



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODE PENELITIAN

3.1 Metode Penelitian

Dalam pembangunan aplikasi ini, digunakan metode penelitian dimulai dari dengan melakukan studi literatur, mengumpulkan data yang digunakan sebagai bahan penunjang aplikasi, kemudian melakukan perancangan dan pembangunan aplikasi dan yang terakhir melakukan pengujian dan perbaikan pada aplikasi tersebut.

3.1.1 Studi Literatur

Tahap awal dalam membangun sistem penjualan ini adalah dengan melakukan studi literatur terhadap referensi-referensi yang berhubungan dengan aplikasi penjualan dan *fuzzy time series*. Referensi–referensi tersebut dapat berupa artikel, jurnal-jurnal, buku, dan lain-lain.

3.1.2 Pengumpulan Data

Dalam membangun *forecasting* guna melakukan peramalan barang yang terdapat dalam sistem penjualan tersebut, maka dibutuhkan data transaksi penjualan selama kurang lebih enam tahun yang lalu. Dari data transaksi penjualan yang telah ada, maka peramalan untuk jenis barang yang paling banyak diminati akan dapat diramalkan sehingga dapat membantu perusahaan dalam aktivitas penjualan ke depan nanti.

3.1.3 Perancangan dan Pembangunan Aplikasi

Perancangan dari dibentuknya *user interface* guna membantu selama proses penjualan, hingga dilakukan proses perhitungan menggunakan metode *fuzzy time series* untuk menampilkan hasil peramalan dalam bentuk grafik dengan menggunakan bahasa pemrograman web.

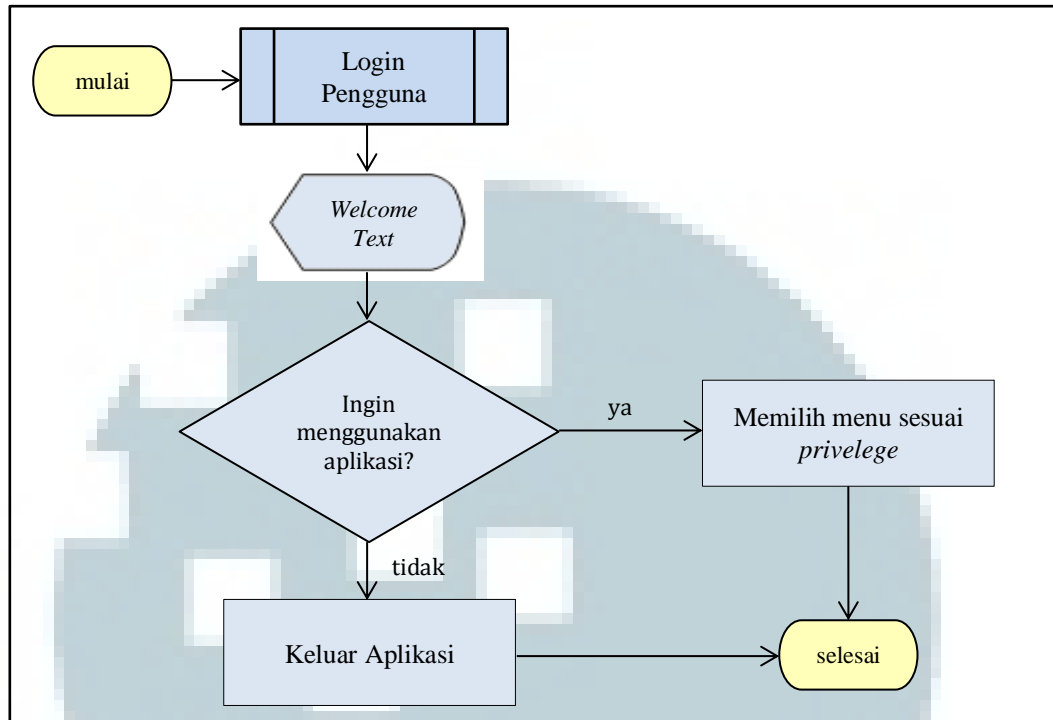
3.1.4 Pengujian dan Perbaikan

Aplikasi yang telah dibuat kemudian akan diuji coba untuk melihat apakah aplikasi tersebut dapat berjalan dengan baik atau masih ada *bug* pada aplikasi tersebut. Apabila diperlukan perbaikan, maka akan dilakukan perbaikan pada saat itu juga.

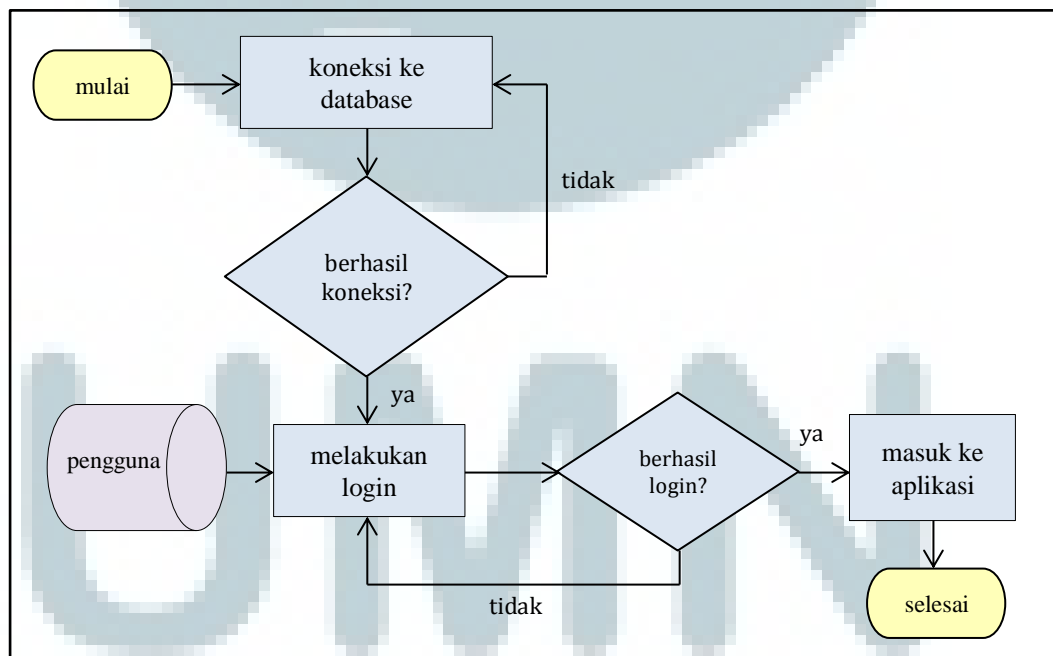
3.2 Perancangan Aplikasi

Perancangan aplikasi yang dibuat pada penelitian ini dimulai dengan *flowchart diagram* yang berguna dalam menggambarkan alur proses jalannya aplikasi ini, *data flow diagram* yang berguna dalam menggambarkan aliran data dari suatu proses ke proses yang lain dalam aplikasi ini dan *entity relationship diagram* yang bermanfaat untuk menampilkan hubungan antar tabel yang dibutuhkan dalam aplikasi ini. Aplikasi ini memiliki dua entitas eksternal yaitu *user* dan *admin*.

3.2.1 Flowchart Diagram



Gambar 3.1 Flowchart Keseluruhan Aplikasi



Gambar 3.2 Flowchart Subroutine Login Pengguna

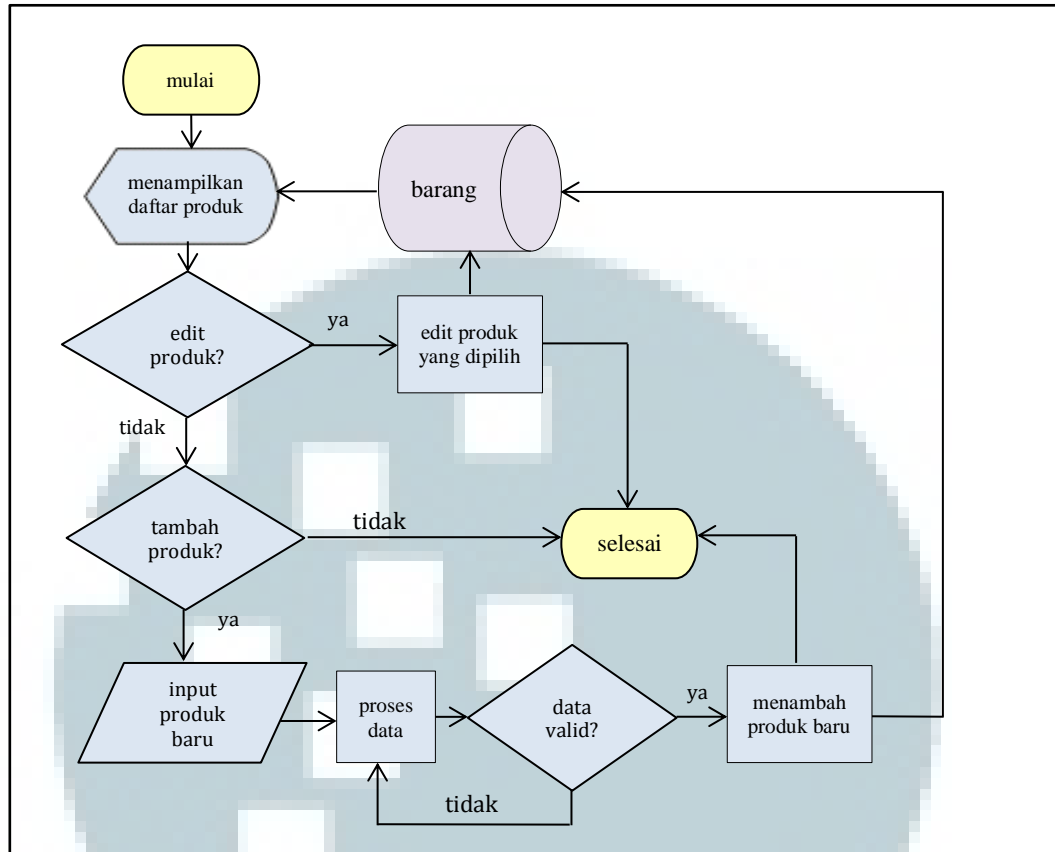
Pada *flowchart subroutine login* di atas, langkah pertama pengguna (*user/admin*) harus melakukan koneksi terlebih dahulu ke *database*. Apabila koneksi *database* berhasil, maka *user/admin* perlu melakukan *login* terlebih dahulu dengan memasukkan *username* dan *password* untuk dapat masuk ke dalam aplikasi ini.

Setelah berhasil melakukan *login*, maka *user/admin* dapat melihat daftar produk, menambah dan mengedit produk (*admin privilege*), melihat pelanggan, menambah dan mengedit pelanggan (*admin privilege*), melihat dan menambah transaksi, melihat piutang, menambah piutang dan melihat *forecasting* produk (*admin privilege*).

Di bagian kiri tampilan aplikasi, terdapat beberapa menu seperti transaksi, produk, pelanggan, pengguna (*admin privilege*), piutang, laporan, *forecasting* (*admin privilege*) dan keluar (*logout*).

Apabila *user/admin* memilih menu produk, maka tampilan akan langsung menampilkan daftar produk dengan tombol tambah produk dan *edit* produk.

U
M
N



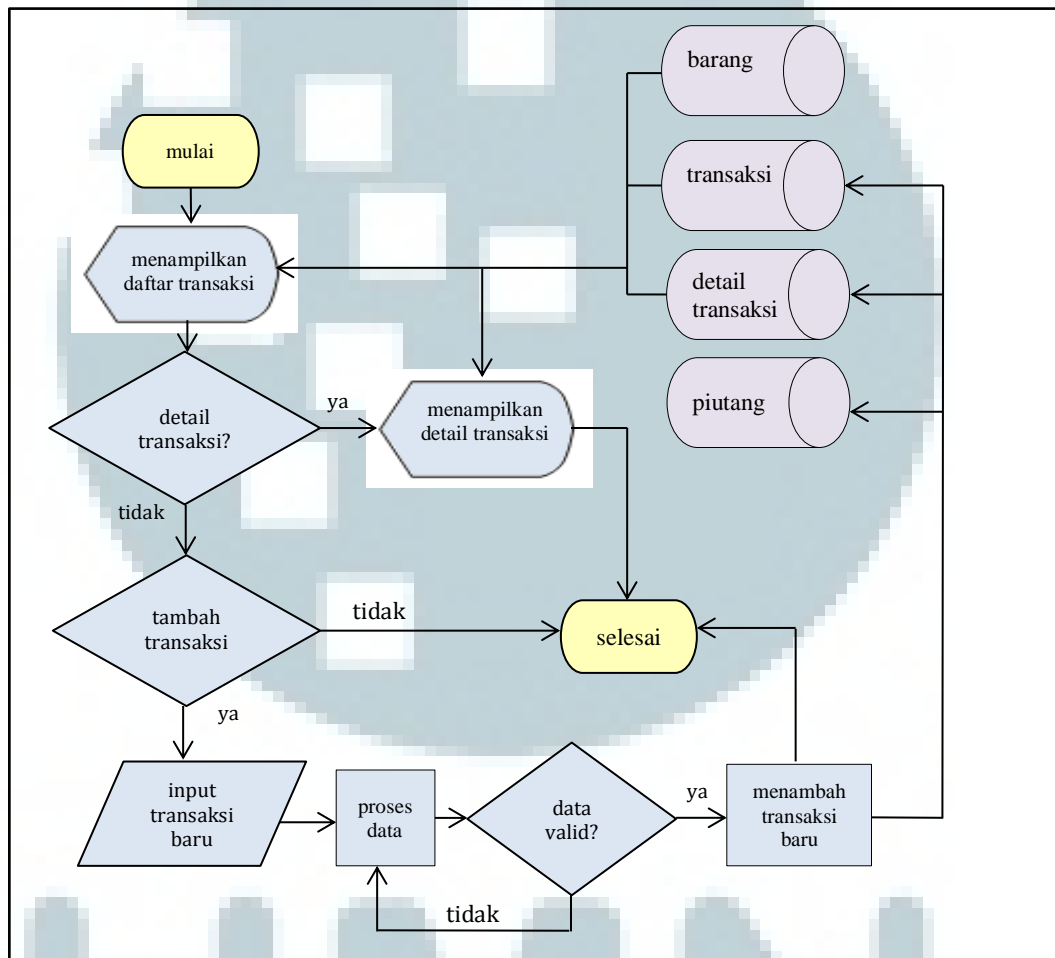
Gambar 3.3 Flowchart Menu Produk

Pada *flowchart* menu produk yang ditampilkan pada gambar 3.3, ketika *user/admin* memilih menu produk, maka aplikasi akan menampilkan halaman dengan daftar produk yang dimilikinya. Daftar produk yang tersimpan dalam *database* akan ditampilkan dalam bentuk *paging*. Apabila *admin* ingin melakukan *edit* suatu produk, maka aplikasi akan menampilkan halaman dimana *admin* dapat melakukan *edit* produk tersebut. Data yang dimasukkan kemudian akan disimpan ke dalam *database* barang.

Ketika *admin* ingin melakukan penambahan suatu produk tertentu, maka dapat dipilih tombol tambah barang dan aplikasi akan menampilkan halaman untuk penambahan produk. Data yang telah diisi oleh *admin*, kemudian akan divalidasi terlebih dahulu. Apabila belum *valid*, maka *admin* harus menambahkan

data yang diperlukan, namun apabila data tersebut sudah *valid*, maka data tersebut akan tersimpan ke dalam *database*.

Apabila *user/admin* memilih menu transaksi, maka tampilan akan langsung menampilkan daftar transaksi dengan masing-masing detail transaksi dan tombol tambah transaksi.

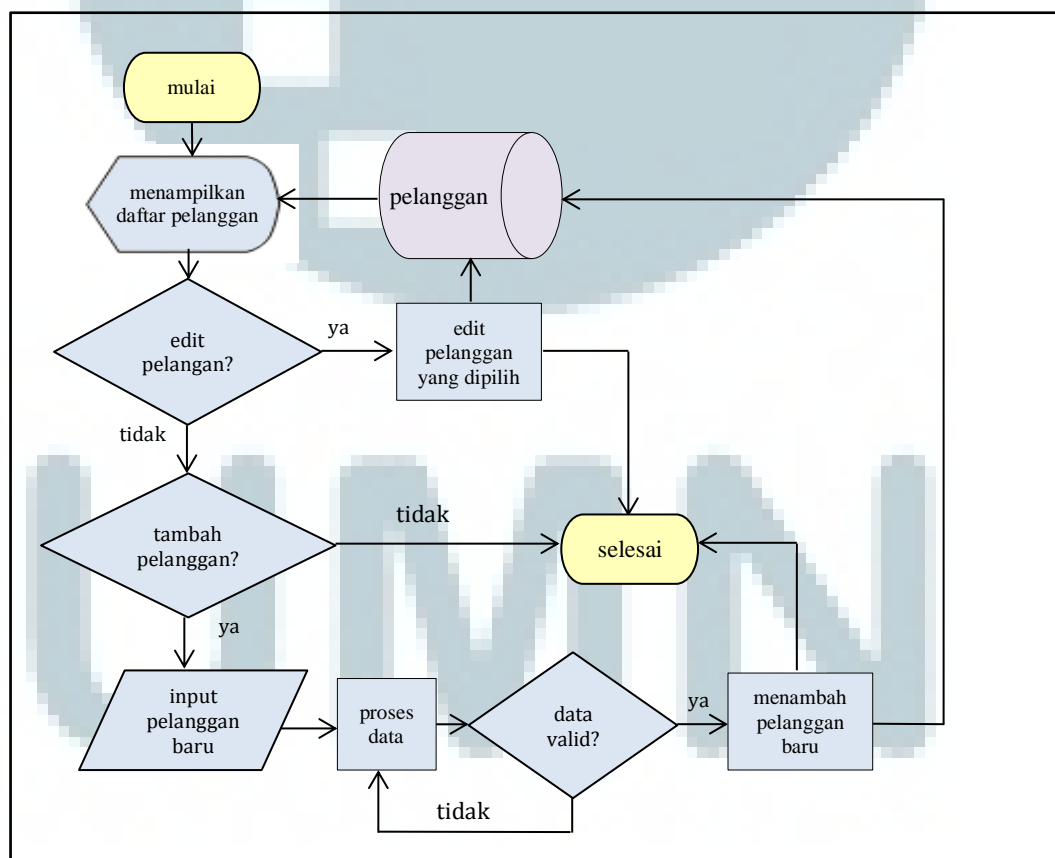


Gambar 3.4 *Flowchart* Menu Transaksi

Pada *flowchart* gambar 3.4 di atas, *user/admin* yang memilih menu transaksi akan melihat tampilan daftar transaksi yang telah dilakukan. Apabila *user/admin* ingin melihat detail dari suatu transaksi, maka dapat memilih keterangan yang berada pada masing-masing transaksi.

Apabila *user/admin* ingin melakukan penambahan transaksi, maka dapat memilih tombol tambah transaksi. *User/admin* akan memasukkan data transaksi yang kemudian sebelum disimpan ke *database*, perlu dilakukan validasi data terlebih dahulu. Apabila data yang dimasukkan sudah valid, maka aplikasi akan melakukan penyimpanan data transaksi tersebut ke *database* dan melakukan pengurangan jumlah stok pada barang/produk yang dibeli. Namun apabila data transaksi tersebut belum valid, maka data yang kurang harus diisi.

Apabila *user/admin* memilih menu pelanggan, maka tampilan akan langsung menampilkan daftar pelanggan dalam *paging*. *User/admin* dapat menambahkan data pelanggan dan juga dapat melakukan *edit* data pelanggan (*admin privilege*).



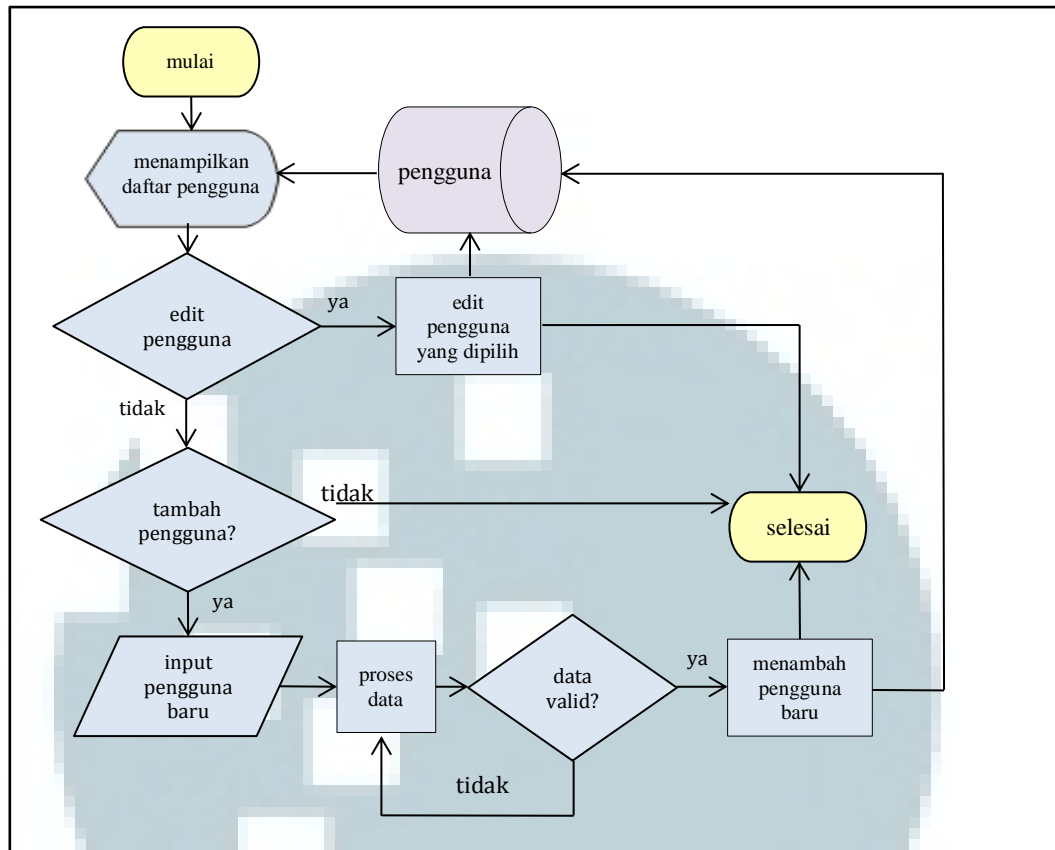
Gambar 3.5 Flowchart Menu Pelanggan

Pada *flowchart* menu pelanggan yang ditampilkan pada gambar 3.5, ketika *user/admin* memilih menu pelanggan, maka aplikasi akan menampilkan halaman dengan daftar pelanggan yang dimilikinya. Daftar pelanggan yang tersimpan dalam *database* akan ditampilkan dalam bentuk *paging*. Apabila *admin* ingin melakukan *edit* pada seorang pelanggan, maka aplikasi akan menampilkan halaman dimana *admin* dapat melakukan *edit* data pelanggan tersebut. Data yang dimasukkan kemudian akan disimpan ke dalam *database*.

Ketika *user/admin* ingin melakukan penambahan seorang pelanggan, maka dapat dipilih tombol tambah pelanggan dan aplikasi akan menampilkan halaman untuk penambahan pelanggan. Data yang telah diisi oleh *user/admin*, kemudian akan divalidasi terlebih dahulu. Apabila belum *valid*, maka *user/admin* harus menambahkan data yang diperlukan, namun apabila data tersebut sudah *valid*, maka data tersebut akan tersimpan ke dalam *database*.

Apabila *admin* memilih menu pengguna, maka tampilan akan langsung menampilkan daftar pengguna dalam *paging*. *Admin* dapat menambahkan data pengguna dan juga dapat melakukan *edit* data pengguna. Seorang pengguna dapat memiliki status sebagai seorang *admin* atau sebagai seorang *user*.

U
M
N



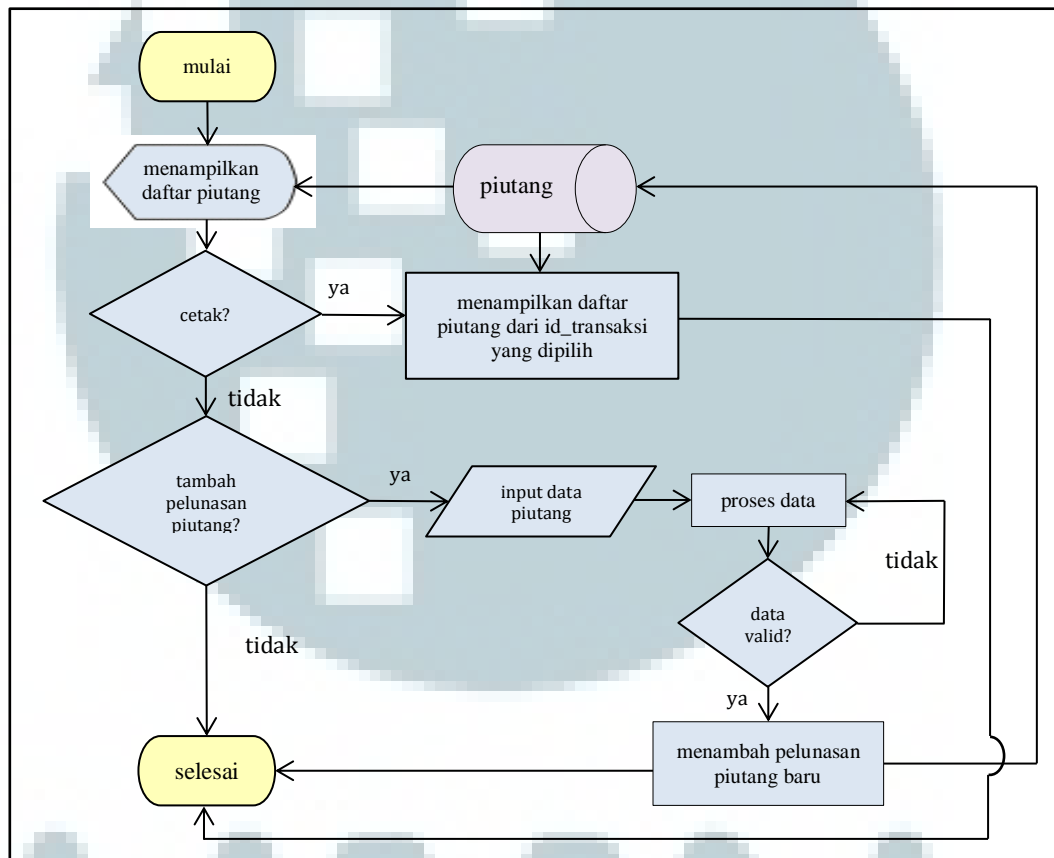
Gambar 3.6 *Flowchart* Menu Pengguna

Pada *flowchart* menu pengguna yang ditampilkan pada gambar 3.6, ketika *admin* memilih menu pengguna, maka aplikasi akan menampilkan halaman dengan daftar pengguna. Daftar pengguna yang tersimpan dalam *database* akan ditampilkan dalam bentuk *paging*. Apabila *admin* ingin melakukan *edit* pada seorang pengguna, maka aplikasi akan menampilkan halaman dimana *admin* dapat melakukan *edit* data pengguna tersebut. Data yang dimasukkan kemudian akan disimpan ke dalam *database*.

Ketika *admin* ingin melakukan penambahan seorang pengguna, maka dapat dipilih tombol tambah pengguna dan aplikasi akan menampilkan halaman untuk penambahan pelanggan. Data yang telah diisi oleh *admin*, kemudian akan divalidasi terlebih dahulu. Apabila belum *valid*, maka *admin* harus menambahkan

data yang diperlukan, namun apabila data tersebut sudah *valid*, maka data tersebut akan tersimpan ke dalam *database*.

Apabila *user/admin* memilih menu piutang, maka tampilan akan langsung menampilkan daftar piutang dalam *paging*. *User/admin* dapat menambahkan data piutang apabila ada pelanggan yang ingin melunasi pembayaran.

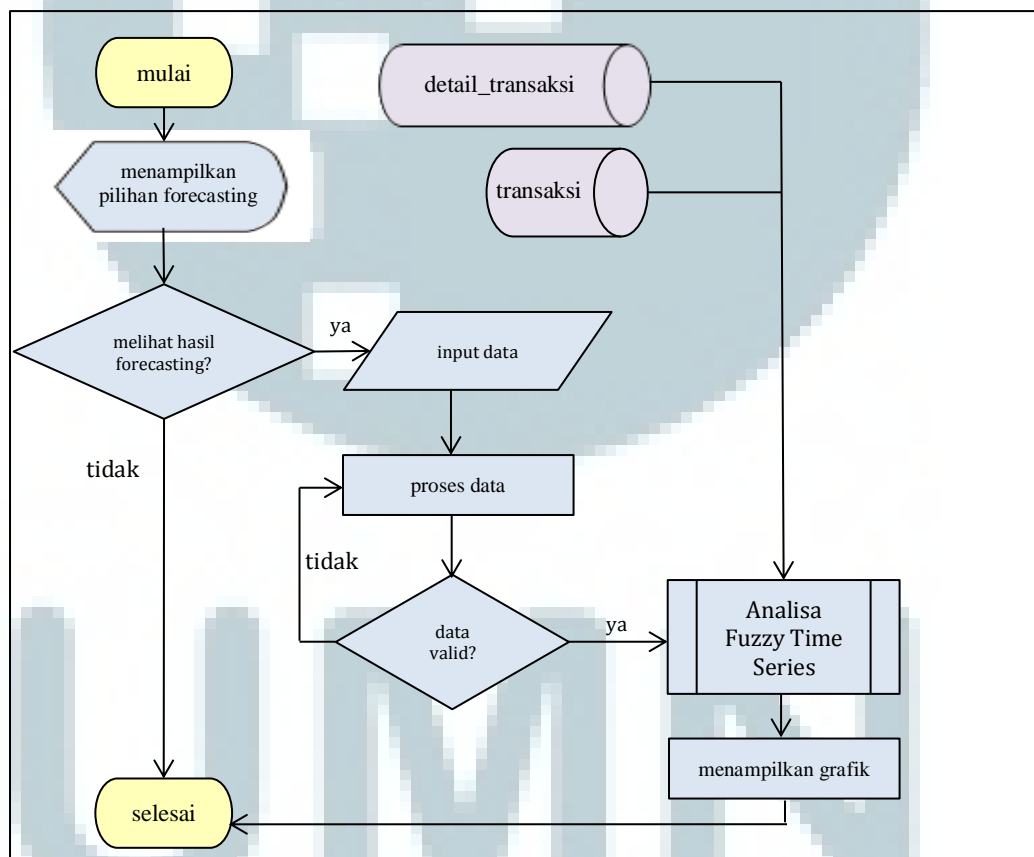


Gambar 3.7 *Flowchart* Menu Piutang

Pada *flowchart* menu piutang yang ditampilkan pada gambar 3.7, ketika *user/admin* memilih menu piutang, maka aplikasi akan menampilkan halaman dengan daftar piutang. Daftar piutang yang tersimpan dalam *database* akan ditampilkan dalam bentuk *paging*. Bila *user/admin* ingin mencetak daftar piutang, maka ia dapat memilih keterangan cetak suatu piutang. Apabila terdapat

pelanggan yang ingin melunasi pembayaran, maka *user/admin* dapat menambahkan data baru pelunasan pembayaran transaksi pelanggan tersebut. Apabila data yang ditambahkan *valid*, maka data tersebut akan disimpan ke dalam *database*, dan piutang transaksi pelanggan tersebut akan berkurang pada setiap pembayarannya.

Apabila *admin* memilih menu *forecasting*, maka tampilan akan menampilkan halaman dimana akan terdapat pilihan mengenai kategori barang yang ingin diforecasting. *Admin* dapat menginput tanggal untuk melakukan perhitungan *forecasting*.

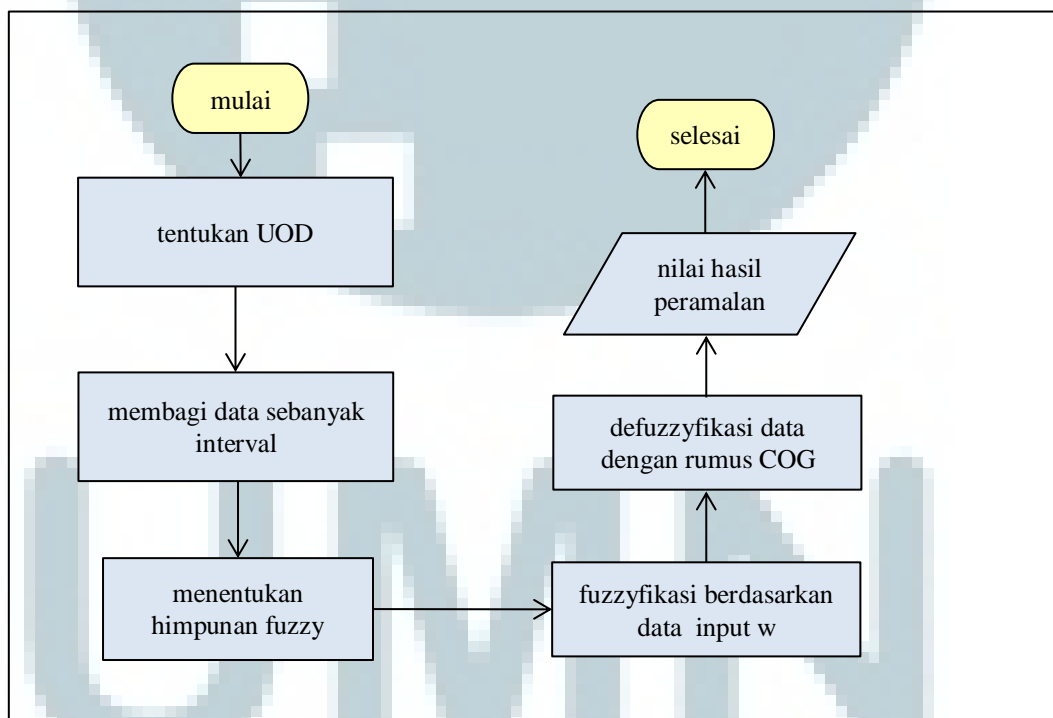


Gambar 3.8 Flowchart Menu Forecasting

Pada *flowchart* menu piutang yang ditampilkan pada gambar 3.8, ketika *admin* memilih menu *forecasting*, maka aplikasi akan menampilkan halaman

dimana *admin* memilih kategori barang mana yang ingin dilihat hasil *forecasting*-nya. Terdapat dua kategori barang yang dapat dipilih untuk melihat hasil *forecasting*, yaitu As Balok dan Transmisi.

Selain memilih kategori barang yang ingin dilihat, *admin* harus melakukan *input weight* (w) dan tanggal yang menentukan perhitungan berdasarkan nilai w yang dimulai dari tanggal dimulainya perhitungan *forecasting* yang akan dilakukan terhadap kategori barang yang telah dipilih tersebut. Apabila *admin* telah melakukannya, maka hasil *forecasting* dari data yang telah dipilih akan ditampilkan dalam bentuk grafik garis. Proses analisa *fuzzy time series* akan ditampilkan pada *flowchart* subroutine analisa *fuzzy time series* seperti pada gambar 3.9.

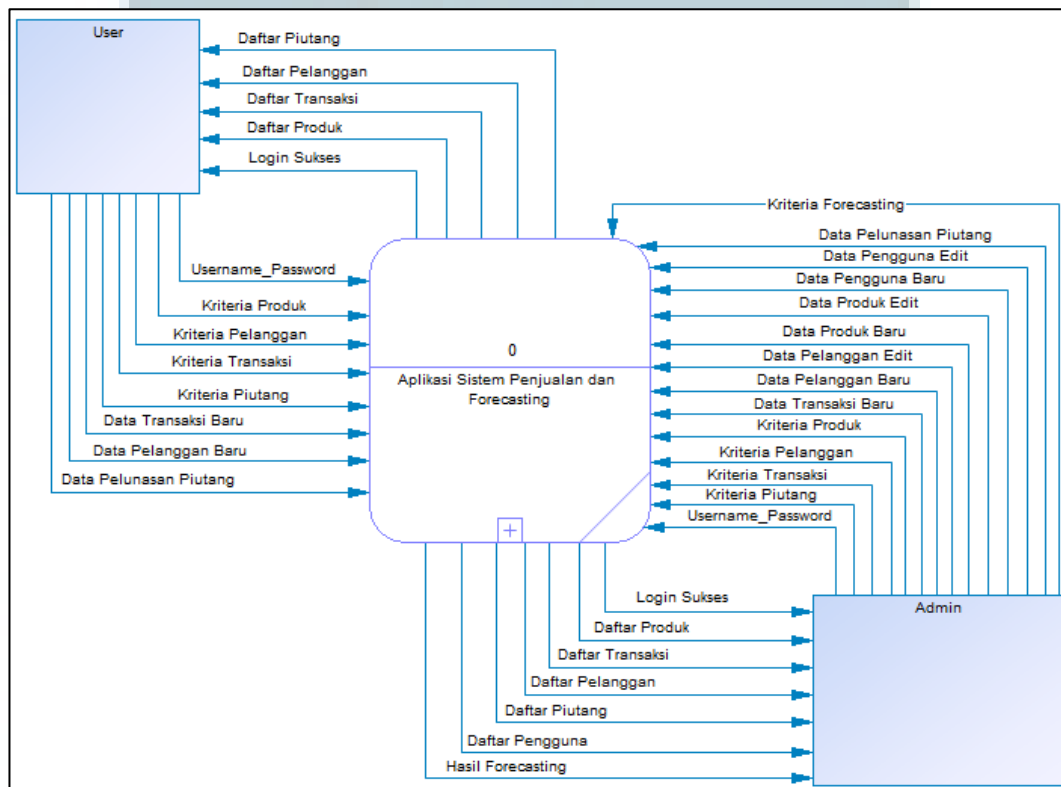


Gambar 3.9 Flowchart Subroutine Analisa Fuzzy Time Series

Pada gambar 3.9, proses analisa *fuzzy time series* dimulai dari menentukan *Universe of Discourse* (UOD) dari data transaksi berdasarkan rentang

waktu yang telah diinput oleh *admin*. Kemudian setelah itu dilakukan pembagian data berdasarkan jumlah interval yang telah dihitung. Setelah itu berdasarkan himpunan *fuzzy* yang telah dibentuk, dilakukan perhitungan fuzzyfikasi terhadap data berdasarkan nilai *w* yang telah dimasukkan oleh *admin*. Hasil perhitungan fuzzyfikasi tersebut kemudian didefuzzyfikasi dengan menggunakan metode *Center of Gravity* (COG), sehingga mendapatkan nilai hasil peramalan.

3.2.2 Data Flow Diagram



Gambar 3.10 Context Diagram

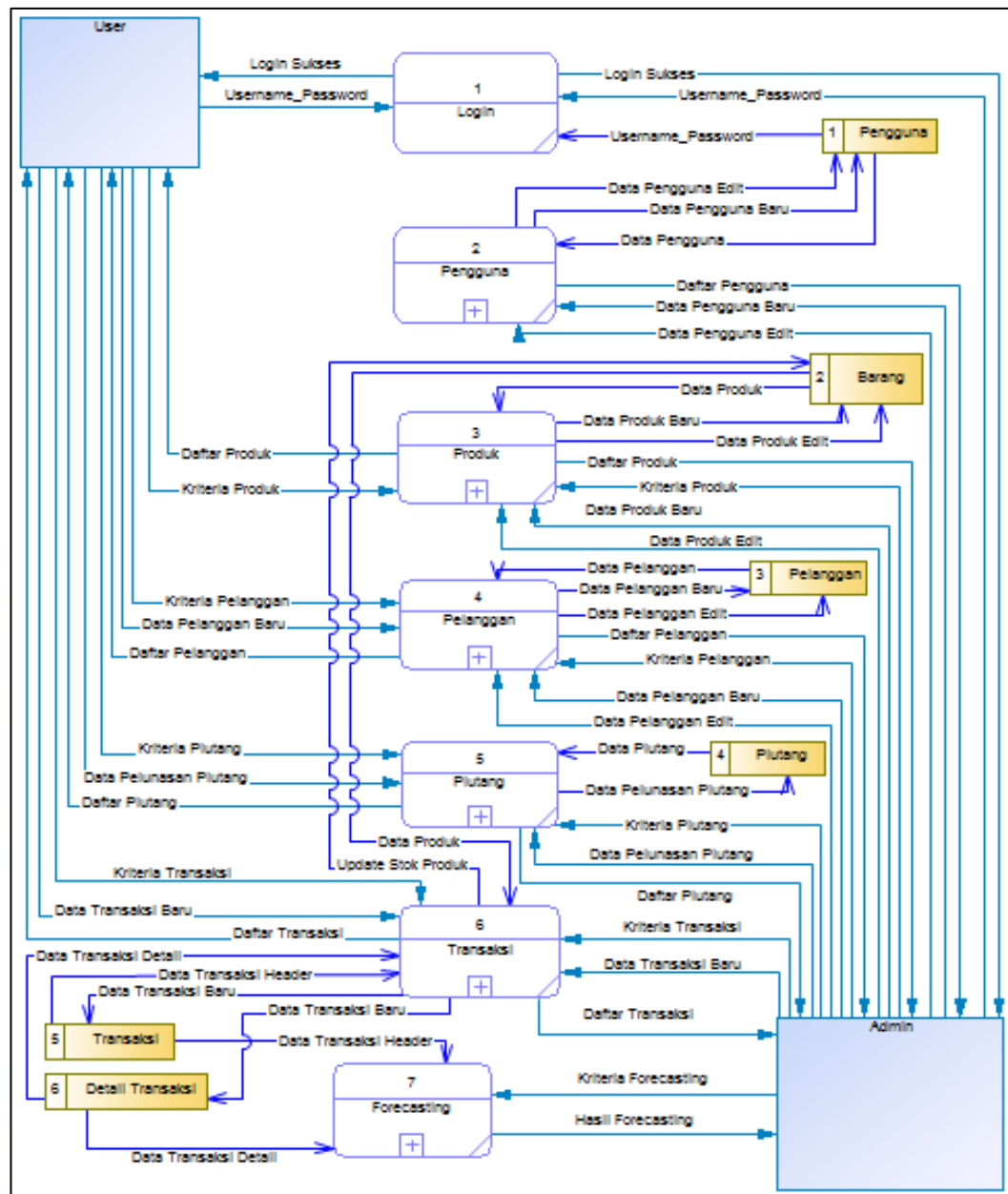
Gambar 3.10 adalah *context* diagram dari aplikasi sistem penjualan dan *forecasting* yang dibangun ini. Terdapat satu buah proses utama dengan dua buah entitas, yaitu *user* dan *admin*. *Context* diagram tersebut kemudian dipecah menjadi tujuh proses yang lebih rinci pada DFD *level* 1 seperti pada gambar 3.11,

yaitu proses *Login*, proses Pengguna, proses Produk, proses Pelanggan, proses Piutang, proses Transaksi, dan proses *Forecasting*.

Ketika memulai aplikasi ini, maka *user* atau *admin* perlu untuk melakukan *login* dengan memasukkan *username* dan *password*. *Username* dan *password* tersebut kemudian akan divalidasi dengan data yang terdapat dalam *database* dan mencocokkan *privilege* yang dimiliki oleh pengguna yang melakukan *login* untuk masuk ke dalam aplikasi.



UMN

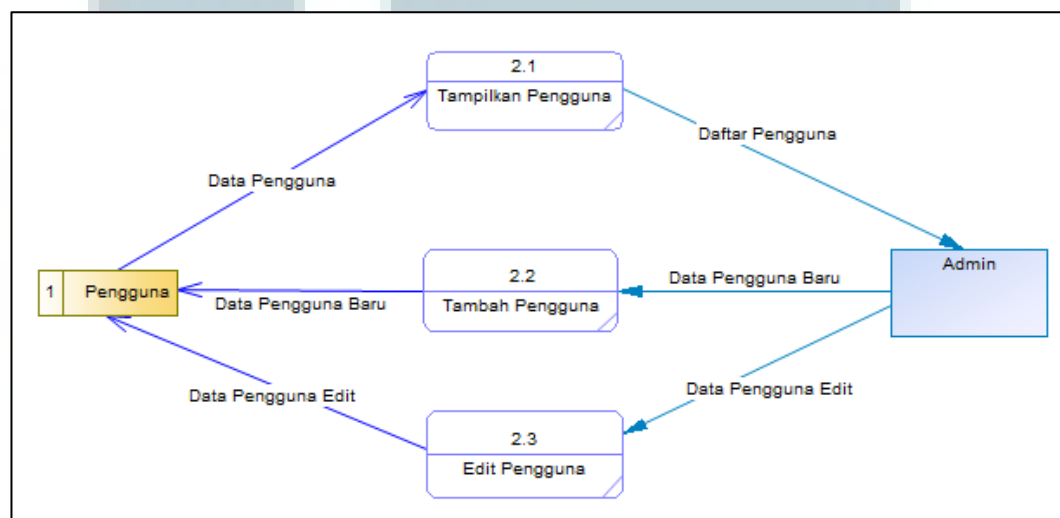


Gambar 3.11 DFD Level 1

Proses Pengguna hanya dapat diakses oleh seorang *admin*. Pada proses Pengguna ini dipecah lagi menjadi tiga proses yang lebih rinci pada DFD Level 2 Proses 2 seperti pada gambar 3.12. Ketiga proses tersebut yaitu proses Tampilkan Pengguna, proses Tambah Pengguna dan proses Edit Pengguna. Proses Tampilkan

Pengguna mengambil data dari *storage* Pengguna dan kemudian menampilkan data pada *admin* dalam bentuk daftar pengguna.

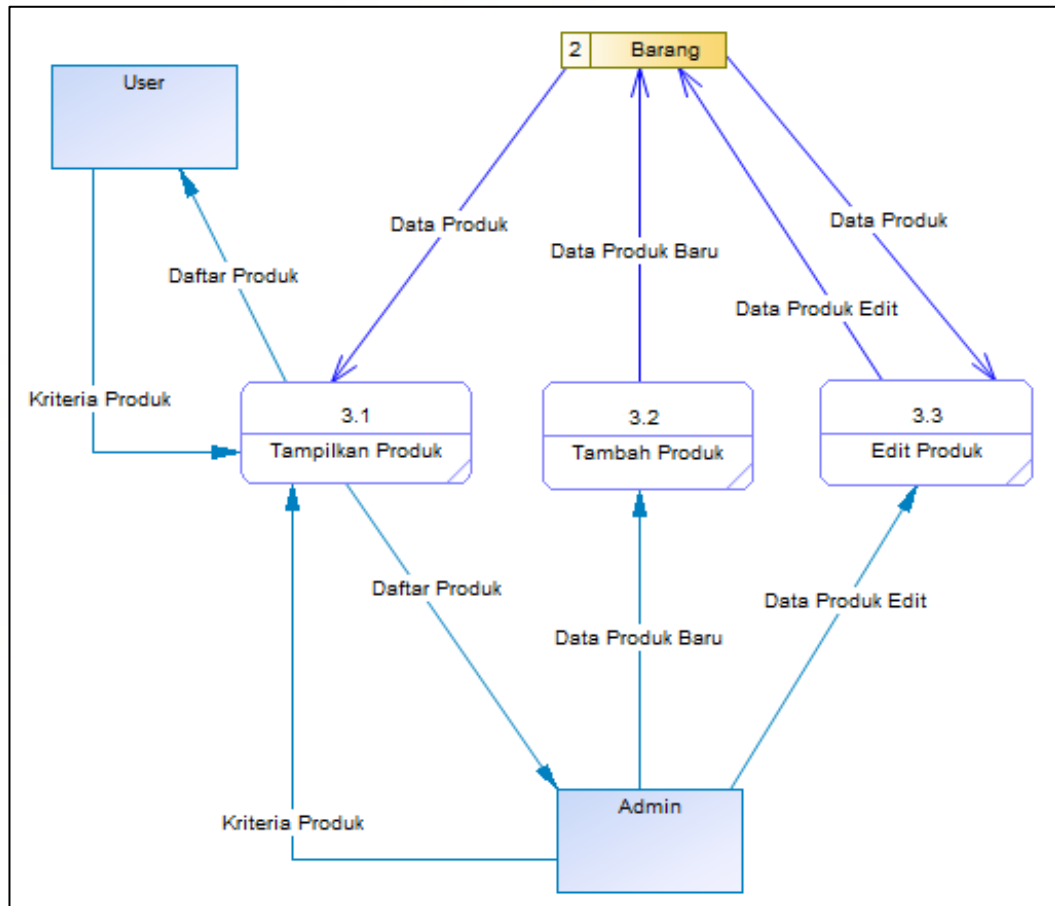
Proses Tambah Pengguna menerima data pengguna baru yang di *input* oleh *admin* dan kemudian menambahkan data baru tersebut ke dalam *storage* pengguna. Kemudian proses Edit Pengguna, menerima data pengguna yang ingin di *edit* dan kemudian melakukan *update* terhadap data pengguna yang di *edit* tersebut.



Gambar 3.12 DFD Level 2 Proses 2

Proses Produk dapat diakses oleh *user* atau *admin*. Proses Produk dipecah menjadi tiga proses lebih rinci seperti pada gambar 3.13, yaitu proses Tampilkan Produk, proses Tambah Produk dan proses Edit Produk. Proses Tampilkan Produk menerima data berupa kriteria pencarian produk yang di *input* oleh *user* atau *admin*, dan mengambil data dari *storage* Barang dan mengembalikan dalam bentuk daftar produk kepada *user* atau *admin*. Proses Tambah Produk hanya dapat diakses oleh *admin*, dimana *admin* melakukan *input* data produk baru yang kemudian akan ditambahkan ke dalam *storage* Barang.

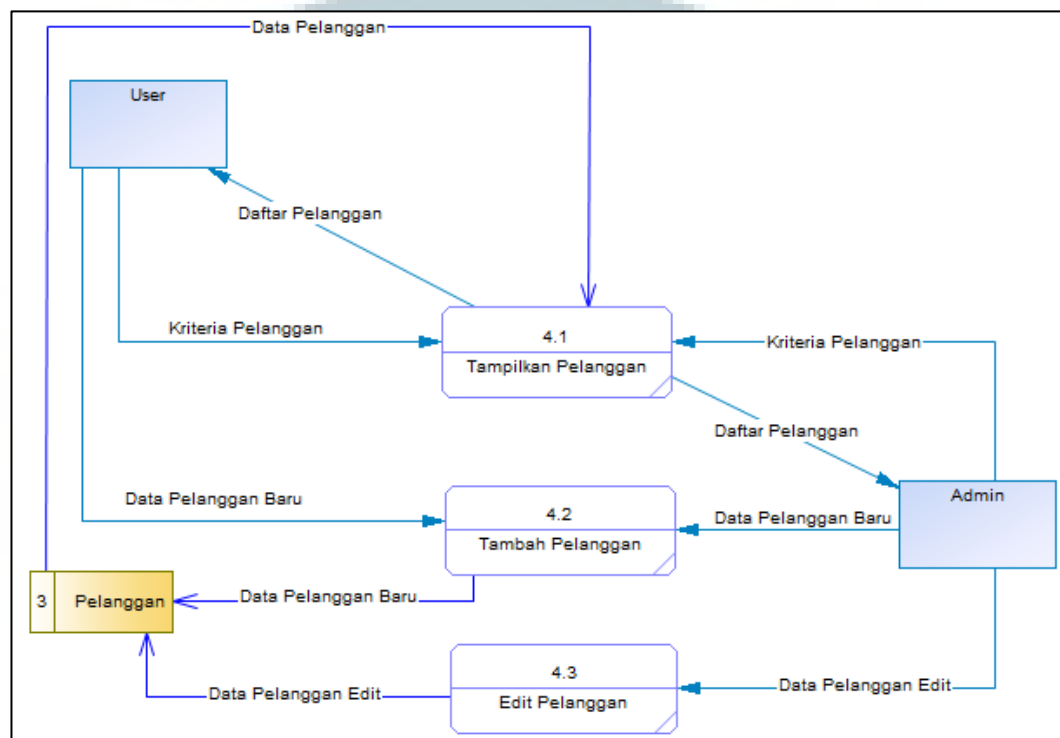
Proses Edit Produk juga hanya dapat diakses oleh *admin*, dimana *admin* melakukan *input* data pada produk yang ingin diubah dan kemudian melakukan *update* data pada *storage* Barang.



Gambar 3.13 DFD Level 2 Proses 3

Proses Pelanggan dapat diakses oleh *user* atau *admin*. Proses Pelanggan dipecah menjadi tiga proses lebih rinci seperti pada gambar 3.14, yaitu proses Tampilkan Pelanggan, proses Tambah Pelanggan dan proses Edit Pelanggan. Proses Tampilkan Pelanggan menerima data berupa kriteria pencarian pelanggan yang di *input* oleh *user* atau *admin*, yang kemudian mengambil data dari *storage* Pelanggan dan mengembalikan dalam bentuk daftar pelanggan kepada *user* atau *admin*. Proses Tambah Produk dilakukan dengan melakukan *input* data pelanggan

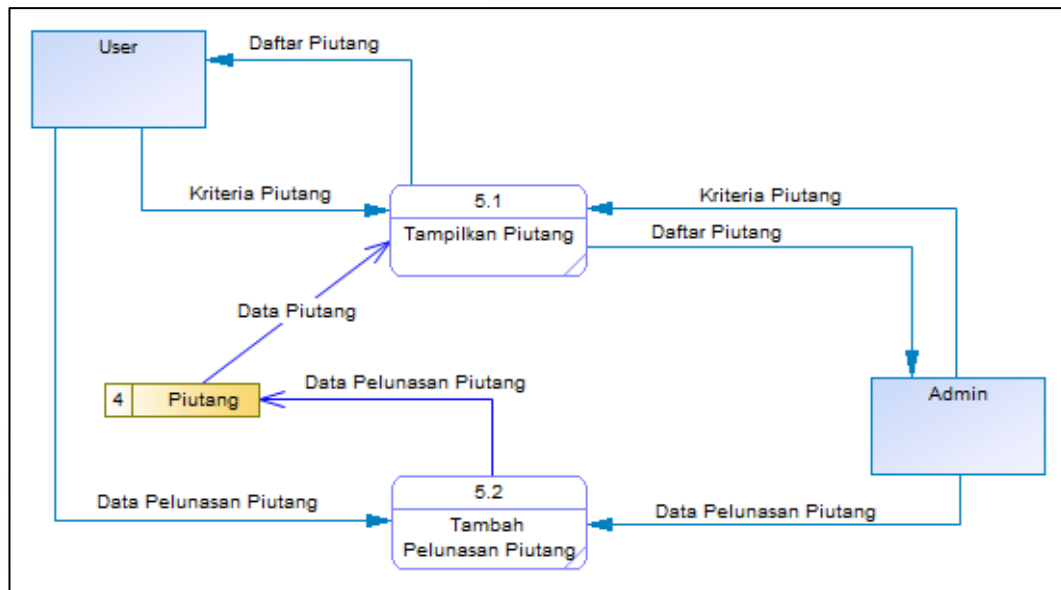
baru yang kemudian akan ditambahkan ke dalam *storage* Pelanggan. Proses Edit Pelanggan juga hanya dapat diakses oleh *admin* dimana *admin* melakukan *input* data pada pelanggan yang ingin diubah dan kemudian melakukan *update* data pada *storage* Pelanggan.



Gambar 3.14 DFD *Level 2* Proses 4

Proses Piutang dapat diakses oleh *user* atau *admin*. Proses Piutang dipecah menjadi dua proses lebih rinci seperti pada gambar 3.15, yaitu proses Tampilkan Piutang dan proses Tambah Pelunasan Piutang. Proses Tampilkan Piutang menerima data berupa kriteria pencarian piutang yang di *input* oleh *user* atau *admin*, yang kemudian mengambil data dari *storage* Piutang dan mengembalikan dalam bentuk daftar piutang kepada *user* atau *admin*. Proses Tambah Pelunasan Piutang dilakukan dengan melakukan *input* data pelunasan

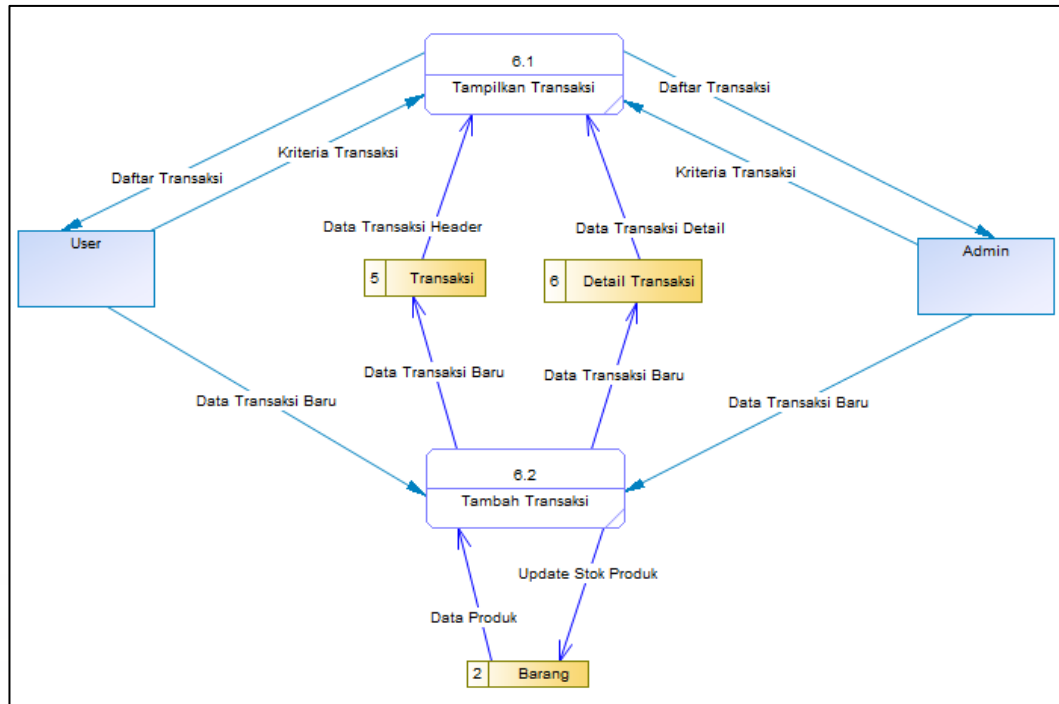
piutang baru sesuai dengan nomor transaksi yang kemudian akan ditambahkan ke dalam *storage* Piutang.



Gambar 3.15 DFD Level 2 Proses 5

Proses Transaksi dapat diakses oleh *user* atau *admin*. Proses Transaksi dipecah menjadi dua proses lebih rinci seperti pada gambar 3.16, yaitu proses Tampilkan Transaksi dan proses Tambah Transaksi. Proses Tampilkan Transaksi menerima data berupa kriteria pencarian transaksi yang di *input* oleh *user* atau *admin*, yang kemudian mengambil data dari *storage* Transaksi dan *storage* Detail Transaksi, dan mengembalikan dalam bentuk daftar transaksi kepada *user* atau *admin*.

Proses Tambah Transaksi dilakukan dengan melakukan *input* data yang kemudian akan ditambahkan ke dalam *storage* Transaksi dan *storage* Detail Transaksi. Setelah melakukan penyimpanan data transaksi, proses ini kemudian melakukan *update* data pengurangan stok barang pada *storage* Barang.



Gambar 3.16 DFD *Level 2* Proses 6

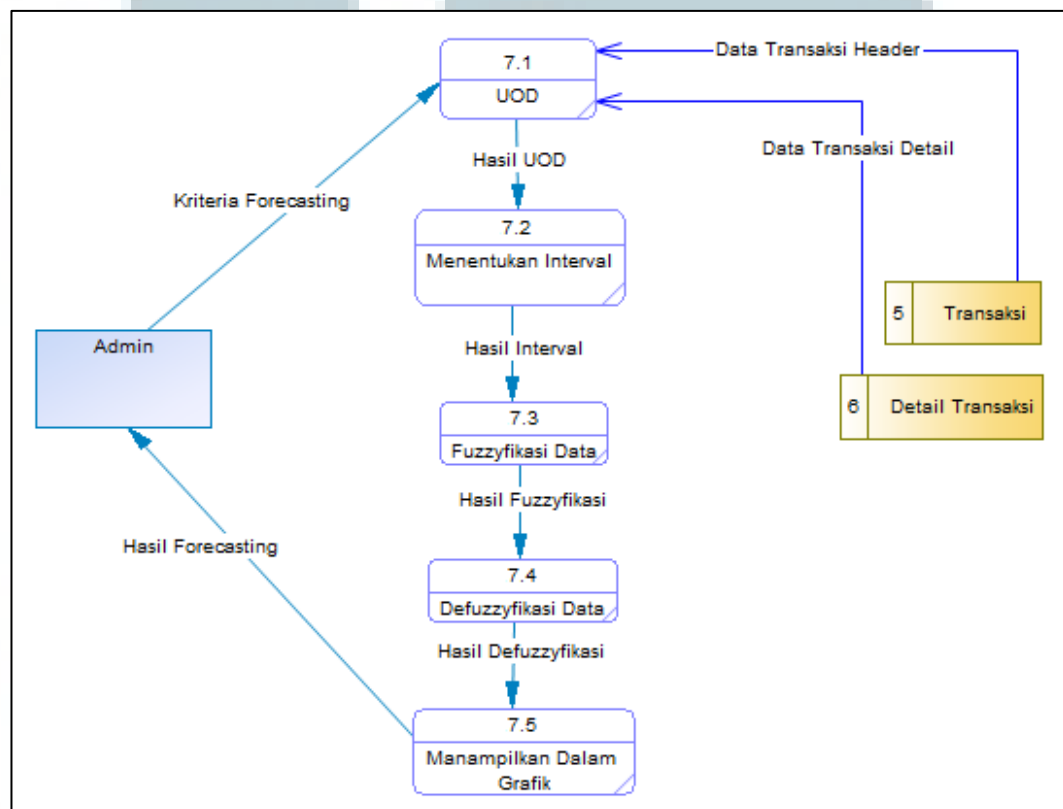
Proses Forecasting hanya dapat diakses oleh *admin*. Proses Forecasting dipecah menjadi lima proses lebih rinci seperti pada gambar 3.17, yaitu proses UOD, proses Menentukan Interval, proses Fuzzyfikasi Data, proses Defuzzyfikasi Data, dan proses Menampilkan Dalam Grafik.

Admin melakukan *input* data yang akan digunakan dalam perhitungan untuk hasil *forecasting*. *Admin* melakukan *filter* kriteria pada kategori barang yang akan dilakukan perhitungan *forecasting*. Pada proses UOD, akan ditentukan nilai minimal dan maksimal dari data berdasarkan rentang waktu yang di *input* oleh *admin*. Data yang digunakan dalam proses ini diambil dari data transaksi yang telah dilakukan dalam jangka waktu tertentu. Data transaksi diambil dari *storage* Transaksi dan *storage* Detail Transaksi.

Kemudian pada Proses Menentukan Interval dari data tersebut, ditentukan jumlah interval sesuai dengan UOD yang kemudian data tersebut

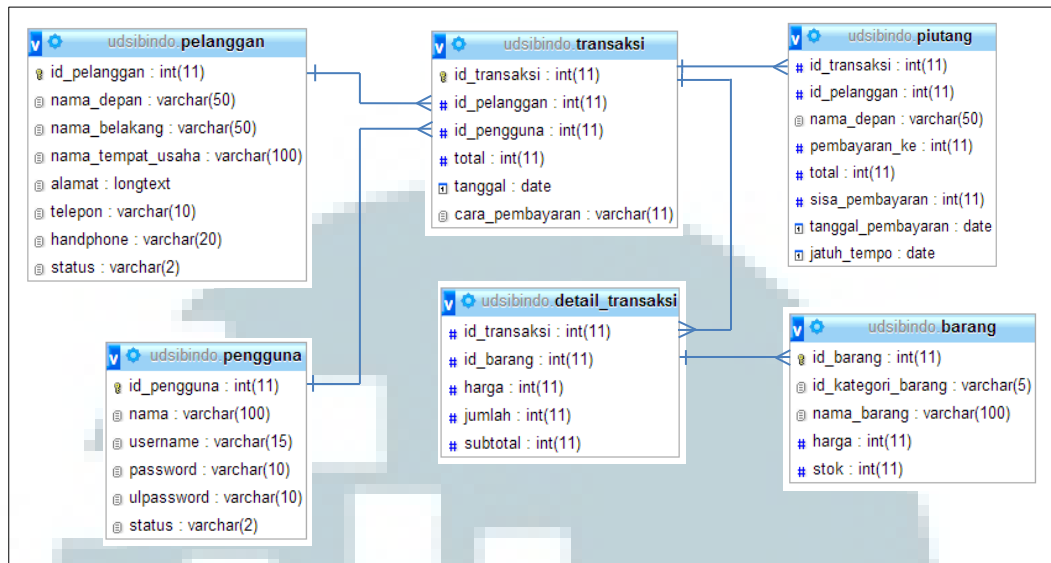
dibagi sejumlah interval yang telah diperhitungkan dan kemudian akan membentuk himpunan *fuzzy*. Pada Proses Fuzzyfikasi Data, data penjualan kemudian difuzzyfikasi berdasarkan nilai w yang telah di *input* oleh *admin*. Hasil fuzzyfikasi tersebut kemudian didefuzzyfikasi menggunakan metode *Center of Gravity* (COG).

Pada Proses Menampilkan Dalam Grafik, hasil perhitungan COG tersebut ditampilkan dalam grafik garis, sehingga *admin* dapat melihat hasil peramalan jenis barang tertentu.



Gambar 3.17 DFD *Level 2* Proses 7

3.2.3 Entity Relationship Diagram



Gambar 3.18 Entity Relationship Diagram

Pada gambar 3.18, dapat dilihat bahwa terdapat enam buah tabel yaitu tabel pelanggan, tabel pengguna, tabel transaksi, tabel detail transaksi, tabel piutang dan tabel barang. *Relationship* antara tabel pelanggan dengan tabel transaksi adalah *one to many* yang berarti seorang pelanggan dapat melakukan banyak transaksi. Kemudian *relationship* antara tabel pengguna dengan tabel transaksi adalah *one to many* dimana seorang user (pengguna) dapat meng *input* banyak transaksi. Lalu *relationship* antara tabel transaksi dengan tabel detail_transaksi adalah *one to many* yang berarti satu transaksi bisa memiliki lebih dari satu detail transaksi. *Relationship* antara tabel transaksi dengan tabel piutang adalah *one to many* dimana satu transaksi memiliki lebih dari satu pelunasan utang. Kemudian *relationship* antara tabel detail_transaksi dengan tabel barang adalah *one to many* yang berarti satu detail transaksi dapat memiliki lebih dari sebuah produk (barang).

Selain enam buah tabel tersebut terdapat dua buah tabel lagi, yaitu tabel `tabel_sementara` dan tabel `kategori_barang` guna menunjang tabel-tabel sebelumnya, sehingga terdapat delapan buah tabel yang digunakan dalam aplikasi ini.

3.2.4 Struktur Tabel

Dalam pembangunan aplikasi ini, terdapat beberapa tabel yang dibutuhkan. Terdapat dua tabel yang digunakan untuk menyimpan data penjualan yang masuk dan enam tabel lain yang digunakan untuk menunjang kedua tabel yang telah disebutkan sebelumnya. Berikut adalah struktur tabel yang digunakan dalam pembangunan aplikasi ini.

Nama Tabel : transaksi

Fungsi : digunakan untuk menyimpan data *header* transaksi yang masuk

Tabel 3.1 Tabel transaksi

Nama Field	Tipe Data	Keterangan
id_transaksi	int	Kode transaksi
id_pelanggan	int	Kode pelanggan
id_pengguna	int	Kode pengguna
total	int	Total transaksi
tanggal	date	Tanggal transaksi
cara_pembayaran	varchar(11)	Metode pembayaran

Nama Tabel : detail_transaksi

Fungsi : digunakan untuk menyimpan data detail transaksi yang masuk

Tabel 3.2 Tabel detail_transaksi

Nama Field	Tipe Data	Keterangan
id_transaksi	Int	Kode transaksi
id_barang	Int	Kode barang
harga	Int	Harga barang
jumlah	Int	Jumlah barang
subtotal	Int	Harga subtotal barang

Nama Tabel : pelanggan

Fungsi : digunakan untuk menyimpan data pelanggan

Tabel 3.3 Tabel pelanggan

Nama Field	Tipe Data	Keterangan
id_pelanggan	Int	Kode pelanggan
nama_depan	varchar(50)	Nama depan pelanggan
nama_belakang	varchar(50)	Nama belakang pelanggan
nama_tempat_usaha	varchar(100)	Nama tempat usaha pelanggan
alamat	longtext	Alamat pelanggan
telepon	varchar(10)	Nomor telepon
handphone	varchar(20)	Nomor <i>handphone</i>

Tabel 3.3 Tabel pelanggan (lanjutan)

Nama Field	Tipe Data	Keterangan
status	varchar(2)	Status pelanggan

Nama Tabel : pengguna

Fungsi : digunakan untuk menyimpan data pengguna

Tabel 3.4 Tabel pengguna

Nama Field	Tipe Data	Keterangan
id_pengguna	Int	Kode pengguna
nama	varchar(100)	Nama pengguna
username	varchar(15)	<i>Username</i> pengguna
password	varchar(10)	<i>Password</i> pengguna
ulpassword	varchar(10)	Konfirmasi <i>password</i>
status	varchar(2)	Status pengguna

Nama Tabel : barang

Fungsi : digunakan untuk menyimpan data produk

Tabel 3.5 Tabel barang

Nama Field	Tipe Data	Keterangan
id_barang	Int	Kode barang
id_kategori_barang	varchar(5)	Kode kategori barang

Tabel 3.5 Tabel barang (lanjutan)

Nama Field	Tipe Data	Keterangan
nama_barang	varchar(100)	Nama barang
harga	Int	Harga barang
stok	Int	Stok barang

Nama Tabel : kategori_barang

Fungsi : digunakan untuk menyimpan kategori produk

Tabel 3.6 Tabel kategori_barang

Nama Field	Tipe Data	Keterangan
id_kategori_barang	varchar(5)	Kode kategori barang
nama_kategori_barang	varchar(100)	Nama kategori barang

Nama Tabel : piutang

Fungsi : digunakan untuk menyimpan data pelunasan piutang

Tabel 3.7 Tabel piutang

Nama Field	Tipe Data	Keterangan
id_transaksi	Int	Kode transaksi
id_pelanggan	Int	Kode pelanggan
nama_depan	varchar(50)	Nama depan pelanggan
pembayaran_ke	Int	Pembayaran ke-n

Tabel 3.7 Tabel piutang (lanjutan)

Nama Field	Tipe Data	Keterangan
total	Int	Jumlah pelunasan piutang
sisa_pembayaran	Int	Sisa piutang
tanggal_pembayaran	Date	Tanggal pelunasan piutang
jatuh_tempo	Date	Tanggal jatuh tempo pelunasan

Nama Tabel : tabel_sementara

Fungsi : digunakan untuk menyimpan data *order* sementara

Tabel 3.8 Tabel sementara

Nama Field	Tipe Data	Keterangan
id_barang	Int	Kode barang
nama_barang	varchar(100)	Nama barang
harga	Int	Harga barang
jumlah	Int	Jumlah barang
subtotal	Int	Subtotal harga barang

3.2.5 Rancangan Antarmuka

Dalam aplikasi ini memiliki rancangan antarmuka sebagai berikut:

- Rancangan tampilan (halaman *login*) ketika aplikasi dijalankan

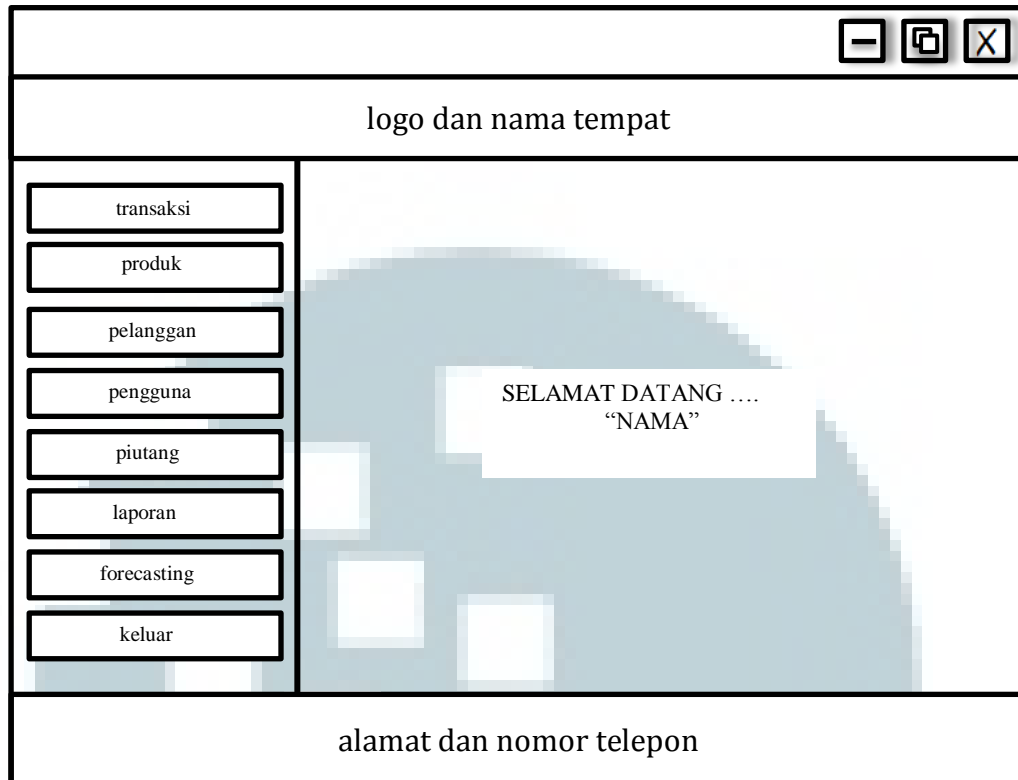
Ketika aplikasi dijalankan, maka akan muncul halaman *login*. Halaman *login* berisi logo dan nama tempat usaha, dua buah *text field* yang

digunakan sebagai *username* dan *password*, dan satu tombol yang digunakan untuk masuk.

The image shows a login form window with a title bar containing minimize, maximize, and close buttons. The window title is "Selamat Datang". On the left side, there are two rectangular boxes: one labeled "LOGO" and another labeled "NAMA PERUSAHAAN". On the right side, there are three input fields: "Username", "Password", and a button labeled "MASUK". The background of the window features a large, faint watermark of the UMN logo.

Gambar 3.19 Rancangan Tampilan Halaman Login

- Rancangan tampilan awal aplikasi setelah melakukan *login*.
Setelah melakukan *login*, maka bagian kiri rancangan tampilan aplikasi memiliki delapan buah menu yaitu transaksi, produk, pelanggan, pengguna, piutang, laporan, *forecasting*, dan keluar. Untuk seorang *user* ketika masuk ke aplikasi, hanya terdapat lima buah menu yaitu transaksi, produk, pelanggan, piutang, dan keluar. Pada bagian *header* terdapat logo dan nama tempat usaha. Sedangkan pada bagian *footer*, akan ditampilkan alamat dan nomor telepon tempat usaha.



Gambar 3.20 Rancangan Tampilan Halaman Setelah *Login*

- Rancangan tampilan daftar transaksi
Ketika menu transaksi dipilih, maka aplikasi akan menampilkan daftar transaksi. Halaman ini berisi sebuah *button* untuk melakukan penambahan transaksi, sebuah *text field* untuk pencarian kriteria dan sebuah *list* yang digunakan untuk menampilkan daftar transaksi. Daftar transaksi yang ditampilkan pada halaman ini menggunakan fitur *paging*, sehingga dapat membuat tampilan lebih rapi dan teratur. *List* transaksi memiliki empat kolom yaitu Id Transaksi, Tanggal, Total dan Cetak. Kolom Cetak berfungsi untuk mencetak detail transaksi dari transaksi yang dipilih.

logo dan nama tempat

transaksi

produk

pelanggan

pengguna

piutang

laporan

forecasting

keluar

Tambah Transaksi

Mencari kriteria

Id Transaksi	Tanggal	Total	Cetak

Halaman 1 dari 10 1 2 3 4 ... 9 10

alamat dan nomor telepon

Gambar 3.21 Rancangan Tampilan Halaman Daftar Transaksi

- Rancangan tampilan tambah transaksi

Pada halaman tambah transaksi, pada bagian kiri atas terdapat status pelanggan dengan dua *radio button*. Apabila status pembeli adalah pelanggan, maka terdapat sebuah *text field autocomplete* untuk nama tempat usaha pelanggan tersebut. Bagian kanan atas terdapat *text field* nomor transaksi yang otomatis bertambah satu ketika melakukan penambahan transaksi. Di sebelah *text field* nomor transaksi terdapat *text field* tanggal. Kemudian *user* atau *admin* akan menginput nama barang pada *combo box* dan jumlah barang yang dibeli pada *text field* jumlah. Ketika menekan tombol tambah, maka barang tersebut akan masuk ke dalam tabel dibawahnya, beserta total harga pembayaran.

Apabila ingin melakukan pembayaran, maka terdapat dua status pembayaran, yaitu secara tunai atau secara kredit. Apabila melakukan pembayaran secara kredit, maka *user* atau *admin* perlu memasukkan tanggal jatuh tempo pembayaran dan uang muka pembayaran dari transaksi tersebut. Setelah itu *user* atau *admin* dapat menekan tombol proses untuk menyelesaikan transaksi tersebut.

Gambar 3.22 Rancangan Tampilan Halaman Tambah Transaksi

- Rancangan tampilan daftar produk

Ketika menu produk dipilih, maka aplikasi akan menampilkan daftar produk. Halaman ini berisi sebuah *button* untuk melakukan penambahan produk, sebuah *text field* untuk pencarian kriteria dan sebuah *list* yang digunakan untuk menampilkan daftar produk. Pada kolom keterangan, akan terdapat tulisan *edit* yang berfungsi untuk melakukan perubahan

terhadap data produk tersebut. Daftar produk yang ditampilkan pada halaman ini menggunakan fitur *paging*, sehingga dapat membuat tampilan lebih rapi dan teratur.

logo dan nama tempat

transaksi
produk
pelanggan
pengguna
piutang
laporan
forecasting
keluar

Tambah Produk

Mencari kriteria

Id Barang	Nama Barang	Harga	Stok	Ket

Halaman 1 dari 10 1 2 3 4 ... 9 10

alamat dan nomor telepon

Gambar 3.23 Rancangan Tampilan Halaman Daftar Produk

- Rancangan tampilan *edit* produk.

Pada tampilan *edit* produk terdapat dua buah *text field* yang *disabled* (tidak dapat diganti), yaitu id kategori dan id barang, dan tiga buah *text field* untuk mengganti data produk, yaitu nama barang, stok dan harga. Kemudian terdapat dua buah tombol, yaitu tombol *edit* produk yang berguna untuk meng *update* data produk yang telah di *edit* dan tombol kembali untuk pembatalan aksi meng *edit* produk.

logo dan nama tempat

Edit Produk

Id Kategori

Id Barang

Nama Barang

Stok

Harga

alamat dan nomor telepon

Gambar 3.24 Rancangan Tampilan Halaman *Edit Produk*

- Rancangan tampilan tambah produk
 Pada halaman tambah produk, terdapat sebuah *combo box* yang berguna untuk memilih kategori barang dari produk baru, dan tiga buah *text field* sebagai data dari produk baru, yaitu nama barang, harga dan jumlah. Kemudian terdapat dua buah tombol, yaitu tombol tambah produk untuk melakukan penyimpanan data dan tombol kembali untuk melakukan pembatalan aksi.

Gambar 3.25 Rancangan Tampilan Halaman Tambah Produk

- Rancangan tampilan daftar pelanggan

Ketika menu pelanggan dipilih, maka aplikasi akan menampilkan daftar pelanggan. Halaman ini berisi sebuah *button* untuk melakukan penambahan pelanggan, sebuah *text field* untuk pencarian kriteria dan sebuah *list* yang digunakan untuk menampilkan daftar pelanggan. *List* pelanggan memiliki tujuh kolom yaitu ID, Nama, Tempat Usaha, Alamat, Telepon, Handphone dan Keterangan. Pada kolom keterangan, akan terdapat tulisan *edit* yang berfungsi untuk melakukan perubahan terhadap data pelanggan tersebut. Daftar pelanggan yang ditampilkan pada halaman ini menggunakan fitur *paging* sehingga dapat membuat tampilan lebih rapi dan teratur.

logo dan nama tempat

transaksi

produk

pelanggan

pengguna

piutang

laporan

forecasting

keluar

Tambah Pelanggan

Mencari kriteria

ID	Nama	Tempat Usaha	Alamat	Telepon	Handphone	Ket

Halaman 1 dari 10 1 2 3 4 ... 9 10

alamat dan nomor telepon

Gambar 3.26 Rancangan Tampilan Halaman Daftar Pelanggan

- Rancangan tampilan *edit* pelanggan

Pada tampilan *edit* pelanggan terdapat satu buah *text field* yang *disabled* (tidak dapat diganti), yaitu id pelanggan, dan lima buah *text field* untuk meng *edit* data pelanggan yaitu nama, tempat usaha, alamat, telepon dan handphone. Kemudian terdapat dua buah tombol, yaitu tombol *edit* pelanggan yang berguna untuk meng *update* data pelanggan yang telah di *edit* dan tombol kembali untuk pembatalan aksi meng *edit* pelanggan.

logo dan nama tempat

Edit Pelanggan

Id Pelanggan

Nama

Tempat Usaha

Alamat

Telepon

Handphone

alamat dan nomor telepon

Gambar 3.27 Rancangan Tampilan Halaman *Edit Pelanggan*

- Rancangan tampilan tambah pelanggan
Pada halaman tambah pelanggan, terdapat lima buah *text field* sebagai data dari pelanggan baru yaitu nama, tempat usaha, alamat, telepon dan handphone. Kemudian terdapat dua buah tombol, yaitu tombol tambah pelanggan untuk melakukan penyimpanan data dan tombol kembali untuk melakukan pembatalan aksi.

logo dan nama tempat

Tambah Pelanggan

Nama

Tempat Usaha

Alamat

Telepon

Handphone

alamat dan nomor telepon

Gambar 3.28 Rancangan Tampilan Halaman Tambah Pelanggan

- Rancangan tampilan daftar pengguna
Ketika menu pengguna dipilih, maka aplikasi akan menampilkan daftar pengguna. Halaman ini berisi sebuah *button* untuk melakukan penambahan pengguna, sebuah *text field* untuk pencarian kriteria dan sebuah *list* yang digunakan untuk menampilkan daftar pengguna. *List* pengguna memiliki lima kolom yaitu ID, Nama, Username, Status dan Keterangan. Pada kolom keterangan terdapat tulisan *edit* yang berfungsi untuk melakukan perubahan terhadap data pengguna tersebut.

logo dan nama tempat

transaksi

produk

pelanggan

pengguna

piutang

laporan

forecasting

keluar

Tambah Pengguna

Mencari kriteria

ID	Nama	Username	Status	Keterangan

Halaman 1 dari 10 1 2 3 4 ... 9 10

alamat dan nomor telepon

Gambar 3.29 Rancangan Tampilan Halaman Daftar Pengguna

- Rancangan tampilan *edit* pengguna

Pada tampilan *edit* pengguna terdapat satu buah *text field* yang *disabled* (tidak dapat diganti), yaitu id pengguna, dan tiga buah *text field* untuk meng *edit* data pengguna yaitu nama, *username* dan *password*. Lalu terdapat dua *radio button* sebagai status pengguna, yaitu sebagai *user* atau *admin*. Kemudian terdapat dua buah tombol yaitu tombol *edit* pengguna yang berguna untuk meng *update* data pengguna yang telah di *edit* dan tombol kembali untuk pembatalan aksi meng *edit* pengguna.

logo dan nama tempat

Edit Pengguna

Id Pengguna

Nama

Username

Password

Status User Admin

alamat dan nomor telepon

Gambar 3.30 Rancangan Tampilan Halaman *Edit Pengguna*

- Rancangan tampilan tambah pengguna

Pada tampilan tambah pengguna terdapat empat buah *text field* untuk menambah data pengguna baru yaitu nama, *username*, *password* dan ulangi *password*. Lalu terdapat dua *radio button* sebagai status pengguna, yaitu sebagai *user* atau *admin*. Kemudian terdapat dua buah tombol, yaitu tombol tambah pengguna yang berguna untuk melakukan penyimpanan data pengguna yang baru dan tombol kembali untuk pembatalan aksi menambah pengguna.

logo dan nama tempat

Tambah Pengguna

Nama

Username

Password

Ulangi Password

Status User Admin

alamat dan nomor telepon

Gambar 3.31 Rancangan Tampilan Halaman Tambah Pengguna

- Rancangan tampilan daftar piutang

Ketika menu piutang dipilih, maka aplikasi akan menampilkan daftar piutang. Halaman ini berisi sebuah *text field* untuk pencarian kriteria dan sebuah *list* yang digunakan untuk menampilkan daftar piutang. *List* piutang memiliki lima kolom yaitu Id Transaksi, Id Pelanggan, Sisa Pembayaran, Jatuh Tempo dan Keterangan. Pada kolom keterangan terbagi menjadi dua kolom yang nantinya pada kolom pertama akan terdapat tulisan cetak yang berfungsi untuk melakukan cetak daftar piutang dari sebuah transaksi dan kolom kedua terdapat tulisan pelunasan yang berfungsi untuk menambah data pelunasan dari sebuah transaksi.

logo dan nama tempat

transaksi

produk

pelanggan

pengguna

piutang

laporan

forecasting

keluar

Mencari kriteria

Id Transaksi	Id Pelanggan	Sisa Pembayaran	Jatuh Tempo	Keterangan

Halaman 1 dari 10 ...

alamat dan nomor telepon

Gambar 3.32 Rancangan Tampilan Halaman Daftar Piutang

- Rancangan tampilan tambah pelunasan piutang
- Pada halaman ini, terdapat delapan *text field* yaitu pembayaran ke, id transaksi, id pelanggan, nama pelanggan, sisa pembayaran, jatuh tempo, tanggal, dan total. Namun hanya *text field* total yang bersifat *enabled*, sehingga dapat dimasukkan jumlah pembayaran untuk melakukan pelunasan. Apabila sudah memasukkan, maka dapat menekan tombol tambah pelunasan, sehingga data yang telah di *input* masuk dan melakukan pengurangan sisa pembayaran dari transaksi tersebut.

logo dan nama tempat

Tambah Pelunasan

Pembayaran Ke

Id Transaksi

Id Pelanggan

Nama Pelanggan

Sisa Pembayaran

Jatuh Tempo

Tanggal

Total

alamat dan nomor telepon

Gambar 3.33 Rancangan Tampilan Halaman Tambah Pelunasan Piutang

- Rancangan tampilan laporan
Ketika menu laporan dipilih, maka aplikasi akan menampilkan daftar laporan. Halaman ini berisi sebuah *text field* untuk pencarian kriteria dan sebuah *list* yang digunakan untuk menampilkan daftar laporan. *List* pengguna memiliki empat kolom yaitu tanggal, jumlah, total dan cetak. Pada kolom cetak terdapat tulisan cetak yang berfungsi untuk mencetak data transaksi yang terjadi pada tanggal tersebut.

logo dan nama tempat

transaksi

produk

pelanggan

pengguna

piutang

laporan

forecasting

keluar

Mencari kriteria

Tanggal	Jumlah	Total	Cetak

Halaman 1 dari 10 1 2 3 4 ... 9 10

alamat dan nomor telepon

Gambar 3.34 Rancangan Tampilan Halaman Laporan

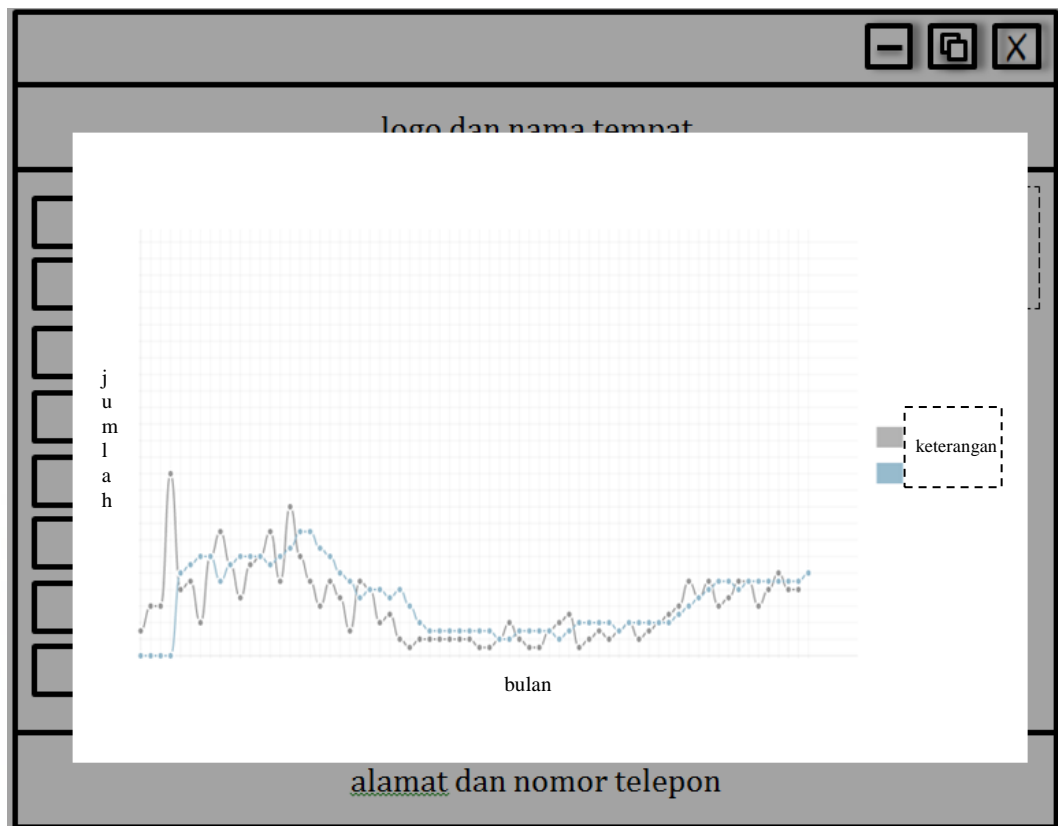
- Rancangan tampilan *forecasting*

Pada halaman ini terdapat dua *combo box* untuk memasukkan bulan dan tahun dimulainya perhitungan *forecasting*, satu *text field* untuk memasukkan *weight* guna patokan setiap berapa bulan perhitungan dilakukan, kemudian terdapat *radio button* untuk memilih kategori barang yang akan di*forecasting* dan terakhir sebuah tombol pilih untuk memulai perhitungan. Ketika tombol pilih ditekan, akan muncul grafik setiap jenis barang. Pada bagian kanan atas terdapat keterangan dimana menjelaskan jumlah jenis barang paling banyak dan paling sedikit dari hasil peramalan satu bulan berikutnya.

Gambar 3.35 Rancangan Tampilan Halaman *Forecasting*

Apabila ingin melihat detail peramalan suatu jenis barang, maka dapat menekan langsung gambar grafik jenis barang tersebut. Gambar grafik dari suatu jenis barang akan muncul seperti pada gambar 3.36. Pada bagian kanan grafik terdapat keterangan warna dari grafik.

UMMN



Gambar 3.36 Rancangan Tampilan Halaman Detail Grafik

UMMN