

Implementasi Business Process Modeling dalam Workflow Aplikasi Monitoring Electronic Data Capturer

Umi Proboyekti¹, Charoline Septa², Budi Susanto^{3,*}

^{1,2}*Sistem Informasi, Universitas Kristen Duta Wacana
Jln. Dr Wahidin Sudirohusodo, Yogyakarta
othie@staff.ukdw.ac.id
charoline.septa@si.ukdw.ac.id*

³*Teknik Informatika, Universitas Kristen Duta Wacana
Jln. Dr Wahidin Sudirohusodo, Yogyakarta*

*Email korespondensi: budsus@ti.ukdw.ac.id

Abstract— Business Process Modeling (BPM) is a data-based visual representation to model business workflows to optimize business processes. BPM notations make the model easy to understand for all stakeholders who play a role in developing business processes. This paper will describe implementing BPM in the Electronic Data Capturer (EDC) Monitoring Application (EDCMA) workflow. The workflow modeling process uses Bonitasoft software that includes business process creation for user interface and process engine. EDCMA is a process to solve the problems in the EDC terminal repair process and the submission as a new merchant. Based on the existing manual procedure of EDCMA, the new EDCMA process model uses electronic form and semi-automatic processing that implement some Bonitasoft BPM features to develop a high-fidelity application prototype. In the evaluation process, we develop some scenarios with data tests that occur in manual procedures. Based on the evaluation, the prototype has successfully represented the entire process that occurs on each feature. In the future, developing EDCMA that uses web and mobile application frameworks will provide the EDCMA system for end-users.

Intisari— Business Process Modeling (BPM) adalah representasi visual berbasis data untuk memodelkan alur kerja bisnis untuk mengoptimalkan proses bisnis. Notasi BPM membuat model mudah dipahami oleh semua pemangku kepentingan yang berperan dalam mengembangkan proses bisnis. Makalah ini akan menjelaskan penerapan BPM dalam alur kerja Electronic Data Capturer (EDC) Monitoring Application (EDCMA). Proses pemodelan alur kerja menggunakan perangkat lunak Bonitasoft yang mencakup pembuatan proses bisnis untuk antarmuka pengguna dan mesin proses. EDCMA merupakan proses untuk menyelesaikan permasalahan pada proses perbaikan terminal EDC dan pengajuan sebagai merchant baru. Berdasarkan prosedur manual EDCMA yang ada, model proses EDCMA yang baru menggunakan bentuk elektronik dan pemrosesan semi-otomatis yang mengimplementasikan beberapa fitur Bonitasoft BPM untuk mengembangkan prototipe aplikasi dengan ketelitian tinggi. Dalam proses evaluasi, kami mengembangkan beberapa skenario dengan pengujian data yang terjadi pada prosedur manual. Berdasarkan evaluasi, prototipe telah berhasil merepresentasikan seluruh proses yang terjadi pada setiap fitur. Di masa depan, pengembangan EDCMA yang menggunakan kerangka kerja aplikasi web dan

seluler akan menyediakan sistem EDCMA untuk pengguna akhir.

Kata Kunci— business process modeling, electronic data capturer monitoring, business workflow

I. PENDAHULUAN

Electronic Data Capturer (EDC) merupakan sebuah alat yang digunakan sebagai media pembayaran dengan menggunakan kartu debit atau kredit. EDC dinilai praktis karena dalam melakukan transaksi tidak perlu menggunakan uang *cash* tetapi dapat langsung menggunakan kartu debit yang sudah terisi saldo atau menggunakan kartu kredit. Oleh karena penggunaannya yang praktis, banyak usaha retail seperti *supermarket*, toko pakaian dan bahkan kegiatan sosial yang menggunakannya sebagai media pembayaran untuk setiap transaksi yang dilakukan.

Setiap harinya ada banyak usaha yang mengajukan penggunaan EDC kepada bank. Untuk mendapatkan EDC, setiap usaha diharuskan untuk menjadi *merchant* bank terlebih dahulu, pemilik usaha diharuskan untuk mengisi *Form Data Nasabah & Merchant* (FDNM). Dalam pengajuan sebagai *merchant* baru tersebut, seringkali terjadi masalah seperti contohnya proses analisis kelayakan usaha yang lebih lama sehingga membuat proses usaha menjadi *merchant* bank menjadi dari target yang diharapkan.

Penggunaan EDC sendiri tidak terlepas dari permasalahan seperti kesalahan *install* pada aplikasi EDC, *human error* (kesalahan manusia), kertas yang habis dan mesin EDC yang korsleting. Untuk memperbaiki masalah yang terjadi pada mesin EDC, *merchant* atau pemilik *merchant* perlu menghubungi bank untuk melakukan pengajuan perbaikan.

Seringkali, pemilik *merchant* akan bertanya mengenai status pengajuan ataupun status perbaikan. Saat pemilik bertanya, operator yang bertugas tidak dapat menjawab pertanyaan mengenai status pengajuan atau perihal teknis yang tidak kunjung datang memperbaiki mesin terminal EDC. Hal tersebut dikarenakan belum adanya notifikasi secara *real time* kepada *merchant*, sehingga proses *tracking* status masih dilakukan secara manual. Untuk itu dibutuhkan

sebuah aplikasi untuk memantau kondisi mesin EDC dan pengajuannya.

Sebelum membangun aplikasi tersebut dibutuhkan sebuah model untuk menjelaskan serta merepresentasikan proses bisnis yang ada di dalam aplikasi tersebut dan digunakanlah sebuah metode yaitu *Business Process Modeling* (BPM) untuk merepresentasikan proses bisnis yang akan dibuat. BPM sendiri merupakan suatu kegiatan atau metode untuk memodelkan proses bisnis dengan menggunakan berbagai notasi.

Artikel ini mendeskripsikan hasil pemodelan proses bisnis untuk *Electronic Data Capturer* (EDC) *Monitoring Application* (EDCMA) dengan memanfaatkan BPM *Notation* (BPMN). Model yang dikembangkan didasarkan pada dua kebutuhan penting dari EDCMA, yaitu menghubungkan pengguna dan bank saat mengajukan perbaikan terminal mesin EDC dan pengajuan sebagai *merchant* baru. Model bisnis yang dikembangkan juga menghasilkan sebuah prototipe *high fidelity* dengan memanfaatkan fasilitas yang tersedia dari Bonita Studio Community Version 7.6.1.

II. TINJAUAN PUSTAKA

Deskripsi yang jelas tentang proses bisnis dapat berfungsi sebagai titik awal untuk desain sistem TI, untuk mengatur pengukuran kinerja, untuk memberlakukan proses oleh BPMS (*Business Process Management System*), melakukan studi simulasi, dll. Berbagai macam tujuan ini memotivasi bagaimana proses pemodelan telah muncul sebagai tradisi yang kuat dalam penelitian tentang BPM [1]. Pemodelan proses bisnis sudah menjadi praktik yang relevan dalam Sistem Informasi. Mengutip dari Indulska, et.al. [2], pemodelan proses bisnis adalah sebuah pendekatan untuk menampilkan secara grafis bagaimana organisasi melakukan proses bisnis mereka. Pemodelan proses bisnis telah muncul sebagai domain penting dan relevan dari pemodelan konseptual [3].

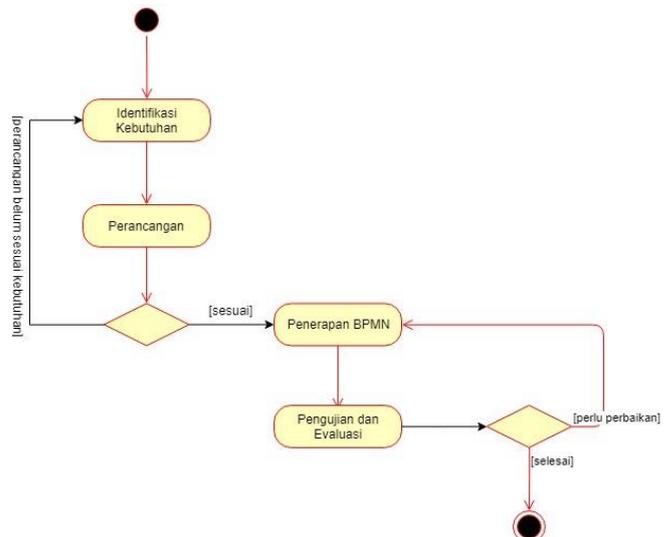
Sebagian besar pakar bisnis bertaruh pada Model dan Notasi Proses Bisnis OMG [4] (BPMN - *Business Process Modeling and Notation*) sebagai standar de-facto (ISO/IEC 19510:2013) dan teknologi terpilih untuk memodelkan proses [5]. Selain BPMN, sebetulnya ada beberapa model lain yang dapat digunakan, antara lain UML (*Unified Modeling Language*) [6] atau *Domain Specific Business Process Modeling* dengan *Domain Specific Languages* (DSL) [7] [8] yang didasarkan pada UML dan menerapkan *Model Driven Architecture* (MDA) [9].

Dalam pemodelan dengan menggunakan BPMN, berdasar studi literatur yang dilakukan oleh Corradini, et.al. [10] terdapat 50 petunjuk pemodelan yang terasosiasi dengan metric dan ambang batas sebagai ukuran kualitas model. Pedoman adalah tujuan umum, dan ini memungkinkan para pemodel bebas memilih yang lebih relevan sesuai dengan tujuan pemodelan – misalnya, proses pembelajaran, pengembangan sistem informasi. Kajian literasi terkait petunjuk pemodelan BPMN juga dilakukan oleh Avila, et.al. [11] yang mengidentifikasi 45 petunjuk dalam 5 kategori. Petunjuk pemodelan ini dapat menjadi pedoman pola dalam pemodelan proses bisnis yang baik dan benar.

Saat ini tersedia beberapa *framework* dan termasuk *integrated development environment* (IDE). Framework tersebut sering disebut sebagai *process-engine*. Beberapa framework yang tersedia di bawah lisensi open-source, antara lain: jBPMN atau Jboss BPM [12], Activiti [13], dan BonitaSoft BPMN [14]. Activiti dan jBPM memiliki kemiripan, salah satunya adalah keduanya fokus sebagai framework *process-engine* yang berorientasi pada pengembang. Sedangkan BonitaSoft berbasis pada solusi tool-based di mana pengguna dapat melakukan *click* dan *drag* dalam pendefinisian proses dan form. Sebuah library yang serupa dengan Activiti atau jBPM, tersedia dalam bahasa Python, yaitu pyBPMN [15], yang kemudian diperluas fungsinya untuk dapat menangani otomatisasi proses dengan pendekatan Model Driven System Engineering Architecture (MDSEA) [16].

III. METODOLOGI PENELITIAN

Dalam melakukan pengembangan model bisnis dengan menerapkan BPMN, metodologi yang diterapkan pada prinsipnya dilakukan dengan tahapan yang berulang pada beberapa bagian (Lihat Gambar 1).



Gambar. 1. Metodologi Penelitian

Pada tahap Identifikasi kebutuhan dan perancangan akan berulang sampai disepakatinya rancangan proses bisnis yang sudah sesuai dengan kebutuhan dari EDCMA. Demikian juga pada tahap pengembangan dengan BPM Notation dan pengujian akan berulang sampai seluruh rancangan telah diterapkan dengan baik. Berikut adalah langkah-langkah penelitian yang dilakukan:

1. Identifikasi Kebutuhan

Melakukan identifikasi terhadap masalah yang terjadi saat ini. Identifikasi yang dilakukan dengan cara melakukan wawancara kepada tim analis bisnis yang mengembangkan proses bisnis untuk aplikasi *monitoring* EDC. Selain itu, proses identifikasi juga didasarkan pada hasil pengamatan terhadap proses bisnis yang terjadi saat ini sehingga dapat diketahui proses dan permasalahan dalam kejadian nyata.

2. Perancangan

Setelah proses identifikasi dilakukan, dirancanglah sebuah proses bisnis dengan menggunakan *flow chart* untuk menggambarkan proses bisnis baru yang menyelesaikan permasalahan saat ini. Selain merancang proses bisnis, dirancang juga tugas-tugas yang dapat aktor lakukan pada aplikasi *monitoring* EDC. Dalam perancangan

3. Penerapan model dengan BPM Notation

Setelah melakukan perancangan, proses bisnis diimplementasi dengan menggunakan metode *Business Process Modeling* yang dikemas dengan berbagai anotasi *Business Process Model & Anotation*. Anotasi tersebut dikonversi menjadi sebuah *blue print* proses bisnis menggunakan aplikasi Bonita Studio, sehingga *blue print* proses bisnis yang dihasilkan dapat dieksekusi (*execute*) untuk melihat jalannya proses bisnis pada setiap aktor.

4. Pengujian dan Analisis Proses Bisnis

Setelah proses bisnis diimplementasi, proses yang sudah menjadi sebuah *blue print* akan diuji untuk mengetahui proses berjalan dengan benar dan hasil keluaran (*output*) yang dihasilkan sesuai dengan perancangan. Pengujian dilakukan dengan mengeksekusi proses bisnis dan memberi masukan (*input*) sesuai dengan formulir atau data yang diserahkan. Setelah itu, dilihat notifikasi yang keluar sesuai dengan perancangan dan penerima.

IV. HASIL DAN ANALISIS

A. Identifikasi Masalah

Penelitian dilakukan dengan membuat proses bisnis pada pengajuan *merchant* baru dan pengajuan perbaikan mesin terminal *Electronic Data Capturer* (EDC). Pada pengajuan *merchant* baru, seseorang yang memiliki usaha dapat melakukan pengajuan untuk menjadi nasabah sekaligus mendaftarkan usahanya menjadi *merchant* bank. Proses ini terjadi ketika pemilik usaha merasa bahwa usahanya membutuhkan dukungan dari bank untuk mendapatkan mesin terminal EDC.

Selain pengajuan *merchant* baru, bagi usaha yang telah terdaftar sebagai *merchant* bank dan telah mendapatkan EDC dapat melakukan pengajuan perbaikan mesin terminal EDC. Pengajuan perbaikan ini dilakukan jika terdapat masalah pada terminal EDC yang mengakibatkan tidak dapat dilakukannya transaksi. Dalam pengajuan *merchant* baru dan perbaikan mesin EDC, terdapat beberapa masalah, adapun masalah-masalah tersebut dijabarkan sebagai berikut:

1. Penilaian atas usaha masih dilakukan secara manual. Penilaian dilakukan dengan pengecekan formulir satu persatu yang akan memakan banyak waktu serta memiliki resiko kesalahan manusia atau *human error* jika penilaian tidak dilakukan secara teliti.
2. Proses persetujuan masih manual, untuk saat ini proses pengecekan dokumen masih dilakukan satu per satu dan mengakses satu persatu sistem yang tentunya akan memakan waktu yang tidak sebentar serta memiliki resiko kesalahan manusia atau *human error*.
3. Dokumentasi Manual, dokumentasi dan perpindahan dokumen yang ada saat ini tidak digital (*paper-based*) sehingga ada resiko dokumen tersebut hilang ataupun rusak.

4. *Service Level Agreement* (SLA) tidak sesuai standar. Dikarenakan sistem yang belum terintegrasi dan beberapa proses dilakukan secara manual, SLA atau waktu respon menjadi semakin lama dari standar yang diharapkan 1 minggu dapat menjadi lebih dari waktu tersebut.
5. Informasi tidak *real time*. Masalah ini juga merupakan runtut dari sistem yang tidak terintegrasi dan proses yang dilakukan secara manual sehingga saat nasabah atau pemilik *merchant* menanyakan mengenai informasi pengajuan *merchant* atau perbaikan terminal EDC, informasi tidak diberikan secara *real time*.

A. Kebutuhan Fungsional

Aplikasi *monitoring* EDC untuk mengatasi beberapa masalah yang sering terjadi di dalam proses pengajuan EDC dan pelaporan mesin EDC rusak. Aplikasi *monitoring* EDC juga dirancang untuk dapat pengisian data-data seperti data pengajuan *merchant* baru dan data perbaikan EDC yang dapat diakses oleh operator. Berdasarkan identifikasi masalah yang telah dilakukan, dirancang kebutuhan fungsional untuk menunjang proses bisnis dalam aplikasi *monitoring*. Adapun kebutuhan fungsional sebagai berikut:

1. Fasilitas *Maintenance* EDC, fasilitas ini akan dihadirkan di dalam sistem sehingga operator hanya perlu melakukan input data perbaikan. *Job order* akan dikirim kepada eksekutor secara otomatis oleh sistem dan nantinya *job order* tersebut akan dipantau oleh *checker*. Di dalam fasilitas ini terdapat beberapa fitur antara lain sebagai berikut:
 - a. Menerbitkan *job order* untuk perbaikan serta pemeliharaan terminal EDC. *Job order* ini akan diterbitkan otomatis oleh sistem
 - b. Melakukan konfirmasi menuju lokasi perbaikan ataupun selesai melakukan perbaikan terminal EDC yang dilakukan oleh Eksekutor.
 - c. Melakukan pemantauan *job order* perbaikan yang telah dilakukan oleh Eksekutor. Pemantauan ini dilakukan oleh *Checker*.
2. Fasilitas Permohonan nasabah sekaligus *merchant* Baru. Dalam perancangan, aplikasi ini memfasilitasi permohonan menjadi nasabah baru sekaligus *merchant* baru yang akan dioperasikan oleh operator untuk mendaftarkan calon nasabah dan *merchant* yang mengajukan penggunaan EDC. Di dalam fasilitas ini terdapat beberapa fitur antara lain sebagai berikut:
 - a. Masukan pendaftaran data calon nasabah dan *merchant* baru melalui operator.
 - b. Melakukan analisis kelayakan calon *merchant* melalui sistem.
 - c. Memberikan persetujuan oleh *approver* sebagai *merchant* melalui sistem setelah analisis yang dilakukan oleh sistem.
 - d. Menerbitkan *job order* untuk pemasangan terminal EDC. *Job order* diterbitkan oleh sistem saat *approver* sudah melakukan persetujuan calon *merchant*.
 - e. Melakukan konfirmasi menuju lokasi pemasangan ataupun selesai melakukan pemasangan yang dilakukan oleh Eksekutor.

- f. Melakukan pemantauan *job order* pemasangan yang telah dilakukan oleh Eksekutor. Pemantauan ini dilakukan oleh *Checker*.

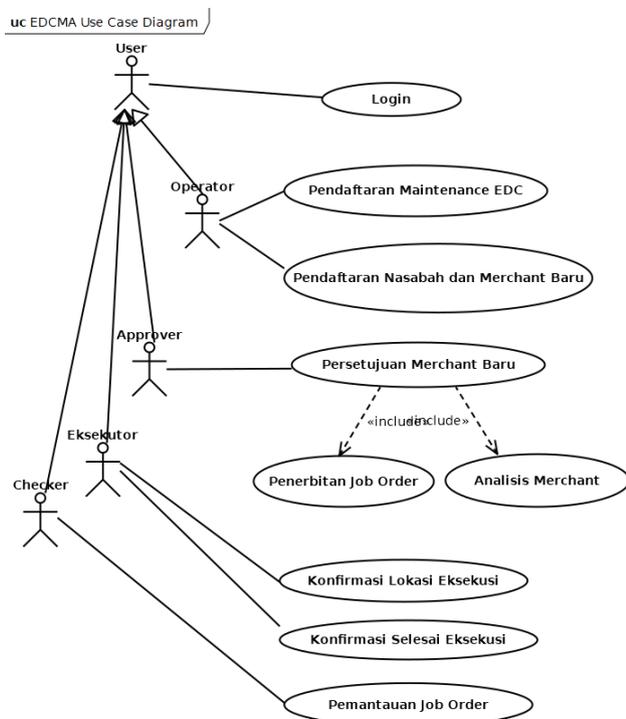
B. Identifikasi User

Dalam aplikasi *monitoring* EDC ini terdapat *user* yang akan mengoperasikan serta menggunakan fasilitas pada aplikasi ini. Adapun *user* (aktor) dalam aplikasi *monitoring* ini sebagai berikut:

1. Operator, yang merupakan *customer service officer* yang bertugas untuk mengisi data pengajuan perbaikan terminal EDC atau pengajuan *merchant* baru sesuai dengan data pada formulir diserahkan oleh pemilik usaha atau *merchant*.
2. Eksekutor atau teknisi, yang bertugas untuk melakukan eksekusi *job order* sesuai dengan permintaan.
3. *Approver*, yang bertugas untuk menyetujui pengajuan *merchant* sesuai dengan hasil analisis.
4. *Checker*, yang bertugas untuk memeriksa *job order* yang telah dieksekusi oleh eksekutor
5. Sistem, yang akan melakukan analisis terhadap data calon *merchant* dan menerbitkan *job order* untuk eksekutor secara otomatis ketika terdapat data baru atau permintaan perbaikan.

C. Use Case

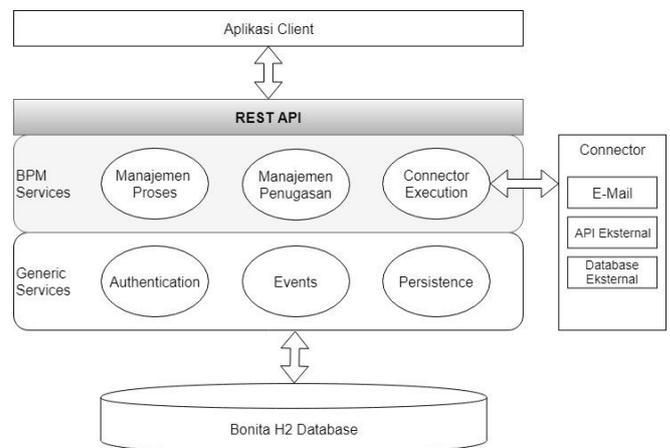
Berdasarkan kebutuhan fungsional yang telah dirancang, dirancang pula *use case* untuk merepresentasikan fasilitas yang dapat dioperasikan atau digunakan oleh setiap aktor berikut ini rancangan *use case* untuk merepresentasikan setiap fungsi pada aplikasi *monitoring*. Gambar 2 memperlihatkan hasil identifikasi kebutuhan untuk EDCMA yang digambarkan dalam bentuk *use case diagram*.



Gambar 2. Use Case Aplikasi Monitoring EDC

D. Implementasi Arsitektur

Arsitektur proses bisnis pada tugas akhir ini mengikuti arsitektur yang digunakan oleh Bonita BPM (Gambar 3). Gambaran arsitektur tersebut didasarkan pada dokumentasi yang disediakan oleh Bonitasoft BPM [14]. Bonita BPM Engine memiliki 2 layer yaitu *layer Generic* dan *Business Services*. Engine sendiri terhubung dengan H2 Database yang menyimpan berbagai *Business Data Model*. *Layer generic* mengatur autentikasi *user* sesuai dengan *role* dan juga *password*, *event* yang digunakan untuk memulai sebuah proses dan *persistence* yang digunakan untuk konversi data menjadi sebuah objek. Selain itu Engine juga memiliki layer *BPM services*, pada layer ini elemen yang ada pada BPM dimanajemen seperti proses, *task* yang tersedia dalam proses dan *connector*. *Connector* sendiri dapat terhubung dengan sistem atau *service external* contohnya *email* dan *external database*.



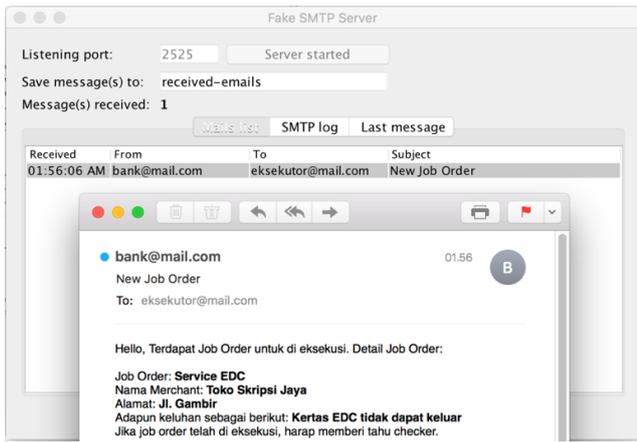
Gambar 3. Arsitektur Proses Bisnis dengan Bonitasoft

Task dan *Business Data Model* yang sudah dibuat pada *layer* sebelumnya dapat diakses oleh pengguna melalui REST API dengan *end point* yang telah ditentukan oleh Bonita. Sehingga, *client* dapat mengetahui hasil keluaran dari *task* atau *Business Data Model* dalam bentuk JSON.

E. Implementasi Proses Bisnis Pengajuan Perbaikan Terminal EDC

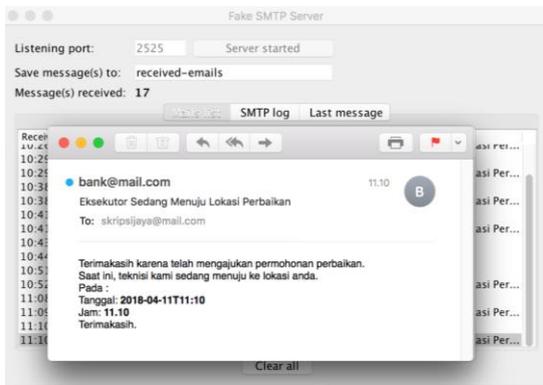
Berdasarkan hasil implementasi yang dilakukan, didapatkan proses bisnis dalam proses pengajuan perbaikan terminal EDC sebagai berikut:

1. Proses pertama, pemilik *merchant* datang kepada operator untuk mengajukan perbaikan terminal EDC. Pada proses ini operator mengisi data pada formulir pengajuan perbaikan pada sistem. Adapun data-data yang akan diisi seperti nama *merchant*, alamat serta kontak dari *merchant*. Pada formulir ini, operator juga akan mengisi keluhan terkait kerusakan dari terminal EDC.
2. Kemudian, setelah data pengajuan perbaikan terminal EDC disimpan, sistem akan langsung mengirimkan *job order* sesuai dengan data kepada eksekutor (teknisi) melalui email. Adapun notifikasi berisi nama dan alamat *merchant* dan *job order* yang diterbitkan serta keluhan kerusakan EDC. Gambar 4 merupakan contoh notifikasi *job order* baru untuk eksekutor.



Gambar 4. Notifikasi Job Order baru

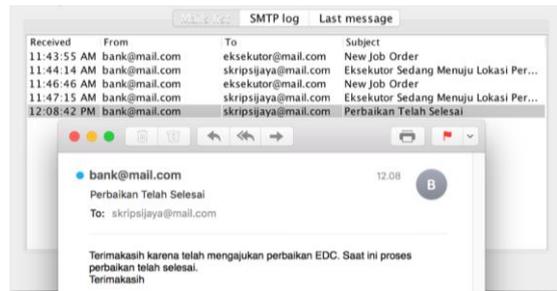
- Setelah mendapatkan email notifikasi *job order*, eksekutor akan melakukan konfirmasi menuju lokasi *on the way*. Untuk melakukan konfirmasi, eksekutor harus mengisi sebuah formulir yang membutuhkan masukan (*input*) yaitu waktu *on the way* atau waktu eksekutor melakukan perjalanan.
- Setelah data konfirmasi menuju lokasi *on the way* disimpan oleh eksekutor, sistem akan mengirim notifikasi melalui email kepada *merchant* bahwa eksekutor menuju lokasi pemasangan. Di dalam email notifikasi, terdapat waktu eksekutor melakukan perjalanan sehingga pemilik *merchant* mengetahui waktu eksekutor menuju lokasi perbaikan. Gambar 5 merupakan contoh email notifikasi untuk *merchant*.



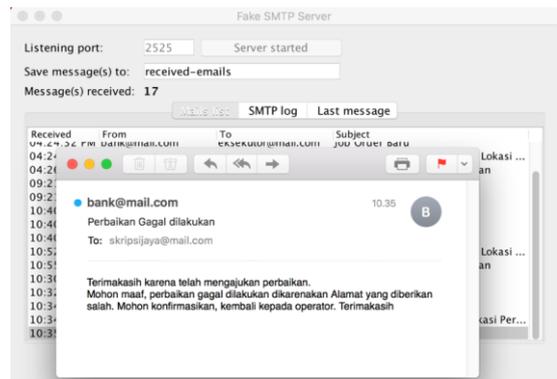
Gambar 5. Notifikasi Eksekutor menuju lokasi perbaikan

- Jika eksekutor berhasil melakukan pemasangan, eksekutor akan melakukan konfirmasi bahwa pemasangan telah selesai. Pada konfirmasi ini, eksekutor akan mengisi formulir yang berisi waktu selesai eksekusi, status eksekusi *job order* dan keterangan eksekusi *job order* (apabila ada).
- Lalu *job order* yang sudah dieksekusi akan diperiksa atau dipantau oleh *checker*. Pada formulir pantau *job order*, disajikan data-data seperti informasi *merchant* yang mengajukan perbaikan lalu status serta waktu dari eksekusi *job order* yang telah dilakukan oleh eksekutor.
- Jika pemasangan berhasil dilakukan, sistem akan mengirimkan notifikasi email bahwa perbaikan selesai. Namun, jika pemasangan gagal dilakukan, sistem akan mengirimkan notifikasi bahwa perbaikan gagal

dilakukan beserta alasan gagalnya perbaikan. Gambar 6 merupakan email notifikasi perbaikan selesai dan Gambar 7 merupakan email notifikasi perbaikan gagal dilakukan.



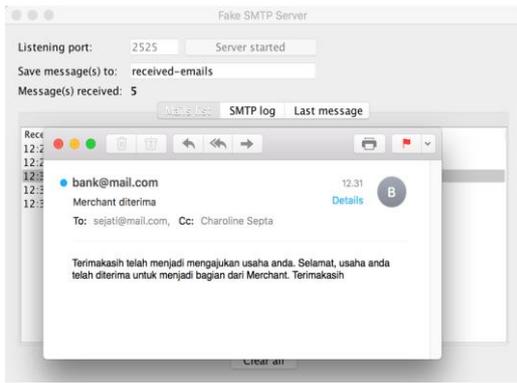
Gambar 6. Notifikasi Perbaikan selesai dilakukan



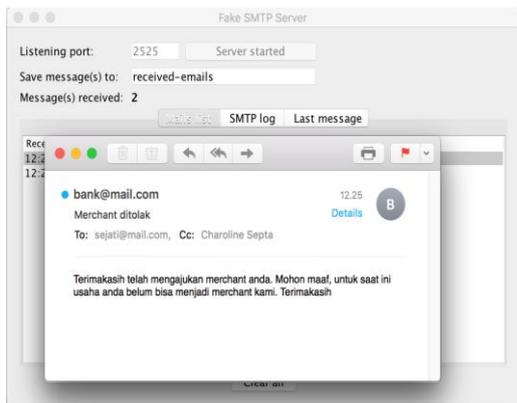
Gambar 7. Notifikasi Perbaikan selesai dilakukan

F. Implementasi Proses Bisnis Pengajuan Merchant Baru

- Untuk mengajukan *merchant* baru, nasabah atau pemilik usaha akan menemui operator atau *customer service officer* untuk menyerahkan data-data terkait pemilik seperti nama, alamat, kontak serta informasi pribadi lainnya. Setelah informasi pemilik, data terkait usaha serta perijinan usaha juga harus diberikan kepada operator untuk diisi. Contoh data usaha antara lain nama usaha, alamat usaha, nomor akta pendirian, nomor akta pengesahan, nomor surat izin usaha, nomor surat izin tempat usaha, dan lain-lain. Pemilik usaha juga harus memberi tahu operator mengenai jumlah rata-rata omzet, jumlah transaksi dan pengujung per bulan. Setelah data lengkap, operator akan mengisi formulir sesuai dengan data dan akan disimpan.
- Setelah data-data disimpan, sistem akan melakukan analisis terhadap data yang telah disimpan. Analisis dilakukan dengan menggunakan data-data terkait dengan omzet dan pengujung usaha. Jika data yang disimpan memenuhi kriteria penilaian maka sistem akan memberikan hasil analisis baik.
- Untuk usaha yang mendapatkan hasil analisis baik, *approver* akan memberikan persetujuan untuk menjadi *merchant*. Bagi usaha yang mendapatkan hasil analisis buruk, sistem akan memberikan notifikasi melalui email bahwa usaha tidak dapat menjadi *merchant* bank. Gambar 8 merupakan contoh notifikasi untuk usaha yang diterima sebagai *merchant* dan Gambar 9 merupakan contoh notifikasi untuk usaha yang ditolak sebagai *merchant*

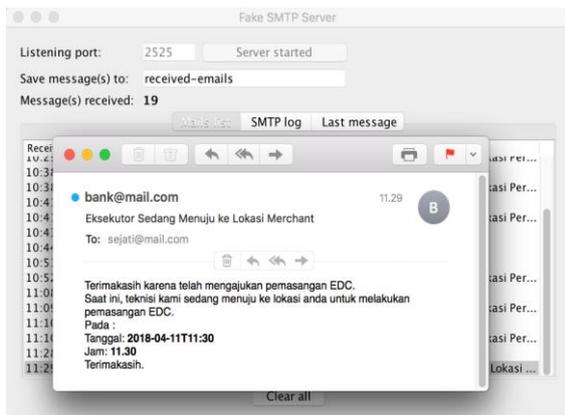


Gambar 8. Notifikasi Merchant diterima



Gambar 9. Notifikasi Merchant ditolak

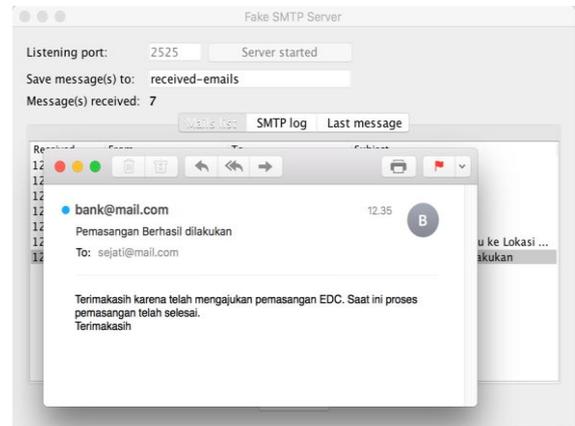
- Setelah usaha disetujui menjadi *merchant*, sistem akan mengirimkan notifikasi *job order* kepada eksekutor melalui email. Notifikasi *job order* ini berisi nama dan alamat *merchant* serta *job order* untuk pemasangan terminal EDC. Gambar 10 merupakan contoh notifikasi *job order* pemasangan EDC untuk eksekutor.



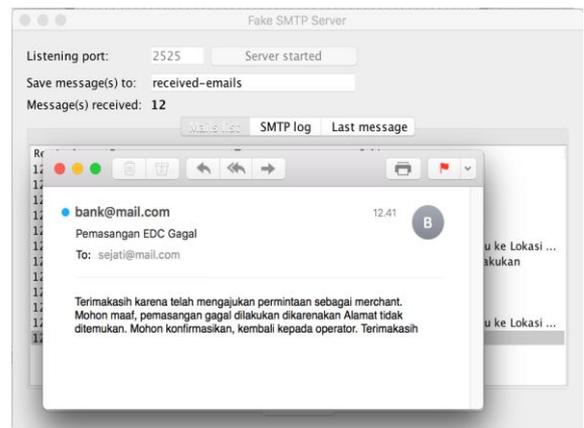
Gambar 10. Notifikasi Job order pemasangan

- Setelah mendapatkan *job order*, eksekutor akan melakukan konfirmasi menuju lokasi *on the way*. Setelah data konfirmasi disimpan, sistem akan mengirim notifikasi kepada *merchant* bahwa eksekutor menuju lokasi pemasangan.
- Jika eksekutor berhasil melakukan pemasangan, eksekutor akan melakukan konfirmasi bahwa pemasangan telah selesai melalui sebuah formulir.

- Lalu *job order* yang sudah dieksekusi akan diperiksa oleh *checker*. Jika pemasangan berhasil dilakukan, sistem akan mengirimkan notifikasi email bahwa pemasangan telah selesai. Namun, jika pemasangan gagal dilakukan, sistem akan mengirimkan notifikasi bahwa pemasangan gagal dilakukan beserta alasan gagalnya pemasangan. Gambar 11 merupakan contoh notifikasi bahwa pemasangan berhasil dan Gambar 12 merupakan contoh notifikasi bahwa pemasangan gagal dilakukan.

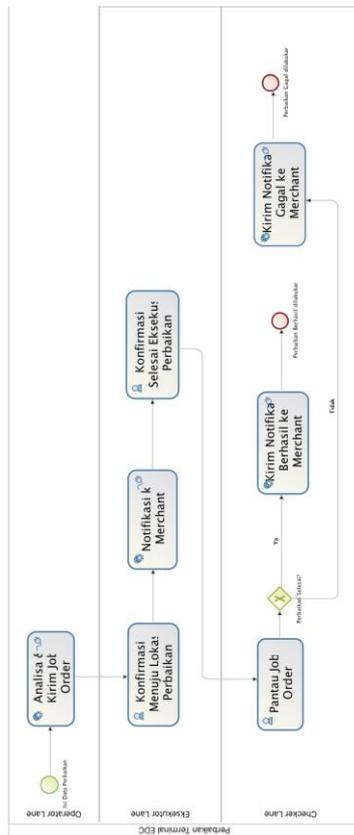


Gambar 11. Notifikasi Pemasangan berhasil

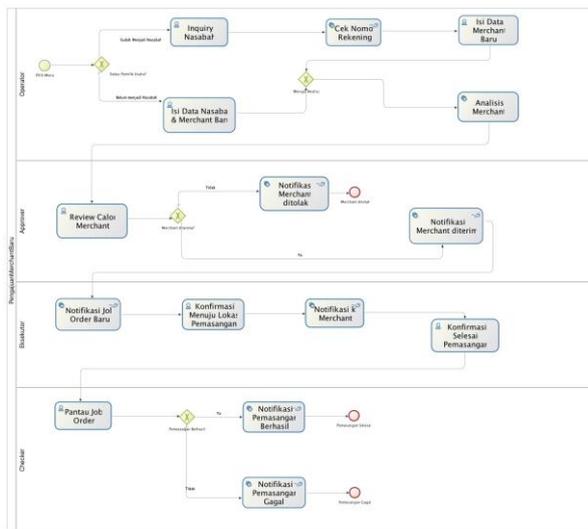


Gambar 12. Notifikasi Pemasangan berhasil

Selain itu, untuk representasi proses bisnis yang lebih jelas telah disajikan diagram proses bisnis untuk pengajuan perbaikan terminal EDC dijabarkan pada Gambar 13 dan Diagram proses bisnis untuk pengajuan sebagai *merchant* baru dijabarkan pada Gambar 14.



Gambar 13. Diagram Perbaikan Terminal EDC



Gambar 14. Diagram Pengajuan Merchant Baru

G. Pengujian Proses Bisnis

Berdasarkan proses bisnis yang telah diimplementasi, dilakukan pengujian. Pengujian dilakukan dengan menggunakan beberapa data. Hasil pengujian akan menyajikan proses bisnis dapat berjalan dengan lancar atau tidak dan umpan balik (*output*) yang dihasilkan sesuai dengan masukan (*input*) yang diberikan. Adapun hasil pengujian sebagai berikut:

- 1) Proses Bisnis Pengajuan Perbaikan Terminal EDC

- a) Data berhasil disimpan ke dalam *Business Data Model*
- b) *Job Order* sesuai keluhan berhasil diterbitkan
- c) Eksekutor berhasil mendapatkan notifikasi *job order* sesuai dengan permintaan oleh pemilik *merchant*
- d) *Merchant* berhasil mendapatkan notifikasi eksekutor menuju lokasi dan status perbaikan.
- e) Semua *task* berhasil dilewati.
- 2) Proses Bisnis Pengajuan *Merchant* Baru
 - a) Data pengajuan *merchant* baru berhasil disimpan
 - b) Analisis sesuai parameter berhasil dilakukan.
 - c) Eksekutor berhasil mendapatkan notifikasi *job* sesuai dengan permintaan oleh pemilik *merchant*.
 - d) *Merchant* berhasil mendapatkan notifikasi eksekutor menuju lokasi dan status pemasangan.
 - e) Semua *task* berhasil dilewati.

V. KESIMPULAN

Penerapan *Business Process Modeling and Notation* (BPMN) pada aplikasi monitoring *Electronic Data Capturer* untuk menghubungkan pengguna dan bank berhasil merepresentasikan sebuah prototipe model proses yang dapat digunakan untuk menghubungkan antara bank dan pengguna saat mengajukan perbaikan terminal mesin *Electronic Data Capturer* dan pengajuan sebagai *merchant*.

Pemanfaatan Bonita Studio untuk penerapan model proses bisnis berhasil dikonversi menjadi sebuah prototipe dari aplikasi *monitoring*. Dengan prototipe yang dihasilkan, proses bisnis dapat lebih dipahami oleh seluruh stakeholder yang terkait dengan proses pengembangan aplikasi. Proses bisnis dapat lebih dipahami karena setiap tugas direpresentasikan dengan anotasi yang jelas dan proses dapat dijalankan (*run as process*). Dengan demikian, alur proses dan kebutuhan data untuk menunjang proses bisnis dapat diketahui.

Salah satu kelemahan dari prototipe yang dikembangkan, Bonita Studio memiliki perbedaan cara dalam pembuatan diagram dari tiap versinya. Selain itu Bonita Studio masih cukup jarang digunakan sehingga dokumentasi yang tersedia juga kurang tersedia. Hal tersebut mengakibatkan penulis tidak dapat melakukan parsing data nasabah existing kedalam tiap text box. Penelitian selanjutnya akan dilakukan penyempurnaan prototipe dengan tetap memanfaatkan Bonita Studio sekaligus sebagai pembelajaran dalam penguasaan pemanfaatan tool.

UCAPAN TERIMA KASIH

Dengan penuh rasa syukur, pada kesempatan ini penulis menyampaikan ucapan terima kasih untuk semua pihak yang telah membantu proses penulisan penelitian ini.

DAFTAR PUSTAKA

- [1] H. A. Reijers, "Business Process Management: The evolution of a discipline," *Computers in Industry*, vol. 126, p. 103404, 2021.
- [2] M. Indulska, J. Recker, M. Rosemann dan P. Green, "Business Process Modeling: Current Issues and Future

- Challenges,” dalam *21st International Conference, CAiSE 2009*, Amsterdam, Netherlands, 2009.
- [3] I. Davies, P. Green, M. Rosemann, M. Indulska dan S. Gallo, “How do practitioners use conceptual modeling in practice?,” *Data & Knowledge Engineering*, vol. 58, no. 3, pp. 358-380, 2006.
- [4] OMG, “Object Management Group Business Process Model and Notation,” OMG Standards Development Organization, [Online]. Available: <https://www.bpmn.org/>.
- [5] C. Arevalo, M. J. Escalona, I. Ramos dan M. Domínguez-Muñoz, “A metamodel to integrate business processes time perspective in BPMN 2.0,” *Information and Software Technology*, vol. 77, pp. 17-33, 2016.
- [6] Object Management Group, “Unified Modeling Language,” OMG, [Online]. Available: <https://www.uml.org/>.
- [7] S. Brahe dan K. Østerbye, “Business process modeling: Defining domain specific modeling languages by use of UML profiles,” dalam *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2006.
- [8] J. Bicevskis, J. Cerina-Berzina, G. Karnitis, L. Lace, I. Medvedis dan S. Nesterovs, “Domain Specific Business Process Modeling in Practice,” dalam *Conference: Proceedings of the 9th International Baltic Conference on Databases and Information Systems*, 2010.
- [9] OMG Standards Development Organization, “MDA® - THE ARCHITECTURE OF CHOICE FOR A CHANGING WORLD,” OMG Standards Development Organization, [Online]. Available: <https://www.omg.org/mda/>.
- [10] F. Corradini, A. Ferrari, F. Fornari, S. Gnesi, A. Polini, B. Re dan G. O. Spagnolo, “A Guidelines framework for understandable BPMN models,” *Data and Knowledge Engineering*, vol. 113, pp. 129-154, 2018.
- [11] D. T. Avila, R. I. dos Santos, J. Mendling dan L. H. Thom, “A systematic literature review of process modeling guidelines and their empirical support,” *Business Process Management Journal*, vol. 27, no. 1, pp. 1-23, 2021.
- [12] “jBPM - Open Source Business Automation Toolkit,” RedHat, [Online]. Available: <https://www.jbpm.org/>.
- [13] “Activiti - Open Source Business Automation,” Alfresco, [Online]. Available: <https://www.activiti.org/>.
- [14] Bonitasoft, “Bonita Engine Architecture,” Bonitasoft, [Online]. Available: <https://documentation.bonitasoft.com/bonita/7.11/engine-architecture-overview>. [Diakses 14 11 2021].
- [15] P. Bocciarelli, A. D'Ambrogio, A. Giglio dan E. Paglia, “A BPMN Extension to Enable the Explicit Modeling of Task Resources,” dalam *Conference: INCOSE Italia Conference on Systems Engineering, CIISE 2016*, Turin, Italy, 2016.
- [16] K. Ougaabal, Y. Ducq, G. Zacharewicz dan S. Tazi, “Distinguishing Resource Type in BPMN Workflows at Simulation Phase,” dalam *SpringSim 2019 - Spring Simulation Conference*, Tucson, United States, 2019