**BAB IV**

**ANALISA DAN PERANCANGAN**

* 1. **Analisa Sistem yang Dibangun**

Melihat sistem pelayanan jasa service komputer yang berjalan pada CV Nik.Com dirasa kurang optimal karena dalam pelayanan service komputer masih mempunyai beberapa kendala dalam sistem kerjanya. Kendala yang muncul dalam proses pelayanan ini adalah dalam proses pengolahan data saat ini sudah menggunakan komputer dan aplikasi *Ms. Excel* namun belum memanfaatkan sistem informasi berbasis *database* sehingga mengakibatkan kegiatan-kegiatan yang kurang terstruktur. Penyimpanan data membutuhkan ruang penyimpanan yang besar, memerlukan waktu cukup lama dalam pencarian data jika sewaktu-waktu diperlukan kembali. Pelaporan data service juga tidak tersimpan pada satu tempat penyimpanan yang mengakibatkan proses pelaporan nota kurang cepat. Oleh karena itu penulis ingin mengoptimalkan sistem yang berjalan sebelumnya dengan menggantinya menjadi sebuah sistem yang lebih efektif. Dengan adanya Sistem Informasi Pelayanan Jasa Service Komputer ini diharapkan dapat meningkatkan kinerja pada CV. Nik Com.

* 1. **Analisa Kebutuhan Data dan Informasi**

Untuk membangun sebuah sistem dibutuhkan adanya masukan berupa data yang nantinya akan diproses oleh sistem sehingga sistem dapat memberikan informasi yang bermanfaat kepada penggunanya. Kebutuhan data dan informasi untuk sistem informasi pendaftaran pelatihan adalah sebagai berikut:

1. Kebutuhan data

Data-data yang dibutuhkan antara lain:

1. Data user/pengguna sistem
2. Data konsumen
3. Data sparepart
4. Data kategori sparepart
5. Data service
6. Data transaksi
7. Kebutuhan informasi

Informasi yang ingin dihasilkan penulis adalah sebagai berikut:

1. Informasi data sparepart
2. Informasi riwayat transaksi data service
3. Nota transaksi service
   1. **Analisa Kebutuhan Perangkat Keras**

Perangkat keras yang dibutuhkan untuk implementasi sistem ini minimal memiliki spesifikasi sebagai berikut:

1. *Monitor*, segala bentuk dan merk. Disarankan monitor berwarna dan mampu beresolusi 1024X800 *pixel.*
2. *CPU* ( *Central Processing Unit*). Terdiri dari *processor, motherboard, harddisk, RAM, VGA,* dll.
3. *Input device.* Terdiri dari *Keyboard* dan *Mouse.*
4. *Output device.* Terdiri dari *Printer, Scanner* dll.
5. Koneksi internet. Perangkat pendukung lain seperti *printer, mouse, keyboard, scanner,* dll.
   1. **Analisa Kebutuhan Perangkat Lunak**

Sedangkan *Software* untuk pengembangan sistem ini antara lain sebagai berikut:

1. Sistem operasi *Windows* atau *Linux.*
2. *Sublime Text* untuk menulis *Script PHP.*
3. *Web Server (Apache)* dan *MySQL* dimana kedua *Software* tersebut sudah tersedia di dalam paket *Xampp.*
4. *Web browser* untuk menjalankan *web* misalnya *Google Chrome.*

   2. **Perancangan Pemodelan Sistem**
5. **Analisa Aktor Sistem**

Aktor menggambarkan semua pengguna sistem. Aktor dalam Sistem Informasi Pelayanan Jasa Service Komputer pada CV. Nik Com antara lain sebagai berikut:

Konsumen

Konsumen merupakan aktor yang melakukan transaksi di CV. Nik Com.

* + - 1. Admin

Administrasi adalah orang yang bertugas dan memiliki hak akses untuk melakukan operasi pengelolaan konsumen, data sparepart, data service, melayani konsumen dan laporan.

* + - 1. Teknisi

Teknisi merupakan orang yang bertugas sebagai eksekutor perbaikan komputer dan sekaligus pengguna sistem yang bertugas sebagai pemberi informasi kerusakan komputer kepada konsumen melalui Admin.

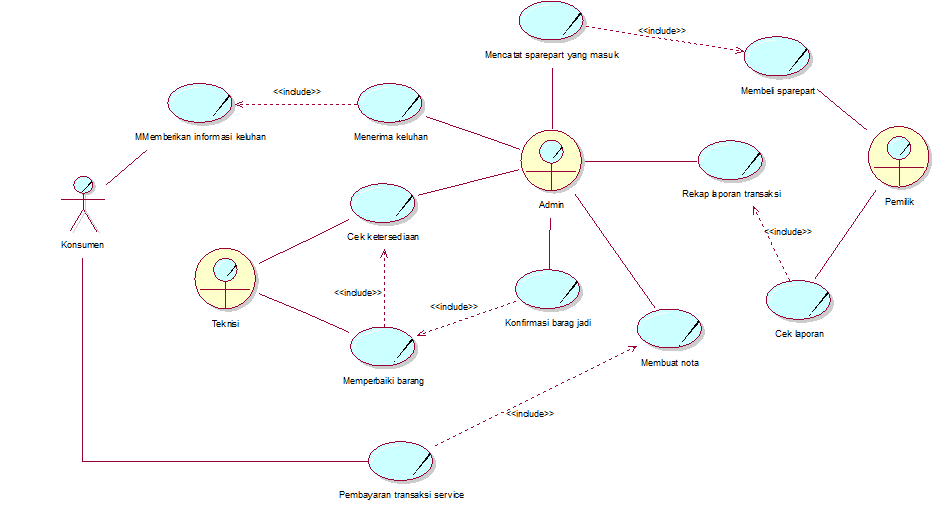
* + - 1. Pemilik

Pemilik adalah orang yang mempunyai kebijakan dan hak akses dalam proses pemesanan sparepart dan memeriksa laporan transaksi yang dibuat oleh admin.

.

1. ***Business use Case***

Diagram *Use Case* bisnis memperlihatkan hubungan-hubungan antara aktor-aktor bisnis, use case bisnis, dan pekerja-pekerja bisnis dalam organisasi. Diagram ini memberi model lengkap tentang apa yang dilakukan organisasi, siapa yang ada didalam organisasi, dan diluar organisasi. Gambaran *business use case* yang menggambarkan tentang proses yang dilakukan oleh pekerja bisnis maupun aktor bisnis dapat dilihat pada diagram business use case yang ada pada gambar 4.1 berikut ini.



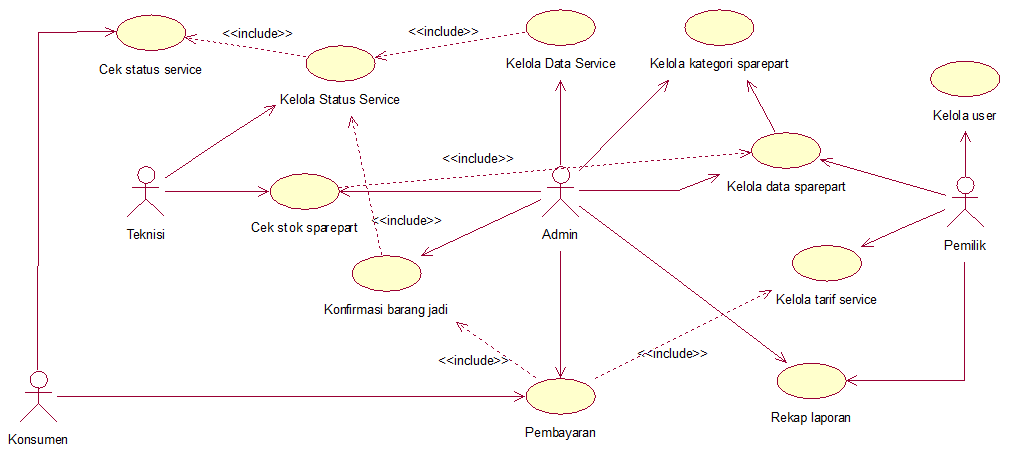
**Gambar 4. 1 Diagram Business Use case Pelayanan Service Komputer CV. Nik Com**

Dari diagram *business use case* pada gambar 4.1 diatas, maka diperoleh gambaran aktifitas bisnis seperti pada tabel 4.1 berikut:

**Tabel 4. 1 Gambaran Aktifitas Bisnis**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Proses Bisnis** | **Aktor** | ***Business Use Case*** |
| 1 | Pemilik membeli sparepart | Pemilik | Membeli sparepart |
| 2 | Admin mencatat sparepart yang masuk | Admin | Mencatat sparepart yang masuk |
| 3 | Konsumen memberikan informasi keluhan komputer | Konsumen | Memberikan informasi keluhan |
| 4 | Admin menerima keluhan komputer | Admin | Menerima keluhan |
| 5 | Admin dan Teknisi memeriksa ketersediaan sparepart | Admin, Teknisi | Cek ketersediaan |
| 6 | Teknisi memperbaiki sparepart atau kerusakan komputer | Teknisi | Memperbaiki sparepart |
| 7 | Admin melakukan konfirmasi sparepart jadi | Admin | Konfirmasi sparepart jadi |
| 8 | Admin membuat nota | Admin | Membuat nota |
| 9 | Konsumen melakukan pembayaran transaksi service | Konsumen | Pembayaran transaksi service |
| 10 | Admin melakukan rekap laporan transaksi | Admin | Rekap laporan transaksi |
| 11 | Pemilik memeriksa laporan transaksi | Pemilik | Memeriksa laporan transaksi |

* + 1. **Sistem use case**

 Diagram sistem *use case* akan menjelaskan mengenai siapa saja yang terlibat dalam sistem (aktor) dan apa saja yang dikerjakan oleh sistem (*use case*). Adapun diagram sistem *use case* yang terbentuk dapat dilihat pada gambar 4.2 berikut ini.

**Gambar 4. 2 Diagram Sistem Use Case Pelayanan Service Komputer CV. Nik Com**

* + 1. **Skenario *Use Case (Flow of Event)***

*Use Case* menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan. Dari Sistem *Use Case* diatas, maka dapat dibuat beberapa skenario antara lain:

**Skenario *Use Case* Kelola *User***

Skenario dari *Use Case* Kelola *User* dapat dilihat pada tabel 4.2 berikut ini.

**Tabel 4. 2 Skenario Use Case Kelola User**

|  |  |
| --- | --- |
| **Aktor Utama :** Pemilik toko  **Nama Use Case :** Kelola data user  **Tujuan Use Case :** Menambahkan data karyawan bagian administrasi dan teknisi.  **Kondisi Awal :** Data karyawan bagian administrasi dan teknisi masih kosong  **Kondisi Akhir :** Data karyawan bagian administrasi dan teknisi sudah terisi dan karyawan bisa login | ***Optimistic flow :***   1. Pemilik melakukan *login* ke sistem 2. Pemilik memilih menu *user* 3. Tampil *form user* 4. Pemilik menambah *user* 5. Pemilik memasukkan data User Pada *form* input data *User*. 6. Pemilik menyimpan data *user* 7. Sistem menyimpan data *user*   ***Pesimistic flow :***  1a, 4a *Login* gagal  **Solusi:** Masukkan *username* dan *password* dengan benar.  7a Simpan data gagal  **Solusi:** Periksa koneksi internet, ulang kembali proses simpan |

**Skenario *Use Case* Kelola Data Sparepart**

Skenario dari *Use Case* Kelola Data Sparepart dapat dilihat pada tabel 4.3 berikut ini.

**Tabel 4. 3 Skenario Use Case Kelola Data Sparepart**

|  |  |
| --- | --- |
| **Aktor Utama :** Admin, Pemilik  **Nama Use Case :** Kelola data sparepart  **Tujuan Use Case :** Menambahkan data detail sparepart yang ada di toko.  **Kondisi Awal :** Data sparepart belum terisi atau masih kosong.  **Kondisi Akhir :** Data sparepart sudah terisi dan sudah siap untuk melakukan transaksi penjualan. | ***Optimistic flow :***   1. Admin melakukan *login* ke sistem. 2. Admin memilih menu sparepart. 3. Tampil *form* sparepart 4. Admin memasukkan data sparepart pada *form* input data sparepart 5. Admin menyimpan data sparepart 6. Sistem menyimpan data sparepart 7. Pemilik memeriksa sparepart yag diinputkan   ***Pesimistic flow :***  1b. *Login* gagal  Solusi:Masukkan *username* dan *password* dengan benar.  5b,6b Simpan data gagal  **Solusi:** Periksa lagi data yang akan dimasukkan, ulang kembali proses simpan.  . |

**Skenario *Use Case* Kelola Tarif *Service***

Skenario dari *Use Case* Kelola Tarif *Service* dapat dilihat pada tabel 4.4 berikut ini.

**Tabel 4.4. Skenario Proses Kelola Tarif *Service***

|  |  |
| --- | --- |
| **Aktor Utama:** Pemilik  **Nama Use Case:** Kelola tarif *service*  **Tujuan Use Case:** Mengelola tarif *service* dengan ketentuan kerusakan umum yang sering terjadi.  **Kondisi Awal:** Tarif *service* belum diinputkan dan belum siap digunakan untuk transaksi *service*.  **Kondisi Akhir:** Data tarif *service* sudah terisi dan siap digunakan untuk melakukan transaksi *service*. | ***Optimistic flow* :**   1. Pemilik Pemilik melakukan *login* ke sistem. 2. Pemilik memilih menu tarif *service*. 3. Tampil *form* tarif service 4. Pemilik memasukkan data tarif service pada *form* input data tarif service. 5. Pemilik menyimpan data tarif service. 6. Sistem menyimpan data tarif service.   ***Pesimistic flow* :**  1c. *Login* gagal  **Solusi:** Masukkan *username* dan *password* dengan benar.  4c,5c Simpan data gagal  **Solusi:** Periksa lagi data yang akan dimasukkan, ulang kembali proses simpan. |

**Skenario *Use Case* Kategori Sparepart**

Lanjutan Tabel 3.7

Lanjutan Tabel 3.7

Skenario *use case* Kategori Sparepart untuk lebih jelasnya dapat dilihat pada tabel 4.5 berikut.

**Tabel 4.5. Skenario *Use Case* Kategori Sparepart**

|  |  |
| --- | --- |
| **Aktor Utama :** Admin  **Aktor Tambahan :** -  **Nama *Use Case* :** Kategori  **Tujuan *Use Case* :** Melakukan aktifitas data kategori  **Kondisi Awal :** Admin membuka form kategori  **Kondisi Akhir :** Sistem berhasil menyimpan data kategori | **Alur Optimistik :**   1. Admin membuka form kategori 2. Sistem menampilkan halaman kategori 3. Admin memasukkan data kategori 4. Sistem berhasil menyimpan data kategori 5. Admin mengubah data kategori 6. Sistem berhasil mengubah data kategori 7. Admin menghapus data kategori 8. Sistem berhasil menghapus data kategori   **Alur Pesimistik :**  5d. Data kategori gagal untuk disimpan.  7d. Ubah data kategori gagal  **Solusi :**  5d. Admin memeriksa kembali inputan data kategori sudah benar atau belum, kemudian ulangi proses simpan.  6d. Admin memeriksa kembali inputan data kategori sudah benar atau belum, kemudian ulangi proses ubah. |

**Skenario *Use Case* Kelola Data *Service***

Skenario dari *Use Case* Kelola Data *Service* dapat dilihat pada tabel 4.6 berikut ini.

**Tabel 4.6 Skenario Use Case Kelola Data *Service***

|  |  |
| --- | --- |
| **Aktor Utama :** Admin  **Aktor Tambahan :** -  **Nama *Use Case*:**Kelola Data Service  **Tujuan Use Case:** Membuat transaksi *service*, mencetak nota bukti *service* dan transaksi *service*.  **Kondisi Awal:** Transaksi *service* belum dibuat.  **Kondisi Akhir:** Mencetak bukti transaksi *service*. | ***Optimistic flow*:**   * 1. Nomer bukti *service* berhasil dibuat dan siap digunakan.   2. Nota bukti service berhasil dicetak   ***Pesimistic flow*:**  1e. Terdapat kesalahan dalam pembuatan transaksi.  **Solusi:** Membatalkan transaksi dan membuat transaksi baru.  2e. Nota tidak dapat dicetak.  **Solusi:** periksa *device* printer apakah masih normal atau tidak, seperti tinta dan catride. |

**Skenario Use Case Kelola Status *Service***

Skenario dari *Use Case* Kelola Status *Service* dapat dilihat pada tabel 4.7 berikut ini.

**Tabel 4. 7 Skenario Use Case Kelola Status *Service***

|  |  |
| --- | --- |
| **Aktor Utama:** Teknisi  **Nama Use Case:** Kelola status service  **Tujuan Use Case:** Menambahkan histori perbaikan atau service didalam sistem untuk informasi konsumen.  **Kondisi Awal:** Tidak ada histori perbaikan.  **Kondisi Akhir:** Ada histori perbaikan dan bisa di akses oleh konsumen. | ***Optimistic flow*:**   * + 1. Konsumen dapat melihat histori perbaikan.   ***Pesimistic flow*:**  1f. Konsumen tidak bisa mengakses histori perbaikan.  **Solusi:** Merefresh browser atau mengecek satu persatu kode bukti *service*. |

**Skenario Use Case Cek Status *Service***

Skenario dari *Use Case* Cek Status Service dapat dilihat pada tabel 4.8 berikut ini.

**Tabel 4. 8 Skenario Use Case Cek Status *Service***

|  |  |
| --- | --- |
| **Aktor Utama:** Konsumen  **Nama Use Case:** Cek status *service*  **Tujuan Use Case:** Melihat histori perbaikan atau service sparepart.  **Kondisi Awal:** Belum meng-inputkan kode nota *service*.  **Kondisi Akhir :** Tampil data histori perbaikan dari sparepart milik konsumen. | ***Optimistic flow*:**   * + - 1. Konsumen dapat melihat histori perbaikan.   ***Pesimistic flow*:**  1g. Konsumen tidak bisa mengakses histori perbaikan.  **Solusi:** Merefresh browser atau mengecek satu persatu kode bukti *service*. |

**Skenario Use Case Cek Stok Sparepart**

Skenario dari *Use Case* Cek Stok Sparepart dapat dilihat pada tabel 4.9 berikut ini.

**Tabel 4. 9 Skenario Use Case Cek Stok Sparepart**

|  |  |
| --- | --- |
| **Aktor Utama:** Admin, Teknisi  ***A*ktor Tambahan:**  **Nama *Use Case*:**Kelola Cek Stok Sparepart  **Tujuan *Use Case*:** Melakukan pengecekan sparepart sebelum service, apakah masih tersedia atau tidak  **Kondisi Awal:** Belum melakukan pengecekan sparepart.  **Kondisi Akhir:** Data update sparepart dicatat didalam sistem dan bisa digunakan apabila ada perbaikan komputer. | ***Optimistic flow :***   1. Admin melakukan *login.* 2. Admin membuka menu cek stok sparepart. 3. Tampil *form* cek stok sparepart. 4. Admin meng*update* data cek stok sparepart. 5. Sistem menyimpan cek stok sparepart. 6. Teknisi melihat sparepart yang masih tersedia. 7. ***Pesimistic flow*:**   5h. Simpan data gagal  **Solusi:** Periksa lagi data yang akan dimasukkan, ulang kembali prosesnya. |

**Skenario Use Case Kelola Konfirmasi Sparepart Jadi**

Skenario dari *Use Case* Kelola Konfirmasi Sparepart Jadi dapat dilihat pada tabel 4.10 berikut ini.

**Tabel 4. 10 Skenario Use Case Konfirmasi Sparepart Jadi**

|  |  |
| --- | --- |
| **Aktor Utama:** Konsumen  **Nama Use Case:** Kelola Konfirmasi Sparepart  **Tujuan Use Case:** Memberikan informasi kepada konsumen bahwa sparepart yang di *service* sudah selesai.  **Kondisi Awal:** Belum ada konfirmasi sparepart sudah selesai.  **Kondisi Akhir:** Ada konfirmasi bahwa sparepart sudah selesai diperbaiki. | ***Optimistic flow :***   1. Konsumen melakukan login. 2. Konsumen membuka menu konfirmasi sparepart jadi. 3. Konsumen mendapatkan konfirmasi sparepart sudah selesai diperbaiki.   ***Pesimistic flow*:**  3i. Tidak dapat mengirim konfirmasi.  **Solusi:** Merefresh browser atau mengecek credit pulsa yang ada didalam kartu perdana yang terhubung dengan sistem. |

**Skenario Use Case Kelola Pembayaran**

Skenario dari *Use Case* Kelola Pembayarandapat dilihat pada tabel 4.11 berikut ini.

**Tabel 4.11 Skenario Use Case Pembayaran**

|  |  |
| --- | --- |
| **Aktor Utama :** Konsumen, Admin  **Aktor Tambahan :** -  **Nama Use Case :** Pembayaran  **Tujuan Use Case :** Melakukan aktifitas transaksi pembayaran  **Kondisi Awal :** Konsumen membuka form transaksi pembayaran  **Kondisi Akhir :** Sistem berhasil menyimpan data transaksi pembayaran | ***Optimistic flow :***   1. Konsumenmasuk ke menu form Pembayaran 2. Sistem menampilkan Pembayaran yang harus dibayar oleh Konsumen 3. Konsumenmengupload bukti pembayaran pada form yang tersedia 4. Konsumen menyimpan data bukti pembayaran 5. Sistem berhasil menyimpan data bukti pembayaran 6. Admin masuk ke menu form Pembayaran 7. Sistem menampilkan Pembayaran 8. Admin mengubah status pembayaran 9. Sistem berhasil mengubah status pembayaran   ***Pesimistic flow*:**  5j. Simpan data bukti pembayaran gagal  4).j. Ubah status data pembayaran gagal  **Solusi :**  5j. Mengulangi proses transaksi pembayaran  4)j. Mengulangi proses ubah status pembayaran dari *Konsumen* |

**Skenario *Use Case* Rekap Laporan**

Skenario dari *Use Case* Rekap Laporan dapat dilihat pada tabel 4.12 berikut ini.

|  |  |
| --- | --- |
| **Aktor Utama :** Admin, Pemilik  **Aktor Tambahan :** -  **Nama *Use Case :***Kelola rekap laporan  **Tujuan *Use Case* :** Membuat laporan data transaksi service  **Kondisi Awal :** Akan ditampilkan form laporan transaksi service  **Kondisi Akhir :** Laporan transaksi service tercertak | ***Optimistic flow :***   1. Admin atau Pemilik melakukan *login*. 2. Admin atau Pemilik membuka menu Laporan. 3. Pemilik memantau data laporan peserta pelatihan. 4. Admin mencari data yang akan dicetak. 5. Data yang dicari ditampilkan. 6. Cetak Laporan Pelatihan 7. ***Pesimistic flow :***   1k*. Login* gagal  **Solusi:** Masukkan *username* dan *password* dengan benar.  4k. Data yang akan dicetak tidak ada  **Solusi:** Memeriksa kembali pencarian data, apakah data tersimpan atau tidak.  6k. Cetak gagal  Solusi: Cek kondisi *printer* atau ganti *printer* lain. |

**Tabel 4.12 Skenario Use Case Laporan**

* + 1. ***Class Diagram***

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan berguna untuk membangun sistem.

* + - 1. ***Class* *User***

*Class user* berfungsi menampung data pengguna untuk kepentingan *login* ke sistem. *Class user* dapat dilihat pada gambar 4.3 dibawah ini:

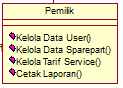


Gambar 4.3 *Class User*

*Class user* selanjutnya diturunkan menjadi 4 *class*,yaitu :

1. *Class* Pemilik Toko

*Class* pemilik toko merupakan *Class* turunan dari *Class user,* seperti pada gambar 4.4 berikut:



Gambar 4.4 *Class* Pemilik Toko

1. *Class* Teknisi

*Class* Teknisi merupakan *Class* turunan dari *Class user,* seperti pada gambar 4.5 berikut:

\

Gambar 4.5 *Class* Teknisi

1. *Class* Admin

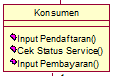
*Class* Admin merupakan *Class* turunan dari *Class user,* seperti pada gambar 4.6 berikut:



Gambar 4.6 *Class* Administrasi

1. *Class* Konsumen

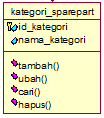
*Class* Konsumen merupakan *Class* turunan dari *Class user,* seperti pada gambar 4.7 berikut :



Gambar 4.7 *Class* Konsumen

1. ***Class* Kategori Sparepart**

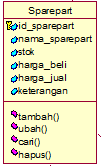
*Class* Kategori Sparepart berfungsi menampung data kategori sparepart. *Class* sparepart dapat dilihat pada gambar 4.8 dibawah ini :



Gambar 4.8 *Class* Kategori Sparepart

1. ***Class* Sparepart**

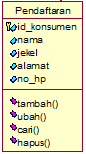
*Class* Sparepart berfungsi menampung data sparepart. *Class* sparepart dapat dilihat pada gambar 4.9 dibawah ini :



Gambar 4.9 *Class* Sparepart

1. ***Class* Pendaftaran**

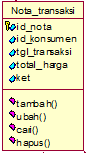
*Class* pendaftaran merupakan class yang digunakan untuk menampung data pendaftaran konsumen. Adapun penggambarannya dapat dilihat pada gambar 3.10 berikut.



Gambar 4.10 *Class* Pendaftaran

1. ***Class* Nota Transaksi**

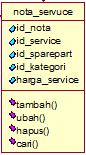
*Class* Nota Transaksi berfungsi menampung data transaksi penjualan ataupun servise yang dilakukan konsumen. *Class* nota transaksi dapat dilihat pada gambar 4.11 dibawah ini.



Gambar 4.11 *Class* Nota Transaksi

1. ***Class* Nota *Service***

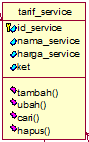
*Class* Nota *Service* berfungsi menampung data detail service yang digunakan oleh konsumen. *Class* nota *service* dapat dilihat pada gambar 4.12 dibawah ini.



Gambar 4.12 *Class* Nota *Service*

1. ***Class* Tarif *Service***

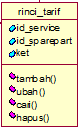
*Class* Tarif *Service* berfungsi menampung data tarif *service*. *Class* tarif *service* dapat dilihat pada gambar 4.13 dibawah ini.



Gambar 4.13 *Class* Tarif *Service*

1. ***Class* Rinci Tarif**

*Class* Rinci Tarif berfungsi menampung data rincian sparepart atau sparepart yang digunakan dalam *service*. *Class* rincitarif dapat dilihat pada gambar 4.14 dibawah ini.



Gambar 4.14 *Class* Rinci Tarif

1. ***Class* Histori Perbaikan**

*Class* Histori Perbaikan berfungsi menampung histori perbaikan atau sampai dimana sparepart diperbaiki oleh teknisi. *Class* histori perbaikan dapat dilihat pada gambar 4.15 dibawah ini.



Gambar 4.15 *Class* Histori Perbaikan

Setelah menganalisa kelas yang dibutuhkan dalam pembuatan rancangan sistem, maka class diagram Sistem Informasi Pengelolaan Service Komputer di CV. NikCom Berbasis Web dapat dilihat pada gambar 4.16 berikut ini.



**Gambar 4.16 *Class Diagram* Sistem Informasi Pengelolaan Service Komputer CV. Nik Com**

* + 1. ***Sequence Diagram***

*Sequence Diagram* menjelaskan hubungan antara komponen dalam sistem yang menghasilkan informasi berupa pesan. Adapun *Sequence Diagram* yang dapat dilihat sebagai berikut :

* + - 1. *Sequence Diagram* Kelola Data User

Menjelaskan bagaimana pemilik toko mengelola data pegawai. *Sequence diagram* kelola data user dapat dilihat pada gambar 4.17 berikut ini :



Gambar 4.17 *Sequence Diagram* Kelola Data User

* + - 1. *Sequence Diagram* Kelola Kategori Sparepart

Menjelaskan bagaimana admin mengelola kategori sparepart. *Sequence diagram* kelola kategori sparepart dapat dilihat pada gambar 4.18 berikut ini :



Gambar 4.18 *Sequence Diagram* Kelola Data Sparepart

* + - 1. *Sequence Diagram* Kelola Data Sparepart

Menjelaskan bagaimana pemilik toko mengelola data sparepart. *Sequence diagram* kelola sparepart dapat dilihat pada gambar 4.19 berikut ini :



Gambar 4.18 *Sequence Diagram* Kelola Data Sparepart

* + - 1. *Sequence Diagram* Tarif Service

Menjelaskan bagaimana pemilik toko mengelola tarif *service*. *Sequence diagram* kelola tarif *service* dapat dilihat pada gambar 4.19 berikut ini :



Gambar 4.19 *Sequence Diagram* Tarif Service

* + - 1. *Sequence Diagram* Pendaftaran

Menjelaskan bagimana aktivitas dari aktor *client* dalam melakukan pendaftaran *user* sistem dengan mengisikan biodata secara lengkap. Adapun penggambarannya dapat dilihat pada gambar 4.20 berikut.



Gambar 4.20 Sequence Diagram Pendaftaran

* + - 1. *Sequence Diagram* Transaksi Penjualan

Menjelaskan bagaimana admin melakukan transaksi penjualan dengan konsumen. *Sequence diagram* Transaksi Penjualan dapat dilihat pada gambar 4.21 berikut ini :



Gambar 4.21 *Sequence Diagram* Transaksi Penjualan

* + - 1. *Sequence Diagram* Kelola Data *Service*

Menjelaskan bagaimana admin melakukan pencatatan sparepart service dari konsumen. *Sequence diagram* Kelola Data *Service* dapat dilihat pada gambar 4.22 berikut ini :



Gambar 4.23 *Sequence Diagram* Kelola Data *Service*

* + - 1. *Sequence Diagram* Kelola Status *Service*

Menjelaskan bagaimana teknisi melakukan pencatatan histori perbaikan pada sparepart service dari konsumen. *Sequence diagram* Kelola Status *Service* dapat dilihat pada gambar 4.24 berikut ini :



Gambar 4.24 *Sequence Diagram* Kelola Status *Service*

* + - 1. *Sequence Diagram* Cek Status *Service*

Menjelaskan bagaimana konsumen melakukan pemantauan histori perbaikan dari sparepart miliknya. *Sequence diagram* Cek Status *Service* dapat dilihat pada gambar 4.25 berikut ini :



Gambar 4.25 *Sequence Diagram* Cek Status *Service*

* + - 1. *Sequence Diagram* Konfirmasi Sparepart Jadi

Menjelaskan bagaimana admin melakukan konfirmasi sparepart jadi. *Sequence diagram* Konfirmasi Sparepart Jadi dapat dilihat pada gambar 4.26 berikut ini :



Gambar 4.26 *Sequence Diagram* Konfirmasi Sparepart Jadi

* + - 1. *Sequence Diagram* Melakukan Transaksi Service

Menjelaskan bagaimana admin melakukan transaksi service dengan konsumen dan konsumen mengambil sparepart service. *Sequence diagram* melakukan transaksi *service* dapat dilihat pada gambar 4.27 berikut ini :



Gambar 4.27 *Sequence Diagram* Melakukan Transaksi *Service*

* + - 1. *Sequence Diagram* Cek Laporan Transaksi

Menjelaskan bagaimana pemilik melakukan pengecekan laporan transaksi bulanan atau sesuai tanggal yang diinginkan. *Sequence diagram* Cek Laporan Transaksi dapat dilihat pada gambar 4.28 berikut ini :



Gambar 4.28 *Sequence Diagram* Cek Laporan Transaksi

* + 1. ***Activity Diagram***

*Activity diagram* menggambarkan berbagai aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing aliran berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Adapun *Activity Diagram*nya adalah sebagai berikut.

* 1. ***Activity Diagram* Kelola User**

Pada *activity* *diagram* kelola *user* ini menjelaskan tentang bagaimana aktivitas dari pemilik toko dalam mengelola data *user*. Adapun penggambarannya seperti gambar 4.29 berikut.

Gambar 4.29 Activity Diagram Kelola User

* 1. ***Activity Diagram* Kelola Kategori Sparepart**

Pada proses ini menjelaskan bagaimana Admin melakukan kelola kategori untuk sparepart agar bisa digunakan untuk transaksi, seperti pada gambar 4.30 berikut ini :



Gambar 4.30 *Activity Diagram* Kelola Kategori Sparepart

* 1. ***Activity Diagram* Kelola Data Sparepart**

Pada proses ini menjelaskan bagaimana Pemilik Toko melakukan kelola sparepart untuk bisa digunakan untuk transaksi, seperti pada gambar 4.31 berikut ini :



Gambar 4.31 *Activity Diagram* Kelola Data Sparepart

* 1. ***Activity Diagram* Kelola Tarif *Service***

Pada proses ini menjelaskan bagaimana pemilik toko mengelola tarif *service* dengan data perbaikan secara umum, seperti pada gambar 4.32 berikut ini :



Gambar 4.32 *Activity Diagram* Kelola Tarif *Sevice*

* 1. ***Activity Diagram* Pendaftaran**

*Activity diagram* pendaftaran menggambarkan tentang bagaimana aktivitas dari konsumen dalam melakukan pendaftaran akun. Adapun penggambarannya seperti gambar 4.33 berikut.



Gambar 4.33 Activity Diagram Pendaftaran

* 1. ***Activity Diagram* Transaksi Penjualan**

Pada proses ini menjelaskan bagaimana admin melakukan transksi penjualan dengan konsumen, seperti pada gambar 4.34 berikut ini :

Gambar 4.34 *Activity Diagram* Transaksi Penjualan

* 1. ***Activity Diagram* Status *Service***

Pada proses ini menjelaskan bagaimana teknisi menambahkan histori perbaikan sparepart yang di *service*, seperti pada gambar 4.35 berikut ini :



Gambar 4.35 *Activity Diagram* Status *Service*

* 1. ***Activity Diagram* Cek Status *Service***

Pada proses ini menjelaskan bagaimana konsumen melihat histori perbaikan sparepart yang di *service*, seperti pada gambar 4.36 berikut ini :



Gambar 4.36 *Activity Diagram Cek* Status *Service*

* 1. ***Activity Diagram* Konfirmasi Sparepart**

Pada proses ini menjelaskan bagaimana admin memberikan informasi tentang sparepart sudah jadi dan bisa diambil kepada konsumen, seperti pada gambar 4.37 berikut ini :



Gambar 4.37 *Activity Diagram* Konfirmasi Sparepart Jadi

* 1. ***Activity Diagram* Melakukan Transaksi *Service***

Pada proses ini menjelaskan bagaimana admin mengelola transaksi service untuk pengambilan sparepart dan pembayaran, seperti pada gambar 4.38 berikut ini :

****

Gambar 4.38 *Activity Diagram* Melakukan Transaksi *Service*

* 1. ***Activity Diagram* Kelola Laporan**

Pada proses ini menggambarkan mengenai aliran aktivitas dari admin dalam mengelola laporan, pemilik toko mencetak laporan. Adapun penggambarannya seperti gambar 4.39 berikut.



Gambar 4.39 *Activity Diagram* Kelola Laporan

* + 1. ***Statechart Diagram***

*Statechart diagram* digunakan untuk menggambarkan perubahan status atau transisi dari sebuah mesin atau sistem. Berikut adalah *statechart diagram* dari Sistem Informasi Pengelolaan Service Komputer di CV. Nik Com *Berbasis Web*.

1. ***Statechart Diagram Class User***

*Statechart diagram user* menjelaskan aktifitas yang dapat dilakukan oleh pemilik toko terhadap sistem dalam siapa saja yang bisa mengelola sistem dengan batasan-batasannya.

***Statechart diagram* tambah *user***

Adapun penggambarannya seperti gambar 4.40 berikut.



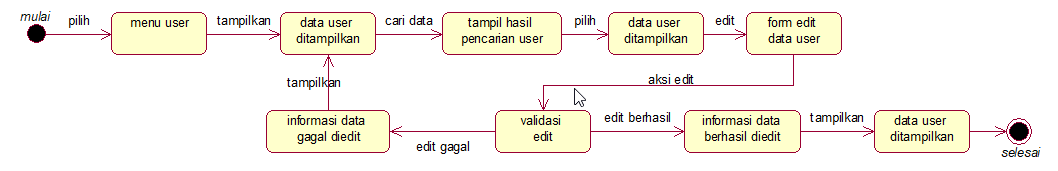
Gambar 4.40 Statechart diagram method tambah user

1. ***Statechart diagram* cari *user***

Adapun penggambarannya seperti gambar 4.41 berikut.

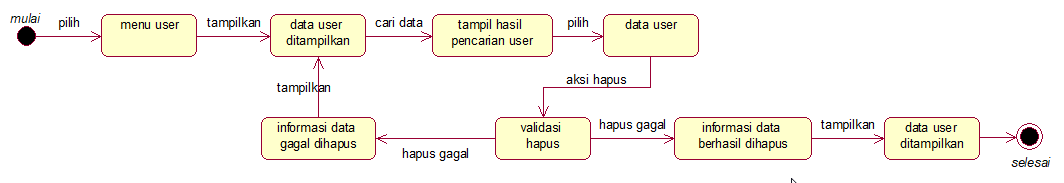
Gambar 4.41 Statechart diagram method cari user

1. ***Statechart diagram edit user***

Adapun penggambarannya seperti gambar 4.42 berikut.

Gambar 4.42 Statechart diagram method edit user

1. ***Statechart diagram* hapus *user***

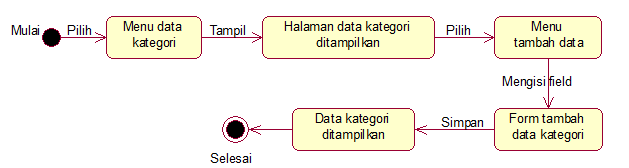
Adapun penggambarannya seperti gambar 4.43 berikut.

Gambar 4.43 Statechart diagram method hapus user

1. ***Statechart Diagram* Kategori**

*Statechart diagram* kategori menjelaskan aktifitas yang dapat dilakukan oleh admin terhadap sistem dalam mengelola form kategori.

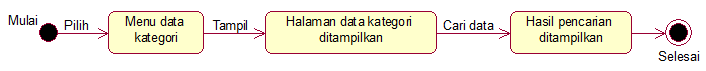
* 1. ***Statechart diagram* tambah kategori**

Adapun penggambarannya seperti gambar 4.44 berikut.

Gambar 4.44 Statechart diagram method tambah kategori

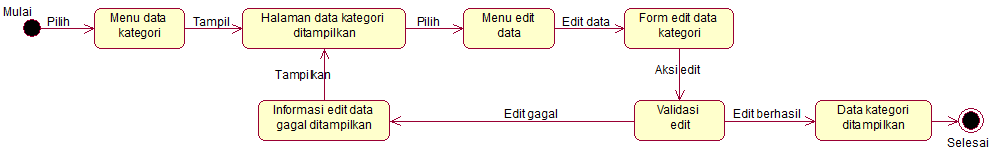
* 1. ***Statechart diagram* cari kategori**

Adapun penggambarannya seperti genap 4.45 berikut.



Gambar 4.45 *Statechart diagram method* cari kategori

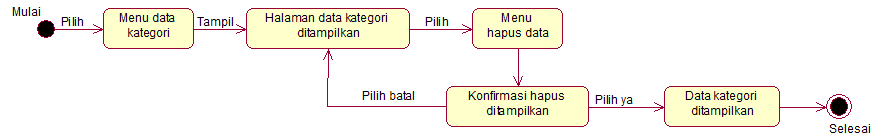
* 1. ***Statechart diagram method* edit kategori**

 Adapun penggambarannya seperti gambar 4.46 berikut.

Gambar 4.46 Statechart diagram method edit kategori

* 1. ***Statechart diagram method* hapus kategori**

Adapun penggambarannya seperti gambar 4.47 berikut.



Gambar 4.47 Statechart diagram method hapus kategori

1. ***Statechart diagram class* sparepart**

*Statechart diagram class* sparepartmenjelaskan aktifitas yang dilakukan pemilik toko dalam mengelola data sparepart.

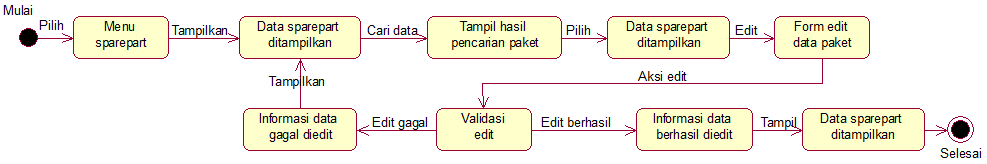
* + 1. ***Statechart diagram method* tambah sparepart**

Adapun seperti gambar 4.48 sebagai berikut ini :



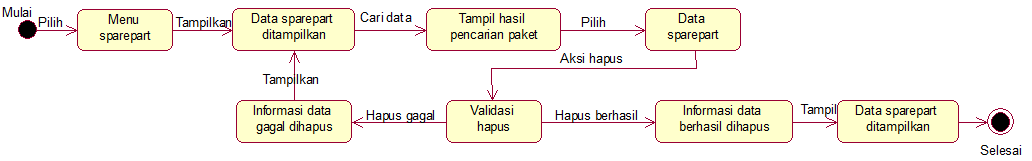
Gambar 4.48 *Statechart diagram method* tambah sparepart

* + 1. ***Statechart diagram method* edit sparepart**

 Adapun seperti gambar 4.49 sebagai berikut ini :

Gambar 4.49 *Statechart diagram method* edit sparepart

* + 1. ***Statechart diagram method* hapus sparepart**

 Adapun seperti gambar 4.50 sebagai berikut ini :

Gambar 4.50 *Statechart diagram method* hapus sparepart

* + 1. ***Statechart diagram method* cari sparepart**

Adapun seperti gambar 4.51 sebagai berikut ini :



Gambar 4.51 *Statechart diagram method* cari sparepart

1. ***Statechart diagram class* tarif service**

*Statechart diagram class* tarif service menjelaskan aktifitas yang dilakukan pemilik toko dalam mengelola tarif service.

* + - 1. ***Statechart diagram method* tambah tarif service**

Adapun seperti gambar 4.52 sebagai berikut ini:



Gambar 4.52 *Statechart diagram method* tambah tarif service

* + - 1. ***Statechart diagram method* edit tarif service**

Adapun seperti gambar 4.53 sebagai berikut ini:

Gambar 4.53 *Statechart diagram method* edit tarif service

* + - 1. ***Statechart diagram method* cari tarif service**

Adapun seperti gambar 4.54 sebagai berikut ini:

Gambar 4.54 *Statechart diagram method* cari tarif service

* + - 1. ***Statechart diagram method* tampil tarif service**

Adapun seperti gambar 4.55 sebagai berikut ini:



Gambar 4.55 *Statechart diagram method* tampil tarif service

1. ***Statechart diagram class* rinci tarif**

*Statechart diagram class* rinci tarif menjelaskan aktifitas yang dilakukan administrasi dalam mengelola rinci tarif service.

* + - * 1. ***Statechart diagram method* tambah rinci tarif**

Adapun seperti gambar 4.56 sebagai berikut ini :

Gambar 4.56 *Statechart diagram method* tambah rinci tarif

* + - * 1. ***Statechart diagram method* edit rinci tarif**

Adapun seperti gambar 4.57 sebagai berikut ini :

Gambar 4.57 *Statechart diagram method* edit rinci tarif

* + - * 1. ***Statechart diagram method* hapus rinci tarif**

Adapun seperti gambar 4.58 sebagai berikut ini :

Gambar 4.58 *Statechart diagram method* hapus rinci tarif

* + - * 1. ***Statechart diagram method* tampil rinci tarif**

Adapun seperti gambar 4.59 sebagai berikut ini:



Gambar 4.59 *Statechart diagram method* tampil rinci tarif

1. ***Statechart Diagram Class* Pendaftaran**

*Statechart diagram* pendaftaran menjelaskan aktifitas yang dapat dilakukan oleh Petugas terhadap sistem dalam mengelola *form* pendaftaran.

* + - 1. ***Statechart diagram* tambah pendaftaran**

Adapun penggambarannya seperti gambar 4.60 berikut.

Gambar 4.60 Statechart diagram method simpan pendaftaran

1. ***Statechart diagram* cari pendaftaran**

Adapun penggambarannya seperti gambar 4.61 berikut.



Gambar 4.61 Statechart diagram method cari pendaftaran

1. ***Statechart diagram method edit* pendaftaran**

Adapun penggambarannya seperti gambar 4.62 berikut.



Gambar 4.62 Statechart diagram method edit pendaftaran

1. **Statechart diagram method hapus pendaftaran**

Adapun penggambarannya seperti gambar 4.63 berikut.

Gambar 4.63 Statechart diagram method hapus pendaftaran

1. ***Statechart diagram class* nota transaksi**

*Statechart diagram class* nota transaksi menjelaskan aktifitas yang dilakukan administrasi dalam mengelola nota transaksi.

* 1. ***Statechart diagram method* tambah nota transaksi**

Adapun seperti gambar 4.64 sebagai berikut ini :



Gambar 4.64 *Statechart diagram method* tambah nota transaksi

* 1. ***Statechart diagram method* batal nota transaksi**

Adapun seperti gambar 4.65 sebagai berikut ini :



Gambar 4.65 *Statechart diagram method* batal nota transaksi

* 1. ***Statechart diagram method* cari nota transaksi**

Adapun seperti gambar 4.66 sebagai berikut ini :



Gambar 4.66 *Statechart diagram method* cari nota transaksi

1. ***Statechart diagram class* nota service**

*Statechart diagram class* nota service menjelaskan aktifitas yang dilakukan administrasi dalam mengelola nota transaksi bagian service.

* + 1. ***Statechart diagram method* tambah nota service**

Adapun seperti gambar 4.67 sebagai berikut ini :



Gambar 4.67 *Statechart diagram method* tambah nota service

* + 1. ***Statechart diagram method* edit nota service**

Adapun seperti gambar 4.68 sebagai berikut ini :



Gambar 4.68 *Statechart diagram method* edit nota service

* + 1. ***Statechart diagram method* hapus nota service**

Adapun seperti gambar 4.69 sebagai berikut ini :



Gambar 4.69 *Statechart diagram method* hapus nota service

* 1. ***Statechart diagram method* tampil nota service**

Adapun seperti gambar 4.70 sebagai berikut ini :



Gambar 4.70 *Statechart diagram method* tampil nota service

1. ***Statechart diagram class* histori perbaikan**

*Statechart diagram class* histori perbaikan menjelaskan aktifitas yang dilakukan teknisi dalam mengelola histori perbaikan.

* + 1. ***Statechart diagram method* tambah histori perbaikan**

Adapun seperti gambar 4.71 sebagai berikut ini :



Gambar 4.71 *Statechart diagram method* tambah histori perbaikan

* + 1. ***Statechart diagram method* edit histori perbaikan**

Adapun seperti gambar 4.72 sebagai berikut ini :



Gambar 4.72 *Statechart diagram method* edit histori perbaikan

* + 1. ***Statechart diagram method* hapus histori perbaikan**

Adapun seperti gambar 4.73 sebagai berikut ini :



Gambar 4.73 *Statechart diagram method* hapus histori perbaikan

* + 1. ***Statechart diagram method* tampil histori perbaikan**

Adapun seperti gambar 4.74 sebagai berikut ini :



Gambar 4.74 *Statechart diagram method* tampil histori perbaikan

1. ***Statechart Diagram Class* Laporan**

*Statechart diagram* Laporan menjelaskan aktifitas yang dapat dilakukan oleh petugas terhadap sistem dalam mencetak laporan.

***Statechart diagram* cetak laporan**

Adapun penggambarannya seperti gambar 4.75 berikut.



Gambar 4.75 Statechart diagram method cetak laporan

**4.6 *Entity Relationship Diagram***

*Entity Relationship Diagram* (ERD) merupakan diagram yang digunakan untuk merancang suatu basis data, untuk memperlihatkan hubungan atau relasi antar entitas atau objek yang terlihat beserta atributnya. Objek utama dari pembuatan diagram ERD adalah menunjukan objek-objek (himpunan entitas) apa saja yang ingin dilibatkan dalam sebuah basis data dan bagaimana hubungan yang terjadi antara objek-objek tersebut. Berikut langkah-langkah untuk menghasilkan ERD.

* + 1. Menentukan Entitas

Adapun entitas yang terbentuk pada gambar 4.98 sebagai berikut :



Gambar 4.98 Menentukan Entitas

* + 1. Menentukan atribut *key*

Setelah menentukan entitas selanjutnya yaitu menentukan *atribut key* disetiap entitasnya, adapun seperti gambar 4.99 sebagai berikut :



Gambar 4.99 Menentukan Atribut *Key*

* + 1. Menentukan kardinalitas dan melengkapi *atribut deskriptif* beserta kunci tamu *(Foreign key)*
       - 1. Atribut *deskriptif* beserta *foreign key* dan kardinalitas relasi *one to many* (1:N) yaitu Memesan beberapa barang kepada satu pemasok , adapun seperti gambar 4.100 sebagai berikut :