

**IMPLEMENTASI PENDEKATAN BERORIENTASI OBJEK TERPADU
UNTUK PENGEMBANGAN SISTEM INFORMASI YANG TERINTEGRASI
PADA PT. SAPTA KARYA**

Oleh:

Yovita A. Maria Aziz^{†1}, Mohammad Fajar², Afifah³

^{1,3}Sistem Informasi, ²Informatika
STMIK KHARISMA Makassar

Abstrak: Penelitian ini bertujuan untuk menganalisis dan merancang sistem informasi penjualan menggunakan pendekatan berorientasi objek terpadu (unified) pada PT. Sapta Karya. Pengumpulan data untuk mendefinisikan spesifikasi kebutuhan sistem dilakukan melalui studi literatur, observasi, dan wawancara. Pada fase analisis, digunakan diagram aktivitas untuk memodelkan proses bisnis sistem, diagram usecase untuk mendefinisikan spesifikasi fungsional sistem, sequence diagram untuk memodelkan interaksi antara bagian penjualan, bagian gudang, dan keuangan, serta diagram class untuk spesifikasi struktur statik sistem yang terdiri dari entitas Pelanggan, Pegawai, Barang, dan Transaksi. Pada fase perancangan, tipe atribut dan metode pada diagram class didefinisikan, dan diagram aktivitas untuk pemodelan logika aplikasi. Pendekatan unified dalam pengembangan sistem informasi yang terintegrasi dapat diterapkan, dimana setiap bagian mengerjakan fungsinya masing-masing dan dapat berkomunikasi melalui pertukaran data antara satu bagian dengan bagian lainnya melalui sistem informasi yang dikembangkan. Diagram-diagram yang digunakan pada rancangan yaitu diagram class dapat digunakan untuk memetakan hasil rancangan ke kode platform target, baik itu struktur tabel database maupun kode program Java. Sementara kode logik dapat dipetakan dari diagram aktifitas hasil rancangan. Hasil eksekusi kode program Java memperlihatkan fitur-fitur yang digunakan seperti transaksi penjualan dapat dijalankan dan digunakan sesuai dengan spesifikasi yang telah didefinisikan.

Kata kunci: Analisis Berorientasi Objek, Desain Berorientasi Objek, Unified Process, Sistem Informasi Penjualan, UML

Abstract: The aim of this study is to analyze and design a sales information system using a unified approach at PT. Sapta Karya. Literature studies, observations, and interviews are performed to define specifications of the system. In the analysis phase, activity diagrams is used to model business processes of the system, use case diagrams to define functional specifications, and sequence diagrams to specify interactions between sales, warehouse, and finance departments, and class diagrams are used to establish the static structure of the system which consists of a number of entities, such as Customers, Employees, Goods, and Transactions. In the design phase, the types of attributes and methods of the class diagram are defined, and activity diagrams for modeling the application logic. The unified approach in the development of an integrated system can be applied, where each division performs its own function and communicates through data exchange between them. The designed class diagrams can be used to map the database structure and Java program code. While the logic code can be mapped from the designed activity diagram. Evaluation of execution of the Java program shows that the functional features such as Sales Transactions can be executed and used according to the specifications.

Keywords: Object Oriented Analysis, Object Oriented Design, Unified Process, Sales Information System, UML

[†] Corresponding author : Yovita A. Maria (vitac@yahoo.com)

PENDAHULUAN

Pemanfaatan teknologi dan sistem informasi berbasis komputer telah diakui mampu meningkatkan pengelolaan sumber daya perusahaan dan layanan kepada pelanggan. Pemanfaatan teknologi informasi ini berlaku bagi setiap jenis industri termasuk perusahaan-perusahaan distributor yang menyediakan layanan distribusi produk-produk principal (pabrik) agar dapat sampai ke tangan konsumen dengan baik, cepat dan tepat.

Dalam kegiatan operasionalnya perusahaan penyedia jasa layanan distribusi barang, distributor ini seringkali terbentur kepada kerugian yang disebabkan oleh kegagalan membuat rencana penjualan dan pembuatan order produk atau barang terhadap pabrik, karena kelebihan stok, kekurangan stok atau stok yang tidak sesuai antara sistem pencatatan dengan stok fisik di tempat penyimpanan (gudang), serta masalah-masalah lain yang berkaitan dengan pengelolaan dan validitas data atau informasi yang digunakan. Sebagai contoh kasus dalam penelitian ini, penulis mengidentifikasi sejumlah masalah pada perusahaan distributor obat PT. Sapta Karya yang berlokasi di kota Makassar. Diantaranya berkaitan dengan koordinasi antara satu bagian dengan bagian lainnya. Meskipun telah menggunakan sistem berbasis komputer, akan tetapi koordinasi dan keterhubungan antara satu bagian dengan bagian lainnya masih sulit dilakukan, sistem berbasis komputer yang digunakan tidak dirancang untuk kebutuhan sistem yang terintegrasi, hal ini berdampak pada tidak efektif dan efisien-nya pekerjaan yang dilakukan.

Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sistem informasi pada PT. Sapta Karya, secara khusus menganalisis dan merancang sistem informasi penjualan perusahaan yang dapat meningkatkan interaksi dan aktivitas operasional antara satu bagian dengan bagian lainnya. Pengembangan sistem menggunakan pendekatan berorientasi objek yang telah menggabungkan beberapa pendekatan atau dikenal dengan pendekatan unified process.

PENDEKATAN BERORIENTASI OBJEK UNIFIED

Dalam tulisan [1] pendekatan berorientasi objek atau model berorientasi objek memperkenalkan modularisasi dalam ruang masalah, membuat praktek proses penggunaan kembali (reuse) yang baik dalam proses pemodelan konseptual. Sedangkan studi [2] menyatakan pendekatan berorientasi objek bertujuan untuk membantu menyederhanakan cara kita melihat dunia nyata sebagaimana mestinya (atau sebagaimana seharusnya) dan menterjemahkan pandangan kita ke dalam sistem perangkat lunak.

Pengembangan sistem dengan pendekatan berorientasi objek berbeda dari pengembangan konvensional yang memandang perangkat lunak sebagai fungsi dan data yang terisolasi. Pada pendekatan konvensional umumnya berfokus pada fungsi (prosedural) atau berfokus pada data (basis data). Sementara itu, pandangan berorientasi objek berpusat pada objek yang mengkombinasikan data dan fungsionalitas secara bersama.

Konsep Dasar Pendekatan Objek

- 1 Suatu teknik atau cara pendekatan baru dalam melihat permasalahan dari sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya).
- 2 Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata.
- 3 Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dari dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis, perancangan, pemrograman, dan pengujian perangkat lunak.
- 4 Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu.

Akan tetapi, menurut [3] kesalahan yang sering kali dibuat oleh pengembang perangkat lunak yaitu menggunakan teknik berorientasi objek hanya bagian-bagian tertentu saja atau pada sebagian tahapan saja. Misalnya, banyak kasus yang analisis dan desain sistemnya masih dilakukan dengan cara konvensional atau terstruktur, tetapi hanya pengembangan aplikasinya saja yang menggunakan implementasi atau pemrograman berorientasi objek. Seharusnya, teknologi / pendekatan berorientasi objek diterapkan mulai dari awal (tahap analisis) atau tidak sama sekali. Edward Berard dan Mayer juga menyatakan bahwa rekayasawan perangkat lunak harus mempertimbangkan penerapan semua tahap pengembangan perangkat lunak berbasis objek, termasuk analisis dan desainnya, Bukan hanya terbatas pada penggunaan bahasa pemrograman berorientasi objeknya saja. [3]

Karakteristik Pendekatan Berorientasi Objek

Menurut [4], beberapa karakteristik yang menjadi ciri-ciri dari pendekatan berorientasi objek adalah :

1. Pendekatan lebih pada data dan bukannya pada prosedur/fungsi.
2. Program besar dibagi menjadi objek-objek
3. Struktur data dirancang dan menjadi karakteristik dari objek-objek
4. Fungsi-fungsi yang mengoperasikan data tergabung dalam suatu objek yang sama.
5. Data tersembunyi dan terlindung dari fungsi/prosedur yang ada di luar.
6. Objek-objek dapat saling berkomunikasi dengan saling mengirim pesan satu sama lain.

Beberapa Metode Berorientasi Objek

Ada beberapa metode pengembangan perangkat lunak berorientasi objek yang sudah dikenal dan diantaranya adalah :

- Object Oriented Analysis (OOA) dan Object Oriented Design (OOD) dari Peter Coad dan Edward Yourdan [1990].
- Object Modeling Technique (OMT) dari James Rumbaugh, Michael Blaha, William Premerlan, Frederick Eddy dan William Lorensen [1991].

- Object Oriented Software Engineering (OOSE) dari Ivar Jacobson [1992].
- Booch Method dari Grady Booch [1994].
- Syntropy dari Steve Cook dan John Daniels [1994].
- UA (Unified Approach) dari James Rumbaugh, Grady Booch dan Ivar Jacobson [1997].

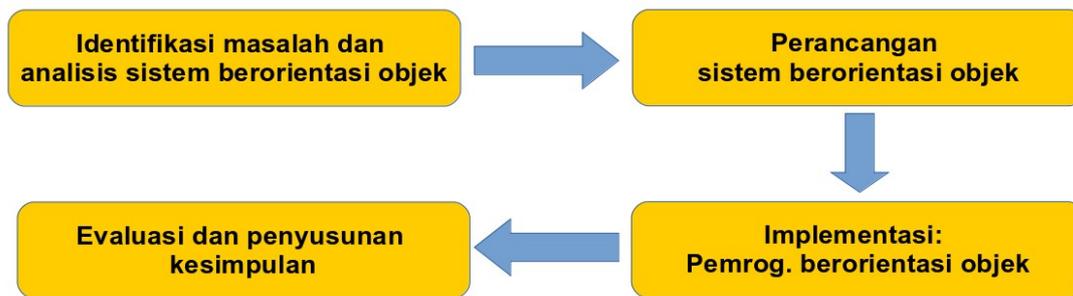
Metode Unified Approach (UA)

Unified Approach adalah sebuah metodologi untuk pengembangan perangkat lunak yang merupakan sebuah metodologi terpadu atau gabungan oleh beberapa metode utama yaitu metode Booch, Rumbaugh, dan Object Modeling Technique/Jacobson. Oleh [5], pendekatan UA ini, pada tahapan analisis melibatkan definisi diagram use-case, membangun model proses bisnis menggunakan diagram aktifitas, klasifikasi (diagram class) dan diagram interaksi. Sementara pada tahapan perancangan dilakukan dengan memodelkan atau melengkapi diagram class pada tahapan sebelumnya, membuat rancangan interface dan mendefinisikan alur logik program menggunakan diagram aktifitas. Jantung dari metode Unified Approach ini adalah use-case milik Jacobson yang saat ini telah mengembangkan pendekatan use case yang lebih baru atau disebut use case 2.0 [6]. Use-case tersebut memperlihatkan interaksi-interaksi yang terjadi antara user dan sistem, yang digunakan untuk menangkap kebutuhan-kebutuhan user atau sistem. Metode UA mencoba menggabungkan praktek terbaik, proses, dan panduan dengan menggunakan Unified Modeling Language (UML). UML merupakan bahasa pemodelan berorientasi objek yang terdiri dari kumpulan notasi-notasi yang digunakan dalam mendefinisikan spesifikasi, visualiasi dan mendokumentasi sistem atau perangkat lunak [7][8][9].

Diagram-diagram UML dapat digunakan untuk memodelkan struktur statik maupun dinamis dari sistem, beberapa diagram termasuk diantaranya use case diagram, class diagram, sequence diagram, statechart diagram, activity diagram, component diagram, and deployment diagram [10]. Meskipun UML cukup lengkap dalam menyediakan notasi pemodelan. Tetapi, ia bukanlah sebuah metodologi khusus, tidak ada langkah-langkah panduan di dalamnya. Sehingga pemodel memerlukan metodologi dalam pemakaiannya, seperti metode UA yang digunakan dalam studi ini.

METODE PENELITIAN

Sumber data dalam penelitian ini terdiri dari sumber data primer yang dikumpulkan melalui observerasi dan wawancara oleh penulis, serta data sekunder yang didapat melalui studi literatur. Pengambilan data terkait informasi spesifikasi kebutuhan dan analisis sistem yang sedang berjalan dilakukan pada PT. SAPTAKARYA, Jl. Hertasing Regency No. 15, Makassar – Sulawesi Selatan. Tahapan penelitian meliputi: identifikasi masalah dan analisis sistem, perancangan sistem, implementasi dan evaluasi sistem yang dikembangkan serta penyusunan kesimpulan penelitian.



Gambar 1. Desain Penelitian

Pengumpulan data melalui studi literatur dilakukan untuk mendapatkan data/informasi terkait pengembangan sistem berorientasi objek menggunakan pendekatan unified meliputi tahapan analisis, perancangan dan implementasinya, observasi dan wawancara untuk mengetahui proses-proses bisnis sistem, hubungan setiap bagian di perusahaan yang terdiri dari tiga divisi/bagian besar yaitu penjualan, gudang dan keuangan. Data/informasi yang diperlukan dalam interaksi antara bagian tersebut. Setelah sistem yang dirancang diimplementasikan ke dalam sistem berbasis komputer yang dapat dieksekusi, penulis melakukan pengambilan data pengujian untuk mengevaluasi sistem apakah sistem informasi yang dikembangkan dapat menyelesaikan masalah yang dibicarakan pada latarbelakang masalah penelitian ini. Metode analisis dan desain yang digunakan merujuk pada pendekatan UA, tetapi dengan tetap menjaga proses analisis dan desain yang lebih sederhana dengan melakukan analisis atau pemodelan proses bisnis terlebih dahulu sebelum pemodelan use-case. Gambar 1 memperlihatkan desain penelitian yang digunakan.

1. Tahapan analisis awal yaitu menganalisis kondisi awal sistem untuk memahami proses bisnis penjualan, tools yang digunakan dalam membantu proses analisis ini yaitu dokumen flowchart untuk mendapatkan gambaran secara detail aliran dokumen dari satu bagian ke bagian lain.
2. Tahapan pemodelan analisis berorientasi objek (OOA). Setelah aliran dokumen dari satu bagian ke bagian lain dipahami, maka selanjutnya dilakukan pemodelan analisis perilaku sistem. Pemodelan analisis terdiri dari pemodelan proses bisnis menggunakan diagram aktivitas (activity diagram), pemodelan fungsional sistem informasi penjualan secara menyeluruh menggunakan usecase (diagram dan text/skenario), pemodelan interaksi antara objek di sistem menggunakan diagram sequence, serta pemodelan klasifikasi menggunakan diagram class.
3. Tahapan desain berorientasi objek (OOD). Dari artifak analisis, selanjutnya penulis memilih sejumlah fitur dari usecase untuk dituangkan dalam proses rancangan berorientasi objek. Pada tahapan ini proses yang dirancang yaitu proses order yang melibatkan pelanggan, bagian penjualan, gudang dan bagian keuangan. Proses rancangan berorientasi objek menggunakan diagram class dan diagram aktivitas.

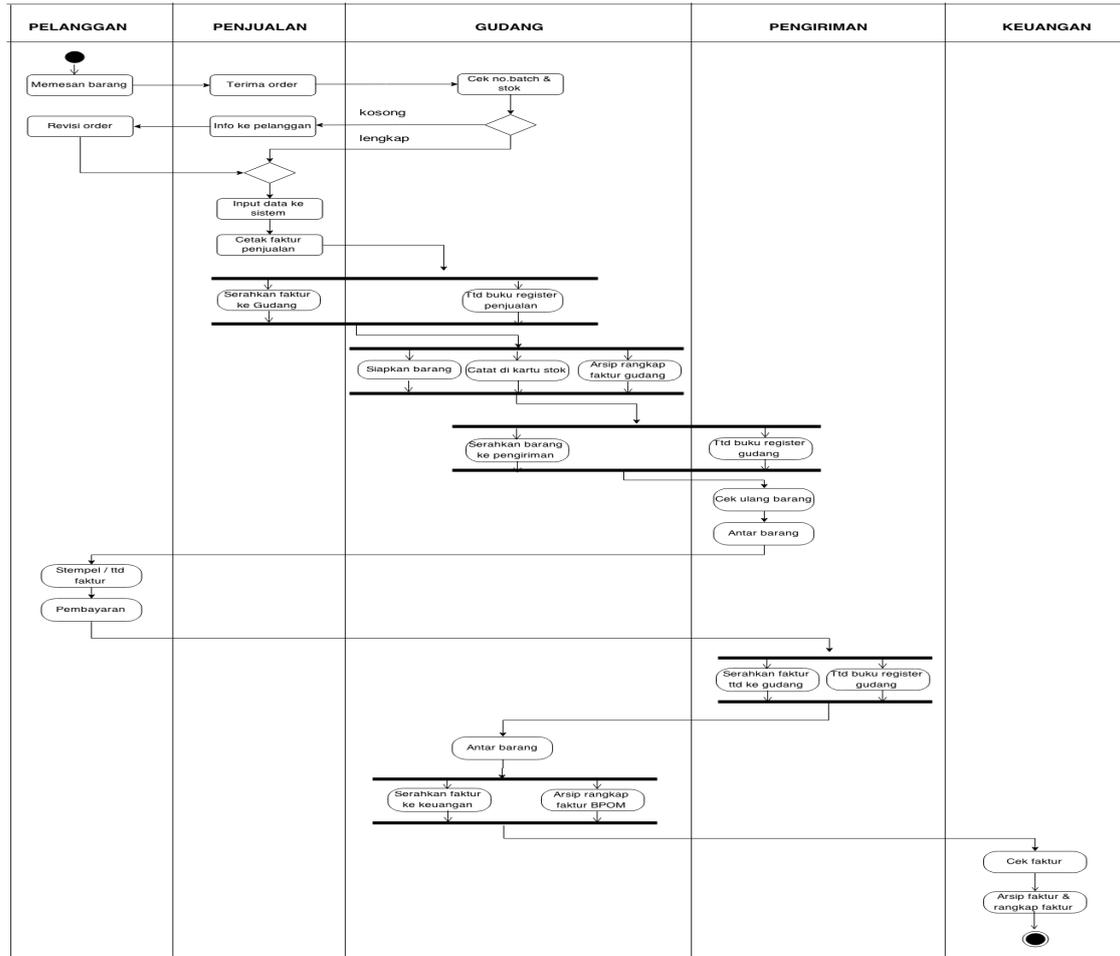
4. Tahapan pemrograman berorientasi objek (OOP). Hasil rancangan selanjutnya diimplementasikan kedalam bahasa pemrograman berorientasi objek java. Pada tahapan ini diperlihatkan bagaimana pemetaan (mapping) diagram class kedalam kode program Java. Kelas diagram digunakan untuk pemetaan struktur data dan operasi, juga kebutuhan sistem basis datanya. Sementara diagram aktivitas digunakan untuk pemetaan logika aplikasi.
5. Pengujian. Tahapan ini memperlihatkan proses pengujian hasil implementasi dalam program Java. Apakah fitur dan proses pilihan yang dirancang serta diimplementasikan dapat berjalan sesuai dengan spesifikasi yang ditetapkan. Hasil pengujian ini dijadikan sebagai dasar penyusunan kesimpulan penelitian.

HASIL DAN PEMBAHASAN

Analisis Sistem Berorientasi Objek

A. Analisis Proses Bisnis

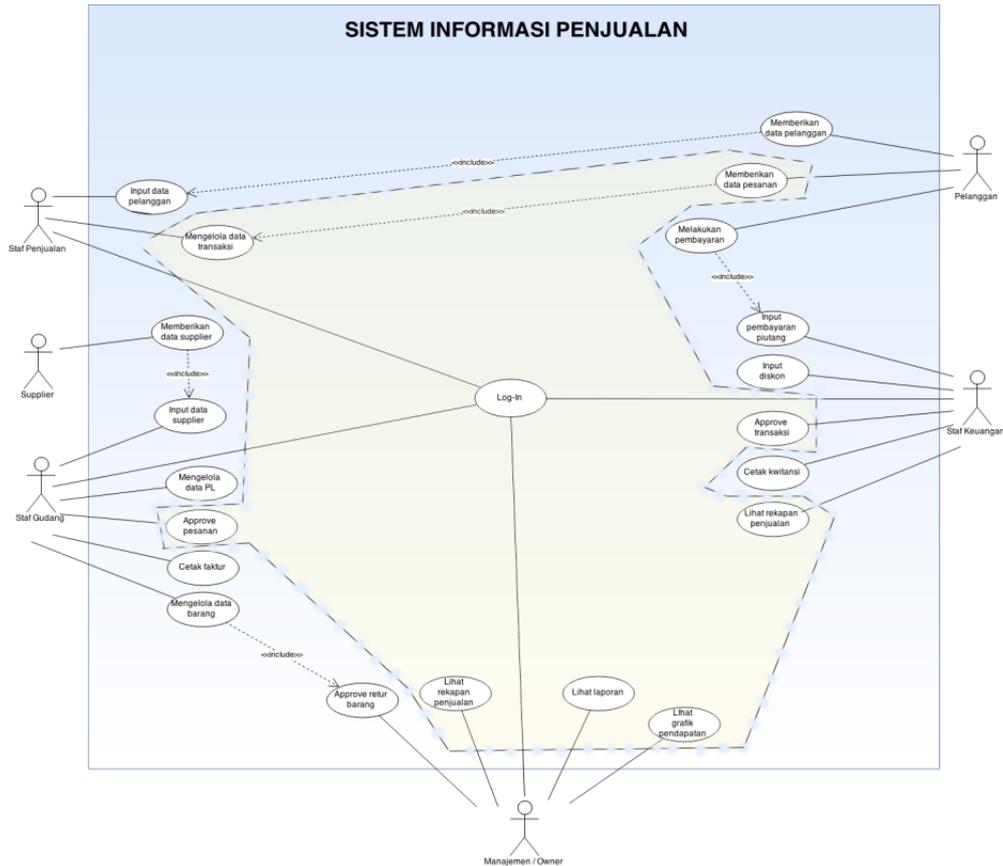
Analisis sistem Proses bisnis penjualan pada PT. Saptakarya dimulai ketika pelanggan memesan barang dan memberikan data ordernya tersebut ke bagian penjualan. Kebutuhan analisis proses bisnis sistem dimodelkan menggunakan activity diagram yang memperlihatkan objek Pelanggan, Penjualan, Gudang, Pengiriman dan Keuangan serta aktifitas-aktifitas yang terjadi antara objek tersebut. Hasil pemodelan ditunjukkan pada Gambar 2 dimana proses dimulai dari Pelanggan yang melakukan pemesanan barang, kemudian diterima oleh bagian Penjualan yang diteruskan ke bagian Gudang untuk diperiksa apakah Stok tersedia. Bagian Pengiriman melakukan pengiriman barang, dan proses berakhir pada bagian Keuangan dengan memeriksa dan mengarsipkan faktur penjualan.



Gambar 2. Diagram Aktifitas Proses Bisnis Penjualan PT.Saptakarya

B. Analisis Spesifikasi Kebutuhan Sistem

Proses analisis selanjutnya yaitu memodelkan spesifikasi kebutuhan sistem menggunakan use-case diagram. Aktor-aktor yang teridentifikasi dan terlibat di sistem yaitu: Pelanggan, Staf Penjualan, Supplier, Staf Keuangan dan Staf Gudang. Sementara use-case yang diidentifikasi diantaranya, yaitu: Menginput data pesanan, mengelola data transaksi, Cetak Kwitansi, Approve Pesanan, dan Approve Transaksi. Pada use-case diagram yang diperlihatkan pada Gambar 3, terpetakan kebutuhan sistem yang merupakan hasil analisis penulis yang didapatkan melalui diagram aktivitas pada proses analisa kondisi awal dan juga merupakan hasil dari tinjauan pustaka yang dilakukan oleh penulis terhadap penelitian-penelitian seputar sistem informasi penjualan. Bagian yang diarsir pada diagram, merupakan proses-proses yang dipilih penulis untuk dijadikan contoh fitur yang dilanjutkan ke tahap desain (OOD).



Gambar 3. Diagram Use-case

Pada Tabel 1 disajikan salah satu skenario dari usecase diagram yang dijabarkan dalam bentuk deksripsi use case.

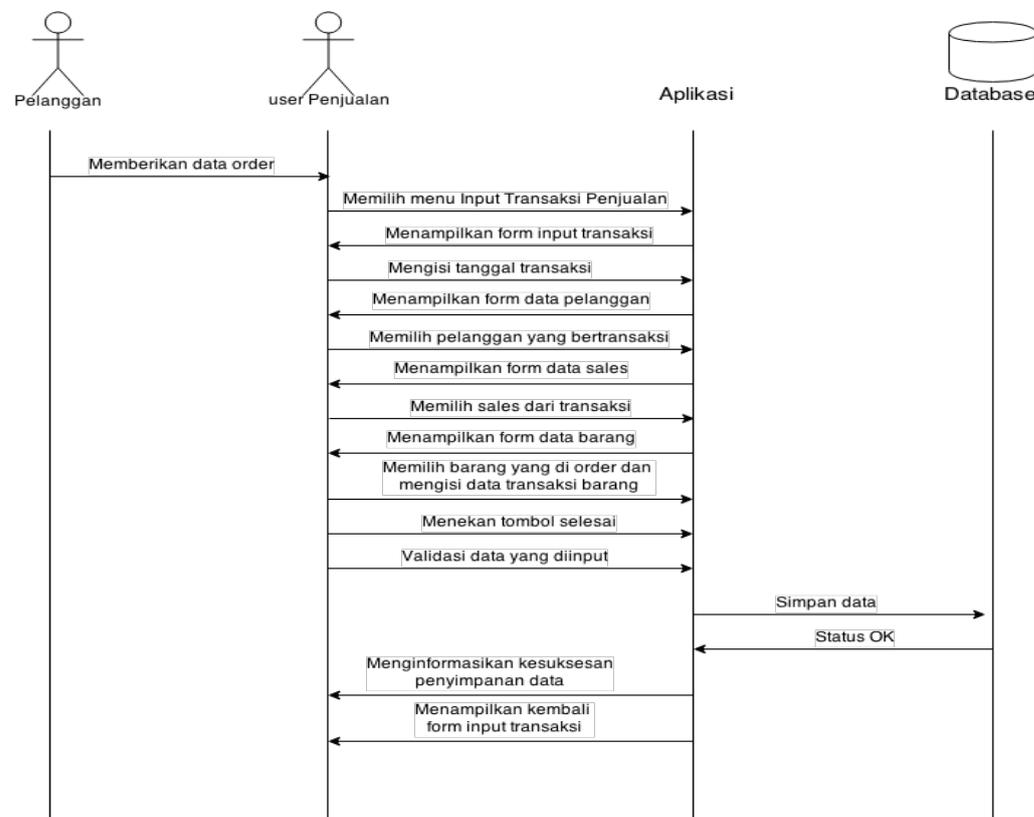
Tabel 1: Dekscrip use-case data transaksi

Nama Use-case : Input data transaksi	
Aktor : Pelanggan, Staf Penjualan	
Deskripsi : Memasukkan / mengentri data transaksi ke dalam sistem.	
Exception : Proses penyimpanan transaksi gagal.	
Triggering Event : Aktor pelanggan memberikan data order kepada staf penjualan, dan aktor penjualan memilih menu Penjualan - Input Transaksi Penjualan.	
Trigger type : Eksternal	
Pre Condition : Pelanggan telah memberikan data order dan aktor penjualan telah login dalam sistemnya.	
Langkah-langkah	Informasi
1. Aktor penjualan memilih menu input transaksi penjualan.	Form menu penjualan
2. Aktor penjualan mengisi tanggal transaksi.	Form input transaksi penjualan Di bawah ini terlampir salah satu skenario dari usecase diagram.
3. Aktor penjualan memilih data pelanggan yang bertransaksi.	Data pelanggan, Form data pelanggan
4. Aktor penjualan memilih sales yang memberikan order (kantor/salesman).	Data sales, Form data sales

5 Aktor penjualan memilih jenis transaksi.	Form input transaksi penjualan
6. Aktor penjualan memilih barang-barang yang dipesan dan mengisi / memilih data yang diperlukan, seperti mencocokkan harga, mengisi jumlah / kuantiti, dan no.batch.	Data barang, Data batch, Form data barang
7. Setelah data lengkap, aktor penjualan menekan tombol selesai.	Form input transaksi penjualan
8. Validasi data yang diinput	
9. Data transaksi disimpan ke database dengan status 'NA' (Not Approved yet)	Data transaksi , Data detail transaksi, Message Box
Post Condition : Aktor penjualan berhasil menginput transaksi,	

C. Analisis Interaksi

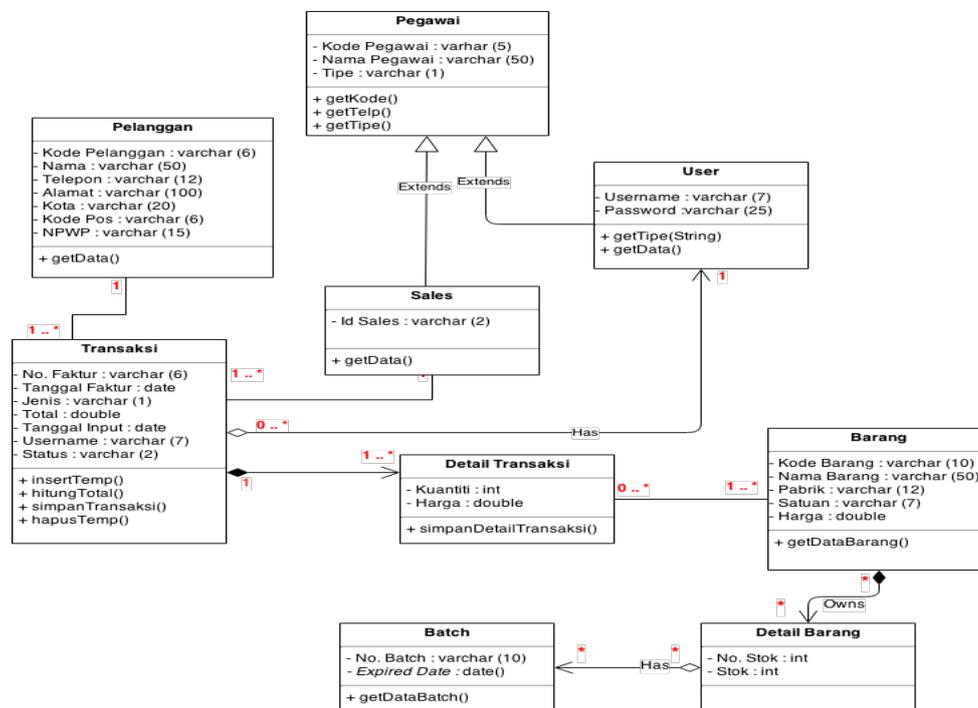
Berdasarkan skenario use-case yang telah dibuat, penulis memodelkan interaksi antar objek di sistem dengan menggunakan sequence diagram, yang melibatkan aktor, layer aplikasi, dan layer database. Gambar 4 memperlihatkan salah contoh pemodelan interaksi yang terjadi untuk proses transaksi.



Gambar 4. Diagram Interaksi

Desain Sistem Berorientasi Objek

Pemodelan klasifikasi yang dilakukan terbagi dalam dua tahap, yaitu diagram kelas pada tahap analisis, dan diagram kelas pada tahap desain. Diagram kelas pada tahap analisis digunakan untuk mengidentifikasi entitas-entitas apa saja yang ada dan bagaimana hubungannya, sementara pada tahapan desain, kelas diagram hasil analisis dikembangkan sehingga berisi informasi yang telah lengkap, seperti definisi atribut dan metode serta tipe yang digunakan, sehingga memungkinkan melakukan pemetaan ke kode program target. Kelas-kelas yang digunakan, diantaranya yaitu kelas Pegawai, Pelanggan, User, Transaksi, Salses, Detail Transaksasi dan Barang. Sementara untuk rancangan logik sistem secara detail dimodelkan menggunakan diagram aktifitas. Gambar 5 menunjukkan contoh pemodelan diagram class yang dilakukan pada tahap desain sistem.

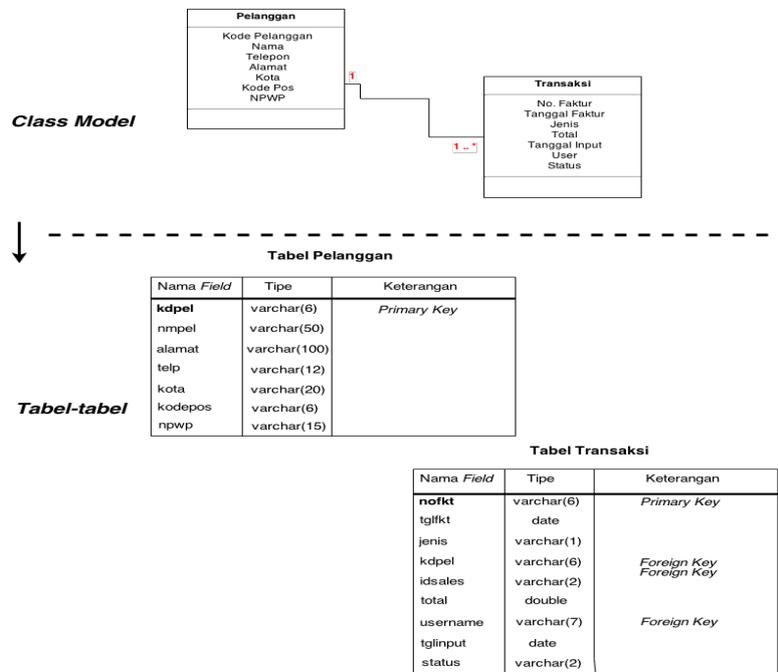


Gambar 5. Class Diagram pada tahapan desain

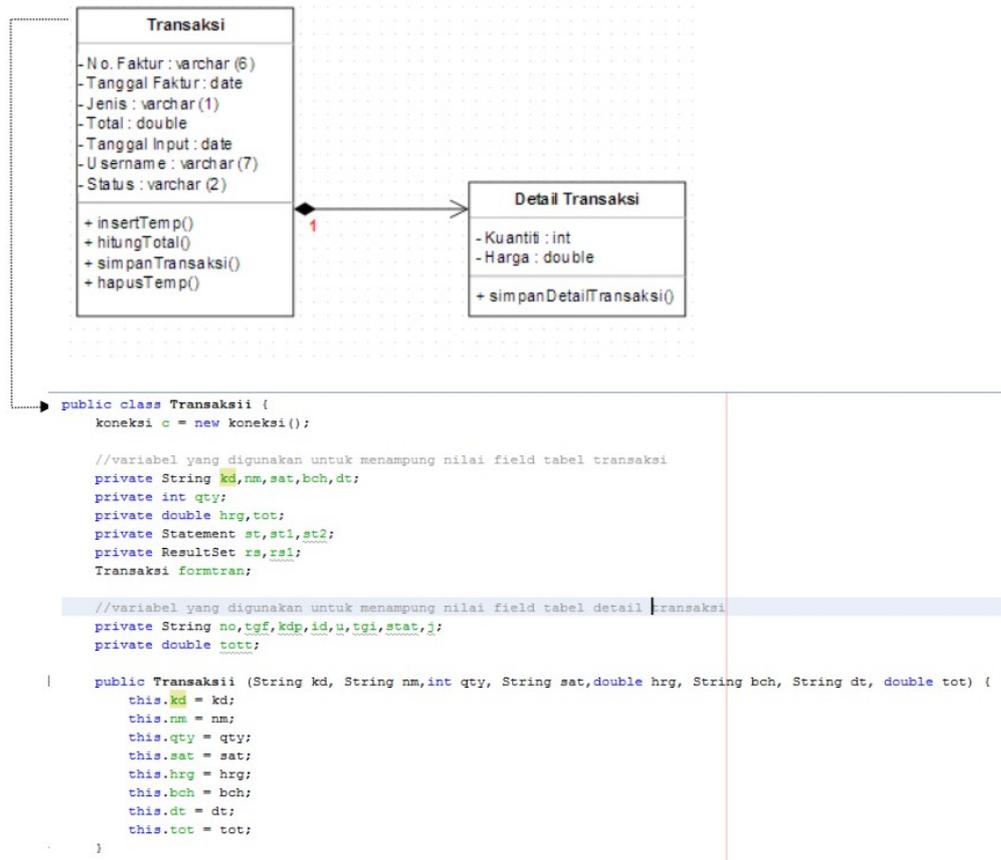
Pemetaan Hasil Rancangan ke Platform Target

Setelah proses desain selesai. Selanjutnya dapat dilakukan pemetaan diagram kelas hasil desain (OOD) ke database (struktur data), dan ke kode program Java sebagai contoh platform targetnya (OOP). Pada gambar 6 memperlihatkan cara penulis memetakan diagram kelas transaksi menjadi tabel transaksi dan detail transaksi dalam database. Sebagai contoh pada kelas Pelanggan terdapat atribut kode pelanggan dengan tipe varchar(6) dipetakan menjadi field kdpel dengan tipe yang sama. Demikian pula untuk atribut yang lain dan kelas-kelas lainnya yang perlu dipetakan dilakukan teknik yang sama. Untuk pemetaan ke kode program Java diperlihatkan pada Gambar 7. Nama kelas diagram dipetakan menjadi nama kelas di Java, atribut-atribut pada kelas diagram dipetakan menjadi atribut atau field kelas

beserta tipe-nya. Demikian pula metode-metode yang didefinisikan pada kelas diagram dapat dengan mudah dipetakan ke metode-metode dalam kode program Java. Sementara untuk logika yang ada di dalam metode dapat dipetakan melalui diagram aktifitas hasil rancangan yang memuat informasi detail logika atau pemrosesan yang ada pada metode, jika metode tersebut merupakan metode pemrosesan data.



Gambar 6. Pemetaan Rancangan Diagram Class ke Tabel Database

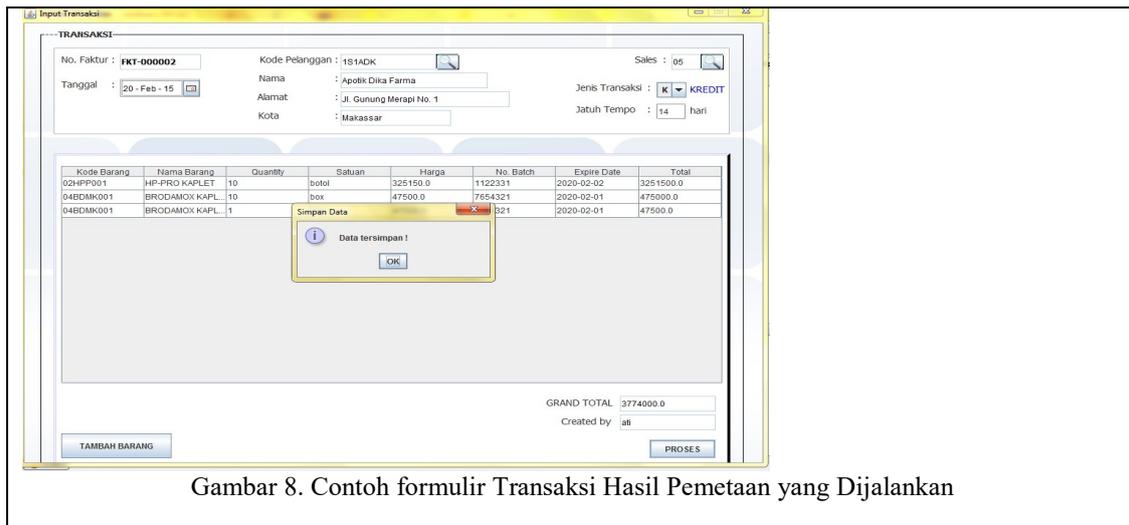


Gambar 7. Pemetaan Rancangan Diagram Class ke Kode Program Java

Pengujian Sistem

Pada tahapan ini, penulis melakukan pengujian terhadap hasil rancangan dan hasil pemetaan ke kode program Java. Fitur-fitur yang dipilih sebagai contoh implementasi, selanjutnya dieksekusi pada platform Target yaitu bahasa pemrograman Java, setelah proses kompilasi kode sumber. Selanjutnya kode hasil kompilasi dijalankan diatas sistem operasi java runtime environment pada platform sistem operasi windows. Dari hasil pengujian fitur-fitur yang dipilih didapat bahwa kode program hasil pemetaan dapat dijalankan dan digunakan. Gambar 8 memperlihatkan salah satu fitur utama yang diuji yaitu Transaksi.

Test Factor	Hasil	Keterangan
Proses simpan data transaksi	☐	Berhasil, karena data transaksi berhasil disimpan
Tampilan		



Gambar 8. Contoh formulir Transaksi Hasil Pemetaan yang Dijalankan

KESIMPULAN

Dari penelitian yang dilakukan, disimpulkan bahwa:

1. Pendekatan terpadu (unified) yang digunakan dalam proses analisis dan desain berorientasi objek yang melibatkan diagram aktifitas, usecase, interaksi dan diagram class dalam mengembangkan sistem informasi penjualan pada PT. Saptakarya dapat diterapkan dalam mengembangkan sistem informasi yang terintegrasi.
2. Diagram-diagram yang digunakan pada rancangan yaitu diagram class dapat digunakan untuk memetakan hasil rancangan ke kode platform target, baik itu struktur Tabel database maupun kode program Java. Sementara kode logik dapat dipetakan dari diagram aktifitas hasil rancangan. Hasil eksekusi kode program Java memperlihatkan fitur-fitur yang digunakan seperti Transaksi Penjualan dapat dijalankan dan digunakan sesuai dengan spesifikasi yang telah didefinisikan.

DAFTAR PUSTAKA

- [1] O. Pastor, J. Gomez, E. Insfran, V. Pelechano, "The OO-Method approach for information systems modeling: from object-oriented conceptual modeling to automated programming," *Information Systems*, Vol. 26, No.7, pp.507-534, 2001.
- [2] J.M. Nerson, "Applying Object Oriented Analysis and Design," *Communications of the ACM*, Vol. 35, No. 9, pp.63-74. September 1992.
- [3] A. Kahate, *Object Oriented Analysis and Design*. New Delhi: Tata McGraw-Hill Publishing Company Limited, 2004.
- [4] A. Nugroho, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*. Bandung: Penerbit Informatika Bandung, 2002.
- [5] A. Behrami, *Object Oriented Systems Development*. The McGraw Hill Book Co., Singapore, 1999.
- [6] I. Jacobson, "Use-Case 2.0 The Lightness of User Stories with the Power of Modeling," Diakses pada: June 13, 2016. [Online]. Tersedia di: <http://www.dreamevent.nl/pdf/DREAM14/IvarJacobson-UML2.0.pdf>

- [7] G. Booch, J. Rumbaugh, I. Jacobson, *The Unified Modeling Language User Guide*, First Edition. Publisher: Addison Wesley, October 20, 1998
- [8] T. Clark, A. Evans, "Foundation of the Unified Modeling Language," Diakses pada: June 13, 2016. [Online]. Tersedia di https://www.researchgate.net/publication/2464719_Foundations_of_the_Unified_Modeling_Language
- [9] A. Knapp , H. Storrle, "Unified Modeling Language 2.0," in 2005 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'05), 2005, pp. 9-9, doi: 10.1109/VLHCC.2005.65.
- [10] S. Lee, "Unified Modeling Language (UML) for Database Systems and Computer Applications," *International Journal of Database Theory and Application*, Vol. 5, No.1, pp.157-164, march 2012.