

ANALISA PROSES BISNIS SISTEM PENGGAJIAN DAN PINJAMAN PEGAWAI STUDI KASUS PERUSAHAAN INDUSTRI KERTAS PT UNIPA DAYA

Ririn Ikana Desanti, Suryasari, Grecia Puspita Gunawan

Jurusan Sistem Informasi Universitas Pelita Harapan

Jl. MH. Thamrin Boulevard No. 2, Lippo Karawaci – Tangerang 15811 Telp (021) 5460901 Ext. 1355

e-mail : ririn.desanti@staff.uph.edu, suryasari@staff.uph.edu, gesa_pusguna@yahoo.com

Abstrak

Sistem penggajian dan pinjaman pegawai yang digunakan pada perusahaan ini masih menggunakan catatan kertas. Oleh karena itu masih banyak terjadi kesalahan dalam penulisan data dan penghitungan uang. Data absensi pegawai juga masih dicatat dalam bentuk kertas sehingga cukup menyulitkan petugas yang melakukan rekapitulasi gaji di setiap akhir minggu.

Penelitian ini bertujuan untuk menyelesaikan permasalahan tersebut dengan cara melakukan analisa terhadap proses bisnis dari sistem yang lama dan memberikan usulan rancangan sistem yang baru. Perancangan sistem baru ini termasuk merancang proses bisnis penambahan pegawai, proses bisnis penggajian pegawai, proses bisnis absensi pegawai, proses bisnis pinjaman pegawai, dan proses bisnis permintaan ijin atau cuti pegawai. Metode yang digunakan untuk mengumpulkan data adalah dengan cara observasi, wawancara, dan menganalisa dokumen yang berkaitan (document analysis). Sedangkan bahasa pemodelan yang digunakan untuk merancang sistem baru adalah UML (unified modeling language).

Hasil analisa dari sistem yang lama adalah sebuah daftar kebutuhan sistem atau pengguna (requirements list) yang digunakan untuk memodelkan rancangan sistem baru. Dan dengan sistem yang baru, perusahaan akan mampu mengatur proses penggajian dan pinjaman pegawai dengan lebih baik karena kesalahan-kesalahan yang terjadi pada sistem lama dapat diminimalisasi.

Kata kunci : *analisa proses bisnis, uml (unified modeling language), sistem penggajian*

1. PENDAHULUAN

Salah satu dokumen penting yang dibutuhkan oleh perusahaan adalah dokumen kepegawaian. Yang dapat termasuk dalam dokumen tersebut adalah data penambahan pegawai, data absensi pegawai, data penggajian pegawai, data pengajuan cuti pegawai, dan data pinjaman pegawai. PT. Unipa Daya adalah perusahaan yang bergerak di bidang industri kertas dan tentu saja pada perusahaan seperti ini, keseluruhan data tersebut sangat dibutuhkan karena mempengaruhi dalam proses penghitungan gaji pegawai.

Saat ini proses pengelolaan data kepegawaian di PT. Unipa Daya masih menggunakan catatan kertas (*paper-based*). Dengan jumlah pegawai sekitar 300 orang, maka ada beberapa kelemahan yang timbul akibat proses pengelolaan data secara manual tersebut seperti misalnya terjadi kesalahan dalam pencatatan data pegawai, kesalahan pencatatan absensi pegawai yang bisa menyebabkan kesalahan penghitungan gaji pegawai, dan masih banyak lagi kesalahan lainnya.

Pembatasan masalah pada penelitian ini hanya mencakup analisa proses bisnis untuk penambahan pegawai, proses bisnis penggajian pegawai, proses bisnis absensi pegawai, proses bisnis pengajuan cuti pegawai, dan proses bisnis pengajuan pinjaman pegawai.

Tujuan dari penelitian adalah untuk menghasilkan sebuah rancangan proses bisnis yang baru untuk sistem penggajian dan pinjaman pegawai di PT. Unipa Daya yang mampu meminimalkan kelemahan yang ada pada sistem lama.

2. TINJAUAN PUSTAKA

Object-Oriented System Analysis and Design (OOAD)

Menurut Grady Booch, Ivar Jacobson, dan James Rumbaugh, pendekatan berorientasi objek untuk membangun suatu sistem informasi harus *use-case driven, architecture centric, iterative and incremental* (Dennis, 2010).

Use-case driven mempunyai arti bahwa *use case* menjadi alat pemodelan yang utama untuk mengetahui perilaku sistem. *Use case* bersifat sederhana karena hanya berfokus pada suatu aktivitas pada suatu waktu.

Architecture centric mempunyai arti bahwa arsitektur *software* yang mendasari spesifikasi sistem yang terus berkembang mengendalikan spesifikasi, konstruksi, dan dokumentasi dari sistem. Analisis dan perancangan sistem berorientasi objek minimum harus mendukung tiga *architectural views* dari sebuah sistem, yaitu *functional, static, dan dynamic*. *Functional (external) view* menggambarkan perilaku eksternal sistem dari perspektif *user*. *Static (structural) view* menggambarkan struktur dari sistem berupa *attributes, methods, classes, dan relationships*. *Dynamic (behavioral) view* menggambarkan perilaku internal dari sistem berupa pesan-pesan (*messages*) yang melewati objek dan perubahan *state* dari objek-objek tersebut.

Iterative and incremental. Analisis dan perancangan sistem berorientasi objek menekankan pada pengembangan yang *iterative* dan *incremental* sehingga pengujian dan perbaikan dilakukan terus-menerus untuk mencapai sistem yang sesuai dengan kebutuhan *user*.

Unified Modeling Language 2.0

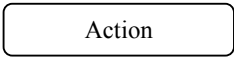

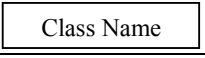
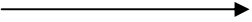
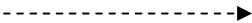



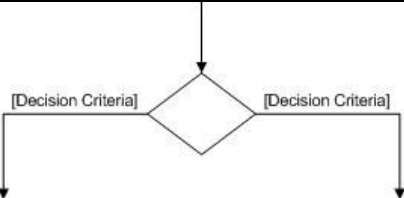
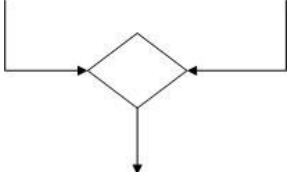
Pada November 1997, *Object Management Group* (OMG, secara formal telah menerima UML sebagai suatu standar untuk semua *object developers*. UML versi 2.0 diterima pada tahun 2003 dan mendefinisikan 14 teknik-teknik pembuatan diagram yang digunakan untuk menggambarkan sistem (Dennis, 2010). Diagram-diagram tersebut dikelompokkan menjadi dua kelompok besar, yaitu untuk permodelan struktur dan untuk permodelan perilaku. Diagram-diagram yang menggambarkan struktur adalah *class*, *object*, *package*, *deployment*, *component*, dan *composite structure diagram*. Diagram-diagram yang menggambarkan perilaku adalah *activity*, *sequence*, *communication*, *interaction overview timing*, *behavior state machine*, *protocol state machine*, dan *use case diagram*.

Dalam menganalisis proses bisnis yang terjadi di PT. Unipa Daya, permodelan UML 2.0 yang digunakan adalah *activity diagram* dan *use case diagram*.

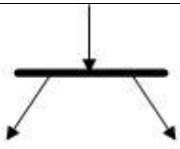
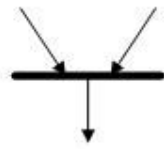
Activity Diagram

Activity diagram digunakan untuk memodelkan perilaku di dalam suatu bisnis. *Activity diagram* dapat dilihat sebagai sebuah *sophisticated data flow diagram* (DFD) yang digunakan pada analisis structural. Akan tetapi, berbeda dengan DFD, *activity diagram* mempunyai notasi untuk memodelkan aktivitas yang berlangsung secara paralel, bersamaan, dan juga proses pengambilan keputusan yang kompleks.

Tabel 1. Elemen-elemen dalam *Activity Diagram*

No.	Nama elemen	Fungsi	Notasi
1.	<i>Action</i>	Untuk menggambarkan perilaku yang sederhana dan bersifat <i>non-decomposable</i> .	
2.	<i>Activity</i>	Untuk mewakili kumpulan aksi (<i>action</i>)	
3.	<i>Object Node</i>	Untuk mewakili objek yang terhubung dengan kumpulan <i>object flow</i> .	
4.	<i>Control Flow</i>	Menunjukkan rangkaian pelaksanaan	
5.	<i>Object Flow</i>	Menunjukkan aliran sebuah objek dari sebuah aktivitas atau aksi ke aktivitas atau aksi lainnya.	
6.	<i>Initial Node</i>	Menandakan awal dari kumpulan aksi atau aktivitas.	
7.	<i>Final-Activity Node</i>	Untuk menghentikan seluruh <i>control flows</i> atau <i>object flows</i> pada sebuah aktivitas (atau aksi).	
8.	<i>Final-Flow Node</i>	Untuk menghentikan <i>control flow</i> atau <i>object flow</i> tertentu.	
9.	<i>Decision Node</i>	Untuk mewakili suatu kondisi pengujian yang bertujuan untuk memastikan bahwa <i>control flow</i> atau <i>object flow</i> hanya menuju ke satu arah.	
10.	<i>Merge Node</i>	Untuk menyatukan kembali <i>decision path</i> yang dibuat dengan menggunakan <i>decision node</i> .	

Tabel 1. Elemen-elemen dalam *Activity Diagram* (lanjutan)

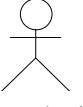
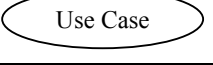
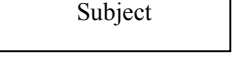

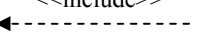
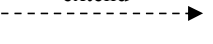

No.	Nama elemen	Fungsi	Notasi				
11.	<i>Fork Node</i>	Untuk memisahkan perilaku menjadi kumpulan aktivitas yang berjalan secara paralel atau bersamaan.					
12.	<i>Join Node</i>	Untuk menyatukan kembali kumpulan aktivitas yang berjalan secara paralel atau bersamaan.					
13.	<i>Swimlane</i>	Untuk membagi sebuah <i>activity diagram</i> menjadi kolom guna menempatkan aktivitas atau aksi tertentu pada individu atau objek yang bertanggung jawab untuk melaksanakan aktivitas atau aksi tersebut.	<table border="1" style="width: 100%;"> <thead> <tr> <th style="width: 50%;">Swimlane 1</th> <th style="width: 50%;">Swimlane 2</th> </tr> </thead> <tbody> <tr> <td style="height: 40px;"></td> <td style="height: 40px;"></td> </tr> </tbody> </table>	Swimlane 1	Swimlane 2		
Swimlane 1	Swimlane 2						

Sumber : Dennis, Alan, Wixom, Barbara H., Tegarden, David, 2010, *System Analysis and Design with UML An Object-Oriented Approach*, 3rd ed, hal.160-164

Use Case Diagram

Sebuah *use case diagram* menggambarkan secara sederhana fungsi-fungsi utama dari sistem dan berbagai *user* yang akan berinteraksi dengan sistem tersebut.

Tabel 2. Elemen-elemen dalam *Use Case Diagram*

No.	Nama Elemen	Fungsi	Notasi
1.	<i>Actor</i>	Menggambarkan tokoh atau sistem yang memperoleh keuntungan dan berada di luar dari sistem. <i>Actor</i> dapat berasosiasi dengan <i>actor</i> lainnya dengan menggunakan <i>specialization/superclass association</i> . <i>Actor</i> ditempatkan di luar <i>subject boundary</i> .	 Actor/Role
2.	<i>Use Case</i>	Mewakili sebuah bagian dari fungsionalitas sistem dan ditempatkan dalam <i>system boundary</i> .	 Use Case
3.	<i>Subject Boundary</i>	Menyatakan lingkup dari subjek.	 Subject
4.	<i>Association Relationship</i>	Menghubungkan <i>actor</i> untuk berinteraksi dengan <i>use case</i> .	 * *
5.	<i>Include Relationship</i>	Menunjukkan <i>inclusion</i> fungsionalitas dari sebuah <i>use case</i> dengan <i>use case</i> lainnya. Arah panah dari <i>base use case</i> ke <i>included use case</i> .	 <<include>>
6.	<i>Extend Relationship</i>	Menunjukkan <i>extension</i> dari sebuah <i>use case</i> untuk menambahkan <i>optional behavior</i> . Arah panah dari <i>extension use case</i> ke <i>base use case</i> .	 <<extend>>
7.	<i>Generalization relationship</i>	Menunjukkan generalisasi dari <i>use case</i> khusus ke yang lebih umum	

Sumber : Dennis, Alan, Wixom, Barbara H., Tegarden, David, 2010, *System Analysis and Design with UML An Object Oriented Approach*, 3rd ed, hal.174

3. METODE PENELITIAN

Metode yang digunakan untuk mengumpulkan data adalah dengan cara observasi, wawancara, dan menganalisa dokumen yang berkaitan (*document analysis*).

Unified Process

Unified Process adalah metodologi khusus yang menggambarkan kapan dan bagaimana menggunakan berbagai macam teknik UML untuk melakukan *object-oriented analysis and design* (Dennis, 2010). Seperti halnya UML, dalam *unified process* pun suatu sistem informasi harus *use-case driven, architecture centric, iterative and incremental*.

Unified process merupakan proses pengembangan sistem dua dimensional yang digambarkan oleh fase-fase (*phases*) dan *workflows*. Fase-fasenya adalah *inception, elaboration, construction, dan transition*. Di dalam *workflows* terdapat *business modeling, requirements, analysis, design, implementation, test, deployment, project management, configuration and change management, dan environment*.

Berdasarkan analisa terhadap proses bisnis saat ini, diketahui adanya kendala dalam melakukan proses pengelolaan penggajian dan pinjaman pegawai sebagai berikut:

- a. Adanya kesalahan dalam penghitungan gaji.
Karena proses penggajian masih secara manual maka sering terjadi kesalahan dalam perhitungan gaji. Hal ini dapat memperlambat proses penggajian dan merugikan perusahaan.
- b. Penggunaan buku yang memboroskan biaya.
Karena semua data dicatat di dalam buku, maka dibutuhkan banyak buku. Hal ini dapat menyebabkan pemborosan biaya bagi perusahaan karena harus membeli buku dan peralatan lainnya.
- c. Pengecekan dan pencarian data mengalami kesulitan.
Semua pencatatan data absensi, gaji, cuti, surat ijin masih dicatat di dalam buku dan belum berbasis komputer. Hal ini mengakibatkan kesulitan dalam pencarian data dan pengecekan data karena terlalu banyak data yang disimpan.
- d. Keterlambatan dalam memutakhirkan data.
Karena proses masih dilakukan secara manual, sehingga proses untuk meng-*update* data mengalami kesulitan. Sebagai contoh, pada proses peminjaman dana sering mengalami kesalahan dalam pencatatan status pelunasan dana.
- e. Data dan surat mengalami kerusakan atau hilang
Semua data dicatat di dalam buku dan surat disimpan di dalam map. Jika hal ini terus dilakukan, maka semua data dan surat semakin lama akan mengalami kerusakan atau bahkan hilang.

4. HASIL DAN PEMBAHASAN

Dari analisa yang telah dilakukan terhadap proses bisnis yang lama, maka telah berhasil dikumpulkan beberapa *requirement* penting yang dibutuhkan untuk merancang proses bisnis sistem yang baru. Adapun *requirement* tersebut adalah:

1. Data pegawai dapat ditambah dan dimutakhirkan.
2. Sistem menyediakan perubahan status pegawai aktif atau tidak aktif.
3. Petugas absensi dapat memasukkan dan memutakhirkan data absensi pegawai
4. Terdapat beberapa status absensi yg dapat disimpan yaitu masuk, tidak masuk tanpa keterangan, cuti, dan ijin.
5. Pegawai dapat mengisi form pengajuan cuti.
6. Kepala Unit memiliki akses untuk menyetujui atau menolak pengajuan cuti pegawai.
7. Petugas absensi dapat mencetak form pengajuan cuti per pegawai yang telah disetujui.
8. Pegawai dapat melihat sisa jatah cuti.
9. Sistem dapat menampilkan status pengajuan cuti pegawai jika disetujui ataupun ditolak.
10. Secara otomatis, sistem akan memotong jatah cuti pegawai jika pengajuan cuti telah disetujui.
11. Petugas absensi dapat menambah dan memutakhirkan data penggajian pegawai.
12. Sistem mampu menghitung rekapitulasi gaji pegawai per minggu secara otomatis.
13. Sistem dapat melakukan pemotongan gaji jika data absensi pegawai masih mengalami masalah.
14. Petugas absensi dapat mencetak laporan penggajian untuk masing-masing pegawai.
15. Pegawai memiliki akses untuk mengisi formulir pengajuan pinjaman dana.
16. Petugas koperasi memiliki akses untuk menyetujui atau menolak pengajuan pinjaman dana.
17. Petugas koperasi dapat mencetak formulir pengajuan pinjaman dana yang telah disetujui.
18. Pegawai dapat melihat status pinjaman dana yaitu lunas atau belum lunas.
19. Pegawai dapat melihat status pengajuan pinjaman dana yaitu disetujui atau ditolak.
20. Sistem mampu mengubah status pinjaman dana secara otomatis menjadi lunas jika pegawai sudah melunasi seluruh pinjaman.

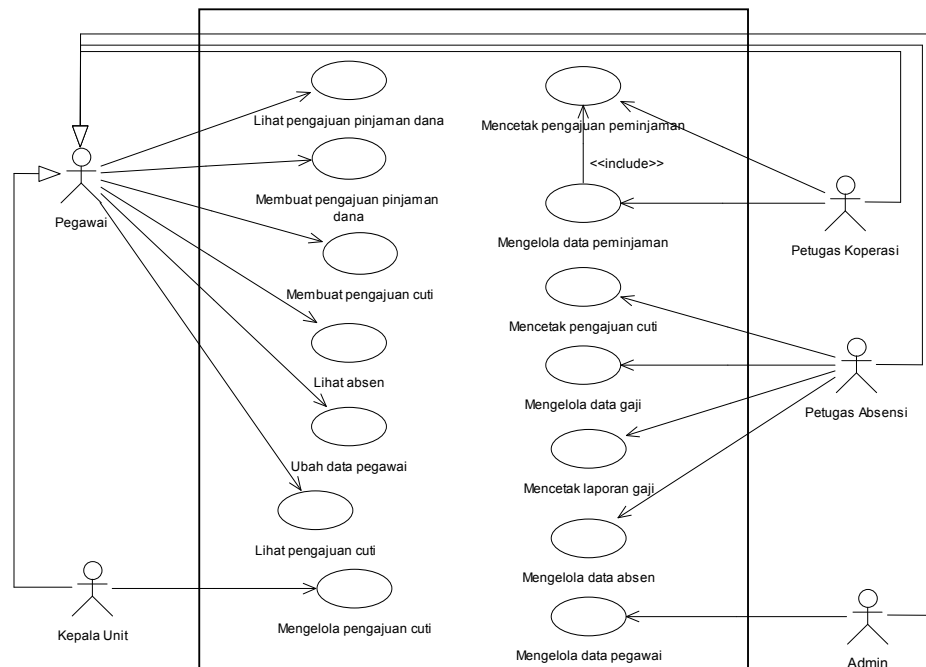
21. Sistem mampu mengubah status pinjaman dana secara otomatis menjadi belum lunas jika pengajuan pinjaman dana pegawai telah disetujui.

Rancangan proses bisnis untuk sistem yang baru akan dimodelkan dengan beberapa diagram UML seperti *activity diagram* dan *use case diagram*. Berikut ini adalah daftar aktor dari rancangan sistem baru yang dibedakan berdasarkan peran masing-masing:

Tabel 3. Daftar aktor pada rancangan sistem baru

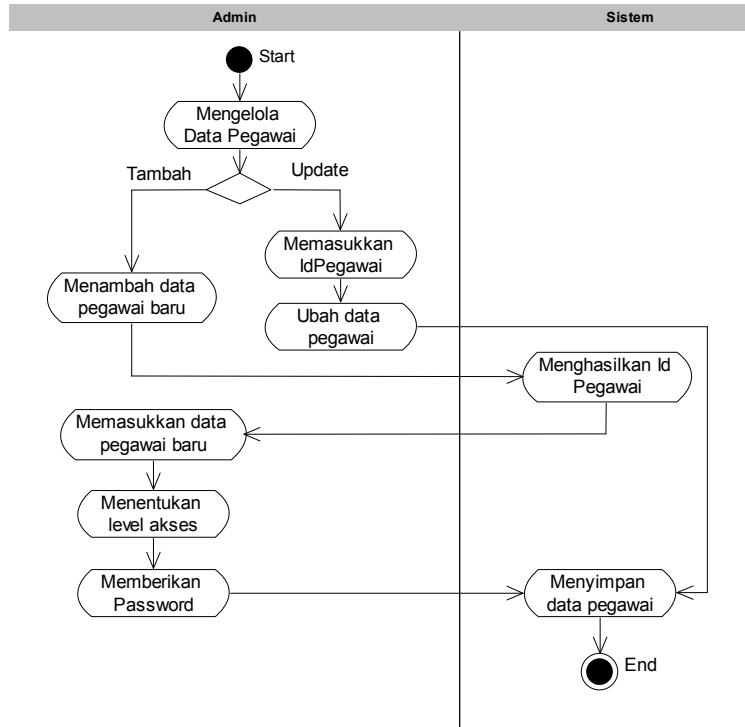
Nama Aktor	Keterangan
Admin	Berperan dalam mengelola data karyawan.
Pegawai	Berperan untuk mengajukan cuti dan pinjaman dana, memutakhirkan data pribadi pegawai, melihat status cuti dan pinjaman dana, melihat data absensi.
Petugas Absensi	Berperan untuk memasukkan dan mengelola data absensi pegawai setiap hari dan mengolah data gaji per minggu, mencetak laporan absensi, mencetak pengajuan cuti pegawai, dan mencetak laporan gaji
Kepala Unit	Berperan untuk mengelola data cuti pegawai, menyetujui atau menolak pengajuan cuti.
Petugas Koperasi	Berperan dalam mengelola data pinjaman dan pelunasan dana, memberi keputusan untuk menyetujui atau menolak pengajuan pinjaman, mencetak laporan pinjaman dana yang disetujui.

Berikut ini *use case diagram* dari rancangan sistem yang baru:



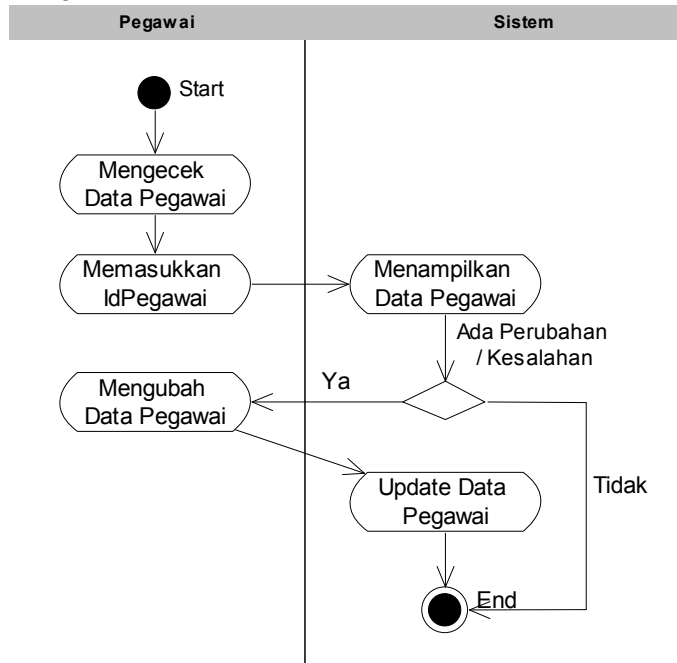
Gambar 1. Use case diagram rancangan sistem yang baru

Kemudian untuk menggambarkan proses bisnis dari rancangan sistem yang baru, akan dijelaskan melalui beberapa *activity diagram*. Yang pertama adalah *activity diagram* proses mengelola data pegawai, dimana aktor yang berperan adalah Admin.



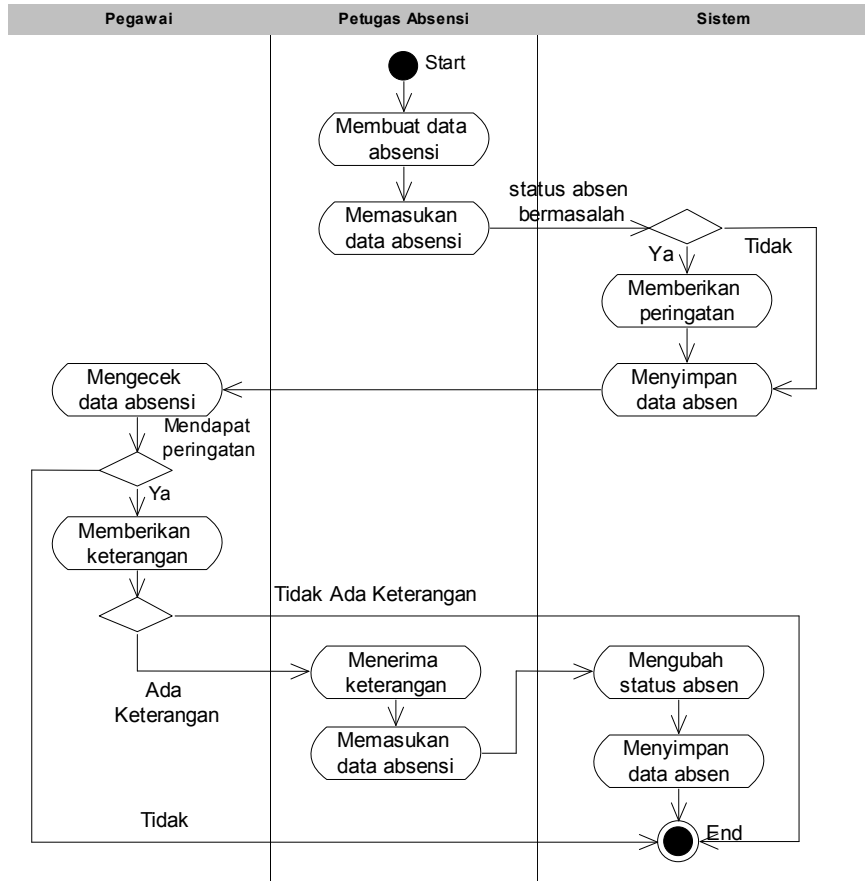
Gambar 2. Activity diagram proses mengelola data pegawai

Yang kedua adalah *activity diagram* ubah data pegawai dimana yang memiliki akses untuk melakukan proses ini adalah pegawai yang bersangkutan.



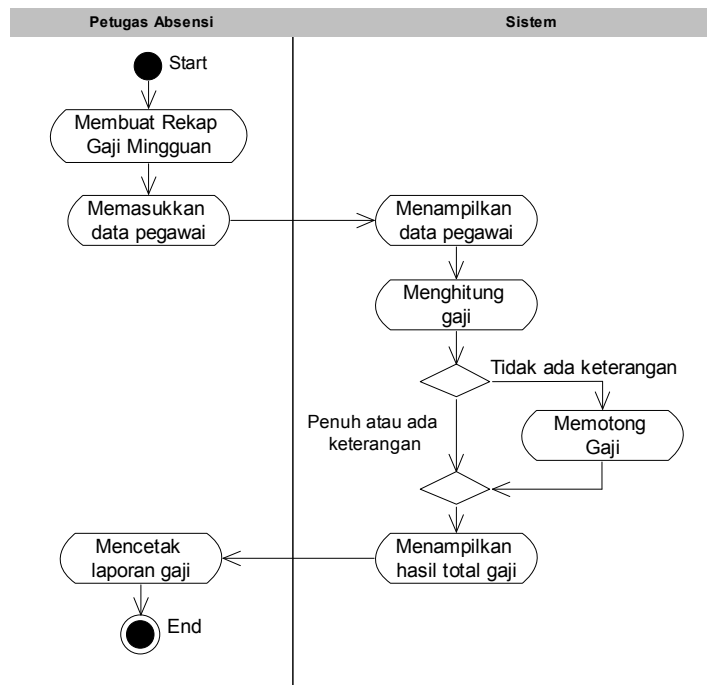
Gambar 3. Activity diagram ubah data pegawai

Ketiga adalah *activity diagram* mengelola data absensi dimana yang berperan dalam proses ini adalah actor pegawai dan petugas absensi.



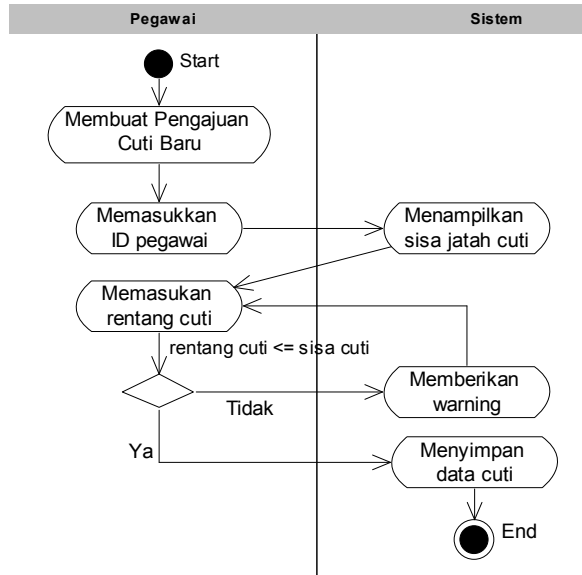
Gambar 4. Activity diagram mengelola data absensi

Keempat adalah *activity diagram* mengelola data gaji, dimana yang berperan untuk melakukan proses ini adalah petugas absensi.



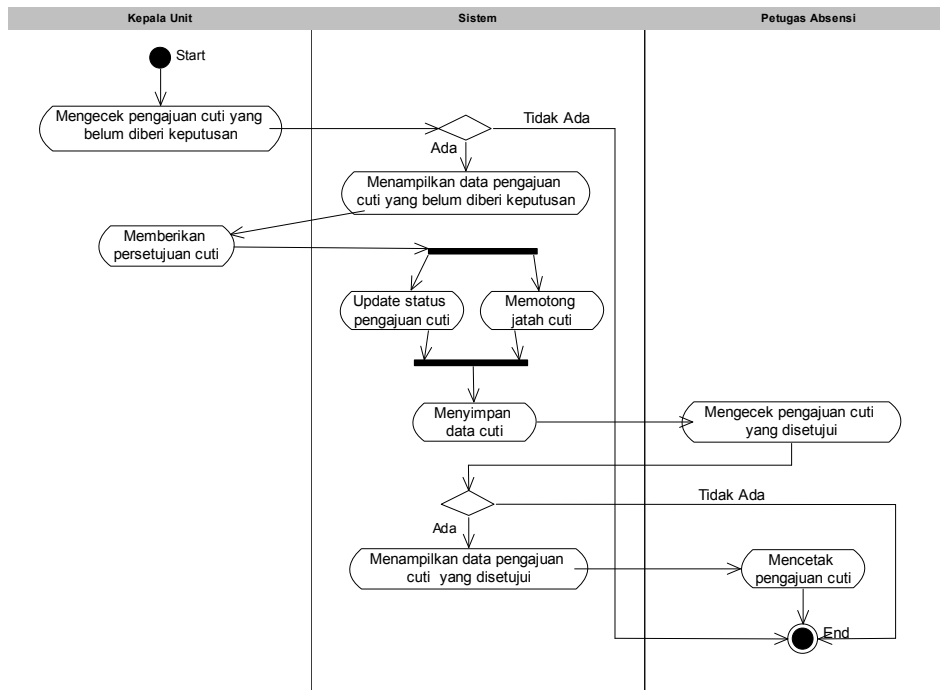
Gambar 5. Activity diagram mengelola data gaji

Kelima adalah *activity diagram* pengajuan cuti pegawai, dimana yang memiliki akses untuk melakukan proses ini adalah aktor pegawai.



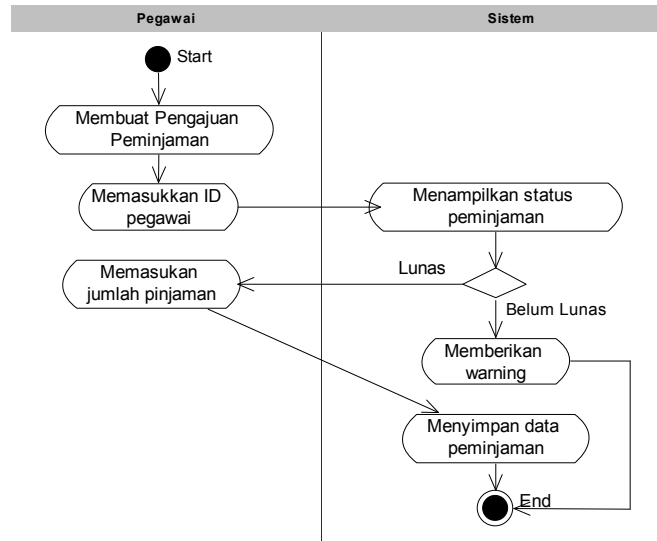
Gambar 6. Activity diagram pengajuan cuti pegawai

Keenam adalah *activity diagram* mengelola dan mencetak pengajuan cuti, dimana aktor yang berperan adalah kepala unit dan petugas absensi.



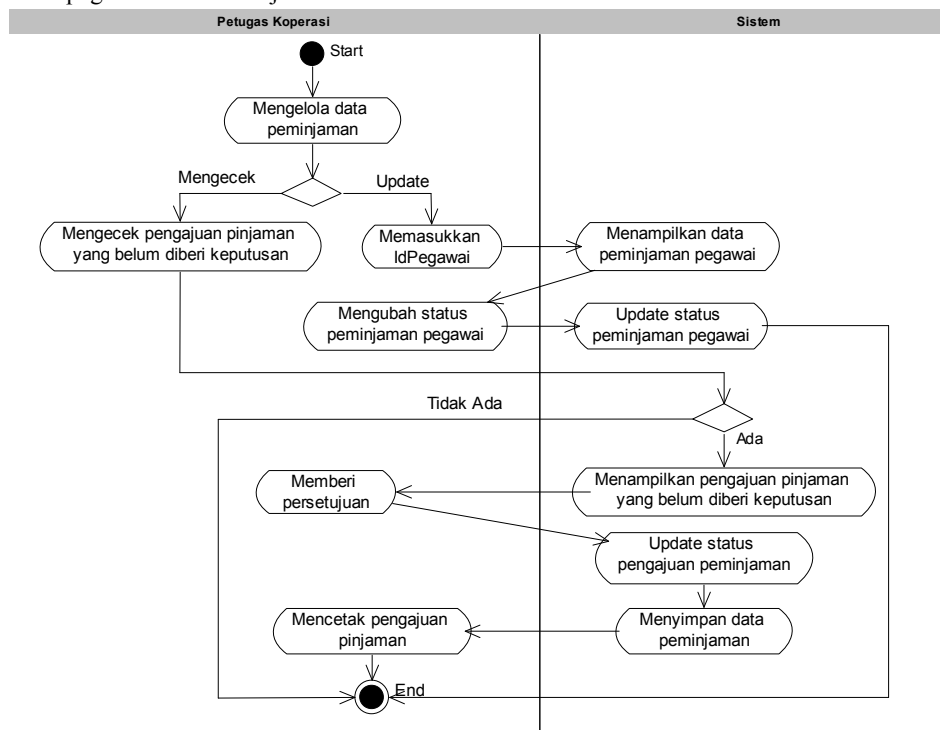
Gambar 7. Activity diagram mengelola dan mencetak pengajuan cuti

Ketujuh adalah *activity diagram* pengajuan pinjaman dana, dimana yang memiliki akses untuk dapat melakukan proses ini adalah pegawai yang bersangkutan.



Gambar 8. Activity diagram pengajuan pinjaman dana

Kedelapan adalah *activity diagram* mengelola dan mencetak pengajuan pinjaman dana, dimana yang berperan untuk mengerjakan proses ini adalah petugas koperasi. Proses ini dapat dilakukan pada saat status pengajuan pinjaman dana pegawai telah disetujui.



Gambar 9. Activity diagram mengelola dan mencetak pengajuan pinjaman dana

5. KESIMPULAN DAN SARAN

Kendala yang dijabarkan pada bagian metode penelitian telah dapat diminimalkan dengan cara memperbaiki proses bisnis dari setiap kegiatan utama dalam proses pengelolaan penggajian pegawai dan pinjaman pegawai, seperti yang terlihat pada *activity diagram* di bagian hasil dan pembahasan.

Rancangan proses bisnis dari sistem penggajian dan pinjaman pegawai ini dapat dikembangkan dan diimplementasi dalam bentuk aplikasi perangkat lunak sehingga dapat memberikan manfaat bagi perusahaan.

6. DAFTAR PUSTAKA

- Dennis, Alan, Wixom, Barbara H., Tegarden, David, 2010, *System Analysis and Design with UML An Object-Oriented Approach*, 3rd ed, Asia: John Wiley & Sons, Inc., pp. 17-237
- Pender, Tom, 2003, *UML Bible*, Canada: Wiley Publishing, Inc.