

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Menurut Soendoro Herlambang dan Haryanto Tanuwijaya (2005: 116), Sistem dapat dilakukan dengan 2 pendekatan komponen yaitu dengan pendekatan prosedur dan pendekatan komponen. Dengan pendekatan prosedur, sistem didefinisikan sebagai kumpulan dari beberapa prosedur yang mempunyai tujuan tertentu, sebagai contoh sistem informasi penjualan barang. Dengan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu, contoh dari definisi ini adalah sistem komputer.

Menurut Tata Sutabri (2005: 2), Sistem sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu.

Menurut Jogiyanto (2005: 1), Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

Menurut Jogiyanto (2009: 34), Sistem (*system*) dapat didefinisikan dengan pendekatan prosedur dan dengan pendekatan komponen. Dengan pendekatan prosedur, sistem dapat didefinisikan sebagai kumpulan dan prosedur-prosedur tertentu yang mempunyai tujuan tertentu. Dengan pendekatan komponen, sistem dapat didefinisikan sebagai kumpulan komponen yang saling berhubungan satu

dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu.

Menurut Bonnie Soeherman dan Marion Pinontoan (2008: 3), Sistem dapat diartikan sebagai serangkaian komponen-komponen yang saling berinteraksi dan bekerja sama untuk mencapai tujuan tertentu.

Dari definisi sistem di atas, dapat disimpulkan sistem adalah suatu jaringan kerja yang saling memiliki keterkaitan antar bagian dan prosedur-prosedur yang ada.

II.2. Informasi

Menurut Soendoro Herlambang dan Haryanto Tanuwijaya (2005: 122), Pengertian mengenai data dan informasi sering rancu. Ada yang mengatakan bahwa data itu sama dengan informasi, atau data itu berisi informasi, data sebenarnya merupakan fakta-fakta atau kejadian yang dapat berupa angka-angka atau kode-kode tertentu. Data berupa angka atau wujud yang lain belum mempunyai arti bagi penggunanya. Sedangkan informasi adalah data yang telah diolah dan mempunyai arti bagi penggunanya.

Menurut Jogiyanto (2005: 8), Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Menurut Jogiyanto (2009: 36), Informasi (*information*) adalah data yang diolah menjadi bentuk yang berguna bagi para pemakainya.

Menurut Bonnie Soeherman dan Marion Pinontoan (2008: 4), Informasi merupakan hasil pemrosesan data (fakta) menjadi sesuatu yang bermakna dan bernilai untuk pengambilan keputusan.

Menurut Edhy Sutanta (2011: 13), Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang. Informasi diperlukan oleh pemakai (manajemen) pada seluruh level manajemen dalam seluruh fungsi *organisatoris*. Informasi tersebut dapat mempunyai beberapa fungsi, antara lain :

1. Menambah pengetahuan

Adanya informasi akan menambah pengetahuan bagi penerimanya yang dapat digunakan sebagai bahan pertimbangan yang mendukung proses pengambilan keputusan.

2. Mengurangi ketidakpastian

Adanya informasi akan mengurangi ketidakpastian karena apa yang akan terjadi dapat diantisipasi dengan baik sehingga menghindari keraguan pada saat pengambilan keputusan.

3. Mengurangi risiko kegagalan

Adanya informasi akan mengurangi risiko kegagalan karena apa yang akan terjadi dapat diantisipasi dengan baik sehingga kemungkinan terjadinya kegagalan akan dapat dikurangi dengan pengambilan keputusan yang tepat.

4. Mengurangi keanekaragaman yang tidak diperlukan

Adanya informasi akan mengurangi keanekaragaman yang tidak diperlukan karena keputusan yang diambil lebih terarah.

II.3. Sistem Informasi

Menurut Jogiyanto (2005: 11), Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Menurut Bonnie Soeherman dan Marion Pinontoan (2008: 5), Sistem informasi merupakan serangkaian komponen berupa manusia, prosedur, data, dan teknologi (seperti komputer) yang digunakan untuk melakukan sebuah proses untuk menghasilkan informasi yang bernilai untuk pengambilan keputusan.

II.4. Basis Data

Menurut Janner Simarmata (2007: 4-5), Basis data adalah koleksi data yang bisa mencari secara menyeluruh dan secara sistematis memelihara retrieve informasi.

Sistem basis data terus dikembangkan oleh para ahli agar dapat diperoleh cara pengorganisasian data yang efisien dan efektif. Hal ini diperlukan karena sekarang ini berbagai bidang usaha telah menjadikannya sebagai tumpuan manajemen informasi perusahaannya. Adapun penerapan sistem *database* ini antara lain untuk membangun sebuah sistem informasi seperti persediaan, pegawai, akuntansi, pemasaran, produksi, reservasi, layanan pelanggan yang digunakan dalam perusahaan retail, perbankan, perhotelan, pariwisata, rumah sakit, institusi pendidikan dan lain sebagainya. Data dalam *database* disusun berdasarkan sistem hierarki yang unik yaitu :

1. *Database*, merupakan kumpulan *file* yang saling terkait satu sama lain. Kumpulan *file* yang tidak saling terkait antara satu dengan lainnya tidak dapat disebut basis data.
2. *File* yaitu kumpulan dari *record* yang saling terkait dan memiliki *format field* yang sama dan sejenis.
3. *Record* yaitu kumpulan *field* yang menggambarkan suatu unit data individu tertentu.
4. *Field* yaitu atribut dari *record* yang menunjukkan suatu *item* dari data seperti nama, alamat dan lain sebagainya.
5. *Byte* yaitu atribut dari *field* yang berupa huruf yang membentuk nilai dari sebuah *field*. Huruf tersebut dapat berupa numerik maupun abjad atau karakter khusus.

Dalam pembangunan *database*, analisis sistem harus dapat menentukan dalam model arsitektur mana *database* itu akan diletakkan. Dinilai dari penempatannya, arsitektur *database* dapat dikategorikan dalam tiga bagian yaitu :

1. Sistem *database* tunggal

Pada arsitektur ini, *database* dan aplikasinya diletakkan pada komputer yang sama yang tidak berada dalam lingkungan jaringan, sehingga *database* itu hanya dapat diakses oleh aplikasi tunggal. Sistem ini biasanya digunakan oleh perusahaan kecil.

2. Sistem *database* terpusat

Pada arsitektur ini, lokasi *database* secara fisik berada dalam komputer pusat dalam suatu lingkungan jaringan. Meskipun pemasukan dan akses data dapat

dilakukan dari berbagai terminal yang terhubung dengan komputer tersebut, namun proses pengolahan data hanya berlangsung di komputer pusat. Dengan sistem ini, komputer pusat menjadi titik kritis dari proses pengolahan *database*. Bila komputer pusat terganggu, maka secara keseluruhan sistem informasi akan terganggu.

3. Sistem *database* terdistribusi

Pada arsitektur ini salinan *database*, baik sebagian maupun secara keseluruhan terdistribusi di beberapa lokasi. Pada model ini, titik kritis pada sistem terpusat dapat dihindari. Namun pada sistem ini, tantangan terbesar yang dihadapi adalah proses pengintegrasian untuk menjaga konsistensi data yang tersebar di beberapa lokasi.

II.5. Normalisasi

Menurut Janner Simarmata (2007: 77-84), Normalisasi merupakan suatu proses pengelompokan elemen data menjadi tabel-tabel yang menunjukkan entitas dan normalisasi bisa dipakai oleh para perancang *database* untuk melakukan verifikasi terhadap tabel-tabel yang telah dibuat sehingga tidak menimbulkan suatu permasalahan saat data diperbaharui maupun saat data dihapus. Aturan-aturan normalisasi dinyatakan dalam istilah bentuk normal :

1. Bentuk Normal Kesatu (1 NF).

Skema tabel disebut dalam bentuk normal pertama jika nilai atribut tidak terpisahkan.

2. Bentuk Normal Kedua (2 NF)

Skema tabel disebut bentuk normal kedua jika semua atribut informasi

(atribut yang tidak memiliki kunci manapun) adalah atribut dari entitas lain di dalam skema tabel dan bukan dari kelas entitas lainnya.

3. Bentuk Normal Ketiga (3 NF)

Untuk bentuk normal ketiga, maka relasi haruslah dalam bentuk normal kedua dan semua atribut bukan primer dan tidak mempunyai hubungan yang transitif, dengan kata lain setiap atribut tergantung pada *subset* yang sesuai dengan kunci.

4. *Boyce Codd Normal Form (BCNF)*

BCNF merupakan bentuk normal sebagai perbaikan terhadap 3NF. Suatu relasi BCNF selalu memenuhi 3NF, tetapi tidak sebaliknya. Suatu relasi yang memenuhi 3NF belum tentu memenuhi BCNF. BCNF adalah perbaikan dari 3NF karena bentuk normal ketiga pun masih memungkinkan mengandung anomali sehingga masih perlu dinormalisasikan lebih lanjut. Skema tabel disebut bentuk ketiga apabila semua atribut harus bergantung pada *superkey*.

II.6. *Unified Modeling Language*

Menurut Munawar (2005: 17-22), UML (*Unified Modelling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembangan sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain.

UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan

oleh Booch, *Object Modelling Technique* (OMT) dan *Object Oriented Software Engineering* (OOSE). Dengan UML, metode Booch, OMT dan OOSE digabungkan dengan membuang elemen-elemen yang tidak praktis ditambah dengan elemen-elemen dari metode lain yang lebih efektif dan elemen-elemen baru yang belum ada pada metode terdahulu sehingga UML lebih ekspresif dan seragam daripada metode lainnya.

UML mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Karena ini merupakan sebuah bahasa, UML mempunyai sejumlah aturan untuk menggabungkan/mengkombinasikan elemen-elemen tersebut. Untuk lebih jelasnya akan lebih baik bila kita melihat diagram apa saja yang ada di UML, yaitu:

1. *Use Case Diagram*
2. *State Machine Diagram*
3. *Sequence Diagram*
4. *Collaboration Diagram*
5. *Activity Diagram*
6. *Component Diagram*
7. *Deployment Diagram*
8. *Package Diagram*
9. *Object Diagram*
10. *Communication Diagram*
11. *Composite Structure Diagram*
12. *Interaction Overview Diagram*

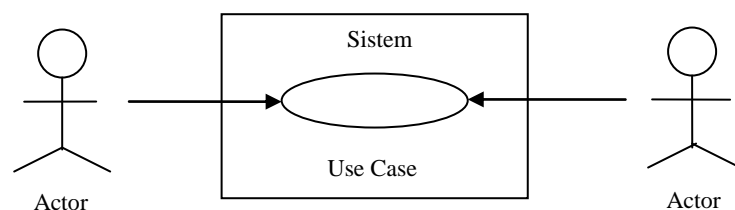
13. Timing Diagram

Berikut ini adalah keterangan dari masing-masing diagram pada *Unified Modelling Language*:

1. Use Case Diagram

Menurut Munawar (2005: 63), *Use Case* adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. *Use Case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut *scenario*. Setiap *scenario* mendeskripsikan urutan kejadian. Setiap urutan diinisialisasi oleh orang, sistem yang lain, perangkat keras, atau urutan waktu. Dengan demikian secara singkat bisa dikatakan *use case* adalah serangkaian *scenario* yang digabungkan bersama-sama oleh tujuan umum pengguna.

Use Case Diagram menunjukkan 3 aspek dari sistem yaitu actor, *use case*, dan sistem/sub sistem boundary. Actor mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan *use case*. Gambar II.1 mengilustrasikan actor, *use case* dan boundary.

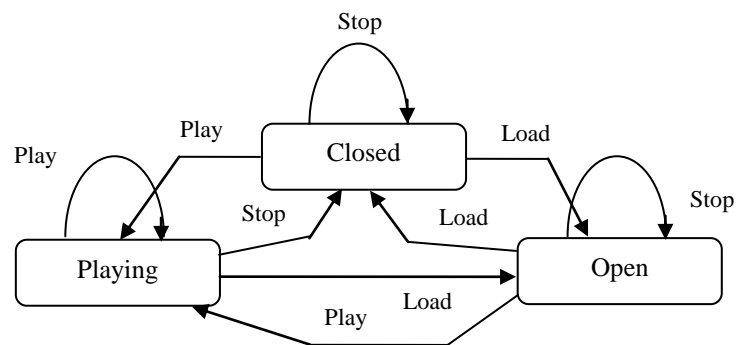


Gambar II.1 : Contoh Use Case Model

Sumber : Munawar (2005 : 64)

2. State Machine Diagram

Menurut Munawar (2005: 73), *Interaction diagram* dan *state chart* menampilkan dua pandangan yang saling melengkapi tentang perilaku dinamis sebuah sistem. *Interaction diagram* menunjukkan pesan-pesan yang dilewatkan diantara obyek-obyek di dalam sistem selama periode waktu yang pendek. Sedangkan *state chart diagram* menelusuri individu-individu obyek melalui keseluruhan daur hidupnya, menspesifikasikan semua urutan yang mungkin dari pesan-pesan yang akan diterima obyek tersebut, bersama-sama dengan tanggapan atas pesan-pesan tersebut. Sebagai contoh, jika sebuah CD player dalam keadaan terbuka, penekanan tombol *load* akan menyebabkan *drawer* CD player berpindah ke *state close*. *State chart* lengkap untuk CD player ini bisa dilihat pada gambar II.2 berikut :



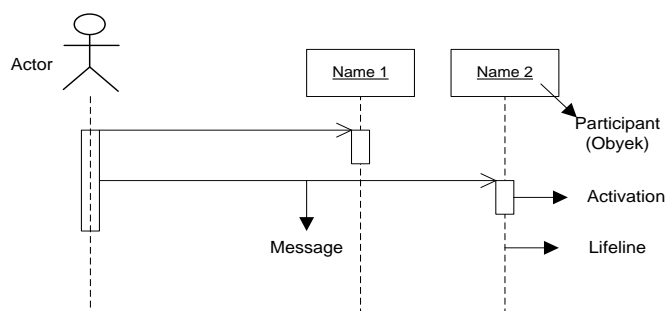
Gambar II.2 : Contoh State Machine untuk CD Player

Sumber : Munawar (2005 : 75)

3. Sequence Diagram

Menurut Munawar (2005: 87), *Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah scenario. Diagram ini menunjukkan sejumlah contoh obyek dan *message* (pesan) yang diletakkan diantara obyek-obyek ini di dalam *use case*.

Komponen utama *sequence diagram* terdiri atas obyek yang dituliskan dengan kotak segiempat. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan *progress vertical*. Gambar II.3 menunjukkan esensi symbol dari *sequence diagram* dan symbol kerjanya secara bersama-sama.

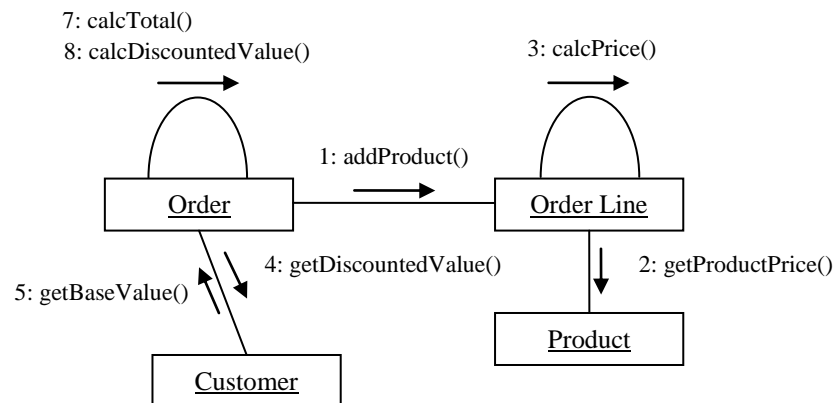


Gambar II.3 : Simbol-simbol yang ada pada *Sequence Diagram*

Sumber : Munawar (2005 : 89)

4. *Collaboration Diagram*

Menurut Munawar (2005: 101), *Collaboration Diagram* adalah perluasan dari *obyek diagram* (*Obyek diagram* menunjukkan obyek-obyek dan hubungannya satu dengan yang lain). *Collaboration diagram* menunjukkan *message-message* obyek yang dikirimkan satu sama lain. Untuk menunjukkan sebuah pesan, buatlah tanda panah didekat garis asosiasi diantara 2 obyek. Arah panah menunjukkan obyek yang menerima pesan. Label didekat panah menunjukkan pesannya apa. Tipikal *message* meminta kepada obyek yang menerimanya untuk menjalankan salah satu operationnya. Sepasang tanda kurung digunakan untuk mengakhiri *message*. Jika ada parameter bisa diletakkan diantara tanda kurung. Gambar II.4 menunjukkan *collaboration diagram* untuk kasus order.

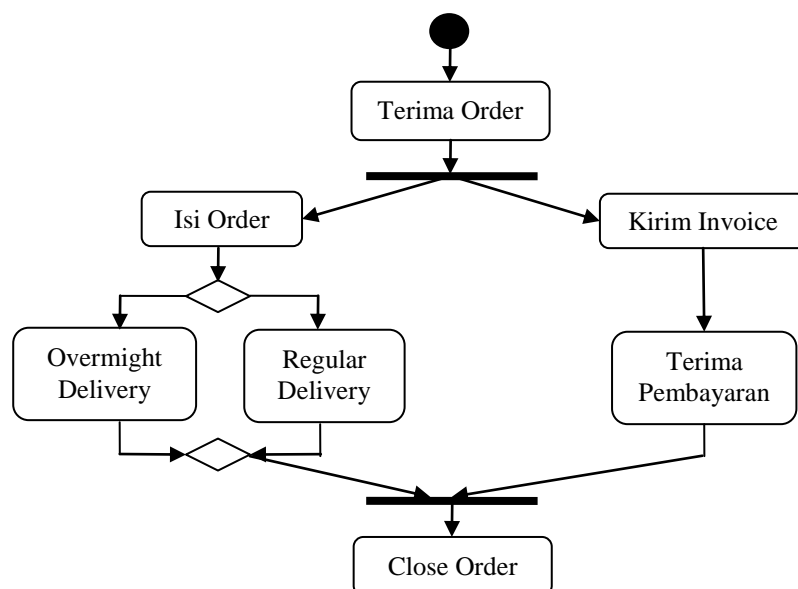


Gambar II.4 : Collaboration Diagram untuk Order

Sumber : Munawar (2005 : 102)

5. Activity Diagram

Menurut Munawar (2005: 109), *Activity Diagram* adalah teknik untuk mendiskripsikan logika procedural, proses bisnis, dan aliran kerja dalam banyak kasus. *Activity Diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Contoh sederhana *activity diagram* bisa dilihat pada gambar II.5 berikut :

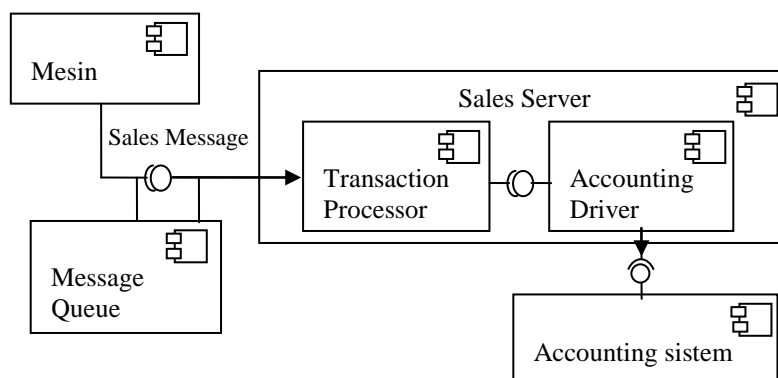


Gambar II.5 : Contoh Activity Diagram Sederhana

Sumber : Munawar (2005 : 111)

6. Component Diagram

Menurut Munawar (2005: 124), *Component Diagram* merepresentasikan dunia riil item yaitu *component software*. *Component Software* menetap di computer bukan dibenak para analis. *Component* bisa diakses melalui interfacenya yaitu koleksi operasi-operasi. Relasi antara *component* dan interfacenya disebut *realization*. Suatu *component* bisa mengakses *service-service* yang ada di *component* lain dengan cara *import interface*. Sedangkan *component* yang menyediakan *service* menggunakan *export interface*. Gambar II.6 menunjukkan contoh *component diagram*.



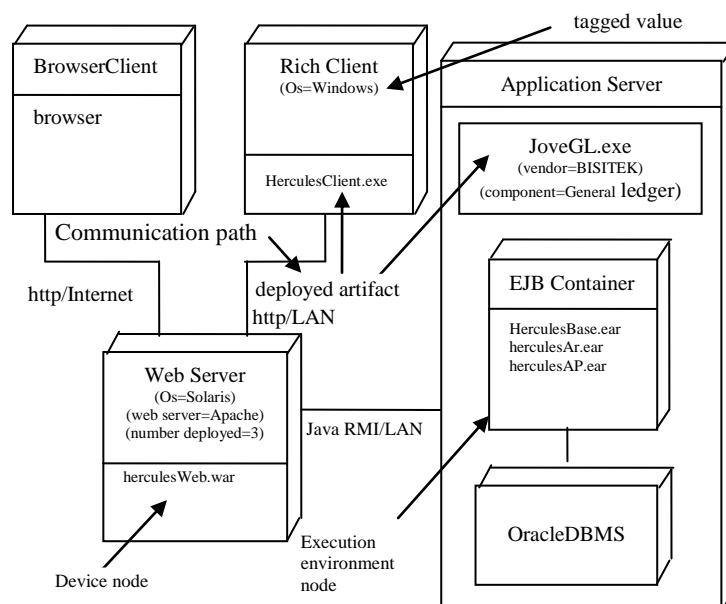
Gambar II.6 : Contoh Component Diagram

Sumber : Munawar (2005 : 123)

7. Deployment Diagram

Menurut Munawar (2005: 125), *Deployment Diagram* menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian software yang berjalan pada bagian-bagian hardware. Bagian utama hardware/perangkat keras adalah node, yaitu nama umum untuk semua jenis sumber komputasi. Ada 2 tipe node yang mungkin. *Processor* adalah node yang bisa mengeksekusi sebuah component, sedangkan device tidak. *Device* adalah perangkat keras (seperti

printer atau monitor) tipikalnya menjadi interface dengan dunia luar. Contoh penerapan *deployment diagram* bisa dilihat pada gambar II.7 berikut :



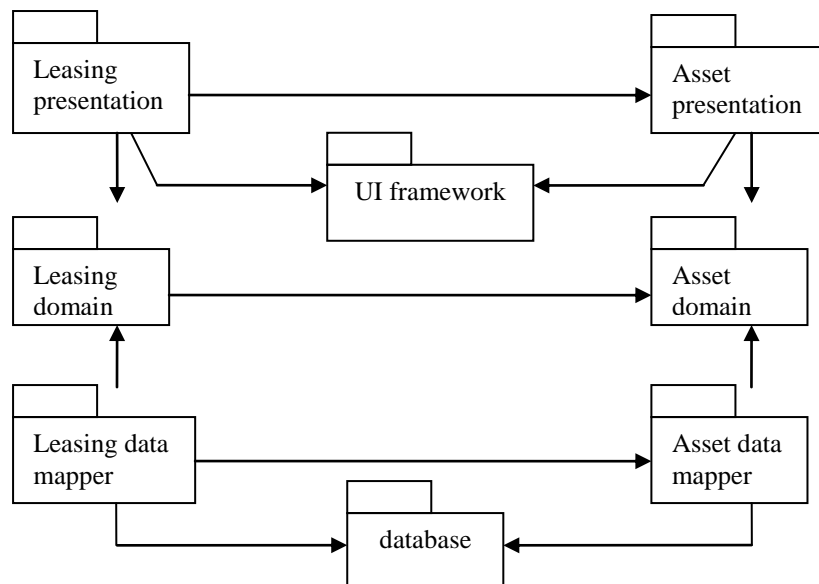
Gambar II.7 : Contoh Penerapan *Deployment Diagram*

Sumber : Munawar (2005 : 127)

8. Package Diagram

Menurut Munawar (2005: 134), *Package Diagram* adalah pengelompokan konstruksi ke level yang lebih tinggi. Sebuah package bisa menjadi anggota package yang lain. Bila dibuat hierarki, package yang paling tinggi akan mengandung package-package yang lain. Demikian seterusnya hingga yang paling bawah adalah class.

Manfaat utama menggunakan package adalah penerapannya pada sistem skala besar untuk mendapatkan gambaran saling ketergantungan diantara component-component utama pada sistem. Gambar II.8 menunjukkan contoh *package diagram* untuk aplikasi enterprise yang terstruktur cukup baik dan mempunyai aliran yang jelas.



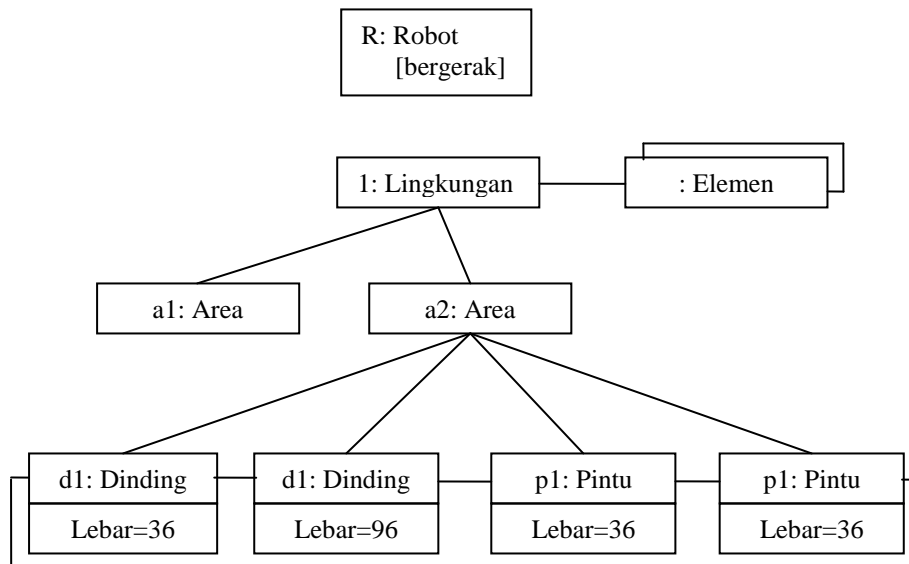
Gambar II.8 : Contoh *Package Diagram* untuk Aplikasi Enterprise

Sumber : Munawar (2005 : 132)

9. *Object Diagram*

Menurut Munawar (2005: 135), *Object Diagram* adalah gambaran obyek-obyek secara ringkas disebuah sistem pada suatu waktu. *Object Diagram* sering disebut sebagai instance diagram karena menunjukkan instance-instance dari class. *Obyek diagram* bisa digunakan untuk menunjukkan contoh konfigurasi dari obyek-obyek. Hal ini sangat berguna untuk menunjukkan hubungan yang mungkin ada diantara obyek-obyek yang sangat kompleks.

Object Diagram bisa digunakan untuk memodelkan pandangan dari rancangan atau proses yang statis dari suatu sistem. Termasuk disini adalah pemodelan sistem secara ringkas pada suatu waktu dan membuat serangkaian obyek, state, dan relasinya. Gambar II.9 menunjukkan serangkaian obyek yang digambar dari implementasi sebuah robot.

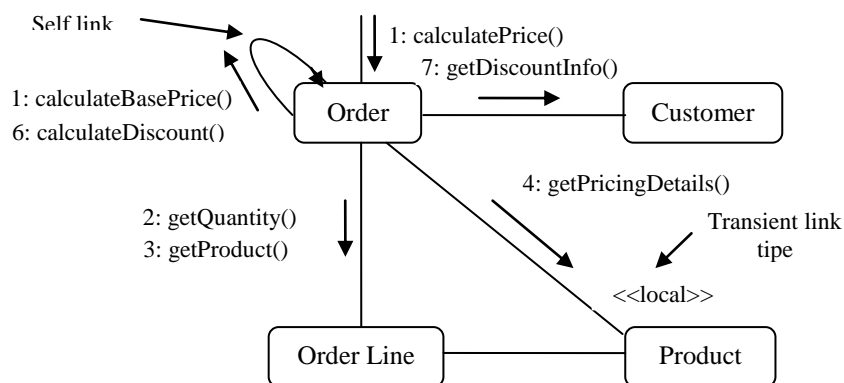


Gambar II.9 : Pemodelan Struktur Obyek

Sumber : Munawar (2005 : 137)

10. Communication Diagram

Menurut Munawar (2005: 141), *Communication Diagram* adalah sejenis dengan diagram interaksi, yang lebih menekankan pada link data diantara bermacam-macam participant pada interaksi tersebut. Pada *Communication Diagram* kita bebas menempatkan participant, boleh menggambarkan link untuk menunjukkan bagaimana participant berhubungan serta boleh juga memberikan penomoran untk menunjukkan urutan *message*.

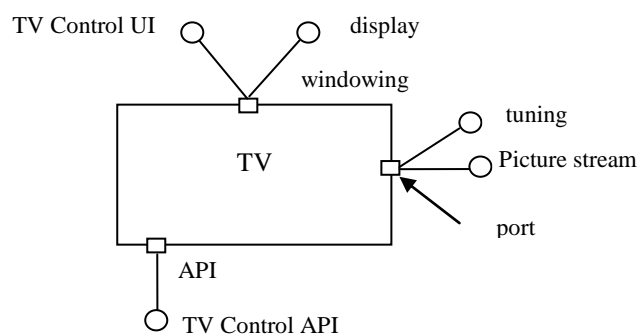


Gambar II.10 : Communication Diagram dengan Control Tersentralisasi

Sumber : Munawar (2005 : 142)

11. Composite Structure Diagram

Menurut Munawar (2005: 148), *Composite Structure Diagram* adalah diagram untuk menunjukkan dekomposisi secara hirarkis sebuah class ke sebuah struktur internal. Hal ini memungkinkan untuk memecah obyek yang kompleks menjadi bagian-bagian kecil. Perbedaan antara package dengan *composite structure* adalah package digunakan untuk pengelompokan saat kompilasi (*compiletime*), sedangkan *composite structure* digunakan untuk pengelompokan saat dijalankan (*runtime*). Dengan port bias dimungkinkan untuk mengelompokkan interface yang dibutuhkan dan disiapkan ke dalam interaksi logis yang dimiliki komponen dengan dunia luar sebagaimana tampak pada gambar II.11 berikut :



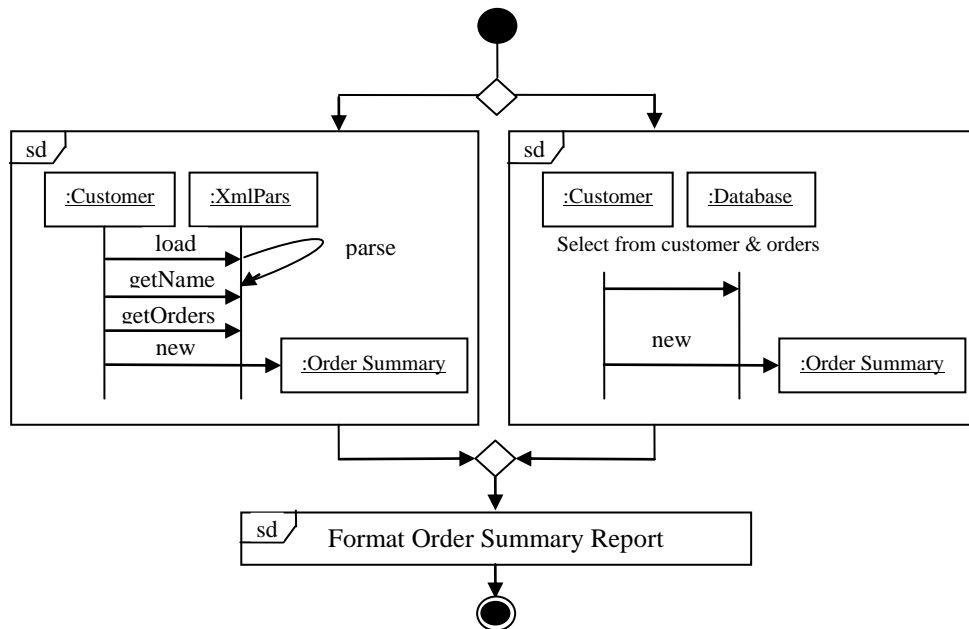
Gambar II.11 : Sebuah Komponen dengan Beberapa Port

Sumber : Munawar (2005 : 147)

12. Interaction Overview Diagram

Menurut Munawar (2005: 149), *Interaction Overview Diagram* adalah pencangkakan secara bersama antara activity diagram dengan *sequence diagram*. *Interaction Overview Diagram* bisa dianggap sebagai *activity diagram* dimana semua aktivitas diganti dengan sedikit *sequence diagram*, atau bisa juga dianggap sebagai *sequence diagram* yang dirincikan dengan notasi *activity diagram* yang

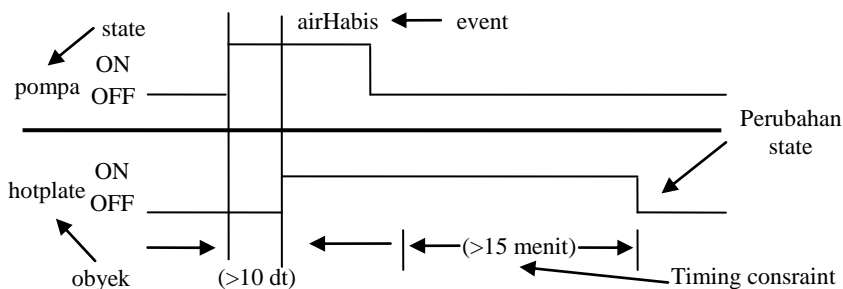
digunakan untuk menunjukkan aliran pengawasan. Dengan kata lain *interaction diagram* adalah gabungan diantara keduanya yang membuat campuran sedikit ganjil. Contoh sederhana penerapan diagram ini bisa dilihat pada gambar II.12 berikut :



Gambar II.12 : Interaction Summary Diagram
 Sumber : Munawar (2005 : 150)

13. Timing Diagram

Menurut Munawar (2005: 151), *Timing Diagram* adalah bentuk lain dari interaction diagram, dimana focus utamanya lebih ke waktu. *Timing diagram* ini bisa dipakai untuk obyek tunggal maupun sekelompok obyek.



Gambar II.13 : Timing Diagram Menunjukkan State sebagai Garis
 Sumber : Munawar (2005 : 152)

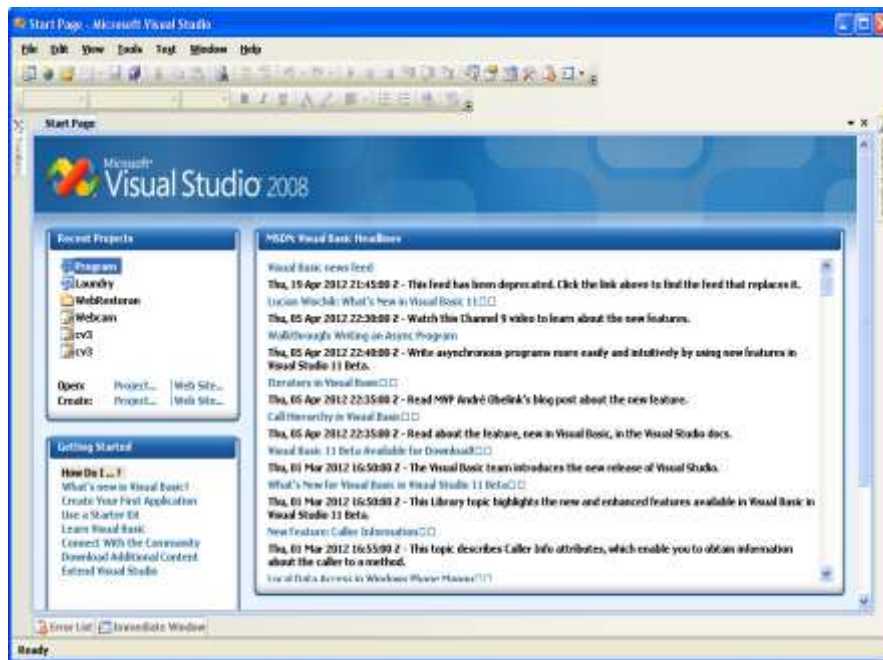
II.7. *Microsoft Visual Studio 2008*

Menurut Ketut Darmayuda (2010: 1-3), *Microsoft Visual Studio 2008* merupakan kelanjutan dari *Microsoft Visual Studio* sebelumnya yang diproduksi oleh Microsoft yang dapat membangun, menjalankan, dan meningkatkan generasi lanjut dari aplikasi terdistribusi.

Menurut Muhammad Sadeli (2009: 2), *Visual Studio 2008* merupakan suatu perangkat lunak yang dapat digunakan untuk pengembangan berbagai macam aplikasi yang memiliki berbagai macam tipe antara lain aplikasi *desktop* (*windows form, command line (console)*), aplikasi *web*, *windows mobile (pocket pc)*. *Visual studio 2008* diluncurkan microsoft pada tanggal 19 November 2007 dan memiliki lebih dari satu kompilator, *SDK (Software Development Kit)* dan dokumentasi tutorial (*MSDN Library*).

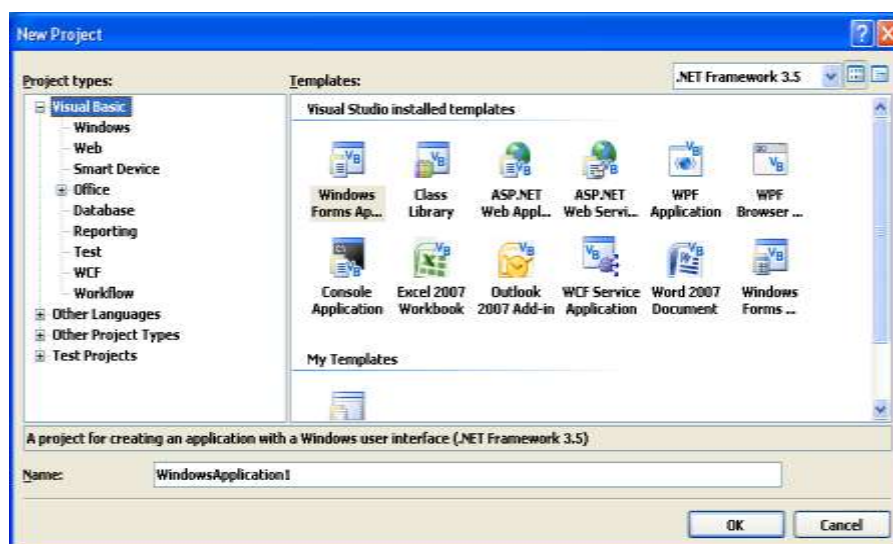
Untuk memulai program *Visual Studio 2008* dapat dilakukan dengan langkah-langkah di bawah ini:

1. Klik tombol *start* pada *windows taskbar*.
2. Klik menu *All Program > Microsoft Visual Studio 2008 > Microsoft Visual Studio 2008*.



Gambar II.14 : Tampilan Microsoft Visual Studio 2008

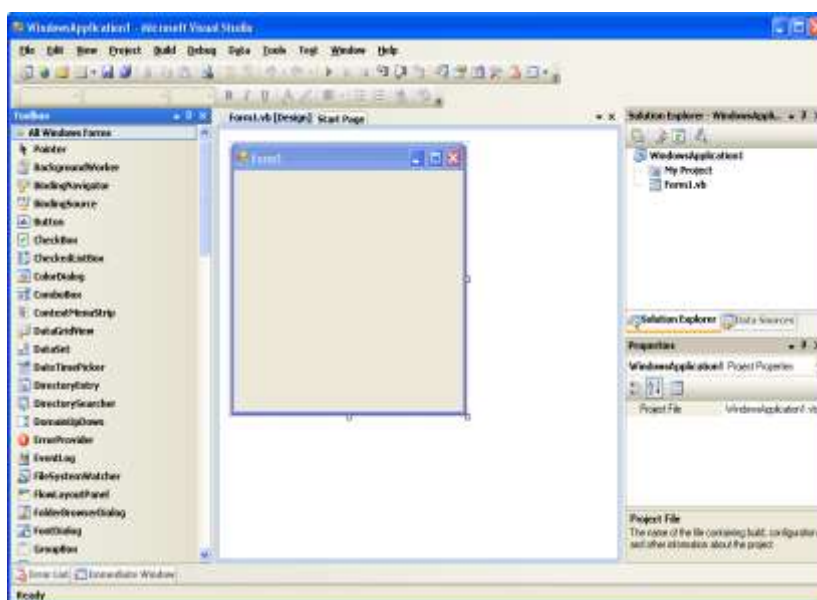
Untuk membuat project baru klik menu *Creat Project* atau Anda klik menu file > *New Project* seperti pada gambar berikut ini :



Gambar II.15 : Cara Membuka Project Baru Di Visual Basic 2008

Pada VB 2008 terdapat banyak tipe *project* yang bisa dibuat antara lain *Windows* (*Windows Form, Command line*), *Web* (*ASP.Net 3.5*), *Smart Device*

(Pocet PC), Office (2003, 2007), Database (Access, SQL Server), Report (Crystal Report), Test, WCF, dan Workflow. Pilih Windows (Windows Form, Command line) seperti gambar di bawah ini.



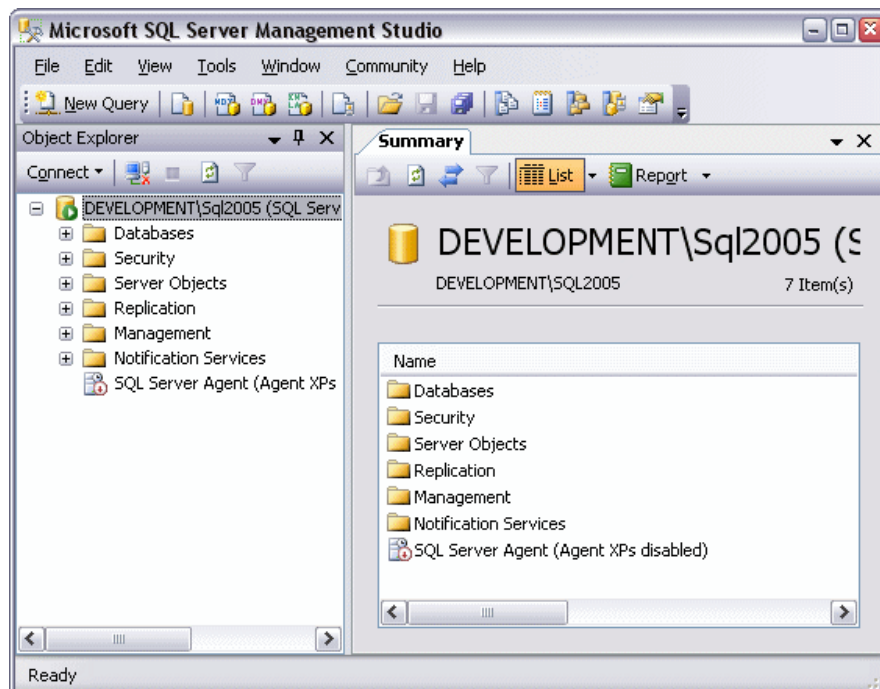
Gambar II.16 : Jendela New Project VB 2008.

II.8. SQL Server

Menurut Ketut Darmayuda (2010: 1-3), *SQL Server 2005* adalah produk lanjutan dari *Microsoft SQL Server 2000*, yang merupakan perangkat lunak yang dirancang untuk penanganan pengolahan data transaksi yang cukup besar.

Menurut Ema Utami dan Sukrisno (2008: 1), *SQL (Structure Query Language)* pada dasarnya adalah bahasa komputer standar yang ditetapkan untuk mengakses dan memanipulasi sistem database. Sebuah *database* terdiri dari satu atau tabel atau lebih dan memiliki nama yang berbeda untuk masing-masing tabel. Masing-masing tabel memiliki satu kolom (*field*) atau lebih dan memiliki baris (*record*). *Query* digunakan untuk mengakses dan mengolah database. SQL terdiri dari 5 bagian utama, yaitu:

1. *Retrieving data*: perintah untuk menampilkan data dari *database (SELECT)*.
2. *Data Definition Language (DDL)*: bahasa yang digunakan untuk membuat dan menghapus tabel atau *database* itu sendiri (*CREATE, DROP, ALTER*)
3. *Data Manipulation Language (DML)*: merupakan bahasa untuk memanipulasi/ mengubah isi tabel (*INSERT, DELETE, UPDATE*)
4. *Data Control Language (DCL)*: bahasa yang berhubungan dengan pengendalian akses ke *database (GRANT, REVOKE)*
5. *Data Transaction Language (DTL)*: bahasa yang digunakan untuk transaksi *database (COMMIT, ROLLBACK)*.



Gambar II.17 : Jendela Microsoft SQL Server 2005

II.9. Sejarah Singkat Perusahaan

PT. Buana Pilarjaya Mandiri Medan adalah salah satu perusahaan swasta nasional yang bergerak di bidang *General Contractor* yang melayani jasa antara lain :

1. Jasa pembangunan fasilitas telekomunikasi
2. Jasa pelaksanaan pondasi bored pile
3. Jasa pelaksanaan kontruksi gedung
4. Jasa pemeliharaan / perawatan gedung
5. Jasa pelaksanaan instalation radio transmission
6. General supplier dan leveransir

PT. Buana Pilarjaya Mandiri Medan didirikan di hadapan Notaris Fifi Rizani, S.H, di Kota Medan dengan bentuk Perseroan Komanditer berdasarkan Akte No.14 tertanggal 13 Juli 2006 dan Akte Perubahan No.11 tertanggal 23 Februari 20087dengan maksud dan tujuan menjalankan kegiatan usaha dalam bidang :

1. General Contactor
2. Supplier
3. General Tradding
4. Transportasi

Dalam menjalankan kegiatan usaha, PT. Buana Pilarjaya Mandiri Medan menganut prinsip kebersamaan dalam bekerja untuk kepuasan pelanggan.

PT. Buana Pilarjaya Mandiri Medan memiliki Kantor di Jl. Sidorukun No.12C Kelurahan Pulo Brayan I Kecamatan Medan Timur Medan – 20239. Telp. (061) 6644411, 6644422 Fax. (061) 6644433, Email : bpm_medan@yahoo.com dan berwork shop di Jl. Sidorukun No.88 Kelurahan Pulo Brayan Darat I Kecamatan Medan Timur Medan-20239. Telp.(061) 6613578, Fax. (061) 6644433, Email : bpm_medan@yahoo.com. Dengan Registrasi perusahaan :

1. TDP (Tanda Daftar Perusahaan) No.02.12.3.51.14620 diterbitkan oleh Dinas Perindustrian dan Perdagangan Kota Medan Tanggal 20 Agustus 2006.
2. SIUP (Surat Izin Usaha Perdagangan) No.2880/02.13/PK/VIII/2006 diterbitkan oleh Dinas Perindustrian dan Perdagangan Kota Medan Tanggal 16 Agustus 2006.
3. IUJK (Izin Usaha Jasa Kontruksi) No.1.131509.1275.2.0284/K.
4. SIGTU (Surat Izin Gangguan Tempat Usaha) No. 503/3507/BI/WAS/VIII/2006 diterbitkan oleh Dinas Perindustrian dan Perdagangan Kota Medan Tanggal 14 Agustus 2006.
5. NPWP (Nomor Pokok Wajib Pajak) 02.502.297.1-113.000 ditebitkan Tanggal 10 Agustus 2006.

Sebagai perusahaan yang baru berdiri, PT. Buana Pilarjaya Mandiri Medan belum mempunyai pengalaman yang cukup memadai, namu sebagai perusahaan yang berbasis pada pelayanan jasa telekomikasi khususnya pembangunan fasilitas Tower Telekomunikas (CME), dan akan terus berusaha memperbaiki pelayanan kepada pelanggan seperti prinsip yang dianut oleh perusahaan.

Berikut daftar pekerjaan yang telah berhasil di kerjakan oleh PT. Buana Pilarjaya Mandiri Medan sebagai berikut :

1. Project Siemens HCPT Tower 2007

Maincontractor	: PT. QDC Technologies
Jumlah	: 5 Site
Area	: North Sumatera
Sistem Pembayaran:	DP
	30% Pembayaran
	50% Proses Penyelesaian Project
	30% Pembayaran
	100% Proses Penyelesaian Project
	35% Pembayaran
	Jaminan 3 Bulan
	5% Pembayaran

2. Project Colocation 2007,2008,2009 (HCPT, Bakrie Telecom, Indosat, M8, XL)
 - Maincontactor : PT. Protelindo
 - Jumlah : 60 Site
 - Area : Sumatera, Jawa
 - Sistem Pembayaran: DP 30% Pembayaran
 - 80% Proses Penyelesaian Project 50% Pembayaran
 - 100% Proses Penyelesaian Project 20% Pembayaran

3. Project Telkomsel Tinem 2. 2007, 2008, 2009,2010
 - Maincontractor : PT. Nestic Bukaka
 - Jumlah : 10 Site
 - Area : North Sumatera
 - Sistem Pembayaran: DP 30% Pembayaran
 - 100% Proses Penyelesaian Project 65% Pembayaran
 - Jaminan 3 Bulan 5% Pembayaran

4. Project Excelcomindo Pratama 2007
 - Maincontractor : PT. Era Bangun Jaya
 - Jumlah : 3 Site
 - Area : North Sumatera, NAD
 - Sistem Pembayaran: DP 30% Pembayaran
 - 80% Proses Penyelesaian Project 50% Pembayaran
 - 100% Proses Penyelesaian Project 20% Pembayaran

5. Project Telkomsel Tinem 2. 2007, 2008
 - Maincontactor : PT. Badilany Parti
 - Jumlah : 2 Site
 - Area : North Sumatera
 - Sistem Pembayaran: DP 30% Pembayaran
 - 100% Proses Penyelesaian Project 65% Pembayaran
 - Jaminan 3 Bulan 5% Pembayaran

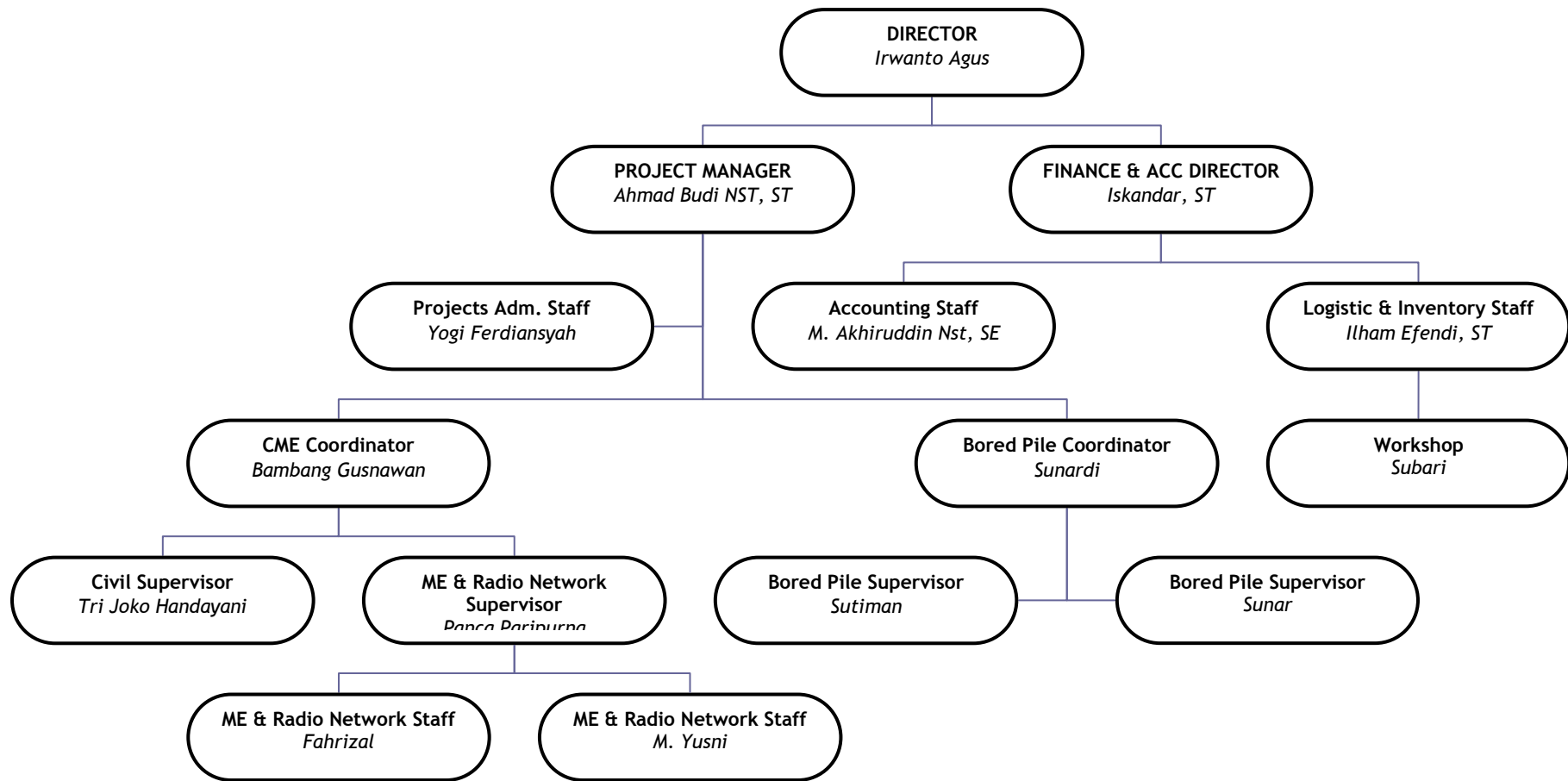
6. Project Installation ALFO/ELFO 2008-2009
 - Maincontractor : PT. Suryajaya Teknotama
 - Jumlah : 26 Site
 - Area : North Sumatera
 - Sistem Pembayaran: 100% Pengerjaan Project 100% Pembayaran

7. Project Borepile
 - Maincontractor : -
 - Jumlah : 150
 - Area : North Sumatera, NAD
 - Sistem Pembayaran: 100% Pengerjaan Project 100% Pembayaran

II.10. Struktur Organisasi

Di dalam setiap perusahaan terdapat struktur organisasi yang di dalamnya tercermin wewenang, tugas dan tanggung jawab masing-masing bagian dalam perusahaan. Wewenang adalah hak untuk melakukan suatu atau memerintah orang lain untuk melakukan sesuatu agar tercapai tujuan tertentu. Tanggung jawab adalah kewajiban untuk melaksanakan sesuatu yang timbul apabila seorang bawahan menerima wewenang untuk mendelegasikan tugas atau fungsi tertentu. Sedangkan tugas adalah sesuatu yang dimiliki oleh anggota dari suatu organisasi karena telah diberi tanggung jawab dari atasannya.

Struktur organisasi yang baik merupakan pengawasan terhadap organisasi dan merupakan salah satu sistem pertanggung jawaban diterapkan dalam suatu perusahaan. Untuk lebih jelasnya mengenai struktur organisasi PT. Buana Pilarjaya Mandiri dapat dilihat pada gambar II.18 berikut ini :



Gambar II.18 : Struktur Organisasi PT. Buana Pilarjaya Mandiri

Sumber : PT. Buana Pilarjaya Mandiri, 2012

Di dalam struktur organisasi tersebut dapat dijelaskan mengenai tugas, tanggung jawab dan wewenang dari masing-masing bagian sebagai berikut :

1. Director

Director mempunyai tugas sebagai berikut :

- a. Merencanakan kegiatan perusahaan dalam jangka pendek maupun jangka panjang. Mengawasi dan mengkoordinir dalam bidang sebagaimana tercantum dalam susunan organisasi untuk mencapai tujuan
- b. Merumuskan strategi perusahaan dan menjalankan kebijakan yang ditetapkan oleh badan pengawas dalam pelaksanaan operasi perusahaan sesuai dengan peraturan/ketentuan yang berlaku.
- c. Mewakili Perusahaan PT. Buana Pilarjaya Mandiri di dalam maupun di luar perusahaan dalam urusan operasional perusahaan

2. Project Manager

Project Manager mempunyai tugas sebagai berikut :

- a. Melakukan perencanaan pekerjaan, estimasi biaya, serta analisis financial dan resiko dari proyek
- b. Melakukan penjadwalan, pembelian, pengadaan dan mobilisasi semua sumber daya yang diperlukan untuk memulai dan mempertahankan proyek
- c. Memantau status dan kemajuan proyek,
- d. Membuat laporan pelaksanaan proyek untuk diserahkan kepada direktur.

3. Finance & Acc Director

Finance & Acc Director memiliki tugas dan tanggung jawab sebagai berikut :

- a. Mendapatkan sumber daya perusahaan atau sering disebut kas guna menjalankan operasi perusahaan untuk mencapai tujuan yang ditetapkan oleh perusahaan PT. Buana Pilarjaya Mandiri Medan.
- b. Menggunakan kas perusahaan baik pengeluaran operasi usaha maupun melakukan investasi aktiva tetap serta merencanakan kebutuhan kas perusahaan periode mendatang.
- c. Melaksanakan urusan pembukuan laporan keuangan
- d. Mengendalikan kegiatan-kegiatan dalam bidang keuangan perusahaan.

4. Administrasi

Administrasi memiliki tugas sebagai berikut :

- a. Melakukan koordinasi dan operasional umum perusahaan
- b. Menjaga agar seluruh dokumen tertata dan tidak rusak/hilang
- c. Mencatat dan menyimpan pesan-pesan penting
- d. Menyampaikan seluruh dokumen penting kepada atasan
- e. Menjaga detil informasi penyimpanan dan bukti peminjaman dokumen
- f. Membantu koordinasi proyek tertutup/terbatas dalam hal pengarsipan
- g. Mempersiapkan administrasi dan kontrak personal

5. CME (*Civil Mechanical Elektrical*)

CME (*Civil Mechanical Elektrical*) memiliki tugas sebagai berikut :

- a. Melakukan pembersihan lahan
- b. Melakukan penggalian dan pengecoran pondasi tower
- c. Melakukan ereksi / mendirikan tower
- d. Memasang Grounding (anti petir)
- e. Memasang lampu tower