



BAB II
LANDASAN TEORI

BAB II

LANDASAN TEORI

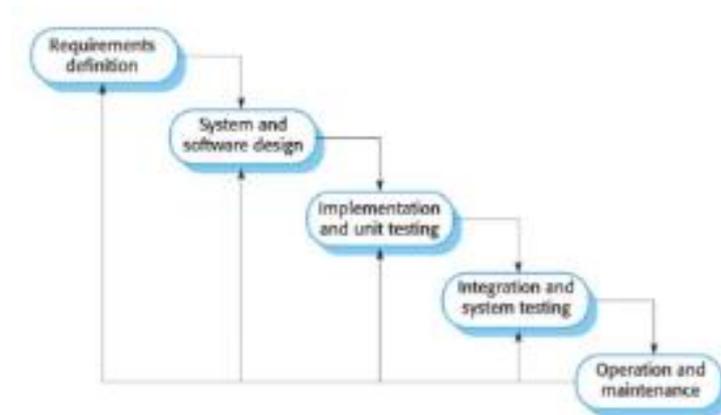
1.1 Landasan Teori

1.1.1 Rekayasa Perangkat Lunak

Rekayasa Perangkat Lunak merupakan suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak (*software*), dimulai dari tahap awal kajian spesifikasi / kebutuhan sistem sampai pemeliharaan sistem setelah digunakan. Menurut IEEE (*Standard Glossary of Software Engineering Terminology*,1990) mendefinisikan perangkat lunak sebagai program komputer, prosedur, dan dokumentasi serta data yang terkait dengan pengoperasian sistem komputer. Jadi, perangkat lunak bukan sekedar sebuah program komputer, akan tetapi termasuk didalamnya dokumen – dokumen yang menyertainya. Dokumen – dokumen penyerta berupa dokumen desain, dokumen pengoperasian, serta dokumen pemeliharaan [11].

A. Metode Pengembangan Sistem

Rancang Bangun Sistem Informasi Penjualan Gas Lpg Pada Distributor Pangkalan Gas Lpg Berbasis Website ini menggunakan metode SDLC Waterfall dalam pengembangan perangkat lunak sistemnya. Hal ini dikarenakan pengembangan sistem menggunakan metode ini setiap tahapannya harus diselesaikan secara sistematis. Adapun tahapan pada metode pendekatan SDLC Waterfall menurut Ian Sommerville dapat dilihat pada Gambar 2. 1 berikut penjelasan tahapannya:



Gambar 2. 1 Model SDLC Waterfall – Ian Sommerville [12]

1. Tahap Analisis Kebutuhan (*Requirement Analysis*)

Pada tahap ini dilakukan analisis kebutuhan sistem dengan cara melakukan observasi pada Pangkalan Gas dan melakukan wawancara pada karyawan Pangkalan Gas. Dilakukannya analisis ini untuk mengetahui permasalahan yang ada, kebutuhan dari sistem yang akan dibangun mulai dari jumlah aktor dan kebutuhan fungsional mulai dari input-proses-output sistem.

2. Tahap Desain (*Design*)

Pada desain ini sistem akan dibuat dengan data yang didapatkan pada tahap analisis sistem. Desain harus dibuat detail agar mudah dalam memasukkannya dalam koding. Desain sistem yang dibuat meliputi use case diagram, sequence diagram, dan activity diagram.

3. Perkembangan (*Development*)

Pada tahap ini sistem mulai dibangun atau dikembangkan. Pada proses ini dilakukan dengan pembuatan program dalam membangun Sistem Informasi.

4. Pengujian (*Testing*)

Tahap ini merupakan tahap dimana seorang pembuat sistem atau programmer secara nyata melakukan coding atau pembuatan sistem dengan bahasa pemrograman sesuai desain yang telah disepakati. Sebelum sistem yang dibuat diserahkan kepada user, perlu dilakukan pengujian terlebih dahulu. Pengujian dilakukan untuk mengetahui apakah dapat diterima. Pemakai akan melakukan pengecekan apakah sistem atau aplikasi sudah sesuai dengan kebutuhan atau belum. Pada tahap ini seorang user juga dapat memberikan tanggapan mengenai sistem yang telah dibuat.

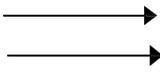
5. Pemeliharaan (*Maintenance*)

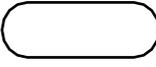
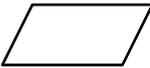
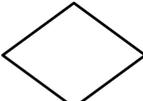
Pada tahap ini dilakukan evaluasi pada seluruh program dengan mengetahui data-data fungsional sistem berjalan sesuai rencana fungsional sistem yang belum berjalan sebagaimana mestinya. Tahap ini juga mengkaji untuk pengembangan berikutnya dalam membangun Sistem Informasi yang dibuat.

B. Flowchart

Bagan alir merupakan bagan-bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika [11]. Berikut simbol – simbol dalam flowchart dapat dilihat pada Tabel 2. 1 :

Tabel 2. 1 *Simbol Flowchart*

| No | Nama | Simbol | Keterangan |
|----|---|---|---|
| 1. | Flow Direction Symbol/Con necting Line |  | Berfungsi untuk meghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses. |
| 2. | Connector (on-page referance) |  | Digunakan untuk menyatakan sambungan proses yang satu |

| No | Nama | Simbol | Keterangan |
|-----|---------------------------------------|---|---|
| | | | ke proses berikutnya dalam halaman yang sama |
| 3. | Connector (off-page referance) |  | Digunakan untuk menyatakan sambungan proses yang satu ke proses berikutnya dalam halaman yang berbeda |
| 4. | Terminal |  | Digunakan untuk memulai atau mengakhiri program |
| 5. | <i>Input</i> dan <i>Output</i> |  | Menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya |
| 6. | <i>Processing/</i> Proses |  | Digunakan untuk menunjukkan pengolahan yang akan dilakukan dalam komputer |
| 7. | <i>Decision /</i> <i>keputusan</i> |  | Digunakan untuk memilih proses yang dilakukan berdasarkan kondisi tertentu |
| 8. | <i>Manual</i> <i>Operation</i> |  | Digunakan untuk menunjukkan kegiatan yang tidak dilakukan oleh komputer |
| 9. | <i>Document/</i> Dokumen |  | Simbol yang menyatakan bahwa <i>input</i> berasal dari dokumen dalam bentuk fisik atau <i>output</i> yang perlu dicetak |
| 10. | <i>Manual Input</i> |  | Digunakan untuk menginput data secara manual dengan keyboard |

| No | Nama | Simbol | Keterangan |
|-----|---------------------------|---|---|
| 11. | <i>Predefined process</i> |  | Digunakan untuk mempersiapkan penyimpanan yang sedang/akan digunakan dengan memberikan harga awal |

C. UML

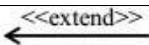
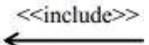
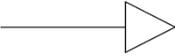
UML adalah istilah berbasis objek dan teknik diagram untuk memodelkan proyek pengembangan sistem dari analisis hingga desain [11].

1. Use Case Diagram (*Unified Modelling Language*)

Use Case Diagram adalah sebuah diagram yang menjelaskan apa yang harus dilakukan oleh sistem. *Use case* diagram digunakan untuk menjelaskan dan mendokumentasikan interaksi yang diperlukan antara pengguna dengan sistem [11]. Adapun simbol – simbol *use case* diagram dapat dilihat pada Tabel 2. 2.

Tabel 2. 2 Simbol *Use Case Diagram*

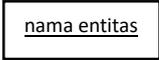
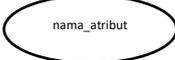
| No | Nama | Simbol | Keterangan |
|----|--------------------------------|---|---|
| 1. | Use Case |  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor. |
| 2. | Aktor/ <i>Actor</i> |  | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang dibuat |
| 3. | Asosiasi/ <i>Asosiation</i> |  | Penghubung atau komunikasi antar aktor dan <i>use case</i> |

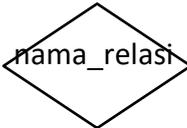
| No | Nama | Simbol | Keterangan |
|----|---|---|--|
| 4. | Ekstensi/ <i>Ekstend</i> |  | Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri meski tanpa use case tambahan itu. |
| 5. | <i>Include</i> |  | Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya. |
| 6. | Generalisasi <i>/generalizati on</i> |  | Hubungan generalisasi dan spesialisasi antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |

2. Entity Relationship Diagram (ERD)

Entity Relationship Diagram merupakan tools yang digunakan untuk memodelkan struktur data dengan menggambarkan entitas dan hubungan antar entitas [13]. Adapun simbol – simbol *entity relationship* diagram dapat dilihat pada Tabel 2. 3

Tabel 2. 3 Simbol Entity Relationship Diagram

| No | Nama | Simbol | Keterangan |
|----|----------------------------|---|---|
| 1. | Entitas/ entity |  | Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel. |
| 2. | Atribut |  | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas. |
| 3. | Atribut kunci primer |  | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan. |

| | | | |
|----|--|---|---|
| 4. | Atribut multinilai <i>/ multivalue</i> |  | Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu. |
| 5. | Relasi |  | Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja. |
| 6. | Asosiasi / <i>association</i> |  | Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. |

2.1.2 Sistem Informasi

Sistem Informasi merupakan suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian dan laporan perbulan yang dibutuhkan oleh pimpinan. Maka dengan adanya sistem informasi bisa memudahkan dalam pengelolaan data [14].

2.1.3 Penjualan

Penjualan adalah ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang atau jasa yang ditawarkan. Maka dari adanya penjualan bisa menciptakan suatu proses pertukaran barang atau jasa antara penjual dan pembeli [15].

2.1.4 Distribusi

Distribusi merupakan sekumpulan organisasi yang membuat sebuah proses kegiatan penyaluran suatu barang atau jasa siap untuk di pakai atau di konsumsi oleh para konsumen. Maka dari adanya distribusi dapat memperlancar dan mempermudah penyampaian barang dan jasa dari produsen ke konsumen, sehingga penggunaanya sesuai yang diperlukan misalkan harga [16].