proses sebelum dinormalisasi, maka nilai elemen yang bersesuaian dalam matrik normalisasi akan sama dengan pada matrik proses. Sedangkan setiap *node* yang ditambahkan melalui proses normalisasi, maka nilai elemen dalam matrik normalisasi akan ditetapkan dengan nilai 0.

4.1.9 Penghitungan Distance Measure

Dalam membandingkan *dependency graph* antara dua model proses, dapat dilakukan penghitungan *Dependency Difference Metric* (d) sesuai dengan Definisi 5. Matrik normalisasi NM_1 dan NM_2 digunakan sebagai input untuk menghitung *Dependency Different Matrix*

antara model proses bisnis standar dengan model proses bisnis aktual. *Dependency Different Matrix* tersebut dihitung berdasarkan *trace* dari matrik $(NM_1 - NM_2)(NM_1 - NM_2)^T$ yang berukuran $m \times m$. Matrik hasil perkalian antara matrik $(NM_1 - NM_2)$ dengan matrik *transpose*-nya, yaitu $(NM_1 - NM_2)(NM_1 - NM_2)^T$ ditunjukkan pada Gambar 13 dalam Lampiran. Matrik $(NM_1 - NM_2)(NM_1 - NM_2)^T$ yang berukuran 11×11 didefinisikan sebagai matrik bujur sangkar, oleh karena itu *trace* untuk menghitung *Dependency Different Matrix* pada matrik $(NM_1 - NM_2)(NM_1 - NM_2)^T$ dinotasikan $tr[(NM_1 - NM_2)(NM_1 - NM_2)^T]$, dihitung berdasarkan jumlah elemen diagonal utama pada matrik tersebut, sehingga:

$$\begin{aligned} d(DG_1, DG_2) &= \sum_{i=1}^m a_{ii} \\ &= \sum_{i=1}^{11} a_{ii} \\ &= 0+1+2+0+1+3+1+1+3+1+0 \\ &= 13 \end{aligned}$$

Dengan demikian berdasarkan penghitungan *Dependency Different Matrix*, maka nilai deviasi yang menunjukkan ketidaksesuaian jumlah *edge* antara *dependency graph* model proses bisnis standar dengan model proses bisnis aktual pengadaan spareparts di perusahaan Produsen Semen adalah 13.

4.2 Analisis

Berdasarkan hasil penghitungan *distance measure* dapat diketahui bahwa nilai deviasi pada proses bisnis pengadaan spareparts di perusahaan Produsen Semen adalah 13. Jika dilihat berdasarkan *dependency graph* pada Gambar 7 dan 8, jumlah *edge* pada *dependency graph* model proses bisnis standar adalah 9, sedangkan pada *dependency graph* model proses bisnis aktual adalah 14. Selain itu, jumlah *node* pada *dependency graph* model proses bisnis standar adalah 10, sedangkan pada *dependency graph* model proses bisnis aktual adalah 11. Perbedaan jumlah *node* tersebut disebabkan adanya aktivitas tambahan pada proses bisnis aktual yang tidak didefinisikan dalam proses bisnis standar. Pada *dependency graph* model proses bisnis aktual, terdapat *node End* yang menandai aktivitas terakhir pada setiap case. Ketidaksesuaian 13 *edge* antara *dependency graph* model proses bisnis standar dengan aktual yang didapatkan dari hasil penghitungan *distance measure*, diantaranya adalah sebagai berikut.

4.2.1 Ketidaksesuaian Hubungan antar Aktivitas *Approve Purchase Requisition*, *Create Request for Quotation*, dan *Create Purchase Order*

Pada DG_1 setelah aktivitas *Approve Purchase Requisition* dieksekusi selanjutnya aktivitas yang dieksekusi adalah *Create Request for Quotation*. Namun, pada DG_2 setelah aktivitas *Approve Purchase Requisition* dieksekusi selanjutnya terdapat percabangan 2 *edge* ke 2 *node* yang menunjukkan ada 2 aktivitas yang dieksekusi, yaitu *Create Purchase Order* dan *Create Request for Quotation*. Berdasarkan prosedur standar perusahaan, seharusnya aktivitas pembuatan RFQ dilakukan terlebih dahulu setelah itu baru dilakukan aktivitas berikutnya yaitu pembuatan PO. Namun pada kenyataannya, setelah aktivitas persetujuan PR dapat diikuti oleh aktivitas PO atau aktivitas pembuatan RFQ. Dengan demikian terdapat 2 *edge* pada DG_2 yang tidak sesuai dengan DG_1 .

4.2.2 Ketidaksesuaian Hubungan antar Aktivitas *Create Request for Quotation*, *Create Purchase Order* dan *Receive Spareparts*

Pada DG_1 aktivitas *Receive Spareparts* dieksekusi setelah aktivitas *Create Purchase Order*. Namun, pada DG_2 sebelum aktivitas *Receive Spareparts* dieksekusi, terdapat penggabungan 2 *edge* dari 2 *node* yang menunjukkan ada 2 aktivitas sebelumnya yang harus dieksekusi terlebih dahulu, yaitu *Create Purchase Order* dan *Create Request for Quotation* dimana yang satu bisa mendahului yang lain. Berdasarkan prosedur standar perusahaan, seharusnya aktivitas penerimaan spareparts terjadi setelah aktivitas pembuatan PO dilakukan. Namun pada kenyataannya, selain kondisi normal di mana aktivitas penerimaan spareparts dapat dilakukan setelah pembuatan PO yang didahului oleh pembuatan RFQ, ternyata ada kemungkinan aktivitas penerimaan sparepart terjadi setelah pembuatan RFQ yang sebelumnya didahului oleh pembuatan PO (*Variant ke-5* pada Tabel 1). Dengan demikian terdapat tambahan 2 *edge* lagi pada DG_2 yang tidak sesuai dengan DG_1 .

4.2.3 Ketidaksesuaian Hubungan antar Aktivitas *Receive Spareparts*, *Check GR into Blocked Stock (GR 103)*, dan *Release GR from Blocked Stock (GR 105)*

Pada DG_1 setelah aktivitas *Receive Spareparts* dieksekusi selanjutnya aktivitas yang dieksekusi adalah *Check GR into Blocked Stock (GR 103)*. Namun, pada DG_2 setelah aktivitas *Receive Spareparts* dieksekusi selanjutnya terdapat

percabangan 2 *edge* ke 2 *node* yang menunjukkan ada 2 aktivitas yang dieksekusi, yaitu *Check GR into Blocked Stock (GR 103)* dan *Release GR from Blocked Stock (GR 105)*. Berdasarkan prosedur standar perusahaan, seharusnya aktivitas *Check GR into Blocked Stock (GR 103)* dilakukan terlebih dahulu untuk melakukan inspeksi kualitas pada material spareparts sesuai dengan PO. Selanjutnya, terdapat percabangan dua jalur alternatif yang dapat dilewati, salah satunya merupakan aktivitas *Release GR from Blocked Stock (GR 105)* yang menandakan material spareparts lolos inspeksi kualitas. Namun pada kenyataannya, setelah aktivitas penerimaan spareparts terdapat pilihan untuk memulai aktivitas *Release GR from Blocked Stock (GR 105)* terlebih dahulu sebelum aktivitas *Check GR into Blocked Stock (GR 103)*. Dengan demikian terdapat tambahan 2 *edge* lagi pada DG₂ yang tidak sesuai dengan DG₁.

4.2.4 Ketidaksesuaian Hubungan antar Aktivitas *Check GR into Blocked Stock (GR 103)*, *Release GR from Blocked Stock (GR 105)*, dan *Invisible Task*

Pada DG₁ aktivitas *i0* merupakan salah satu dari dua jalur alternatif yang dapat dieksekusi setelah aktivitas *Release GR from Blocked Stock (GR 105)*. Namun, pada DG₂ sebelum aktivitas *i0* dieksekusi, terdapat penggabungan 2 *edge* dari 2 *node* yang menunjukkan ada 2 aktivitas sebelumnya yang harus dieksekusi terlebih dahulu, yaitu *Check GR into Blocked Stock (GR 103)* dan *Release GR from Blocked Stock (GR 105)*. Dengan demikian terdapat tambahan 2 *edge* lagi pada DG₂ yang tidak sesuai dengan DG₁.

4.2.5 Ketidaksesuaian Hubungan antar Aktivitas *GR to Blocked Rev. (GR 104)*, *GR from Blocked Stock Rev. (GR 106)*, dan *Invisible Task*

Pada DG₁ setelah aktivitas *invisible task (i0)* dieksekusi tidak terdapat aktivitas selanjutnya yang dieksekusi. Namun, pada DG₂ setelah aktivitas *i0* dieksekusi selanjutnya terdapat percabangan 3 *edge* ke 3 *node* yang menunjukkan ada 3 aktivitas yang dieksekusi, yaitu *GR from Blocked Stock Rev. (GR 106)*, *GR to Blocked Rev. (GR 104)*, dan *End*. Berdasarkan prosedur standar perusahaan, seharusnya aktivitas *Check GR into Blocked Stock (GR 103)* dilakukan terlebih dahulu untuk melakukan inspeksi kualitas pada material spareparts sesuai dengan PO. Selanjutnya, terdapat percabangan dua jalur alternatif yang dapat dipilih yaitu aktivitas *GR from Blocked Stock (GR 105)* atau *GR to Blocked Rev. (GR*

104) yang menandakan material spareparts tidak lolos uji dan akan dikembalikan ke vendor. Dengan adanya aktivitas *GR 105* yang mendahului *GR 103 (Variant ke-7 pada Tabel 1)* maka DG₂ menggambarkan bahwa aktivitas *GR to Blocked Rev. (GR 104)* dilakukan setelah aktivitas *i0*. Selain itu, berdasarkan prosedur standar perusahaan, *GR from Blocked Stock Rev. (GR 106)* dilakukan setelah terjadi pembatalan dari aktivitas sebelumnya yaitu *GR from Blocked Stock (GR 105)*. Hal ini diakibatkan karena adanya kesalahan *user* pada saat melakukan input data pada aktivitas *Release GR from Blocked Stock (GR 105)*. Namun dengan adanya data aktivitas *GR 105* mendahului *GR 103* maka DG₂ menunjukkan bahwa aktivitas *GR from Blocked Stock Rev. (GR 106)* dilakukan setelah aktivitas *i0*. Aktivitas *End* pada DG₂ salah satunya dieksekusi setelah aktivitas *i0*, namun berdasarkan prosedur standar perusahaan, aktivitas *End* tersebut tidak ada. Dengan demikian, terdapat tambahan 3 *edge* lagi pada DG₂ yang tidak sesuai dengan DG₁.

4.2.6 Ketidaksesuaian Hubungan antar Aktivitas *GR to Blocked Rev. (GR 104)*, *GR from Blocked Stock Rev. (GR 106)*, *Invisible Task*, dan *End*

Pada DG₁ tidak terdapat aktivitas *End* seperti yang ada pada DG₂, karena aktivitas *End* tidak terdapat dalam prosedur standar. Namun, pada DG₂ sebelum aktivitas *End* dieksekusi, terdapat penggabungan 3 *edge* dari 3 *node* yang menunjukkan ada 3 aktivitas sebelumnya yang harus dieksekusi terlebih dahulu, yaitu *i0*, *GR to Blocked Rev. (GR 104)*, dan *GR from Blocked Stock Rev. (GR 106)*. Berdasarkan standar perusahaan, tidak ada aktivitas selanjutnya yang dieksekusi setelah aktivitas *GR to Blocked Rev. (GR 104)* maupun aktivitas *GR from Blocked Stock Rev. (GR 106)*. Namun, pada kenyataannya, ada aktivitas *End* yang menandai berakhirnya suatu *case*, apabila salah satu dari ketiga aktivitas tersebut selesai dieksekusi. Dengan demikian, terdapat tambahan 2 *edge* lagi pada DG₂ yang tidak sesuai dengan DG₁. Satu *edge* dari aktivitas *i0* menuju aktivitas *End* telah disebutkan sebelumnya.

5. SIMPULAN dan SARAN

Simpulan dari penelitian ini sebagai berikut:

1. Berdasarkan hasil penghitungan *distance measure* ditemukan 13 ketidaksesuaian hubungan antar aktivitas antara model proses bisnis standar dan aktual. Hal ini menunjukkan bahwa proses pengadaan spareparts di perusahaan Produsen Semen

- ini belum sepenuhnya memenuhi standar yang telah ditetapkan oleh perusahaan.
2. Berdasarkan hasil analisis deviasi, deviasi lebih banyak dipengaruhi oleh adanya ketidakkonsistenan dalam urutan pelaksanaan aktivitas percabangan dan penggabungan aktivitas, dimana pada proses bisnis standar seharusnya *Create RFQ* dilakukan setelah *Create PO* dan tidak dimungkinkan tidak ada percabangan dari *Approve PR* ke *Create RFQ*. Proses standar menunjukkan bahwa *Create RFQ* dilakukan setelah *Create PO*. Namun dalam kenyataannya dimungkinkan bahwa *Create RFQ* dilakukan setelah *Approve PR*, tanpa melewati *PO* terlebih dahulu. Hal ini menunjukkan ketidakkonsistenan yang perlu ditelusuri lebih lanjut penyebabnya.
 3. Deviasi juga dipengaruhi oleh adanya aktivitas *End* yang terdapat pada model proses bisnis aktual, dimana pada model proses bisnis standar seharusnya tidak memiliki aktivitas *End*. Proses standar menunjukkan bahwa setelah *invisible task*, *GR to Blocked Rev. (GR 104)*, dan *GR from Blocked Stock Rev. (GR 106)* dieksekusi tidak ada aktivitas selanjutnya yang dieksekusi. Namun dalam kenyataannya, terdapat aktivitas *End* yang dieksekusi untuk menandakan berakhirnya suatu *case*.

Pengembangan untuk penelitian selanjutnya, antara lain:

1. Penggunaan algoritma lain misalnya *Evolutionary Tree Miner* (ETM) dan *Integer Linear Programming* (ILP) dalam memodelkan proses bisnis aktual berdasarkan *event log perusahaan*.
2. Perbandingan antara hasil analisis deviasi berdasarkan *distance measure* dengan hasil analisis perbedaan *variant* yang terdapat pada prosedur standar dan *event log perusahaan*.

6. DAFTAR RUJUKAN

- [1] Leon, A. 2014. *Enterprise Resource Planning* (3rd ed.). New Delhi, India: McGraw Hill Education (India) Private Limited.
- [2] Bae, J., Liu, L., Caverlee, J., Zhan, L.-J., & Bae, H. 2007. Development of Distance Measures for Process Mining, Discovery, and Integration. *International Journal of Web Services Research*, 4(4), pp.1-17.
- [3] Wang, J., Wong, R. K., Ding, J., Guo, Q., & Wen, L. 2013. Efficient Selection Process Mining Algorithms. *IEEE Transactions on Services Computing*, 6 (4), pp.484-496.
- [4] Bolt, A., Leoni, M., & Aalst, W. 2015. Scientific Workflows for Process Mining: Building Blocks, Scenarios, and Implementation. *International Journal on Software Tools for Technology Transfer*.
- [5] Leemans, S. J., Fahland, D., & Aalst, W. M. 2013. Discovering Block-Structured Process. In: *34th International Conference, PETRI NETS 2013*. Milan, Italy 24-28 June, 2013. Springer Berlin Heidelberg.
- [6] Poonia, V. S. 2010. *Production and Operations Management*. Delhi: Gennext Publication.
- [7] Meland, P., & Gjaere, E. 2012. Representing Threats in BPMN 2.0. *2012 Seventh International Conference on Availability, Reliability and Security (ARES)* (pp. 542 - 550). Prague: IEEE.
- [8] Wang, D., Ge, J., Hu, H., Luo, B., & Huang, L. 2012. Discovering Process Model from Event Multiset. *Expert System with Application*, 39, 11970-11978.
- [9] Khodabandelou, G., Hug, C., Deneckère, R., & Salinesi, C. 2013. Process Mining Versus Intention Mining. *14th International Conference, BPMDS 2013, 18th International Conference, EMMSAD 2013*. 147, pp. 466-480. Valencia: Springer Berlin Heidelberg.
- [10] Conforti, R., Dumas, M., Iacobanuelos, L. G., & Rosa, M. L. (2015, March 31). BPMN Miner: Automated Discovery of BPMN Process Models with Hierarchical Structure. *Information System*, 56, pp.284-303.
- [11] Dijkman, R. M., Dumas, M., & Ouyang, C. 2008. Semantics and analysis of business process models in BPMN. *Information and Software Technology*, 50(12), pp.1281-1294.
- [12] Burattin, A. 2015. *Process Mining Techniques in Business Environment: Theoretical Aspects, Algorithms, Techniques and Open Challenges in Process Mining*. Springer International Publishing.

Lampiran

	Create Purchase Requisition	Approve Purchase Requisition	Create Request for Quotation	Create Purchase Order	Recieve Spareparts	Check GR into Blocked Stock (GR 103)	GR to Blocked Rev. (GR 104)	Release GR from Blocked Stock (GR 105)	i0	GR from Blocked Stock Rev. (GR 106)
Create Purchase Requisition	0	1	0	0	0	0	0	0	0	0
Approve Purchase Requisition	0	0	1	0	0	0	0	0	0	0
Create Request for Quotation	0	0	0	1	0	0	0	0	0	0
Create Purchase Order	0	0	0	0	1	0	0	0	0	0
Recieve Spareparts	0	0	0	0	0	1	0	0	0	0
Check GR into Blocked Stock (GR 103)	0	0	0	0	0	0	1	1	0	0
GR to Blocked Rev. (GR 104)	0	0	0	0	0	0	0	0	0	0
Release GR from Blocked Stock (GR 105)	0	0	0	0	0	0	0	0	1	1
i0	0	0	0	0	0	0	0	0	0	0
GR from Blocked Stock Rev. (GR 106)	0	0	0	0	0	0	0	0	0	0

Gambar 9. Matrik Proses dari Dependency Graph Proses Bisnis Standar Pengadaan Spareparts (M_1)

	Create Purchase Requisition	Approve Purchase Requisition	Create Request for Quotation	Create Purchase Order	Recieve Spareparts	Check GR into Blocked Stock (GR 103)	GR to Blocked Rev. (GR 104)	Release GR from Blocked Stock (GR 105)	i0	GR from Blocked Stock Rev. (GR 106)	End
Create Purchase Requisition	0	1	0	0	0	0	0	0	0	0	0
Approve Purchase Requisition	0	0	1	1	0	0	0	0	0	0	0
Create Request for Quotation	0	0	0	0	1	0	0	0	0	0	0
Create Purchase Order	0	0	0	0	1	0	0	0	0	0	0
Recieve Spareparts	0	0	0	0	0	1	0	1	0	0	0
Check GR into Blocked Stock (GR 103)	0	0	0	0	0	0	0	0	1	0	0
GR to Blocked Rev. (GR 104)	0	0	0	0	0	0	0	0	0	0	1
Release GR from Blocked Stock (GR 105)	0	0	0	0	0	0	0	0	1	0	0
i0	0	0	0	0	0	0	1	0	0	1	1
GR from Blocked Stock Rev. (GR 106)	0	0	0	0	0	0	0	0	0	0	1
End	0	0	0	0	0	0	0	0	0	0	0

Gambar 10. Matrik Proses dari Dependency Graph Proses Bisnis Aktual Pengadaan Spareparts (M_2)

	Create Purchase Requisition	Approve Purchase Requisition	Create Request for Quotation	Create Purchase Order	Recieve Spareparts	Check GR into Blocked Stock (GR 103)	GR to Blocked Rev. (GR 104)	Release GR from Blocked Stock (GR 105)	i0	GR from Blocked Stock Rev. (GR 106)	End
Create Purchase Requisition	0	1	0	0	0	0	0	0	0	0	0
Approve Purchase Requisition	0	0	1	0	0	0	0	0	0	0	0
Create Request for Quotation	0	0	0	1	0	0	0	0	0	0	0
Create Purchase Order	0	0	0	0	1	0	0	0	0	0	0
Recieve Spareparts	0	0	0	0	0	1	0	0	0	0	0
Check GR into Blocked Stock (GR 103)	0	0	0	0	0	0	1	1	0	0	0
GR to Blocked Rev. (GR 104)	0	0	0	0	0	0	0	0	0	0	0
Release GR from Blocked Stock (GR 105)	0	0	0	0	0	0	0	0	1	1	0
i0	0	0	0	0	0	0	0	0	0	0	0
GR from Blocked Stock Rev. (GR 106)	0	0	0	0	0	0	0	0	0	0	0
End	0	0	0	0	0	0	0	0	0	0	0

Gambar 11. Matrik Normalisasi dari Matrik Proses M_1 (NM_1)

	Create Purchase Requisition	Approve Purchase Requisition	Create Request for Quotation	Create Purchase Order	Recieve Spareparts	Check GR into Blocked Stock (GR 103)	GR to Blocked Rev. (GR 104)	Release GR from Blocked Stock (GR 105)	i0	GR from Blocked Stock Rev. (GR 106)	End
Create Purchase Requisition	0	1	0	0	0	0	0	0	0	0	0
Approve Purchase Requisition	0	0	1	1	0	0	0	0	0	0	0
Create Request for Quotation	0	0	0	0	1	0	0	0	0	0	0
Create Purchase Order	0	0	0	0	1	0	0	0	0	0	0
Recieve Spareparts	0	0	0	0	0	1	0	1	0	0	0
Check GR into Blocked Stock (GR 103)	0	0	0	0	0	0	0	0	1	0	0
GR to Blocked Rev. (GR 104)	0	0	0	0	0	0	0	0	0	0	1
Release GR from Blocked Stock (GR 105)	0	0	0	0	0	0	0	0	1	0	0
i0	0	0	0	0	0	0	1	0	0	1	1
GR from Blocked Stock Rev. (GR 106)	0	0	0	0	0	0	0	0	0	0	1
End	0	0	0	0	0	0	0	0	0	0	0

Gambar 12. Matrik Normalisasi dari Matrik Proses M_2 (NM_2)

	Create Purchase Requisition	Approve Purchase Requisition	Create Request for Quotation	Create Purchase Order	Recieve Spareparts	Check GR into Blocked Stock (GR 103)	GR to Blocked Rev. (GR 104)	Release GR from Blocked Stock (GR 105)	i0	GR from Blocked Stock Rev. (GR 106)	End
Create Purchase Requisition	0	0	0	0	0	0	0	0	0	0	0
Approve Purchase Requisition	0	1	-1	0	0	0	0	0	0	0	0
Create Request for Quotation	0	-1	2	0	0	0	0	0	0	0	0
Create Purchase Order	0	0	0	0	0	0	0	0	0	0	0
Recieve Spareparts	0	0	0	0	1	-1	0	0	0	0	0
Check GR into Blocked Stock (GR 103)	0	0	0	0	-1	3	0	0	-1	0	0
GR to Blocked Rev. (GR 104)	0	0	0	0	0	0	1	0	1	1	0
Release GR from Blocked Stock (GR 105)	0	0	0	0	0	0	0	1	-1	0	0
i0	0	0	0	0	0	-1	1	-1	3	1	0
GR from Blocked Stock Rev. (GR 106)	0	0	0	0	0	0	1	0	1	1	0
End	0	0	0	0	0	0	0	0	0	0	0

Gambar 13. Matrik Hasil Perkalian $(NM_1 - NM_2)(NM_1 - NM_2)^T$