

BAB 3

METODOLOGI PENELITIAN

3.1 Metode Penelitian

Penelitian ini menggunakan beberapa metode antara lain sebagai berikut:

1. Telaah Literatur

Telaah Literatur yang digunakan dalam penelitian ini dilakukan dengan mencari dan mempelajari beberapa teori yang bersumber dari pakar terkait, jurnal ilmiah dan karya tulis ilmiah lainnya. Teori-teori yang digunakan antara lain menganalisa masalah pada perusahaan, rancang bangun, aplikasi, prediksi, prediksi penjualan, *fuzzy time series*, algoritma genetika, AFER (*Average Forecasting Error Rate*).

2. Analisa Kebutuhan

Analisa Kebutuhan dilakukan untuk mengetahui dan mendapatkan informasi yang diperlukan dalam proses pembuatan rancang bangun aplikasi prediksi penjualan menggunakan metode *fuzzy time series* dengan optimasi algoritma genetika. Informasi yang dibutuhkan berupa data penjualan berupa angka selama kurun waktu tiga tahun terakhir.

3. Pengumpulan Data

Proses pengumpulan data dilakukan bertujuan untuk memperoleh hasil

prediksi penjualan Toko Kue Kris. Pengumpulan data dilakukan dengan cara melakukan wawancara mengenai masalah yang dihadapi oleh Toko Kue Kris dan meminta data penjualan sebagai informasi pendukung dalam memprediksi penjualan di masa yang akan datang. Data yang didapatkan merupakan data penjualan Januari 2018 sampai dengan Desember 2020.

4. Perancangan Sistem

Perancangan sistem dilakukan dengan cara membuat *sitemap*, *flowchart*, skema basis data, struktur tabel basis data dan desain antarmuka sistem dari aplikasi prediksi penjualan ini serta pembuatan tabel berupa data penjualan yang didapat dari pakar terkait.

5. Implementasi

Implementasi dilakukan dengan membuat aplikasi berdasarkan desain yang telah dibuat sebelumnya menggunakan HTML, CSS dan Javascript. Pembuatan aplikasi menggunakan bahasa pemrograman PHP dan menggunakan basis data MySQL.

6. Testing dan Debugging

Proses ini bertujuan untuk menguji apakah metode *fuzzy time series* dan algoritma genetika berhasil diimplementasikan serta dapat berjalan dengan baik. Apabila terjadi kesalahan, akan dilakukan perbaikan pada *source code*.

7. Uji Coba dan Evaluasi

Uji coba dan evaluasi dilakukan dengan beberapa metode seperti uji coba *population size*, generasi dan tingkat akurasi dengan menggunakan AFER (*Average Forecasting Error Rate*).

8. Konsultasi dan Penulisan

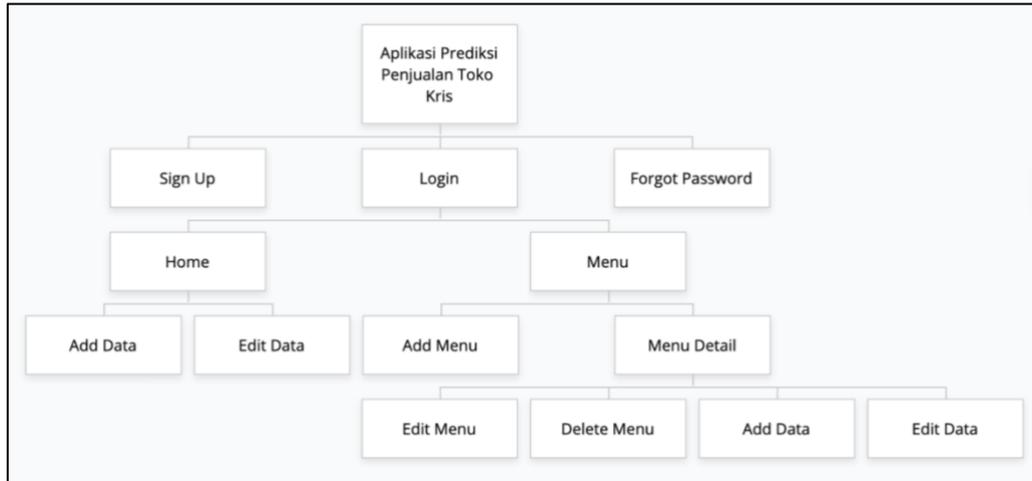
Penulisan laporan dibuat sebagai dokumentasi dari penelitian dan pembuatan aplikasi sehingga dapat memberikan informasi untuk penelitian serupa lainnya.

3.2 Perancangan Sistem

Sebelum melakukan implementasi sistem prediksi penjualan, dibutuhkan perancangan sistem yang terdiri dari *sitemap*, *flowchart*, skema basis data, struktur tabel basis data dan desain antarmuka sistem. Perancangan sistem dilakukan untuk memudahkan tahap implementasi agar lebih sistematis dan lebih terarah.

3.2.1 Sitemap

Sitemap merupakan rancangan alur aplikasi dari keseluruhan halaman *website* dan menggambarkan hubungan dari satu halaman ke halaman lainnya. Dengan adanya *sitemap*, diharapkan dapat mempermudah tahap implementasi setiap halaman yang akan dibangun.



Gambar 3.1 Sitemap halaman aplikasi prediksi penjualan Toko Kue Kris

Gambar 3.1 merupakan sitemap untuk setiap halaman yang akan diimplementasikan ke dalam aplikasi prediksi penjualan Toko Kue Kris. Saat aplikasi pertama kali dibuka, sistem akan menampilkan halaman *Login*. Apabila *user* belum memiliki akun, maka *user* dapat membuat akun melalui halaman *Sign Up*. Apabila *user* memiliki akun tetapi lupa *password*, maka *user* dapat membuat *password* baru melalui halaman *Forgot Password*. Apabila *user* sudah memiliki akun, *user* dapat melakukan *login*. Setelah berhasil melakukan *login*, *user* diarahkan menuju halaman *Home* dimana *user* dapat melihat data historis total penjualan, grafik garis dan nilai prediksi total penjualan untuk bulan berikutnya. Selain itu, *user* dapat menambahkan data baru ke dalam tabel data historis dan *user* dapat merubah nilai total penjualan perbulan dari data yang sudah ada. Pada halaman *Menu*, *user* dapat melihat beberapa menu yang telah ditambahkan ke *database* dan membuat menu baru. Apabila *user* memilih salah satu menu, *user* akan diarahkan ke halaman *menu detail* dimana *user* dapat melihat data historis dari menu tersebut, grafik garis dan nilai prediksi total penjualan untuk bulan berikutnya. *User* dapat merubah detail menu dan dapat menghapus menu tersebut.

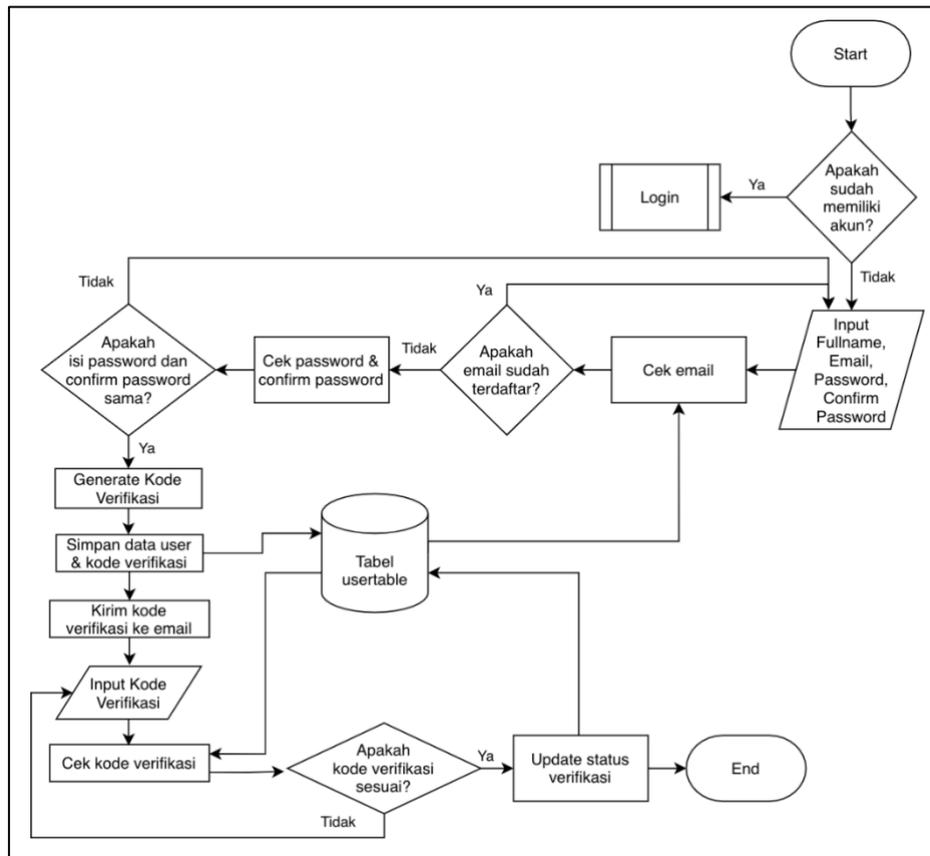
Selain itu, *user* dapat menambahkan data baru ke dalam tabel data historis dan *user* dapat merubah nilai total penjualan perbulan dari data yang sudah ada.

3.2.2 Flowchart

Flowchart merupakan sebuah bagan alur yang berisi bentuk serta simbol dengan makna tertentu, yang disusun secara sistematis sebagai urutan dan hubungan antar proses secara detail. *Flowchart* akan dibagi menjadi beberapa bagian antara lain *flowchart* halaman *sign up*, *flowchart* halaman *login*, *flowchart* halaman *forgot password*, *flowchart* halaman *home*, *flowchart* halaman *add new data*, dan *flowchart* halaman *edit data*.

A. Flowchart Halaman *Sign Up*

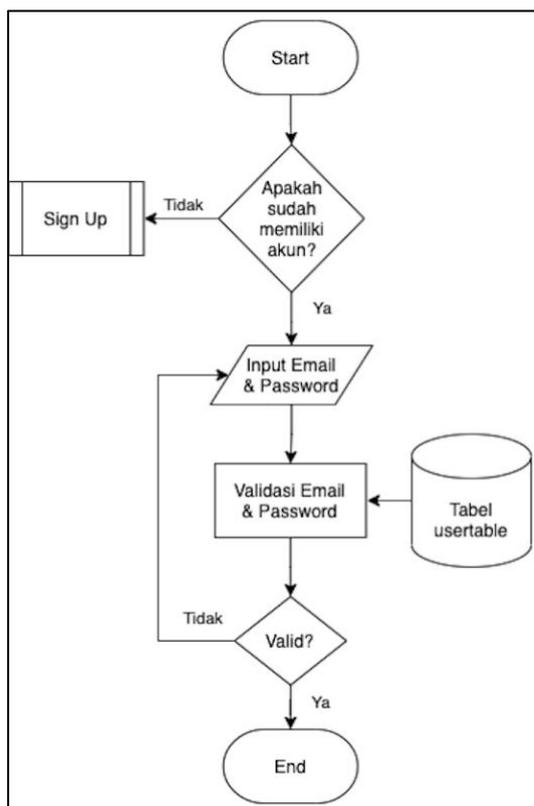
Gambar 3.2 merupakan *flowchart* dari halaman *sign up*. Apabila *user* belum memiliki akun, maka *user* diminta untuk melakukan *sign up* terlebih dahulu dengan memberikan input data berupa nama, *email*, *password* dan *confirm password*. Apabila *email* yang diinput merupakan *email* dari akun yang telah terdaftar di *database* maka pengguna harus kembali mengisi ulang dengan alamat *email* lain. Jika isi *password* dan *confirm password* tidak sama, maka pengguna diminta untuk mengisi ulang *field-field* tersebut. Apabila sudah berhasil diinput, sistem akan melakukan *generate* kode verifikasi dan data *user* serta kode verifikasi dimasukkan ke dalam *database*. Kemudian kode dikirim menuju *email user*. Lalu *user* diminta untuk memberikan input kode verifikasi dimana jika kode tersebut sesuai, maka status verifikasi akun di *database* akan berubah menjadi *verified*.



Gambar 3.2 Flowchart halaman *sign up*

B. Flowchart Halaman *Login*

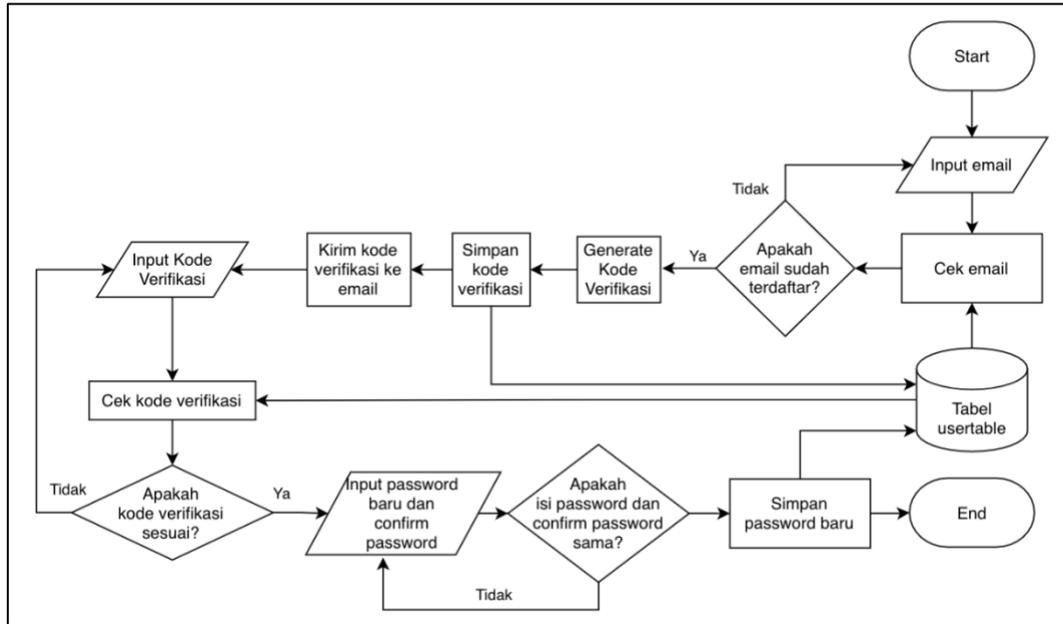
Gambar 3.3 merupakan *flowchart* dari halaman *login*. Apabila *user* telah memiliki akun, maka *user* diminta untuk melakukan *login* dengan memberikan input data berupa *email* dan *password*. Sistem akan melakukan validasi data inputan dengan data yang berada pada *database*. Jika tidak valid, *user* diminta untuk melakukan input kembali. Sedangkan apabila valid, *user* telah berhasil melakukan *login*.



Gambar 3.3 Flowchart aplikasi login

C. Flowchart Halaman *Forgot Password*

Gambar 3.4 merupakan *flowchart* dari halaman *forgot password*. Apabila *user* telah memiliki akun namun lupa *password*, maka *user* diarahkan menuju halaman *forgot password* dan diminta untuk memberikan input data berupa *email*. Kemudian sistem akan melakukan pengecekan apakah email sudah terdaftar atau belum. Jika sudah terdaftar maka sistem akan melakukan generate kode verifikasi dan menyimpan kode tersebut ke dalam database. Kemudian kode dikirim menuju *email user*. Lalu *user* diminta untuk memberikan input kode verifikasi dimana jika kode tersebut sesuai maka *user* dapat membuat *password* baru dan mengisi *confirm password*. Apabila isi *password* dan *confirm password* tidak sama, maka pengguna diminta untuk mengisi ulang *field-field* tersebut. Jika sudah sesuai, sistem akan menyimpan *password* baru ke dalam *database*.

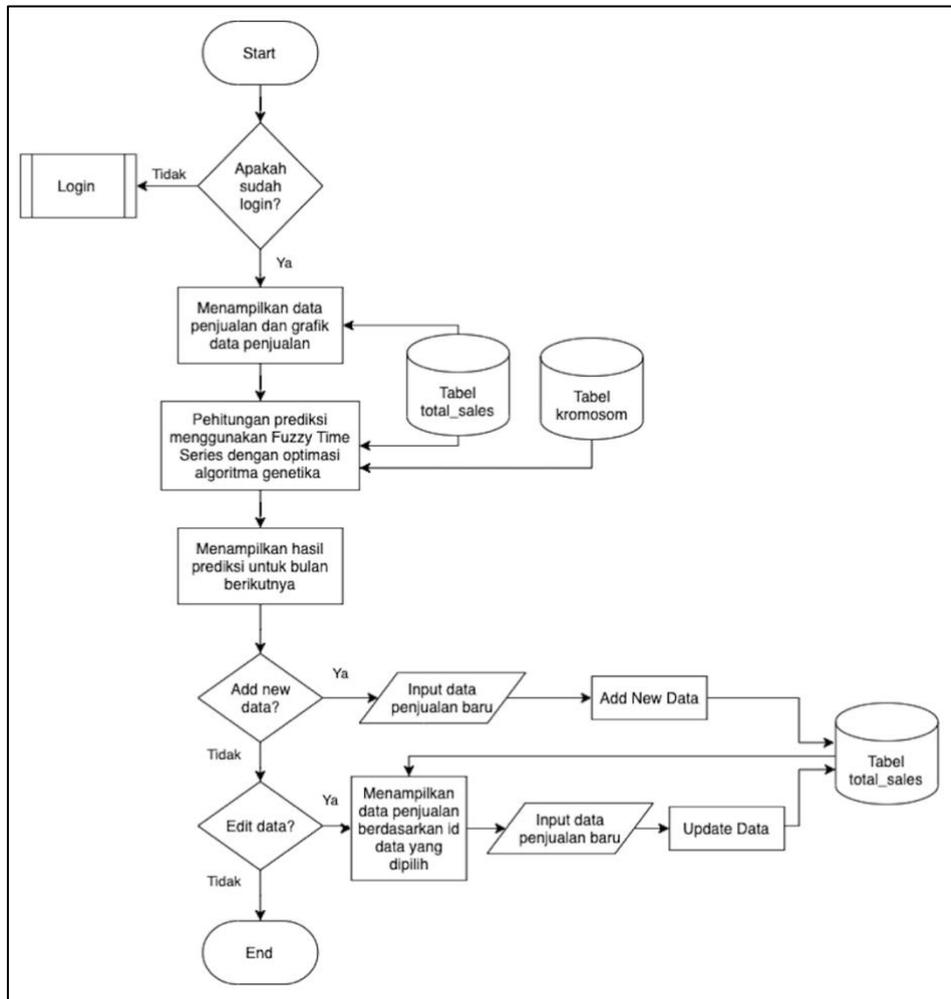


Gambar 3.4 Flowchart aplikasi forgot password

D. Flowchart Halaman Home

Gambar 3.5 merupakan *flowchart* dari halaman *home*. Pada halaman *home*, *user* dapat melihat data historis, grafik garis dan nilai prediksi total penjualan untuk bulan berikutnya. Data historis dan grafik garis ditampilkan berdasarkan data yang ada pada di *database*. Sedangkan prediksi total penjualan untuk bulan berikutnya didapatkan dari proses perhitungan dengan menggunakan *fuzzy time series* yang dioptimasi algoritma genetika berdasarkan data penjualan dan kromosom yang diakses dari *database*. *User* juga dapat menambahkan data historis berupa angka total penjualan bulan selanjutnya dari bulan terakhir yang ada pada data historis di *database*. Setelah data berhasil ditambahkan, data disimpan ke dalam *database* dan data akan otomatis muncul pada tabel yang ada di halaman *home* dan sistem akan menampilkan hasil perhitungan prediksi pada bulan selanjutnya berdasarkan data historis yang terbaru. Selain itu, *user* dapat mengubah data historis total penjualan per bulan yang ada pada *database*. Apabila perubahan

berhasil dilakukan, maka sistem melakukan *update* data di *database* dan data pada tabel serta prediksi penjualan yang ada pada halaman *home* akan berubah mengikuti data historis terbaru.

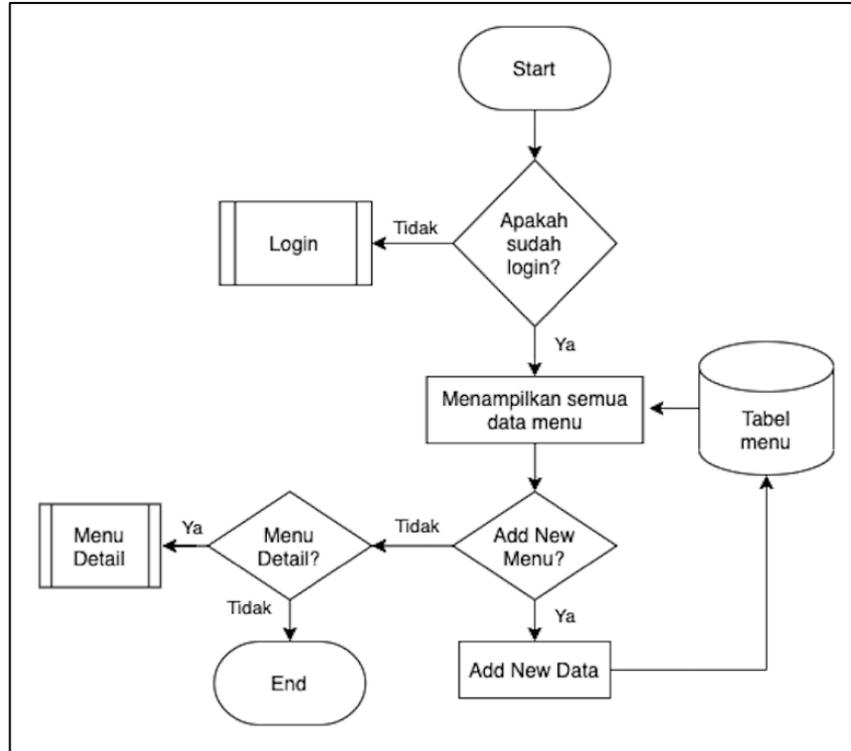


Gambar 3.5 Flowchart aplikasi home

E. Flowchart Halaman Menu

Gambar 3.6 merupakan *flowchart* dari halaman menu. Pada halaman menu, *user* dapat melihat seluruh data menu yang telah ditambahkan ke *database*. Selain itu *user* dapat menambah data menu baru dengan mengisi nama menu. Setelah berhasil melakukan input, data akan disimpan ke dalam *database* dan menu baru

akan otomatis muncul pada halaman menu. User juga dapat menuju halaman *menu detail* dengan cara memilih menu yang diinginkan.

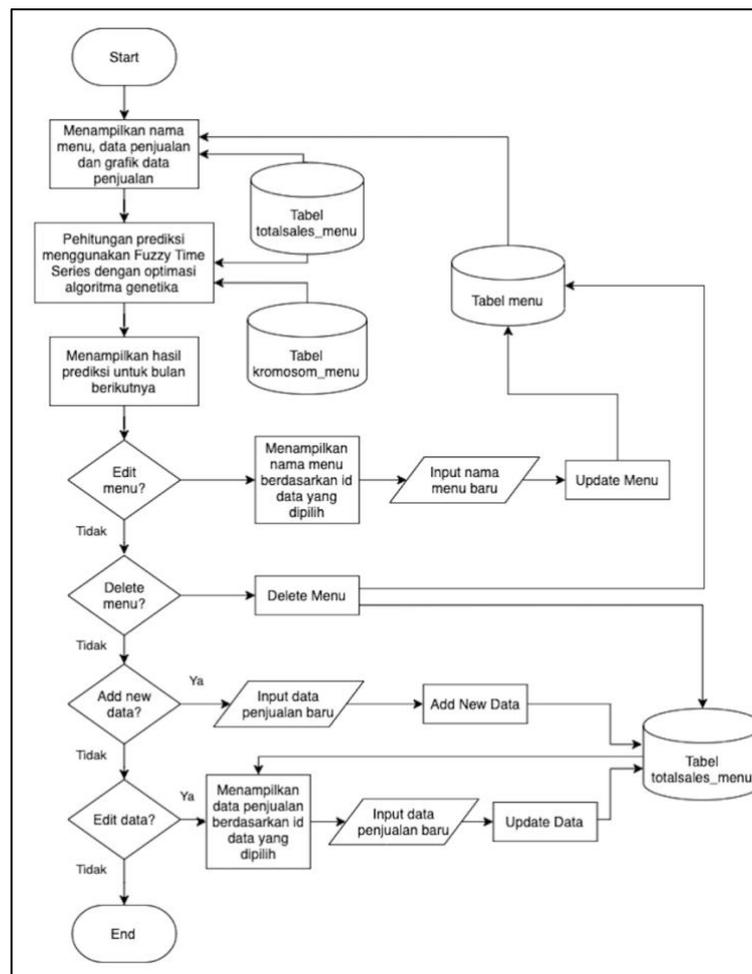


Gambar 3.6 *Flowchart* aplikasi menu

F. *Flowchart* Halaman *Menu Detail*

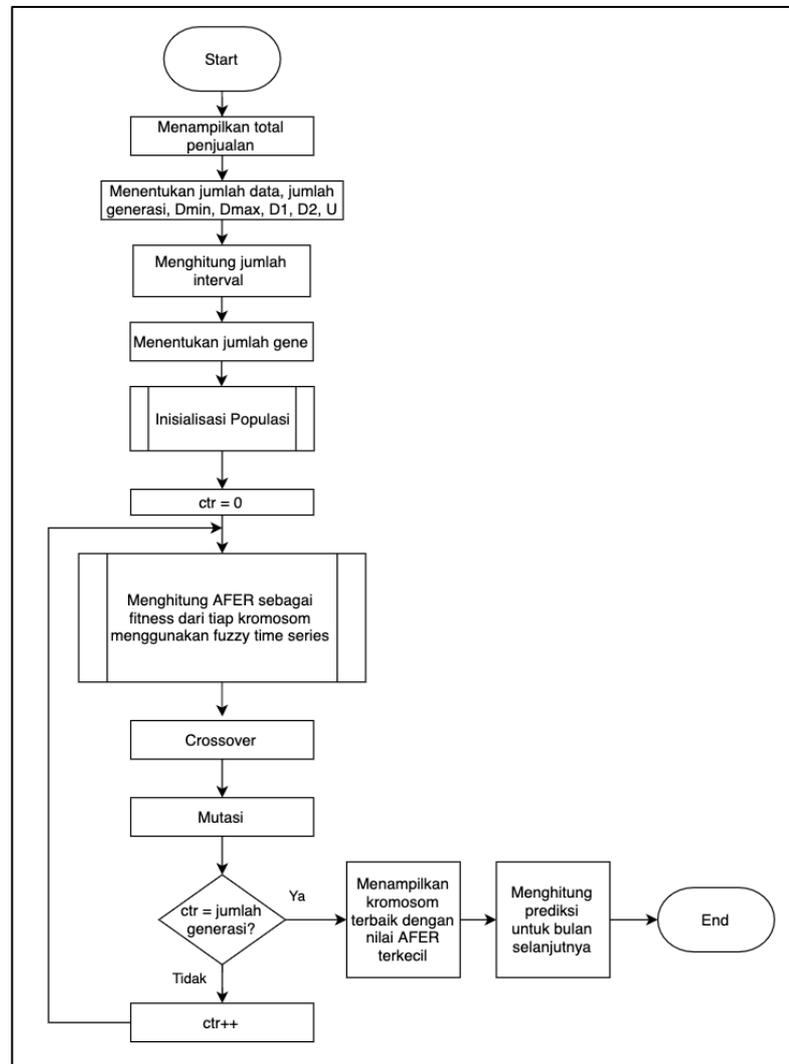
Gambar 3.7 merupakan flowchart halaman *menu detail*. Pada halaman ini, *user* dapat melihat data historis, grafik garis dan nilai prediksi total penjualan dari menu tersebut untuk bulan berikutnya. Data historis dan grafik garis ditampilkan berdasarkan data yang ada pada *database*. Sedangkan prediksi total penjualan untuk bulan berikutnya didapatkan dari proses perhitungan dengan menggunakan *fuzzy time series* yang dioptimasi algoritma genetika berdasarkan data penjualan dari menu tersebut yang diakses dari *database*. *User* dapat melakukan edit nama menu yang ada pada *database*. Apabila perubahan berhasil dilakukan, maka sistem akan melakukan *update* pada *database* dan nama menu di halaman Menu serta *Menu*

Detail akan otomatis berubah. *User* juga dapat menghapus menu yang ada. Selain itu, *user* dapat menambahkan data historis berupa angkat total penjualan bulan selanjutnya dari bulan terakhir yang ada pada data historis di database. Setelah data berhasil ditambahkan, data disimpan ke dalam database dan data akan otomatis muncul pada tabel yang ada di halaman *menu detail* dan sistem akan menampilkan hasil perhitungan prediksi pada bulan selanjutnya berdasarkan data historis yang terbaru. Selain itu, *user* dapat mengubah data historis total penjualan per bulan yang ada pada *database*. Apabila perubahan berhasil dilakukan, maka sistem melakukan *update* data di database dan data pada tabel serta prediksi penjualan yang ada pada halaman *menu detail* akan berubah mengikuti data historis terbaru.



Gambar 3.7 Flowchart aplikasi menu detail

G. *Flowchart* Perhitungan Prediksi menggunakan *Fuzzy Time Series* dengan optimasi Algoritma Genetika

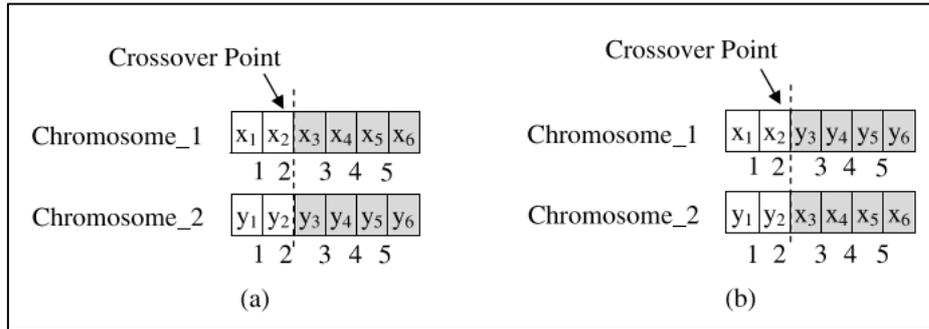


Gambar 3.8 *Flowchart* perhitungan prediksi menggunakan *fuzzy time series* dengan optimasi algoritma genetika

Berikut merupakan langkah-langkah perhitungan prediksi total penjualan menggunakan *Fuzzy Time Series* dengan optimasi Algoritma Genetika:

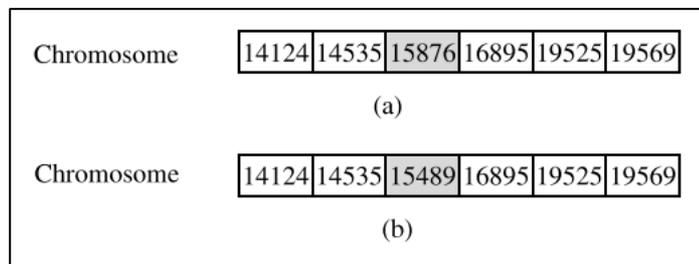
- a. Menampilkan total penjualan yang didapat dari *database*

- b. Menentukan beberapa parameter yaitu jumlah data, jumlah generasi, D_{min} , D_{max} , $D1$, $D2$, U (*universe*). Dimana D_{min} adalah nilai terkecil dan D_{max} adalah nilai terbesar dari keseluruhan data historis. Sedangkan $D1$, $D2$ merupakan nilai konstanta sembarang serta *universe* adalah himpunan semesta dan *universe* didapat dengan rumus 2.1.
- c. Menentukan jumlah interval menggunakan rumus Sturges dengan mengacu pada rumus 2.2 dan Dimana n = jumlah data historis.
- d. Tentukan jumlah gene. Jumlah gene didapat dari $K-1$ dimana K adalah jumlah interval.
- e. Kemudian dilakukan proses inisialisasi populasi yang akan dijelaskan lebih detail pada *flowchart* selanjutnya.
- f. Definsikan variabel *counter* = 0.
- g. Hitung AFER tiap kromosom menggunakan *fuzzy time series*. Hasil AFER akan digunakan sebagai nilai *fitness*. Detail akan dijelaskan pada *flowchart* selanjutnya.
- h. *Crossover*. Jenis *crossover* yang dipakai yaitu *one-point crossover*, dimana hanya ada satu *cross point* acak yang digunakan. Misal *cross point* = 2, dimana persilangan anggota antara 2 kromosom akan dimulai dari gene ketiga sampai terakhir sehingga didapatkan kromosom *child*.



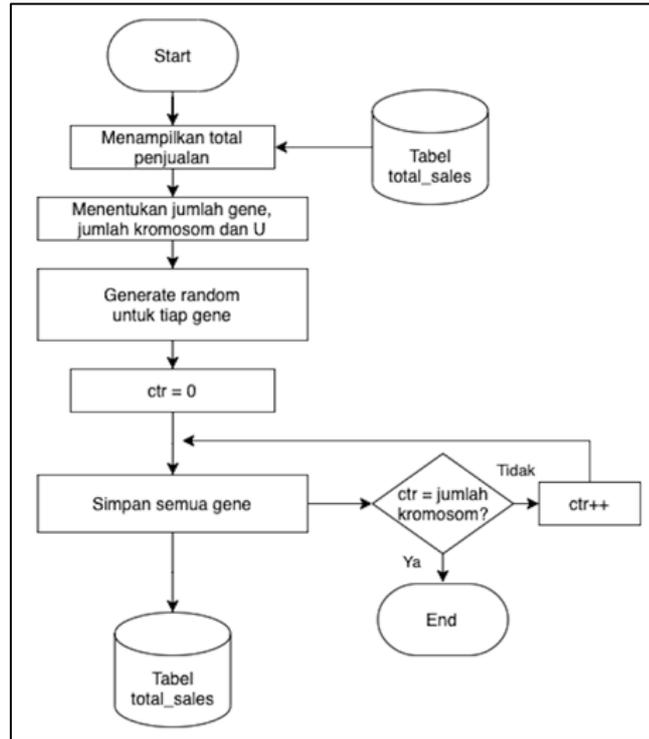
Gambar 3.9 (a) Sebelum *crossover* (b) Sesudah *crossover*

- i. Mutasi. Pilih salah satu generasi secara acak, lalu diubah secara acak dengan *range* yang sudah ditentukan berdasarkan *mutation rate*.



Gambar 3.10 (a) Sebelum mutasi (b) Sesudah mutasi

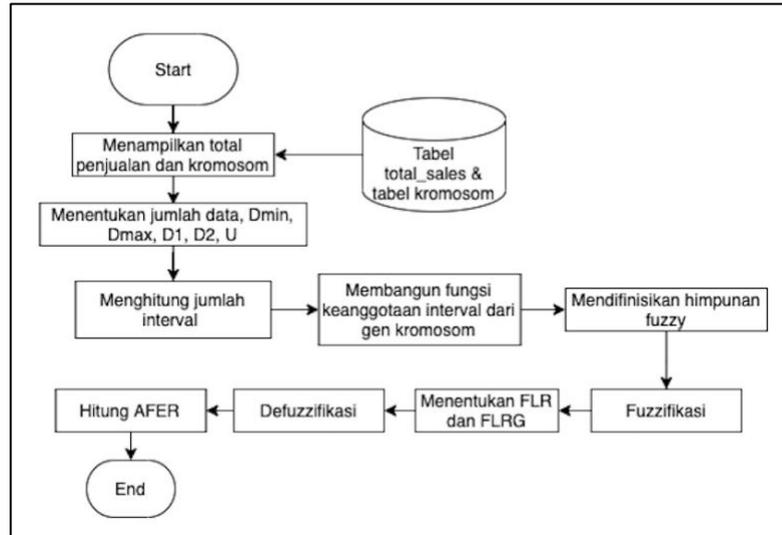
- j. Lakukan perulangan dengan jumlah generasi yang telah ditentukan. Perulangan dilakukan dengan tujuan untuk mendapatkan hasil AFER terkecil.
 - k. Untuk menentukan prediksi total penjualan pada bulan berikutnya, gunakan rumus $t - 1$ dimana t adalah waktu sehingga prediksi untuk bulan berikutnya merupakan prediksi pada bulan terakhir yang ada pada data historis.
- H. *Flowchart* Inisialisasi Populasi



Gambar 3.11 *Flowchart* inialisasi populasi

Berikut merupakan langkah-langkah melakukan inialisasi populasi:

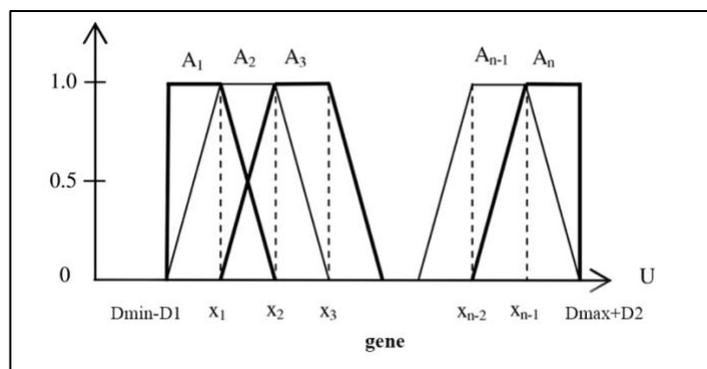
- a. Menampilkan total penjualan yang didapat dari *database*
- b. Menentukan jumlah gene, jumlah kromosom dan U yang didapat dari proses sebelumnya.
- c. Sistem melakukan *generate random* angka untuk tiap gene dengan range berdasarkan batas bawah dan batas atas dari U.
- d. Definsikan variabel *counter* = 0
- e. Simpan semua gene yang telah terbentuk ke dalam *database*
- f. Lakukan perulangan sampai dengan jumlah kromosom yang telah ditentukan sebelumnya
- g. *Flowchart* Menghitung AFER tiap kromosom menggunakan *Fuzzy Time Series*



Gambar 3.12 Flowchart menghitung AFER tiap kromosom menggunakan *fuzzy time series*

Berikut merupakan langkah-langkah melakukan Menghitung AFER tiap kromosom menggunakan *Fuzzy Time Series*:

- a. Menampilkan total penjualan dan total kromosom yang didapat dari proses sebelumnya
- b. Menentukan jumlah data, Dmin, Dmax, D1, D2 dan U serta jumlah interval yang juga didapat dari proses sebelumnya
- c. Membangun fungsi keanggotaan interval dari gen kromosom



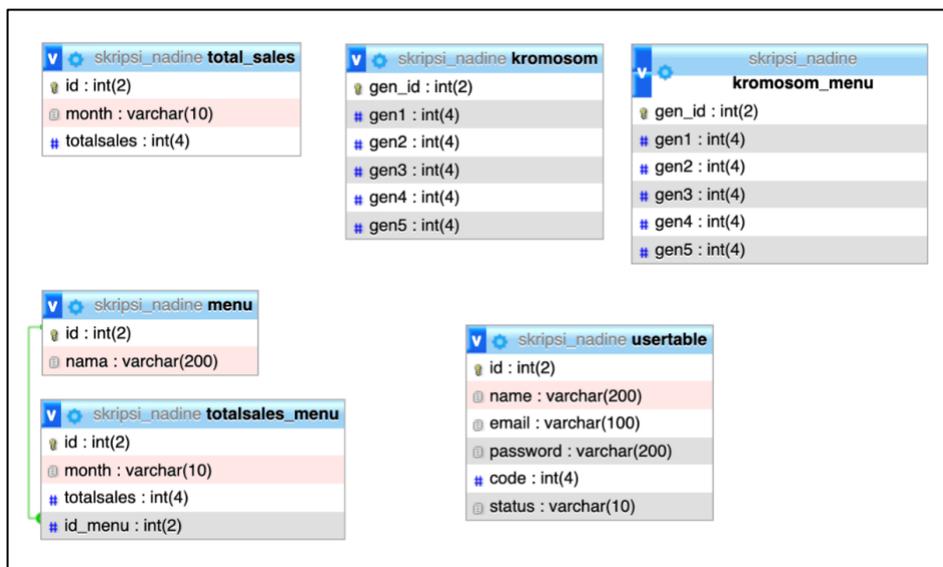
Gambar 3.13 Fungsi keanggotaan yang dibangun dari gene

- d. Mendefinisikan himpunan *fuzzy*, melakukan fuzzifikasi, meenentukan FLR dan FLRG serta Defuzzifikasi dilakukan seperti langkah-langkah perhitungan prediksi menggunakan *fuzzy time series* yang telah dijabarkan sebelumnya.
- e. Kemudian lakukan perhitungan AFER dengan rumus 2.6 untuk menghitung tingkat kesalahan yang dihasilkan oleh sistem.

3.2.3 Skema Basis Data

Skema Basis Data untuk memudahkan implementasi dalam melakukan relasi antar tabel dengan menunjukkan relasi entitas *primary key* dengan entitas *foreign key* tabel lainnya yang memiliki hubungan satu sama lain.

Gambar 3.11 merupakan skema basis data yang digunakan. Terdapat 5 tabel yang digunakan, antara lain *total_sales*, *menu*, *totalsales_menu*, *usertable*, *kromosom*, *kromosom_menu*. Entitas *id_menu* pada tabel *totalsales_menu* merupakan *foreign key* yang berasal dari tabel *menu*. Hal tersebut dilakukan untuk mendapatkan menu yang ada pada data total penjualan menu.



Gambar 3.14 Skema basis data

3.2.4 Struktur Tabel Basis Data

A. Tabel total_sales

Tabel total_sales merupakan tabel yang digunakan untuk menyimpan data historis dari total penjualan Toko Kue Kris.

Tabel 3.1 Tabel total_sales

No	Nama	Tipe Data	Panjang
1	id	Int	2
2	month	Varchar	10
3	totalsales	Int	4

Tabel 3.1 merupakan struktur dari tabel total_sales pada *database* yang digunakan dalam aplikasi yang dibangun. Total penjualan perbulan dari keseluruhan penjualan Toko Kue Kris disimpan dalam tabel ini.

B. Tabel menu

Tabel menu merupakan tabel yang digunakan untuk menyimpan data menu yang ada di Toko Kue Kris.

Tabel 3.2 Tabel menu

No	Nama	Tipe Data	Panjang
1	id	Int	2
2	nama	Int	200

Tabel 3.2 merupakan tabel yang berisi informasi mengenai menu yang ada di Toko Kue Kris.

C. Tabel totalsales_menu

Tabel totalsales_menu merupakan tabel yang digunakan untuk menyimpan data historis dari total penjualan dari setiap menu yang ada di Toko Kue Kris

Tabel 3.3 Tabel totalsales_menu

No	Nama	Tipe Data	Panjang
1	id	Int	2
2	month	Varchar	10
3	totalsales	Int	4
4	id_menu	Int	2

Tabel 3.3 merupakan struktur dari tabel totalsales_menu pada database yang digunakan dalam aplikasi yang dibangun. Pada tabel ini, id_menu merupakan *foreign key* dari tabel menu. Hal tersebut dilakukan untuk mendapatkan menu yang ada pada data total penjualan menu.

D. Tabel kromosom

Tabel kromosom merupakan tabel yang digunakan untuk menyimpan data kromosom hasil dari inisialisasi populasi yang dilakukan dalam melakukan perhitungan prediksi total penjualan Toko Kue Kris.

Tabel 3.4 Tabel kromosom

No	Nama	Tipe Data	Panjang
1	gen_id	Int	2
2	gen1	Int	4
3	gen2	Int	4
4	gen3	Int	4
5	gen4	Int	4
6	gen5	Int	4

Tabel 3.4 merupakan struktur dari tabel kromosom pada *database* yang digunakan dalam aplikasi yang dibangun. Setiap gen yang ada pada tabel akan digunakan sebagai sub himpunan semesta sehingga didapatkan nilai AFER dari setiap kromosom yang ada pada *database*.

E. Tabel kromosom_menu

Tabel kromosom_menu merupakan tabel yang digunakan untuk menyimpan data kromosom hasil dari inisialisasi populasi yang dilakukan dalam melakukan perhitungan prediksi total penjualan setiap menu yang ada di Toko Kue Kris.

Tabel 3.5 Tabel kromosom_menu

No	Nama	Tipe Data	Panjang
1	gen_id	Int	2
2	gen1	Int	4
3	gen2	Int	4
4	gen3	Int	4
5	gen4	Int	4
6	gen5	Int	4

Tabel 3.5 merupakan struktur dari tabel kromosom_menu pada *database* yang digunakan dalam aplikasi yang dibangun. Setiap gen yang ada pada tabel akan digunakan sebagai sub himpunan semesta sehingga didapatkan nilai AFER dari setiap kromosom yang ada pada *database*.

F. Tabel usertable

Tabel usertable merupakan tabel yang digunakan untuk menyimpan informasi user yang telah melakukan pendaftaran akun.

Tabel 3.6 Tabel usertable

No	Nama	Tipe Data	Panjang
1	id	Int	2
2	name	Varchar	200
3	email	Varchar	100
4	password	Varchar	200
5	code	Int	4
6	status	Varchar	10

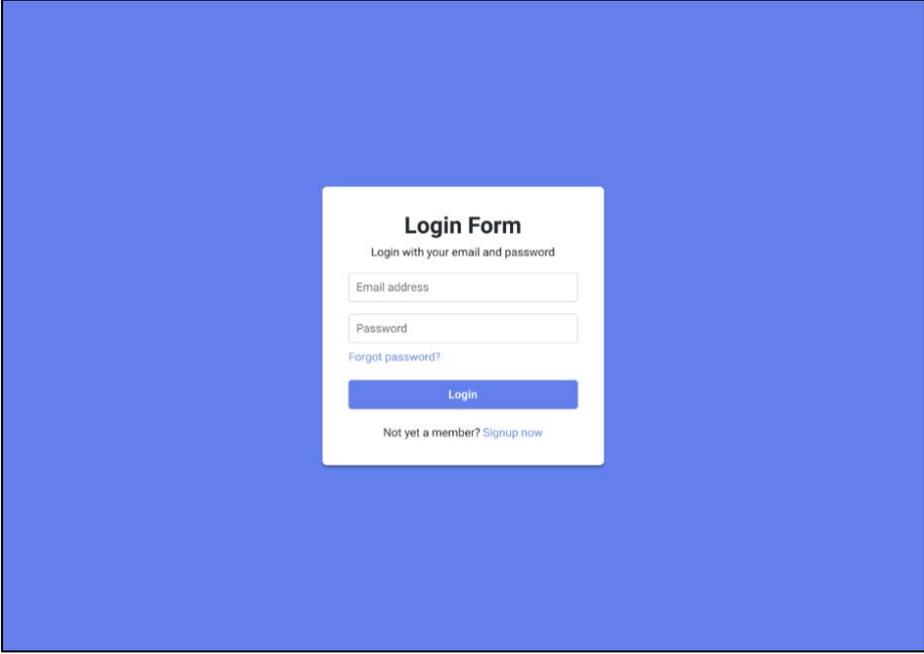
Tabel 3.6 merupakan tabel berisi informasi *user* seperti *email* dan *password* yang digunakan sebagai *login credential*. Entitas *code* digunakan sebagai kode verifikasi dalam melakukan *sign up* ataupun *forgot password*. Sedangkan entitas *status* digunakan sebagai informasi bahwa akun telah terverifikasi melalui email.

3.2.5 Desain Antarmuka Sistem

Desain antarmuka sistem dirancang sesuai dengan fungsionalitas dari setiap halaman aplikasi prediksi total penjualan Toko Kue Kris dengan memperhatikan penyampaian kepada pengguna sehingga dapat lebih mudah dipahami. Berikut merupakan desain antarmuka sistem setiap halaman pada aplikasi.

A. Desain Antarmuka Halaman *Login*

Halaman *login* akan ditampilkan pada saat pertama kali *user* memasuki aplikasi prediksi penjualan Toko Kue Kris.



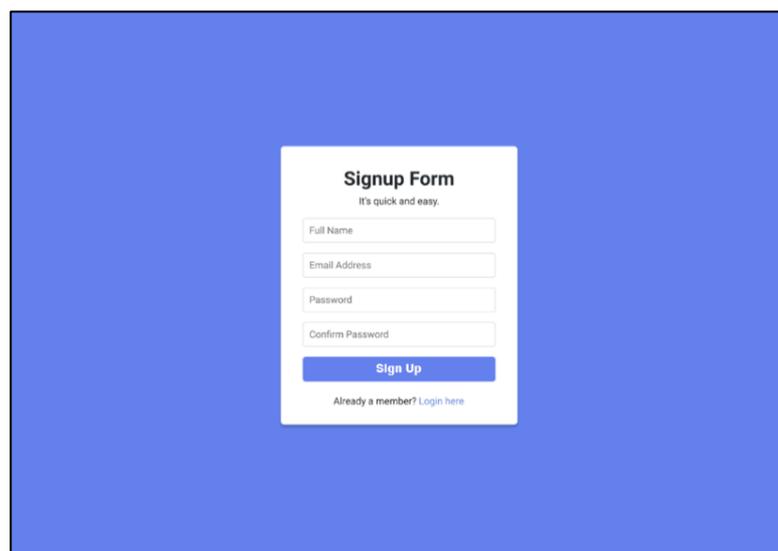
The image shows a login form centered on a blue background. The form is white and contains the following elements: a title 'Login Form', a subtitle 'Login with your email and password', an input field for 'Email address', an input field for 'Password', a link for 'Forgot password?', a blue 'Login' button, and a link for 'Not yet a member? Signup now'.

Gambar 3.12 Desain antarmuka halaman *login*

Form seperti pada Gambar 3.12 akan ditampilkan pada halaman *login*, *form* tersebut berguna sebagai tahap awal sebelum *user* dapat menggunakan aplikasi prediksi penjualan Toko Kue Kris. Jika *user* sudah terdaftar maka *user* dapat langsung mengisi *form*, *form* tersebut memiliki dua buah *field* yang harus diisi oleh *user* yaitu email *address* dan *password*. Setelah *user* mengisi *form* pada halaman *login*, *user* dapat menekan tombol “*Login*” untuk masuk ke dalam aplikasi dan mengakses halaman *home*. Jika *user* belum pernah mendaftarkan dirinya pada aplikasi prediksi penjualan Toko Kue Kris, *user* dapat mendaftarkannya terlebih dahulu dengan menekan tombol link dengan tulisan “*Signup now*” yang akan memindahkan *user* ke halaman *sign up*. Jika *user* lupa akan *password* yang didaftarkan sebelumnya, *user* dapat menekan tombol link dengan tulisan “*Forgot password*” yang akan memindahkan *user* ke halaman *forgot password*.

B. Desain Antarmuka Halaman *Sign Up*

Halaman *sign up* akan berguna untuk melakukan pendaftaran akun. Akun yang didaftarkan dapat berguna untuk *login* masuk ke aplikasi prediksi penjualan Toko Kue Kris.

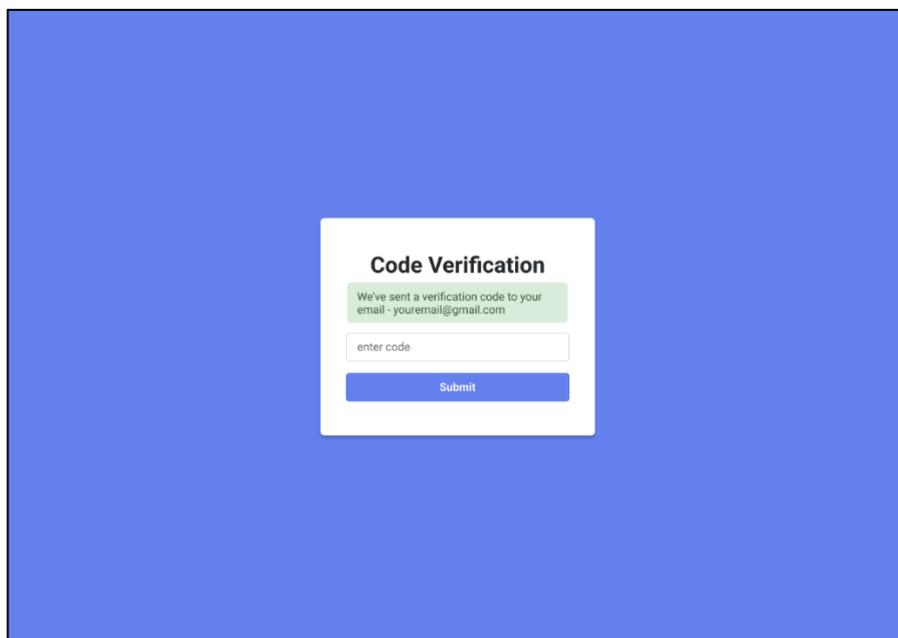
The image shows a 'Signup Form' centered on a blue background. The form is white and contains the following elements: a title 'Signup Form' with the subtitle 'It's quick and easy.', four input fields labeled 'Full Name', 'Email Address', 'Password', and 'Confirm Password', a blue 'Sign Up' button, and a link 'Already a member? Login here'.

Gambar 3.13 Desain antarmuka halaman *sign up*

Form seperti pada Gambar 3.13 akan ditampilkan pada halaman *sign up*, *form* tersebut memiliki 4 buah *field* yang harus diisi oleh *user* yaitu *full name*, *email address*, *password*, dan *confirm password*. Setelah *user* mengisi *form* pada halaman *sign up*, *user* dapat mendaftarkan akun-nya dengan menekan tombol “*Sign Up*”.

C. Desain Antarmuka Halaman Kode Verifikasi – *Sign Up*.

Halaman kode verifikasi *sign up* digunakan untuk melakukan verifikasi akun dengan cara mengisi kode verifikasi yang dikirim oleh sistem melalui *email*.



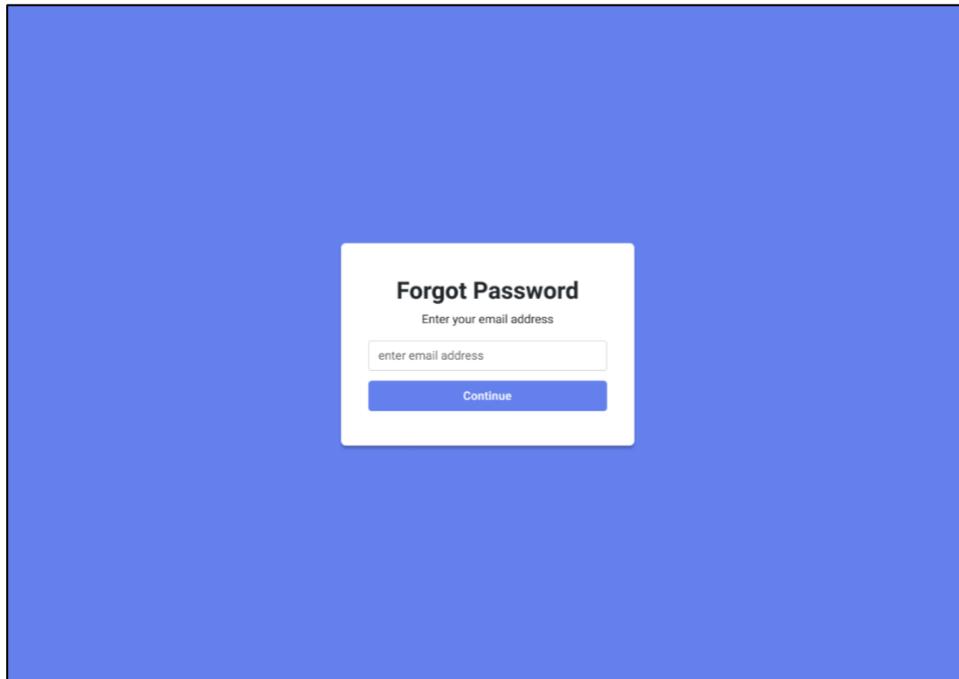
The image shows a central white form box on a solid blue background. The form is titled "Code Verification" in bold black text. Below the title, there is a green message box containing the text "We've sent a verification code to your email - youremail@gmail.com". Underneath the message is a white input field with the placeholder text "enter code". At the bottom of the form is a blue button with the white text "Submit".

Gambar 3.14 Desain antarmuka halaman kode verifikasi - *sign up*

Halaman kode verifikasi – *sign up* menampilkan *form* seperti pada Gambar 3.14, *form* tersebut hanya memiliki satu buah *field* untuk diisi dengan kode verifikasi yang telah diterima oleh *user*. Setelah *user* mengisi *form* pada halaman kode verifikasi – *sign up*, *user* dapat menekan tombol “*Submit*” untuk memverifikasi akun yang telah didaftarkan oleh *user*. *User* dapat melakukan *login* menggunakan akun tersebut jika sudah terverifikasi.

D. Desain Antarmuka Halaman *Forgot Password*

Halaman *forgot password* merupakan tahap awal yang dapat dilakukan oleh *user* jika melupakan *password* yang dimilikinya dan ingin membuat *password* baru.

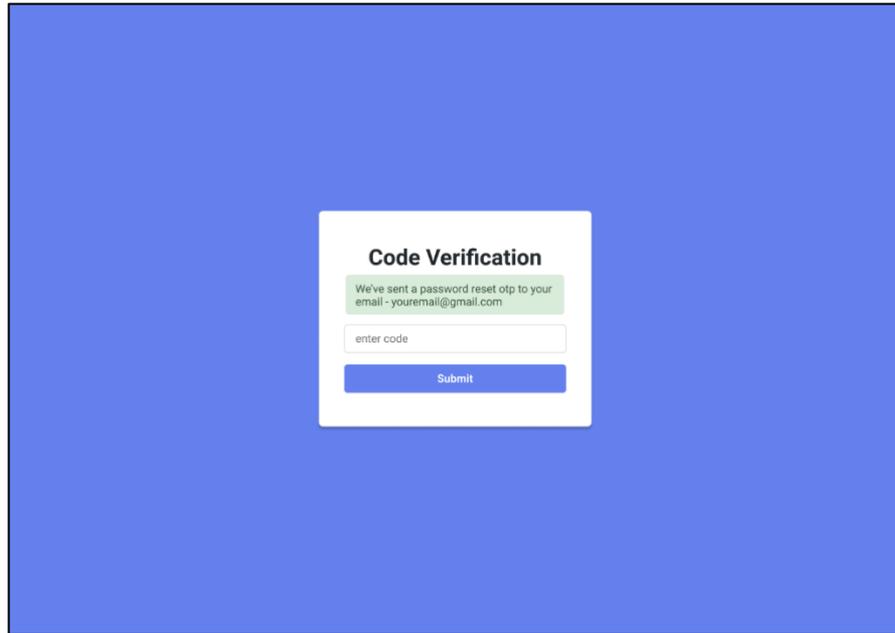


Gambar 3.15 Desain antarmuka halaman *forgot password*

Gambar 3.15 menunjukkan tampilan yang dimiliki oleh halaman *forgot password*, pada halaman ini terdapat *form* dengan satu *field* yang meminta *user* untuk mengisi *field* tersebut dengan alamat email pada akun *user* tersebut. Setelah *user* mengisi *form* pada halaman *forgot password*, *user* dapat menekan tombol “*Continue*” melanjutkan ke halaman selanjutnya.

E. Desain Antarmuka Halaman Kode Verifikasi – *Forgot Password*

Halaman kode verifikasi – *forgot password* merupakan tahap lanjutan dari halaman *forgot password*. *User* diminta untuk melakukan verifikasi akun dengan cara mengisi kode verifikasi yang telah dikirim oleh sistem melalui *email*.

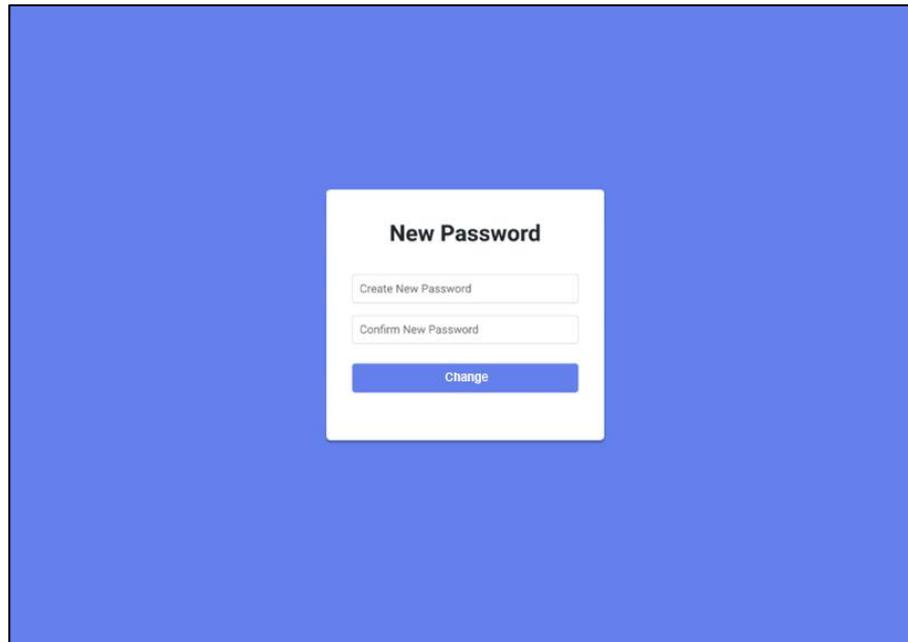


Gambar 3.16 Desain antarmuka halaman kode verifikasi – *forgot password*

Halaman kode verifikasi – *forgot password* menampilkan form seperti pada Gambar 3.16, *form* tersebut hanya memiliki satu buah *field* untuk diisi dengan kode verifikasi yang telah diterima oleh *user*. Setelah *user* mengisi *form* tersebut, *user* dapat menekan tombol “*Submit*” untuk memverifikasi penggantian *password*. *User* akan dipindahkan ke halaman *new password* setelah *user* menekan tombol “*Submit*”.

F. Desain Antarmuka Halaman *New Password*

Halaman *new password* merupakan tahap akhir dari proses *Forgot Password*, halaman ini berguna untuk mendaftarkan *password* baru pada akun yang dimiliki oleh *user*.

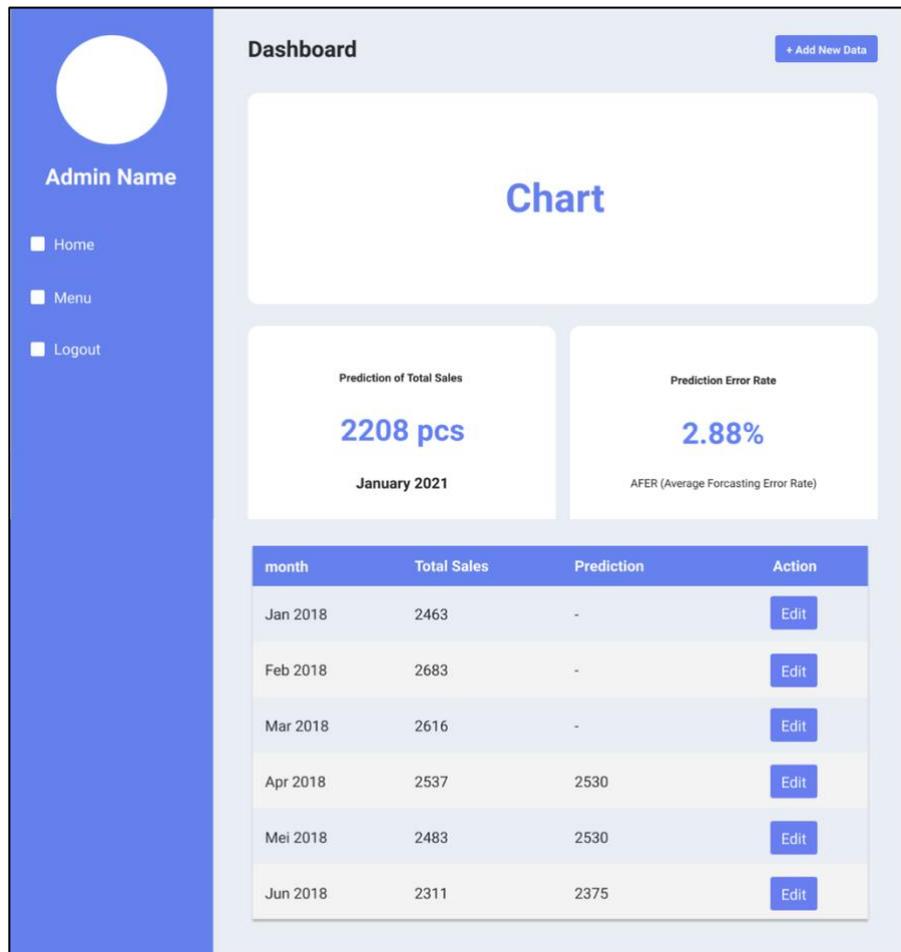


Gambar 3.17 Desain antarmuka halaman *new password*

Gambar 3.17 menunjukkan tampilan yang dimiliki oleh halaman *new password*, pada halaman ini terdapat *form* dengan dua *field* yang meminta *user* untuk mengisi *field* tersebut dengan *password* baru yang ingin didaftarkan oleh *user*. Setelah *user* mengisi *form* tersebut, *user* dapat menekan tombol “*Change*” yang akan mendaftarkan *password* baru untuk akun yang dimiliki oleh *user* dan menyelesaikan proses *Forgot Password*. *User* akan dipindahkan ke Halaman *Login* setelah menekan tombol “*Change*”.

G. Desain Antarmuka Halaman *Home*

Halaman *home* merupakan halaman utama pada aplikasi prediksi total penjualan Toko Kue Kris, berbagai macam *menu* dapat diakses pada halaman ini.



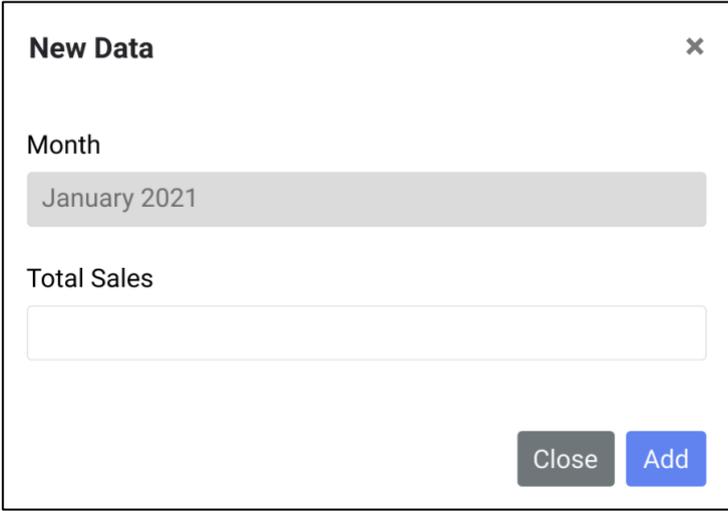
Gambar 3.18 Desain antarmuka halaman *home*

Halaman *home* memiliki tampilan seperti pada Gambar 3.18, terdapat *menu* navigasi pada bagian kiri pada halaman *home* yang menampilkan *avatar* dan nama dari *user* dan 3 buah *menu* yaitu *home*, *menu*, dan *logout*. Pada bagian *dashboard* pada halaman *home* menampilkan *chart* dan prediksi dari perhitungan sistem aplikasi prediksi total penjualan Toko Kue Kris. Bagian *dashboard* juga menampilkan tabel berisi *data* penjualan untuk tiap tahunnya, *user* dapat mengubah data tersebut dengan menekan tombol “*Edit*” yang terletak pada tabel tersebut. *User*

juga dapat menambahkan *data* baru dengan menekan tombol “*Add New Data*” yang terletak pada kanan atas halaman *home*.

H. Desain Antarmuka Halaman Add New Data - Home

Halaman *add new data – home* akan ditampilkan jika *user* menekan tombol “*Add New Data*” pada halaman *home*.



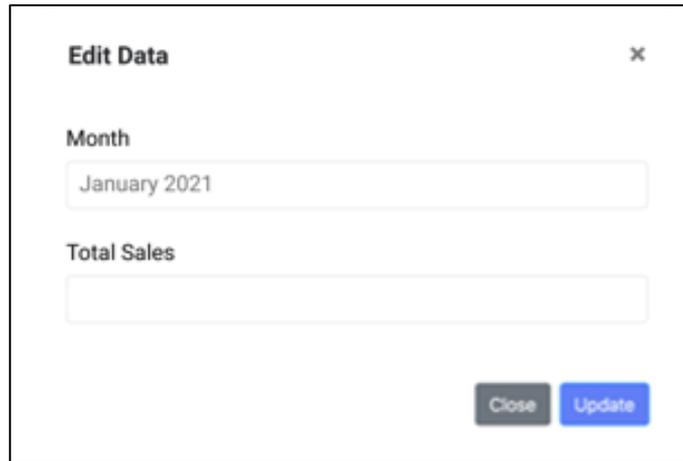
The image shows a modal window titled "New Data" with a close button (X) in the top right corner. Below the title, there are two input fields. The first is labeled "Month" and has a dropdown menu showing "January 2021". The second is labeled "Total Sales" and is an empty text input field. At the bottom right of the modal, there are two buttons: "Close" (grey) and "Add" (blue).

Gambar 3.19 Desain antarmuka halaman *add new data - home*

Halaman *add new data – home* memiliki tampilan seperti pada Gambar 3.19, halaman ini memiliki *form* yang meminta *user* untuk mengisi total sales pada bulan yang telah ditentukan. *User* dapat menambahkan *data* baru tersebut dengan menekan tombol “*Add*”, *user* juga bisa keluar dari halaman ini jika tidak ingin menambahkan data baru dengan cara menekan tombol dengan ikon silang pada kanan atas halaman ini atau dengan cara menekan tombol “*Close*”.

I. Desain Antarmuka Halaman *Edit Data - Home*

Halaman *edit data - home* akan ditampilkan jika *user* menekan tombol “*Edit*” pada halaman *home*.

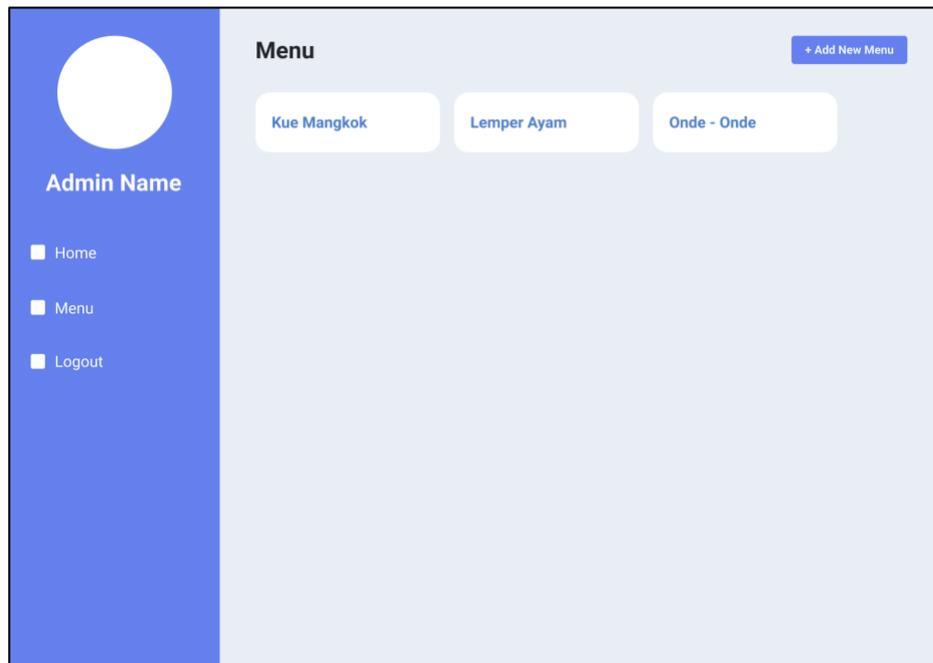


Gambar 3.20 Desain antarmuka halaman *edit data - home*

Halaman *edit data – home* memiliki tampilan seperti pada Gambar 3.20, halaman ini memiliki *form* mengenai informasi data tersebut yang memiliki 2 *field* yaitu *month* dan total sales pada *data* tersebut. *User* dapat mengubah *field* tersebut dan menyimpan perubahannya dengan menekan tombol “*Update*”, *user* juga bisa keluar dari halaman ini jika tidak ingin menambahkan *data* baru dengan cara menekan tombol dengan ikon silang pada kanan atas halaman ini atau dengan cara menekan tombol “*Close*”.

J. Desain Antarmuka Halaman *Menu*

Halaman *menu* merupakan *menu* lain yang dapat diakses pada aplikasi prediksi total penjualan Toko Kue Kris.



Gambar 3.21 Desain antarmuka halaman *menu*

Halaman *menu* memiliki tampilan seperti pada Gambar 3.21, halaman ini menampilkan daftar *menu* yang terdapat pada aplikasi prediksi total penjualan Toko Kue Kris. *User* dapat melihat *detail* tiap *menu*-nya dengan cara menekan *card menu* kue tersebut. *User* dapat menambahkan *menu* baru dengan menekan tombol “*Add New Menu*” pada kanan atas halaman *menu*.

K. Desain Antarmuka Halaman *Add New Menu*

Halaman *add new menu* akan ditampilkan jika *user* menekan tombol “*Add New Menu*” pada halaman *menu*.

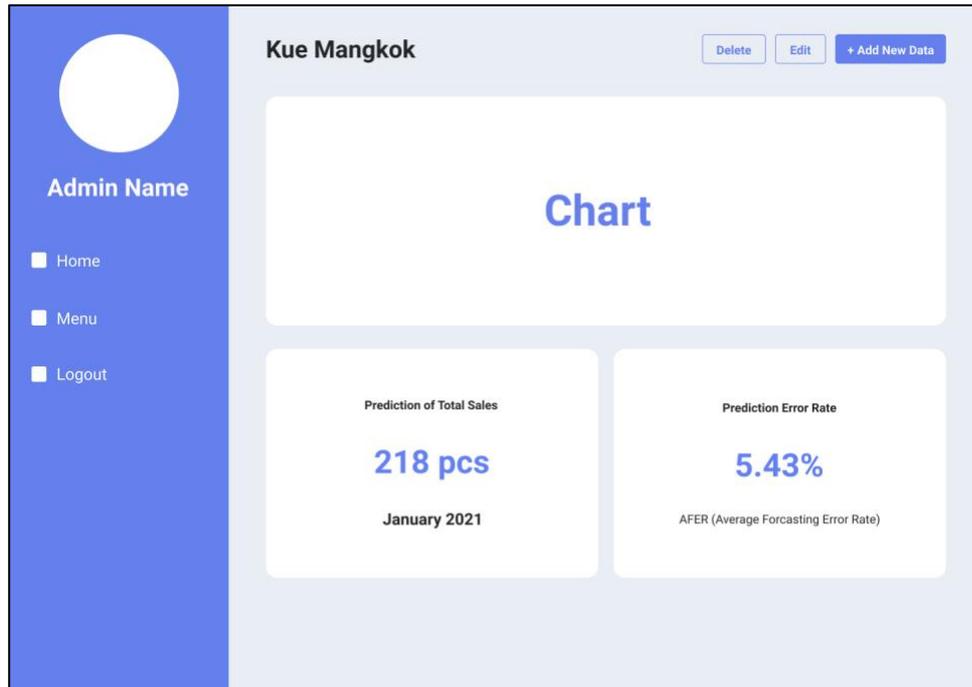
The image shows a modal window titled "New Data" with a close button (X) in the top right corner. Below the title is a label "Nama Menu" followed by a text input field. At the bottom right of the modal, there are two buttons: a grey "Close" button and a blue "Add" button.

Gambar 3.22 Desain antarmuka halaman *add new menu*

Halaman *add new menu* memiliki tampilan seperti pada Gambar 3.22, halaman ini memiliki *form* yang meminta *user* untuk mengisi nama *menu*. *User* dapat menekan tombol “*Add*” jika telah mengisi *form* dan ingin menambahkan *menu* tersebut. *User* juga bisa keluar dari halaman ini jika tidak ingin menambahkan data baru dengan cara menekan tombol dengan ikon silang pada kanan atas halaman ini atau dengan cara menekan tombol “*Close*”.

L. Desain Antarmuka Halaman *Menu Detail*

Halaman *menu detail* menampilkan informasi mendetail mengenai *menu* yang telah dipilih/ditekan oleh *user* pada halaman *menu*.



Gambar 3.23 Desain antarmuka halaman *menu detail*

Halaman *menu detail* memiliki tampilan seperti pada Gambar 3.23, halaman ini menampilkan *chart* dan prediksi yang dimiliki oleh *menu* yang dipilih. *User* dapat mengubah *menu* tersebut dengan menekan tombol “*Edit*”, dan menghapus *menu* tersebut dengan menekan tombol “*Delete*”. *User* juga dapat menambahkan *data* untuk *menu* yang dipilih dengan menekan tombol “*Add New Data*”.

M. Desain Antarmuka Halaman *Edit Menu Detail*

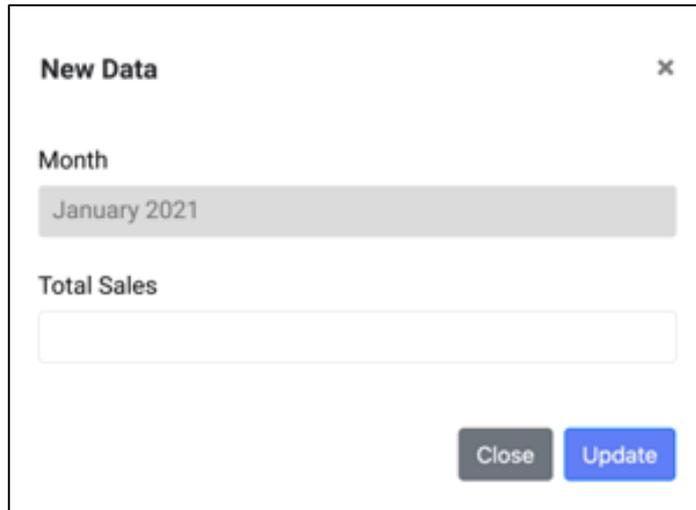
Halaman *edit menu detail* akan ditampilkan jika *user* menekan tombol “*Edit*” pada halaman *menu detail*.

Gambar 3.24 Desain antarmuka halaman *edit menu detail*

Halaman *edit menu detail* memiliki tampilan seperti pada Gambar 3.24, halaman ini memiliki *form* mengenai informasi *menu* tersebut yang memiliki *field* nama *menu*. *User* dapat mengubah nama *menu* tersebut dan menyimpan perubahannya dengan menekan tombol “*Update*”, *user* juga bisa keluar dari halaman ini jika tidak ingin menambahkan data baru dengan cara menekan tombol dengan ikon silang pada kanan atas halaman ini atau dengan cara menekan tombol “*Close*”.

N. Desain Antarmuka Halaman *Add New Data – Menu Detail*

Halaman *add new data – menu detail* akan ditampilkan jika *user* menekan tombol “*Add New Data*” pada halaman *menu detail*.



Gambar 3.25 Desain antarmuka halaman *add new data – menu detail*

Halaman *add new data – menu detail* memiliki tampilan seperti pada Gambar 3.25, halaman ini memiliki *form* yang meminta *user* untuk mengisi total sales pada bulan yang telah ditentukan untuk *menu* yang dipilih. *User* dapat menambahkan *data* baru untuk *menu* yang dipilih dengan menekan tombol “Add”, *user* juga bisa keluar dari halaman ini jika tidak ingin menambahkan *data* baru dengan cara menekan tombol dengan ikon silang pada kanan atas halaman ini atau dengan cara menekan tombol “Close”.

O. Desain Antarmuka Halaman *Edit Data – Menu Detail*

Halaman *edit data – menu detail* akan ditampilkan jika *user* menekan tombol “Edit” pada halaman *menu detail*.

The image shows a modal window titled "Edit Data" with a close button (X) in the top right corner. Inside the modal, there are two input fields. The first field is labeled "Month" and contains the text "January 2021". The second field is labeled "Total Sales" and is currently empty. At the bottom right of the modal, there are two buttons: a grey "Close" button and a blue "Update" button.

Gambar 3.26 Desain antarmuka halaman *edit data - menu detail*

Halaman *edit data – menu detail* memiliki tampilan seperti pada Gambar 3.26, halaman ini memiliki *form* mengenai informasi *data* pada *menu* yang dipilih yang memiliki 2 *field* yaitu *month* dan total sales pada data untuk menu yang dipilih. *User* dapat mengubah *field* tersebut dan menyimpan perubahannya dengan menekan tombol “*Update*”, *user* juga bisa keluar dari halaman ini jika tidak ingin menambahkan *data* baru dengan cara menekan tombol dengan ikon silang pada kanan atas halaman ini atau dengan cara menekan tombol “*Close*”.