# 

**BAB I**

PENDAHULUAN

* 1. Latar Belakang

Meningkatnya persaingan dalam dunia bisnis membuat teknologi informasi menjadi komponen penunjang bagi perusahaan agar bisa tetap bersaing dengan perusahaan-perusahaan lainnya. Salah satu teknologi yang terus berkembang dan semakin banyak yang menggunakan adalah internet. Internet membuat tidak ada batasan dalam berbisnis, karena membuat manusia saling terhubung secara global untuk saling bertukar informasi atau data, berbelanja dan melakukan kegiatan sosial. Perusahaan yang memanfaatkan teknologi internet dapat memperluas jangkauan pasar, karena tidak ada batasan waktu dan tempat. Sehingga peluang bisnis berbasis internet lebih menguntungkan. Hal ini semakin ditandai dengan maraknya bisnis yang berbasis *online*, seperti toko *online* dan reservasi tiket *online*.

Hotel Garuda Wira merupakan salah satu hotel di Jakarta Pusat dengan konsep *low budget*, tepatnya terletak di jalan Biduri Pandan No.70. Letaknya yang strategis sekitar 1,15 km dari stasiun senen dan 2,68 km dari pusat belanja ITC Cempaka Mas membuat hotel ini merupakan salah satu hotel favorit bagi wisatawan ber*budget* hemat. Untuk kamar, hotel ini memiliki 60 kamar dengan menawarkan 2 tipe kamar yang dibedakan berdasarkan fasilitas pendingin ruangan, dan ukuran kamar. Harga kamar yang ditawarkan Hotel Garuda Wira ini relatif murah, hanya dengan 500 ribuan sudah bisa menginap di hotel ini dengan fasilitas lengkap.

Hotel Garuda Wira saat ini belum memiliki *website* dan layanan pemesanan kamar *online* sendiri, oleh karena itu Hotel Garuda Wira berencana untuk bergabung dengan *website* pencarian hotel seperti pegipegi.com tetapi menjadi anggota *website* pencarian hotel memiliki kekurangan dan kerugian seperti persaingan secara langsung dengan hotel-hotel lain, harga hotel yang tidak bisa kompetitif dikarenakan ada tambahan biaya layanan, pembayaran dari pihak *customer* tidak dapat langsung diterima dan tidak dapat melakukan inovasi layanan dan promosi sendiri. Selain permasalahan tersebut, untuk reservasi di Hotel Garuda Wira masih melakukan proses manajemen secara manual, seperti proses terima tamu, status dan kondisi kamar dicatat menggunakan buku, sehingga saat proses terima tamu dan pemilihan kamar harus dilakukan cek pada buku. Sering kali petugas lupa *update* status kamar, sehingga tidak dapat diketahui jumlah kamar yang dapat digunakan secara tepat dan cepat.

Dengan maksud meningkatkan pelayanan, memperluas jangkauan pasar dan perubahan konsep hotel. Pemilik Hotel Garuda Wira menginginkan sistem reservasi hotel dengan layanan *self service.* Layanan *self service* yang diberikan adalah pemilihan posisi dan lantai kamar, ukuran kamar, pilihan tempat tidur, pemilihan fasilitas kamar seperti televisi, kulkas, pendingin ruangan, kelengkapan mandi, selimut dan pelanggandapat memilih menu sarapan pagi atau tidak sarapan pagi. Dengan adanya layanan *self service*, pelanggan dapat menentukan sendiri kebutuhannya sesuai dana yang tersedia. Sistem yang dibuat juga memberikan informasi jumlah ketersediaan kamar, sehingga pelanggan dipastikan mendapatkan kamar yang dipesan. Saat melakukan pemesanan, pelanggan ditampilkan denah posisi kamar di setiap lantai yang dipilih. Dengan adanya denah, pelanggan dapat mengetahui posisi kamar menghadap ke arah mana. Hal ini dapat meningkatkan pilihan dan kenyamanan pelanggan.

Setelah melakukan proses pemesanan, pelanggan mendapatkan nomer reservasi dan cara pembayaran. Setelah melakukan pembayaran, pelanggan melakukan konfirmasi pembayaran dengan menyertakan bukti pembayaran. Petugas hotel akan melakukan verifikasi pembayaran pelanggan, dan merubah status pemesanan jika pembayaran telah diterima. Pelanggan mendapatkan bukti pembayaran melalui email untuk dibawa saat *check in.*  Jika pelanggan tidak membayar lebih dari 4 jam dari waktu pemesanan, maka pemesanan dibatalkan secara otomatis oleh sistem.

Penulis berharap dengan adanya “**Sistem Reservasi *Online* di Hotel Garuda Wira dengan Layanan *Self Service****”* meningkatkan pelayanan dan memberikan nilai tambah bagi Hotel Garuda Wira untuk mendapatkan jangkauan pasar yang lebih luas dan memenangkan persaingan usaha dibidang perhotelan.

* 1. Rumusan Masalah

Berdasarkan dari latar belakang permasalahan yang terjadi, diperoleh rumusan masalah sebagai berikut

1. Bagaimana merancang bangun sistem reservasi online hotel dengan algoritma metode self service?
2. Bagaimana penerapan layanan self service saat proses reservasi dilakukan ?
   1. Tujuan dan Manfaat
      1. **Tujuan**

Tujuan dari penelitian ini adalah sebagai berikut

1. Memberikan layanan kepada pelanggansesuai dengan yang dibutuhkan saat menginap di Hotel Garuda Wira.
2. Memberikan harga yang terjangkau sesuai dengan kebutuhanpelanggan.
3. Memberikan kemudahan bagi pelanggan untuk mendapatkan informasi mengenai ketersediaan kamar dan fasilitas yang ada pada hotel.
4. Memperluas jangkauan pasar dan meningkatkan daya saing Hotel Garuda Wira.
5. Dapat dilakukan monitor jumlah kamar yang tersedia dan jumlah pemesanan untuk memudahkan *staff* melihat status kamar serta menghindari terjadinya *overbooking* yang sering terjadi saat proses reservasi secara manual.
   * 1. **Manfaat**

Manfaat yang diharapkan dari penelitian ini yaitu:

1. Bagi Penulis, sebagai media penerapan ilmu yang didapatkan dari hasil perkuliahan.
2. Bagi Hotel Garuda Wira bermanfaat untuk meningkatkan pelayanan kepada pelanggan dengan memberikan kemudahaan pemesanan kamar hotel yang sesuai dengan kebutuhan dan budjet.
3. Bagi akademisi atau mahasiswa dapat menjadi refrensi dalam melakukan penelitian yang terkait dengan sistem pelayanan reservasi hotel dengan menggunakan algoritma self service.
   1. Batasan Masalah

Agar penelitian yang dilakukan dapat terarah dengan baik dan tidak menyimpang dari pokok permasalahan, maka penulis membatasi permasalahan yang akan dibahas, yakni:

1. Sistem yang dibuat hanya sebatas layanan pemesanan di Hotel Garuda Wira.
2. Pelanggan melakukan konfirmasi pembayaran dengan menyertakan bukti pembayaran melalui menu sistem yang disediakan.
3. Pemesanan memiliki batas waktu 4 jam jika pelanggan tidak melakukan pembayaran dan meng*upload* bukti bayar, pemesanan hotel dianggap batal dan kamar yang sudah dipesan akan tersedia lagi.
4. Algoritma *self service* yang diterapkan bekerja dengan dasar memilih sendiri fasilitas yang diperlukan yaitu pemilihan posisi dan lantai kamar, ukuran kamar, pilihan tempat tidur, pemilihan fasilitas kamar seperti televisi, pendingin ruangan, kelengkapan mandi, selimut, dan pelanggan dapat memilih untuk mendapatkan sarapan atau tidak sarapan pagi.
5. Laporan yang dihasilkan adalah laporan jumlah pemesanan kamar, laporan status pesanan kamar dan laporan fasilitas yang paling diminati oleh pelanggan.
   1. Metode Penelitian
      1. Jenis Penelitian

Jenis penelitian ini berdasarkan prinsip penelitian terapan. Hal ini dikarenakan dengan dasar dari penelitian bersifat pengembangan solusi dari permasalahan yang ada. Oleh karena itu analisis sistem menggunakan metode kualitatif, karena berdasarkan proses analisa terhadap permasalahan yang terjadi. Hasil dari penelitian dilakukan analisa asosiatif untuk mendapatkan pengaruh pemakaian sistem oleh pengguna.

**1.5.1 Metode** **Pengumpulan Data**

Dalam melakukan pengumpulan data-data yang diperlukan dalam melakukan penelitian ini adalah sebagai berikut:

1. Wawancara

Proses wawancara dilakukan kepada pemilik dan pihak manajemen Hotel Garuda. Hasil wawancara diharapkan mendapatkan informasi permasalahan yang terjadi dan harapan pengembangan sistem yang baru.

1. Studi Data dan Dokumentasi

Pada tahap ini, dilakukan analisa data yang diberikan oleh Hotel Garuda berkaitan dengan kegiatan layanan reservasi kamar dan pelayanan Tamu di Hotel Garuda.

**1.5.2 Metoda Pengembangan Perangkat Lunak**

Metodelogi yang digunakan untuk membangun sistem ini adalah Model *Waterfall* berdasar teori *Roger Pressman (2012).* Tahapan proses pengembangan perangkat lunak adalah sebagai berikut:

1. *System Engineering*

Pada tahapan ini dilakukan penetapan fokus permasalahan yang akan diteliti yaitu di bidang layanan reservasi pelanggan di Hotel Garuda.

1. *Analysis*

Melakukan identifikasi permasalahan dengan melakukan wawancara dan survey ke tempat studi kasus untuk mengetahi permasalahan-permasalahan yang terjadi dan kebutuhan pengembangan sistem untuk menjadi solusi.

1. *Design*

Pada tahapan ini dilakukan penetapan solusi atas permasalahan yang terjadi dan pembuatan desain sistem yang akan dibuat. Desain sistem menggunakan diagram *Unified Modelling Language (UML),* diagram relasi database dan desain tampilan sitem. Diagram *UML* yang digunakan adalah *Use Case Diagram, Activity diagram, Sequance Diagram, Class Diagram, Component Diagram* dan *Deployment Diagram.*

1. *Coding*

Tahapan proses pengembangan desain sistem dalam bentuk sistem reservasi menggunakan metode self service berbasis website. Pengkodean sistem menggunakan bahasa *PHP*  dengan *framework* *Code igneiter (CI)* dan editor pemprograman *notepad++.*  Management database sistem menggunakan *MySQL*

1. *Testing*

Uji coba dilakukan dengan menggunaknametode *Black Box Testing.* Dengan melakukan uji coba diharapkan, sistem yang dibuat sesuai dengan desain dan tujuan penelitian.

* 1. Sistematika Penulisan

Uraian singkat mengenai struktur penulisan pada masing-masing bab adalah sebagai berikut:

|  |
| --- |
| **PENDAHULUAN**  Membahas Latar Belakang Masalah, Identifikasi Masalah, Batasan Masalah, Tujuan Penelitian, Metodelogi Penelitian serta Sistematika Penulisan.  **LANDASAN TEORI**  Memaparkan teori-teori yang didapat dari sumber-sumber yang relevan untuk digunakan sebagai panduan dalam penelitian serta penyusunan laporan tugas akhir.  **PERANCANGAN SISTEM**  Berisi tentang perancangan sistem serta komponen-komponen pemodelan sistem yang digunakan.  **IMPLEMENTASI SISTEM**  Bab ini menguraikan langkah-langkah dalam implementasi sistem, disertai dengan komponen-komponen kebutuhan sistem.  **PENUTUP**  Mengemukakan kesimpulan yang diambil dari hasil penelitian dan perancangan sistem, serta saran-saran untuk pengembangan selanjutnya, agar dapat dilakukan perbaikan-perbaikan di masa depan. |

BAB II

LANDASAN TEORI

2.1 Rekayasa Perangkat Lunak

Perangkat lunak adalah program komputer, prosedur, aturan, dan dokumentasi yang berkaitan serta data, yang bertalian dengan operasi suatu sistem komputer [ Roger S. Pressman : 2011 ] .

Perangkat lunak lebih dikenal dengan elemen lojik daripada fisik, oleh karena itu perangkat lunak memiliki karakteristik :

1. Perangkat lunak itu dikembangkan dan direkayasa bukan di produksi
2. Produk yang unik dan tidak akan pernah rusak
3. Tidak terlihat (*invisible*) dan pada umumnya dibentuk sesuai keinginan
4. Mudah untuk dimodifikasi dan bisa langsung terhubung dengan komputer

Rekayasa perangkat lunak adalah Pembentukan dan penggunaan prinsip rekayasa ( *engineering* ) untuk mendapatkan perangkat lunak secara ekonomis namun andal dan dapat bekerja secara efesien pada computer [ Roger S. Pressman, 2012 ]

Tujuan dari rekayasa perangkat lunak adalah untuk menghasilkan perangkat lunak yang berkualitas yang dilihat dari 3 sisi yaitu sponsor ( individu atau organisasi yang membiayai proyek ), sis pemakai dan juga sisi *management* ( pemeliharaan dan pemodifikasi).

2.2 Aplikasi

Aplikasi adalah perangkat lunak yang menjadi tahapan awal dalam sebuah sistem yang digunakan untuk mengolah beberapa data menjadi suatu informasi yang berguna bagi orang-orang dan sistem yang bersangkutan. [ M. Hilmi Masruri & Java Creativity : 2015 ].

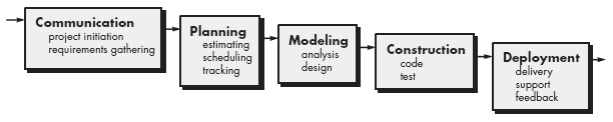
Aplikasi berasal dari bahasa inggris yaitu “application” yang berarti penerapan ataupun penggunaan. Sedangkan secara istilah, pengertian aplikasi adalah suatu program yang siap untuk digunakan dan dibuat untuk menyelesaikan dan melaksanakan suatu fungsi bagi pengguna jasa aplikasi yang bersangkutan serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Contoh untuk sebuah aplikasi adalah aplikasi absensi yang dibuat dengan tujuan meningkatkan kedisiplinan kehadiran siswa dan siswi dan juga sebagai aset tolak ukur dalam proses pembelajaran di sekolah

2.3 Metodologi Pengembangan Sistem

Dalam pengembangan sebuah sistem dibutuhkan beberapa proses yang disebut *System Development Life Cycle (SDLC)* atau Siklus hidup dalam pengembangan sistem. SDLC adalah proses perancangan sistem serta metodologi yang digunakan untuk mengembangkan sistem yang akan dibuat [ Azhar Susanto : 2012 : 356 ]. Dalam perancangan aplikasi tugas akhir ini, metode yang digunakan adalah metode *Waterfall* sebagai pola dalam perancangan sistem yang akan dibuat.

2.3.1 Definisi Waterfall

Metodologi pengembangan sistem dengan cara *Waterfall* adalah metodologi terlama yang pernah digunakan. Metodologi ini pertama kali muncul di tahun 1970. Pada model ini dilakukan pendekatan secara sistematis dan urut dari setiap level pengembangan perangkat lunaknya [ Roger S. Pressman : 2011 ]



**Gambar 2.1** *Waterfall Model* [ Roger S. Pressman, 2010 ]

Penjelasan dari gambar *Waterfall Model* sebagai berikut :

1. Analisa kebutuhan ( Communication )

Proses pengumpulan kebutuhan dalam perancangan sistem untuk memahami dasar dari sistem yang akan dibuat. Analis harus mengetahui ruang lingkup, fungsi – fungsi dan kemampuan kinerja yang ingin dihasilkan oleh sistem. Biasanya dilakukan pertemuan dengan Konsumen yang akan memakai aplikasi yang akan dibuat.

1. Perencanaan Sistem ( Planning )

Setelah ditetapkan untuk data apa saja yang akan digunakan dalam sistem, selanjutnya adalah perencaan sistem ini akna dibuat seperti apa. Pada tahap ini menghasilkan *user requirement* sebagai data yang langsung menjelaskan tentang kebutuhan konsumen dalam pembuatan *software.*

1. Perancangan ( Modeling )

Setelah *user requirement* disetujui oleh pihak konsumen dan developer, digunakan metode *modeling* yang berisi lebih kearah analisa sistem yang akan dibuat. Dalam menganalisa biasanya digunakan konsep *design* atau gambar untuk lebih mudah dibaca konsepnya.

1. *Construction / Coding & Testing*

Setelah sistem sudah dianalisa dan di *mapping* dengan *design* yang sesuai, kemudian dilakukan pemrosesan alur sistem kedalam bahasa program yang nantinya akan diproses oleh komputer. Proses ini dilakukan oleh *developer*. Setelah program sudah masuk kedalam sistem dan hasil sudah tersedia, data output akan di *testing* terlebih dahulu sebelum di implementasikan ke konsumen. Proses *testing*  ini ditujukan untuk membandingkan output dari proses sebelumnya dengan hasil proses yang sudah dilakukan sistem. Jika data sudah sama atau lebih *valid*, sistem sudah siap untuk digunakan. *Testing* biasanya dilakukan oleh tim QC ( *Quality Control* )

1. *Deployment*

Setelah aplikasi atau sistem sudah dites dan hasilnya *valid*, maka aplikasi siap untuk ditunjukkan dan dicoba oleh konsumen. Disitu tidak hanya penyampaian atau pembelajaran tentang aplikasi yang harus diberikan, tetapi juga dilakukan pengecekan jika ada data yang masih kurang sesuai dengan konsumen. Darisitu biasanya juga dilakukan *maintenance* aplikasi tersebut oleh tim dari pembuat aplikasi tersebut untuk memberikan arahan-arahan jika konsumen masih belum terlalu paham dengan aplikasi yang sudah dibuat.

2.4 *UML ( Unified Modeling Language )*

Tujuan dari *Unified Modeling Language* adalah untuk menyediakan istilah-istilah atau simbol secara teknik berbasis objek dan penggunaan teknik diagram yang cukup banyak untuk menggambarkan proyek pengembangan sistem mulai dari analisa sampai desain [ Alan Dennis, Barbara Haley Wixom dan Roberta M. Roth, 012:513 ].

Diagram-diagram pada *Unified Modeling Language* (UML) dibagi menjadi 2 jenis yaitu *structure diagrams* dan *behavior diagrams. Structure Diagrams* yaitu diagram yang digunakan untuk menggambarkan data dan relasi paten yang terdapat pada sistem informasi. Sedangkan *Behavior Diagrams* adalah diagram yang menyediakan analisa dengan cara menggambarkan relasi yang dinamis antar objek yang mewakili sistem informasi yang akan dibangun. Dalam perancangan aplikasi ini digunakan beberapa step Diagram UML yang digunakan yaitu :

1. Usecase Diagram ( Analisa sistem )
2. *Activity Diagram* ( Analisa dan Design sistem )
3. *Sequence Diagram* (Analisa dan Design sistem)
4. *Class Diagram* ( Analisa dan Design sistem )

2.4.1 Use Case Diagram

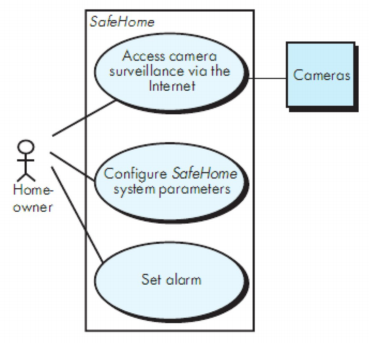
*Use Case Diagram* merupakan model diagram *UML ( Unified Modeling Language )* yang digunakan untuk menggambarkan kesepakatan fungsional yang diharapkan dari sebuah sistem. Use Case Diagram menekankan pada “siapa” dan melakukan “apa” dalam lingkungan sistem preangkat lunak yang akan dibangun. Intinya, didalam diagram ini seorang user bisa melakukan apa saja ke dalam kesepakatan atau kebutuhan dari perangkat lunak yang akan dibuat.

*Use Case Diagram* adalah design untuk menggambarkan interaksi antara aktor dengan sistem yang membentuk peran dan aktifitas dalam suatu sistem [ Roger S. Pressman, 2015 ]. Diagram ini menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada diluar sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar. Berikut adalah beberapa simbol yang digunakan di usecase :

**Table 2.1** Komponen *Use Case Diagram* [ Roger S. Pressman, 2015 ]

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama Elemen** | **Keterangan** |
|  | Aktor | * Aktor adalah seseorang atau sesuatu yang berperan untuk berinteraksi dengan sistem tetapi tidak memiliki kontrol akan *use case*. |
|  | UseCase | * Simbol UseCase menggambar kan fungsionalitas dari sistem, sehingga pengguna paham mengenai manfaat dari sistem yang dibangun |
|  | Asosiasi | * Penghubung antar elemen ( aktor / UseCase ) di dalam sistem |
|  | Generalisasi | * Sebuah elemen yang dihasilkan dari pewarisan elemen lain |
|  | Batasan sistem | * Area yang digunakan untuk menempatkan usecase sebagai batasan apa yang dilakukan sistem |

Contoh untuk penggunaan *Use Case Diagram* dalam pembangunan sistem.



**Gambar 2.2** *Use Case Diagram* [ Roger S. Pressman, 2015 ]

2.4.2 *Activity Diagram*

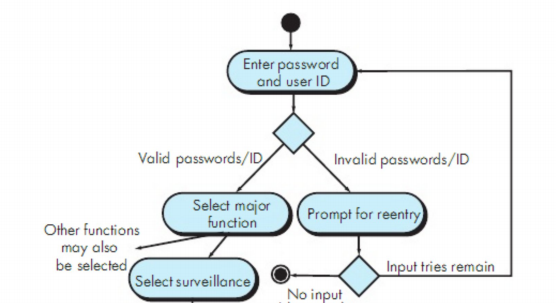
*Activity Diagram* adalah bagan Diagram yang merepresentasikan analis dengan tujuan untuk memodelkan alur proses dan hubungan antar proses dalam suatu sistem informasi dan juga berisikan tentang scenario yang ada dalam sistem tersebut [ Roger S. Pressman, 2015 ].

Berikut komponen – komponen yang ada didalam sebuah model *Activity Diagram* :

**Table 2.2** Komponen *Activity Diagram* [ Roger S. Pressman, 2015 ]

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama Elemen** | **Keterangan** |
|  | Aktifitas | * Notasi yang menggambarkan aktor dari proses dalam aliran pekerjaan |
|  | Transisi | * Notiasi yang digunakan mem- perlihatkan aliran kontrol dari satu aktifitas ke aktifitas lain |
|  | *Decision* | * Notasi yang menandakan ada- nya kontrol cabang berdasarkan *decision point* |
|  | Awalan & Akhiran | * Notasi yang menandakan awal dan akhir dari sebuah aktifitas sistem |

Contoh dari penerapan *activity diagram* adalah sebagai berikut:



**Gambar 2.3** *Activity Diagram* [ Roger S. Pressman,2015]

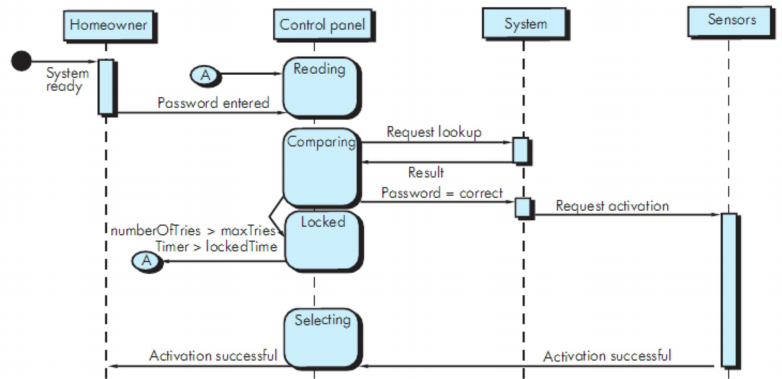
2.4.3 Sequence Diagram

*Sequence Diagram* adalah diagram yang menjelaskan interaksi antar objek yang disusun berdasarkan waktu proses berlangsung. Diagram ini digunakan untuk menggambarkan tahap demi tahap yang harus dilakukan oleh pengguna sistem untuk menghasilkan sesuatu dari *use case diagram* yang sudah dibuat [Roger S. Pressman , 2015]. Berikut adalah beberapa komponen yang sangat berpengaruh dalam pembentukan sebuah *Sequence Diagram* :

**Table 2.3** Komponen *Sequence Diagram* [ Roger S. Pressman, 2015 ]

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama Elemen** | **Keterangan** |
|  | Objek | * Komponen yang menjadikan sebuah objek dalam membuat diagram |
|  | *Stimulus* | * Untuk menandakan hubungan komunikasi antar objek |
|  | *Self Stimulus* | * Fungsi sama dengan *Stimulus,* tetapi pesan yang disampaikan dikirimkan untuk objek itu sendiri |
|  | *Focus Control* | * Sebagai tempat untuk hasil input atau output dari sebuah proses yang dilakukan oleh objek ataupun aktor yang ada dalam sistem |

Contoh dari penerapan *sequence diagram* adalah sebagai berikut:



**Gambar 2.4** *Sequence Diagram* [ Roger S. Pressman,2015]

2.4.4 *Class Diagram*

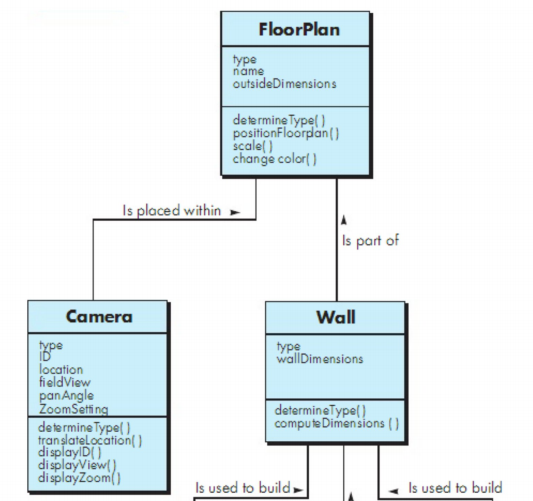
*Class Diagram* adalah diagram yang menunjukkan *class-class­* yang ada dari sebuah sistem dan saling berhubungan secara logika [ Roger S. Pressman , 2015 ]. Diagram ini menggambarkan struktur statis dari sebuah sistem. Karena itu, *class diagram* merupakan tulang punggung atau kekuatan dasar dari hamper setiap metode berorientasi objek termasuk *UML ( Unified Modeling Language ).*

Berikut komponen – komponen yang harus dipakai dalam pembuatan *class diagram :*

**Table 2.4** Komponen *Class Diagram* [ Roger S. Pressman, 2015 ]

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama Elemen** | **Keterangan** |
|  | Kelas | * Simbol untuk membangun sebuah pemrograman dengan objek * Terdiri 3 bagian, bagian **atas** adalah nama kelas, bagian **tengah** adalah atribut, dan bagian **bawah** adalah metode dari kelas tersebut |
|  | Garis Asosiasi | * Simbol yang menggambarkan adanya hubungan antara satu kelas dengan kelas lainnya |
|  | Generalisasi | * Simbol yang menandakan adanya generalisasi dari kelas input untuk menghasilkan data yang dibutuhkan |

Contoh dari penerapan *Class diagram* adalah sebagai berikut:



**Gambar 2.5** *Class Diagram* [R.S. Pressman,2015]