

## BAB III

### ANALISIS

#### **3.1 Analisis Kebutuhan BPEL untuk Aplikasi e-Commerce**

*E-commerce* yang telah dijelaskan pada II.3 melibatkan beberapa aplikasi yang masing-masing memiliki peran dalam suatu proses bisnis tertentu. Hubungan antar aplikasi tersebut dapat digambarkan dalam suatu sistem terdistribusi dengan setiap *node* adalah aplikasi dengan fungsionalitasnya masing-masing dalam proses bisnis. Oleh karena *e-commerce* menjalankan proses bisnisnya melalui komunikasi internet, maka teknologi *web service* sangat cocok untuk diterapkan pada aplikasi jenis ini. Sebelum adanya teknologi *web service*, kebutuhan dalam sistem terdistribusi dipenuhi oleh beberapa teknologi dasar. Teknologi tersebut antara lain *Remote invocation protocols* digunakan untuk menghubungkan komponen di server yang berbeda.

Salah satu contohnya adalah *Remote Procedure Call (RPC)* semacam DCOM dan CORBA, Adapters yang digunakan untuk memungkinkan komunikasi antar lojik dari aplikasi yang berada pada platform berbeda, *Object request broker* dan perangkat lunak lain yang menyediakan mekanisme *runtime* pada aplikasi sistem terdistribusi. Akan tetapi ada suatu kendala dalam penerapan teknologi-teknologi terdahulu yaitu ketergantungannya terhadap platform. Jadi, komunikasi antar komponen dalam sistem terdistribusi membutuhkan semacam persetujuan format pesan yang dipakai dan tidak ada standard umum yang berlaku. Hal ini berbeda dengan teknologi *web service* yang memiliki suatu standard khusus yang didefinisikan dalam spesifikasi WSDL dan SOAP. Seperti yang telah dibahas dalam Bab II.1.1, *web service* membentuk semacam *integration layer* yang menghubungkan service yang berkaitan. Dengan adanya kemampuan *web-service* yang telah dijelaskan sebelumnya, maka kebutuhan akan teknologi di atas dapat digantikan oleh teknologi *web service* yang memiliki sifat yang lebih fleksibel terhadap platform aplikasi. Dari keterangan di atas, dapat dilihat bahwa teknologi *web service* dapat digunakan untuk komunikasi antara aplikasi yang berperan dalam *e-commerce*.

Perusahaan *e-commerce* tersebut tentu saja akan berkoordinasi dengan perusahaan pendukung bisnisnya yang memiliki aplikasi sendiri. Dalam komunikasi antara perusahaan *e-commerce* dengan perusahaan rekanan, digunakan komunikasi *web service* dengan mekanisme pertukaran pesan melalui protokol HTTP. Dalam komunikasi tersebut akan digunakan komponen dalam teknologi *web service* seperti SOAP dan WSDL seperti yang telah dijelaskan pada Bab II.1.1. Suatu perusahaan pasti memiliki proses bisnis yang dijalankan dan dibutuhkan suatu bahasa untuk mendefinisikannya. Dari penjelasan pada Bab II.3 dapat dilihat bahwa BPEL dapat

digunakan untuk mengatasi permasalahan definisi proses bisnis. BPEL dapat digunakan untuk menggambarkan jalannya suatu proses bisnis dalam perusahaan melalui definisi dari aktivitas yang berperan. Aktivitas tersebut dapat berupa aktivitas dasar dan juga aktivitas terstruktur seperti yang telah dijelaskan pada Bab II.3. BPEL merupakan suatu ekstensi *web service* yang telah didefinisikan untuk menangani proses bisnis dari aplikasi yang dimiliki oleh perusahaan *e-commerce*.

Pemodelan proses bisnis *e-commerce* dengan BPEL memiliki keuntungan karena kesederhanaannya dalam pendefinisian. Pada Bab II.3 telah dijelaskan bahwa BPEL memanfaatkan deskripsi WSDL untuk mendefinisikan pesan yang dikomunikasikan.

Dari penjelasan BPEL pada Bab II.3 dan analisis penerapannya pada aplikasi *e-commerce*, maka dapat disimpulkan beberapa keuntungan penerapan BPEL pada aplikasi *e-commerce* adalah sebagai berikut:

1. Pendefinisian dan otomatisasi proses bisnis tidak perlu dilakukan pada level *coding* dan *programming*, hanya perlu dilakukan pada level desain dan abstraksi. Sehingga pengembangan dan modifikasi proses bisnis dapat dilakukan dengan cepat dan lebih sederhana. Manfaat lain yang dicapai dengan adanya hal ini adalah dapat dipenuhi kebutuhan web service dalam aplikasi *e-commerce* untuk melakukan integrasi.
2. Proses bisnis *e-commerce* biasanya melibatkan *long-running processes*. Masalah yang timbul dengan adanya *long-running processes* adalah persistensi dari proses. Semakin lama proses diharapkan untuk berjalan, maka semakin penting masalah persistensi harus dijaga. Manfaat utama dari implementasi *long-running processes* sebagai proses BPEL adalah bahwa persistensi proses dapat diatur oleh BPEL *engine*.
3. Dalam aplikasi *e-commerce*, salah satu masalah penting adalah manajemen dari *endpoint*. Seperti yang telah dijelaskan pada bab II.1.1 tentang WSDL, *endpoint* mengandung lokasi servis, data endpoint termasuk *physical address* dan juga informasi protokol. Akan tetapi dalam WSDL hanya didukung definisi endpoints yang statis. Aplikasi ini membutuhkan manajemen endpoints yang dinamik dan mudah untuk dikendalikan. BPEL dapat mengatasi permasalahan ini dengan adanya properti *partner link*. Seperti yang telah dijelaskan pada bab II.3 bahwa *partner link* dapat dispesifikasikan pada saat *deployment* ataupun *runtime* dan dapat dimanipulasi secara langsung dalam proses, maka BPEL menyediakan solusi yang baik untuk manajemen *endpoint* yang awalnya statik menjadi lebih dinamik.

### 3.2 Analisis Kebutuhan Penerapan ACID pada E-Commerce

Dalam proses bisnis *e-commerce* sangat mungkin terjadi operasi pembatalan yang dilakukan oleh pengguna sehingga mengakibatkan adanya beberapa aktivitas yang harus dibatalkan. Mekanisme untuk menangani masalah ini tidaklah semudah mekanisme pembatalan aktivitas biasa karena melibatkan beberapa aktivitas yang terjadi pada aplikasi *e-commerce* dan juga pada aplikasi partner yang berkaitan. Selain karena aksi pembatalan dari pengguna, masalah ini juga mungkin muncul jika terjadi *failure* ataupun kegagalan pada tengah-tengah eksekusi proses bisnis *e-commerce*.

Seperti yang telah dijelaskan pada bab II.4.2 bahwa ada dua tipe transaksi dalam hubungannya dengan BPEL dan SOA yaitu *atomic transaction* dan *business activity*. Mekanisme penanganan *failure* pada transaksi tipe pertama mirip dengan penanganan semantik ACID pada sistem basisdata relasional yaitu melalui mekanisme *locking resource* dengan protokol tertentu seperti *two-phase commit protocol* (2PC). Teknologi web service menangani transaksi jenis ini dengan spesifikasi dari WS-Transaction.

Penanganan *failure* dengan cara yang sama tidak dapat diterapkan pada transaksi yang tipe *business activity* karena *resource locking* untuk waktu yang lama sangat tidak praktis. Pendekatan yang dilakukan adalah dengan mendefinisikan aktivitas *compensation* yang dapat mengulangi aksi sebelumnya disamping tetap mengikat *resource* dengan *locking-based protocol* seperti 2PC. Akan tetapi dalam hal ini, *resource* tidak diikat selama transaksi, dan informasi status dari *resource* dapat diakses meskipun aktivitas proses bisnis belum selesai. Teknologi *web service* menangani masalah pada transaksi jenis ini dengan spesifikasi WS-BusinessActivity.

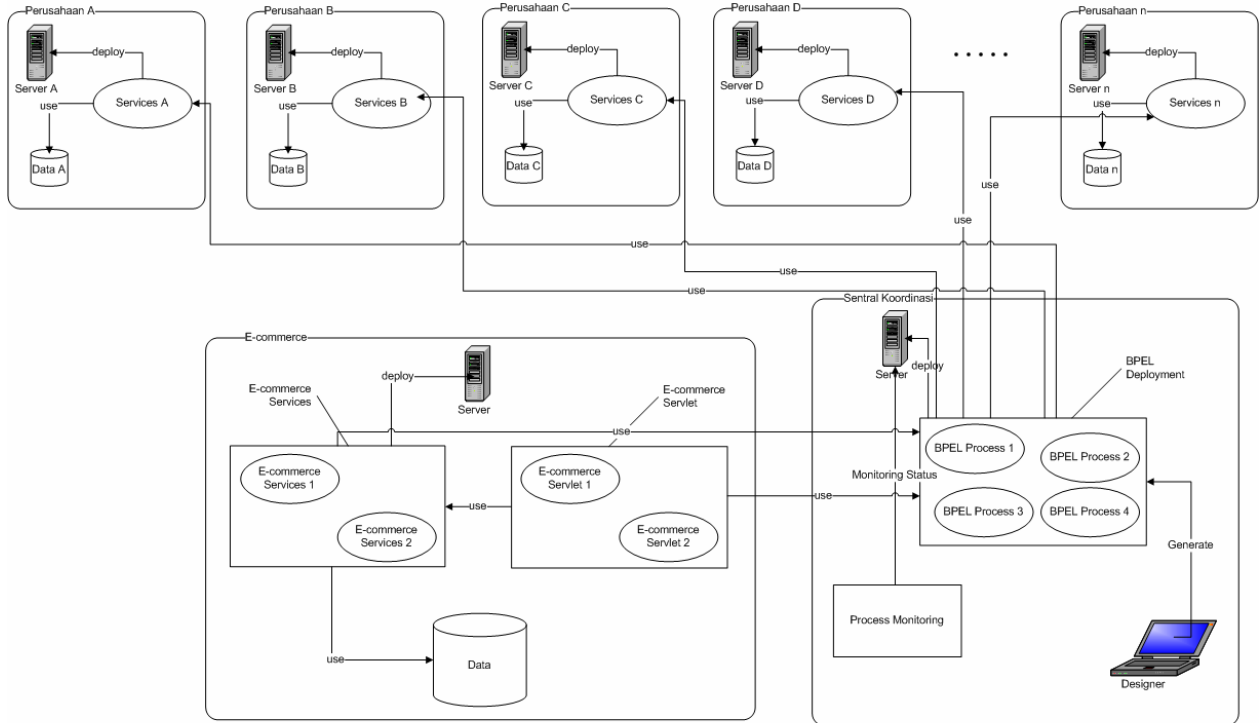
WS-AtomicTransaction dan WS-BusinessActivity secara bersamaan menggantikan spesifikasi WS-Transaction yang telah ada sebelumnya. Tapi dua spesifikasi ini belum secara luas diterima, dan BPEL tidak bergantung pada kedua spesifikasi tersebut. Akan tetapi BPEL mendukung mekanisme penanganannya. Beberapa aktivitas dalam BPEL secara eksplisit menspesifikasikan semantik dari ACID (misalnya, aktivitas *assign* adalah *atomic* dan *scopes* mungkin didesain sebagai serializable yang berkaitan dengan semantik *isolation*). BPEL juga mendukung jenis transaksi yang *long-running* melalui adanya aktivitas *compensations* dan *compensations handler*. Mekanisme *compensation* ini terintegrasi dengan mekanisme *fault-handling* yang akan dieksekusi jika terjadi suatu kegagalan.

### 3.3 Model Generik Aplikasi e-Commerce

#### 3.3.1 Gambaran Umum Model

Pengertian model dalam hal ini adalah model koordinasi antar aplikasi perusahaan yang memiliki peranan dalam proses bisnis *e-commerce*. Secara umum, model dalam bisnis *e-commerce* dapat melibatkan banyak jenis perusahaan dengan berbagai jenis layanannya tergantung kepada bisnis dari *e-commerce*. Antara perusahaan tersebut terdapat interaksi untuk menggambarkan proses bisnis keseluruhan yang dijalankan oleh *e-commerce*. Gambaran interaksi tersebut dapat dilihat pada gambar III-1. Sebagai contoh, perusahaan penyedia produk berinteraksi dengan konsumen untuk mengambil pemesanan produk dari pelanggan. Setelah masukan diterima, maka transaksi dijalankan dengan melibatkan dua subproses yaitu *account* dan *delivery*. Subproses *account* akan menangani pembayaran dari konsumen dengan melibatkan layanan perusahaan kredit dan bank. Subproses *delivery* menangani pengiriman barang pesanan dengan melibatkan *delivery agent* yang terhubung ke beberapa *delivery transport* yang disediakan.

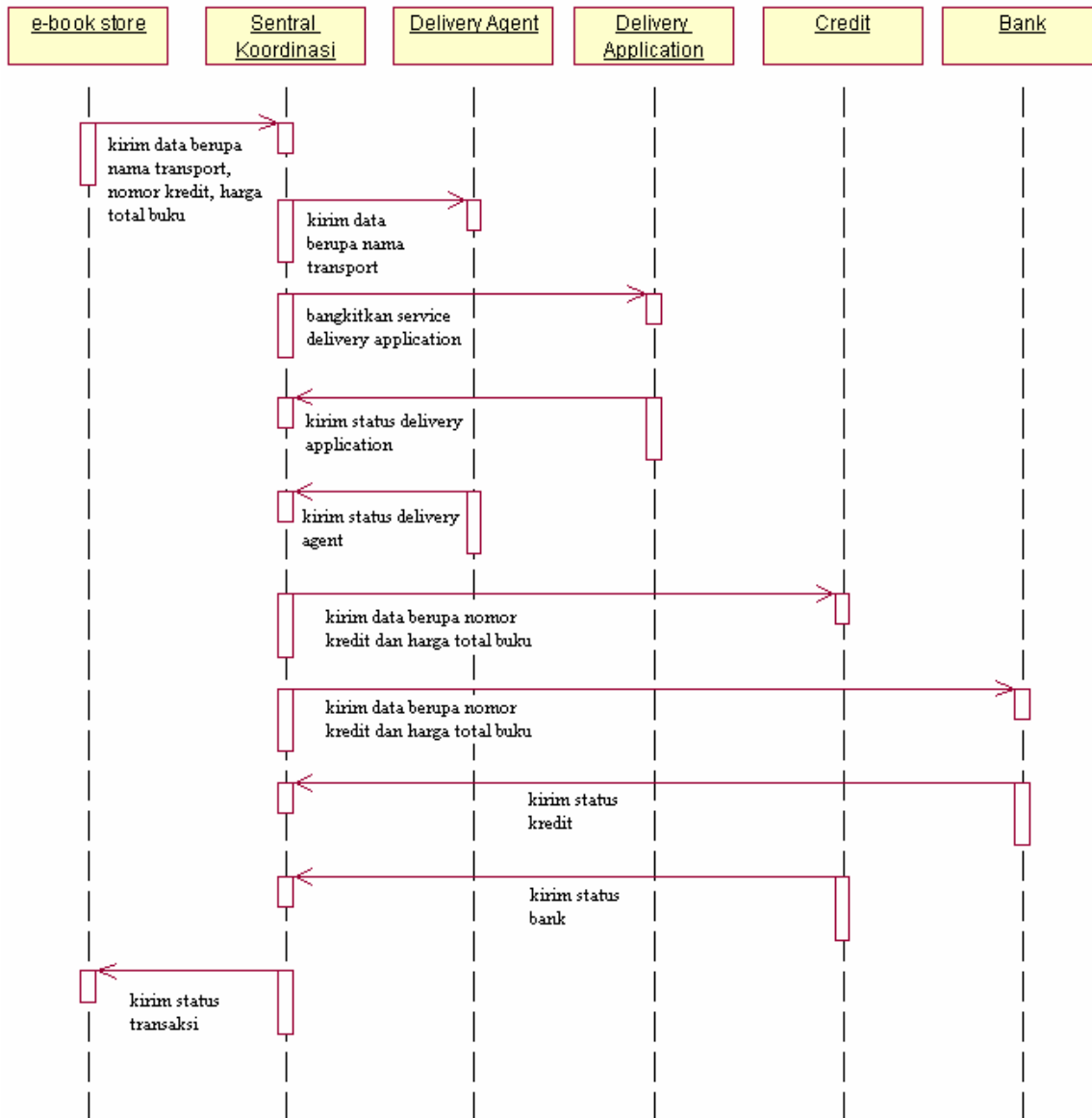
Suatu *service* yang ada pada aplikasi *e-commerce* akan meminta *service* lain yang ada pada perusahaan rekanan lain untuk menjalankan proses bisnis yang diinginkan. Jika terjadi transaksi pembelian pada aplikasi *e-commerce* maka pesan akan dikirimkan kepada *service* perusahaan kredit untuk menangani proses pembayaran konsumen melalui kartu kredit. Untuk selanjutnya, *service* ini akan memanfaatkan *service* lain yang ada pada bank untuk melanjutkan proses pembayaran. Selain ke perusahaan kredit, pesan juga akan dikirimkan pada perusahaan *delivery* untuk menangani proses pengiriman barang. Pesan ini akan membangkitkan layanan yang terdapat pada perusahaan transportasi untuk mekanisme pengiriman barang lebih lanjut. Interaksi semacam ini akan ditangani BPEL untuk menggambarkan proses bisnis yang akan dijalankan. Hasil yang diharapkan adalah aktivitas dalam interaksi antara *service* perusahaan dapat berjalan sesuai dengan proses bisnis dengan tetap menjaga semantik ACID pada proses bisnis.



**Gambar III-1 Model Generik E-commerce**

Salah satu contoh yang dapat digunakan untuk menggambarkan interaksi antar komponen dalam model yang terbentuk adalah skenario transaksi pembelian buku pada e-book store. Perusahaan yang berperan dalam skenario tersebut ada lima perusahaan, yaitu *bank*, *credit*, *delivery agent*, *delivery application*, dan aplikasi *e-commerce* tersebut adalah *e-book store*. Sesuai dengan gambaran model pada Gambar III-1, maka *e-book store* memanggil proses yang ada pada sentral koordinasi. Proses dalam sentral koordinasi menggunakan *services* yang disediakan oleh kelima perusahaan yang telah disebutkan untuk menjalankan proses bisnis. Skenario tersebut hanya diterapkan pada submodel yang merupakan komponen inti dalam proses bisnis transaksi pembelian. Gambaran skenario tersebut dapat dilihat pada Gambar III-2.

Dalam skenario transaksi pembelian, *e-book store* mengirimkan masukan berupa transport yang dipilih, nomor kartu kredit, dan total harga buku yang dibeli ke proses yang ada di sentral koordinasi. Dari sentral koordinasi, informasi jenis transport akan dilanjutkan ke perusahaan *delivery agent* dan *delivery application* untuk mengembalikan status *delivery* dari transaksi pembelian. Sementara informasi nomor kartu kredit dan total harga disalurkan ke perusahaan *credit* dan bank untuk mengambil status pembayaran. Semua status dari perusahaan rekanan akan disimpan sebagai hasil keluaran proses sentral koordinasi dan akan diambil oleh *e-book store* sebagai keluaran dari transaksi pembelian.



Gambar III-2 Skenario Transaksi Pembelian

### 3.4 Analisis BPMS Tools

#### 3.4.1 Alternative Tools

Eksekusi dan pemodelan BPEL membutuhkan suatu *tools* untuk mempermudah implementasinya. Pada subbab berikut akan dijelaskan dua *tools* yang dapat digunakan sebagai alternative yaitu Intallio dan jBoss. Kedua aplikasi ini menjadi alternatif karena bersifat *open source* sehingga dapat secara bebas digunakan dan dimodifikasi sesuai dengan kebutuhan.

### 3.4.1.1 Intalio BPMS Community Edition

Intalio BPMS Community Edition adalah BPMS yang dikembangkan oleh Intalio. Pada akhir tahun 2006 BPMS ini dirilis sebagai produk *open source*, sehingga pihak manapun dapat secara bebas menggunakannya atau memberikan usulan dalam pengembangan perangkat lunak ini. Intalio dapat didapatkan secara gratis pada <http://www.intalio.com>.

Aplikasi ini terdiri dari dua subaplikasi utama yaitu Intalio Designer dan Intalio Server. Intalio Designer berfungsi untuk merancang proses bisnis perusahaan dan Intalio Server akan mengeksekusi model yang telah dihasilkan pada Intalio Designer dan memberikan laporan terhadap hasil eksekusi proses bisnis.

Perancangan proses bisnis pada Intalio Designer dilakukan dengan *Business Process Modelling Notation* (BPMN) dan selanjutnya akan dikonversi ke BPEL untuk dieksekusi. Intalio juga mendukung perancangan *form* untuk proses bisnis yang melibatkan interaksi manusia. Intalio Designer dibangun dengan Eclipse yang merupakan IDE *opensource* dan berbasis pada *desktop*. Sedangkan Intalio Server dibangun dengan menggunakan Apache ODE (*Orchestration Designer Engine*) untuk mesin eksekusi BPEL.

### 3.4.1.2 jBoss jBPM

Aplikasi ini dikembangkan oleh jBoss (<http://www.jboss.org>). Sama halnya dengan Intalio, aplikasi ini terdiri atas server yang mengeksekusi proses yaitu jBPM Process Engine dan sebuah aplikasi untuk mendesain proses yaitu jBPM Process Designer.

Aplikasi server jBPM Process Engine digunakan sebagai alat untuk mengintegrasikan aplikasi-aplikasi dalam sebuah *enterprise* dan dapat juga menjadi mesin eksekusi proses bisnis dalam sebuah aplikasi. jBPM Process Engine mendukung eksekusi proses yang didefinisikan dengan bahasa jPDL sebagai bahasa utama. Selain jPDL juga disediakan ekstensi untuk dukungan eksekusi proses dalam bahasa BPEL. Namun pengembangan dukungan untuk bahasa ini masih dalam tahap beta.

Aplikasi jBPM Process Designer yang disediakan oleh jBPM dapat menggambarkan proses bisnis, namun terbatas dalam bahasa jPDL. Untuk eksekusi dengan BPEL, dibutuhkan pembuatan kode BPEL secara manual.

## 3.4.2 Tools Yang Dipilih

Model aplikasi dengan interaksi *service* dapat didefinisikan dengan menuliskan bahasa dalam format XML yang merepresentasikan *basic activities* dan *structured activities* sesuai dengan yang telah dibahas pada bab 2.3. Akan tetapi cara tersebut tidak efektif karena terlalu abstrak untuk

menggambarkan model interaksi yang terjadi. Oleh karena itu digunakan *tools* yang dapat mendefinisikan aktivitas dan pemodelan dalam BPEL untuk aplikasi *e-commerce*. Tools yang dipertimbangkan untuk membantu pengerjaan tugas akhir ini adalah Intalio dan JBoss. Setelah melakukan eksplorasi terhadap keduanya, akhirnya dipilih Intalio sebagai *tools* yang akan digunakan dalam pengerjaan tugas akhir ini. Hal ini disebabkan oleh beberapa faktor, antara lain:

1. Kemudahan penggunaan Intalio karena antarmuka yang sederhana dan mudah dimengerti.
2. JBoss tidak mendukung pemodelan proses BPEL secara visual sehingga pemodelan tersebut membutuhkan kakas lain. Mekanisme pemodelan BPEL secara visual ini didukung pada Intalio.

### **3.5 Analisis Studi Kasus**

#### **3.5.1 Pemilihan Studi Kasus**

Dalam pengerjaan tugas akhir ini dipilih satu aplikasi *e-commerce* untuk dijadikan studi kasus, yaitu aplikasi *e-book store*. Suatu aplikasi *e-book store* menyediakan fasilitas penjualan buku secara *online* dengan sistem pembayaran yang dilakukan melalui sistem elektronik kartu kredit dan produk yang dibeli didistribusikan ke konsumen melalui *delivery service*.

Alasan pemilihan aplikasi *e-book store* sebagai studi kasus adalah:

1. Aplikasi *e-commerce* yang sebagian besar sukses dalam dunia nyata adalah aplikasi *e-book store*. Salah satu contoh dari aplikasi yang tergolong sukses tersebut adalah *amazon.com*. Dengan demikian, referensi tentang aplikasi cukup mudah didapatkan dalam internet.
2. Aplikasi *e-book store* memiliki proses bisnis yang cukup sederhana tetapi tetap dapat menggambarkan pentingnya BPEL untuk diterapkan. Hal ini lebih memudahkan dalam pengerjaan studi kasus dalam tugas akhir ini. Proses bisnis yang ditangani dalam tugas akhir ini adalah khusus untuk transaksi pembelian.

#### **3.5.2 Deskripsi Umum Sistem**

Aplikasi *e-book store* adalah aplikasi pendukung penjualan buku secara *online*. Pemodelan aplikasi ini hanya melibatkan perusahaan yang benar-benar terlibat langsung dalam transaksi pembelian. Perusahaan yang terlibat merupakan perusahaan yang memiliki peran dalam proses bisnis mulai dari dipesannya barang sampai barang tersebut diterima oleh konsumen. Jadi perusahaan semisal pemasok barang yang tidak langsung terlibat dalam proses bisnis tidak

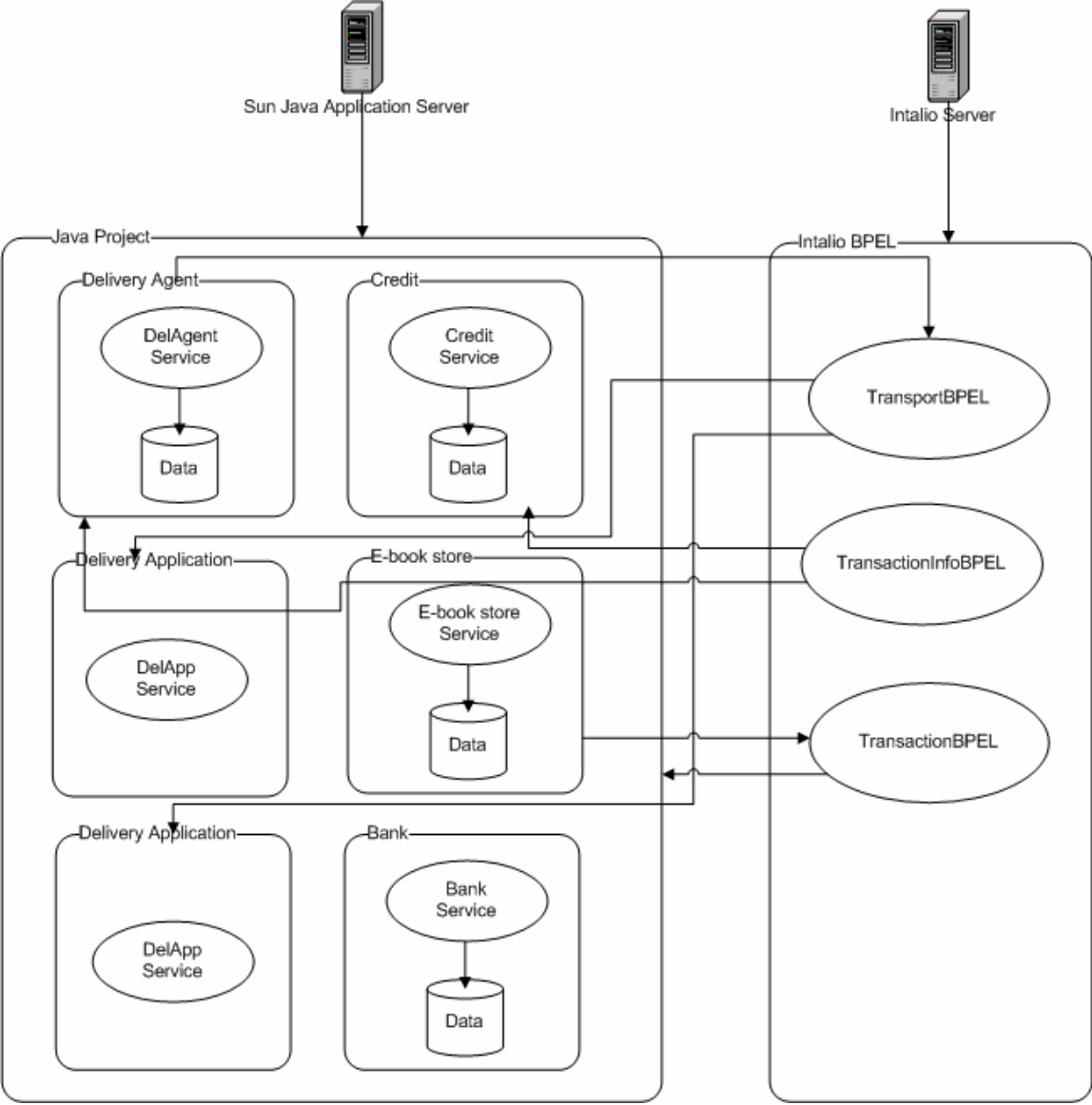
dimasukkan dalam pemodelan aplikasi. Aplikasi ini melibatkan perusahaan rekanan seperti perusahaan kredit untuk menangani proses pembayaran dalam bentuk kredit dan perusahaan *delivery* yang menangani distribusi produk kepada konsumen. Pembangunan aplikasi ini bertujuan untuk menerapkan proses bisnis transaksi pembelian buku dengan melibatkan perusahaan rekanan yang telah disebutkan di atas.

Dari penjelasan di atas, dapat disimpulkan bahwa pihak-pihak yang terlibat dalam aplikasi ini antara lain perusahaan kredit, perusahaan *delivery*, pelanggan, dan perusahaan *e-book store* itu sendiri. Pelanggan adalah pengguna *e-book store* yang melakukan transaksi pembelian produk buku dari perusahaan. Perusahaan kredit merupakan rekanan yang menangani mekanisme pembayaran biaya transaksi dari konsumen melalui sistem pembayaran elektronik dengan kartu kredit. *Delivery partner* berurusan dengan distribusi produk buku yang dibeli konsumen untuk diantarkan ke alamat konsumen.

Aplikasi *e-book store* akan diimplementasikan dengan antarmuka berbasis web. Untuk perusahaan kredit dan *delivery partner* hanya akan diimplementasikan dalam bentuk *service* yang akan dipanggil oleh aplikasi *e-book store*. Untuk mengatur orkestrasi BPEL antar komponen tersebut akan digunakan *tools* yang telah dipilih pada Bab 3.2.2 yaitu Intalio. Gambaran umum sistem dapat dilihat pada Gambar III-3.

Pengguna pada aplikasi ini akan berinteraksi dengan aplikasi melalui tampilan *web* yang ada pada *web browser*. Interaksi antara pengguna dengan aplikasi akan menghasilkan pemanggilan fungsi BPEL yang telah didefinisikan. Fungsi tersebut akan menjalankan proses bisnis dengan melakukan pemanggilan *service* yang ada dan dilakukan orkestrasi terhadap proses bisnis yang dirancang.

Pada tugas akhir ini, implementasi studi kasus hanya dibatasi pada penanganan fungsi pengelolaan transaksi pembelian buku oleh pelanggan mengingat cukup besar dan rumitnya sistem secara keseluruhan. Dalam sistem ini terdapat fungsi lain selain fungsi pengelolaan transaksi pembelian seperti fungsi manajemen *inventaris* ataupun pengelolaan karyawan. Akan tetapi fungsi-fungsi tersebut dapat ditangani dengan pemrograman biasa dan tidak dapat memunculkan pentingnya konsep proses bisnis dalam perusahaan. Oleh karena itu, tidak dapat dilihat pentingnya fungsionalitas BPEL sebagai kajian dalam Tugas Akhir ini. Sedangkan dalam fungsi pengelolaan transaksi pembelian terlihat jelas bagaimana interaksi antar *service* dalam membentuk proses bisnis. Dengan adanya gambaran interaksi tersebut maka kajian tentang fungsionalitas BPEL dapat dilihat dengan jelas termasuk juga untuk mekanisme penanganan konsep ACID pada eksekusi proses bisnis.



Gambar III-3. Gambaran Umum Sistem

### 3.5.3 Metodologi Pembangunan Perangkat Lunak

Pada subbab 1.5 telah dijelaskan bahwa perangkat lunak akan dibuat dengan menggunakan dua tahap. Tahap pertama adalah pembangunan perangkat lunak dari sisi *client e-book store* dan tahap kedua adalah pembangunan perangkat lunak dari sisi *service* perusahaan rekanan. Metode yang digunakan untuk membangun aplikasi *client* adalah *unified process*. Ada empat fase dengan kegiatan sesuai dengan kebutuhan masing-masing fase. Tahap kedua merupakan aktivitas tambahan pada *unified process* untuk mendukung pembentukan perangkat lunak dalam studi kasus ini yang berbasis SOA. Aktivitas tambahan berupa pendefinisian *service* perlu ditambahkan

karena perangkat lunak tidak hanya dipandang sebagai berdasarkan sekumpulan objek saja tapi juga sekumpulan *service* [ERL05].

Beberapa aktivitas tambahan yang perlu dilakukan dalam pembangunan perangkat lunak ini antara lain [ERL05] :

1. Identifikasi kandidat *service* yang akan digunakan. Aktivitas ini dilakukan pada fase *Elaboration* dari RUP pada saat analisis perangkat lunak.
2. Perancangan *service* yang telah didefinisikan sebelumnya. Aktivitas ini dilakukan pada fase *Elaboration* pada tahap perancangan perangkat lunak.
3. Aktivitas terakhir adalah revisi dari perancangan *service* yang telah ada. Aktivitas ini dilakukan pada fase *Construction* pada aktivitas perancangan perangkat lunak.

Identifikasi kandidat *service* dapat dilakukan dengan menggunakan *use case*. Dari skenario *use case* dapat diperkirakan jenis *service* yang akan diimplementasikan dalam perangkat lunak. Langkah selanjutnya adalah pemodelan *business process* berdasarkan *service* yang telah diidentifikasi dan dari pemodelan tersebut dapat didefinisikan operasi yang dilakukan oleh *service* yang bersangkutan.

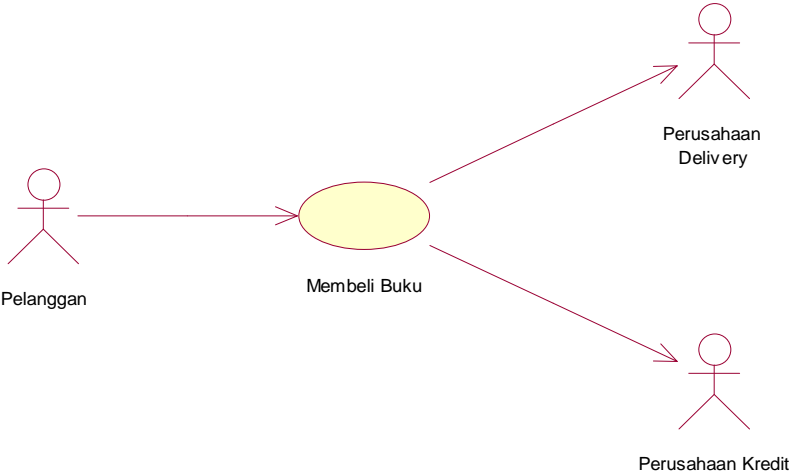
### **3.5.4 Pemodelan Perangkat Lunak**

Pemodelan perangkat lunak akan lebih memperjelas spesifikasi perangkat lunak yang akan dibangun sesuai dengan hasil analisis pada bab sebelumnya. Pemodelan ini akan menghasilkan beberapa diagram antara lain *use case diagram* dan *sequence diagram*, model kandidat *service* yang digunakan, dan identifikasi paket dan kelas.

#### **3.5.4.1 Pemodelan Fungsional**

Pemodelan ini akan menghasilkan diagram *use case* dan setiap *use case* akan menghasilkan beberapa skenario sesuai dengan proses bisnis yang terjadi. Untuk skenario *use case* yang lengkap dapat dilihat pada Lampiran A. Diagram *use case* yang dihasilkan akan menggambarkan bagaimana sistem secara keseluruhan jika dilihat dari lima aplikasi yang terlibat. Jadi akan ada lima diagram *use case* yaitu satu diagram untuk aplikasi *e-book store* itu sendiri dan empat diagram untuk perusahaan rekanan yaitu perusahaan *delivery*, perusahaan kredit, bank, dan perusahaan transportasi.

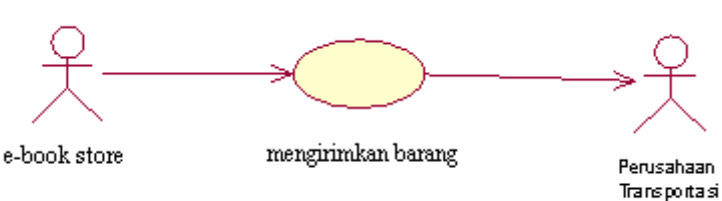
Diagram *use case* untuk aplikasi *e-book store* dapat dilihat pada Gambar III-3.



**Gambar III-4** Gambar Use Case Aplikasi E-book Store

Ada tiga aktor yang akan berhubungan dengan aplikasi ini yaitu pelanggan, perusahaan *delivery*, dan perusahaan kredit. Pelanggan adalah seseorang yang melakukan transaksi pembelian buku dengan *e-book store*. Dalam hal ini pelanggan adalah pengguna yang telah terdaftar sebagai *member*. Sedangkan *e-book store* sendiri mempunyai kewenangan untuk mengelola transaksi pembelian yang diajukan oleh pelanggan. Pengelolaan transaksi pembelian ini akan melibatkan beberapa *service* yang berhubungan dengan proses bisnis pembelian buku yaitu *service* dari perusahaan *delivery* dan perusahaan kredit.

Dari sisi perusahaan *delivery*, diagram *use case* yang terkait dengan proses bisnis pembelian buku dapat dilihat pada Gambar III-4.

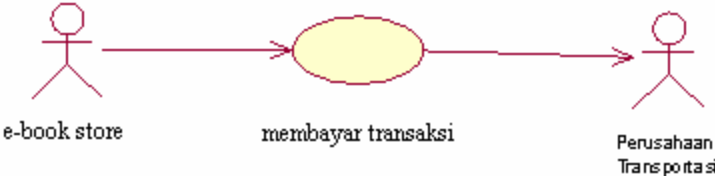


**Gambar III-5** Gambar Use Case Perusahaan Delivery

Ada dua aktor yang menjalankan fungsionalitas dalam perusahaan *delivery* ini yaitu perusahaan *delivery* dan perusahaan transportasi. Perusahaan *delivery* akan menerima permintaan dari aplikasi *e-book store* untuk menangani pengiriman barang dari konsumen. Selanjutnya

perusahaan ini akan meneruskan proses dengan mengirimkan pesan ke perusahaan transportasi yang bersangkutan.

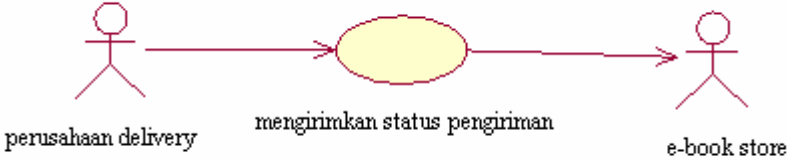
Jika dilihat dari sisi perusahaan kredit, diagram *use case* yang dihasilkan dapat dilihat pada Gambar III-5.



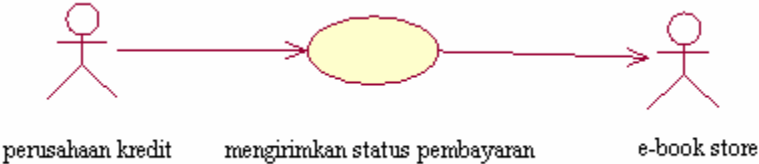
**Gambar III-6 Gambar Use Case Perusahaan Kredit**

Ada dua aktor yang menjalankan fungsionalitas dalam perusahaan kredit ini yaitu perusahaan kredit dan bank. Perusahaan kredit akan menerima permintaan dari aplikasi *e-book store* untuk menangani pembayaran dari konsumen. Selanjutnya perusahaan ini akan meneruskan proses dengan mengirimkan pesan ke bank.

Ada dua entitas lain yang tidak berkaitan langsung dengan aplikasi *e-book store* yaitu bank dan perusahaan transportasi. Meskipun tidak berkaitan langsung dengan aplikasi, dua perusahaan ini tetap memiliki peran dalam menjalankan proses bisnis. Oleh karena itu diagram *use case* dari dua entitas ini tetap harus dijelaskan. Diagram *use case* untuk dua entitas ini dapat dilihat pada gambar III-6.



**Gambar III-7 Gambar Use Case Perusahaan Transportasi**



**Gambar III-8 Gambar Use Case Bank**

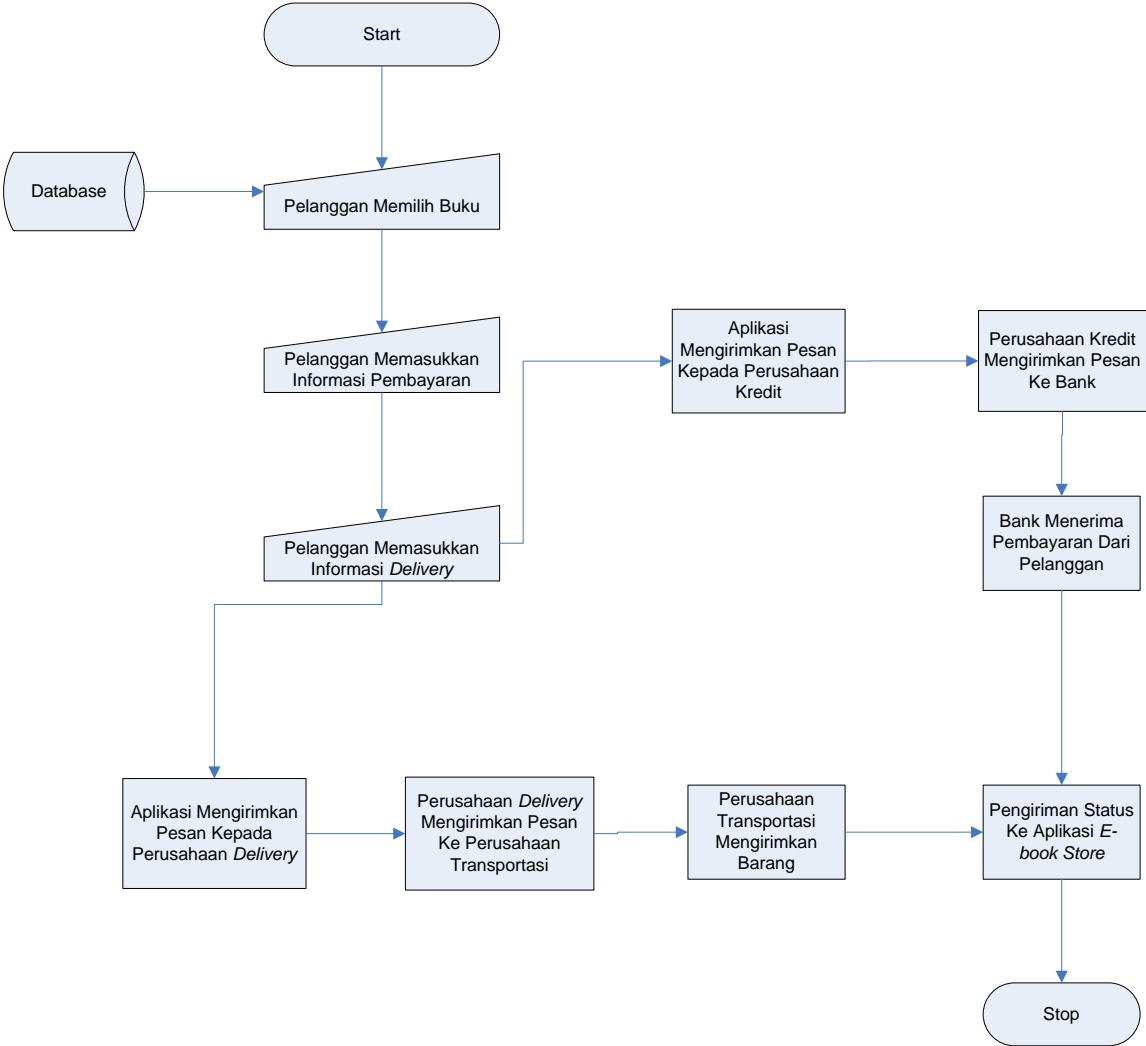
Bank dan perusahaan transportasi merupakan *node* akhir yang menjadi *endpoint* bagi proses bisnis. Bank berperan dalam penerimaan pembayaran dari rekening konsumen setelah menerima informasi dari perusahaan kredit. Perusahaan transportasi akan melakukan pengiriman barang ke alamat konsumen yang didapat dari pesan yang dikirim oleh perusahaan *delivery*. Perusahaan transportasi ini dapat berupa perusahaan yang bergerak dalam transportasi darat, air, atau udara. Dalam tugas akhir ini diasumsikan perusahaan transportasi adalah salah satu dari ketiga jenis di atas, jadi tidak akan diimplementasikan ketiga jenisnya.

#### **3.5.4.2 Pemodelan Proses Bisnis dan Identifikasi Service**

Pemodelan *service* merupakan aktivitas tambahan pada UP agar mendukung aplikasi SOA sesuai dengan yang telah dijelaskan pada bab 3.3.3. Pada tahap ini akan dihasilkan kandidat *service* yang muncul sebagai hasil dari tahap analisis dan berpotensi untuk diimplementasikan jika telah masuk pada tahap perancangan perangkat lunak.

Pemodelan *service* sangatlah bergantung pada proses bisnis yang dijalankan. Dalam studi kasus *e-book store*, proses bisnis yang ditangani adalah bagaimana menangani transaksi pembelian buku oleh pelanggan. Gambaran *workflow* dari proses bisnis ini dapat dilihat pada gambar III-7. Proses bisnis tersebut dapat mengidentifikasi aktivitas yang ada di dalamnya dan dari aktivitas tersebut pendefinisian kandidat operasi dari setiap *service* dapat dilakukan.

Identifikasi *service* dilakukan dengan cara analisis terhadap entitas-entitas yang berhubungan dengan sistem dan bagaimana keterhubungannya. Aktivitas-aktivitas yang dihasilkan dari analisis terhadap proses bisnis lalu dikelompokkan berdasarkan entitas hasil analisis di atas. Selanjutnya akan dihasilkan kandidat *service* dan operasinya. Pada *e-book store* terdapat lima *service* secara keseluruhan yang terdiri dari empat *service* sesuai dengan entitas hasil analisis di atas dan satu *service* lain untuk melakukan orkestrasi yaitu *service* Transaksi Pembelian.



Gambar III-9 Proses Bisnis Pembelian Buku

Adapun fungsi-fungsi umum dari *service* yang ada adalah sebagai berikut :

1. *Service* Perusahaan *Delivery* berfungsi untuk menangani proses pengiriman barang pada pelanggan. *Service* ini juga akan melakukan pemanggilan *service* lain dari perusahaan transportasi seperti yang telah dijelaskan pada 3.2.1.
2. *Service* Perusahaan Kredit berfungsi untuk menangani proses pembayaran dari pelanggan melalui kartu kredit. *Service* ini juga akan melakukan pemanggilan *service* lain dari bank sesuai dengan kartu kredit yang dimiliki oleh pelanggan seperti yang telah dijelaskan pada 3.2.1.
3. *Service* Bank berfungsi untuk menerima pembayaran transaksi sesuai dengan rekening pelanggan pada Bank. *Service* ini akan berjalan setelah menerima pesan dari perusahaan kredit.

4. *Service* Perusahaan Transportasi berfungsi untuk mengirimkan barang ke alamat pelanggan. *Service* ini akan berjalan setelah menerima pesan dari perusahaan *delivery*.
5. *Service* Transaksi Pembelian berfungsi untuk menangani jika pelanggan melakukan transaksi pembelian dengan *e-book store*. *Service* ini akan melakukan pemanggilan *service* lain dan melakukan orkestrasi antar *service* untuk merepresentasikan proses bisnis yang dijalankan.

*Service-service* yang dimiliki oleh sistem dan operasinya dapat dilihat pada Tabel III-1. Untuk penjelasan detail tentang identifikasi *service* dan operasi yang dimiliki dapat dilihat pada Lampiran A.

**Tabel III-1. Kandidat *Service* dan Operasi Dalam *e-book store***

<i>No</i>	<i>Service</i>	Operasi
S-01	Transaksi Pembelian	<ul style="list-style-type: none"> <li>• Mengambil informasi pengguna</li> <li>• Melakukan transaksi pembelian</li> <li>• Mengirimkan status proses</li> </ul>
S-02	Perusahaan Kredit	<ul style="list-style-type: none"> <li>• Mengelola pembayaran melalui kartu kredit</li> <li>• Mengirimkan informasi kepada bank</li> </ul>
S-03	Perusahaan <i>Delivery</i>	<ul style="list-style-type: none"> <li>• Mengelola pengiriman barang</li> <li>• Mengirimkan informasi kepada perusahaan transportasi.</li> </ul>
S-04	Perusahaan Transportasi	<ul style="list-style-type: none"> <li>• Mengirimkan barang</li> </ul>
S-05	Bank	<ul style="list-style-type: none"> <li>• Menerima Pembayaran</li> </ul>