

## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

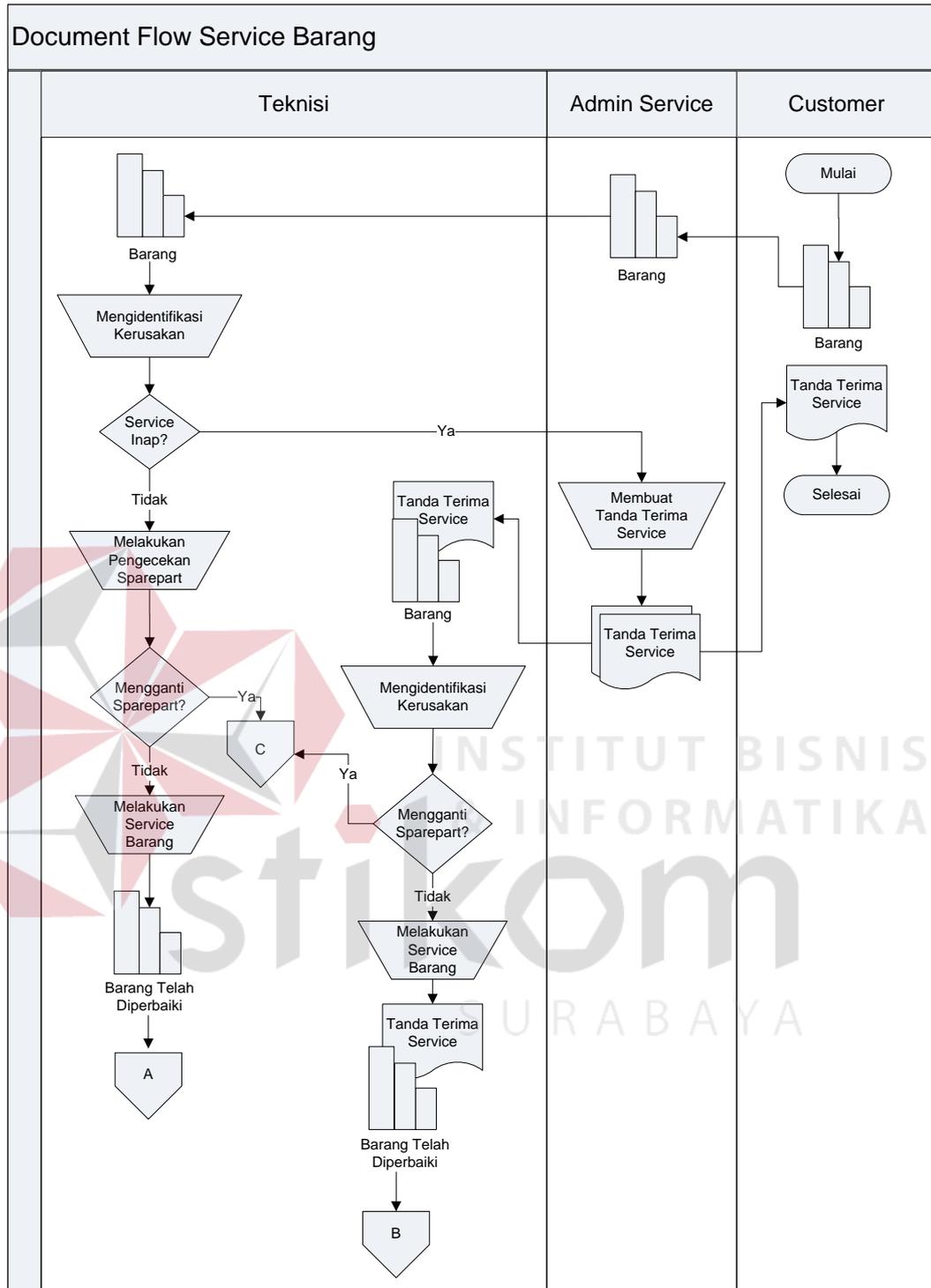
#### 3.1 Analisis Sistem

Tahap analisis merupakan tahap mendasar yang harus dilakukan pertama kali sebelum melanjutkan pada tahap perancangan sistem. Kesalahan dalam tahap ini harus dihindari karena dapat menyebabkan kesalahan yang berhubungan pada tahap berikutnya.

Kegiatan dalam menganalisa sistem yaitu mengidentifikasi dan mengevaluasi permasalahan serta membuat model, spesifikasi sistem dan solusi dalam membuat rancang bangun aplikasi.

Untuk menyempurnakan informasi dari tahap perencanaan sistem yang lebih terperinci tentang kondisi instansi saat ini, maka metode yang digunakan untuk mengumpulkan data adalah pengamatan (observasi), wawancara (*interview*), studi pustaka dan identifikasi masalah.

Berikut ini adalah rancangan dokumen *flow service* barang yang menjelaskan tentang alur proses transaksi oleh *admin service*. Dapat dilihat pada gambar 3.1.



Gambar 3.1 *Document Flow* Proses Transaksi Service Barang

Analisis masalah yang digambarkan dalam *document flow service barang* pada PT. Indo Bismar Komputer dimulai dari *customer* melakukan *service barang*

di ruang *service* PT. Indo Bismar Komputer. *Customer* menaruh barang yang perlu diperbaiki pada admin *service*. Kemudian diserahkan kepada teknisi oleh admin *service* dan teknisi bertanya pada *customer* tentang apa yang dikeluhkan agar dapat ditentukan seberapa lama pengerjaannya.

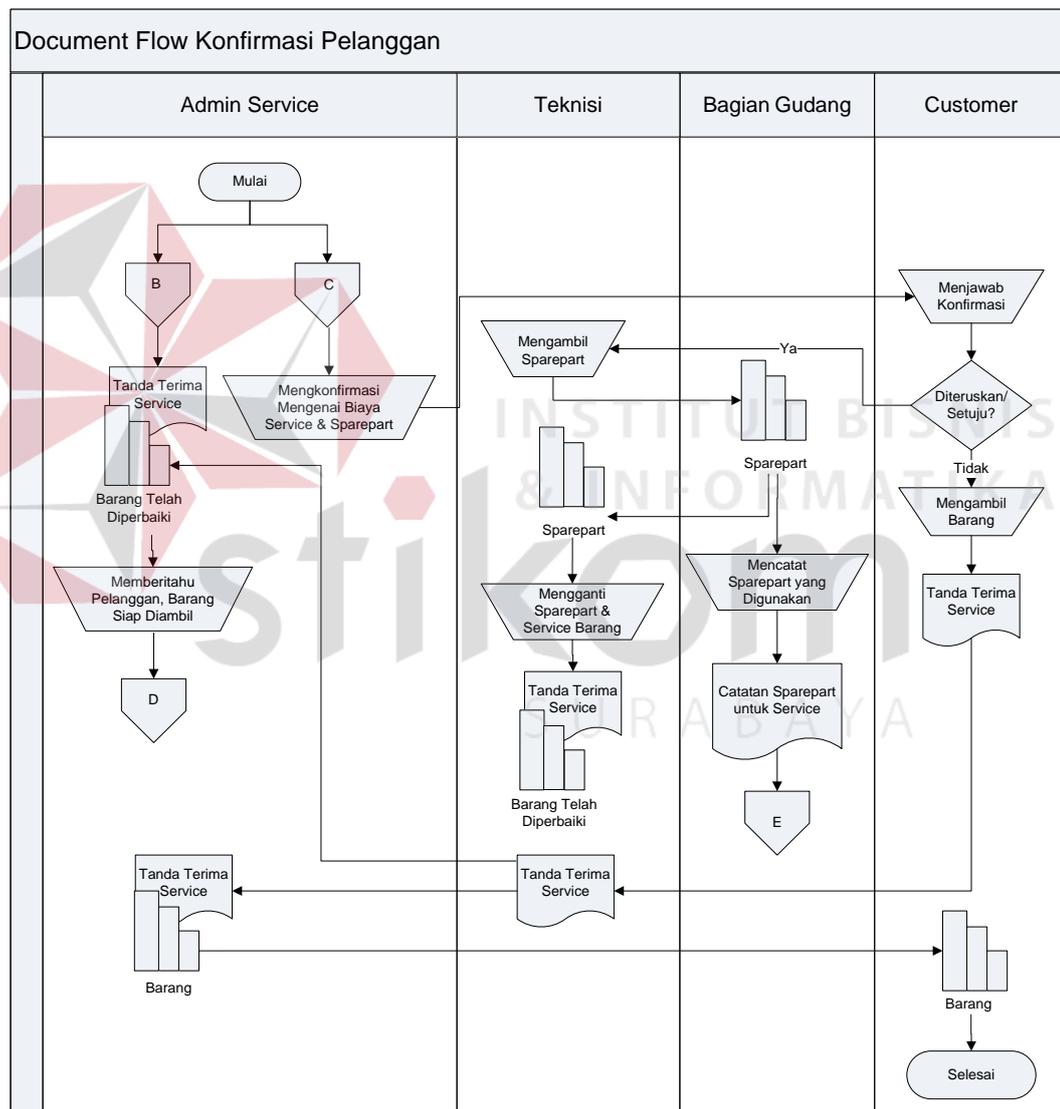
Jika pengerjaan *service* butuh waktu yang lama, teknisi menyarankan agar barang yang akan di-*service* ditinggal saja. Jika *service* membutuhkan waktu sekitar kurang dari satu jam, teknisi menyarankan kepada *customer* untuk menunggu.

Proses *service* barang inap: Admin *service* membuat tanda terima *service* dua rangkap (berwarna putih dan merah). Tanda terima *service* warna putih diberikan pada *customer* dan tanda terima *service* warna merah ditempel menggunakan isolasi pada barang *service* milik *customer*. Kemudian teknisi mencari kerusakan. Jika kerusakan barang berasal dari *sparepart* yang rusak parah dan *sparepart* perlu diganti, maka admin *service* mengkonfirmasi *customer* mengenai *sparepart* yang perlu diganti beserta biaya yang akan dikenakan. Namun jika *sparepart* hanya perlu diperbaiki saja, maka teknisi langsung melakukan *service* barang hingga barang selesai diperbaiki.

Proses *service* barang langsung: Jika *service* dapat dikerjakan langsung (tidak sampai satu jam), maka *customer* biasanya menunggu dan admin *service* tidak perlu membuat tanda terima *service*. Kemudian teknisi mencari kerusakan. Jika kerusakan barang berasal dari *sparepart* yang rusak parah dan *sparepart* perlu diganti, maka admin *service* bertanya kepada *customer* apakah *customer* bersedia mengganti *sparepart* atau tidak. Namun jika *sparepart* hanya perlu

diperbaiki saja, maka teknisi langsung melakukan *service* barang hingga barang selesai diperbaiki.

Berikut ini adalah rancangan dokumen *flow* konfirmasi *customer* yang menjelaskan tentang alur proses transaksi oleh *admin service*. Dapat dilihat pada gambar 3.2.



Gambar 3.2 Document Flow Proses Transaksi Konfirmasi Customer

Analisis masalah yang digambarkan dalam *document flow* konfirmasi *customer* pada PT. Indo Bismar Komputer dimulai dari dua jenis transaksi yaitu barang *service* dengan mengganti *sparepart* (C) dan barang *service* yang hanya memperbaiki *sparepart* saja (B).

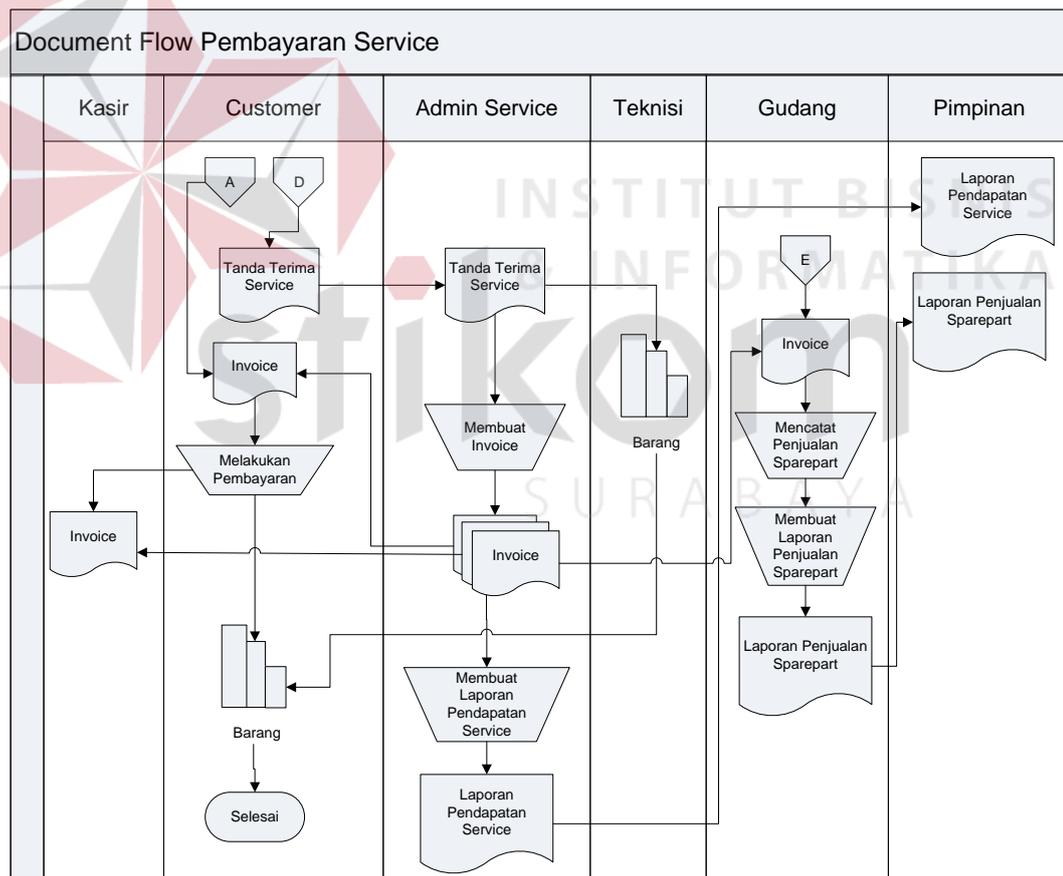
Proses bisnis transaksi perbaikan *sparepart* (B): Untuk transaksi yang hanya memperbaiki *sparepart* saja maka dimulai dari bagian admin *service* yang akan diberi tanda terima *service* oleh teknisi. Lalu admin *service* mengkonfirmasi *customer* melalui pesan atau telepon bahwa barang sudah diperbaiki dan bisa diambil.

Proses bisnis transaksi penggantian *sparepart* (C): Untuk transaksi yang perlu mengganti *sparepart* maka alur *document flow* dimulai dari bagian admin *service* yang hendak mengkonfirmasi *customer* melalui pesan atau telepon untuk memberitahu biaya *service* yang akan dikenakan dan memberitahu tentang penggantian *sparepart* lalu menanyakan kepada *customer* bahwa *service* akan dilanjutkan atau tidak. Jika *customer* menjawab dilanjutkan saja, maka teknisi akan mengambil *sparepart* ke bagian gudang. Ketika *sparepart* telah diambil oleh teknisi, bagian gudang mencatat *sparepart* yang digunakan teknisi ke dalam sebuah buku tulis.

Setelah teknisi mengambil *sparepart* di gudang, kemudian teknisi mengganti *sparepart* dan mulai men-*service* barang. Jika barang telah selesai di-*service*, teknisi menyerahkan barang pada admin *service* dan admin *service* mengirim pesan atau menelepon untuk memberitahu bahwa barang sudah diperbaiki dan bisa diambil.

Kemudian, *admin service* mencatat jawaban dari *customer* mengenai tanggal pengambilan barang *service* oleh *customer*. Dan sebaliknya jika *customer* membatalkan *service*, maka *customer* bisa langsung datang dan mengambil barang ke ruang *service* PT. Indo Bismar Komputer dengan memberikan tanda terima *service* pada *admin service* untuk bukti pengambilan barang.

Berikut ini adalah rancangan dokumen *flow* pembayaran *service* yang menjelaskan tentang alur proses transaksi oleh *admin service*. Dapat dilihat pada gambar 3.3.



Gambar 3.3 Document Flow Proses Transaksi Pembayaran Service

Analisis masalah yang digambarkan dalam *document flow* pembayaran *service* pada PT. Indo Bismar Komputer dimulai dari dua jenis transaksi yaitu barang *service* yang ditinggal atau *service* inap atau pengerjaan *service* yang membutuhkan waktu lama dan barang *service* yang langsung ditangani atau tidak menginap atau pengerjaan *service* cepat.

Untuk proses barang yang akan di-*service* tersebut ditinggal atau *service* inap atau pengerjaan *service* yang butuh waktu yang lama, maka *customer* harus membawa tanda terima *service* sebagai tanda bukti kepemilikan barang dan diberikan pada admin *service*. Admin *service* ke bagian teknisi untuk mengambil barang dan *running* barang *service* agar *customer* mengecek kembali barangnya. Dalam waktu yang sama, admin *service* mencatat dan membuat *invoice* di Microsoft Excel untuk dicetak menjadi rangkap tiga, yaitu: warna putih, merah dan kuning.

*Invoice* berwarna putih diserahkan kepada *customer* dan *invoice* berwarna merah diserahkan pada bagian kasir. Lalu *customer* membayar biaya *service* pada kasir, kasir mencocokkan *invoice* putih milik *customer* dengan *invoice* merah yang sudah dicetak dan diserahkan oleh admin *service*.

Untuk *invoice* berwarna kuning akan diarsipkan dan setiap bulannya diserahkan kepada bagian gudang agar dibuatkan laporan penjualan *sparepart*. Setiap akhir bulan admin *service* membuat dan mencetak laporan pendapatan *service* yang datanya berasal dari arsip nota *invoice* warna kuning tersebut untuk diserahkan kepada pimpinan.

Untuk proses barang *service* yang langsung ditangani atau tidak menginap atau pengerjaan *service* cepat maka *customer* langsung menerima *invoice* dan

langsung melakukan pembayaran kepada kasir dan kasir mencocokkan dengan *invoice* merah yang ada di dalam buku transaksinya. Selanjutnya *customer* mengambil barang dan *running* barang *service* untuk dicek kembali oleh *customer*.

### 3.1.1 Wawancara

Wawancara terhadap bagian admin *service*, bagian teknisi, bagian gudang, kasir dan pimpinan perusahaan. Tujuan dari kegiatan ini adalah untuk mendapatkan informasi penting yang sangat dibutuhkan dalam mencari dan mengidentifikasi proses bisnis pada perusahaan, deskripsi pekerjaan pada beberapa bagian perusahaan, hingga masalah operasional yang dihadapi oleh perusahaan agar dapat mengidentifikasi masalah yang sedang terjadi pada PT. Indo Bismar Komputer.

Hasil dari wawancara atau *interview* atau tanya jawab yang dilakukan oleh bagian admin *service* dan bagian teknisi maka didapatkan informasi dan data sebagai berikut:

Ketika admin *service* menerima *service* inap dari *customer* maka admin *service* membuat nota tanda terima *service* untuk *customer*. Tanda terima *service* yang dibuat oleh admin *service* dicatat dalam sebuah nota manual rangkap dua dan untuk pembukuannya dicatat dalam sebuah buku tulis biasa.

Pada saat kerusakan barang telah berhasil ditemukan apakah perlu mengganti *sparepart* atau tidak, maka admin *service* menghubungi *customer* melalui telepon untuk menanyakan *service* dilanjutkan atau tidak. *Customer* menjawab konfirmasi dan menyetujui untuk penggantian *sparepart* dan admin

*service* segera memberitahu teknisi untuk mengganti *sparepart*. Teknisi langsung mengambil *sparepart* ke bagian gudang. Bagian gudang mencatat nama barang dan jumlah *sparepart* yang diambil oleh teknisi ke dalam sebuah buku tulis.

Ketika barang telah selesai di-*service* maka admin *service* menghubungi *customer* untuk memberitahu dan menanyakan kapan sekiranya barang akan diambil. Lalu *customer* datang dan mengambil barang, admin *service* membuat nota *invoice* untuk *customer*, kasir dan gudang. Sedangkan admin *service* sendiri juga mencatat ulang ke dalam buku tulis tentang pendapatan *service*.

Untuk laporan pendapatan *service* setiap bulannya, admin *service* membuat laporan pendapatan *service* dengan mengetikkan lagi data yang dicatat dari buku tulis dan nota *invoice* warna kuning ke dalam Microsoft Excel dan mencetaknya untuk diserahkan kepada pimpinan.

Setiap akhir bulan, bagian gudang membuat laporan penjualan *sparepart* dengan mengetikkan lagi data yang dicatat dari buku tulis ke dalam Microsoft Excel dan mencetaknya untuk diserahkan kepada pimpinan. Namun laporan yang diserahkan kepada pimpinan adalah laporan penjualan seluruhnya, bukan laporan penjualan *service* saja. Dalam laporan penjualan *sparepart* tidak dibedakan antara laporan penjualan *sparepart service* dengan laporan penjualan oleh PT. Indo Bismar Komputer sendiri.

### **3.1.2 Identifikasi Masalah**

PT Indo Bismar Komputer merupakan perusahaan yang beroperasi dibidang penjualan, *service* dan distributor komputer. Pada sektor *service*, terdapat bagian administrasi *service* yang mengelola proses *service*. Bagian

administrasi *service* ini bertanggung jawab mulai dari mencatat, membuat dan mencetak nota tanda terima *service* serta *invoice*, merekap nota tanda terima *service* dengan nota *invoice*, mengkonfirmasi *customer*, membantu teknisi menginstall *software*, install ulang OS serta *running* aplikasi, setiap akhir bulan admin *service* membuat laporan pendapatan *service* untuk diserahkan kepada pimpinan perusahaan.

Alur proses bisnis yang menjadi pokok permasalahan yaitu mekanisme pencatatan data tanda terima *service* dicatat secara langsung dalam sebuah buku tulis sehingga menyebabkan redundansi data. Pencatatan biaya jasa perbaikan atau *service* dan penjualan *sparepart* yang dikelola admin *service* seringkali mengalami kesusahan dalam perhitungan karena belum adanya pengelolaan perhiungan *service* yang akurat. Admin *service* memberi harga jasa *service* berdasarkan taksiran (ingatan) dari teknisi sehingga biaya jasa *service* sering berubah sesuai permintaan teknisi dan tidak akurat.

Seringkali barang-barang yang sudah diperbaiki belum dikonfirmasi admin *service* yang berakibat terjadinya penumpukan barang karena selama ini admin *service* mengkonfirmasi *customer* hanya mengandalkan ingatan saja apakah *customer* telah dikonfirmasi atau belum dan *customer* mana saja yang belum dikonfirmasi agar dilakukan konfirmasi ulang. Dengan kata lain tidak ada pencatatan status konfirmasi yang jelas sehingga mengakibatkan penumpukan barang *service* karena keterbatasan luas ruang penyimpanan untuk barang *service*.

Apabila ada beberapa barang telah di-*service* namun admin *service* lupa mengkonfirmasi. Barang-barang yang belum dikonfirmasi tentunya suatu saat akan ditanyakan langsung oleh *customer*. Hal ini menjadi perhatian paling utama

jika *customer* mengeluh pelayanan yang lama sebab admin *service* lupa mengkonfirmasi sehingga berkurangnya kepercayaan dan kepuasan *customer*. Karena berkurangnya kepuasan *customer*, biasanya *customer* mencari tempat *service* lain yang pelayanannya lebih cepat sehingga PT. Indo Bismar Komputer kehilangan *customer* dan pemasukan *service* menjadi kurang maksimal.

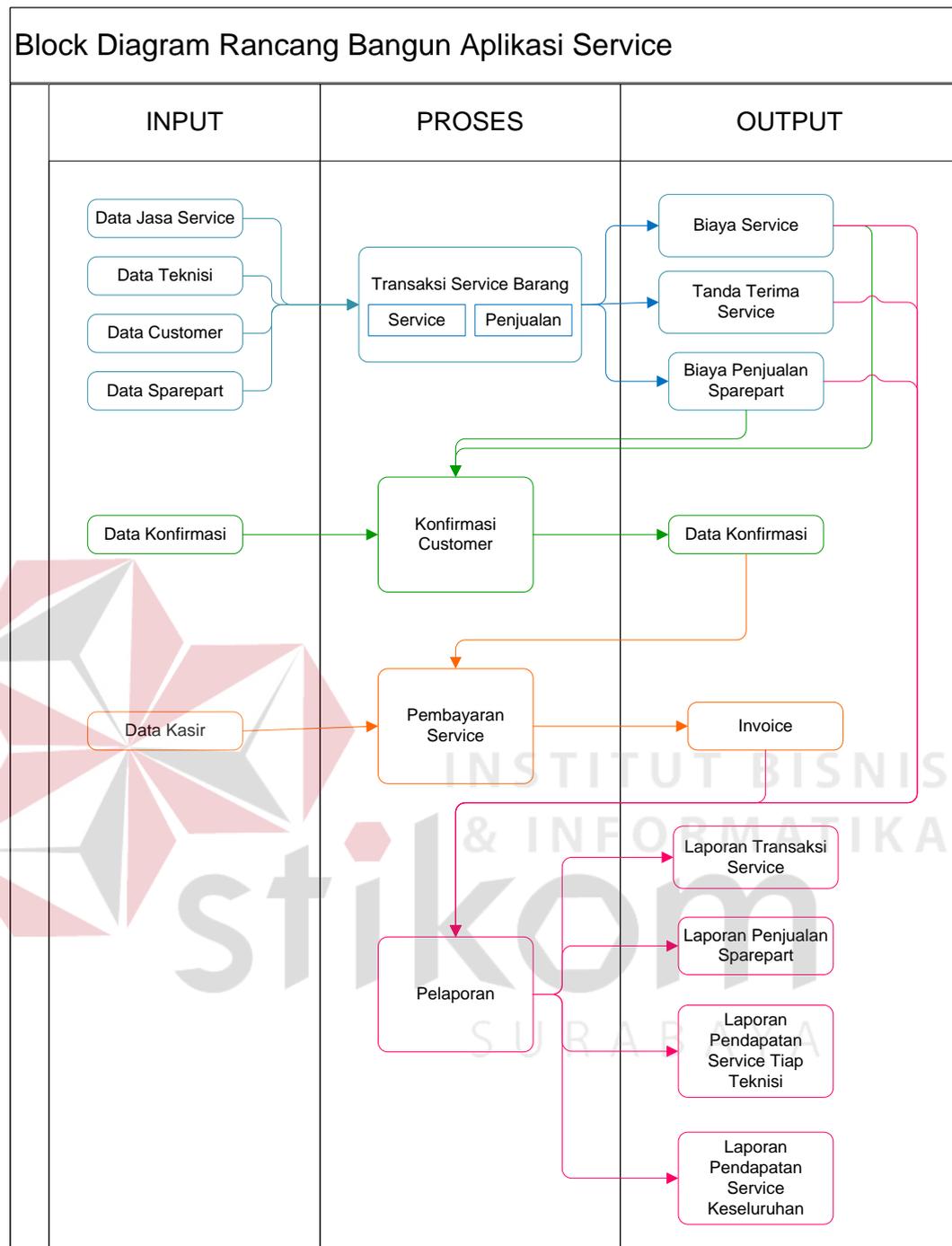
### 3.2 Perancangan Sistem

Tahap ini sangat berguna agar aplikasi yang dihasilkan dapat berfungsi selaras dengan kebutuhan yang diharapkan, yaitu membantu pimpinan dalam menganalisa data dan dalam pengambilan keputusan yang lebih informatif. Rancang bangun aplikasi administrasi *service* ini dibangun dalam bentuk *desktop* dan menggunakan *visual basic* sebagai bahasa pemrogramannya.

Pembuatan rancang bangun aplikasi *service* pada PT. Indo Bismar Komputer dijabarkan secara rinci dan bertahap. Tahapan-tahapan dalam proses perancangan sistem yang dipakai dalam rancang bangun aplikasi *service* ini adalah sebagai berikut:

#### 3.2.1 IPO Diagram

Dalam tahap perancangan sistem ini, dibuatlah *IPO diagram* sebagai acuan dalam menyelesaikan rancang bangun aplikasi *service* ini. *IPO Diagram* sangat dibutuhkan dalam mengetahui masukan (*input*) dan keluaran (*output*), yang digambarkan pada gambar 3.4 dibawah ini:



Gambar 3.4 *Input-Proses-Output* Diagram Rancang Bangun Aplikasi *Service*

Berdasarkan gambar 3.4 *Input-Proses-Output* Diagram yang akan digunakan pada PT. Indo Bismar Komputer diatas, maka dapat dilihat pada penjelasan berikut:

a. *Input*

1. Data Teknisi yang digunakan berasal dari proses pendaftaran teknisi baru yang sebelumnya telah diterima oleh pihak yang berwenang dalam penerimaan pegawai baru pada PT. Indo Bismar Komputer. Data teknisi mencakup identitas atau profil lengkap teknisi. Di dalam data teknisi juga disematkan *username* dan *password* untuk dapat login pada rancang bangun aplikasi *service*.
2. Data *Customer* yang digunakan berasal dari proses *service* barang. Ketika *customer* datang ke ruang *service* PT. Indo Bismar Komputer untuk memperbaiki barangnya maka data *customer* sangat dibutuhkan sebagai tanda bukti bahwa barang tersebut milik *customer*. Data *customer* mencakup nama, nomor yang bisa dihubungi dan alamat lengkap rumah saja.
3. Data *Jasa Service* yang digunakan berasal dari proses pencatatan jenis-jenis jasa *service* serta biaya yang dikenakan bervariasi tergantung tingkat kesulitan dan waktu pengerjaan *service* tersebut. Hal-hal yang bisa masuk ke dalam kategori jasa *service* adalah penjualan jasa *service* teknisi saja. Data jasa *service* mencakup kerusakan dan biaya *service* saja.
4. Data *Sparepart* yang digunakan berasal dari proses pembelian *sparepart* yang dibutuhkan dalam melakukan *service* barang. Data *sparepart* menyimpan harga beli dan harga jual *sparepart* yang dibutuhkan ketika teknisi melakukan penggantian *sparepart* pada proses *service* barang. Data *sparepart* mencakup nama *sparepart*, harga beli *sparepart* dan harga jual *sparepart*.

5. Data Konfirmasi yang digunakan berasal dari proses konfirmasi *customer* ketika *service* perlu mengganti sparepart atau ketika barang sudah selesai diperbaiki dan siap untuk diambil. Data konfirmasi menyimpan jenis kerusakan pada barang *service customer* dan *sparepart* yang telah diganti sehingga menghasilkan perhitungan biaya *service* yang berguna dalam pengambilan keputusan bagi *customer*. Di dalam data konfirmasi terdapat tanggal konfirmasi dan total bayar.

6. Data Kasir yang digunakan berasal dari proses pendaftaran kasir baru yang sebelumnya telah diterima oleh pihak yang berwenang dalam penerimaan pegawai baru pada PT. Indo Bismar Komputer. Data kasir mencakup identitas atau profil lengkap kasir. Di dalam data kasir juga disematkan *username* dan *password* untuk dapat login pada rancang bangun aplikasi *service*.

b. *Process*

Berdasarkan data *input* diatas maka tahapan selanjutnya yang akan dibuat adalah proses. Berikut adalah penjelasan mengenai proses-proses yang terkait dalam perancangan sistem ini:

1. Transaksi *Service* Barang

Proses transaksi *service* barang ini digunakan pada saat *customer* melakukan *service* barang. Di dalam transaksi *service* barang ini ada proses dimana admin *service* mencatat apa yang dikeluhkan *customer* mengenai barang *service* miliknya agar teknisi dapat melakukan pengecekan jenis kerusakan barang dan mencatat data pemilik atau *customer*, tanggal *service* serta nama teknisi yang menangani *service* barang tersebut. Dalam

transaksi ini tidak ada perhitungan atau rumus tertentu yang digunakan, hanya menyimpan data saja.

## 2. Konfirmasi *Customer*

Proses konfirmasi *customer* ini digunakan pada saat proses *service* barang telah selesai diperbaiki atau juga dapat digunakan untuk mengkonfirmasi *customer* ketika diperlukan penggantian *sparepart* penting dengan harga yang cukup mahal. Jika kerusakan barang berada dibagian *sparepart* dan kerusakannya sangat parah sehingga perlu mengganti *sparepart* maka proses ini berlaku. Proses konfirmasi *customer* ini untuk mencatat *customer* yang belum atau telah dikonfirmasi serta mengkalkulasi total penjualannya.

## 3. Pembayaran *Service*

Proses pembayaran *service* digunakan ketika *customer* melunasi pembayaran terhadap barang yang telah selesai *service*. Pembayaran *service* mencakup total biaya dari jasa *service* yang dijumlahkan dengan total biaya dari *sparepart* yang dibeli.

## 4. Pelaporan

Proses pelaporan yang berlangsung di PT. Indo Bismar Komputer berasal dari *output* pembayaran *service* yang diakhiri dengan pembuatan laporan untuk pimpinan.

c. *Output*

1. *Biaya Service*

*Biaya service* ini digunakan untuk menampilkan transaksi *service* yang menggunakan jasa *service* teknisi saja. *Biaya service* ini nantinya akan menjadi hasil dari *input* dalam pembuatan *invoice*.

2. *Tanda Terima Service*

*Tanda terima service* merupakan nota yang berguna sebagai bukti atau tanda terima kepada *customer* bahwa teknisi telah menerima barang yang perlu di-*service* dari *customer*. Nota tanda terima *service* ini berasal dari data teknisi, data *customer*, keterangan barang *service* milik *customer* dan keluhan yang terjadi pada barang *service*. Nota ini harus dicetak untuk diberikan kepada *customer* sebagai jaminan atau tanda bukti bahwa *customer* adalah pemilik barang *service* tersebut.

3. *Data Konfirmasi*

Data konfirmasi berisi tentang kumpulan data *customer* yang melakukan *service* barang dan proses *service* tersebut statusnya masih belum selesai atau sudah selesai namun belum dikonfirmasi.

4. *Biaya Penjualan Sparepart*

Data ini berasal dari data *sparepart* yang telah diproses dan diambil dari proses penjualan *sparepart*. *Biaya penjualan sparepart* merupakan kumpulan beberapa data *sparepart* yang telah dipakai dalam proses *service* barang sehingga data ini akan menampilkan *sparepart* apa saja yang telah dijual atau telah dipakai dalam mengganti atau *service* barang *customer*.

#### 5. *Invoice*

Nota *Invoice* yang dihasilkan ini diambil dari data biaya *service* pertransaksi, data penjualan *sparepart* pertransaksi dan data kasir. Nota *Invoice* tersebut dicetak dan diberikan pada *customer* sebagai tanda bukti jika telah melunasi atau telah membayar *invoice* dan mengambil barang yang telah selesai di-*service*.

#### 6. Laporan Transaksi *Service*

Laporan transaksi *service* merupakan hasil perhitungan dari transaksi *service* barang dan penjualan *sparepart* yang mana dalam laporan ini berisi data transaksi *service* selama satu bulan.

#### 7. Laporan Penjualan *Sparepart*

Laporan penjualan *sparepart* adalah hasil kalkulasi dari transaksi *service* barang dengan penjualan *sparepart*. Dalam laporan penjualan *sparepart* ini berisi data penjualan *sparepart* yang sangat dibutuhkan dalam proses *service* barang dalam kurun waktu satu bulan.

#### 8. Laporan Pendapatan *Service* Tiap Teknisi

Laporan pendapatan *service* tiap teknisi ini berasal dari *invoice*. Laporan ini sangat berguna bagi pimpinan untuk mengetahui dan mengevaluasi kinerja teknisi dalam kurun waktu satu bulan. Di dalam laporan ini pimpinan dapat melakukan penilaian terhadap kinerja teknisi mengenai seberapa banyak masing-masing teknisi menangani *service* barang milik *customer*. Pimpinan juga dapat mengetahui, mengevaluasi dan menilai keunggulan teknisi dalam memperbaiki jenis kerusakan beserta total pendapatan *service* yang diperoleh teknisi dalam kurun waktu satu bulan.

## 9. Laporan Pendapatan *Service* Keseluruhan

Laporan pendapatan *service* keseluruhan ini mencakup semua pendapatan kotor dan pendapatan bersih yang dihasilkan oleh seluruh teknisi. Laporan ini berasal dari *invoice* yang telah dilunasi atau dibayar oleh *customer*. Laporan ini sangat berguna dalam mengetahui arus aliran kas yang terjadi selama satu bulan, total pendapatan *service* dalam satu bulan, total penjualan *sparepart* dalam satu bulan dan total jasa *service* dalam satu bulan. Laporan ini digunakan oleh pimpinan untuk menganalisa naik turunnya pendapatan *service* agar dilakukan sebuah tindakan yang sesuai.

### 3.2.2 Kebutuhan Sistem

Dalam suatu rancang bangun aplikasi *service* pada PT. Indo Bismar Komputer, informasi yang sangat dibutuhkan ialah *output* yang dihasilkan dari suatu aplikasi. PT. Indo Bismar Komputer tentunya sangat membutuhkan laporan pendapatan *service* dari tiap teknisi maupun laporan pendapatan *service* secara keseluruhan yang diperoleh perusahaan dalam periode tertentu, laporan transaksi *service* berperiode serta laporan penjualan *sparepart* dalam kurun waktu tertentu.

Berdasarkan laporan tersebut maka informasi yang dibutuhkan dalam pembuatan rancang bangun aplikasi *service* ini adalah bagaimana pegawai memasukkan data *sparepart* dalam perusahaan, bagaimana pegawai menentukan jenis jasa *service* apa saja yang dikuasai untuk dimasukkan dalam bentuk data ke dalam aplikasi, bagaimana pegawai membuat tanda terima *service*, mengkonfirmasi *customer* hingga pada proses pembayaran *invoice*. Informasi-

informasi ini tentunya sangat berhubungan erat dengan transaksi *service* yang ada pada PT. Indo Bismar Komputer. Berikut ini merupakan pengguna yang memiliki wewenang dalam melakukan akses pada rancang bangun aplikasi *service*:

1. Direktur: dapat mengakses seluruh *form* laporan.
2. Bagian Keuangan: dapat mengakses *form* master *sparepart*, master jasa *sparepart* dan seluruh *form* pada laporan.
3. Administrasi: Adalah sebutan bagi bagian admin *service* yang memiliki wewenang dalam mengakses seluruh *form* master, transaksi dan laporan.

Admin *service* bertugas membuat jenis-jenis jasa *service*, tanda terima *service* jika ada barang *service* masuk dan mengkonfirmasi *customer*.

4. Bagian Pembelian: dapat mengakses master *sparepart* saja.
5. Bagian Gudang: dapat mengakses master *sparepart* saja.
6. Kasir: dapat mengakses *form* transaksi *invoice* karena berhubungan dengan pembayaran.
7. Teknisi: dapat mengakses *form* laporan pendapatan *service*.

Dari penjelasan diatas maka dapat ditarik kesimpulan bahwa terdapat beberapa hal yang menjadi keebutuhan dalam sistem yang diusulkan oleh peneliti, yaitu:

1. Kebutuhan data jenis-jenis jasa *service* yang akurat dan konsisten

Masalah : Sulitnya menetapkan biaya jasa *service* secara konsisten

Usulan : Disediakan daftar biaya jasa *service* berdasarkan jenis jasa *service*

2. Kebutuhan data jenis-jenis barang *sparepart* yang akurat dan konsisten

Masalah : Sulitnya menetapkan harga *sparepart* secara konsisten

Usulan : Disediakan daftar harga *sparepart*

3. Kebutuhan proses konfirmasi *customer* yang efektif dan efisien

Masalah : Sulitnya melakukan pencatatan secara manual sehingga sering tidak dapat membedakan status konfirmasi *customer* apakah telah berhasil mengkonfirmasi atau belum menerima konfirmasi atau batal *service* sehingga ada *customer* yang dilewati dan menimbulkan pencitraan kualitas pelayanan yang lama.

Usulan : Disediakan halaman konfirmasi otomatis yang memudahkan pegawai dalam melakukan konfirmasi kepada *customer*. Jika *customer* tidak bisa dikonfirmasi maka status pada *form* diisi belum dikonfirmasi dan jika *customer* berhasil dikonfirmasi maka status pada *form* diganti menjadi berhasil konfirmasi sehingga data transaksi *service* tersebut berpindah ke *form invoice*.

4. Kebutuhan pencetakan laporan pendapatan *service* teknisi

Masalah : Kesalahan ditimpakan pada semua teknisi ketika ada *customer* yang mengeluh *service* lama melebihi batas waktu yang telah ditetapkan teknisi sehingga membuat *customer* enggan menaruh kepercayaan pada PT. Indo Bismar Komputer.

Usulan : Dicitakkan laporan pendapatan *service* teknisi setiap bulan agar dapat mengetahui teknisi yang perlu diberi pengarahan atau juga sebagai acuan untuk mengetahui rata-rata pendapatan *service* setiap bulan agar dapat membuat teknisi selalu termotivasi dan semangat.

5. Kebutuhan pencetakan laporan pendapatan *service* perusahaan/keseluruhan

Masalah : Membutuhkan waktu yang lama dalam membuat laporan yang akurat karena harus mencocokkan antara nota *invoice* dengan transaksi *service* yang dicatat dalam sebuah buku manual.

Usulan : Dibuatkan laporan pendapatan *service* perusahaan dimana pegawai dapat menghemat waktu dan melaporkan secara akurat setiap bulan kepada pimpinan guna memantau naik turunnya transaksi *service*.

#### 6. Kebutuhan pencetakan laporan transaksi *service*

Masalah : Kesulitan dalam menghitung jumlah transaksi karena pencatatan yang tercampur antara *customer* yang batal *service* dengan *customer* yang melanjutkan *service*.

Usulan : Disediakan laporan transaksi *service* agar dapat mengetahui secara tepat jumlah *transaksi service*.

#### 7. Kebutuhan pencetakan laporan penjualan *sparepart*

Masalah : Sulitnya membuat laporan penjualan *sparepart* secara detail. Perusahaan hanya menghitung total laba yang diperoleh saja tanpa mengetahui *sparepart* apa saja yang terjual karena selama ini nota pembayaran hanya menyebutkan jenis jasa *service* dan biaya keseluruhannya saja dan bagian gudang hanya menyimpan nota pembelian *sparepart* dari *supplier* untuk ditulis dalam sebuah buku.

Usulan: Disediakan laporan penjualan *sparepart* sehingga dapat mengetahui *sparepart* apa saja yang sering digunakan dan harus disediakan lebih banyak serta dapat mengetahui secara rinci dan akurat harga dan laba yang diperoleh dari penjualan setiap masing-masing *sparepart*.

### 3.2.3 System Flow

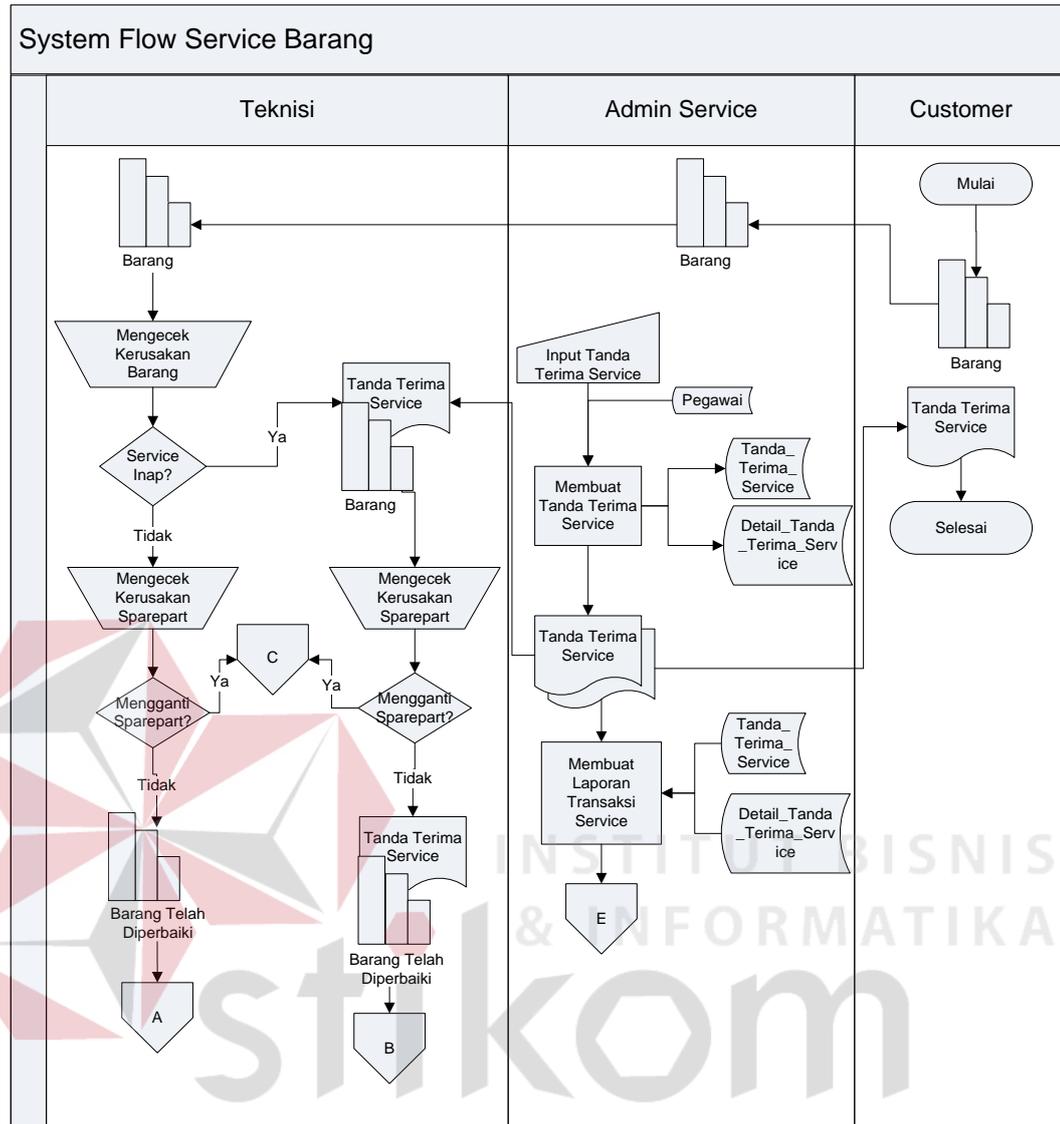
Berdasarkan analisis *IPO* diagram diatas maka dibuatlah suatu pengembangan sistem lebih lanjut untuk mencari solusi dari permasalahan yang terjadi pada PT. Indo Bismar Komputer. Dengan *IPO* diagram maka dapat dihasilkan suatu pengembangan dalam bentuk *system flow*. *System flow* adalah proses bisnis yang akan dijalankan berdasarkan sistem atau solusi yang telah dibuat.

Solusi proses bisnis yang digambarkan dalam *system flow service* barang pada PT. Indo Bismar Komputer dimulai dari *customer* yang melakukan *service* barang pada ruang *service* di PT. Indo Bismar Komputer. *Customer* dilayani oleh admin *service* yang menerima barang *service* dan menyerahkannya kepada bagian teknisi untuk dicek lebih lanjut lagi. Teknisi menerima barang *service* milik *customer* dari admin *service* lalu teknisi mengecek kerusakan barang tersebut berdasarkan keluhan dari *customer*. Jika kerusakannya parah dan membutuhkan waktu yang lama untuk melakukan proses *service* barang, maka admin *service* membuka *form* tanda terima *service* dan mengisi tanda terima *service* serta mencantumkan nama teknisi yang bersangkutan dalam menangani barang *service* milik *customer*. Lalu tanda terima *service* tersebut disimpan dalam *database* tanda terima *service* dan *database* detail tanda terima *service*. Setelah disimpan, admin *service* mencetak tanda terima *service* menjadi dua rangkap untuk diserahkan kepada *customer* sebagai tanda bukti kepemilikan barang dan satunya lagi diserahkan pada teknisi untuk ditempel menggunakan isolasi pada barang *service* milik *customer* untuk menandai bahwa barang tersebut milik *customer*.

Setelah proses tanda terima *service* tersebut selesai, maka selanjutnya teknisi mengecek setiap *sparepart* yang berkaitan dengan kerusakan barang *service* tersebut. Jika ditemukan ada *sparepart* yang rusak maka proses akan berlanjut pada *system flow* konfirmasi *customer*. Jika ditemukan ada *sparepart* yang perlu diperbaiki saja maka teknisi tidak perlu mengganti *sparepart*, namun teknisi hanya perlu memperbaiki *sparepart* saja. Setelah *sparepart* selesai diperbaiki, maka teknisi menaruh barang *service* yang sudah diperbaiki tersebut ke dalam tempat penyimpanan barang *service* agar nantinya dikonfirmasi oleh bagian admin *service*.

Selanjutnya untuk proses yang kerusakannya sepele atau kecil dan membutuhkan waktu setidaknya kurang dari satu jam maka *customer* biasanya menunggu dan tidak perlu dibuatkan tanda terima *service*. Lalu teknisi melakukan pengecekan yang kedua yaitu pengecekan *saprepart* pada komponen yang rusak tersebut. Jika *sparepart* perlu diganti maka prosesnya kembali lagi seperti yang diatas. Namun jika tidak perlu diganti maka teknisi hanya perlu memperbaiki *sparepart* saja kemudian ketika barang telah selesai diperbaiki maka barang *service* tersebut ditaruh dalam lemari khusus barang *service* agar dikonfirmasi lebih lanjut oleh admin *service*.

Berikut ini adalah rancangan *system flow service* barang yang menjelaskan tentang alur proses transaksi oleh *admin service*. Dapat dilihat pada gambar 3.5.



Gambar 3.5 System Flow Proses Transaksi Service Barang

Solusi proses bisnis yang digambarkan dalam *system flow* konfirmasi *customer* pada PT. Indo Bismar Komputer dimulai dari dua jenis transaksi yaitu barang *service* dengan mengganti *sparepart* (C) dan barang *service* yang hanya memperbaiki *sparepart* saja (B).

Proses bisnis transaksi perbaikan *sparepart* (B): Untuk transaksi yang hanya memperbaiki *sparepart* saja maka dimulai dari bagian *admin service* yang akan diberi tanda terima *service* oleh teknisi. Lalu *admin service* mengkonfirmasi

*customer* dengan membuka *form* konfirmasi dan melakukan pencarian transaksi berdasarkan nomor *service* yang tertera dalam tanda terima *service* yang menempel pada barang *service* milik *customer*. Dari data konfirmasi yang berasal dari *database* jasa *service* dan *database* tanda terima *service* tersebut kemudian admin *service* memberitahu melalui pesan atau telepon bahwa barang sudah diperbaiki dan bisa diambil. Setelah admin *service* berhasil mengkonfirmasi *customer*, maka admin *service* mencatat dan menyimpan data konfirmasi tersebut ke dalam *database* konfirmasi.

Proses bisnis transaksi penggantian *sparepart* (C): Untuk transaksi yang perlu mengganti *sparepart* maka alur *system flow* dimulai dari bagian admin *service* yang hendak mengkonfirmasi *customer* melalui pesan atau telepon untuk memberitahu biaya *service* yang akan dikenakan dan memberitahu tentang penggantian *sparepart* lalu menanyakan kepada *customer* bahwa *service* akan dilanjutkan atau tidak. Jika *customer* menjawab dilanjutkan saja, maka teknisi akan mengambil *sparepart* ke bagian gudang.

Setelah teknisi mengambil *sparepart* di gudang, kemudian teknisi mengganti *sparepart* dan mulai men-*service* barang. Jika barang telah selesai di-*service*, teknisi menyerahkan barang pada admin *service* dan admin *service* membuka *form* konfirmasi dan melakukan pencarian transaksi berdasarkan nomor *service* yang tertera dalam tanda terima *service* yang menempel pada barang *service* milik *customer*. Dari data konfirmasi yang berasal dari *database* jasa *service*, *database* *sparepart* dan *database* tanda terima *service* tersebut kemudian admin *service* mencatat dan menyimpan data konfirmasi tersebut ke dalam *database* konfirmasi.



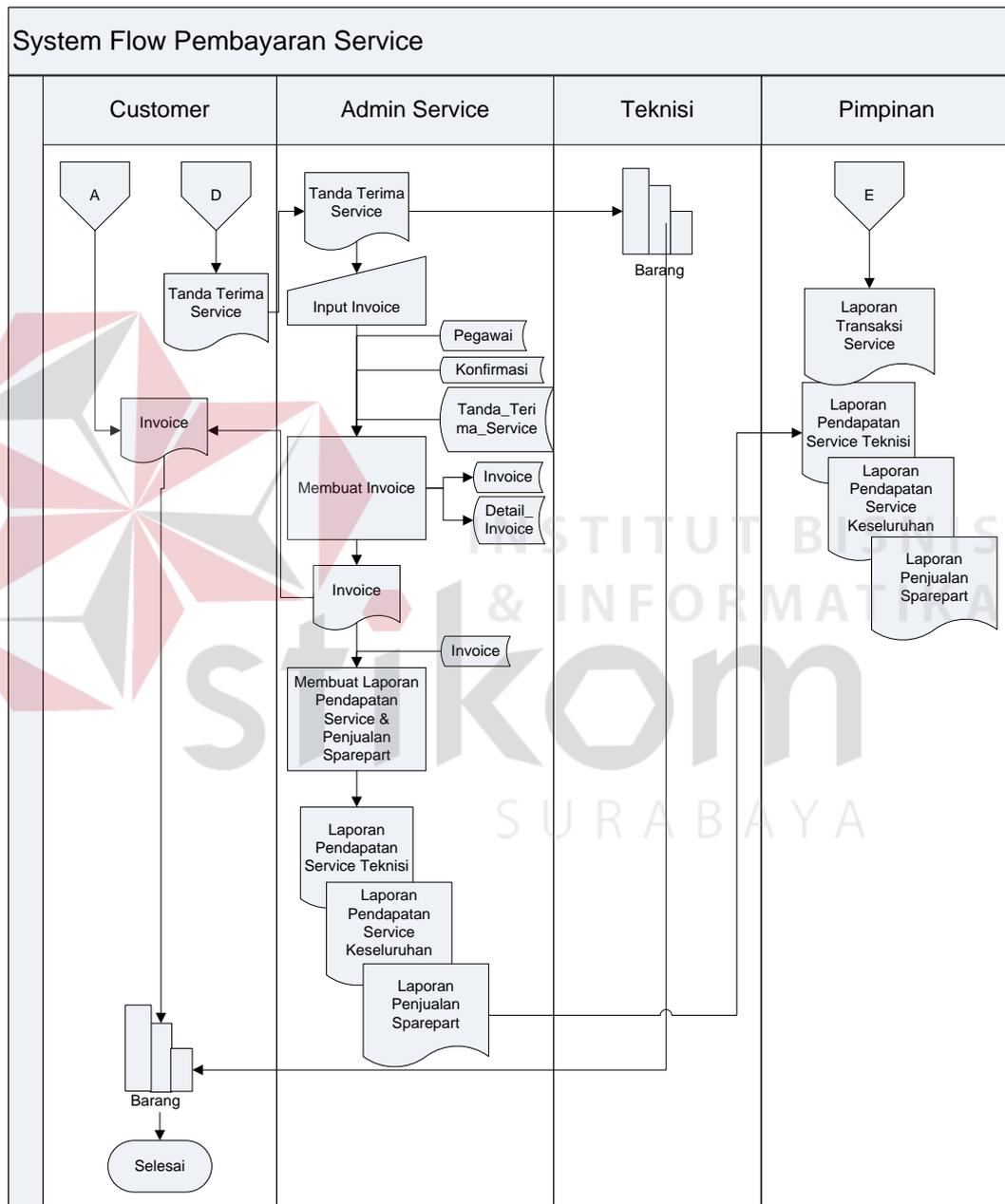
membutuhkan waktu lama (D) dan barang *service* yang langsung ditangani atau tidak menginap atau pengerjaan *service* cepat (A).

Untuk proses barang yang akan di-*service* tersebut ditinggal atau *service* inap atau pengerjaan *service* yang butuh waktu yang lama (D), maka *customer* harus membawa tanda terima *service* sebagai tanda bukti kepemilikan barang dan diberikan pada admin *service*. Admin *service* ke bagian teknisi untuk mengambil barang dan *running* barang *service* agar *customer* mengecek kembali barangnya. Dalam waktu yang sama, admin *service* membuka *form invoice* dan mencari nomor *service* lalu aplikasi akan menampilkan transaksi *invoice customer* yang berasal dari data pegawai, data konfirmasi dan data tanda terima *service*. Lalu admin *service* membuat *invoice* dan menyimpannya ke dalam *database invoice* dan *database detail invoice*. Kemudian admin *service* mencetaknya menjadi nota *invoice*. *Invoice* tersebut diserahkan kepada *customer*. Lalu *customer* membayar biaya *service* pada kasir.

Dari data *invoice* tersebut maka dibuatkan laporan pendapatan *service* tiap teknisi, laporan pendapatan *service* keseluruhan, laporan penjualan *spaerpart* dan laporan transaksi *service* yang setiap bulannya diserahkan kepada pimpinan.

Untuk proses barang *service* yang langsung ditangani atau tidak menginap atau pengerjaan *service* cepat (A), maka *customer* langsung menerima *invoice* dan langsung melakukan pembayaran kepada kasir. Selanjutnya *customer* mengambil barang dan *running* barang *service* untuk dicek kembali oleh *customer*.

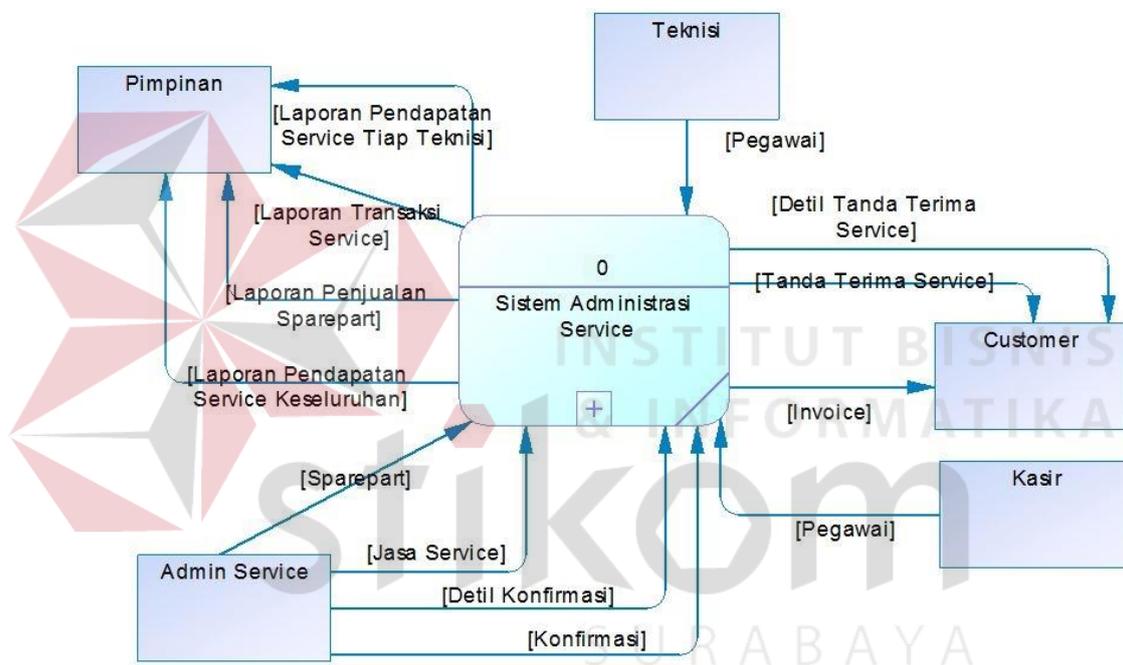
Berikut ini adalah rancangan *system flow* pembayaran *service* yang menjelaskan tentang alur proses transaksi oleh *admin service*. Dapat dilihat pada gambar 3.7.



Gambar 3.7 System Flow Proses Pembayaran Service

### 3.2.4 Context Diagram

*Context Diagram* rancang bangun aplikasi *service* pada PT. Indo Bismar Komputer secara umum menggambarkan tentang proses-proses apa saja yang sedang berjalan pada perusahaan tersebut. Pada *context diagram* dibawah ini melibatkan *entity-entity* yang saling berkaitan. *Context diagram* ini digambarkan pada gambar 3.8.

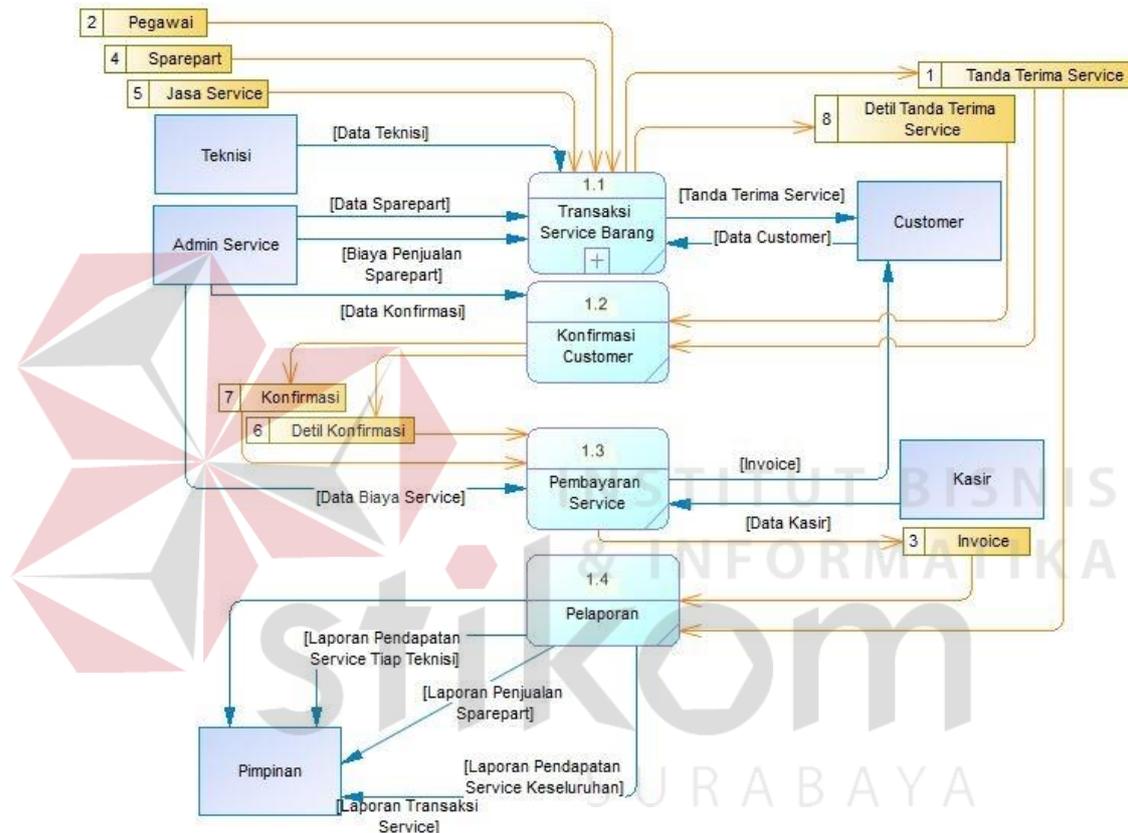


Gambar 3.8 *Context Diagram* Rancang Bangun Aplikasi *Service*

### 3.2.5 DFD Level 0 Rancang Bangun Aplikasi *Service*

*Data Flow Diagram* merupakan aliran data yang memiliki tingkatan atau tahapan desain yang memiliki beberapa tahapan level desain tergantung kebutuhan suatu sistem. Langkah pertama adalah merancang sistem satu persatu pada tingkat level tertinggi dalam DFD, kemudian baru membuat sistem dibawahnya yang lebih detail lagi.

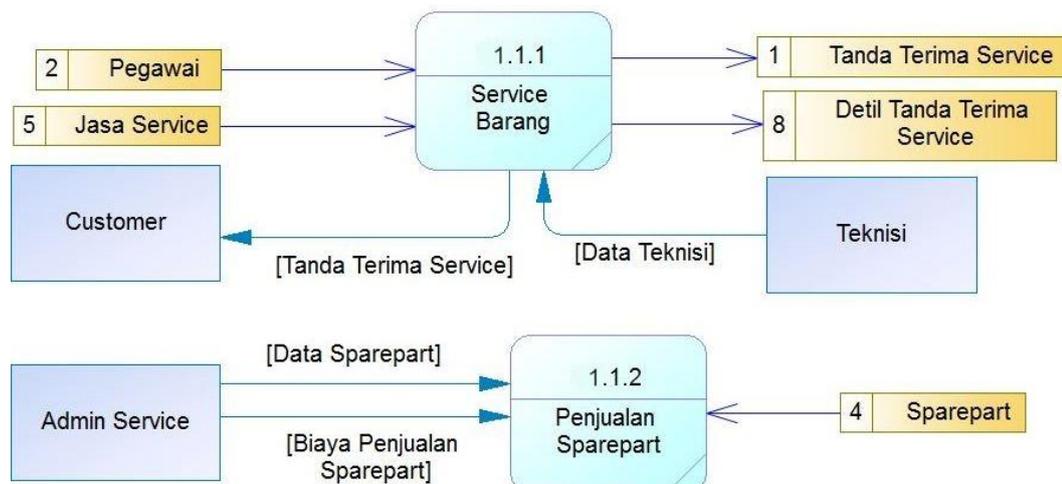
DFD *Level 0* dapat dihasilkan dari hasil *decompose context diagram*. DFD *level 0* ini menggambarkan tentang beberapa yang saling berkaitan dengan rancang bangun aplikasi *service* pada PT. Indo Bismar Komputer. Berikut adalah DFD *Level 0* yang dapat dilihat pada gambar 3.9.



Gambar 3.9 DFD *Level 0* Rancang Bangun Aplikasi *Service*

### 3.2.6 DFD *Level 1* Sub-Proses Transaksi *Service* Barang

Gambar 3.10 adalah sub-proses transaksi *service* barang yang ada pada DFD *Level 0* Aplikasi *Service* pada PT. Indo Bismar Komputer. Pada DFD *Level 1* menjelaskan proses *service* barang dan penjualan *sparepart* pada *customer*.



Gambar 3.10 DFD *Level 1* Sub-Proses Transaksi *Service Barang*

### 3.2.7 *Entity Relationship Diagram*

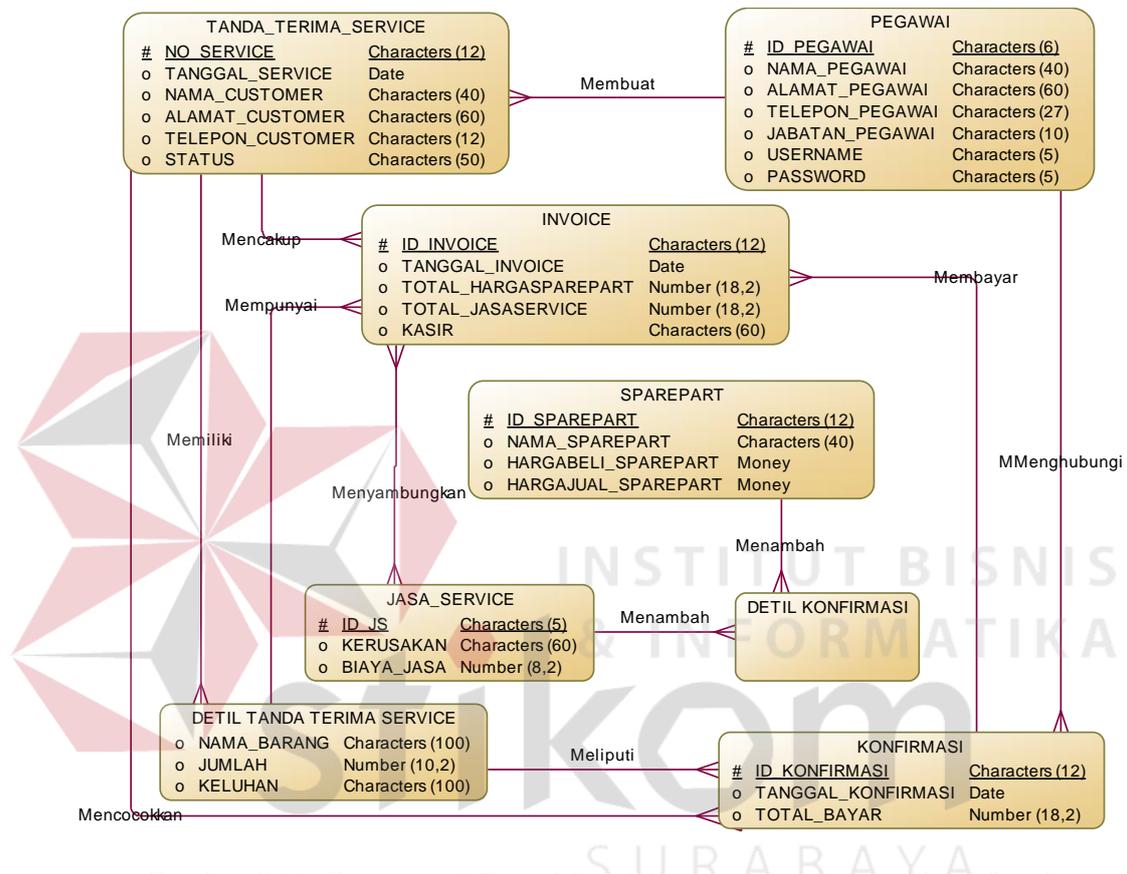
Dari analisis sistem diatas maka dapat dibuat suatu desain system yang digunakan untuk merancang *database* yaitu *Entity Relationship Diagram* (ERD) dengan menggambarkan atau menginterpretasikan, mendokumentasikan dan menentukan kebutuhan-kebutuhan struktur *database* aplikasi administrasi *service* PT. Indo Bismar Komputer.

Dalam *Entity Relationship Diagram* ini memiliki beberapa entitas atau *entity* yang saling berkaitan dalam menampilkan data-data yang dibutuhkan bagi suatu sistem yaitu:

#### A. *Conceptual Data Model*

*Conceptual Data Model* (CDM) merupakan sebuah model yang berfungsi untuk menunjukkan konsep tatanan basis data yang diterapkan pada sebuah aplikasi. Susunan basis data yang ditampilkan dengan tabel beserta *field*-nya mempunyai hubungan namun pada CDM tidak terlihat secara rinci pada kolom mana yang saling terhubung antar tabel. Penggunaan *Primary key*

untuk tabel-tabel penyusun basis data sebuah aplikasi juga dilakukan pada model ini. Pemodelan CDM yang digunakan untuk aplikasi *service* ini bisa dilihat pada gambar 3.11.

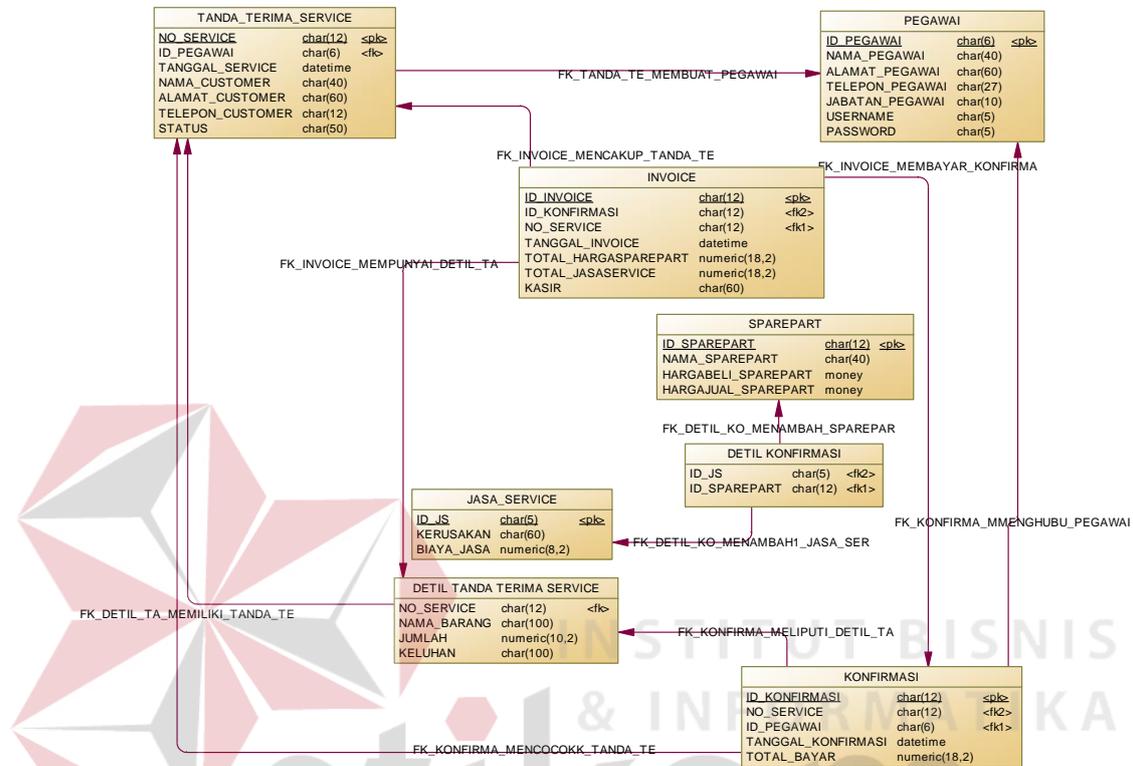


Gambar 3.11 *Conceptual Data Model Rancang Bangun Aplikasi Service*

## B. *Physical Data Model*

*Physical Data Model* (PDM) menjelaskan secara terperinci tentang struktur basis data yang dirancang sedemikian rupa agar terbentuk suatu program aplikasi. PDM merupakan hasil uraian *Conceptual Data Model* (CDM). Pada PDM terdapat sejumlah tabel-tabel yang saling tersusun dan menjadi sebuah *database* lengkap dengan *field-field* yang ada di dalam setiap tabel. Berikut

PDM pada rancang bangun aplikasi *service* PT. Indo Bismar Komputer dapat dilihat pada gambar 3.12.



Gambar 3.12 *Physical Data Model* Rancang Bangun Aplikasi *Service*

### 3.2.8 Struktur Basis Data

Tahapan yang akan dilakukan setelah melalui tahap merancang *Physical Data Model* adalah tahap pengembangan berupa rancangan struktur basis data. dalam struktur basis data ini akan menjelaskan nama tabel dan *field* yang terdapat dalam *Physical Data Model* diatas. Disini akan *Physical Data Model* dijelaskan lebih rinci lagi mengenai fungsi dari setaipvtabel beserta isinya. Struktur basis juga dapat digunakan untuk penyimpanan beberapa data-data yang sangat dibutuhkan dalam merancang bangun aplikasi *service* pada PT. Indo Bismar

Komputer. Struktur tabel aplikasi *service* pada PT. Indo Bismar Komputer adalah sebagai berikut:

a. Tabel Pegawai

Nama Tabel : PEGAWAI

Primary Key : ID\_PEGAWAI

Foreign Key : -

Fungsi : Tabel untuk memasukkan data-data pegawai

Tabel 3.1 Pegawai

No.	Field	Type	Length	Key
1.	ID_PEGAWAI	Char	6	<i>Primary Key</i>
2.	NAMA_PEGAWAI	Char	40	
4.	ALAMAT_PEGAWAI	Char	60	
5.	TELEPON_PEGAWAI	Char	27	
6.	JABATAN_PEGAWAI	Char	10	
7.	USERNAME	Char	5	
8.	PASSWORD	Char	5	

b. Tabel Data *Sparepart*

Nama Tabel : SPAREPART

Primary Key : ID\_SPAREPART

Foreign Key : -

Fungsi : Tabel untuk memasukkan data-data *sparepart*

Tabel 3.2 *Sparepart*

No.	Field	Type	Length	Key
1.	ID_SPAREPART	Char	12	<i>Primary Key</i>
2.	NAMA_SPAREPART	Char	40	

4.	HARGABELI_SPAREPART	Money
5.	HARGAJUAL_SPAREPART	Money

c. Tabel Jasa *Service*

Nama Tabel : JASA SERVICE

Primary Key : ID\_JS

Foreign Key : -

Fungsi : Tabel untuk memasukkan data-data jasa *service*

Tabel 3.3 Jasa *Service*

No.	Field	Type	Length	Key
1.	ID_BS	Char	5	<i>Primary Key</i>
2.	KERUSAKAN	Char	60	
3.	BIAYA_JASA	Numeric	8, 2	

d. Tanda Terima *Service*

Nama Tabel : TANDA\_TERIMA\_SERVICE

Primary Key : NO\_SERVICE

Foreign Key : ID\_PEGAWAI

Fungsi : Tabel untuk memasukkan data-data transaksi *service*

Tabel 3.4 Tanda Terima *Service*

No.	Field	Type	Length	Key
1.	NO_SERVICE	Char	12	<i>Primary Key</i>
2.	ID_PEGAWAI	Char	6	<i>Foreign Key</i>
3.	TANGGAL_SERVICE	Date Time		
4.	NAMA_CUSTOMER	Char	40	
5.	ALAMAT_CUSTOMER	Char	60	
6.	TELEPON_CUSTOMER	Char	12	

7.	STATUS	Char	50
----	--------	------	----

e. Tabel Detail Tanda Terima Service

Nama Tabel : DETIL TANDA TERIMA SERVICE  
 Primary Key : -  
 Foreign Key : NO\_SERVICE  
 Fungsi : Tabel untuk memasukkan data-data detail tanda terima  
*service*

Tabel 3.5 Detail Tanda Terima *Service*

No.	Field	Type	Length	Key
1.	NO_SERVICE	Char	12	<i>Foreign Key</i>
2.	NAMA_BARANG	Char	100	
3.	JUMLAH	Numeric	10, 2	
4.	KELUHAN	Char	100	

f. Tabel Konfirmasi

Nama Tabel : KONFIRMASI  
 Primary Key : ID\_KONFIRMASI  
 Foreign Key : NO\_SERVICE, ID\_PEGAWAI  
 Fungsi : Tabel untuk memasukkan data-data konfirmasi

Tabel 3.6 Konfirmasi

No.	Field	Type	Length	Key
1.	ID_KONFIRMASI	Char	12	<i>Primary Key</i>
2.	NO_SERVICE	Char	12	<i>Foreign Key</i>
3.	ID_PEGAWAI	Char	6	<i>Foreign Key</i>
4.	TANGGAL_KONFIRMASI	Datetime		
5.	TOTAL_BAYAR	Numeric	18, 2	

## g. Tabel Detail Konfirmasi

Nama Tabel : DETIL KONFIRMASI

Primary Key : ID\_BS, ID\_KONFIRMASI

Foreign Key : -

Fungsi : Tabel untuk memasukkan data-data detail konfirmasi

Tabel 3.7 Detail Konfirmasi

No.	Field	Type	Length	Key
1.	ID_JS	Char	5	Primary Key
2.	ID_KONFIRMASIhju	Char	12	Primary Key
3.	TOTAL	Numeric	16, 2	

## h. Tabel Invoice

Nama Tabel : INVOICE

Primary Key : ID\_INVOICE

Foreign Key : ID\_KONFIRMASI, NO\_SERVICE

Fungsi : Tabel untuk memasukkan data-data *invoice*

Tabel 3.8 Invoice

No.	Field	Type	Length	Key
1.	ID_INVOICE	Char	12	Primary Key
2.	ID_KONFIRMASI	Char	12	Foreign Key
3.	NO_SERVICE	Char	12	Foreign Key
4.	TANGGAL_INVOICE	Datetime		
5.	TOTAL_HARGASPAREPART	Numeric	18, 2	
6.	TOTAL_JASASERVICE	Numeric	18, 2	
7.	KASIR	Char	60	

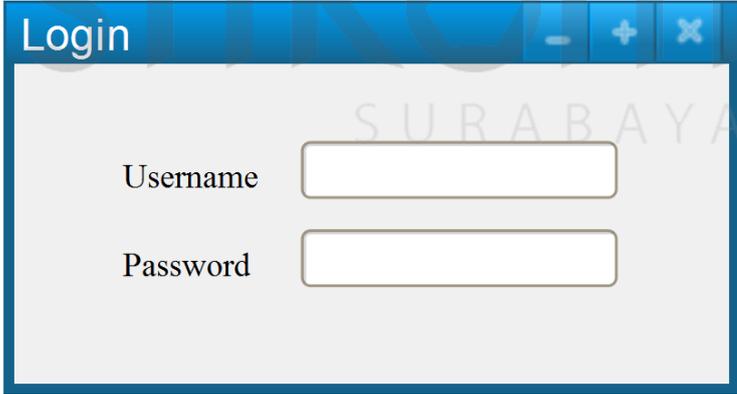
### 3.2.9 Desain *Input Output*

Desain *input output* merupakan langkah yang harus dibuat setelah menyusun sekumpulan perancangan sistem yang berguna untuk menjelaskan secara detail mengenai jalannya suatu sistem tersebut. Di dalam desain *input output* terdapat rancangan *form-form* khusus untuk memasukkan data dan *form-form* khusus untuk menampilkan data yang ditampilkan dalam bentuk laporan-laporan dan digunakan sebagai mendokumentasikan suatu proses bisnis.

Pengimplementasian dari desain *input output* dapat dijelaskan dalam keterangan dibawah ini:

#### 1. Desain Halaman *Login*

Dibawah ini adalah tampilan dari *form* atau halaman *login* yang digunakan untuk validasi identitas pengguna atau *user* yang membuka dan menggunakan aplikasi *service*. Desain *form* ini dapat diamati pada gambar 3.13 dibawah.

The image shows a screenshot of a web browser window with a blue title bar containing the text 'Login' and standard window control buttons (minimize, maximize, close). The main content area has a light gray background and contains two input fields. The first field is labeled 'Username' and the second is labeled 'Password'. Both fields are empty and have a thin gray border. A large, semi-transparent watermark for 'stikom SURABAYA' is overlaid on the background of the form.

Gambar 3.13 Desain Halaman *Login*

## 2. Desain Halaman Menu Utama

Dibawah ini adalah tampilan dari *form* atau halaman menu utama yang digunakan ketika pengguna atau *user* berhasil *login* atau masuk sebagai *username*. Desain *form* ini dapat diamati pada gambar 3.14 dibawah.



Gambar 3.14 Desain Halaman Menu Utama

## 3. Desain Halaman *Master* Pegawai

Dibawah ini adalah tampilan dari *form master* pegawai yang digunakan untuk memasukkan dan menyimpan data pegawai. Desain *form* ini dapat diamati pada gambar 3.15 dibawah.

Gambar 3.15 Desain Halaman *Master Pegawai*

#### 4. Desain Halaman *Master Sparepart*

Dibawah ini adalah tampilan dari *form master sparepart* yang digunakan untuk memasukkan dan menyimpan data *sparepart*. Desain *form* ini dapat diamati pada gambar 3.16 dibawah.

Gambar 3.16 Desain Halaman *Master Sparepart*

### 5. Desain Halaman Master Jasa Service

Dibawah ini adalah tampilan dari *form* jasa service yang digunakan untuk memasukkan dan menyimpan data jasa service. Desain *form* ini dapat diamati pada gambar 3.17 dibawah.

The image shows a software interface for managing service data. The main window, titled "Jasa Service", features a form for entering new records. The form includes fields for "ID", "Kerusakan" (Damage Type), and "Biaya Jasa" (Service Cost), with a "Baru" button for adding new entries. Below the form are "SIMPAN" (Save), "UBAH" (Edit), and "BATAL" (Cancel) buttons. A secondary window, "Data Jasa Service", displays a search field for "Cari Jenis Kerusakan" and a table with columns for "ID", "Kerusakan", and "Biaya Jasa". The table is currently empty. A large watermark for "stikom" and "INSTITUT BISNIS & INFORMATIKA SURABAYA" is overlaid on the image.

Gambar 3.17 Desain Halaman Master Jasa Service

### 6. Desain Halaman Transaksi Tanda Terima Service

Dibawah ini adalah tampilan dari *form* halaman transaksi tanda terima service yang digunakan untuk memasukkan, mencatat *input*/data dan mencetak transaksi tanda terima service dari *customer*. Desain *form* ini dapat diamati pada gambar 3.18 dibawah.

Gambar 3.18 Desain Halaman Transaksi Tanda Terima *Service*

## 7. Desain Halaman Transaksi Konfirmasi

Dibawah ini adalah tampilan dari *form* halaman transaksi konfirmasi yang digunakan untuk memasukkan atau mencatat *input*/data transaksi konfirmasi untuk mengkonfirmasi *customer*. Desain *form* ini dapat diamati pada gambar 3.19 dibawah.

Gambar 3.19 Desain Halaman Transaksi Konfirmasi

### 8. Desain Halaman Transaksi *Invoice*

Dibawah ini adalah tampilan dari *form* halaman transaksi *invoice* yang digunakan untuk memasukkan, mencatat *input*/data dan mencetak transaksi *invoice* sebagai bukti pembayaran bagi *customer*. Desain *form* ini dapat diamati pada gambar 3.20 dibawah.

The image shows a software interface for creating an invoice. It features a search bar for service numbers, a table for listing services, and various input fields for customer and invoice details. A table at the bottom is designed for itemized charges, including a 'Total Bayar' column. Navigation buttons for printing and canceling are located on the right side of the form.

Gambar 3.20 Desain Halaman Transaksi *Invoice*

### 9. Desain Halaman Laporan Transaksi *Service*

Dibawah ini adalah tampilan dari *form* halaman laporan transaksi *service* yang diperoleh dari kumpulan-kumpulan tanda terima *service*. Halaman ini bertujuan untuk mengetahui laporan *list* atau daftar *customer* yang melakukan *service* barang pada PT. Indo Bismar Komputer dalam periode waktu tertentu. Desain *form* ini dapat diamati pada gambar 3.21 dibawah.

The screenshot shows a software window titled "Laporan Transaksi Service". The window contains a form on the left for selecting transaction periods and a table on the right for displaying service records.

**PILIH PERIODE TRANSAKSI**

Periode Harian  
 Tanggal Mulai : dd/mm/yy  
 Tanggal Akhir : dd/mm/yy  
 Bulan : Pilih Bulan...  
 Tahun : Pilih Tahun...

Periode Bulanan  
 Bulan : Pilih Bulan...  
 Tahun : Pilih Tahun...

Periode Tahunan  
 Tahun : Pilih Tahun...

TAMPIL

No.	Tanggal Service	Teknisi	Customer	Telp	Barang	Jumlah Kerusakan

Gambar 3.21 Desain *Form* Laporan Transaksi *Service*

#### 10. Desain Halaman Laporan Penjualan *Sparepart*

Dibawah ini adalah tampilan dari *form* halaman laporan penjualan *sparepart* yang berasal dari pembelian *sparepart* oleh *customer* dalam proses *service* barang jika ada *sparepart* yang rusak. Halaman ini bertujuan untuk mengetahui laporan *list* atau daftar *sparepart* yang telah terjual pada PT. Indo Bismar Komputer dalam periode waktu tertentu. Desain *form* ini dapat diamati pada gambar 3.22 dibawah

The image shows a software window titled "Laporan Penjualan Service" with a sub-header "LAPORAN PENJUALAN SPAREPART". On the left, there is a section titled "PILIH PERIODE TRANSAKSI" with three sub-sections: "Periode Harian" (Daily), "Periode Bulanan" (Monthly), and "Periode Tahunan" (Yearly). Each section contains input fields for "Tanggal Mulai" (Start Date), "Tanggal Akhir" (End Date), "Bulan" (Month), and "Tahun" (Year). The "Periode Harian" section has "dd/mm/yy" in the date fields. The "Periode Bulanan" section has a "Pilih Bulan..." dropdown menu. The "Periode Tahunan" section has "dd/mm/yy" in the date field. Below these filters are two buttons: "TAMPIL" and "CETAK". To the right of the filters is a table with the following columns: "No.", "Tanggal", "Nama Sparepart", "Pcs", "Harga Beli", "Harga Jual", and "Laba". The table is currently empty.

Gambar 3.22 Desain *Form* Laporan Penjualan *Sparepart*

#### 11. Desain Halaman Laporan Pendapatan *Service*

Dibawah ini adalah tampilan dari *form* halaman laporan pendapatan *service* yang berasal dari pelayanan *service* dan penjualan *sparepart*. Halaman ini terpisah menjadi dua *textbox* yaitu Laporan Pendapatan *Service* Teknisi yang menampilkan pendapatan *service* setiap teknisi yang telah dipilih dan Laporan Pendapatan *Service* Perusahaan yang menampilkan pendapatan *service* keseluruhan pada PT. Indo Bismar Komputer dalam periode waktu tertentu. Desain *form* ini dapat diamati pada gambar 3.23 dan 3.24 dibawah

Gambar 3.23 Desain *Form* Laporan Pendapatan *Service* Teknisi

Gambar 3.24 Desain *Form* Laporan Pendapatan *Service* Keseluruhan

## 12. Desain Tanda Terima *Service*

Nota ini berisi tentang barang *service* dari *customer* dan identitas *customer*.

Nota ini berfungsi sebagai tanda bukti bahwa *customer* telah menaruh barang *service* kepada PT. Indo Bismar Komputer dan juga sebagai tanda bukti bagi

teknisi bahwa barang yang di-*service* ini milik *customer* yang telah tertulis identitasnya di nota. Desain nota ini dapat diamati pada gambar 3.25 dibawah.

 <b>PT. INDO BISMAR</b> Bendul Merisi Permai R-9/R-11 Telp. (031) 8474685, 8437701 Fax. (031) 8490632		<b>TANDA TERIMA SERVICE</b>	
Tanggal :		Nama :	
No. Service :		No. Telp/HP :	
Teknisi :		Alamat :	
Nama Barang	Jumlah	Keluhan	

Menyerahkan, Penerima,

.....

- Barang tidak diambil lebih dari 3 (tiga) bulan bukan tanggung jawab kami  
 - Tanpa tanda terima ini, barang tidak dapat diambil  
 - Garansi berjalan setelah 3 (tiga) hari konirmasi barang selesai *service*

Gambar 3.25 Desain Tanda Terima *Service*

### 13. Desain *Invoice*

Nota ini berisi tentang bukti pembayaran dari dan untuk *customer*. Nota ini berfungsi sebagai tanda bukti bahwa *customer* telah melunasi biaya *service* kepada PT. Indo Bismar Komputer dan juga sebagai tanda bukti bagi teknisi bahwa barang yang di-*service* telah diserahkan kepada *customer*. Desain nota ini dapat diamati pada gambar 3.26 dibawah



### 15. Desain Laporan Penjualan *Sparepart*

Laporan ini berisi tentang kumpulan seluruh data-data penjualan *sparepart* dalam proses *service* barang dengan target penjualan *sparepart* yang telah ditentukan oleh pimpinan dalam penjualan per periode tertentu. Desain laporan ini dapat diamati pada gambar 3.28 dibawah.

**PT. INDO BISMAR**  
Bendul Merisi Pemai R-9/R-11  
Telp. (031) 8474685, 8437701  
Fax. (031) 8490632

**LAPORAN PENJUALAN  
SPAREPART**

Periode:

No.	Tanggal	Nama Sparepart	Qty	Harga Beli	Harga Jual	Laba
<b>Total</b>						

Surabaya,.....

Gambar 3.28 Desain Laporan Penjualan *Sparepart*

### 16. Desain Laporan Pendapatan *Service* Teknisi

Laporan ini berisi tentang kumpulan data-data penjualan *sparepart* dan biaya jasa *service* teknisi dalam proses *service* barang dengan target penjualan *sparepart* & jasa *service* yang telah ditentukan oleh pimpinan dalam penjualan tiap masing-masing teknisi per periode tertentu. Desain laporan ini dapat diamati pada gambar 3.29 dibawah.

 <b>PT. INDO BISMAR</b> Bendul Merisi Permai R-9/R-11 Telp. (031) 8474685, 8437701 Fax. (031) 8490632		<b>LAPORAN PENDAPATAN SERVICE TEKNISI</b> Periode: Teknisi:					
No.	Invoice	Tanggal	Kerusakan	Biaya Service	Sparepart	Biaya Sparepart	Total
<b>Total</b>							

Surabaya,.....

---

Gambar 3.29 Desain Laporan Pendapatan *Service* Teknisi

### 17. Desain Laporan Pendapatan *Service* Keseluruhan

Laporan ini berisi tentang kumpulan seluruh data-data penjualan *sparepart* dan biaya jasa *service* teknisi dalam proses *service* barang dengan target penjualan *sparepart* & jasa *service* yang telah ditentukan oleh pimpinan per periode tertentu. Desain laporan ini dapat diamati pada gambar 3.30 dibawah.

 <b>PT. INDO BISMAR</b> Bendul Merisi Permai R-9/R-11 Telp. (031) 8474685, 8437701 Fax. (031) 8490632		<b>LAPORAN PENDAPATAN SERVICE KESELURUHAN</b> Periode:					
No.	Invoice	Tanggal	Kerusakan	Biaya Service	Sparepart	Biaya Sparepart	Total
<b>Total</b>							

Surabaya,.....

---

Gambar 3.30 Desain Laporan Pendapatan *Service* Keseluruhan

### 3.3 Rancangan Pengujian dan Evaluasi Sistem

#### 3.3.1 Desain Uji Coba

Desain uji coba dilakukan untuk dapat mengetahui apakah aplikasi telah sesuai dengan fungsi dan kebutuhan. Secara umum pengujian dilakukan pada data *master*, proses transaksi dan, pelaporan. Berikut ini adalah desain uji coba yang akan dilakukan.

##### a. Desain Uji Coba *Form Login*

Tabel 3.9 Desain Uji Coba *Form Login*

Obyek Pengujian			<i>Form Login</i>	
Keterangan			Mengetahui tampilan dan fungsi yang terdapat dalam <i>form login</i> dapat berjalan sesuai yang diharapkan.	
<i>Test Case ID</i>	Uji Coba	<i>Input</i>	<i>Output yang Diharapkan</i>	Status
1.	<i>Textbox username</i>	Memasukkan karakter <i>username</i>	Jika karakter <i>username</i> yang dimasukkan pengguna tidak sesuai dengan karakter <i>username</i> yang telah tersimpan di dalam <i>database</i> maka ketika klik <i>button</i> masuk akan tampil “ <i>Username</i> atau <i>password</i> yang anda masukkan salah. Silahkan coba lagi.”	
			Jika karakter <i>username</i> yang dimasukkan pengguna sudah sesuai dengan karakter <i>username</i> yang telah tersimpan di dalam <i>database</i> maka ketika klik <i>button</i> masuk akan tampil “Selamat Datang. Anda berhasil masuk pada Aplikasi Service PT. Indo Bismar Komputer.”	
2.	<i>Textbox password</i>	Memasukkan karakter <i>password</i>	- Karakter yang dimasukkan akan tampil dalam bentuk simbol “*”	
			- Jika karakter <i>password</i> yang dimasukkan pengguna tidak	

			sesuai dengan karakter <i>password</i> yang telah tersimpan di dalam <i>database</i> maka ketika klik <i>button</i> masuk akan tampil “ <i>Username</i> atau <i>password</i> yang anda masukkan salah. Silahkan coba lagi.”	
			Jika karakter <i>password</i> yang dimasukkan pengguna sudah sesuai dengan karakter <i>password</i> yang telah tersimpan di dalam <i>database</i> maka ketika klik <i>button</i> masuk akan tampil “Selamat Datang. Anda berhasil masuk pada Aplikasi Service PT. Indo Bismar Komputer.”	
3.	<i>Button login</i>	Klik <i>button login</i>	Jika karakter <i>username</i> dan <i>password</i> yang dimasukkan pengguna tidak sesuai dengan karakter <i>username</i> dan <i>password</i> yang telah tersimpan di dalam <i>database</i> maka ketika klik <i>button</i> masuk akan tampil “ <i>Username</i> atau <i>password</i> yang anda masukkan salah. Silahkan coba lagi.”	
			Jika karakter <i>username</i> dan <i>password</i> yang dimasukkan pengguna sudah sesuai dengan karakter <i>username</i> dan <i>password</i> yang telah tersimpan di dalam <i>database</i> maka ketika klik <i>button</i> masuk akan tampil “Selamat Datang. Anda berhasil masuk pada Aplikasi Service PT. Indo Bismar Komputer.”	

b. Desain Uji Coba *Form* Menu Utama

Tabel 3.10 Desain Uji Coba *Form* Menu Utama

<b>Objek Pengujian</b>		<i>Form</i> Menu Utama		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> Menu Utama dapat berjalan sesuai yang diharapkan.		
<b>Test</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>

<b>Case ID</b>				
1.	<i>Button Master</i>	Mengarahkan <i>cursor</i> ke <i>button</i> Master	Menampilkan <i>Link Label</i> Master Pegawai, <i>Link Label</i> Master <i>Sparepart</i> dan <i>Link Label</i> Master <i>Jasa Service</i> .	
		<i>Link Label</i> Pegawai	Menampilkan <i>Form</i> Master Pegawai	
		<i>Link Label</i> <i>Sparepart</i>	Menampilkan <i>Form</i> Master <i>Sparepart</i>	
		<i>Link Label</i> <i>Jasa Service</i>	Menampilkan <i>Form</i> Master <i>Jasa Service</i>	
2.	<i>Button Transaksi</i>	Mengarahkan <i>cursor</i> ke <i>button</i> Transaksi	Menampilkan <i>Link Label</i> Transaksi Tanda Terima <i>Service</i> , <i>Link Label</i> Transaksi Konfirmasi dan <i>Link Label</i> Transaksi <i>Invoice</i> .	
		<i>Link Label</i> Tanda Terima <i>Service</i>	Menampilkan <i>Form</i> Transaksi Tanda Terima <i>Service</i>	
		<i>Link Label</i> Konfirmasi	Menampilkan <i>Form</i> Transaksi Konfirmasi	
		<i>Link Label</i> <i>Invoice</i>	Menampilkan <i>Form</i> Transaksi <i>Invoice</i>	
3.	<i>Button Laporan</i>	Mengarahkan <i>cursor</i> ke <i>button</i> Laporan	Menampilkan <i>Link Label</i> Laporan Pendapatan <i>Service</i> , <i>Link Label</i> Laporan Transaksi <i>Service</i> dan <i>Link Label</i> Laporan Penjualan <i>Sparepart</i> .	
		<i>Link Label</i> Laporan Pendapatan <i>Service</i>	Menampilkan <i>Form</i> Laporan Pendapatan <i>Service</i>	
		<i>Link Label</i> Laporan Transaksi <i>Service</i>	Menampilkan <i>Form</i> Laporan Transaksi <i>Service</i>	
		<i>Link Label</i> Laporan Penjualan <i>Sparepart</i>	Menampilkan <i>Form</i> Laporan Penjualan <i>Sparepart</i>	
4.	<i>Button Log Out</i>	<i>Button Log Out</i>	Mengarahkan pengguna untuk keluar dari aplikasi <i>service</i> .	

c. Desain Uji Coba *Button Form Master*Tabel 3.11 Desain Uji Coba *Button Form Master*

<b>Objek Pengujian</b>		<i>Button Form Master</i>		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>button form</i> master dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> pada <i>Form Master</i>	<i>Button Baru</i>	Memberi kode baru setiap kali memasukkan data master	
		<i>Button Simpan</i>	Menampilkan <i>message box</i> “Data Tersimpan” jika data-data yang diperlukan telah terisi semua sesuai dengan tipe data yang disyaratkan.	
			Menampilkan <i>message box</i> “Isi Data Terlebih Dahulu” jika ada data-data yang belum diisi dan tidak sesuai dengan tipe data yang disyaratkan.	
		<i>Button Ubah</i>	Menampilkan <i>message box</i> “Data Tersimpan” jika data-data yang diperlukan telah terisi semua sesuai dengan tipe data yang disyaratkan.	
			Menampilkan <i>message box</i> “Isi Data Terlebih Dahulu” jika ada data-data yang belum diisi dan tidak sesuai dengan tipe data yang disyaratkan.	
<i>Button Batal</i>	<i>Textbox</i> yang awalnya terisi akan menjadi kosong.			

d. Desain Uji Coba *Form Master Pegawai*Tabel 3.12 Desain Uji Coba *Form Master Pegawai*

<b>Objek Pengujian</b>		<i>Form Master Pegawai</i>		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> master pegawai dapat berjalan sesuai yang diharapkan.		
<b>Test Case</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>

ID				
1.	<i>Button</i> Simpan, Ubah dan Batal	Klik <i>button</i> Batal	<i>Button</i> Simpan, Ubah dan Batal tidak bisa digunakan ketika ID Pegawai masih kosong.	
		Klik <i>button</i> Baru	<i>Button</i> Ubah tidak berfungsi ketika <i>user</i> memasukkan data pegawai baru.	
		Klik <i>datagridview</i>	<i>Button</i> Simpan tidak berfungsi ketika <i>user</i> mengubah, mengecek atau melihat data dalam <i>datagridview</i> untuk ditampilkan ke dalam <i>textbox</i> .	
2.	<i>Combobox</i> Jabatan	Klik <i>combobox</i> jabatan	<i>Combobox</i> jabatan akan menampilkan beberapa macam jabatan ketika <i>user</i> klik <i>combobox</i> jabatan.	
3.	<i>Textbox</i> Username, Password & Cari	Mengisi <i>textbox</i> <i>username</i>	<i>Error Provider</i> “Maksimal 5 Karakter” akan muncul ketika pengguna mengisi <i>textbox</i> <i>username</i>	
			<i>Error Provider</i> “Tidak Boleh Lebih dari 5 Karakter” akan muncul ketika pengguna mengisi <i>textbox</i> <i>username</i> lebih dari lima karakter.	
		Mengisi <i>textbox</i> <i>password</i>	<i>Error Provider</i> “Maksimal 5 Karakter” akan muncul ketika pengguna mengisi <i>textbox</i> <i>password</i>	
			<i>Error Provider</i> “Tidak Boleh Lebih dari 5 Karakter” akan muncul ketika pengguna mengisi <i>textbox</i> <i>password</i> lebih dari lima karakter.	
		Mengisi <i>textbox</i> Cari	Menampilkan data berdasarkan nama pegawai yang diketik dalam <i>textbox</i> cari	

e. Desain Uji Coba *Form* Master *Sparepart*

Tabel 3.13 Desain Uji Coba *Form* Master *Sparepart*

<b>Objek Pengujian</b>	<i>Form</i> Master <i>Sparepart</i>
<b>Keterangan</b>	Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> master <i>sparepart</i> dapat

<i>Test Case ID</i>	<i>Uji Coba</i>	<i>Input</i>	<i>Output yang Diharapkan</i>	<i>Status</i>
			berjalan sesuai yang diharapkan.	
1.	<i>Button Simpan, Ubah dan Batal</i>	Klik <i>button</i> Batal	<i>Button</i> Simpan, Ubah dan Batal tidak bisa digunakan ketika ID <i>Sparepart</i> masih kosong.	
		Klik <i>button</i> Baru	<i>Button</i> Ubah tidak berfungsi ketika <i>user</i> memasukkan data <i>sparepart</i> baru.	
		Klik <i>datagridview</i>	<i>Button</i> Simpan tidak berfungsi ketika <i>user</i> mengubah, mengecek atau melihat data dalam <i>datagridview</i> untuk ditampilkan ke dalam <i>textbox</i> .	
2.	<i>Combobox Status Stok Sparepart</i>	Klik <i>combobox</i> status stok <i>sparepart</i>	<i>Combobox</i> status stok <i>sparepart</i> akan menampilkan status aktif atau tidak aktif ketika <i>user</i> klik <i>combobox</i> status stok <i>sparepart</i> .	
3.	<i>Textbox Harga Beli, Harga Jual &amp; Cari</i>	Mengisi <i>textbox</i> Harga Beli	<i>Error Provider</i> "Data Tidak Boleh Kosong dan Harus Angka" akan muncul ketika pengguna mengisi dengan karakter angka atau dikosongi.	
		Mengisi <i>textbox</i> Harga Jual	<i>Error Provider</i> "Data Tidak Boleh Kosong dan Harus Angka" akan muncul ketika pengguna mengisi dengan karakter angka atau dikosongi.	
		Mengisi <i>textbox</i> Cari	Menampilkan data berdasarkan nama <i>sparepart</i> yang diketik dalam <i>textbox</i> cari	

f. Desain Uji Coba *Form* Master Jasa ServiceTabel 3.14 Desain Uji Coba *Form* Master Jasa Service

<b>Objek Pengujian</b>		<i>Form</i> Master Jasa Service		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> master jasa service dapat berjalan sesuai yang diharapkan.		
<i>Test Case ID</i>	<i>Uji Coba</i>	<i>Input</i>	<i>Output yang Diharapkan</i>	<i>Status</i>

1.	<i>Button</i> Simpan, Ubah dan Batal	Klik <i>button</i> Batal	<i>Button</i> Simpan, Ubah dan Batal tidak bisa digunakan ketika ID <i>Jasa Service</i> masih kosong.	
		Klik <i>button</i> Baru	<i>Button</i> Ubah tidak berfungsi ketika <i>user</i> memasukkan data <i>jasa service</i> baru.	
		Klik <i>button</i> Simpan	<i>Button</i> Simpan tidak berfungsi ketika <i>user</i> mengubah, mengecek atau melihat data dalam <i>datagridview</i> untuk ditampilkan ke dalam <i>textbox</i> .	
2.	<i>Combobox</i> Status	Klik <i>combobox</i> status	<i>Combobox</i> status akan menampilkan beberapa status <i>jasa service</i> apakah masih aktif atau sudah tidak aktif.	
3.	<i>Textbox</i> Biaya <i>Jasa</i> , <i>Cari</i>	Mengisi <i>textbox</i> biaya <i>jasa</i>	<i>Error Provider</i> “Data Tidak Boleh Kosong dan Harus Angka” akan muncul ketika pengguna mengisi <i>textbox</i> biaya <i>jasa</i>	
		Mengisi <i>textbox</i> <i>cari</i>	Menampilkan data berdasarkan nama kerusakan yang diketik dalam <i>textbox</i> <i>cari</i>	

g. Desain Uji Coba *Form* Transaksi Tanda Terima *Service*Tabel 3.15 Desain Uji Coba *Form* Transaksi Tanda Terima *Service*

<b>Objek Pengujian</b>		<i>Form</i> Transaksi Tanda Terima <i>Service</i>		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> transaksi tanda terima <i>service</i> dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> Baru, Batal, Simpan Data <i>Service</i> dan Tambah	Klik <i>button</i> Batal	<i>Button</i> Batal, Simpan Data <i>Service</i> dan Tambah tidak bisa digunakan ketika ID Pegawai masih kosong.	
		Klik <i>button</i> Baru	<i>Button</i> Baru tidak berfungsi ketika <i>user</i> memasukkan data pegawai baru sedangkan <i>button</i> Batal, Simpan Data <i>Service</i> dan Tambah bisa digunakan.	
		Klik	Ketika <i>user</i> mengklik data	

		<i>datagridview</i>	dalam <i>datagridview field</i> atau <i>textbox</i> data barang akan tetap kosong.	
		Klik <i>button</i> Tambah	Data barang akan tampil dalam <i>datagridview</i> ketika klik <i>button</i> Tambah.	
		Klik <i>button</i> Simpan Data <i>Service</i>	Data <i>service</i> tidak akan tersimpan dan ada peringatan “Periksa kembali! Data tidak boleh ada yang kosong” jika ada <i>field</i> yang masihh kosong belum diisi oleh <i>user</i> .	
2.	<i>Combobox</i> Jabatan	Klik <i>dropdownlist</i> jabatan	<i>Dropdownlist</i> nama teknisi akan tampil ketika <i>user</i> klik <i>dropdownlist</i> nama teknisi.	

#### h. Desain Uji Coba *Form* Transaksi Konfirmasi

Tabel 3.16 Desain Uji Coba *Form* Transaksi Konfirmasi

<b>Objek Pengujian</b>		<i>Form</i> Transaksi Konfirmasi		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> transaksi konfirmasi dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> Tambah, Tambah <i>Sparepart</i> , Ubah, Simpan dan Batal		<i>Button</i> tambah, tambah <i>sparepart</i> , ubah, simpan dan batal tidak berfungsi ketika data <i>service</i> dalam <i>datagridview</i> konfirmasi tidak dipilih	
		Klik <i>datagridview</i> konfirmasi	<i>Button</i> tambah, tambah <i>sparepart</i> , simpan dan batal berfungsi ketika data <i>service</i> dalam <i>datagridview</i> konfirmasi dipilih	
		Klik <i>button</i> Tambah	<i>Datagridview</i> kerusakan akan tampil ketika <i>user</i> klik <i>button</i> tambah. Kerusakan bisa lebih dari satu jenis kerusakan.	
		Klik <i>button</i> Tambah <i>Sparepart</i>	<i>Datagridview</i> <i>sparepart</i> akan tampil ketika <i>user</i> klik <i>button</i> tambah <i>Sparepart</i> . <i>Sparepart</i> bisa lebih dari satu jenis	

			barang.	
		Klik <i>button</i> Simpan	Data barang akan tampil dalam <i>datagridview</i> konfirmasi ketika klik <i>button</i> Simpan.	
			Ketika <i>combobox</i> status konfirmasi diisi dengan status "Sudah Dikonfirmasi" maka daftar <i>customer</i> yang sudah dikonfirmasi akan ditampilkan dalam transaksi <i>Invoice</i>	
2.	<i>Combobox</i> Kerusakan & <i>Sparepart</i>	Klik <i>Combobox</i> Kerusakan	Jenis kerusakan akan tampil ketika <i>user</i> klik <i>combobox</i> kerusakan.	
		Klik <i>Combobox</i> <i>Sparepart</i>	<i>Sparepart</i> akan tampil ketika <i>user</i> klik <i>combobox</i> <i>sparepart</i> .	
		Klik <i>Combobox</i> Status Konfirmasi	Jenis status konfirmasi akan tampil ketika <i>user</i> klik <i>combobox</i> status konfirmasi.	
3.	<i>Textbox</i> Cari, No. <i>Service</i> , Tanggal <i>Service</i> , Nama Teknisi, Nama <i>Customer</i> , Telp. <i>Custome</i> & Harga	Mengisi <i>Textbox</i> Cari	Daftar <i>customer</i> yang belum dikonfirmasi akan ditampilkan sesuai dengan nomor <i>service</i> yang diketik pada <i>textbox</i> cari.	
		Klik <i>datagridview</i> konfirmasi	No. <i>Service</i> , Tanggal <i>Service</i> , Nama Teknisi, Nama <i>Customer</i> , Telp. <i>Customer</i> berfungsi ketika data <i>service</i> dalam <i>datagridview</i> konfirmasi dipilih	
		Pilih <i>combobox</i> kerusakan	Nominal harga akan tampil ketika <i>user</i> memilih jenis kerusakan pada <i>combobox</i> kerusakan	
		Pilih <i>combobox</i> <i>sparepart</i>	Nominal harga akan tampil ketika <i>user</i> memilih jenis <i>sparepart</i> pada <i>combobox</i> <i>sparepart</i>	
4.	<i>Datagridview</i> Kerusakan & <i>Sparepart</i>	Pilih <i>datagridview</i> konfirmasi	Ketika klik salah satu daftar <i>customer</i> yang telah diketahui total pembayarannya namun belum dikonfirmasi pada <i>datagridview</i> konfirmasi maka data tersebut akan ditampilkan ke dalam <i>field</i> dan <i>datagridview</i> Kerusakan serta	

			<i>datagridview sparepart</i>	
--	--	--	-------------------------------	--

i. Desain Uji Coba *Form Transaksi Invoice*

Tabel 3.17 Desain Uji Coba *Form Transaksi Invoice*

<b>Objek Pengujian</b>		<i>Form Transaksi Invoice</i>		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> transaksi <i>invoice</i> dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> Simpan, Ubah dan Batal	Klik <i>button</i> Batal	Membuat semua <i>button</i> tidak berfungsi dan <i>textbox</i> menjadi kosong	
		Klik <i>datagridview invoice</i>	Menampilkan <i>button</i> batal, <i>button</i> cetak & <i>datagridview</i> data barang	
2.	<i>Combobox</i> Kair	Klik <i>combobox</i> kasir	<i>Combobox</i> kasir menampilkan beberapa nama kasir untuk dipilih salah satu	
3.	<i>Textbox</i> Cari	Mengisi <i>textbox</i> cari	Menampilkan daftar <i>invoice</i> berdasarkan nomor <i>service</i> yang diketik dalam <i>textbox</i> cari nomor <i>service</i>	
4.	<i>Datagridview</i> Data Barang	Klik <i>datagridview invoice</i>	Menampilkan <i>button</i> batal, <i>button</i> cetak & <i>datagridview</i> data barang	

j. Desain Uji Coba *Form Laporan Pendapatan Service Teknisi*

Tabel 3.18 Desain Uji Coba *Form Laporan Pendapatan Service Teknisi*

<b>Objek Pengujian</b>		<i>Form Laporan Pendapatan Service Teknisi</i>		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> laporan pendapatan <i>service</i> dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> Tampil	Klik <i>button</i> Tampil	Menampilkan laporan pendapatan <i>service</i> teknisi	

2.	<i>Combobox</i> Teknisi	Klik <i>combobox</i> teknisi	<i>Combobox</i> teknisi menampilkan beberapa nama teknisi untuk dipilih salah satu	
----	----------------------------	------------------------------------	---	--

k. Desain Uji Coba *Form* Laporan Pendapatan *Service* Keseluruhan

Tabel 3.19 Desain Uji Coba *Form* Laporan Pendapatan *Service* Keseluruhan

<b>Objek Pengujian</b>		<i>Form</i> Laporan Pendapatan <i>Service</i> Keseluruhan		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> laporan pendapatan <i>service</i> keseluruhan dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> Tampil	Klik <i>button</i> Tampil	Menampilkan laporan pendapatan <i>service</i> keseluruhan yang diterima perusahaan	

l. Desain Uji Coba *Form* Laporan Transaksi *Service*

Tabel 3.20 Desain Uji Coba *Form* Laporan Transaksi *Service*

<b>Objek Pengujian</b>		<i>Form</i> Laporan Transaksi <i>Service</i>		
<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> laporan transaksi <i>service</i> dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button</i> Tampil	Klik <i>button</i> Tampil	Menampilkan laporan transaksi <i>service</i> yang diterima perusahaan	

m. Desain Uji Coba *Form* Laporan Penjualan *Sparepart*

Tabel 3.21 Desain Uji Coba *Form* Laporan Penjualan *Sparepart*

<b>Objek Pengujian</b>	<i>Form</i> Laporan Penjualan <i>Sparepart</i>
------------------------	--

<b>Keterangan</b>		Mengetahui tampilan dan fungsi yang terdapat dalam <i>form</i> laporan penjualan <i>sparepart</i> dapat berjalan sesuai yang diharapkan.		
<b>Test Case ID</b>	<b>Uji Coba</b>	<b>Input</b>	<b>Output yang Diharapkan</b>	<b>Status</b>
1.	<i>Button Tampil</i>	Klik <i>button Tampil</i>	Menampilkan laporan penjualan <i>sparepart</i> yang diterima perusahaan	

