

**LAPORAN KERJA PRAKTEK  
RANCANG BANGUN  
SISTEM INFORMASI SPE CIFICATION MAKER  
STUDI KASUS DI ASCOMPUTER  
YOGYAKARTA**

**Diajukan Sebagai Salah Satu Syarat  
Untuk Memperoleh Gelar Sarjana Teknik Informatika**



Disusun oleh :

**KHOLIS HERMAWAN**

**08650068**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SUNAN KALIJAGA  
YOGYAKARTA  
2012**

## **LEMBAR PENGESAHAN**

### **LAPORAN KERJA PRAKTEK RANCANG BANGUN SISTEM INFORMASI SPECIFICATION MAKER STUDI KASUS DI ASCOMPUTER YOGYAKARTA**

Disusun Oleh :

**Nama : Kholis Hermawan**

**NIM : 08650068**

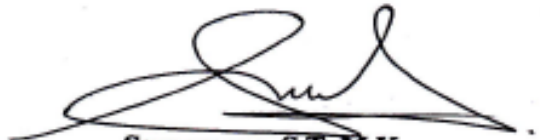
Telah diseminarkan pada tanggal : 25 April 2012

Dosen Pembimbing

Dosen Penguji



**Nurochman, S.Kom, M.Kom**  
NIP. 19801223 200901 1 007



**Sumarsono, S.T, M.Kom**  
NIP. 19710209 200501 1 003

Mengetahui,

A.n. Dekan Fakultas Sains dan Teknologi

Ketua Program Studi Teknik Informatika



**Agus Mulvanto, S.Si, M.Si**  
NIP. 19710823 199903 1 003

## KATA PENGANTAR

*Assalamualaikum wr. wb.*

Segala puji syukur bagi Allah SWT yang telah melimpahkan rahmat, hidayah serta karunianya sehingga pelaksanaan kerja praktek yang dilakukan di ASComputer Yogyakarta dapat terselesaikan dengan baik dan tepat waktu. Pelaksanaan kerja praktek ini merupakan salah satu syarat untuk dapat memperoleh gelar sarjana Teknik Informatika di Universitas Islam Negeri Sunan Kalijaga Yogyakarta.

Selanjutnya penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Allah SWT yang telah melimpahkan rahmat, hidayah serta karuniaNya sehingga penulis dapat menyelesaikan kerja praktek dengan baik dan tepat waktu.
2. Orang tua, istri dan anak yang selalu mendoakan dan memberikan dorongan baik material maupun spiritual.
3. Bapak Nurochman, S.Kom. M.Kom selaku dosen pembimbing yang telah memberikan banyak bimbingan dan bantuan dalam pelaksanaan dan penulisan laporan kerja praktek.
4. Bapak Agus Mulyanto, M. Kom., selaku Ketua Program Studi Teknik Informatika UIN Sunan Kalijaga atas segala dukungan dan bantuannya sehingga pelaksanaan kerja praktek dapat berjalan dengan lancar .

5. Bapak Wahyu Tri Kurniawan, S.T. selaku Direktur CV. Aman Sejahtera Computer yang telah memberikan banyak bantuan, saran, pengarahan dan pengalaman ilmunya selama pelaksanaan kerja praktek.
6. Bapak Vito selaku pembimbing lapangan yang telah memberikan banyak bantuan, saran, pengarahan dan pengalaman ilmunya selama pelaksanaan kerja praktek.
7. Teman-teman khususnya prodi Teknik Informatika angkatan 2008 yang telah banyak memberikan dukungan dan motivasi kepada penulis, sehingga penulis dapat menyelesaikan kerja praktek.
8. Semua pihak yang telah membantu penulis dalam pelaksanaan kerja praktek dan penulisan laporan kerja praktek ini.

Penulis menyadari masih banyak kekurangan dan kelemahan dalam pelaksanaan dan penulisan laporan kerja praktek ini. Oleh karena itu, kritik dan saran dari pembaca sangat penulis harapkan. Semoga laporan ini dapat bermanfaat bagi semua pihak yang membacanya. Amin.

*Wassalamualaikum wr.wb.*

Yogyakarta, 20 April 2012

## DAFTAR ISI

|                                    |     |
|------------------------------------|-----|
| HALAMAN JUDUL .....                | i   |
| LEMBAR PENGESAHAN .....            | ii  |
| KATA PENGANTAR .....               | iii |
| DAFTAR ISI .....                   | v   |
| DAFTAR GAMBAR .....                | ix  |
| DAFTAR TABEL .....                 | xii |
| BAB I PENDAHULUAN .....            | 1   |
| 1.1 Latar Belakang .....           | 1   |
| 1.2 Tujuan Kerja Praktek .....     | 2   |
| 1.3 Batasan Kerja Praktek .....    | 2   |
| 1.4 Manfaat Kerja Praktek .....    | 3   |
| BAB II TEMPAT KERJA PRAKTEK .....  | 4   |
| 2.1 Gambaran Umum Instansi .....   | 4   |
| 2.1.1 Tentang ASC .....            | 4   |
| 2.1.2 Prestasi Penghargaan .....   | 4   |
| 2.1.3 Management .....             | 5   |
| 2.1.4 Outlet ASC Computer .....    | 7   |
| 2.1.5 Bidang Usaha .....           | 7   |
| 2.1.6 Legalitas ASC Computer ..... | 9   |
| 2.1.7 Visi dan Misi .....          | 9   |
| 2.1.8 Core Values .....            | 10  |

|                                             |    |
|---------------------------------------------|----|
| 2.2 Struktur Organisasi Instansi .....      | 12 |
| BAB III LANDASAN TEORI .....                | 13 |
| 3.1 Sistem Informasi .....                  | 13 |
| 3.2 Basis Data .....                        | 13 |
| 3.2.1 Database Server dengan MySQL .....    | 15 |
| 3.2.2 Database MySQL dengan SQL .....       | 15 |
| 3.3 JAVA .....                              | 17 |
| 3.4 UML .....                               | 20 |
| 3.4.1 Usecase Diagram .....                 | 20 |
| 3.4.2 Class Diagram .....                   | 22 |
| 3.4.3 Sequence Diagram .....                | 24 |
| 3.4.4 Activity Diagram .....                | 26 |
| 3.5 Pengujian .....                         | 28 |
| 3.5.1 Teknik Pengujian .....                | 28 |
| 3.5.2 Dasar Pengujian Perangkat Lunak ..... | 28 |
| 3.5.3 Sasaran-sasaran Pengujian .....       | 28 |
| 3.5.4 Prinsip Pengujian .....               | 29 |
| 3.5.5 Testibilitas .....                    | 29 |
| 3.5.6 Pengujian Black Box .....             | 31 |
| BAB IV HASIL DAN PEMBAHASAN .....           | 32 |
| 4.1 Hasil .....                             | 32 |
| 4.1.1 Usecase Diagram .....                 | 32 |
| 4.1.1.1 Deskripsi Usecase .....             | 34 |

|        |                                          |    |
|--------|------------------------------------------|----|
| 4.1.2  | Activity Diagram .....                   | 42 |
| 4.1.3  | Sequence Diagram .....                   | 45 |
| 4.1.4  | Class Diagram .....                      | 56 |
| 4.1.5  | Design Tabel.....                        | 58 |
| 4.1.6  | Relati Tabel .....                       | 61 |
| 4.2    | Implementasi Sistem .....                | 61 |
| 4.2.1  | Installer Specification Maker .....      | 61 |
| 4.2.2  | Form Home.....                           | 62 |
| 4.2.3  | Form Setting .....                       | 62 |
| 4.2.4  | Form Edit.....                           | 63 |
| 4.2.5  | Form Kategori .....                      | 63 |
| 4.2.6  | Form Kurs Setting.....                   | 64 |
| 4.2.7  | Form Akun Setting.....                   | 64 |
| 4.2.8  | Form Simulasi .....                      | 65 |
| 4.2.9  | Form Detail Barang .....                 | 65 |
| 4.2.10 | Report Daftar Simulasi Spesifikasi ..... | 66 |
| 4.3    | Pengujian .....                          | 66 |
| 4.3.1  | Rencana Pengujian Alpha .....            | 66 |
| 4.3.2  | Kasus dan Hasil Pengujian Alpha .....    | 67 |
| 4.3.3  | Kesimpulan Pengujian Alpha .....         | 69 |
| 4.3.4  | Rencana Pengujian Beta .....             | 69 |
| 4.3.5  | Hasil Pengujian Beta .....               | 70 |
| 4.3.6  | Kesimpulan Pengujian Beta .....          | 71 |

|                                 |    |
|---------------------------------|----|
| BAB V PENUTUP .....             | 73 |
| 5.1 Kesimpulan .....            | 73 |
| 5.2 Saran .....                 | 73 |
| DAFTAR PUSTAKA .....            | 75 |
| LAMPIRAN 1 : Kode Program ..... | 76 |

## DAFTAR GAMBAR

|                                                                 |    |
|-----------------------------------------------------------------|----|
| Gambar 2.1 Struktur Organisasi CV Aman Sejahtera Computer ..... | 12 |
| Gambar 3.1 <i>Actor</i> .....                                   | 21 |
| Gambar 3.2 <i>Use Case</i> .....                                | 21 |
| Gambar 3.3 <i>Association</i> .....                             | 22 |
| Gambar 3.4 <i>Class</i> .....                                   | 22 |
| Gambar 3.5 <i>Association</i> .....                             | 23 |
| Gambar 3.6 <i>Composition</i> .....                             | 23 |
| Gambar 3.7 <i>Dependency</i> .....                              | 23 |
| Gambar 3.8 <i>Agregation</i> .....                              | 24 |
| Gambar 3.9 <i>Generalization</i> .....                          | 24 |
| Gambar 3.10 <i>Object</i> .....                                 | 24 |
| Gambar 3.11 <i>Actor</i> .....                                  | 25 |
| Gambar 3.12 <i>Lifeline</i> .....                               | 25 |
| Gambar 3.13 <i>Activation</i> .....                             | 25 |
| Gambar 3.14 <i>Message</i> .....                                | 25 |
| Gambar 3.15 <i>Action state</i> .....                           | 26 |
| Gambar 3.16 <i>Transition</i> .....                             | 26 |
| Gambar 3.17 <i>Initial state</i> .....                          | 26 |
| Gambar 3.18 <i>Final state</i> .....                            | 27 |
| Gambar 3.19 <i>Decision state</i> .....                         | 27 |
| Gambar 3.20 <i>Synchronization bar</i> .....                    | 27 |

|                                                             |    |
|-------------------------------------------------------------|----|
| Gambar 4.1 Usecase Diagram .....                            | 32 |
| Gambar 4.2 Usecase Diagram (Admin) .....                    | 33 |
| Gambar 4.3 Usecase Diagram (User) .....                     | 33 |
| Gambar 4.4 Activity Mengelola Data Barang .....             | 42 |
| Gambar 4.5 Activity Kategori .....                          | 43 |
| Gambar 4.6 Activity Kurs .....                              | 44 |
| Gambar 4.7 Activity Akun .....                              | 44 |
| Gambar 4.8 Activity Simulasi .....                          | 45 |
| Gambar 4.9 Sequence Diagram Login .....                     | 46 |
| Gambar 4.10 Sequence Diagram Menambah Data Barang .....     | 47 |
| Gambar 4.11 Sequence Diagram Mengubah Data B arang.....     | 48 |
| Gambar 4.12 Sequence Diagram Menghapus Data Barang .....    | 49 |
| Gambar 4.13 Sequence Diagram Menambah Data Kategori .....   | 50 |
| Gambar 4.14 Sequence Diagram Mengubah Data Kategori .....   | 51 |
| Gambar 4.15 Sequence Diagram Menghapus Data Kategori .....  | 52 |
| Gambar 4.16 Sequence Diagram Kurs .....                     | 53 |
| Gambar 4.17 Sequence Diagram Akun .....                     | 54 |
| Gambar 4.18 Sequence Diagram Membuat Daftar Simulasi .....  | 55 |
| Gambar 4.19 Sequence Diagram Mencetak Daftar Simulasi ..... | 56 |
| Gambar 4.20 Class Diagram.....                              | 56 |
| Gambar 4.21 Relasi Antar Tabel .....                        | 61 |
| Gambar 4.22 Installer Specification Maker.....              | 61 |
| Gambar 4.23 Form Home.....                                  | 62 |

|                                                      |    |
|------------------------------------------------------|----|
| Gambar 4.24 Form Setting .....                       | 62 |
| Gambar 4.25 Form Edit.....                           | 63 |
| Gambar 4.26 Form Kategori.....                       | 63 |
| Gambar 4.27 Form Kurs Setting.....                   | 64 |
| Gambar 4.28 Form Akun Setting.....                   | 64 |
| Gambar 4.29 Form Simulasi.....                       | 65 |
| Gambar 4.30 Form Detail Barang.....                  | 65 |
| Gambar 4.31 Report Daftar Simulasi Spesifikasi ..... | 66 |

## DAFTAR TABEL

|                                                                       |    |
|-----------------------------------------------------------------------|----|
| Tabel 4.1 Deskripsi Usecase Login .....                               | 34 |
| Tabel 4.2 Deskripsi Usecase Mengelola Barang .....                    | 35 |
| Tabel 4.3 Deskripsi Usecase Mengelola Kategori Barang .....           | 36 |
| Tabel 4.4 Deskripsi Usecase Menyetting Kurs .....                     | 38 |
| Tabel 4.5 Deskripsi Usecase Mengelola Password Admin .....            | 39 |
| Tabel 4.6 Deskripsi Usecase Mensimulasikan Spesifikasi Komputer ..... | 40 |
| Tabel 4.7 Tabel Akun.....                                             | 58 |
| Tabel 4.8 Tabel Barang .....                                          | 58 |
| Tabel 4.9 Tabel Jenis_Barang .....                                    | 59 |
| Tabel 4.10 Tabel PC_sup .....                                         | 59 |
| Tabel 4.11 Tabel Mem_sup .....                                        | 59 |
| Tabel 4.12 Tabel Uang .....                                           | 60 |
| Tabel 4.13 Tabel Rakit .....                                          | 60 |
| Tabel 4.14 Tabel Detail .....                                         | 60 |
| Tabel 4.15 Rencana Pengujian Alpha .....                              | 67 |
| Tabel 4.16 Hasil Pengujian Alpha .....                                | 67 |
| Tabel 4.17 Kuisisioner .....                                          | 70 |
| Tabel 4.18 Pertanyaan 1 .....                                         | 70 |
| Tabel 4.19 Pertanyaan 2 .....                                         | 70 |
| Tabel 4.20 Pertanyaan 3 .....                                         | 70 |
| Tabel 4.21 Pertanyaan 4 .....                                         | 71 |
| Tabel 4.22 Perhitungan Presentase .....                               | 71 |

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar belakang**

Yogyakarta merupakan salah satu kota terbesar di Indonesia. Kota pelajar juga merupakan sebutan bagi kota ini. Setiap tahunnya selalu ada pameran-pameran yang diselenggarakan, baik itu pameran buku, komputer, elektronik, makanan dan lain sebagainya. Pameran komputer merupakan pameran yang paling banyak menarik *animo* masyarakat. Pameran komputer yang sudah terjadwal tiap tahun seperti Mega Bazar Computer (MBC), Festival Komputer Indonesia (FKI), APKOM JOSS, YOGYAKOMTEK.

ASComputer sebagai salah satu toko komputer terbesar di Yogyakarta selalu berpartisipasi dalam setiap pameran komputer. *Stand* yang diikuti oleh ASComputer salah satunya adalah *Hardware* dan *Accesories*. Pameran komputer diadakan selama 5 hari, setiap harinya selama pameran, *stand* ASComputer dikunjungi oleh pengunjung pameran. Beberapa pengunjung ingin membuat spesifikasi komputer sesuai dengan apa yang mereka butuhkan. Mereka biasanya membuat spesifikasi dengan menuliskan di kertas yang disediakan oleh penjaga *stand*. Jika spesifikasi tidak sesuai, mungkin karena melebihi *budget* yang mereka miliki, maka mereka mencoba membuat spesifikasi kembali pada kertas yang baru. Hal ini menyebabkan kurang efektifnya waktu serta pemborosan kertas.

Oleh karena itu, dari persoalan tersebut di atas perlu dibuat suatu Sistem Informasi *Specification Maker* yang mana dengan Sistem Informasi ini akan

memudahkan pengunjung atau calon pembeli dalam membuat spesifikasi komputer sesuai dengan apa yang mereka butuhkan dan dapat langsung melihat hasil atau biaya yang dibutuhkan untuk merealisasikan spesifikasi komputer tersebut. Jika nantinya biaya yang dibutuhkan tidak sesuai dengan *budget*, maka pengunjung dapat langsung mengganti dengan *hardware* lain yang sesuai tanpa menulis ulang di kertas yang baru.

## 1.2 Tujuan Kerja Praktek

Berdasarkan latar belakang masalah yang ada, maka tujuan dari kerja praktek meliputi:

1. Membangun sistem informasi *Specification Maker* yang dapat memberikan informasi mengenai data barang pada *stand* ASComputer secara cepat dan mudah.
2. Membangun sistem informasi *Specification Maker* yang dapat memberikan panduan dalam membuat spesifikasi komputer tanpa didampingi oleh penjaga *stand* ASComputer.

## 1.3 Batasan Kerja Praktek

Batasan masalah dalam Kerja Praktek ini adalah sebagai berikut :

1. Sistem informasi *Specification Maker* diterapkan hanya pada satu komputer pada saat pameran berlangsung.
2. Sistem informasi *Specification Maker* tidak memperhatikan *stock* dari barang.

3. Sistem informasi *Specification Maker* tidak terkait dengan sistem informasi penjualan di *stand* ASComputer.

#### **1.4 Manfaat Kerja Praktek**

Manfaat dari pelaksanaan kerja praktek di ASComputer antara lain:

1. Memberikan layanan informasi kepada pengunjung pameran melalui media *Specification Maker* dengan cepat dan mudah.
2. Mempercepat pelayanan kepada pengunjung pameran untuk membuat spesifikasi komputer melalui media *Specification Maker*.
3. Membuat penulisan spesifikasi komputer menjadi lebih efisien dan *paperless*.

## **BAB II**

### **TEMPAT KERJA PRAKTEK**

#### **2.1 Gambaran Umum Instansi**

##### **2.1.1 Tentang ASC**

Aman Sejahtera Computer (ASC) adalah perusahaan komputer di Yogyakarta yang berdiri pada 1 Mei 2002. ASC adalah perusahaan komputer terdepan yang menerapkan konsep penjualan produk PC rakitan murah untuk konsumen dengan tujuan membentuk *image* ASC menjual barang berkualitas dengan harga terjangkau. Dengan strategi ini ASC telah berkembang menjadi 2 anak cabang di Yogyakarta hingga sekarang. ASC mengembangkan berbagai sektor bisnis IT melalui Penjualan *Retail Consumer*, Distribusi *Reseller*, *Maintenance* dan *Service*, Layanan Penjualan, *Software Development*, *Web Development*, Pengembangan *E-Commerce* yaitu penjualan secara *online* melalui situs [www.ascomputer.co.id](http://www.ascomputer.co.id) dan *Branded PC*.

##### **2.1.2 Prestasi Penghargaan**

Dengan semangat berinovasi dan komitmen untuk meraih predikat Terbaik dan Terdepan di Yogyakarta, ASC telah meraih berbagai penghargaan:

1. Best Achievement EPSON Preferred Reseller FY 2006
2. THE ESQ WAY Baktiku Guru Peduli Pendidikan 2 Yogyakarta 20 08

3. Best Seller Intel Core Duo E6X50 dan Intel Core 2 Quad Q6600 ICC 2008 Intel Indonesia Corporation se-Jawa Tengah dan DIY
4. Runner up Intel Motherboard Dekstop ICC 2008 Intel Indonesia Corporation se-Jawa Tengah dan DIY
5. Best Seller Intel Processor Over All ICC 2009 Intel Indonesia Corporation se-Jawa Tengah dan DIY
6. Best Seller Top 20 Memory Moduler Dealer in 2008
7. Best Seller Visipro Memory Moduler 2008 in Indonesia
8. Intel Technology Provider Platinum 2011 in 2010
9. Asus Gold Partner in 2011
10. Axioo Channel Partner Gold in 2011

### **2.1.3 Management**

#### **2.1.3.1 Kemampuan manajerial dan pemasaran**

ASC menyadari tanpa pelayanan yang baik dan profesionalisme kerja dari staf mereka, prosedur operasi dan pelaksanaannya tidak mungkin dapat memenuhi keinginan pelanggan. Untuk itu ASC selalu menjaga kekompakan tim dan melakukan beberapa strategi pemasaran seperti dengan mengembangkan berbagai sektor bisnis komputer melalui Penjualan *Retail Consumer*, Distribusi *Reseller*, *Maintenance Service*, layanan Purnajual, *Software Development*, *Web Design*, FastaWorld dan Pengembangan *E-Commerce* yaitu penjualan secara online melalui [www.ascomputer.co.id](http://www.ascomputer.co.id) dan juga membuka jalinan kerjasama dengan Perguruan Tinggi di Indonesia,

instansi pemerintah dan swasta sehingga diharapkan ASC semakin masyarakat luas.

Menjalin kerjasama dengan distributor dalam negeri dan luar negeri, sehingga ketersediaan produk terjamin sesuai dengan kebutuhan teknologi yang diperlukan dana akhirnya akan memberikan kenyamanan dan kepuasan kepada pemakai.

#### **2.1.3.2 Mengembangkan teknologi**

Dalam memberikan nilai lebih untuk kenyamanan pelanggan maka ASC ditangani oleh tenaga ahli yang terlatih yang selalu melakukan berbagai macam riset dalam melakukan berbagai pengembangan guna mewujudkan SDM yang profesional mencapai tujuan pengembangan berbagai macam *brand* yang akan diusung ASC.

#### **2.1.3.3 Pelatihan**

ASC selalu melakukan pelatihan untuk melatih Manajemen dan Staf, agar menghasilkan tenaga ahli yang mahir, handal dan berpengalaman, guna memastikan terwujudnya kenyamanan dalam berinteraksi dan menggunakan berbagai produk dari ASC.

#### **2.1.3.4 Memberikan Pelayanan Yang Terbaik**

Hal yang paling penting dalam pelayanan ASC adalah memastikan bahwa setiap pelanggan akan mendapatkan perhatian khusus serta pelayanan

yang profesional sehingga akan terwujudlah “Kepuasan anda adalah keberhasilan kami”.

#### **2.1.4 Outlet ASC Computer**

1. ASC Jakal

Jalan Kaliurang Km 7 Nomor 5

Yogyakarta

Telp: 0274-882 891, 882 817, 882 309

2. ASC Babarsari

Komplek Pertokoan Babarsari No. 69-71

Yogyakarta

Telp: 0274-433 2932, 433 2933, 485 126

3. Fasta (Komputer dan Notebook)

Jalan Kaliurang Km.05 No.22

Yogyakarta

Telp: 0274-543 363, 664 8901

#### **2.1.5 Bidang Usaha**

Dalam menjalankan roda bisnisnya, ASC Group memiliki beberapa bidang usaha yang bias dijelaskan sebagai berikut:

1. Penjualan

Melayani penjualan produk meliputi komputer dan *notebook* baik *hardware* ataupun *software* secara *retail* maupun mekanisme

*tender* baik secara langsung (*direct selling*) maupun penjualan secara *online*.

## 2. Pembelian

Melayani pembelian produk (purna jual) komputer dan *notebook* beserta asesorisnya dalam kondisi tertentu, secara *retail* maupun mekanisme partai besar dan kecil.

## 3. Perawatan

Melayani perawatan *hardware* dan *software* IT, personal maupun instansi baik swasta maupun pemerintah.

## 4. Jaringan IT

Melayani instalasi jaringan komputer pribadi maupun instansi dari skala kecil hingga profesional.

## 5. Software Development

Melayani pembuatan *software* Sistem Informasi untuk pribadi, instansi dan bisnis dari skala kecil hingga profesional.

## 6. Web Development

Melayani pembuatan *website* dan desain untuk pribadi, instansi dari skala kecil hingga profesional.

## 7. Branded PC

Menyediakan *Branded PC* Lokal dengan *brand* FastaWorld berikut dengan perangkat kerasnya. Adapun perangkat yang sudah ASC produksi meliputi *Desktop PC*, *Notebook*, *LCD monitor*, *Keyboard*

*Mouse, Casing, Power Supplay Unit (PSU), Stabilizer* dan berbagai asesoris komputer lainnya.

#### **2.1.6 Legalitas ASC Computer**

1. Akta pendirian CV Aman Sejahtera Computer Nomor 5 tahun 2009
2. Surat Pengukuhan Pengusaha Kena Pajak Nomor PEM - 02222/WPJ.23/KP.0103/2009
3. NPWP Nomor 02.683.375.6-542.000
4. Ijin Gangguan No. 503/4924/HO/2008
5. TDP Nomor 120255205224
6. SIUP Nomor 503/662/3381/PK/VI/2008

#### **2.1.7 Visi dan Misi**

##### **2.1.7.1 Visi**

Setiap orang senang dan puas membeli produk dari ASC

##### **2.1.7.2 Misi**

1. Memberikan layanan yang memuaskan dan solusi yang bernilai tambah kepada seluruh *customer*
2. Membangun Sumber Daya Manusia yang profesional untuk kesejahteraan umat manusia

3. Mengembangkan produk-produk inovatif di bidang teknologi informasi sesuai dengan kebutuhan pasar dengan harga terjangkau
4. Meningkatkan kepedulian dan tanggung jawab terhadap lingkungan social

### 2.1.8 Core Values

#### 2.1.8.1 SMART

1. *Senyum* : Kami berkomitmen untuk selalu tersenyum kepada siapapun, kapanpun dimanapun dan dalam kondisi apapun dengan harapan untuk memberikan kenyamanan
2. *Menarik* : Kami berkomitmen untuk selalu berpenampilan menarik kepada siapapun, kapanpun, dimanapun dan dalam kondisi apapun dengan harapan untuk memberikan rasa percaya diri bagi kami dan *customer* kami
3. *Antusias* : kami berkomitmen untuk antusias dan semangat kepada siapapun, kapanpun, dimanapun dan dalam kondisi apapun dengan harapan untuk memberikan kepuasan kepada *customer* kami
4. *Ramah*: Kami berkomitmen untuk selalu ramah kepada siapapun, kapanpun, dimanapun dan dalam kondisi apapun dengan harapan untuk menumbuhkan rasa menghargai dan menghormati kepada semua pihak

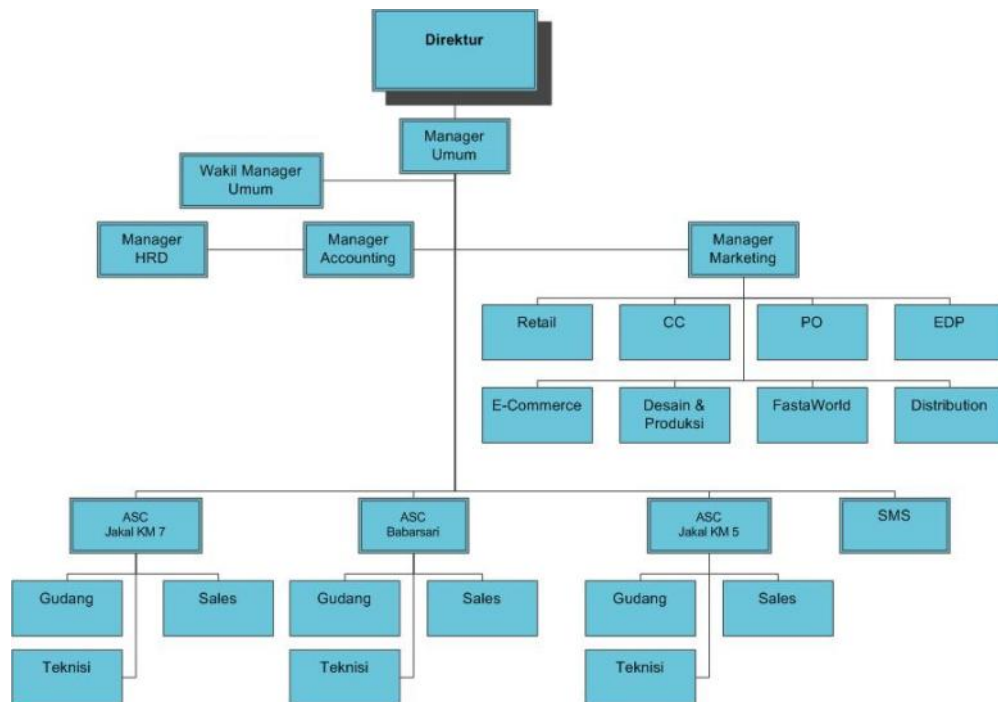
5. *Tanggap*: Kami selalu berkomitmen untuk selalu tanggap kepada siapapun, kapanpun, dimanapun dan dalam kondisi apapun dengan harapan untuk memberikan pelayanan yang cepat, tepat, akurat dan bermanfaat pada semua pihak

#### **2.1.8.2 FASTA**

1. *Fathonah* = cerdas
2. *Amanah* = bertanggung jawab
3. *Shiddiq* = jujur
4. *Tabligh* = menyampaikan kebenaran

## 2.2 Struktur Organisasi Instansi

### Struktur Organisasi CV AMAN SEJAHTERA COMPUTER



**Gambar 2.1** Struktur Organisasi CV Aman Sejahtera Computer

## **BAB III**

### **LANDASAN TEORI**

#### **3.1 Sistem Informasi**

Ada beberapa definisi mengenai sistem informasi. Sistem informasi adalah suatu sistem yang saling berintegrasi dan berinteraksi atau berelasi satu sama lain, secara teratur, baik, dan rapi sehingga sistem dapat memberikan informasi yang bermanfaat yang dibutuhkan oleh pengguna dalam mengambil keputusan pada saat ini ataupun untuk masa mendatang (Supriyanto, 2005). Menurut Hall (2001) sistem informasi merupakan sebuah rangkaian prosedur formal dimana data dikelompokkan, diproses menjadi informasi, dan didistribusikan kepada pemakai. Ada juga yang mendefinisikan sistem informasi sebagai kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi (Alter, 1992). Dari berbagai definisi tersebut, dapat disimpulkan bahwa sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan (Kadir, 2003).

#### **3.2 Basis Data**

Basis data didefinisikan sebagai suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk

memperoleh informasi dan mengatasi problem pada sistem yang memakai pendekatan berbasis berkas (Kadir, 2003).

Pada pemrosesan data yang menggunakan pendekatan berbasis berkas, ada kemungkinan bahwa perancangan sistem masih didasarkan pada kebutuhan individu si pemakai, bukan berdasarkan kebutuhan sejumlah pemakai. Setiap kali terdapat kebutuhan baru dari seorang pemakai, kebutuhan segera diterjemahkan ke dalam program komputer. Akibatnya, setiap program aplikasi menuliskan data tersendiri yang sebetulnya ada kemungkinan data yang sama juga terdapat pada berkas - berkas lain yang digunakan oleh program aplikasi lain. Ini merupakan salah satu kelemahan sistem berbasis berkas, yaitu memungkinkan terjadinya duplikasi data. Di samping itu, kelemahan sistem berbasis berkas lainnya adalah dalam hal dependensi data program dimana perubahan dalam data membuat program juga harus diubah. Kelemahan ini dapat diperbaiki dalam sistem pemrosesan berbasis data yang memiliki kelebihan dalam hal independensi data yang memungkinkan sebuah data tidak tergantung dengan data lainnya. Hal ini mencegah terjadinya perubahan pada program jika struktur berkas diubah ataupun sebaliknya.

Untuk mengelola basis data diperlukan sebuah perangkat lunak yang disebut DBMS (*Data Base Management System*). DBMS adalah perangkat lunak sistem yang memungkinkan pemakai membuat, memelihara, mengontrol, dan mengakses basis data dengan cara yang praktis dan efisien (Kadir, 2003). Beberapa contoh DBMS yang terkenal

yaitu Access (Microsoft Corporation), Sybase (Sybase Inc.), MySQL (The MySQL AB Company), dan Oracle (Oracle Corporation).

### 3.2.1 Database Server dengan MySQL

MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta mampu menggunakan perintah standar SQL ( *Structured Query Language* ). MySQL selain sebagai *database server*, dapat juga berperan sebagai *client* sehingga sering disebut *database client-server*, yang *open source* dengan kemampuan dapat berjalan baik di OS ( *Operating System* ) manapun, dengan *platform Windows* maupun LINUX (Nugroho, 2004).

MySQL memiliki beberapa kelebihan dibanding dengan DBMS lainnya, yaitu :

- a. MySQL sebagai *Relation Database Management System* (RDBMS).
- b. MySQL adalah sebuah *software* yang *open source* artinya program ini bersifat *free* atau bebas digunakan oleh siapa saja tanpa harus membeli dan membayar lisensi kepada pembuatnya.
- c. MySQL mampu menerima *query* yang bertumpuk dalam satu permintaan atau yang disebut *multi-threading*.
- d. MySQL memiliki kecepatan dalam pembuatan tabel maupun peng - *update*-an tabel.

### 3.2.2 Database MySQL dengan SQL

MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user*

serta mampu menggunakan perintah standar SQL (*Structured Query Language*). MySQL selain sebagai *database server*, dapat juga berperan sebagai *client* sehingga sering disebut *database client-server*.

Menurut Kadir (2003), SQL singkatan dari *Structured Query Language* adalah bahasa yang digunakan untuk mengakses basis data yang tergolong relasional. SQL dibuat sebagai bahasa yang dapat merelasikan beberapa tabel dalam *database* maupun merelasikan antar *database* (Nugroho, 2004). Pada dasarnya, SQL tidak terbatas hanya untuk mengambil data (*query*), tetapi juga dapat digunakan untuk menciptakan tabel, menghapus tabel, menambahkan data ke tabel, menghapus data pada tabel, mengganti data pada tabel, dan berbagai operasi lainnya.

SQL dibagi menjadi dua bentuk *query*, yaitu :

1. DDL (*Data Definition Language*)

DDL adalah suatu bahasa yang digunakan untuk mendefinisikan data pada sebuah basis data. Perintah yang termasuk DDL, yaitu :

- a. CREATE : digunakan untuk membuat sebuah tabel atau *database*
- b. DROP : digunakan untuk menghapus sebuah tabel atau *database*

- c. ALTER : digunakan untuk mengubah struktur tabel yang sudah dibuat. Pengubahan struktur bisa dalam hal penambahan kolom (ADD), pengubahan lebar dan jenis kolom (MODIFY), penghapusan kolom dan indeks (DROP), penggantian nama kolom (CHANGE), penggantian nama tabel (RENAME), dan sebagainya.

## 2. DML (*Data Manipulation Language*)

DML adalah suatu bahasa yang digunakan untuk memanipulasi data pada sebuah basis data. Perintah yang termasuk DML, yaitu :

- a. INSERT : digunakan untuk melakukan penambahan atau pemasukan data pada tabel dari sebuah *database*
- b. UPDATE : digunakan untuk melakukan pengubahan terhadap data yang ada pada tabel
- c. DELETE : digunakan untuk melakukan penghapusan data pada tabel
- d. SELECT : dilakukan untuk melihat data yang ada di dalam satu atau beberapa tabel

### 3.3 JAVA

Sebagai sebuah bahasa pemrograman, Java dapat membuat seluruh bentuk aplikasi, *desktop*, *web* dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain.

Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai platform sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat *open source*.

Terdapat dua komponen utama dari *Deployment Environment*. Yang pertama adalah JRE (*Java Runtime Environment*), mengandung kelas-kelas untuk semua paket teknologi Java yang meliputi kelas dasar dari Java, komponen GUI (*Graphical User Interface*) dan sebagainya. Komponen yang lain terdapat pada *Web Browser*. Hampir seluruh *Web Browser* komersial menyediakan interpreter dan *runtime environment* dari teknologi Java.

Berdasarkan *white paper* resmi dari SUN, Java memiliki karakteristik berikut :

#### 1. Sederhana

Bahasa pemrograman Java menggunakan *sintaks* mirip dengan C++ namun *sintaks* pada Java telah banyak diperbaiki terutama menghilangkan penggunaan *pointer* yang rumit dan multiple inheritance. Java juga menggunakan *automatic memory allocation* dan *memory garbage collection*.

#### 2. Berorientasi objek (*Object Oriented*)

Java menggunakan pemrograman berorientasi objek yang membuat program dapat dibuat secara modular dan dapat dipergunakan kembali. Pemrograman berorientasi objek memodelkan dunia nyata ke dalam objek dan melakukan interaksi antar objek-objek tersebut.

### 3. Dapat di distribusi dengan mudah

Java dibuat untuk membuat aplikasi terdistribusi secara mudah dengan adanya *libraries networking* yang terintegrasi pada Java.

### 4. Interpreter

Program Java dijalankan menggunakan interpreter yaitu *Java Virtual Machine* (JVM). Hal ini menyebabkan *source code* Java yang telah di kompilasi menjadi Java *bytecodes* dapat dijalankan pada platform yang berbeda-beda.

### 5. Robust

Java mempunyai *reliabilitas* yang tinggi. Compiler pada Java mempunyai kemampuan mendeteksi *error* secara lebih teliti dibandingkan bahasa pemrograman lain. Java mempunyai *runtime-Exception handling* untuk membantu mengatasi *error* pada pemrograman.

### 6. Aman

Sebagai bahasa pemrograman untuk aplikasi internet dan terdistribusi, Java memiliki beberapa mekanisme keamanan untuk menjaga aplikasi tidak digunakan untuk merusak sistem komputer yang menjalankan aplikasi tersebut.

### 7. Architecture Neutral

Program Java merupakan platform *independent*. Program cukup mempunyai satu buah versi yang dapat dijalankan pada platform yang berbeda dengan *Java Virtual Machine*.

### 8. Portabel

*Source code* maupun program Java dapat dengan mudah dibawa ke platform yang berbeda-beda tanpa harus di kompilasi ulang.

#### 9. *Performance*

*Performance* pada Java sering dikatakan kurang tinggi. Namun *performance* Java dapat ditingkatkan menggunakan kompilasi Java lain seperti buatan Inprise, Microsoft ataupun Symantec yang menggunakan Just In Time Compilers (JIT).

#### 10. *Multithreaded*

Java mempunyai kemampuan untuk membuat suatu program yang dapat melakukan beberapa pekerjaan secara sekaligus dan simultan.

#### 11. Dinamis

Java didesain untuk dapat dijalankan pada lingkungan yang dinamis. Perubahan pada suatu *class* dengan menambahkan *properties* ataupun *method* dapat dilakukan tanpa mengganggu program yang menggunakan *class* tersebut.

### 3.4 UML (*Unified Modeling Language*)

*Unified Modelling Language* (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem peranti lunak.

UML sesuai untuk penulisan peranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET dapat juga digunakan untuk modeling aplikasi *prosedural* dalam VB atau C.

#### 3.4.1 Use Case Diagram

Diagram ini memperlihatkan himpunan *Use Case* dan *actor-actor*. Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.

### 1. Aktor

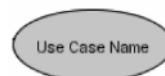
Aktor adalah pengguna sistem. Peran pengguna tertulis di bawah ikon. Pelaku tidak terbatas pada manusia. Jika suatu sistem berkomunikasi dengan aplikasi lain, dan akan memberikan masukan atau keluaran, maka aplikasi tersebut juga dapat dianggap sebagai aktor. Aktor dapat dilihat pada gambar 3.1.



Gambar 3.1 *Actor*

### 2. *Use Case*

*Use Case* adalah fungsi yang disediakan oleh sistem. *Use Case* digambarkan dengan elips. Nama *Use Case* ditulis dalam elips seperti pada gambar 3.2.



Gambar 3.2 *Use Case*

### 3. Association

Asosiasi digunakan untuk menghubungkan Aktor dengan *Use Case*, dan menunjukkan bahwa sebuah Aktor berpartisipasi di dalam

beberapa *form Use Case*. Asosiasi yang digambarkan oleh garis menghubungkan Aktor dan *Uses Case* seperti pada gambar 3.3.



Gambar 3.3 *Association*

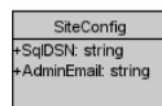
### 3.4.2 Class Diagram

*Class* diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain (Dharwiyanti & Wahono, 2003).

Diagram ini memperlihatkan himpunan kelas, antarmuka, kolaborasi serta relasi.

#### 1. *Class*

Kelas adalah bangunan blok di pemrograman berorientasi objek. Kelas digambarkan menggunakan persegi panjang dibagi menjadi tiga bagian. Bagian atas adalah nama Kelas. Bagian tengah mendefinisikan properti dari Kelas. Di bagian bawah daftar metode kelas seperti pada gambar 3.4..

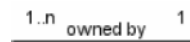


Gambar 3.4 *Class*

#### 2. *Association*

Asosiasi adalah hubungan umum antara dua kelas, dan dimodelkan oleh baris menghubungkan dua kelas. Baris ini bisa memenuhi syarat dengan jenis hubungan, dan juga dapat fitur beragam

aturan (misalnya, satu ke satu, satu ke banyak, banyak ke banyak) untuk hubungan seperti pada gambar 3.5.



Gambar 3.5 *Association*

### 3. *Composition*

Jika satu kelas tidak dapat berdiri dengan sendirinya, melainkan harus menjadi anggota kelas lain, maka kelas tersebut memiliki hubungan Komposisi dengan kelas yang lain. Komposisi hubungan itu ditunjukkan dengan garis yang diisi dengan berlian seperti pada gambar 3.6.



Gambar 3.6 *Composition*

### 4. *Depedency*

Bila menggunakan kelas kelas yang lain, mungkin sebagai anggota variabel atau parameter, dan sebagainya "tergantungan" di kelas itu, sebuah hubungan ketergantungan dibentuk. Ketergantungan hubungan itu ditunjukkan dengan panah garis putus-putus seperti pada gambar 3.7.



Gambar 3.7 *Depedency*

### 5. *Aggregation*

*Aggregations* menunjukkan seluruh bagian-hubungan, dan dikenal sebagai "memiliki-sebuah" hubungan. Penggabungan sebuah

hubungan itu ditunjukkan dengan garis dengan rongga berlian seperti pada gambar 3.8.



Gambar 3.8 *Aggregation*

#### 6. *Generalization*

Hubungan generalisasi adalah sama dengan sebuah warisan hubungan berorientasi objek. Hubungan generalisasi ditunjukkan oleh sebuah panah dengan panah yang berongga pada, atau "orang tua", kelas seperti pada gambar 3.9.



Gambar 3.9 *Generalization*

### 3.4.3 Sequence Diagram

Diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu (Nugroho,2005).

#### 1. *Object*

Objek adalah contoh dari kelas, dan dapat diatur horizontal. Yang menggambarkan perwakilan untuk Objek adalah kotak dengan nama seperti pada gambar 3.10.



Gambar 3.10 *Object*

#### 2. *Actor*

Aktor juga dapat berkomunikasi dengan objek, sehingga mereka juga dapat terdaftar sebagai kolom. Aktor menggunakan simbol seperti pada gambar 3.11.



Gambar 3.11 *Actor*

### 3. *Lifeline*

*Lifeline* yang mengidentifikasi keberadaan objek dari waktu ke waktu seperti pada gambar 3.12.



Gambar 3.12 *Lifeline*

### 4. *Activation*

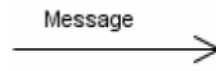
*Activations* sebagai model kotak segi empat di atas *lifeline*, menunjukkan bila objek melakukan tindakan seperti pada gambar 3.13.



Gambar 3.13 *Activation*

### 5. *Message*

Pesan, dimodelkan panah horisontal antara aktifitas, menunjukkan komunikasi antara objek. *Message* dapat dilihat pada gambar 3.14

Gambar 3.14 *Message*

### 3.4.4 Activity Diagram

Diagram aktivitas ini terutama penting dalam pemodelan fungsi - fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek.

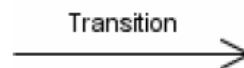
#### 1. *Activity State*

Aktivitas ditandai tindakan oleh obyek. Notasi untuk keadaan ini adalah bulatan seperti pada gambar 3.15.

Gambar 3.15 *Action State*

#### 2. *Transition*

Bila aktivitas sudah selesai, proses berpindah ke aktivitas yang lain. Transisi digunakan untuk menandai gerakan ini. Transisi adalah model menggunakan panah seperti pada gambar 3.16.

Gambar 3.16 *Transition*

#### 3. *Initial State*

*Initial State* menandai awal masuk dan titik awal. Notasinya lingkaran yang penuh seperti pada gambar 3.17. Hanya ada satu *Initial State* pada diagram.

Gambar 3.17 *Initial State*

#### 4. *Final State*

*Final State* menandai akhir model alur kerja. Ada beberapa *Final State* pada diagram, dan menggunakan model lingkaran yang penuh dikelilingi oleh lingkaran lain seperti pada gambar 3.18.

Gambar 3.18 *Final State*

#### 5. *Decision State*

*Decision state* menandai pencabangan kondisi dari suatu proses. Notasinya seperti pada gambar 3.19.

Gambar 3.19 *Decision State*

#### 6. *Synchronization Bar*

Kegiatan sering dapat dilakukan secara paralel. Untuk memecah proses ("*fork*"), atau untuk melanjutkan pemrosesan ketika beberapa kegiatan telah selesai ("*join*"), *Sinkronisasi Bar* digunakan. Model ini adalah sebagai Kotak penuh, dengan beberapa transisi yang terjadi di dalam dan / atau keluar, seperti pada gambar 3.20.

Gambar 3.20 *Synchronization Bar*

### **3.5 Pengujian**

#### **3.5.1 Teknik Pengujian Perangkat Lunak**

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain, dan pengkodean.

#### **3.5.2 Dasar Pengujian Perangkat Lunak**

Pengujian menyajikan anomali yang menarik bagi perekayasa perangkat lunak. Pada proses perangkat lunak, perekayasa pertama-tama berusaha membangun perangkat lunak dari konsep abstrak ke implementasi yang dapat dilihat, baru dilakukan pengujian. Perekayasa menciptakan sederetan test case yang dimaksudkan untuk “membongkar” perangkat lunak yang sudah dibangun.

Pada dasarnya pengujian merupakan salah satu langkah dalam proses rekayasa perangkat lunak yang dianggap sebagai hal yang destruktif daripada konstruktif.

#### **3.5.3 Sasaran-sasaran Pengujian**

Dalam buku klasiknya mengenai pengujian perangkat lunak, Glen Myers [MYE79] menyatakan sejumlah aturan yang berfungsi sebagai sasaran pengujian:

1. Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
2. Test case yang baik adalah test case yang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum ditemukan sebelumnya.
3. Pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya.

#### **3.5.4 Prinsip Pengujian**

Sebelum mengaplikasikan metode untuk mendesain test case yang efektif, perekayasa harus memahami prinsip dasar yang menuntun pengujian perangkat lunak. Davis [DAV95] mengusulkan serangkaian prinsip-prinsip pengujian diantaranya:

1. Semua pengujian harus dapat ditelusuri sampai ke persyaratan pelanggan.
2. Pengujian harus direncanakan lama sebelum pengujian itu mulai.
3. Prinsip pareto berlaku untuk pengujian perangkat lunak.
4. Pengujian harus mulai dari yang kecil dan yang berkembang kepengujian yang lebih besar.
5. Pengujian yang mendalam tidak mungkin.
6. Untuk menjadi paling efektif pengujian harus dilakukan oleh pihak ketiga yang independen.

### 3.5.5 Testibilitas

*Testibilitas* perangkat lunak adalah seberapa mudah program komputer dapat diuji. Karena pengujian sulit, maka perlu diketahui apa yang harus dilakukan agar menjadi lebih mudah. *Cheklis* berikut ini memberikan serangkaian karakteristik yang membawa perangkat lunak yang dapat diuji.

1. Operabilitas, “semakin baik dia bekerja, semakin efisien dia diuji”.
2. Observabilitas, “apa yang anda lihat adalah apa yang anda uji”.
3. Kontrabilitas, semakin baik kita dapat mengontrol perangkat lunak semakin banyak pengujian yang diotomatisasi dan dioptimalkan”.
4. Dekomposabilitas, “dengan mengontrol ruang lingkup pengujian, kita dapat lebih cepat mengisolasi masalah dan melakukan pengujian kembali secara lebih halus”.
5. Kesederhanaan, “semakin cepat yang diuji, semakin sedikit kita dapat mengujinya”.
6. Stabilitas, “semakin sedikit perubahan, semakin gangguan dalam pengujian”.
7. Verifikasi, mengacu kepada rangkaian aktivitas yang memastikan bahwa perangkat lunak secara tepat mengimplementasikan suatu fungsi tertentu.

8. Validasi, mengacu pada rangkaian aktivitas berbeda yang memastikan bahwa perangkat lunak yang dibangun dapat ditelusuri kepersyaratan pelanggan. “Apakah kita membangun produk yang benar”.

### 3.5.6 Pengujian Black Box

Pengujian *Black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi *input* yang sepenuhnya semua persyaratan fungsional untuk suatu program.

Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *Interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

## BAB IV

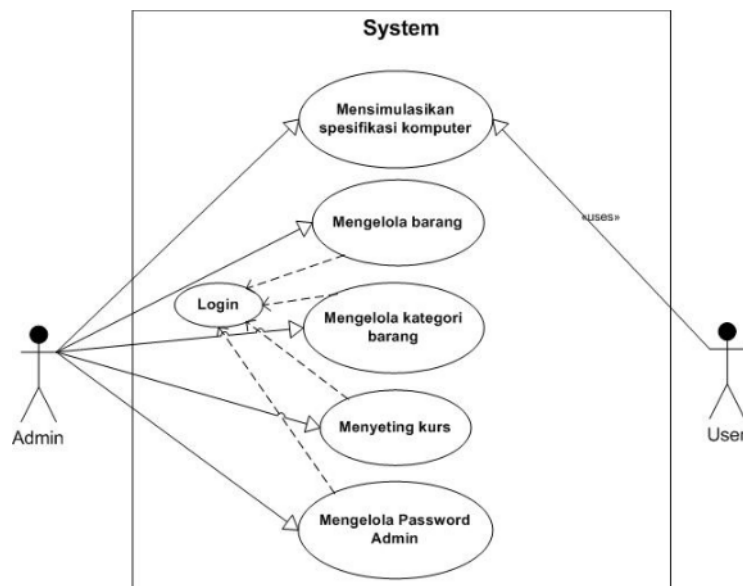
### HASIL DAN PEMBAHASAN

#### 4.1 Hasil

Untuk membuat aplikasi, pemodelan sistem merupakan suatu gambaran yang menjelaskan bagaimana sebuah sistem berjalan, dari mana sistem mendapatkan *input* data, apa saja proses yang terjadi, bagaimana data disajikan, dan dimana data disimpan. Dalam pengembangan aplikasi ini digunakan konsep OOP, oleh karenanya pemodelan sistem menggunakan metode UML.

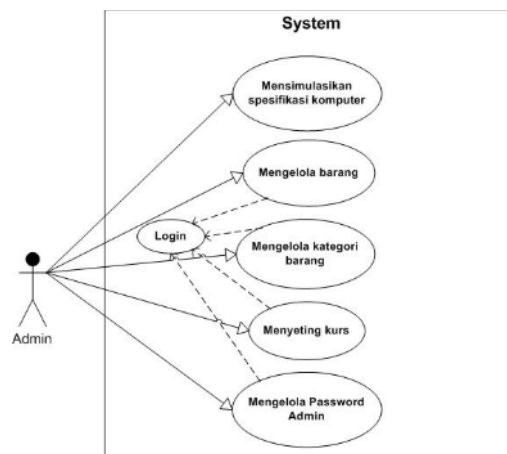
##### 4.1.1 Usecase Diagram

Fungsionalitas dari sistem informasi *Specification Maker* yang mempresentasikan interaksi antara aktor (*Admin* dan pengunjung pameran) dengan sistem dapat dilihat pada gambar 4.1



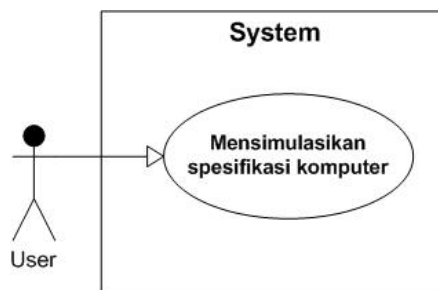
Gambar 4.1 Usecase diagram

Dari gambar 4.1 dapat disimpulkan terdapat dua aktor dalam sistem informasi *Specification Maker* ini, yaitu *Admin* dan *User*. *Admin* dalam hal ini adalah karyawan ASComputer, sedangkan *User* adalah pengunjung pameran. *Admin* dan *User* dapat mengakses simulasi dari sistem informasi *Specification Maker* secara langsung. *Admin* untuk mengakses Edit, Kategori, Kurs dan Akun harus melakukan proses *Login* terlebih dahulu.



**Gambar 4.2** Usecase diagram (Admin)

Gambar 4.2 menggambarkan tentang *usecase* dari *Admin*.



**Gambar 4.3** Usecase diagram (User)

Gambar 4.3 menggambarkan tentang *usecase* dari *User* yang hanya dapat mengakses simulasi dari sistem.

#### 4.1.1.1 Deskripsi *Usecase*

Berikut ini adalah penjelasan detail mengenai skenario dari masing-masing *usecase* melalui beberapa tabel deskripsi. Masing-masing tabel terdiri :

1. Skenario yang merupakan alur *usecase*.
2. *Brief description* adalah rincian deskripsi dari *usecase*
3. Actor menyatakan actor yang terlibat didalam *usecase*.
4. *Related Usecase* adalah *usecase* yang berhubungan dengan *usecase* tersebut.
5. *Precondition* adalah kondisi awal yang harus ada sebelum skenario dijalankan.
6. *Post Condition* adalah kondisi ketika skenario telah dijalankan.
7. *Flow of Event* adalah rangkaian kegiatan yang dilakukan aktor dan sistem yang terlibat.
8. *Exception Condition* adalah kondisi pengecualian yang diberikan ketika sistem tidak berjalan sesuai skenario yang diberikan.

Tabel 4.1 Deskripsi *Usecase Login*

|                     |                                                                                                         |
|---------------------|---------------------------------------------------------------------------------------------------------|
| <i>Usecase Name</i> | <i>Login</i>                                                                                            |
| Skenario            | <i>Login ke Sistem Informasi Specification Maker</i>                                                    |
| Brief Description   | <i>Admin memasukkan Username dan password pada form Login. Sistem melakukan pengecekan Username dan</i> |

|                        |                                                                                                                                 |                                                                                 |
|------------------------|---------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
|                        | <i>password</i> yang dimasukkan dengan data yang tersimpan di <i>database</i> .                                                 |                                                                                 |
| Actors                 | <i>Admin</i>                                                                                                                    |                                                                                 |
| Related <i>Usecase</i> | -                                                                                                                               |                                                                                 |
| Stakeholder            | Karyawan ASComputer                                                                                                             |                                                                                 |
| Precondition           | Data <i>Admin</i> harus sudah ada, didalam sistem sudah tersimpan <i>Admin</i> default yang tersimpan dan tidak dapat dihapus.  |                                                                                 |
| PostCondition          | Sistem Informasi <i>Specification Maker</i> akan terbuka                                                                        |                                                                                 |
| Flows of Events        | Actors                                                                                                                          | System                                                                          |
|                        | <i>Admin</i> membuka Sistem Informasi <i>Specification Maker</i><br><i>Admin</i> memasukkan <i>Username</i> dan <i>password</i> | Sistem melakukan pengecekan <i>Username</i> dan <i>password</i> yang dimasukkan |
| Exception Condition    | -                                                                                                                               |                                                                                 |

Tabel 4.1 menjelaskan tentang alur *usecase* untuk proses *login*.

Proses dimulai dari *Admin* memasukkan *username* dan *password*, setelah itu sistem akan melakukan pemeriksaan terha dap *username* dan *password* yang dimasukkan.

Tabel 4.2 Deskripsi *Usecase* Mengelola Barang

|                     |                                                                                                                                                                                                                               |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Usecase</i> Name | Mengelola barang                                                                                                                                                                                                              |
| Skenario            | <ul style="list-style-type: none"> <li>- Menambah data barang</li> <li>- Mengubah data barang</li> <li>- Menghapus data barang</li> </ul>                                                                                     |
| Brief Description   | <ul style="list-style-type: none"> <li>- <i>Admin</i> membuka <i>form</i> edit data barang</li> <li>- Sistem menampilkan data barang sesuai dengan kategori</li> <li>- <i>Admin</i> menekan tombol insert, mengisi</li> </ul> |

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                                                                                                                                                                                                                     |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                 | <p>data barang, kemudian menekan tombol <i>save</i></p> <ul style="list-style-type: none"> <li>- Sistem akan menambahkan data barang tersebut ke dalam <i>database</i></li> <li>- <i>Admin</i> memilih data barang pada tabel data, kemudian klik kanan <i>update</i> atau menekan tombol <i>update</i>. <i>Admin</i> mengubah data barang kemudian menekan tombol <i>save</i>.</li> <li>- Sistem akan menyimpan hasil perubahan data ke dalam <i>database</i></li> <li>- <i>Admin</i> memilih data barang pada tabel data, kemudian klik kanan <i>delete</i> atau menekan tombol <i>delete</i>.</li> <li>- Sistem akan menghapus data barang yang dipilih oleh <i>Admin</i></li> </ul> |                                                                                                                                                                                                                                     |
| Actors          | <i>Admin</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                                     |
| Related Usecase | <i>Login</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                                                                                                                                                                                                                     |
| Stakeholder     | Karyawan ASComputer                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                     |
| Precondition    | Data barang yang ada dalam <i>database</i> sudah ditampilkan.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                                                     |
| PostCondition   | <p>Data barang bertambah</p> <p>Data barang berkurang</p> <p>Data barang berubah</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                                                     |
| Flows of Events | Actors                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | System                                                                                                                                                                                                                              |
|                 | <ol style="list-style-type: none"> <li>1. <i>Admin</i> membuka <i>form</i> edit data barang</li> <li>2. <i>Admin</i> menambah data barang</li> <li>3. <i>Admin</i> mengubah data barang</li> <li>4. <i>Admin</i> menghapus data barang</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                                                                       | <ol style="list-style-type: none"> <li>1. Sistem menampilkan data barang</li> <li>2. Sistem menyimpan data barang baru</li> <li>3. Sistem menyimpan hasil perubahan data barang</li> <li>4. Sistem menghapus data barang</li> </ol> |

|                     |   |
|---------------------|---|
| Exception Condition | - |
|---------------------|---|

Tabel 4.2 menjelaskan tentang *usecase* mengelola barang.

*Admin* dapat merubah, menambah serta menghapus data barang dari sistem.

Tabel 4.3 Deskripsi *Usecase* Mengelola Kategori barang

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Usecase</i> Name    | Mengelola Kategori Barang                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Skenario               | <ul style="list-style-type: none"> <li>- Menambah data kategori</li> <li>- Mengubah data kategori</li> <li>- Menghapus data kategori</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| Brief Description      | <ul style="list-style-type: none"> <li>- <i>Admin</i> membuka <i>form</i> kategori</li> <li>- Sistem menampilkan data kategori sesuai dengan <i>item</i>.</li> <li>- <i>Admin</i> menekan tombol insert, mengisi data kategori, kemudian menekan tombol <i>save</i></li> <li>- Sistem akan menambahkan data kategori tersebut ke dalam <i>database</i></li> <li>- <i>Admin</i> memilih data kategori pada tabel kategori, kemudian klik kanan <i>update</i> atau menekan tombol <i>update</i>. <i>Admin</i> mengubah data kategori kemudian menekan tombol <i>save</i>.</li> <li>- Sistem akan menyimpan hasil perubahan data ke dalam <i>database</i></li> <li>- <i>Admin</i> memilih data kategori pada tabel kategori, kemudian klik kanan <i>delete</i> atau menekan tombol <i>delete</i>.</li> <li>- Sistem akan menghapus data kategori yang dipilih oleh <i>Admin</i></li> </ul> |
| Actors                 | <i>Admin</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Related <i>Usecase</i> | <i>Login</i>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Stakeholder            | Karyawan ASComputer                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Precondition           | Data kategori yang ada dalam <i>database</i> sudah ditampilkan.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| PostCondition          | Data kategori bertambah<br>Data kategori berkurang<br>Data kategori berubah                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

| Flows of Events     | Actors                                                                                                                                                                           | System                                                                                                                                                                  |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                     | 1. <i>Admin</i> membuka <i>form</i> edit kategori<br>2. <i>Admin</i> menambah data kategori<br>3. <i>Admin</i> mengubah data kategori<br>4. <i>Admin</i> menghapus data kategori | 1. Sistem menampilkan data kategori<br>2. Sistem menyimpan data kategori baru<br>3. Sistem menyimpan hasil perubahan data kategori<br>4. Sistem menghapus data kategori |
| Exception Condition | -                                                                                                                                                                                |                                                                                                                                                                         |

Tabel 4.3 menjelaskan tentang *usecase* mengelola kategori barang. Pada tabel ini *Admin* dapat merubah, menambah serta menghapus data kategori dari sistem.

Tabel 4.4 Deskripsi *Usecase* Menyeting Kurs

|                        |                                                                                                                                                                                                                                                                                        |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Usecase</i> Name    | Menyeting Kurs                                                                                                                                                                                                                                                                         |
| Skenario               | - Mengubah nilai mata uang dollar                                                                                                                                                                                                                                                      |
| Brief Description      | - <i>Admin</i> membuka <i>form</i> kurs<br>- Sistem akan menampilkan nilai mata uang dolar terakhir<br>- <i>Admin</i> memasukkan nilai mata uang dolar terbaru, kemudian menekan <i>save</i><br>- Sistem menyimpan nilai mata uang dolar terbaru, kemudian menampilkan hasil perubahan |
| Actors                 | <i>Admin</i>                                                                                                                                                                                                                                                                           |
| Related <i>Usecase</i> | <i>Login</i>                                                                                                                                                                                                                                                                           |
| Stakeholder            | Karyawan ASComputer                                                                                                                                                                                                                                                                    |

|                     |                                                                                                          |                                                                                                                                                                       |
|---------------------|----------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Precondition        | Data nilai mata uang dollar sudah ditampilkan                                                            |                                                                                                                                                                       |
| PostCondition       | Data nilai mata uang dolar berubah                                                                       |                                                                                                                                                                       |
| Flows of Events     | Actors                                                                                                   | System                                                                                                                                                                |
|                     | 1. <i>Admin</i> membuka <i>form</i> kurs<br><br>2. <i>Admin</i> memasukkan nilai mata uang dolar terbaru | 1. Sistem menampilkan nilai mata uang dolar terakhir<br>2. Sistem menyimpan data nilai mata uang dolar terbaru<br>3. Sistem menampilkan nilai mata uang dolar terbaru |
| Exception Condition | -                                                                                                        |                                                                                                                                                                       |

Tabel 4.4 menjelaskan mengenai *usecase* menyeting kurs.

Bagaimana cara *Admin* merubah nilai dolar dijelaskan pada *usecase* ini.

Tabel 4.5 Deskripsi *Usecase* Mengelola *password Admin*

|                        |                                                                                                                                                                                                                                                                                                          |        |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| <i>Usecase Name</i>    | Mengelola <i>password Admin</i>                                                                                                                                                                                                                                                                          |        |
| Skenario               | - Mengubah <i>password</i>                                                                                                                                                                                                                                                                               |        |
| Brief Description      | - <i>Admin</i> membuka <i>form</i> setting akun<br>- Sistem akan menampilkan <i>form</i> setting akun<br>- <i>Admin</i> memasukkan <i>Username</i> , <i>password</i> lama, <i>password</i> baru dan konfirmasi <i>password</i> baru<br>- Sistem menyimpan perubahan <i>password</i> pada <i>database</i> |        |
| Actors                 | <i>Admin</i>                                                                                                                                                                                                                                                                                             |        |
| Related <i>Usecase</i> | <i>Login</i>                                                                                                                                                                                                                                                                                             |        |
| Stakeholder            | Karyawan ASComputer                                                                                                                                                                                                                                                                                      |        |
| Precondition           | Sistem menampilkan <i>form</i> setting akun                                                                                                                                                                                                                                                              |        |
| PostCondition          | Sistem menyimpan <i>password</i> baru <i>Admin</i>                                                                                                                                                                                                                                                       |        |
| Flows of Events        | Actors                                                                                                                                                                                                                                                                                                   | System |

|                     |                                                                                                                                                                         |                                                                                                         |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
|                     | 1. <i>Admin</i> membuka <i>form</i> setting akun<br>2. <i>Admin</i> memasukkan <i>Username,password</i> lama, <i>password</i> baru, dan konfirmasi <i>password</i> baru | 1. Sistem menampilkan <i>form</i> setting akun<br>2. Sistem menyimpan <i>password</i> baru <i>Admin</i> |
| Exception Condition | -                                                                                                                                                                       |                                                                                                         |

Untuk merubah *password* lama menjadi *password* baru

dijelaskan pada tabel 4.5.*Admin* harus memasukkan *username* dan *password* lama sebelum melakukan perubahan *password*.

Tabel 4.6 Deskripsi *Usecase* Mensimulasikan Spesifikasi Komputer

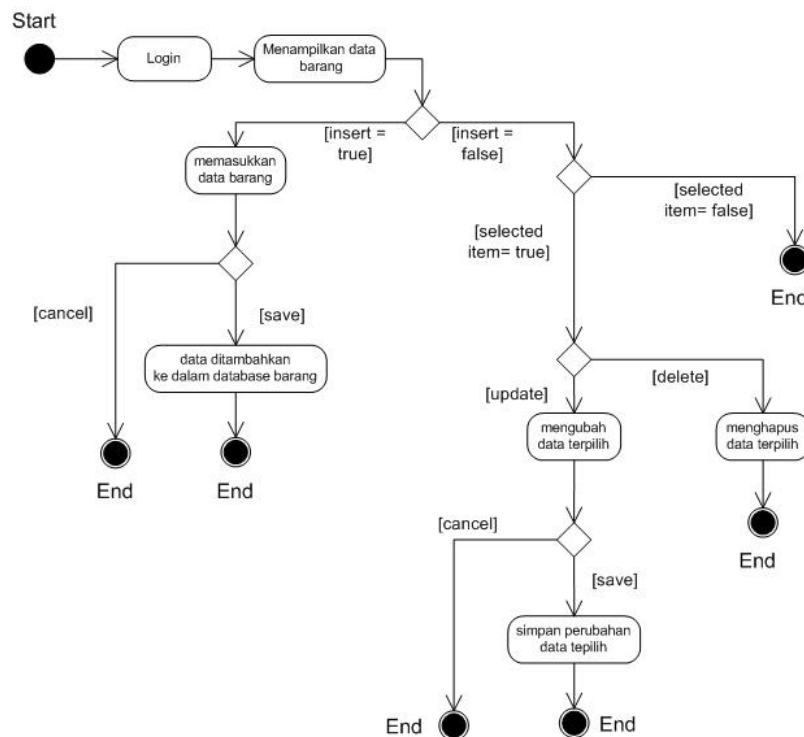
| <i>Usecase</i> Name | Mensimulasikan Spesifikasi Komputer                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Skenario            | <ul style="list-style-type: none"> <li>- <i>User</i> membuat daftar simulasi spesifikasi komputer</li> <li>- <i>User</i> memcetak hasil simulasi spesifikasi komputer</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| Brief Description   | <ul style="list-style-type: none"> <li>- <i>User</i> membuka <i>form</i> simulasi</li> <li>- Sistem menampilkan <i>form</i> simulasi</li> <li>- <i>User</i> memilih jenis barang</li> <li>- Sistem menampilkan data barang sesuai dengan jenis barang</li> <li>- <i>User</i> memilih barang,kemudian menekan tombol <i>Add</i> atau klik kanan <i>Add</i></li> <li>- Sistem memasukkan data barang yang dipilih ke dalam daftar simulasi</li> <li>- <i>User</i> memilih barang, kemudian klik kanan Detail</li> <li>- Sistem menampilkan <i>informasi</i> mengenai barang yang dipilih</li> <li>- <i>User</i> mengubah <i>quantity</i> barang pada daftar simulasi menjadi 0</li> <li>- Sistem menghapus data barang yang <i>quantity</i></li> </ul> |

|                        |                                                                                                                                                                                                                               |                                                                                                                                                                                                                          |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                        | sama dengan 0 dari daftar simulasi<br>- <i>User</i> menekan tombol Clear<br>- Sistem mengosongkan daftar simulasi spesifikasi<br>- <i>User</i> menekan tombol cetak atau print<br>- Sistem menampilkan laporan hasil simulasi |                                                                                                                                                                                                                          |
| Actors                 | <i>Admin</i> dan <i>User</i>                                                                                                                                                                                                  |                                                                                                                                                                                                                          |
| Related <i>Usecase</i> | -                                                                                                                                                                                                                             |                                                                                                                                                                                                                          |
| Stakeholder            | Karyawan ASComputer                                                                                                                                                                                                           |                                                                                                                                                                                                                          |
| Precondition           | Sistem menampilkan <i>form</i> simulasi                                                                                                                                                                                       |                                                                                                                                                                                                                          |
| PostCondition          | Sistem menyimpan <i>password</i> baru <i>Admin</i>                                                                                                                                                                            |                                                                                                                                                                                                                          |
| Flows of Events        | Actors                                                                                                                                                                                                                        | System                                                                                                                                                                                                                   |
|                        | 1. <i>User</i> membuka <i>form</i> simulasi<br>2. <i>User</i> memilih barang dari daftar barang ke daftar simulasi<br>3. <i>User</i> menghapus daftar simulasi<br>4. <i>User</i> mencetak daftar simulasi                     | 1. Sistem menampilkan <i>form</i> simulasi<br>2. Sistem menyalin informasi merk, qty dan harga barang ke dalam daftar simulasi<br>3. Sistem mengosongkan daftar simulasi<br>4. Sistem menampilkan laporan hasil simulasi |
| Exception Condition    | -                                                                                                                                                                                                                             |                                                                                                                                                                                                                          |

Tabel 4.6 menjelaskan alur *usecase* mensimulasikan spesifikasi komputer. *Usecase* simulasi merupakan tahap-tahap bagaimana *user* membuat sebuah spesifikasi komputer.

#### 4.1.2 Activity Diagram

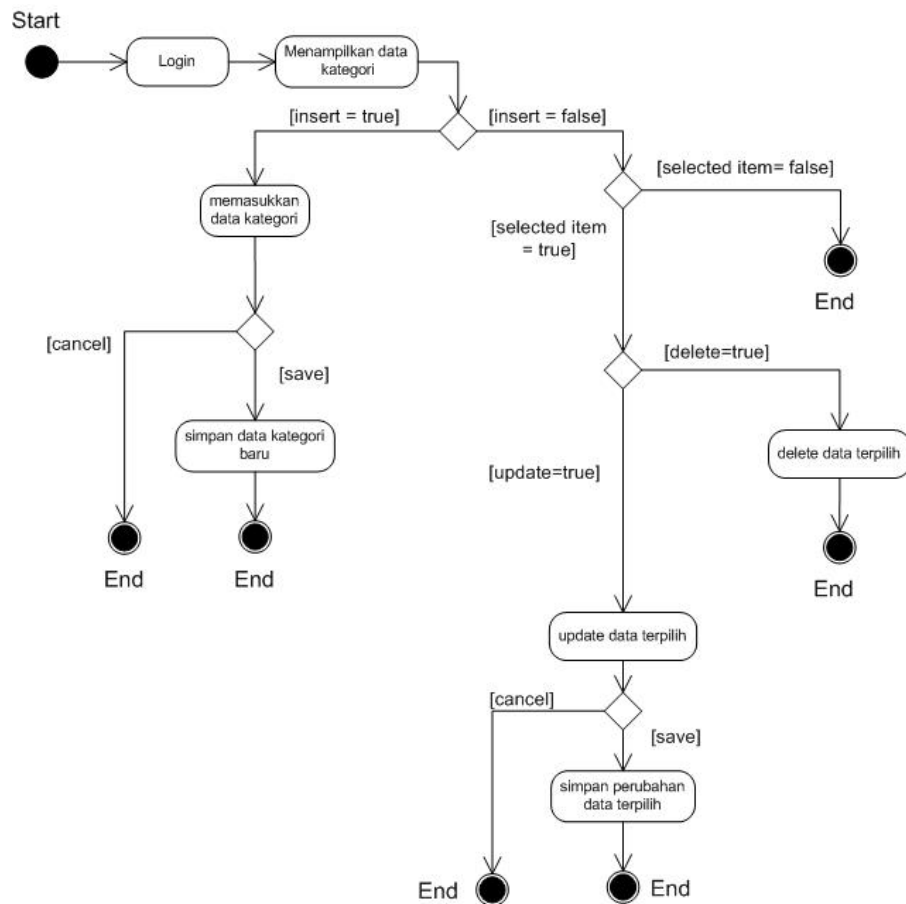
Activity diagram merupakan cara lain dalam memodelkan aliran kejadian. Activity diagram digunakan untuk menggambarkan *workflow* / aliran kerja proses dalam suatu sistem. Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.



**Gambar 4.4** Activity Mengelola Data Barang

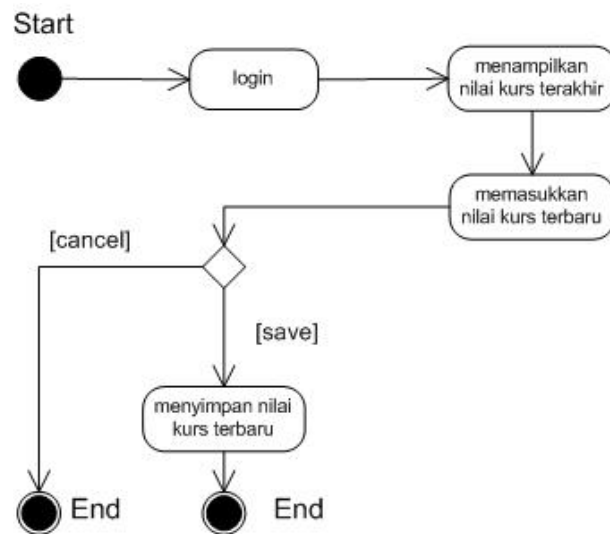
Aktivitas untuk melakukan pemrosesan atau pengaturan data barang ditunjukkan pada gambar 4.4. Pada aktivitas ini *Admin* / *Admin*

dapat melakukan penambahan, pengubahan serta penghapusan data barang.



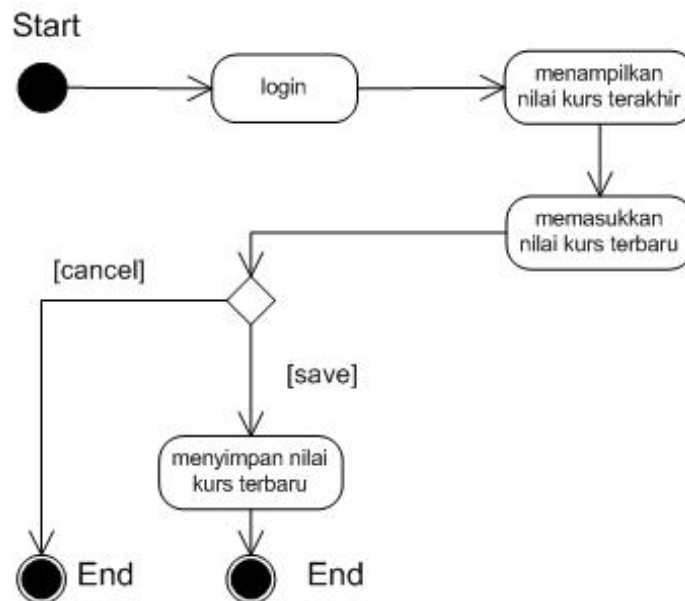
**Gambar 4.5**Activity Kategori

Gambar 4.5 menunjukkan aktivitas *Admin* dalam melakukan pengaturan kategori barang yang meliputi penambahan, pengubahan serta penghapusan data.



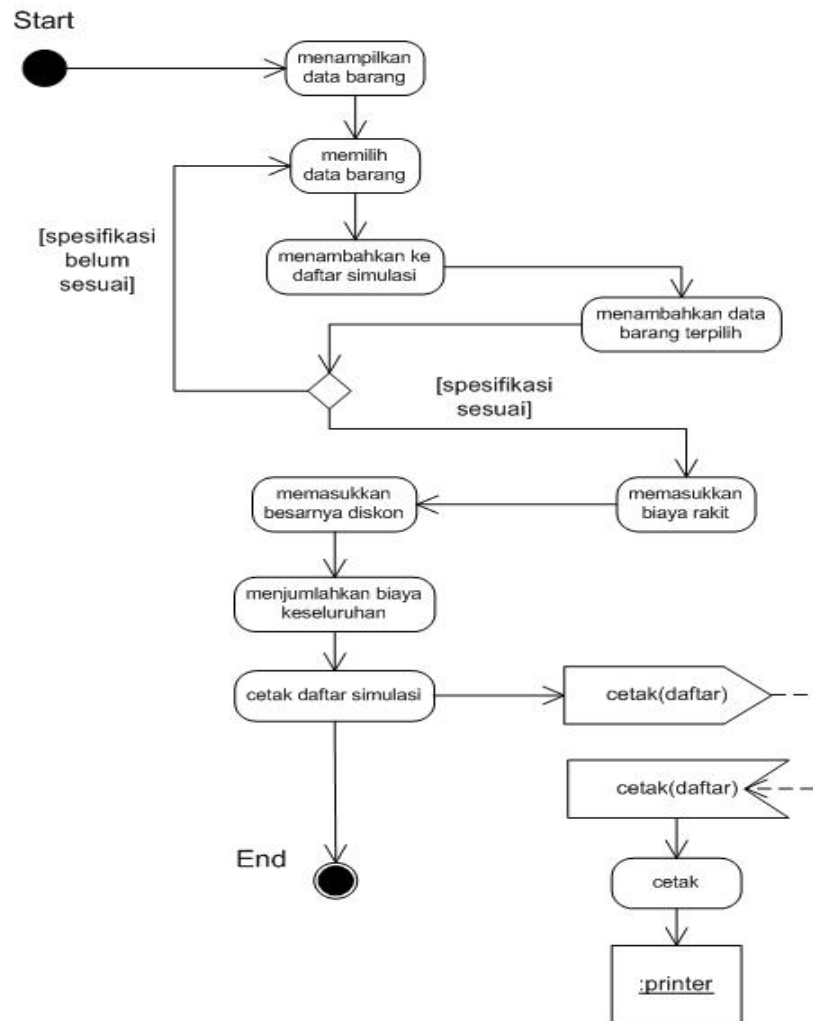
**Gambar 4.6**Activity Kurs

Gambar 4.6 menunjukkan aktivitas *Admin* dalam melakukan pengubahan nilai dolar.



**Gambar 4.7**Activity Akun

Aktivitas merubah *password* lama menjadi *password* baru dijelaskan pada gambar 4.7



**Gambar 4.8** Activity Simulasi

Aktivitas dalam membuat spesifikasi komputer dijelaskan pada gambar 4.8.

### 4.1.3 Sequence Diagram

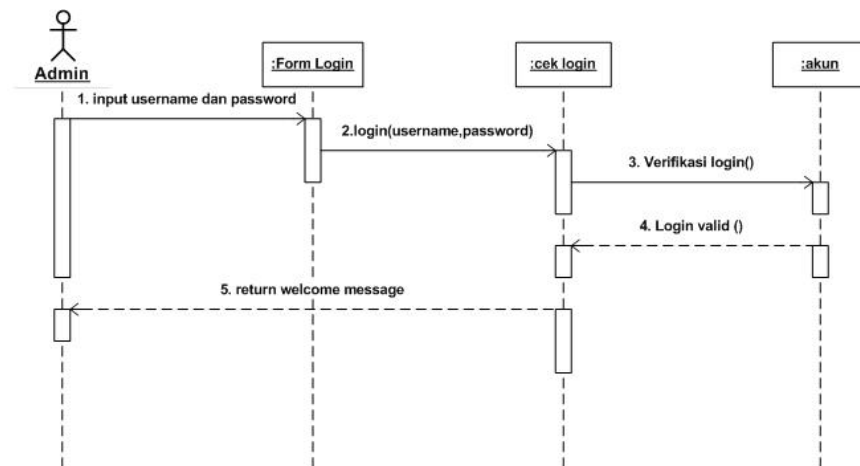
#### 4.1.3.1. Sequence Diagram Login

Sequence Diagram proses *Login* menjelaskan alur untuk masuk bagi *Admin* ke form setting. Proses yang terjadi antara lain:

1. *Admin* memasukkan *username* dan *password*

2. Sistem memasukkan data *username* dan *password* yang dimasukkan
3. Sistem melakukan verifikasi *login*, jika *login* valid maka *return welcome message*.

*Sequence* diagram *login* ditunjukkan pada gambar 4.9



**Gambar 4.9** *Sequence Diagram Login*

#### 4.1.3.2. *Sequence Diagram* Pengolahan Data Barang

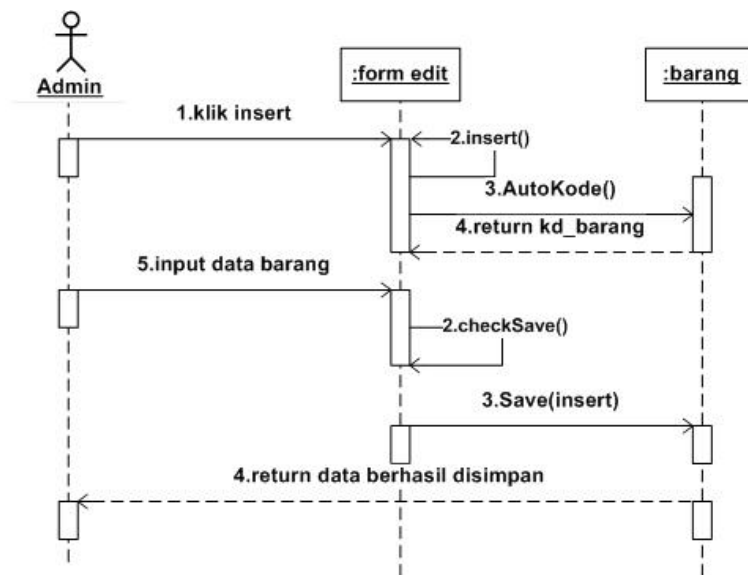
##### 4.1.3.2.1 Menambah data barang

*Sequence* diagram menambah data barang menjelaskan alur memasukkan data-data barang ke dalam database sistem informasi *Specification Maker*. Berikut proses yang terjadi pada *Sequence* diagram menambah data barang:

1. *Admin* memasukkan data barang baru
2. Sistem melakukan pengecekan data.
3. Jika data valid maka data barang disimpan ke database barang.

4. Setelah proses penyimpanan berhasil, maka sistem akan mengirimkan konfirmasi penyimpanan data berhasil.

*Sequence* diagram menambah data barang ditunjukkan pada gambar 4.10



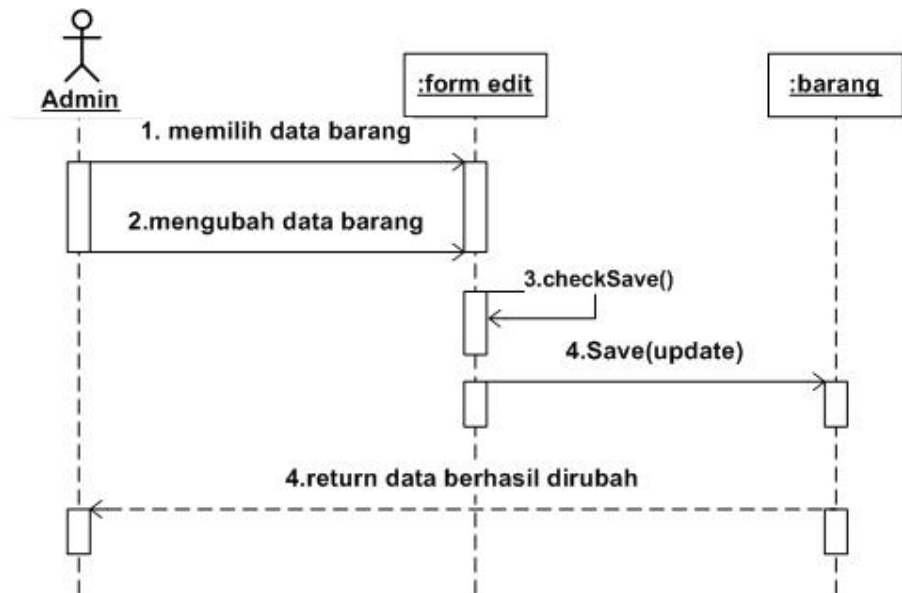
**Gambar 4.10** *Sequence* Diagram menambah data barang

#### 4.1.3.2.2 Mengubah data barang

*Sequence* diagram mengubah data barang menjelaskan mengenai alur memperbarui data barang. Berikut proses yang terjadi pada *Sequence* diagram mengubah data barang:

1. *Admin* mengubah data barang
2. Sistem melakukan pengecekan data barang yang diubah.
3. Jika valid maka data perubahan disimpan ke dalam database.
4. Setelah proses penyimpan data yang diubah berhasil, maka sistem akan mengirimkan konfirmasi perubahan data berhasil.

*Sequence* diagram mengubah data barang ditunjukkan pada gambar 4.11



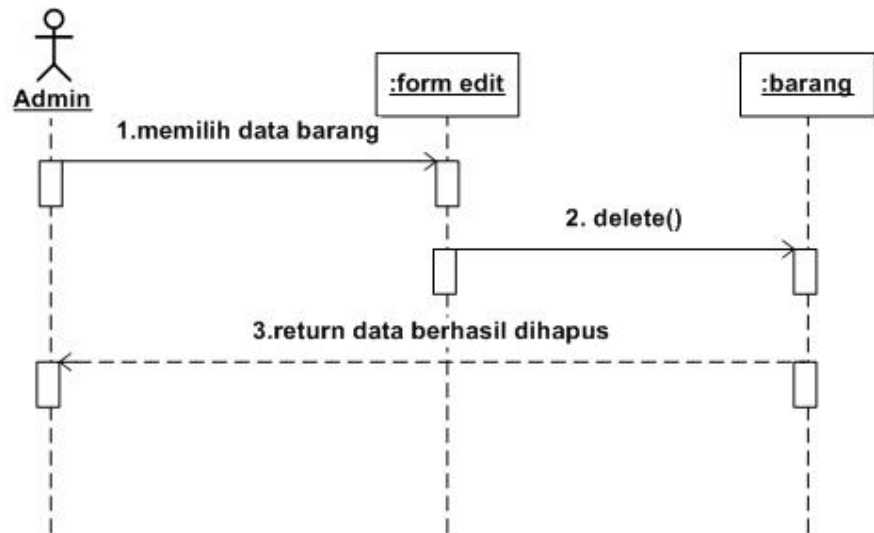
**Gambar 4.11** *Sequence* diagram mengubah data barang

#### 4.1.3.2.3 Menghapus data barang

*Sequence* diagram menghapus data barang menjelaskan mengenai alur menghapus data barang. Berikut proses yang terjadi pada *Sequence* diagram mengubah data barang:

1. *Admin* menghapus data barang
2. Sistem menghapus data barang
3. Setelah proses hapus data yang berrhasil, maka sistem akan mengirimkan konfirmasi hapus data berhasil.

*Sequence* diagram menghapus data barang ditunjukkan pada gambar 4.12



**Gambar 4.12** *Sequence diagram* menghapus data barang

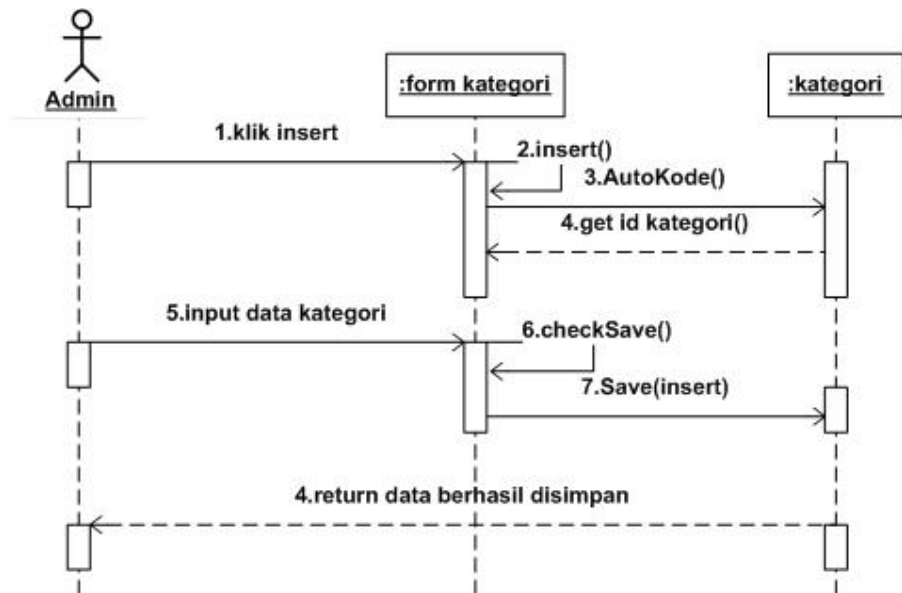
#### 4.1.3.3. *Sequence Diagram* Kategori

##### 4.1.3.3.1 Menambah data kategori

*Sequence diagram* menambah data kategori menje laskan alur memasukkan data-data kategori ke dalam database sistem informasi *Specification Maker*. Berikut proses yang terjadi pada *Sequence diagram* menambah data kategori:

1. *Admin* memasukkan data kategori baru
2. Sistem melakukan pengechekan data.
3. Jika data valid maka data barang disimpan ke database kategori.
4. Setelah proses penyimpanan berhasil, maka sistem akan mengirimkan konfirmasi penyimpanan data berhasil.

Sequence diagram menambah data kategori ditunjukkan pada gambar 4.13



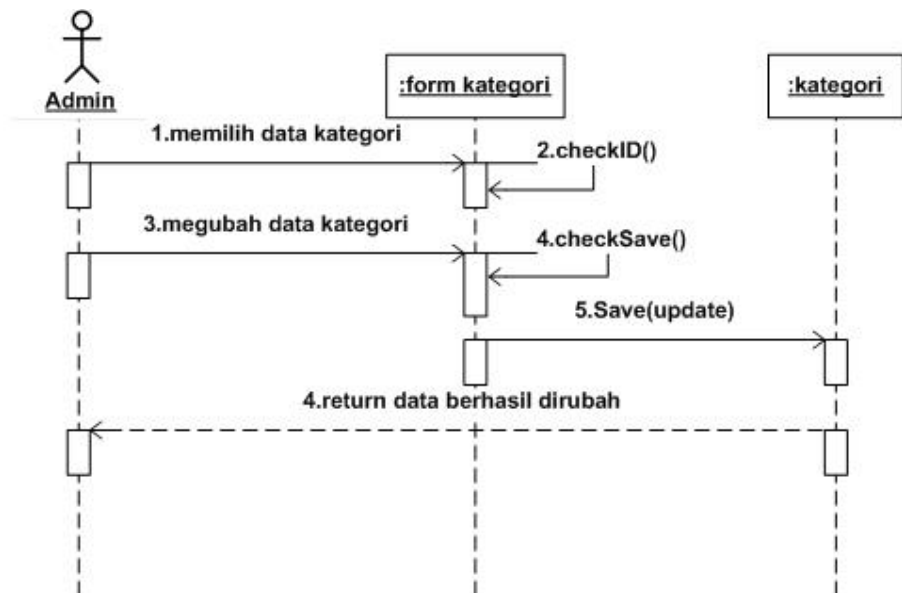
**Gambar 4.13** Sequence diagram menambah data kategori

#### 4.1.3.3.2 Mengubah data kategori

Sequence diagram mengubah data kategori menjelaskan mengenai alur memperbarui data kategori. Berikut proses yang terjadi pada Sequence diagram mengubah data kategori:

1. Admin mengubah data kategori
2. Sistem melakukan pengecekan data kategori yang diubah.
3. Jika valid maka data perubahan disimpan ke dalam database.
4. Setelah proses penyimpanan data yang diubah berhasil, maka sistem akan mengirimkan konfirmasi perubahan data berhasil.

Sequence diagram mengubah data kategori ditunjukkan pada gambar 4.14



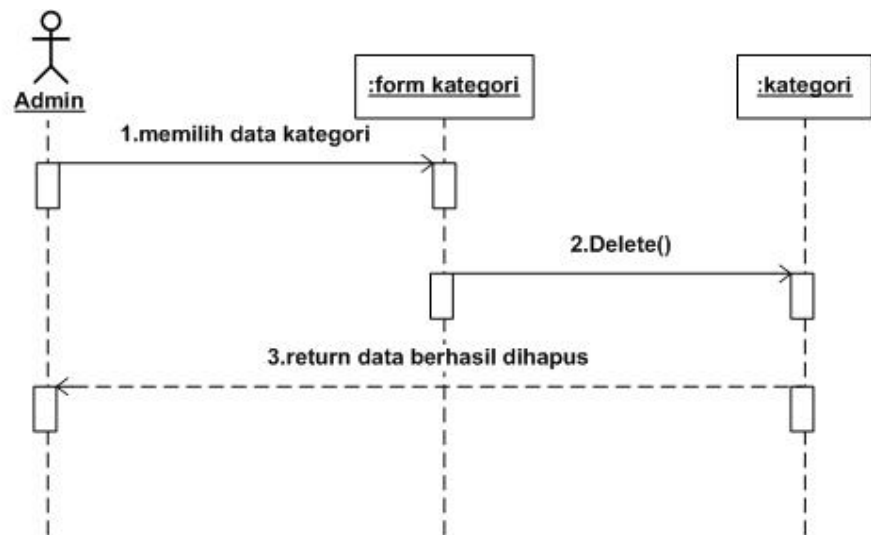
**Gambar 4.14** *Sequence* diagram mengubah data kategori

#### 4.1.3.3 Menghapus data kategori

*Sequence* diagram menghapus data kategori menjelaskan mengenai alur menghapus data kategori. Berikut proses yang terjadi pada *Sequence* diagram mengubah data kategori:

1. *Admin* menghapus data kategori
2. Sistem menghapus data kategori
3. Setelah proses hapus data yang berhasil, maka sistem akan mengirimkan konfirmasi hapus data berhasil.

*Sequence* diagram menghapus data kategori ditunjukkan pada gambar 4.15



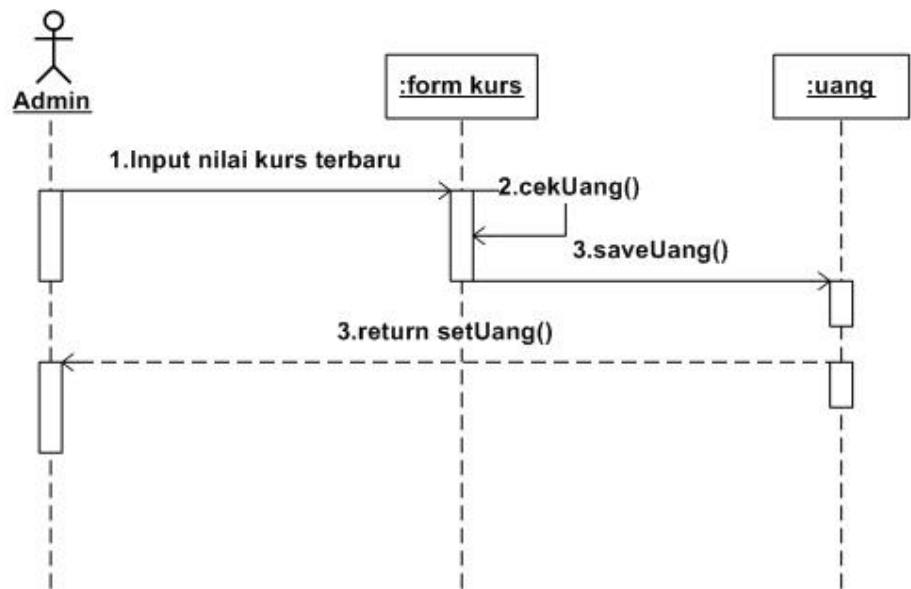
**Gambar 4.15** *Sequence* diagram menghapus data kategori

#### 4.1.3.4. *Sequence* Diagram Kurs

*Sequence* diagram kurs menjelaskan alur memperbarui nilai kurs. Berikut proses yang terjadi pada *Sequence* diagram kurs:

1. *Admin* memasukkan nilai kurs terbaru
2. Sistem menyimpan nilai kurs terbaru
3. Sistem menampilkan nilai kurs terbaru yang telah berhasil disimpan.

*Sequence* diagram kurs ditunjukkan pada gambar 4.16



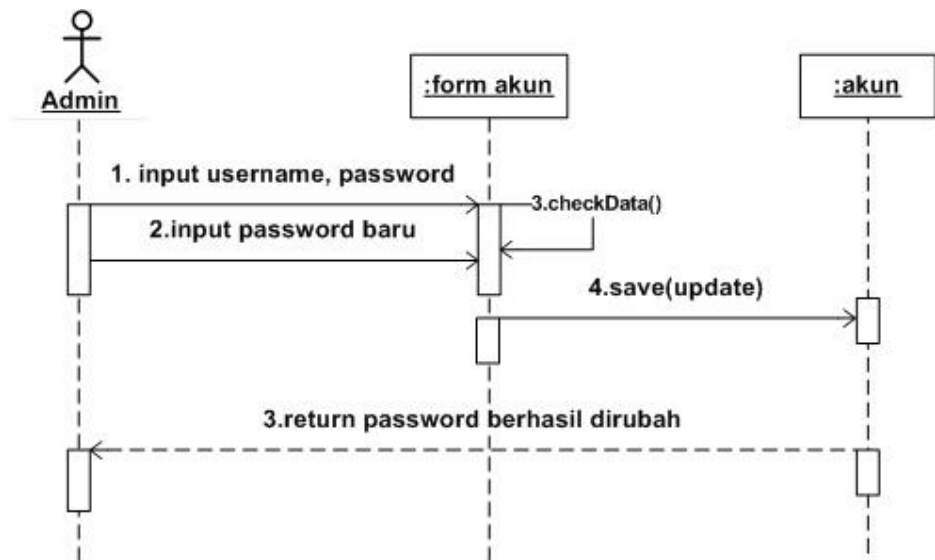
**Gambar 4.16** *Sequence diagram kurs*

#### 4.1.3.5. *Sequence Diagram Akun*

*Sequence Diagram* akun menjelaskan alur mengganti *password* lama dengan *password* baru. Berikut proses yang terjadi pada *Sequence diagram* akun:

1. *Admin* memasukkan *password* baru
2. Sistem menyimpan *password* baru pada databae akun
3. Sistem menampilkan konfirmasi bahwa *password* berhasil dirubah.

*Sequence diagram* akun ditunjukkan pada gambar 4.17



**Gambar 4.17** *Sequence diagram* akun

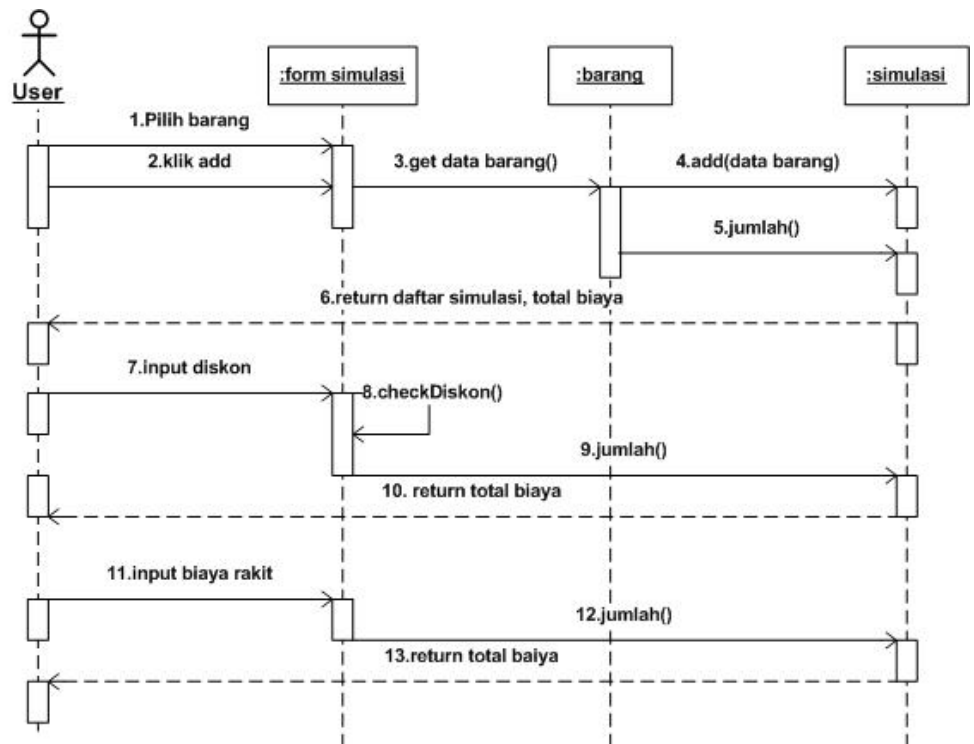
#### 4.1.3.6. *Sequence Diagram* Simulasi

##### 4.1.3.6.1 Membuat daftar simulasi

*Sequence diagram* membuat daftar simulasi menjelaskan alur membuat simulasi spesifikasi komputer. Berikut adalah proses yang terjadi pada *Sequence diagram* membuat daftar simulasi:

1. *User* memilih barang
2. Sistem mengambil data barang yang dipilih oleh *user*
3. Sistem menambah data barang yang dipilih ke tabel simulasi
4. Sistem menampilkan hasil penambahan data barang pada daftar simulasi

*Sequence diagram* membuat daftar simulasi ditunjukkan pada gambar 4.18



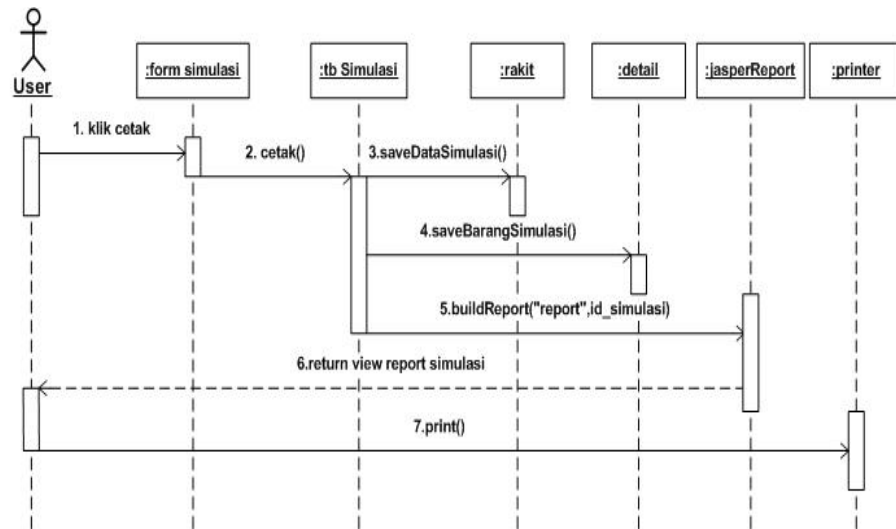
**Gambar 4.18** *Sequence* diagram membuat daftar simulasi

#### 4.1.3.6.2 Mencetak daftar simulasi

*Sequence* diagram mencetak daftar simulasi menjelaskan mengenai alur mencetak ddaftar simulasi yang telah dibuat. Berikut adalah proses yang terjadi pada *Sequence* diagram mencetak daftar simulasi:

1. *User* mencetak daftar simulasi
2. Sistem mengambil daftar simulasi yang telah dibuat
3. Sistem menampilkan gambaran daftar hasil simulasi dalam bentuk *report* sebelum dicetak.

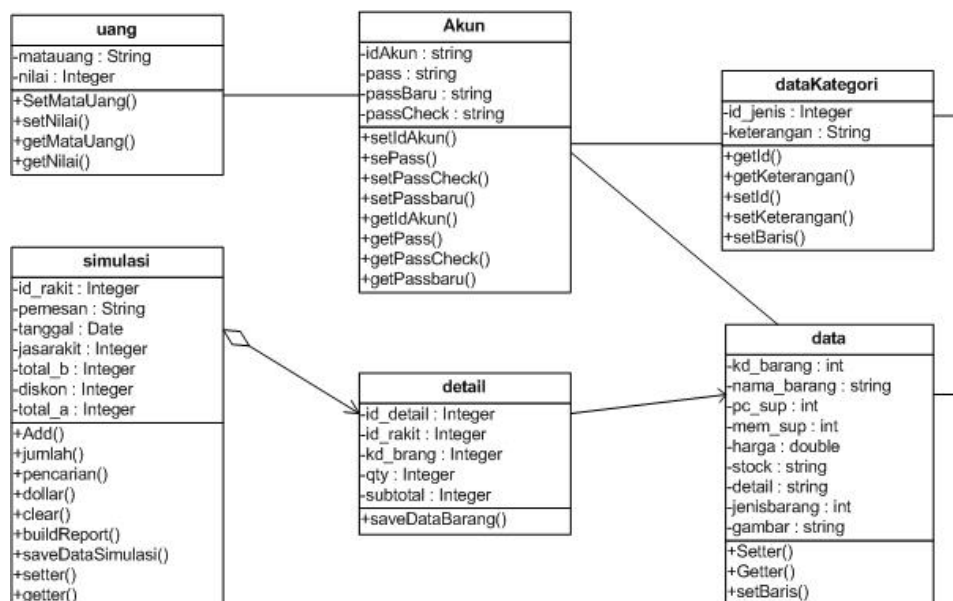
*Sequence* diagram mencetak daftar simulasi ditunjukkan pada gambar 4.19



Gambar 4.19 Sequence diagram mencetak daftar simulasi

#### 4.1.4 Class Diagram

Setelah merancang *Activity diagram* dan *Sequence diagram* maka dapat dibuat rancangan *class diagram* dan menentukan *attribute* dan *method* yang diperlukan untuk masing-masing *class*.



Gambar 4.20 Class Diagram

### 1. Akun

*Class* akun adalah *class* yang menyimpan data *admin*, yaitu berisikan *username* serta *password admin*.

### 2. Uang

*Class* uang adalah *class* yang menyimpan data nilai kurs. Terdapat *attribute* matauang dan nilai.

### 3. Kategori

*Class* kategori adalah *class* yang menyimpan data kategori barang. Terdapat beberapa *attribute* *id\_jenis* dan keterangan.

### 4. Barang

*Class* barang merupakan *class* yang menyimpan data-data barang, pada *class* ini terdapat beberapa *attribute* seperti: *kd\_brang*; *nama\_barang*, *pc\_sup*, *mem\_sup*, *harga*, *stock*, *detail* dan *jenisbarang*. *Kd\_barang* merupakan *primary\_key*.

### 5. Simulasi

*Class* simulasi merupakan *class* yang menyimpan hasil simulasi spesifikasi komputer. *Class* simulasi terdapat beberapa *attribute* antara lain: *id\_rakit*, *pemesan*, *Tanggal*, *jasa\_rakit*, *total\_b*, *diskon*, dan *total\_a*. *ID\_rakit* merupakan *primary\_key*.

### 6. Detail

*Class* detail merupakan *class* yang menyimpan data barang yang terdapat pada daftar simulasi. Pada *class* ini terdapat *attribute* antara lain: *id\_detail*, *id\_rakit*, *kd\_barang*, *qty*, dan *subtotal*.

#### 4.1.5 Desain Tabel

##### 4.1.5.2 Tabel akun

Tabel akun merupakan tabel yang berisikan data-data *Admin* yaitu meliputi *username* dan *password*.

Tabel 4.7 Tabel akun

| Nama Kolom | Tipe Data    | Keterangan  |
|------------|--------------|-------------|
| id_akun    | varchar(50)  | Primary Key |
| pass       | varchar(100) | Password    |

##### 4.1.5.3 Tabel barang

Tabel barang merupakan tabel yang menyimpan data-data barang, dimana *kd\_barang* merupakan *primary key*. *Pc\_sup*, *mem\_sup* dan *jenisbarang* merupakan *foreign key*.

Tabel 4.8 Tabel barang

| Nama Kolom     | Tipe Data     | Keterangan  |
|----------------|---------------|-------------|
| kd_barang      | int(5)        | Primary Key |
| nama_barang    | varchar(100)  |             |
| id_pcsup       | int(5)        | Foreign Key |
| id_memsup      | int(5)        | Foreign Key |
| harga          | double        |             |
| stock          | varchar(15)   |             |
| detail         | varchar(3000) |             |
| id_jenisbarang | int(5)        | Foreign Key |
| matauang       | varchar(50)   |             |
| gambar         | varchar(50)   |             |

##### 4.1.5.4 Tabel jenis\_barang

Tabel jenis barang merupakan tabel yang menyimpan data kategori barang dimana *id\_jenis* merupakan *primary key*.

Tabel 4.9 Tabel jenis\_barang

| Nama Kolom     | Tipe Data   | Keterangan  |
|----------------|-------------|-------------|
| id_jenisbarang | int(5)      | Primary Key |
| keterangan     | varchar(50) |             |

#### 4.1.5.5 Tabel pc\_sup

Tabel pc\_sup merupakan tabel yang menyimpan jenis pc socket seperti Intel 2011, AMD AM2 dan lain sebagainya. Id\_jenis merupakan *primary key*.

Tabel 4.10 Tabel pc\_sup

| Nama Kolom | Tipe Data   | Keterangan  |
|------------|-------------|-------------|
| id_pcsup   | int(5)      | Primary Key |
| keterangan | varchar(50) |             |

#### 4.1.5.6 Tabel mem\_sup

Tabel mem\_sup merupakan tabel yang menyimpan jenis memory socket seperti DDR2, DDR3 dan lain sebagainya. Id\_jenis merupakan *primary key*.

Tabel 4.11 Tabel mem\_sup

| Nama Kolom | Tipe Data   | Keterangan  |
|------------|-------------|-------------|
| id_memsup  | int(5)      | Primary Key |
| keterangan | varchar(50) |             |

#### 4.1.5.7 Tabel uang

Tabel uang merupakan tabel yang menyimpan nilai kurs terbaru. *Primary key* dari tabel uang adalah matauang.

Tabel 4.12 Tabel uang

| Nama Kolom | Tipe Data   | Keterangan  |
|------------|-------------|-------------|
| matauang   | varchar(10) | Primary Key |
| Nilai      | int(10)     |             |

#### 4.1.5.8 Tabel rakit

Tabel rakit berisikan data-data hasil simulasi spesifikasi komputer seperti harga keseluruhan. Id\_rakit merupakan *primary key*.

Tabel 4.13 Tabel rakit

| Nama Kolom | Tipe Data    | Keterangan  |
|------------|--------------|-------------|
| id_rakit   | int(12)      | Primary Key |
| pemesan    | varchar(100) |             |
| tanggal    | date         |             |
| jasarakit  | int(12)      |             |
| total_b    | int(12)      |             |
| diskon     | int(20)      |             |
| total_a    | Int(12)      |             |

#### 4.1.5.9 Tabel detail

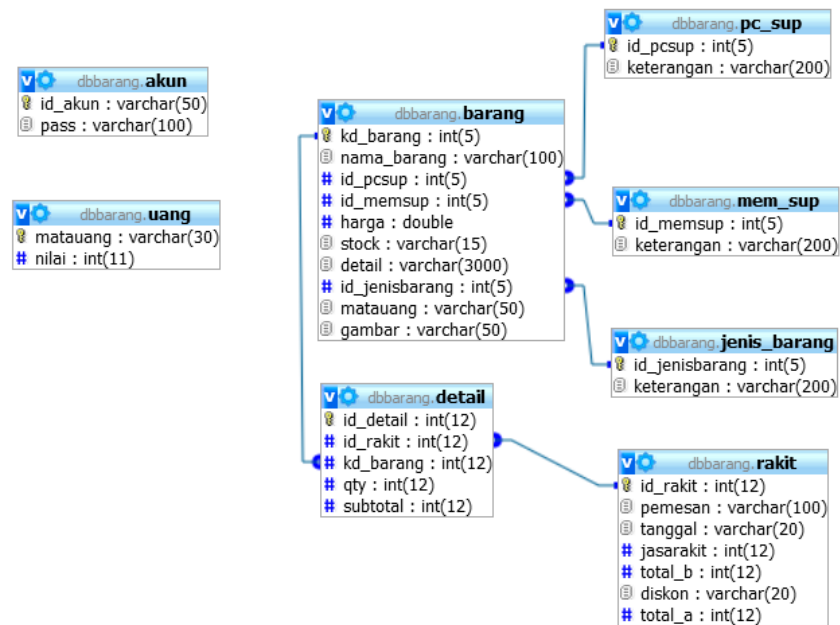
Tabel detail merupakan tabel yang menyimpan data -data hasil simulasi spesifikasi komputer seperti kode barang, id rakit, subtotal dan qty. id\_detail merupakan *primary key* sedangkan id\_rakit dan jenis barang merupakan *foreign key*.

Tabel 4.14 Tabel detail

| Nama Kolom | Tipe Data | Keterangan  |
|------------|-----------|-------------|
| id_detail  | int(12)   | Primary Key |
| id_rakit   | int(12)   | Foreign Key |
| kd_barang  | int(12)   | Foreign Key |
| qty        | int(12)   |             |
| subtotal   | int(12)   |             |

#### 4.1.6 Relasi Tabel

Relasi antar tabel-tabel ditunjukkan pada gambar 4.21

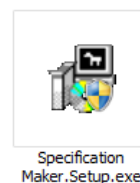


**Gambar 4.21** Relasi antar tabel

## 4.2 Implementasi Sistem

### 4.2.1. Installer *Specification Maker*

Aplikasi *Specification Maker* ini sudah tersedia dalam bentuk installer, sehingga *User* tinggal menjalankan installer tersebut dan langsung dapat dijalankan.



**Gambar 4.22** Installer *Specification Maker*

#### 4.2.2. *Form Home*

*Form Home* adalah *form* yang ditampilkan pertama kali ketika aplikasi dijalankan, melalui *form* ini *User* dapat langsung ke *form* simulasi atau ke *form setting*



**Gambar 4.23** *Form Home*

#### 4.2.3. *Form Setting*

*Form Setting* adalah *form* yang muncul apabila *User* memasukkan *Username* beserta *password* yang valid.



**Gambar 4.24** *Form Setting*

#### 4.2.4. Form Edit

*Form Edit* adalah *Form Internal* dari *Form Setting* yang digunakan untuk *insert*, *update*, serta *delete* data barang.

| Kode Barang | Merk / Tipe  | Kurs | Harga   | Stock |
|-------------|--------------|------|---------|-------|
| 1           | GA-X79-UD3   | IDR  | 1300000 | Ada   |
| 2           | GA-X58A-OC   | IDR  | 1250500 | Ada   |
| 3           | GA-For Intel | IDR  | 850000  | Ada   |
| 4           | GA-900FX     | IDR  | 122000  | Ada   |
| 10          | GA-G41MT     | IDR  | 540000  | Ada   |
| 11          | GA-MAT74     | IDR  | 630000  | Ada   |
| 12          | JW A 611     | IDR  | 59      | Ada   |
| 29          | GA-M52L      | IDR  | 712000  | Ada   |

**Gambar 4.25** *Form Edit*

#### 4.2.5. Form Kategori

*Form Kategori* adalah *Form Internal* dari *form setting* yang digunakan untuk *insert*, *update*, serta *delete* jenis barang, *socket pc* serta *socket memory*.

| No ID | Keterangan  |
|-------|-------------|
| 0     | -           |
| 1     | Motherboard |
| 2     | Processor   |
| 3     | Memory      |
| 4     | VGA         |
| 5     | HardDisk    |
| 6     | Optical     |
| 7     | Keyboard    |
| 8     | PSU         |
| 9     | UPS         |

**Gambar 4.26** *Form Kategori*

#### 4.2.6. *Form Kurs Setting*

*Form Kurs Setting* digunakan untuk melakukan *update* nilai kurs dollar terhadap rupiah.

The screenshot shows a window titled 'Setting' with a menu bar containing 'Edit', 'Setting', 'About', and 'Log Out'. Below the menu bar is a dropdown menu showing 'Kurs Setting'. The main content area has a green dollar sign icon and the text 'KURS MATA UANG DOLLAR'. Below this, there is a label 'Nilai' followed by a text input field containing '9250'. Underneath the input field are two buttons: one with a floppy disk icon and a green arrow, and another with a blue circular arrow icon. At the bottom of the window, it says 'Kurs Sekarang : Rp 9150'.

**Gambar 4.27** *Form Kurs*

#### 4.2.7. *Form Akun Setting*

*Form Akun Setting* digunakan untuk mengubah *password* lama menjadi *password* baru.

The screenshot shows a window titled 'Setting' with a menu bar containing 'Edit', 'Setting', 'About', and 'Log Out'. Below the menu bar is a dropdown menu showing 'Setting Akun'. The main content area has a key icon and the text 'Ganti Password'. Below this, there are four rows of labels and input fields: 'ID Pengguna' with the value 'admin', 'Password Lama' with masked characters '\*\*\*\*\*', 'Password Baru' with masked characters '\*\*\*\*', and 'Password Baru Check' with masked characters '\*\*\*\*'. At the bottom of the window are three buttons: one with a floppy disk icon and a green arrow, one with a red 'X' icon, and one with a blue circular arrow icon.

**Gambar 4.28** *Form Akun Setting*

#### 4.2.8. Form Simulasi

*Form Simulasi* merupakan *form* yang digunakan *User* untuk mensimulasikan sebuah spesifikasi komputer.

The 'Form Simulasi' interface includes a search bar, a category dropdown (Motherboard), and a list of items with their prices and stock status. The simulation table on the right shows the selected components and their total price.

| Kode Bara... | Merk / Tipe  | Kurs | Harga   | Stock |
|--------------|--------------|------|---------|-------|
| 1            | GA-X79-U     | IDR  | 1300000 | Ada   |
| 2            | GA-X58A...   | IDR  | 1250500 | Ada   |
| 3            | GA-For Intel | IDR  | 850000  | Ada   |
| 4            | GA 990FX     | IDR  | 1722000 | Ada   |
| 10           | GA G41MT     | IDR  | 540000  | Ada   |
| 11           | GA MA74GM    | IDR  | 530000  | Ada   |
| 12           | JW A 61 PM   | USD  | 59      | Ada   |
| 26           | GA M52LT     | IDR  | 712000  | Ada   |

| Item     | Merk / Type | Qty | Harga   |
|----------|-------------|-----|---------|
| MB       | GA-X79-UD3  | 1   | 1300000 |
| CPU      |             | 0   | 0       |
| Memory   |             | 0   | 0       |
| VGA      |             | 0   | 0       |
| Harddisk |             | 0   | 0       |
| Optical  |             | 0   | 0       |
| Casing   |             | 0   | 0       |
| LCD      |             | 0   | 0       |
| Mouse    |             | 0   | 0       |
| Keyboard |             | 0   | 0       |

Nilai jual \$1 = Rp 9150

Rakit ☐ Ya ☒ Tidak

TOTAL ( +Rakit ) 1300000

Diskon ☐ % ☐ Nominal

**Total Keseluruhan Rp 1,300,000.00**

**Gambar 4.29** *Form Simulasi*

#### 4.2.9. Form Detail Barang

The 'Form Detail Barang' interface shows details for a speaker system, including its name, code, price, stock, and specifications.

**5. CST 6950N**

Jenis Barang: Speaker

Kode Barang: 96

Harga (Rp): 345000

Stock: Ada

**Spesifikasi**

Speaker System, 36 Watts RMS

Satellite: 2

Subwoofer: 1

Control: Remote Control

Color: Black

**Gambar 4.30** *Form detail barang*

#### 4.2.10. Report Daftar Simulasi Spesifikasi


**ASC COMPUTER**  
Jakal Km 7 No.5 Yogyakarta  
 (0274) 882 891, 882 309, 885 389 Fax 0274 882 817  
 Komplek Ruko Babarsari No.69-71 Yogyakarta  
 (0274) 433 2932, 433 2933, 485 126

**GROUP**  
 Nama Pemesan:  
**Kholis**

14-02-2012

| ITEM                     | MERK/TYPE              | QTY | HARGA (Rp)     |
|--------------------------|------------------------|-----|----------------|
| Motherboard              | GA-X58A-OC             | 1   | 1250500        |
| Processor                | Intel 2012             | 1   | 1108230        |
| Memory                   | V-GEN 2GB PC10600      | 1   | 105000         |
| VGA                      | GF 8400                | 1   | 346885         |
| HardDisk                 | SEAGATE 500GB          | 1   | 750000         |
| Optical                  | DVDRW Samsung 22x SATA | 1   | 165000         |
| Casing                   | Dazumba 522            | 1   | 205000         |
| Monitor                  | LED Philips 16.5       | 1   | 1054170        |
| Mouse                    | M W M235               | 1   | 60000          |
| Keyboard                 | Genius USB             | 1   | 144160         |
| PSU                      | ENLIGHT 300W           | 1   | 315000         |
| UPS                      | Kenika 500W            | 1   | 350000         |
| Speaker                  | Simbada                | 1   | 250000         |
| <b>Rakit</b>             |                        |     | 25000          |
| <b>Total</b>             |                        |     | 6128945        |
| <b>Discount</b>          |                        |     | 25%            |
| <b>Total Keseluruhan</b> |                        |     | <b>4596709</b> |

Setiap Orang Senang dan Puas membeli Produk dari ASC

Marketing :

No Telepon :

Kemudahan di ASC Group  
 1 Fasilitas Purna jual yang Menarik  
 2 Kemudahan pelayanan Garansi  
 3 Konsultasi teknis gratis  
 4 Penjualan Via Online  
 5 Semua barang PC/NB bisa dimiliki secara KREDIT  
  
[www.asccomputer.co.id](http://www.asccomputer.co.id)

**Gambar 4.31** Daftar simulasi spesifikasi

### 4.3 Pengujian

Pengujian yang digunakan untuk menguji Sistem Informasi Specification Maker ini adalah metode pengujian *black box*. Pengujian *black box* berfokus pada fungsional *form simulasi* dari sistem informasi *Specification Maker*.

#### 4.3.1 Rencana Pengujian Alpha

Rencana pengujian yang akan dilakukan dengan menguji *form simulasi* sistem informasi *Specification Maker* secara alpha dan betha. Rencana pengujian selengkapnya terlihat pada tabel 4.15

Tabel 4.15 Rencana pengujian *alpha*

| Item uji | Detail pengujian      | Jenis pengujian  |
|----------|-----------------------|------------------|
| Simulasi | Tampil barang         | <i>Black box</i> |
|          | Detail barang         |                  |
|          | Tambah barang         |                  |
|          | Hapus barang          |                  |
|          | Hapus daftar simulasi |                  |
|          | Cari barang           |                  |
|          | Cetak daftar simulasi |                  |

#### 4.3.2 Kasus dan Hasil Pengujian *Alpha*

Berdasarkan rencana pengujian yang telah dibuat, maka dapat dilakukan pengujian sebagai pada tabel 4.16 berikut :

Tabel 4.16 Hasil pengujian *alpha*

| Kasus dan hasil uji |                                                                                                |
|---------------------|------------------------------------------------------------------------------------------------|
| Data masukan        | Klik item pada combo box kategori                                                              |
| Yang diharapkan     | Tabel menampilkan data barang sesuai dengan kategori yang dipilih                              |
| Pengamatan          | Tabel menampilkan data barang sesuai dengan kategori barang                                    |
| Kesimpulan          | Diterima                                                                                       |
| Kasus dan hasil uji |                                                                                                |
| Data masukan        | Pilih barang, klik kanan, detail                                                               |
| Yang diharapkan     | Menampilkan <i>form</i> detail barang yang dipilih                                             |
| Pengamatan          | Sistem menampilkan <i>Form</i> detail barang yang berisikan informasi dari barang yang dipilih |
| Kesimpulan          | Diterima                                                                                       |
| Kasus dan hasil uji |                                                                                                |
| Data masukan        | Klik 2x pada barang yang dipilih                                                               |
| Yang diharapkan     | Sistem menambahkan barang pada daftar simulasi                                                 |
| Pengamatan          | Daftar simulasi bertambah sesuai dengan barang yang ditambahkan                                |
| Kesimpulan          | Diterima                                                                                       |

|                                        |                                                                                              |
|----------------------------------------|----------------------------------------------------------------------------------------------|
| <b>Kasus dan hasil uji</b>             |                                                                                              |
| Data masukan                           | Pilih barang, klik kanan, <i>Add/tambah</i>                                                  |
| Yang diharapkan                        | Sistem menambahkan barang pada daftar simulasi                                               |
| Pengamatan                             | Daftar simulasi bertambah sesuai dengan barang yang ditambahkan                              |
| Kesimpulan                             | Diterima                                                                                     |
| <b>Kasus dan hasil uji</b>             |                                                                                              |
| Data masukan                           | Pilih barang, klik tombol <i>Add/tambah</i>                                                  |
| Yang diharapkan                        | Sistem menambahkan barang pada daftar simulasi                                               |
| Pengamatan                             | Daftar simulasi bertambah sesuai dengan barang yang ditambahkan                              |
| Kesimpulan                             | Diterima                                                                                     |
| <b>Kasus dan hasil uji(Data Salah)</b> |                                                                                              |
| Data masukan                           | Tidak memilih barang, klik tombol <i>Add/tambah</i>                                          |
| Yang diharapkan                        | Sistem menampilkan pesan “Anda belum memilih barang, Silahkan Pilih Dahulu”                  |
| Pengamatan                             | Menampilkan pesan “Anda belum memilih barang, Silahkan Pilih Dahulu” sesuai yang diharapkan. |
| Kesimpulan                             | Diterima                                                                                     |
| <b>Kasus dan hasil uji</b>             |                                                                                              |
| Data masukan                           | Klik “ <i>down arrow</i> ” JSpinner sampai 0                                                 |
| Yang diharapkan                        | Sistem menghapus barang pada daftar simulasi                                                 |
| Pengamatan                             | Daftar simulasi berkurang sesuai yang diharapkan                                             |
| Kesimpulan                             | Diterima                                                                                     |
| <b>Kasus dan hasil uji</b>             |                                                                                              |
| Data masukan                           | Klik tombol <i>Clear</i>                                                                     |
| Yang diharapkan                        | Sistem mengosongkan daftar simulasi                                                          |
| Pengamatan                             | Daftar simulasi menjadi kosong.                                                              |
| Kesimpulan                             | Diterima                                                                                     |
| <b>Kasus dan hasil uji</b>             |                                                                                              |
| Data masukan                           | Klik tombol Cari                                                                             |
| Yang diharapkan                        | Tabel menampilkan data barang sesuai dengan kriteria pencarian                               |
| Pengamatan                             | Tabel menampilkan data barang sesuai dengan pencarian                                        |
| Kesimpulan                             | Diterima                                                                                     |

| <b>Kasus dan hasil uji</b>              |                                                                                         |
|-----------------------------------------|-----------------------------------------------------------------------------------------|
| Data masukan                            | Klik tombol Cetak                                                                       |
| Yang diharapkan                         | Sistem menampilkan <i>Jasper viewer</i> dari daftar simulasi yang dibuat                |
| Pengamatan                              | Menampilkan <i>Jasper viewer</i> yang berisikan daftar simulasi sesuai yang diharapkan. |
| Kesimpulan                              | Diterima                                                                                |
| <b>Kasus dan hasil uji (Data salah)</b> |                                                                                         |
| Data masukan                            | Daftar Simulasi kosong, Klik tombol Cetak                                               |
| Yang diharapkan                         | Sistem menampilkan pesan “Tidak Ada Daftar Spesifikasi yang dibuat”                     |
| Pengamatan                              | Menampilkan pesan “Tidak Ada Daftar Spesifikasi yang dibuat”                            |
| Kesimpulan                              | Diterima                                                                                |

#### 4.3.3 Kesimpulan Pengujian *Alpha*

Berdasarkan dari hasil pengujian *Alpha* pada *form* simulasi dengan *sample* uji yang telah dilakukan, memberikan kesimpulan bahwa hasil pengujian *alpha* telah bebas dari kesalahan *sintaks*.

#### 4.3.4 Rencana Pengujian *Beta*

Pengujian *beta* merupakan pengujian langsung oleh *user* / pengguna Specification Maker, yaitu orang-orang yang biasa mengunjungi pameran komputer dengan cara mengisi kuisioner mengenai kepuasan *user*/pengguna. Berikut kuisioner yang dibagikan kepada 20 *responden* seperti pada tabel 4.17:

Tabel 4.17 Kuisisioner

|   |   |                                                                                                                                         |
|---|---|-----------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Q | Apakah tampilan dari <i>Specification Maker</i> ini <b>menarik</b> ?                                                                    |
|   | A | <input type="checkbox"/> Sangat Menarik <input type="checkbox"/> Cukup Menarik <input type="checkbox"/> Kurang Menarik                  |
| 2 | Q | Apakah aplikasi <i>Specification Maker</i> <b>mudah dipelajari</b> dan <b>digunakan</b> ?                                               |
|   | A | <input type="checkbox"/> Sangat Mudah <input type="checkbox"/> Cukup Mudah <input type="checkbox"/> Sulit                               |
| 3 | Q | Apakah aplikasi <i>Specification Maker</i> <b>memberikan informasi</b> yang diharapkan dengan <b>cepat</b> dan <b>mudah</b> ?           |
|   | A | <input type="checkbox"/> Sangat Cepat <input type="checkbox"/> Cukup Cepat <input type="checkbox"/> Sulit                               |
| 4 | Q | Apakah aplikasi <i>Specification Maker</i> <b>membantu</b> dan <b>memudahkan</b> anda dalam membuat sebuah <b>spesifikasi</b> komputer? |
|   | A | <input type="checkbox"/> Sangat Membantu <input type="checkbox"/> Cukup Membantu <input type="checkbox"/> Kurang Membantu               |

#### 4.3.5 Hasil Pengujian Beta

##### 4.3.5.1 Apakah tampilan dari *Specification Maker* ini menarik ?

Tabel 4.18 Pertanyaan 1

| Kategori Jawaban  | Sangat menarik | Cukup Menarik | Kurang Menarik |
|-------------------|----------------|---------------|----------------|
| Frekuensi Jawaban | 4              | 14            | 2              |

##### 4.3.5.2 Apakah aplikasi *Specification Maker* mudah dipelajari dan digunakan?

Tabel 4.19 Pertanyaan 2

| Kategori Jawaban  | Sangat Mudah | Cukup Mudah | Sulit |
|-------------------|--------------|-------------|-------|
| Frekuensi Jawaban | 5            | 15          |       |

##### 4.3.5.3 Apakah aplikasi *Specification Maker* memberikan informasi yang diharapkan dengan cepat dan mudah?

Tabel 4.20 Pertanyaan.3

|                   |              |             |              |
|-------------------|--------------|-------------|--------------|
| Kategori Jawaban  | Sangat Cepat | Cukup Cepat | Kurang Cepat |
| Frekuensi Jawaban | 4            | 16          |              |

4.3.5.4 Apakah aplikasi *Specification Maker* membantu dan memudahkan anda dalam membuat sebuah spesifikasi komputer?

Tabel 4.21 Pertanyaan 4

|                   |                 |                |                 |
|-------------------|-----------------|----------------|-----------------|
| Kategori Jawaban  | Sangat Membantu | Cukup Membantu | Kurang Membantu |
| Frekuensi Jawaban | 9               | 11             |                 |

Dari tabel-tabel hasil kuisioner diatas kemudian dibuat presentase jawaban yang iberikan oleh responden untuk setiap kategori pertanyaan dan jawaban yang diberikan. Presentase nilai dibuat dengan menggunakan rumus :

$$(P / n) * 100\% \quad \text{dimana:}$$

P = jumlah jawaban pada satu kategori pertanyaan dan jawaban

n = jumlah responden, yaitu 10 orang

Hasil perhitungan presentasenya seperti pada tabel 4.22.

Tabel 4.22 Perhitungan Presentase

| Kategori jawaban                       | Cepat/Mudah/Menarik | Cukup | Kurang |
|----------------------------------------|---------------------|-------|--------|
| Ketertarikan terhadap tampilan sistem  | 20%                 | 70%   | 10%    |
| Kemudahan mempelajari                  | 25%                 | 75%   | 0%     |
| Kemudahan informasi                    | 20%                 | 80%   | 0%     |
| Kemudahan membuat spesifikasi komputer | 45%                 | 55%   | 0%     |

#### 4.3.6 Kesimpulan Pengujian *Beta*

Berdasarkan hasil pengujian yang telah dilakukan dengan cara membagikan kuisioner kepada *user/pengguna Spesification Maker* kemudian melakukan perhitungan presentase sesuai dengan kategori jawaban, maka dapat disimpulkan bahwa aplikasi *Spesification Maker* mudah untuk digunakan, tampilan menarik, memberikan informasi yang diharapkan serta memudahkan membuat spesifikasi komputer.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Berdasarkan hasil pengujian sistem informasi *Specification Maker* yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Pelaksanaan kerja praktek ini berhasil merancang dan membangun sistem informasi *Specification Maker* yang dapat memberikan informasi mengenai data barang pada *stand* ASComputer secara cepat dan mudah, ditunjukkan dengan presentase kategori jawaban “kemudahan informasi” pada tabel 4.28 dimana 20% Sangat Cepat, 80% Cukup Cepat.
2. Pelaksanaan kerja praktek ini berhasil membangun sistem informasi *Specification Maker* yang dapat memberikan panduan dalam membuat spesifikasi komputer tanpa didampingi oleh penjaga *stand* ASComputer, ditunjukkan dengan presentase kategori jawaban “kemudahan membuat spesifikasi komputer” pada tabel 4.28 dimana 45% Sangat Membantu, 55% Cukup Membantu.

#### **5.2 Saran**

Adapun saran yang diberikan demi pengembangan sistem ini adalah:

1. Sistem ini dapat memberikan keluaran sebuah saran spesifikasi yang sesuai dengan *budget* yang *user* masukan.

2. Penambahan fitur yang memberikan saran untuk spesifikasi tertentu , seperti *office* dan *game*.
3. Sistem ini dapat memberikan penilaian terhadap spesifikasi komputer yang telah dibuat.

## DAFTAR PUSTAKA

- Alter, Steven. *Information System : A Management Perspective*. The Benjamin / Cummings Publishing Company, 1992.
- Davis, Gordon B. *Kerangka Dasar Sistem Informasi Manajemen Bagian I : Pengantar*. Diterj. oleh Andreas S. Adiwardana. Cetakan ke sebelas, PT Ikrar Mandiriabadi, 1999.
- Harianto, Bambang. *Esensi-Esensi Bahasa Pemrograman Java*. Bandung: Informatika, 2008.
- Kadir, Abdul. *Pengenalan Sistem Informasi*. Yogyakarta: ANDI, 2003.
- Miftakhul Huda, Bunafit Komputer. *Membuat Aplikasi Database dengan Java, MySQL dan Netbeans*. Jakarta: PT Elex Media Komputindo, 2010.
- Riyanto, Suprpto dan Hendi Indelarko. *Pengembangan Aplikasi Manajemen Database dengan Java 2 (SE/ME/EE)*. Yogyakarta: Gava Media, 2008.
- Pressman, Rogger. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi, 2002.

## LAMPIRAN

### Kode Program

#### 1 Metode untuk melakukan koneksi database

```
public void koneksi() {  
    config.setServer("localhost");  
    config.setUser("root");  
    config.setPassword("");  
    config.setDatabase("dbBarang");  
    config.makeConnect();  
}
```

#### 2. Metode untuk melakukan pengecekan *username* dan *password* pada *form login*

```
private void checkPassword() {  
    try {  
        Statement st = config.getConnection().createStatement();  
        ResultSet rs = st.executeQuery("Select * from akun where id_akun='"  
            + akun.getIdAkun()+ "' and pass = PASSWORD('" + akun.getPass() +  
            "')");  
        if (rs.next()) {  
            ad.show();  
            FormHome.this.dispose();  
        } else {  
            JOptionPane.showMessageDialog(this, "Password dan ID salah");  
            jPasswordFieldPass.setText("");  
            jTextFieldID.requestFocus();  
        }  
    } catch (SQLException ex) {  
        JOptionPane.showMessageDialog(null, ex, "Error", JOptionPane.ERROR_MESSAGE);  
    }  
}
```

### 3. Metode pengecekan *field-field* pada *form edit* sebelum menyimpan data.

```
private void checkSave() {
    if (jComboBoxKategori.getSelectedIndex() == 0) {
        JOptionPane.showMessageDialog(null, "Anda belum memilih Jenis Barang!!!");
        jComboBoxKategori.requestFocus();
    } else if (jTextFieldKodeBarang.getText().trim().isEmpty() == true) {
        JOptionPane.showMessageDialog(null, "Kode Barang Kosong!!!");
        jTextFieldKodeBarang.requestFocus();
    } else if (jTextFieldMerk.getText().trim().isEmpty() == true) {
        JOptionPane.showMessageDialog(null, "Merk Kosong!!!");
        jTextFieldMerk.requestFocus();
    } else if (jTextFieldHarga.getText().trim().isEmpty() == true) {
        JOptionPane.showMessageDialog(null, "Harga Kosong!!!");
        jTextFieldHarga.requestFocus();
    } else if (jComboBoxStock.getSelectedIndex() == 0) {
        JOptionPane.showMessageDialog(null, "Stock Ada / Kosong ?");
        jComboBoxStock.requestFocus();
    } else if (jTextAreaDetail.getText().trim().isEmpty() == true) {
        JOptionPane.showMessageDialog(null, "Detail Kosong!!!");
        jTextAreaDetail.requestFocus();
    } else {
        if (jComboBoxKategori.getSelectedIndex() == 1) {
            if (jComboBoxPC.getSelectedIndex() == 0) {
                JOptionPane.showMessageDialog(null, "PC Socket Kosong!!!");
                jComboBoxPC.requestFocus();
            } else if (jComboBoxMemory.getSelectedIndex() == 0) {
                JOptionPane.showMessageDialog(null, "Memory Socket Kosong!!!");
                jComboBoxMemory.requestFocus();
            } else {
                save();
            }
        } else if (jComboBoxKategori.getSelectedIndex() == 2) {
            if (jComboBoxPC.getSelectedIndex() == 0) {
                JOptionPane.showMessageDialog(null, "PC Socket Kosong!!!");
                jComboBoxPC.requestFocus();
            } else {
                save();
            }
        }
    }
}
```

```

    } else if (jComboBoxKategori.getSelectedIndex() == 3) {
        if (jComboBoxMemory.getSelectedIndex() == 0) {
            JOptionPane.showMessageDialog(null, "Memory Socket Kosong!!!");
            jComboBoxMemory.requestFocus();
        } else {
            save();
        }
    } else {
        save();
    }
}
}

```

4. Metode untuk menyimpan data, *query insert* atau *update* ditentukan oleh variable “konfirmasi”

```

private void save() {
    if (konfirmasi.equals("insert") == true) {
        try {
            Statement st = config.getConnection().createStatement();
            String sql;
            if (jRadioButtonDollar.isSelected() == true) {
                sql = "insert into barang values " + "("
                    + jTextFieldKodeBarang.getText() + "','"
                    + jTextFieldMerk.getText() + "','"
                    + jComboBoxPC.getSelectedIndex() + "','"
                    + jComboBoxMemory.getSelectedIndex() + "','"
                    + jTextFieldHarga.getText() + "','"
                    + jComboBoxStock.getSelectedIndex() + "','"
                    + jTextAreaDetail.getText() + "','"
                    + jComboBoxKategori.getSelectedIndex() + "','USD','"
                    + nGambar + "')";
            } else {
                sql = "insert into barang values " + "("
                    + jTextFieldKodeBarang.getText() + "','"
                    + jTextFieldMerk.getText() + "','"
                    + jComboBoxPC.getSelectedIndex() + "','"
                    + jComboBoxMemory.getSelectedIndex() + "','"
                    + jTextFieldHarga.getText() + "','"
                    + jComboBoxStock.getSelectedIndex() + "','"

```

```

        + jTextAreaDetail.getText() + "',''"
        + jComboBoxKategori.getSelectedIndex() + "','IDR','"
        + nGambar + "')";
    }
    int mAffect = st.executeUpdate(sql);
    if (mAffect > 0) {
        JOptionPane.showMessageDialog(this, "Insert Data Sukses");
        if (checkG.equals("ya")) {
            SimpanGambar();
        }
    } catch (SQLException ex) {
        jButtonUpdate.setEnabled(false);
    }
}

if (konfirmasi.equals("update") == true) {
    try {
        int app;
        if ((app = JOptionPane.showConfirmDialog(null, "Ubah data " +
            jTextFieldKodeBarang.getText() + " ?", "Perhatian",
            JOptionPane.YES_NO_OPTION)) == 0) {
            Statement st = config.getConnection().createStatement();
            String sql;
            if (jRadioButtonDollar.isSelected() == true) {
                sql = "update barang set nama_barang = '" + jTextFieldMerk.getText()
                + "', pc_sup = '" + jComboBoxPC.getSelectedIndex()
                + "', mem_sup = '" + jComboBoxMemory.getSelectedIndex()
                + "', harga = '" + jTextFieldHarga.getText()
                + "', stock = '" + jComboBoxStock.getSelectedItem()
                + "', detail = '" + jTextAreaDetail.getText()
                + "', matauang = 'USD', gambar = '" + nGambar + "' where kd_barang = '"
                + jTextFieldKodeBarang.getText() + "'";
            } else {
                sql = "update barang set nama_barang = '" + jTextFieldMerk.getText()
                + "', pc_sup = '" + jComboBoxPC.getSelectedIndex()
                + "', mem_sup = '" + jComboBoxMemory.getSelectedIndex()
                + "', harga = '" + jTextFieldHarga.getText()
                + "', stock = '" + jComboBoxStock.getSelectedItem()
                + "', detail = '" + jTextAreaDetail.getText()
                + "', matauang = 'IDR', gambar = '" + nGambar + "' where kd_barang = '"
                + jTextFieldKodeBarang.getText() + "'";
            }
        }
    }
}

```

```

        int mAffect = st.executeUpdate(sql);
        if (mAffect > 0) {
            //System.out.println(checkG);
            if (checkG.equals("ya")) {
                SimpanGambar();
            }
        }
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(null, ex, "Error", JOptionPane.ERROR_MESSAGE);
    }
}
refresh();
settingAwal();
}

```

5. Metode *refresh* merupakan metode yang digunakan untuk me-*refresh* tabel.

```

private void refresh() {
    try {
        int row = tableData.getRowCount();
        for (int i = 0; i < row; i++) {
            modelData.delete(0, row);
        }
        Statement st = config.getConnection().createStatement();
        String sql = "select * from barang where jenisbarang='" +
jComboBoxKategori.getSelectedIndex() + "'";
        ResultSet rs = st.executeQuery(sql);
        while (rs.next()) {
            data data = new data();
            data.setKd_barang(rs.getString(1));
            data.setMerk(rs.getString(2));
            data.setPc_sup(rs.getString(3));
            data.setMem_sup(rs.getString(4));
            data.setHarga(rs.getString(5));
            if (rs.getString(6).equals("Kosong")) {
                data.setStock("<html><font color=\"red\">Kosong</font></html>");
            } else {
                data.setStock(rs.getString(6));
            }
        }
    }
}

```

```

        }
        data.setDetail(rs.getString(7));
        data.setMatauang(rs.getString(9));
        data.setGambar(rs.getString(10));
        modelData.add(data);
    }
    new AturKolom(tableData);
    tableData.revalidate();
} catch (SQLException ex) {
JOptionPane.showMessageDialog(null, ex, "Error", JOptionPane.ERROR_MESSAGE);
}
modelData.refresh();
}

```

6. Metode *add* merupakan metode yang digunakan untuk mengambil data dari tabel barang ke daftar simulasi pada *form* simulasi.

```

private void Add() {
    try {
        int baris = tableData.getSelectedRow();
        data anggota = modelData.get(baris);
        double hargaS = Double.valueOf(anggota.getHarga());
        int harga;
        if (anggota.getMatauang().equals("USD") == true) {
            hargaS = (double) (hargaS * dollar);
        }
        harga = (int) hargaS;
        if (jComboBoxKategori.getSelectedIndex() == 1) {
            kodeMB = anggota.getKd_barang();
            hargaMB = harga;
            PCSocket = anggota.getPc_sup();
            RAMsocket = anggota.getMem_sup();
            jTextFieldRakitMB.setText(anggota.getMerk());
            jTextFieldMBHarga.setText(String.valueOf(harga));
            jSpinnerMB.setValue(1);
            jSpinnerMB.setEnabled(true);
        } else if (jComboBoxKategori.getSelectedIndex() == 2) {
            kodeCPU = anggota.getKd_barang();
            hargaCPU = harga;
            PCSocket = anggota.getPc_sup();

```

```

jTextFieldRakitCPU.setText(anggota.getMerk());
    jTextFieldCPUHarga.setText(String.valueOf(harga));
    jSpinnerCPU.setValue(1);
    jSpinnerCPU.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 3) {
    jTextFieldRakitMemory.setText(anggota.getMerk());
    jTextFieldMemoryHarga.setText(String.valueOf(harga));
    kodeMem = anggota.getKd_barang();
    hargaMem = harga;
    RAMsocket = anggota.getMem_sup();
    jSpinnerMemory.setValue(1);
    jSpinnerMemory.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 4) {
    jTextFieldRakitVGA.setText(anggota.getMerk());
    jTextFieldVGAHarga.setText(String.valueOf(harga));
    kodeVGA = anggota.getKd_barang();
    hargaVGA = harga;
    jSpinnerVGA.setValue(1);
    jSpinnerVGA.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 5) {
    jTextFieldRakitHD.setText(anggota.getMerk());
    jTextFieldHDHarga.setText(String.valueOf(harga));
    kodeHD = anggota.getKd_barang();
    hargaHD = harga;
    jSpinnerHD.setValue(1);
    jSpinnerHD.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 6) {
    jTextFieldRakitOptical.setText(anggota.getMerk());
    jTextFieldOpticalHarga.setText(String.valueOf(harga));
    kodeOptical = anggota.getKd_barang();
    hargaOptical = harga;
    jSpinnerOptical.setValue(1);
    jSpinnerOptical.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 7) {
    jTextFieldRakitCasing.setText(anggota.getMerk());
    jTextFieldCasingHarga.setText(String.valueOf(harga));
    kodeCasing = anggota.getKd_barang();
    hargaCasing = harga;
    jSpinnerCasing.setValue(1);
    jSpinnerCasing.setEnabled(true);
}

```

```

} else if (jComboBoxKategori.getSelectedIndex() == 8) {
    jTextFieldRakitLCD.setText(anggota.getMerk());
    jTextFieldLCDHarga.setText(String.valueOf(harga));
    kodeLCD = anggota.getKd_barang();
    hargaLCD = harga;
    jSpinnerLCD.setValue(1);
    jSpinnerLCD.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 9) {
    jTextFieldRakitMouse.setText(anggota.getMerk());
    jTextFieldMouseHarga.setText(String.valueOf(harga));
    kodeMouse = anggota.getKd_barang();
    hargaMouse = harga;
    jSpinnerMouse.setValue(1);
    jSpinnerMouse.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 10) {
    jTextFieldRakitKeyboard.setText(anggota.getMerk());
    jTextFieldKeyboardHarga.setText(String.valueOf(harga));
    kodeKeyboard = anggota.getKd_barang();
    hargaKeyboard = harga;
    jSpinnerKeyboard.setValue(1);
    jSpinnerKeyboard.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 11) {
    jTextFieldRakitPSU.setText(anggota.getMerk());
    jTextFieldPSUHarga.setText(String.valueOf(harga));
    kodePSU = anggota.getKd_barang();
    hargaPSU = harga;
    jSpinnerPSU.setValue(1);
    jSpinnerPSU.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 12) {
    jTextFieldRakitUPS.setText(anggota.getMerk());
    jTextFieldUPSHarga.setText(String.valueOf(harga));
    kodeUPS = anggota.getKd_barang();
    hargaUPS = harga;
    jSpinnerUPS.setValue(1);
    jSpinnerUPS.setEnabled(true);
} else if (jComboBoxKategori.getSelectedIndex() == 13) {
    jTextFieldRakitStabilizer.setText(anggota.getMerk());
    jTextFieldStabilizerHarga.setText(String.valueOf(harga));
    kodeStab = anggota.getKd_barang();
    hargaStab = harga;

```

```

        jSpinnerStabilizer.setValue(1);
        jSpinnerStabilizer.setEnabled(true);
    } else if (jComboBoxKategori.getSelectedIndex() > 13) { //tambahan
        data d = new data();
        int row = tableLain.getRowCount();
        int i, konf = 0;
        String checkA, checkB;
        for (i = 0; i < row; i++) {
            checkA = anggota.getKd_barang();
            checkB = String.valueOf(modelLain.getValueAt(i, 0));
            if (checkA.equals(checkB) == true) {
                int jml = Integer.parseInt(String.valueOf(modelLain.getValueAt(i, 2)));
                int priceT = harga * (jml + 1);
                modelLain.delete(i, row);
                d.setKd_barang(anggota.getKd_barang());
                d.setMerk(anggota.getMerk());
                d.setJumlah(String.valueOf(jml + 1));
                d.setHarga(String.valueOf(priceT));
                i = row;
                konf = 1;
            }
        }
        if (konf != 1) {
            d.setKd_barang(anggota.getKd_barang());
            d.setMerk(anggota.getMerk());
            d.setJumlah("1");
            d.setHarga(String.valueOf(harga));
        }
        modelLain.add(d);
        new AturKolom(tableLain);
        tableLain.revalidate();
        jScrollPane4.setVisible(true);
        jButtonDeleteLain.setVisible(true);
        jButton1.setVisible(true);
        jButton2.setVisible(true);
        hasilLain = hasilLain + harga;
    }
} catch (Exception e) {
    System.out.println(e + "Error");
} }

```