BAB III

PEMBAHASAN

3.1 Deskripsi Kerja Praktek

3.1.1 Tempat dan Waktu Kerja Praktek

Kerja praktek dilaksanakan di Telkom kantor pusat lantai 5 FBCC (Finance Billing Collection Center), yang beralamat di Jln Japati no 1 Bandung. Dengan durasi selama 1 (satu) bulan, terhitung sejak tanggal 1 Desember 2010 sampai dengan tanggal 31 Desember 2010. Jam kerja praktek disesuaikan dengan jam kerja pegawai di Telkom, yaitu masuk dari hari hari Senin sampai dengan Jumat, kecuali hari libur. Setiap hari kerja dimulai dari jam 8:00 sampai dengan jam 17:00.

3.1.2 Kegiatan Kerja Praktek

Kegiatan yang dilakukan selama kerja praktek adalah sebagai berikut:

1. Pengenalan tentang FBCC (Finance Billing Collection Center) dan kegiatannya yang bertanggung jawab dalam pemrosesan dan penyimpanan data percakapan pelanggan, data billing atau tagihan pelanggan, data pembayaran tagihan pelanggan, dan data tunggakan.
2. Mempelajari cara kerja dan kegiatan yang berhubungan dengan pengolahan data retensi tunda cabut.
3. Membangun aplikasi monitoring data pelangan retensi tunda cabut yang berbasis web.

3.1.3 Data Kerja Praktek

Data kerja praktek yang dimaksud adalah semua data dan informasi yang didapat selama Kerja Praktek berlangsung. Khususnya data dan informasi lengkap mengenai program retensi tunda cabut ini, yaitu data kebutuhan aplikasi, data pelanggan, data pembayaran dan tunggakan, data pelanggan yang termasuk ke dalam program retensi tunda cabut.

3.1.3.1 Data kebutuhan aplikasi

Kebutuhan aplikasi monitoring data tunda cabut ini dikemukakan pada rapat pembahasan kebutuhan aplikasi. Penulis menyimpulkan beberapa hal yang penting yaitu sebagai berikut:

1. Aplikasi yang akan dibangun bertujuan untuk mengurangi proses manual dalam pengolahan data tunda cabut.
2. Data pelanggan yang mempunyai tunggakan dan termasuk ke dalam program retensi tunda cabut diambil dari data yang dikelola secara manual oleh UCS Regional III. Data yang ditangani adalah data produk telepon dan speedy.
3. Aplikasi yang akan dibangun harus bisa mengolah data tunggakan, data status pelanggan, dan data status tunda cabut secara otomatis.
    a. Data tunggakan detail per periode atau bulan tagihan diolah menjadi data tunggakan detail per CCA, sehingga akan tersedia informasi jumlah dan total tunggakan, dan informasi tunggakan pelanggan per periode tagihan dalam satu *record* CCA.
    b. Data status pelanggan adalah status fastel apakah masih aktif atau sudah cabut. Data ini diambil dari data pelanggan.
    c. Data status tunda cabut adalah sebagai berikut :
        - Potensi cabut (pelanggan aktif dan ada tunggakan tapi masih belum ada di periode retensi tunda cabut)

- Pending atau tunda cabut (pelanggan aktif dan masih ada tunggakan)
- Cabut (pelanggan yang sudah cabut dan masih ada tunggakan)
- Lunas sebagian (sudah ada pembayaran tapi masih ada sisa tunggakan)
- Lunas (pelanggan yang sudah tidak ada tunggakan)
- Lunas dan cabut (pelanggan yang sudah tidak ada tunggakan tetapi status pelanggannya sudah cabut)

4. Setiap fastel yang sudah lunas, aplikasi akan mengambil data tanggal pembayarannya. Tanggal ini dipakai sebagai data pelunasan retensi tunda cabut untuk membayarkan bonus atau *winback* untuk pengelola data retensi tunda cabut. Data detail pelanggan (data *winback*) ini nantinya bisa diambil melalui aplikasi yang akan dibuat.
5. Aplikasi yang dibangun harus menggunakan bahasa pemrograman PHP4 karena disesuaikan dengan PHP yang digunakan pada server yang ada.
6. Database yang digunakan di server adalah Oracle 9i.
7. Data yang akan dimonitor harus bisa menampilkan view summary data status rentensi tunda cabut per Produk (telepon dan speedy), per CCAT (*Customer Category*), per CS Area, per Commerce Office, dan per STO.
8. Data usulan tagihan nol untuk data pelanggan retensi tunda cabut terbaru harus bisa diupdate dan diambil (di-*download*) dari aplikasi pada akhir bulan. Data ini digunakan sebagai *attachment* usulan tagihan nol untuk proses billing. Data pelanggan yang termasuk dalam retensi tunda cabut, harus dinolkan tagihannya.

9. Aplikasi ini tidak perlu menggunakan username dan password, karena autentifikasi user sudah ditangani oleh *Infranet Controller Telkom*. Untuk bisa menggunakan jaringan internal telkom, user harus memasukkan username (NIK Nomor Induk Karyawan) dan password ke dalam *Infranet Controller Telkom*. Disamping itu aplikasi ini bersifat publik maka bisa dilihat oleh siapa saja yang bisa masuk ke jaringan internal telkom.

3.1.3.2 Data pelanggan

Data pelanggan Telkom dikelola di dalam suatu aplikasi yang disebut aplikasi i-siska. Aplikasi ini digunakan untuk mengelola pelanggan mulai dari proses pasang baru di plasa telkom, proses billing (proses penghitungan tagihan pelanggan), proses mutasi pelanggan (ganti paket speedy, penambahan fitur telepon, pemblokiran, dan sebagainya), dan update data-data pelanggan (data alamat, data segmen pelanggan, data aktivasi dan data pencabutan pelanggan, dan sebagainya).

Data pelanggan telkom dapat diambil dari database aplikasi atau dari database mirror aplikasi. Untuk keperluan aplikasi ini, data pelanggan diambil dari database mirror. Data pelanggan ini nanti akan diproses dan di-*insert*-kan ke tabel pelanggan di database aplikasi via *procedure* yang dijalankan di *jobs oracle*.

Untuk tahap awal, data yang dipakai adalah data pelanggan Telkom UCS (*Unit Consumer Services*) Regional III Jawa Barat. Meliputi CS (*Consumer Service*) Area Bandung Barat, Bandung Timur, Cirebon, Tasikmalaya dan Sukabumi.

Istilah yang digunakan sebagai kode pelanggan adalah *ncli* (nomor *client*). Satu pelanggan memiliki satu buah *ncli* dan satu pelanggan bisa memiliki lebih dari satu fastel atau produk telkom. Istilah yang digunakan sebagai kode fastel yang dimiliki oleh pelanggan adalah CCA (*Customer Contract Account*).

3.1.3.3 Data pembayaran dan tunggakan

Data pembayaran dan data tunggakan fastel pelanggan dikelola di dalam aplikasi yang disebut TREMS (*Telkom Revenue Management System*) yang berbasis SAP. Untuk keperluan aplikasi monitoring data tunda cabut ini, Telkom sudah menyediakan data pembayaran dan tunggakan di dalam sebuah server datawarehouse. Update data pembayaran dilakukan setiap 15 menit dan update data tunggakan dilakukan setiap tiga jam.

Data tunggakan ini nanti akan diproses dan di-*insert*-kan ke tabel tunggakan di database aplikasi via *procedure* yang dijalankan di *jobs oracle*. Data pembayaran tidak akan diambil karena jumlah *record*-nya besar sehingga membutuhkan *tablespace* yang besar.

3.1.3.4 Data potensi dan retensi tunda cabut

Data potensi tunda cabut adalah data pelanggan yang berpotensi sebagai data retensi tunda cabut. Data ini diambil setelah tanggal 20 setiap bulannya.

Data retensi tunda cabut adalah data telepon yang menunggak pada periode N+3 (tunggakan tiga bulan tagihan) atau data speedy yang menunggak pada periode N+2 (tunggakan dua bulan tagihan). Untuk data history retensi tunda cabut yang lalu diambil dari data yang dikelola oleh Telkom UCS Regional III.

Selama masa retensi pelanggan, tagihan pelanggan ini tidak diproses sehingga tagihan pelanggan sebesar nol rupiah. Selama ada tunggakan, pelanggan tidak akan bisa menggunakan telepon atau speedynya. Jika pelanggan ingin menggunakan fastel lagi, pelanggan harus melunasi tunggakannya.

3.2 Analisis dan Perancangan Sistem

3.2.1 Analisis Sistem

Analisis sistem merupakan sebuah tahapan untuk membangun ataupun mengembangkan suatu sistem.

3.2.1.1 Analisis Masalah

Permasalahan yang muncul terkait dengan retensi tunda cabut ini adalah bagaimana memberikan informasi yang lengkap mengenai retensi tunda cabut tersebut. Karena masih dilakukan proses manual dalam pengolahan data retensi tunda cabut ini, user atau pegawai telkom yang mengelola data ini menjadi kesulitan dalam memproses dan menyajikan data ini.

Berdasarkan hasil analisa permasalahan tersebut, maka dapat diidentifikasikan permasalahan sebagai berikut:

1. Pengolahan data masih dilakukan secara manual karena belum adanya aplikasi.
2. Untuk mendapatkan data status pelanggan retensi memerlukan waktu yang lama, karena harus membutuhkan data pelanggan dan data tunggakan yang didownload dari aplikasi lain secara manual. Waktu yang dibutuhkan untuk mendownload data tersebut relatif lama.
3. Kesulitan dalam menyajikan data detail pelanggan yang termasuk ke dalam usulan tagihan nol (data pelanggan retensi tunda cabut yang terbaru). Karena tidak ada update otomatis maka ada selisih data yang sudah melakukan pembayaran tapi tetap masuk ke dalam usulan tagihan nol.
4. Kesulitan dalam mengambil data *winback* (data fastel yang lunas) karena user harus melakukan pengecekan secara manual.

3.2.1.2 Analisis Pengkodean

Pengkodean untuk data dalam aplikasi ini mengikuti pengkodean yang sudah dipakai di Telkom. Pengkodean untuk data fastel pelanggan menggunakan CCA (*Customer Contract Account*). Setiap fastel pelanggan memiliki CCA yang unik. Misalkan jika pelanggan memiliki empat buah fastel maka pelanggan tersebut memiliki empat buah CCA.

Format CCA adalah 999 99999999 9999 dengan 3 digit pertama adalah ID mesin aplikasi (mesin aplikasi pengelola data pelanggan berbeda-beda di setiap regional), 8 digit kedua adalah ID pelanggan atau biasa disebut NCLI (Nomor Client atau ID pelanggan), dan 4 digit terakhr adalah nomor urutan fastel yang dimiliki satu pelanggan.

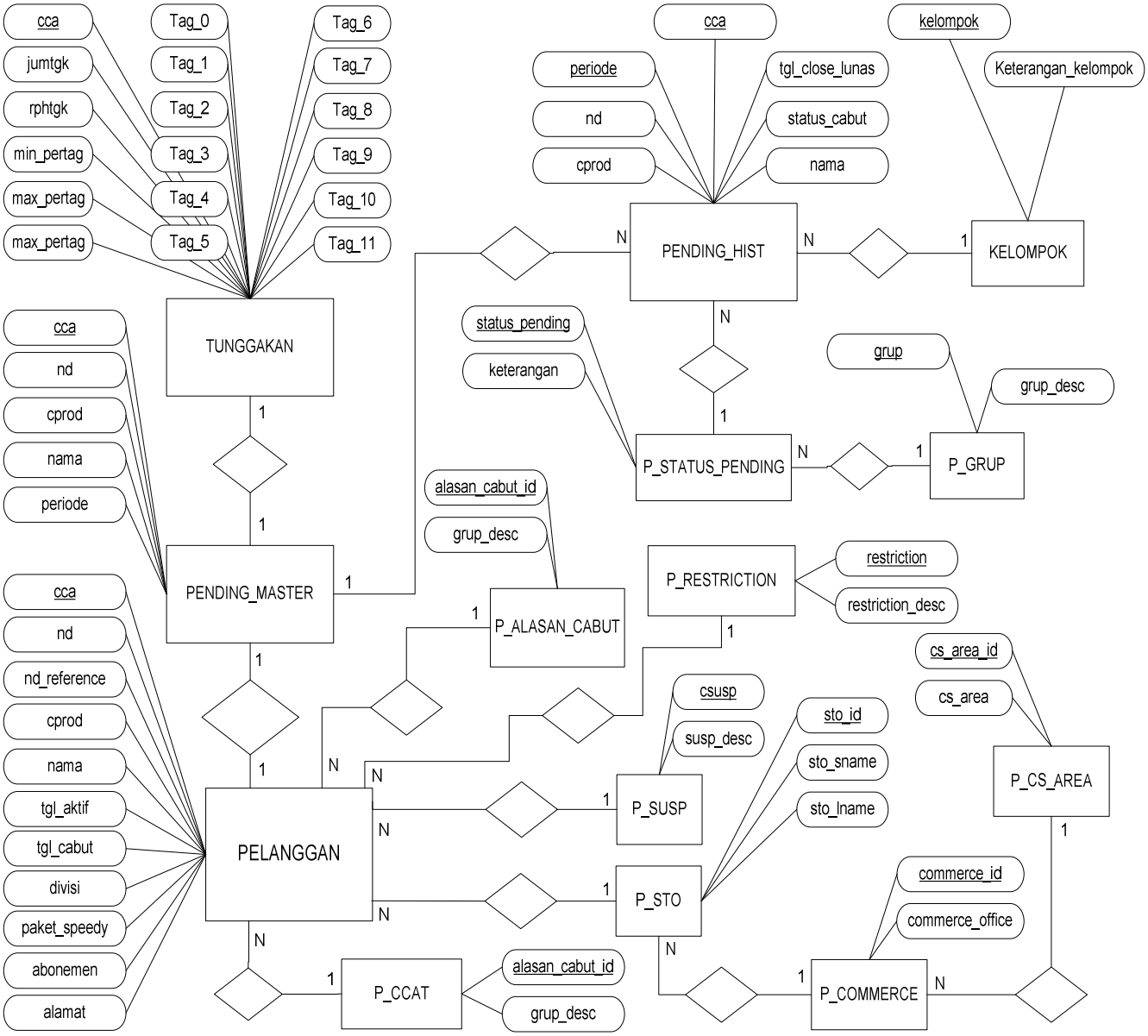
Contoh CCA : 800000138660003, yang artinya ID mesin = 800, ncli = 13866, dan urutan 3 yang artinya ini adalah fastel pertama yang dimiliki pelanggan.

3.2.1.3 Analisis User atau Pengguna

User atau pengguna sistem ini adalah pegawai yang bekerja di TELKOM. Di dalam aplikasi ini tidak diperlukan autentifikasi user, karena semua pegawai di lingkungan jaringan internal TELKOM bisa menggunakan aplikasi ini. Autentifikasi user sudah ditangani oleh *Infranet Controller Telkom* pada saat user ingin menggunakan jaringan internal telkom.

3.2.1.4 Analisis Basis Data

Dari hasil analisis, didapat data-data yang akan dipakai dalam aplikasi ini. Kemudian dari data yang telah diperoleh, dibangun sebuah desain basis data dan desain fitur-fitur lainnya. Penulis menggunakan *Entity Relational Diagram* (ERD) untuk merancang basis data. ERD yang dibuat adalah sebagai berikut :



Gambar 3.1 ERD aplikasi monitoring data retensi tunda cabut

3.2.1.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional merupakan tahap pendefinisian layanan atau fungsi atau proses yang harus disediakan oleh sistem, bagaimana reaksi sistem terhadap input dan apa yang harus dilakukan oleh sistem pada situasi tertentu.

Berdasarkan analisis terhadap permasalahan yang ada, maka fungsional-fungsional yang terdapat pada sistem informasi ini adalah sebagai berikut :

1. Pengambilan data pelanggan

   Proses ini melakukan pengambilan data pelanggan dari server data warehouse telkom. Data pelanggan diupdate setiap hari dan ditransfer ke tabel pelanggan di aplikasi ini via *procedure* yang dijalankan di *jobs oracle*.

2. Pengambilan data tunggakan

   Proses ini melakukan pengambilan data tunggakan dari server data warehouse telkom. Proses ini dijalankan 45 menit setelah update data pelanggan di server data warehouse selesai dilakukan. Data tunggakan ini diupdate setiap 3 jam dan ditransfer ke tabel tunggakan di aplikasi ini via *procedure* yang dijalankan di *jobs oracle*.

3. Pengambilan data potensi retensi tunda cabut

   Proses ini melakukan pengambilan data fastel yang menunggak (speedy tunggakan dua bulan dan telepon tunggakan tiga bulan) pada posisi tanggal 21 setiap bulannya. Jika pada tanggal 1 bulan depannya fastel tersebut masih menunggak maka fastel tersebut diubah statusnya menjadi pelanggan retensi tunda cabut.

4. Update status data potensi dan data retensi tunda cabut

Update status data ini dilakukan setiap ada update data pelanggan dan update data tunggakan. Status data tersebut adalah potensi, pending, lunas sebagian, lunas, cabut, lunas dan cabut.

5. Update data winback

   Proses ini dilakukan untuk mengambil data tanggal pelunasan tagihan pelanggan.

6. View data detail pelanggan usulan tagihan nol

   Data ini adalah data fastel yang merupakan data retensi tunda cabut terbaru. Data ini juga termasuk data fastel yang dulunya pernah tercatat dalam history data retensi tunda cabut yang sudah lunas. Aplikasi akan menampilkan detail fastel-fastel tersebut dan dapat didownload ke dalam file text.

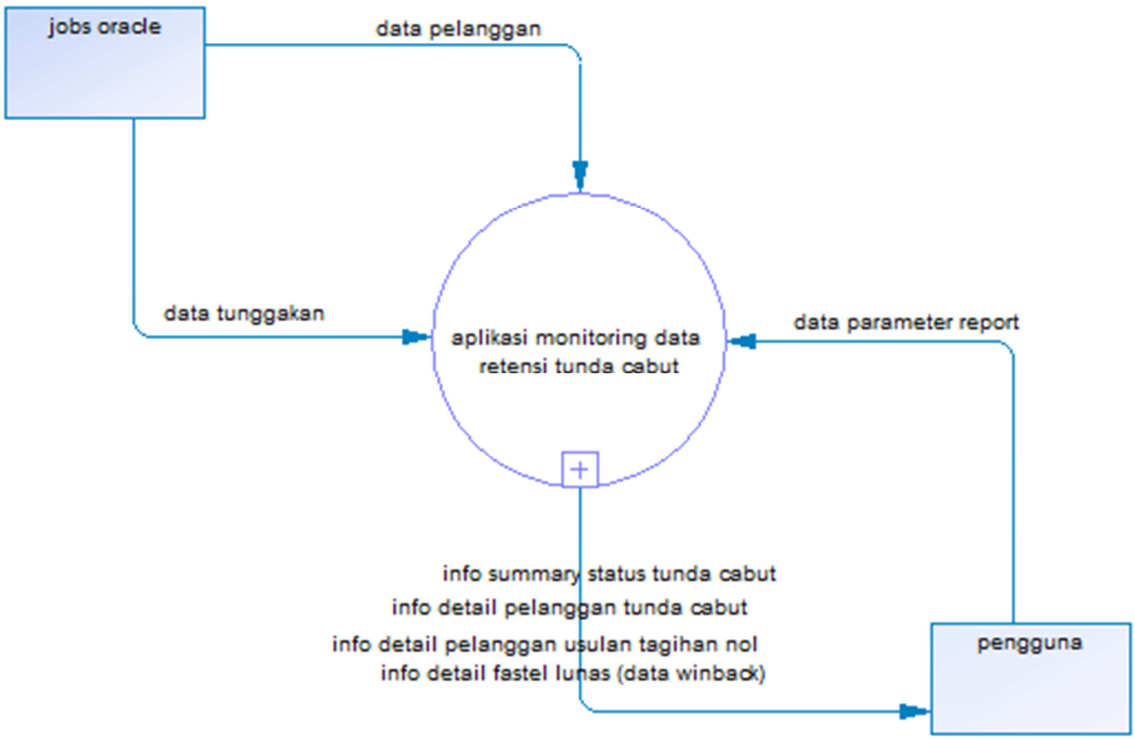
7. View data detail pelanggan winback retensi

   Data ini adalah data fastel retensi tunda cabut yang sudah melunasi tunggakannya dan sudah ada tanggal pelunasannya. Aplikasi akan menampilkan detail fastel-fastel tersebut dan dapat didownload ke dalam file text.

Analisis kebutuhan fungsional dilakukan untuk memberikan gambaran aliran data yang ada pada program aplikasi yang akan dibangun. Kebutuhan fungsional pada aplikasi ini meliputi diagram konteks, data flow diagram, dan spesifikasi proses.

3.2.1.5.1 Diagram Konteks

Diagram konteks merupakan suatu model yang menjelaskan secara global bagaimana data digunakan dan ditransformasikan untuk proses atau dengan kata lain merupakan gambaran umum mengenai sistem yang menggambarkan aliran data kedalam dan keluar sistem. Adapun diagram konteks yang diusulkan adalah sebagai berikut :



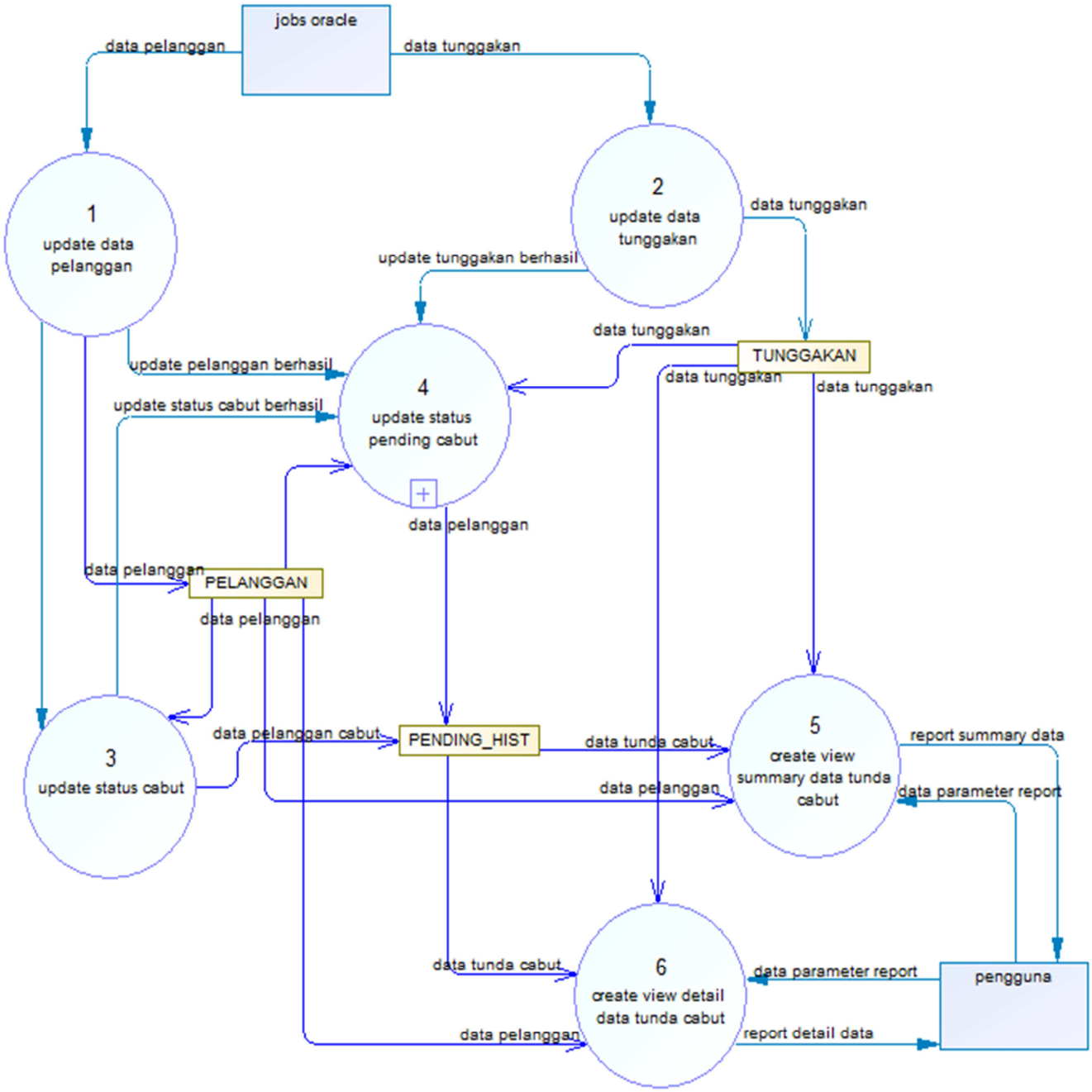
Gambar 3.2 Diagram Konteks aplikasi monitoring data retensi tunda cabut

3.2.1.5.2 Data Flow Diagram

Data Flow Diagram (DFD) adalah representasi dari suatu sistem yang menggambarkan bagian-bagian dari sistem tersebut berserta seluruh keterlibatan diantara bagian yang ada. Arus data pada Flow Diagram (DFD) dapat berupa masukan untuk sistem ataupun keluaran dari sistem, sehingga akan menghasilkan sebuah keluaran yang akan disampaikan kepada pengguna sistem.

1. **DFD Level 0**

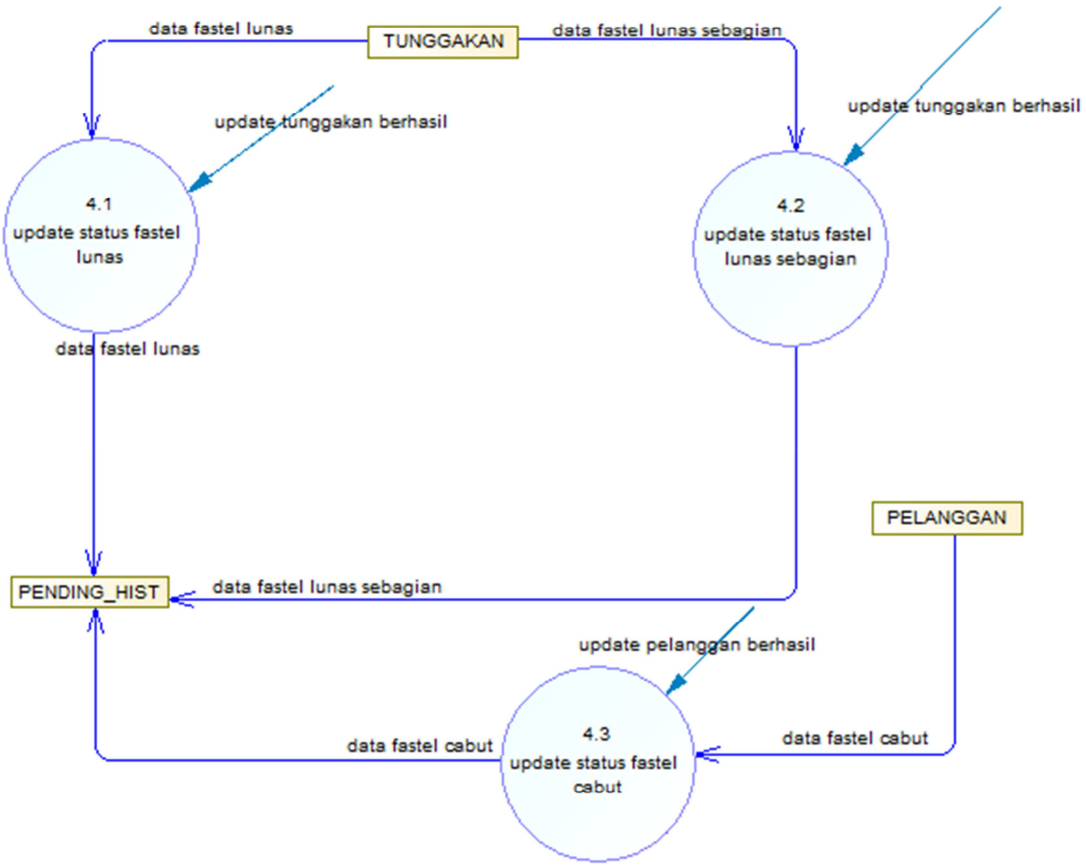
Proses-proses yang ada di dalam DFD level 0 ini dapat dilihat pada gambar di bawah ini :



Gambar 3.3 DFD level 0 aplikasi monitoring data retensi tunda cabut

2. **DFD Level 1 untuk Proses 1.4 (update status pending cabut)**

Proses-proses yang ada di dalam DFD level 0 ini dapat dilihat pada gambar di bawah ini :



Gambar 3.4 DFD level 1 untuk proses 1.4 (update status pending cabut)

3.2.1.5.3 Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan semua proses model aliran data yang disajikan pada diagram aliran data. Spesifikasi proses yang terdapat pada Data Flow Diagram dapat dilihat pada tabel berikut ini:

No	Proses	Keterangan
1	No. Proses	1
	Nama Proses	Update data pelanggan
	Deskripsi	Proses pengambilan data pelanggan terbaru dari database lain ke tabel di database aplikasi ini
	Source (Sumber)	jobs oracle
	Input	Data pelanggan
	Output	Data pelanggan
	Destination (Tujuan)	jobs oracle
	Logika Proses	Begin If exist(data pelanggan) then truncate data Load data pelanggan update End
No	**Proses**	**Keterangan**
2	No. Proses	2
	Nama Proses	Update data tunggakan
	Deskripsi	Proses pengambilan data tunggakan terbaru dari database lain ke tabel di database aplikasi ini
	Source (Sumber)	jobs oracle
	Input	Data tunggakan
	Output	Data tunggakan
	Destination (Tujuan)	jobs oracle
	Logika Proses	Begin If exist(data tunggakan) then truncate data Load data tunggakan update End

No	Proses	Keterangan
3	No. Proses	3
	Nama Proses	Update status cabut
	Deskripsi	Proses update status fastel yang sudah cabut ke table master data tunda cabut

	Source (Sumber)	jobs oracle
	Input	Data fastel cabut
	Output	Data fastel cabut
	Destination (Tujuan)	jobs oracle

	Logika Proses	Begin
		If exist(data fastel cabut) then
		Update status cabut
		End

No	**Proses**	**Keterangan**
4	No. Proses	4.1
	Nama Proses	Update status fastel lunas
	Deskripsi	Proses update status fastel yang sudah lunas (tidak ada tunggakan) ke table master data tunda cabut
	Source (Sumber)	jobs oracle
	Input	Data fastel lunas
	Output	Data fastel lunas
	Destination (Tujuan)	jobs oracle

	Logika Proses	Begin
		If exist(data fastel lunas) then
		Update status lunas
		End

No	**Proses**	**Keterangan**
5	No. Proses	4.2
	Nama Proses	Update status fastel lunas sebagian
	Deskripsi	Proses update status fastel yang sudah lunas sebagian (masih ada sisa tunggakan) ke table master data tunda cabut
	Source (Sumber)	jobs oracle
	Input	Data fastel lunas sebagian
	Output	Data fastel lunas sebagian
	Destination (Tujuan)	jobs oracle

	Logika Proses	Begin
		If exist(data fastel lunas sebagian) then
		Update status lunas sebagian
		End

No	**Proses**	**Keterangan**
6	No. Proses	4.3
	Nama Proses	Update status fastel cabut
	Deskripsi	Proses update status fastel yang sudah cabut ke table master data tunda cabut
	Source (Sumber)	jobs oracle
	Input	Data fastel cabut

	Output	Data fastel cabut
	Destination (Tujuan)	jobs oracle

	Logika Proses	Begin
		If exist(data fastel cabut) then
		Update status cabut
		End

No	**Proses**	**Keterangan**
7	No. Proses	5
	Nama Proses	Create view summary data tunda cabut
	Deskripsi	Proses menampilkan view data summary pelanggan retensi tunda cabut ke dalam aplikasi
	Source (Sumber)	pengguna
	Input	Data parameter report
	Output	Report summary data
	Destination (Tujuan)	pengguna

	Logika Proses	Begin
		If (data parameter report) then
		View data summary pelanggan tunda cabut
		End

No	**Proses**	**Keterangan**
8	No. Proses	6
	Nama Proses	Create view detail data tunda cabut
	Deskripsi	Proses menampilkan view data detail pelanggan retensi tunda cabut ke dalam aplikasi
	Source (Sumber)	pengguna
	Input	Data parameter report
	Output	Report detail data
	Destination (Tujuan)	pengguna

	Logika Proses	Begin
		If (data parameter) then
		View data detail pelanggan tunda cabut
		End

3.2.1.6 Analisis Kebutuhan Non Fungsional

Analisis non fungsional dalam pembangunan aplikasi perangkat lunak sistem informasi Dana Sumbangan Pendidikan dilakukan untuk menghasilkan spesifikasi kebutuhan non fungsional. Spesifikasi kebutuhan non fungsional adalah spesifikasi yang rinci tentang hal-hal yang akan dilakukan sistem ketika diimplementasikan.

3.2.1.6.1 Analisis perangkat keras

Perangkat keras adalah seluruh komponen atau unsur peralatan yang digunakan untuk menunjang pembangunan sistem informasi. Adapun perangkat keras yang saat ini disediakan Telkom untuk server aplikasi ini yaitu:

a. Processor DualCore Intel Xeon 5130, 2000 MHz (6 x 333)
b. Memory berkapasitas 5120 MB
c. VGA Matrox Graphics G200e
d. Harddisk berkapasitas 360 GB
e. CD-Rom berkecepatan 52x
f. LAN *card onboard*
g. Monitor SVGA ukuran 15 inci dengan resolusi maksimal 1024 x 768
h. *Keyboard* dan *Mouse*

Perangkat keras yang dibutuhkan untuk menjalankan aplikasi ini minimal memenuhi spesifikasi komputer berikut ini :

a. Processor dengan kecepatan 1,8 GHz atau lebih
b. Hard Disk 20 GB.
c. RAM 128 MB atau 256 MB.
d. Perangkat masukan standar seperti *keyboard* dan *mouse.*
e. Perangkat keluaran standar seperti monitor dengan resolusi minimal 1024 x 768 dan *printer*.
f. LAN *card onboard.*
g. *Hub* atau *switch* untuk menghubungkan komputer-komputer.
h. Kabel UTP dan RG45.
i. CD-ROM Drive 48X.

3.2.1.6.2 Analisis perangkat lunak

Perangkat lunak yang digunakan untuk mengolah data retensi tunda cabut ini masih menggunakan perangkat lunak yang umum seperti Microsoft Access dan Microsoft Excel. Dengan penggunaan perangkat lunak tersebut pihak telkom di unit operasional masih dihadapkan dengan permasalahan dalam proses dan waktu pengolahan data.

Sedangkan spesifikasi perangkat lunak yang digunakan dalam membangun aplikasi monitoring data retensi tunda cabut ini adalah sebagai berikut :

1. Untuk pembangunan aplikasi menggunakan bahasa pemrograman PHP
2. Untuk pengolahan dan penyimpanan data menggunakan database Oracle 9i
3. Untuk pemodelan DFD dan ERD menggunakan software Sybase PowerDesigner versi 15.3.0.3248

3.2.2 Perancangan Sistem

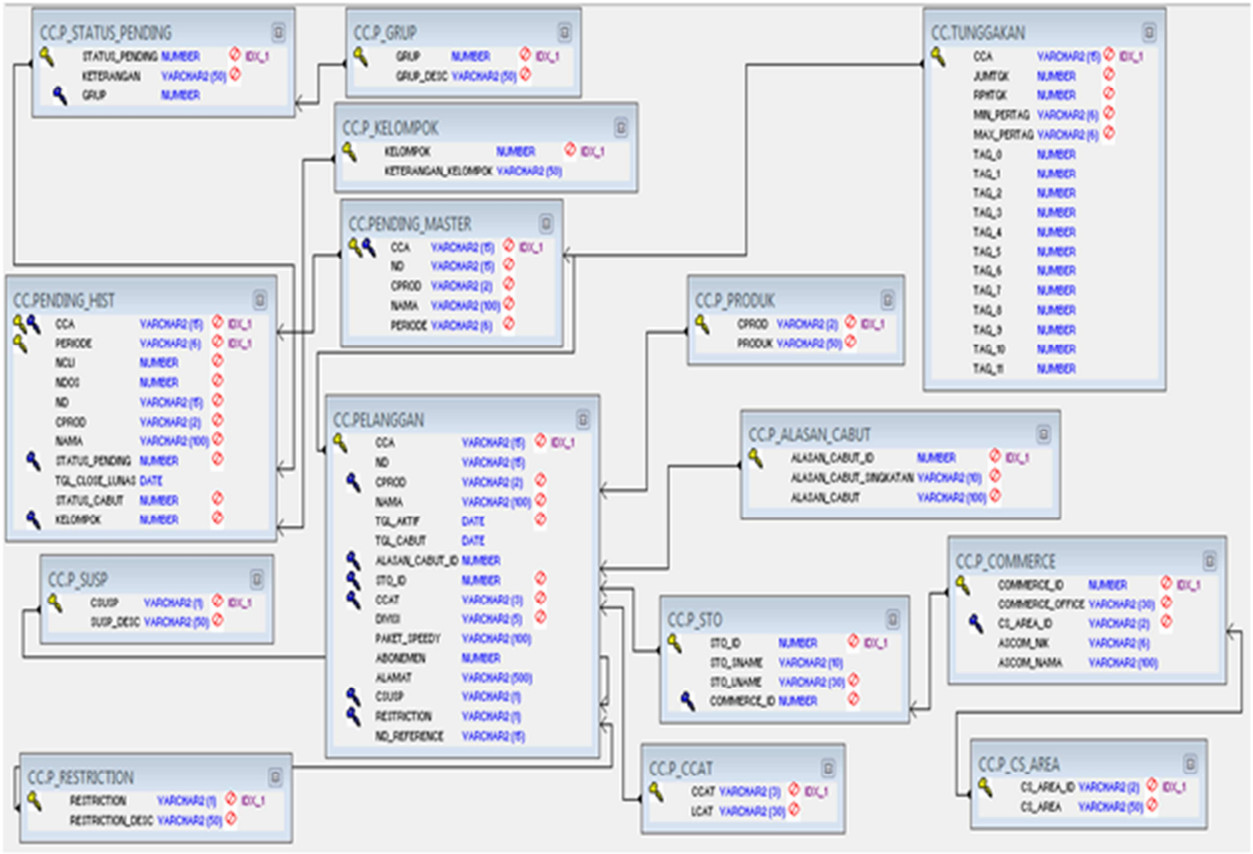
Pada perancangan sistem dilakukan beberapa macam perancangan, yaitu perancangan basis data, perancangan antar muka dan perancangan prosedural.

3.2.2.1 Perancangan Basis Data

Pada perancangan basis data ini dilakukan perancangan skema relasi dan perancangan struktur tabel.

3.2.2.1.1 Skema Relasi

Skema relasi pada aplikasi monitoring data retensi tunda cabut ini di-*generate* dari aplikasi TOAD for Oracle versi 9. Skema relasi yang dimaksud dapat dilihat pada gambar berikut ini :



Gambar 3.5 Skema Relasi Aplikasi monitoring data retensi tunda cabut

3.2.2.1.2 Struktur Tabel

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam sistem yang akan dibangun adalah sebagai berikut:

1. Tabel PENDING_MASTER

   Digunakan untuk menyimpan data master pelanggan tunda cabut, data unik per fastel atau CCA.

Table 3.1 Struktur tabel PENDING_MASTER

Column Name	ID	Pk	Null?	Data Type	Keterangan
CCA	1	1	N	VARCHAR2 (15 Byte)	primary key
ND	2		N	VARCHAR2 (15 Byte)	no telepon atau no speedy

CPROD	3		N	VARCHAR2 (2 Byte)	kode produk
NAMA	4		N	VARCHAR2 (100 Byte)	
PERIODE	5		N	VARCHAR2 (6 Byte)	bulan terbaru mulai retensi tunda cabut

2. Tabel PENDING_HIST

Digunakan untuk menyimpan data history pelanggan yang pernah mengalami tunda cabut, data unik per CCA dan PERIODE.

Table 3.2 Struktur tabel PENDING_HIST

Column Name	ID	Pk	Null?	Data Type	Keterangan
CCA	1	2	N	VARCHAR2 (15 Byte)	primary key
PERIODE	2	1	N	VARCHAR2 (6 Byte)	primary key
NCLI	3		N	NUMBER	nomor pelanggan
NDOS	4		N	NUMBER	nomor fastel
ND	5		N	VARCHAR2 (15 Byte)	
CPROD	6		N	VARCHAR2 (2 Byte)	
NAMA	7		N	VARCHAR2 (100 Byte)	
STATUS_PENDING	8		N	INTEGER	kode status pending
TGL_CLOSE_LUNAS	9		Y	DATE	tanggal pelunasan
STATUS_CABUT	10		N	INTEGER	cabut atau masih aktif
KELOMPOK	11		N	INTEGER	kode kelompok data

3. Tabel PELANGGAN

Digunakan untuk menyimpan master data fastel pelanggan data unik per CCA.

Table 3.3 Struktur tabel PELANGGAN

Column Name	ID	Pk	Null?	Data Type	Keterangan
CCA	1	1	N	VARCHAR2 (15 Byte)	primary key
ND	2		Y	VARCHAR2 (15 Byte)	
CPROD	3		N	VARCHAR2 (2 Byte)	
NAMA	4		N	VARCHAR2 (100 Byte)	
TGL_AKTIF	5		N	DATE	
TGL_CABUT	6		Y	DATE	
ALASAN_CABUT_ID	7		Y	NUMBER	kode alasan cabut
STO_ID	8		N	NUMBER	kode STO atau sentral
CCAT	9		N	VARCHAR2 (3 Byte)	kode kategori pelanggan
DIVISI	10		N	VARCHAR2 (5 Byte)	
PAKET_SPEEDY	11		Y	VARCHAR2 (100 Byte)	
ABONEMEN	12		Y	NUMBER	nilai abonemen fastel

ALAMAT	13		Y	VARCHAR2 (500 Byte)	
CSUSP	14		Y	VARCHAR2 (1 Byte)	kode pengisoliran
RESTRICTION	15		Y	VARCHAR2 (1 Byte)	kode pemblokiran
ND_REFERENCE	16		Y	VARCHAR2 (15 Byte)	no telepon speedy

4. Tabel P_ALASAN_CABUT

Digunakan untuk menyimpan data parameter alasan cabut.

Table 3.4 Struktur tabel P_ALASAN_CABUT

Column Name	ID	Pk	Null?	Data Type	Keterangan
ALASAN_CABUT_ID	1	1	N	NUMBER	primary key
ALASAN_CABUT_SINGKATAN	2		N	VARCHAR2 (10 Byte)	
ALASAN_CABUT	3		N	VARCHAR2 (100 Byte)	

5. Tabel P_CCAT

Digunakan untuk menyimpan data parameter kategori pelanggan.

Table 3.5 Struktur tabel P_CCAT

Column Name	ID	Pk	Null?	Data Type	Keterangan
CCAT	1	1	N	VARCHAR2 (3 Byte)	primary key
LCAT	2		N	VARCHAR2 (30 Byte)	kategori pelanggan

6. Tabel P_RESTRICTION

Digunakan untuk menyimpan data parameter status pemblokiran fastel.

Table 3.6 Struktur tabel P_RESTRICTION

Column Name	ID	Pk	Null?	Data Type	Keterangan
RESTRICTION	1	1	N	VARCHAR2 (1 Byte)	primary key
RESTRICTION_DESC	2		N	VARCHAR2 (50 Byte)	

7. Tabel P_SUSP

Digunakan untuk menyimpan data parameter status pengisoliran fastel.

Table 3.7 Struktur tabel P_SUSP

Column Name	ID	Pk	Null?	Data Type	Keterangan
CSUSP	1	1	N	VARCHAR2 (1 Byte)	primary key
SUSP_DESC	2		N	VARCHAR2 (50 Byte)	

8. Tabel P_KELOMPOK

Digunakan untuk menyimpan data parameter kelompok pelanggan (termasuk data potensi cabut atau termasuk data tunda cabut).

Table 3.8 Struktur tabel P_KELOMPOK

Column Name	ID	Pk	Null?	Data Type	Keterangan
KELOMPOK	1	1	N	NUMBER	primary key
KETERANGAN_KELOMPOK	2		N	VARCHAR2 (50 Byte)	

9. Tabel P_STATUS_PENDING

Digunakan untuk menyimpan data parameter status pending atau tunda cabut.

Table 3.9 Struktur tabel P_STATUS_PENDING

Column Name	ID	Pk	Null?	Data Type	Keterangan
STATUS_PENDING	1	1	N	INTEGER	primary key
KETERANGAN	2		N	VARCHAR2 (50 Byte)	
GRUP	3		Y	NUMBER	

10. Tabel P_GRUP

Digunakan untuk menyimpan data parameter grup pengelompokan status pending cabut yang sudah *closed* (selesai) atau yang belum *closed* (belum selesai).

Table 3.10 Struktur tabel P_GRUP

Column Name	ID	Pk	Null?	Data Type	Keterangan
GRUP	1	1	N	NUMBER	primary key
GRUP_DESC	2		N	VARCHAR2 (50 Byte)	

11. Tabel P_STO

Digunakan untuk menyimpan data parameter STO atau sentral di telkom.

Table 3.11 Struktur tabel P_STO

Column Name	ID	Pk	Null?	Data Type	Keterangan
STO_ID	1	1	N	NUMBER	primary key
STO_SNAME	2		Y	VARCHAR2 (10 Byte)	
STO_LNAME	3		N	VARCHAR2 (30 Byte)	
COMMERCE_ID	4		N	NUMBER	

12. Tabel P_COMMERCE

Digunakan untuk menyimpan data parameter Commerce Office Telkom.

Table 3.12 Struktur tabel P_COMMERCE

Column Name	ID	Pk	Null?	Data Type	Keterangan
COMMERCE_ID	1	1	N	NUMBER	primary key
COMMERCE_OFFICE	2		N	VARCHAR2 (30 Byte)	
CS_AREA_ID	3		N	VARCHAR2 (2 Byte)	
ASCOM_NIK	4		Y	VARCHAR2 (6 Byte)	NIK asman commerce
ASCOM_NAMA	5		Y	VARCHAR2 (100 Byte)	nama asman commerce

13. Tabel P_CS_AREA

Digunakan untuk menyimpan data parameter CS (*Consumer Services*) Area di telkom.

Table 3.13 Struktur tabel P_CS_AREA

Column Name	ID	Pk	Null?	Data Type	Keterangan
CS_AREA_ID	1	1	N	VARCHAR2 (2 Byte)	primary key
CS_AREA	2		N	VARCHAR2 (50 Byte)	

14. Tabel P_PRODUK

Digunakan untuk menyimpan data parameter kode produk fastel telkom.

Table 3.14 Struktur tabel P_PRODUK

Column Name	ID	Pk	Null?	Data Type	Keterangan
CPROD	1	1	N	VARCHAR2 (2 Byte)	primary key
PRODUK	2		N	VARCHAR2 (50 Byte)	

15. Tabel TUNGGAKAN

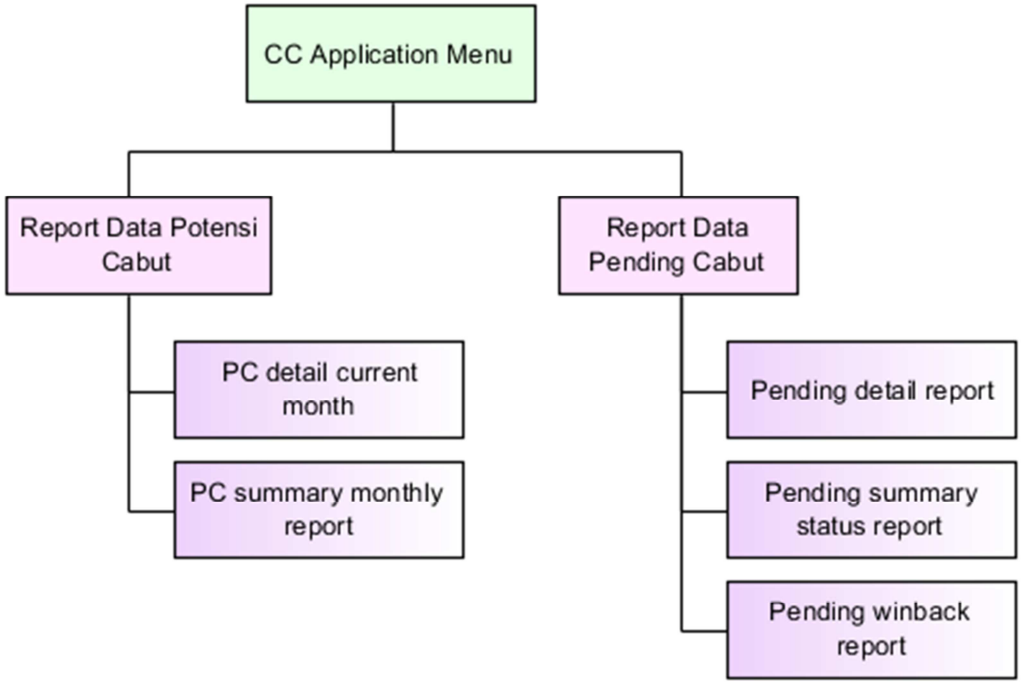
Digunakan untuk menyimpan data tunggakan fastel pelanggan.

Table 3.15 Struktur tabel TUNGGAKAN

Column Name	ID	Pk	Null?	Data Type	Keterangan
CCA	1	1	N	VARCHAR2 (15 Byte)	primary key
JUMTGK	2		N	NUMBER	jumlah bulan tunggakan
RPHTGK	3		N	NUMBER	jumlah total rupiah tunggakan
MIN_PERTAG	4		N	VARCHAR2 (6 Byte)	bulan tagihan tertua
MAX_PERTAG	5		N	VARCHAR2 (6 Byte)	bulan tagihan termuda
TAG_0	6		Y	NUMBER	tagihan bulan ke N
TAG_1	7		Y	NUMBER	tagihan bulan ke N-1
TAG_2	8		Y	NUMBER	tagihan bulan ke N-2
TAG_3	9		Y	NUMBER	tagihan bulan ke N-3
TAG_4	10		Y	NUMBER	tagihan bulan ke N-4
TAG_5	11		Y	NUMBER	tagihan bulan ke N-5
TAG_6	12		Y	NUMBER	tagihan bulan ke N-6
TAG_7	13		Y	NUMBER	tagihan bulan ke N-7
TAG_8	14		Y	NUMBER	tagihan bulan ke N-8
TAG_9	15		Y	NUMBER	tagihan bulan ke N-9
TAG_10	16		Y	NUMBER	tagihan bulan ke N-10
TAG_11	17		Y	NUMBER	tagihan bulan ke N-12

3.2.2.2 Perancangan Struktur Menu

Perancangan struktur menu berisikan menu dan submenu yang berfungsi memudahkan user didalam menggunakan sistem. Berikut ini adalah gambaran mengenai struktur menu pada aplikasi monitoring data retensi tunda cabut ini.



Gambar 3.6 Struktur Menu Aplikasi monitoring data retensi tunda cabut

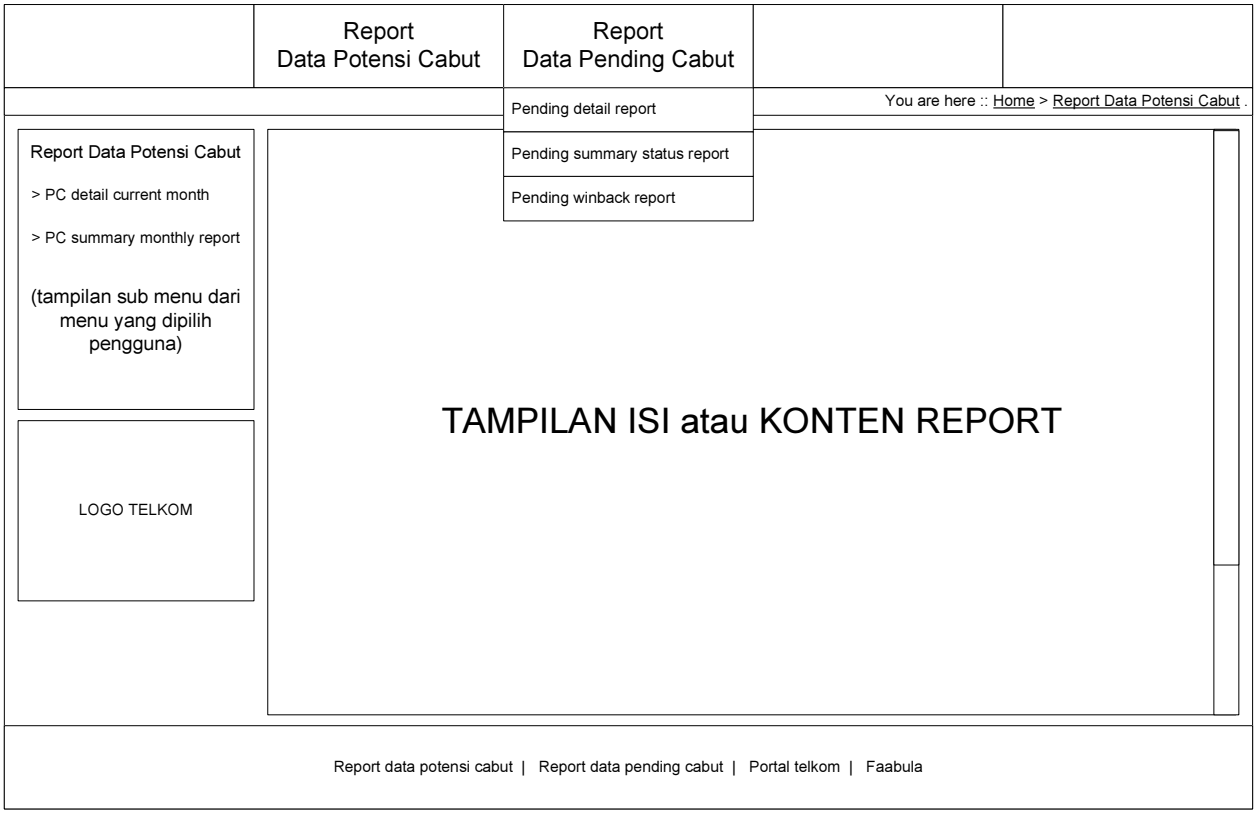
Menu *Report Data Potensi Cabut (PC)* berisi data fastel pelanggan yang merupakan data yang berpotensi menjadi cabutan fastel pelanggan. Submenu *PC detail current month* hanya berisi data detail fastel pelanggan yang statusnya masih menunggak dan berpotensi menjadi cabutan fastel pelanggan. Submenu *PC summary monthly report* berisi laporan data rekap jumlah fastel pelanggan yang pernah menjadi data potensi cabut dari bulan ke bulan. Menu *Report Data Pending Cabut* (pending) berisi data fastel pelanggan yang sudah masuk ke dalam masa retensi tunda cabut.

3.2.2.3 Perancangan Antar Muka

Perancangan antar muka bertujuan untuk memberikan gambaran khususnya tentang tampilan aplikasi yang akan dibangun. Sehingga akan mempermudah dalam mengimplementasikan aplikasi. Dan juga akan memudahkan pembuatan aplikasi yang *user friendly*.

1. **Tampilan Web Template**

   Tampilan web yang akan dibuat menggunakan satu halaman template dengan menggunakan satu *frame* yang isi atau kontennya tergantung pada menu yang dipilih oleh pengguna aplikasi. Tampilan web yang akan dibuat dapat dilihat pada gambar di bawah ini :

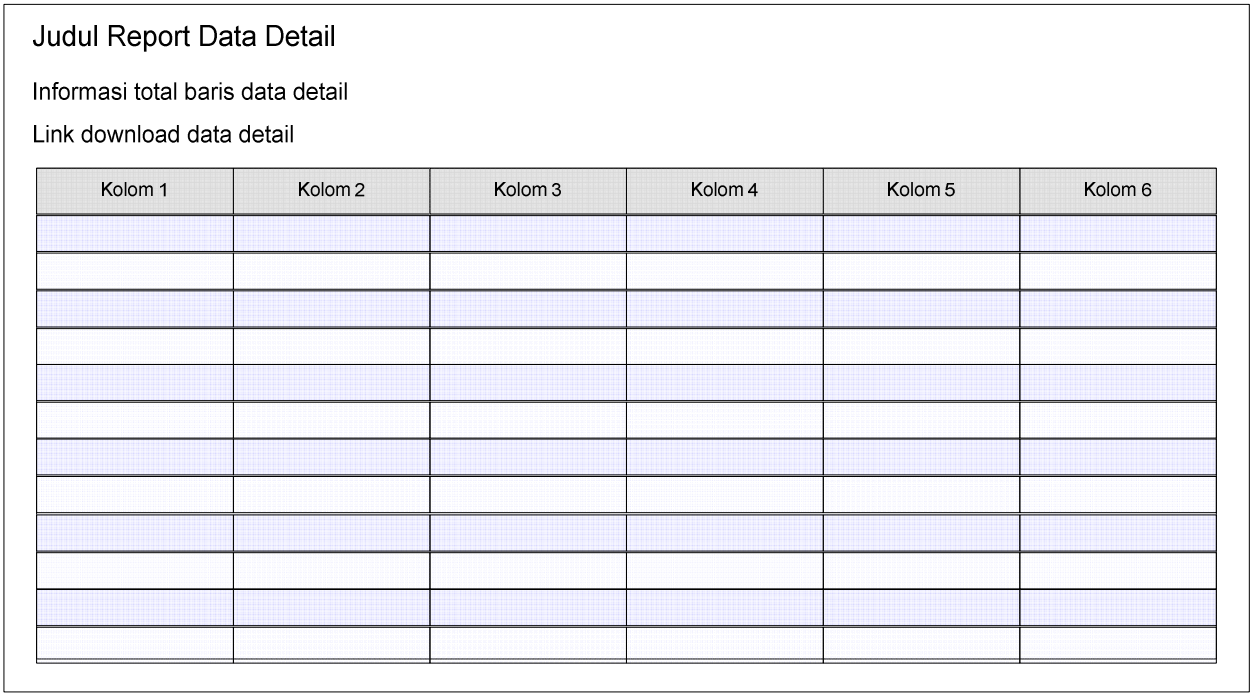


Gambar 3.7 Tampilan web template

2. **Tampilan Konten/Isi Report Data Detail Fastel**

   Tampilan konten atau isi report data detail memiliki template halaman yang sama. Yang membedakan adalah isi halamannya yang tergantung pada menu

yang dipilih oleh pengguna. Secara umum pengguna akan melihat preview data (hanya beberapa baris data sebagai sampling data) beserta informasi jumlah baris data dan link untuk mendownload data detail. Halaman ini akan tampil pada frame pada halaman utama. Tampilan halaman yang akan dibuat dapat dilihat pada gambar di bawah ini :



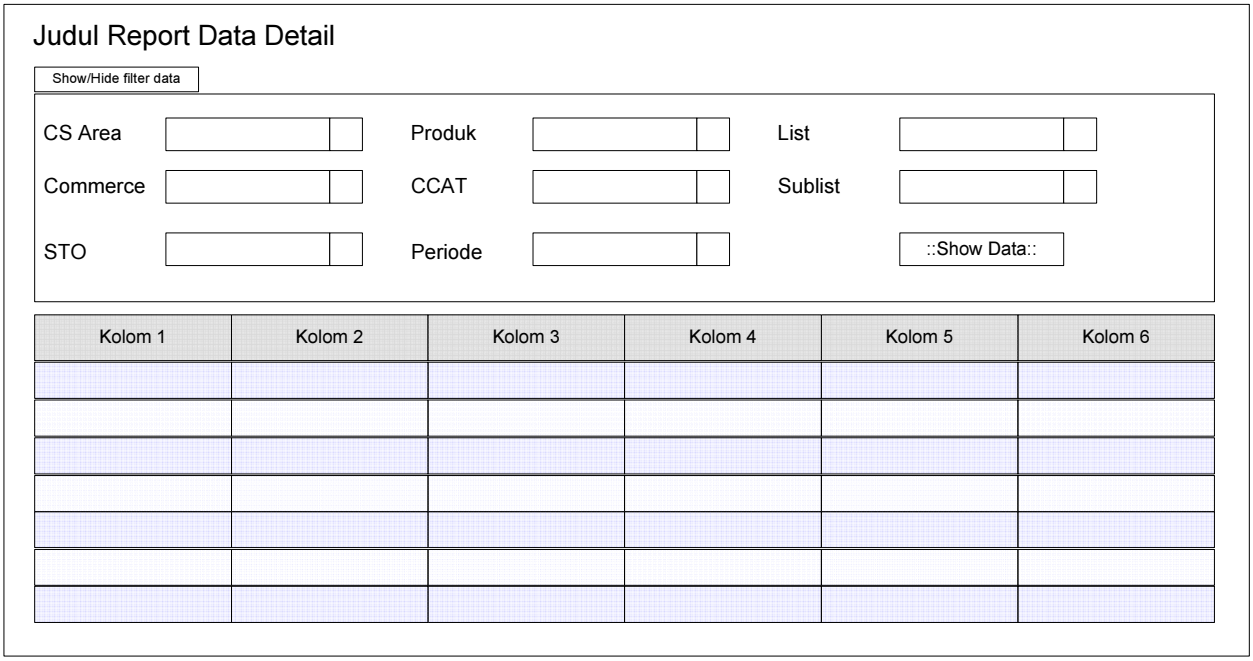
Gambar 3.8 Tampilan halaman report detail fastel

3. Tampilan Konten/Isi Report Data Summary

Tampilan konten atau isi report data summary memiliki template halaman yang sama. Yang membedakan adalah isi halamannya yang tergantung pada menu yang dipilih oleh pengguna. Secara umum pengguna akan memilih parameter data yang ada. Lalu pengguna dapat mendefinisikan sendiri data apa

yang akan ditampilkan. Lalu pengguna akan melihat data report summary berdasarkan konfigurasi parameter yang dipilih setelah menekan tombol *show data*.

Halaman ini akan tampil pada frame pada halaman utama. Tampilan halaman yang akan dibuat dapat dilihat pada gambar di bawah ini :



Gambar 3.9 Tampilan halaman report summary fastel

3.3 Implementasi

Dalam tahap implementasi akan dideskripsikan penggunaan bahasa pemrograman. Sistem informasi ini berbasis web dan dibangun dengan bahasa pemrograman PHP dengan menggunakan database Oracle.

3.3.1 Implementasi Program

Berikut ini adalah deskripsi dari file-file hasil pengkodean dengan menggunakan bahasa pemrograman PHP :

Tabel 3.16 Daftar file hasil pengkodean

No	Nama File	Deskripsi
1	Index.php	File halaman utama yang pertama diakses, merupakan template dari halaman, menggunakan frame yang isinya adalah report sesuai dengan menu yang dipilih oleh user
2	conn.php	File untuk koneksi ke database aplikasi ini
3	select.php	Script untuk melakukan select data dari sql query ke database
4	forms.php	Berisi fungsi dan prosedur yang berhubungan dengan tampilan form pada aplikasi
5	getdata.php	Berisi fungsi dan prosedur yang berfungsi untuk mengambil data-data parameter aplikasi ini
6	menu.php	Menampilkan daftar menu dan submenu pada kotak menu sebelah kiri frame isi atau konten report
7	File-file php dalam folder *pages*	File-file ini digunakan untuk menampilkan report-report yang ada dalam aplikasi ini

3.3.2 Implementasi Data

Database yang dipakai dalam aplikasi ini adalah database Oracle 9i. Tabel-tabel yang digunakan untuk menyimpan data dibuat dengan menggunakan SQL yang disebut DDL (*Data Definition Language*). Beberapa *procedure* dibuat untuk melakukan update data status fastel pelanggan retensi tunda cabut dan untuk melakukan update data pelanggan dan data tunggakan. *Procedure* tersebut dijalankan melalui jobs oracle dalam frekuensi tertentu.

Adapun deskripsi table yang dibuat dan query yang dijalankan adalah sebagai berikut:

```
CREATE TABLE P_GRUP
(
 GRUP      NUMBER,
 GRUP_DESC  VARCHAR2(50 BYTE)              NOT NULL,
 CONSTRAINT P_GRUP_PK
PRIMARY KEY
(GRUP)
)
/
```

```
CREATE TABLE P_ALASAN_CABUT
(
 ALASAN_CABUT_ID       NUMBER,
 ALASAN_CABUT_SINGKATAN  VARCHAR2(10 BYTE)    NOT NULL,
 ALASAN_CABUT          VARCHAR2(100 BYTE)    NOT NULL,
 CONSTRAINT P_ALASAN_CABUT_PK
PRIMARY KEY
(ALASAN_CABUT_ID)
)
/
```

```
CREATE TABLE P_CCAT
(
 CCAT  VARCHAR2(3 BYTE)                  NOT NULL,
 LCAT  VARCHAR2(30 BYTE)                 NOT NULL,
 CONSTRAINT P_CCAT_PK
PRIMARY KEY
(CCAT)
)
/
```

```
CREATE TABLE P_CS_AREA
(
 CS_AREA_ID  VARCHAR2(2 BYTE)              NOT NULL,
 CS_AREA     VARCHAR2(50 BYTE)             NOT NULL,
 CONSTRAINT P_CS_AREA_PK
PRIMARY KEY
(CS_AREA_ID)
)
/
```

```
CREATE TABLE P_COMMERCE
(
 COMMERCE_ID      NUMBER                   NOT NULL,
 COMMERCE_OFFICE  VARCHAR2(30 BYTE)        NOT NULL,
 CS_AREA_ID       VARCHAR2(2 BYTE)         NOT NULL,
 ASCOM_NIK        VARCHAR2(6 BYTE),
 ASCOM_NAMA       VARCHAR2(100 BYTE),
 CONSTRAINT P_COMMERCE_PK
PRIMARY KEY
(COMMERCE_ID),
 CONSTRAINT P_COMMERCE_CON
FOREIGN KEY (CS_AREA_ID)
REFERENCES P_CS_AREA (CS_AREA_ID)
)
/
```

```
CREATE TABLE P_STO
(
 STO_ID       NUMBER                       NOT NULL,
 STO_SNAME    VARCHAR2(10 BYTE),
 STO_LNAME    VARCHAR2(30 BYTE)            NOT NULL,
 COMMERCE_ID  NUMBER                       NOT NULL,
 CONSTRAINT P_STO_PK
PRIMARY KEY
(STO_ID),
 CONSTRAINT P_STO_CON
FOREIGN KEY (COMMERCE_ID)
REFERENCES P_COMMERCE (COMMERCE_ID)
)
/
```

```
CREATE TABLE P_KELOMPOK
(
 KELOMPOK           NUMBER,
 KETERANGAN_KELOMPOK  VARCHAR2(50 BYTE)       NOT NULL,
 CONSTRAINT P_KELOMPOK_PK
PRIMARY KEY
(KELOMPOK)
)
/
```

```
CREATE TABLE P_STATUS_PENDING
(
 STATUS_PENDING  INTEGER                  NOT NULL,
 KETERANGAN      VARCHAR2(50 BYTE)          NOT NULL,
 GRUP          NUMBER,
 CONSTRAINT P_STATUS_NAPR_PK
PRIMARY KEY
(STATUS_PENDING),
 CONSTRAINT P_STATUS_PENDING_FK
FOREIGN KEY (GRUP)
REFERENCES P_GRUP (GRUP)
)
/
```

```
CREATE TABLE TUNGGAKAN
(
 CCA        VARCHAR2(15 BYTE),
 JUMTGK     NUMBER                      NOT NULL,
 RPHTGK     NUMBER                      NOT NULL,
 MIN_PERTAG  VARCHAR2(6 BYTE)              NOT NULL,
 MAX_PERTAG  VARCHAR2(6 BYTE)              NOT NULL,
 TAG_0      NUMBER,
 TAG_1      NUMBER,
 TAG_2      NUMBER,
 TAG_3      NUMBER,
 TAG_4      NUMBER,
 TAG_5      NUMBER,
 TAG_6      NUMBER,
 TAG_7      NUMBER,
 TAG_8      NUMBER,
 TAG_9      NUMBER,
 TAG_10     NUMBER,
 TAG_11     NUMBER,
 CONSTRAINT TUNGGAKAN_PK
 PRIMARY KEY
 (CCA)
)
/
```

```
CREATE TABLE P_PRODUK
(
 CPROD   VARCHAR2(2 BYTE),
 PRODUK  VARCHAR2(50 BYTE)                 NOT NULL,
 CONSTRAINT P_PRODUK_PK
 PRIMARY KEY
 (CPROD)
)
/
```

```
CREATE TABLE P_SUSP
(
 CSUSP      VARCHAR2(1 BYTE),
 SUSP_DESC  VARCHAR2(50 BYTE)              NOT NULL,
 CONSTRAINT P_SUSP_PK
PRIMARY KEY
(CSUSP)
)
/
```

```
CREATE TABLE P_RESTRICTION
(
 RESTRICTION      VARCHAR2(1 BYTE),
 RESTRICTION_DESC  VARCHAR2(50 BYTE)          NOT NULL,
 CONSTRAINT P_RESTRICTION_PK
PRIMARY KEY
(RESTRICTION)
)
/
```

```
CREATE TABLE PELANGGAN
(
 CCA           VARCHAR2(15 BYTE)          NOT NULL,
 ND            VARCHAR2(15 BYTE),
 CPROD         VARCHAR2(2 BYTE)           NOT NULL,
 NAMA          VARCHAR2(100 BYTE)         NOT NULL,
 TGL_AKTIF     DATE                       NOT NULL,
 TGL_CABUT     DATE,
 ALASAN_CABUT_ID  NUMBER,
 STO_ID        NUMBER                     NOT NULL,
 CCAT          VARCHAR2(3 BYTE)           NOT NULL,
 DIVISI        VARCHAR2(5 BYTE)           NOT NULL,
 PAKET_SPEEDY    VARCHAR2(100 BYTE),
 ABONEMEN      NUMBER,
 ALAMAT        VARCHAR2(500 BYTE),
 CSUSP         VARCHAR2(1 BYTE),
 RESTRICTION   VARCHAR2(1 BYTE),
 ND_REFERENCE    VARCHAR2(15 BYTE),
 CONSTRAINT PELANGGAN_PK
PRIMARY KEY
(CCA),
 FOREIGN KEY (STO_ID)
REFERENCES P_STO (STO_ID)
  ON DELETE SET NULL,
 FOREIGN KEY (CCAT)
REFERENCES P_CCAT (CCAT)
  ON DELETE SET NULL,
 FOREIGN KEY (CPROD)
REFERENCES P_PRODUK (CPROD)
  ON DELETE SET NULL,
 FOREIGN KEY (ALASAN_CABUT_ID)
REFERENCES P_ALASAN_CABUT (ALASAN_CABUT_ID),
 FOREIGN KEY (CSUSP)
REFERENCES P_SUSP (CSUSP)
  ON DELETE SET NULL,
 FOREIGN KEY (RESTRICTION)
REFERENCES P_RESTRICTION (RESTRICTION)
  ON DELETE SET NULL
)
/
```

```
CREATE TABLE PENDING_MASTER
(
 CCA     VARCHAR2(15 BYTE)              NOT NULL,
 ND      VARCHAR2(15 BYTE)              NOT NULL,
 CPROD   VARCHAR2(2 BYTE)               NOT NULL,
 NAMA    VARCHAR2(100 BYTE)             NOT NULL,
 PERIODE VARCHAR2(6 BYTE)               NOT NULL,
 CONSTRAINT NAPR_MASTER_PK
PRIMARY KEY
(CCA),
 FOREIGN KEY (CCA)
REFERENCES PELANGGAN (CCA),
 FOREIGN KEY (CCA)
REFERENCES TUNGGAKAN (CCA)
)
/
```

```
CREATE TABLE PENDING_HIST
(
 CCA           VARCHAR2(15 BYTE),
 PERIODE       VARCHAR2(6 BYTE),
 NCLI          NUMBER                  NOT NULL,
 NDOS          NUMBER                  NOT NULL,
 ND            VARCHAR2(15 BYTE)       NOT NULL,
 CPROD         VARCHAR2(2 BYTE)        NOT NULL,
 NAMA          VARCHAR2(100 BYTE)      NOT NULL,
 STATUS_PENDING   INTEGER              NOT NULL,
 TGL_CLOSE_LUNAS  DATE,
 STATUS_CABUT     INTEGER              NOT NULL,
 KELOMPOK      INTEGER                 NOT NULL,
 CONSTRAINT PENDING_HIST_PK
PRIMARY KEY
(PERIODE, CCA),
 FOREIGN KEY (CCA)
REFERENCES PENDING_MASTER (CCA),
 FOREIGN KEY (KELOMPOK)
REFERENCES P_KELOMPOK (KELOMPOK)
   ON DELETE SET NULL,
 FOREIGN KEY (STATUS_PENDING)
REFERENCES P_STATUS_PENDING (STATUS_PENDING)
   ON DELETE SET NULL
)
/
```

Adapun salah satu procedure yang dibuat di dalam database adalah sebagai berikut:

```
CREATE OR REPLACE PROCEDURE cc.update_status_cabut
IS
   ayeuna   VARCHAR (100);

   CURSOR c
   IS
      SELECT a.cca, b.periode, a.tgl_cabut
        FROM pelanggan a, pending_hist b
       WHERE a.cca = b.cca
             AND a.tgl_cabut IS NOT NULL             --AND b.status_cabut = 0
                                        ;

   x        c%ROWTYPE;
BEGIN
   SELECT TO_CHAR (SYSDATE, 'dd/mm/yyyy hh24:mi:ss')
     INTO ayeuna
     FROM DUAL;

   DBMS_OUTPUT.put_line ('mulai jam : ' || ayeuna);

   OPEN c;

   LOOP
      FETCH c
       INTO x;

      UPDATE pending_hist
         SET status_cabut = 1,                                          --cabut
             status_pending = 5                                         --cabut
       WHERE cca = x.cca
         AND periode = x.periode
         AND status_cabut = 0
         AND status_pending IN (2, 3);               --pending, lunas sebagian

      UPDATE pending_hist
         SET status_cabut = 1,                                          --cabut
             status_pending = 14                              --lunas dan cabut
       WHERE cca = x.cca AND periode = x.periode AND status_pending = 4;

      --lunas
      COMMIT;
      EXIT WHEN c%NOTFOUND;
   END LOOP;

   CLOSE c;

   COMMIT;

   SELECT TO_CHAR (SYSDATE, 'dd/mm/yyyy hh24:mi:ss')
     INTO ayeuna
     FROM DUAL;

   DBMS_OUTPUT.put_line ('selesai jam : ' || ayeuna);
END update_status_cabut;
/
```