**Sistem Pemesanan Menu Coffee Shop Berbasis Aplikasi**

|  | **KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET DAN TEKNOLOGI**  **UNIVERSITAS KHAIRUN**  **FAKULTAS TEKNIK PROGRAM STUDI INFORMATIKA**  **Kampus III Universitas Khairun, Kelurahan Jati Kota Ternate Selatan**  <http://if.unkhair.ac.id>, <http://if.unkhair.ac.id/simak> Group FB: if.unkhair |
| --- | --- |



Nama Skil : *System* Analis

Nama Kelompok : BarITzy

Kelas :5if4

Nama Anggota Kelompok : Farida Rustam (07352211047)

Ririn Muhammad (07352111008)

Akbar Saputra (07352211004)

| **NO** | **Tanggal** | **Uraian** | **Paraf** |
| --- | --- | --- | --- |
|  | 03/12/2024 | -gambarkan alur metode waterfallnya |  |
|  |  | -perhatikan penulisan bahasa asing dan singkatan harus sesuai dengan pedoman penulisan Prodi Informatika |  |
|  |  | -SESUAIKAN FORMAT DENGAN CONTOH YANG SUDAH DIBERIKAN |  |
|  |  | -di akhir tambahkan timeline proyek yang mencakup semua tahap, dari Dart Fundamental hingga aplikasi final. dan juga pembagian role tiap anggota |  |
|  | 07/12/2024 | -PERBAIKI TIMELINE |  |
|  | 17/12/2024 | ACC |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**1. Pendahuluan**

**1.1. Tujuan Penulisan Dokumen**

Dokumen ini berisi Spesifikasi Kebutuhan Perangkat Lunak(SKPL) untuk *Sistem pemesanan menu coffee shop.* Tujuan penulisan dokumen ini adalah untuk memberi penjelasan sistem aplikasi yang dibangun. Diharapkan pengembangan dari perangkat lunak jadi menjadi lebih terfokus dan tidak menimbulkan keambiguan bagi pengembang atau *programmer.*

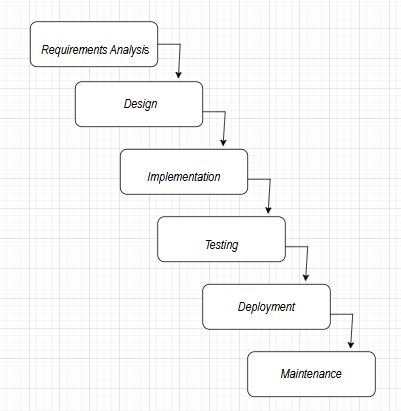
**1.2 Lingkup Masalah**

Semua hal yang tercantum dalam dokumen ini merupakan bagian ruang lingkup kebutuhan pembangunan perangkat lunak yang berupa aplikasi berbasis *android/iOS* yang digunakan untuk pemesanan menu,pembayaran dan lain-lain.

**1.3. Metode Pengembangan**

Metode pengembangan yang digunakan yaitu *Waterfall*. Model *Waterfall*(model air terjun) adalah model SDLC tertua dan paling terkenal. Model ini banyak digunakan dalam projek-projek besar karena pengembangan perangkat lunak dilakukan secara linear dan berurut. Model ini bergerak ke bawah melalui fase analisis kebutuhan, desain, pengkodean, pengujian,penerapan dan pemeliharaan. Dalam model ini, setiap fase dalam siklus pengembangan perangkat lunak dijalankan secara berurutan, dan masing-masing fase bergantung pada fase pertama atau fase yang dilakukan sebelumnya. Keuntungan dari model *Waterfall* adalah dengan melibatkan pendekatan yang terstruktur sehingga mudah dimengerti, dan setiap fase memiliki hasil yang jelas dan terdokumentasi dengan baik. Namun, kelemahan dari model ini berada pada ketidak ampuan untuk menangani perubahan kebutuhan pengguna setelah *proyek* dimulai. Jadi, jika terdapat perubahan kebutuhan pada tahap selanjutnya, maka pengimplementasiannya akan sulit dan mahal. Berikut alur pengembangan metode *waterfall* :

1.3.1 Alur Pengembang *Waterfall*



**Penjelasan:**

### 1. *Requirements Analysi*s (Analisis Kebutuhan)

Tujuan: Mengidentifikasi kebutuhan pengguna dan bisnis yang harus dipenuhi oleh sistem.

Aktivitas:

1. Mengumpulkan kebutuhan dari pelanggan, *admin*, dan pihak manajemen.
2. Mendokumentasikan fitur yang diperlukan, seperti autentikasi, pemesanan, pengelolaan menu, dan laporan.
3. Membuat *Software Requirement Specification* (SRS) yang merinci semua kebutuhan sistem.

### 2. *System Design* (Perancangan Sistem)

Tujuan**:** Menerjemahkan kebutuhan yang telah dikumpulkan menjadi desain sistem.

Aktivitas:

1. Mendesain arsitektur sistem (f*ront-end* untuk aplikasi *mobile* dan *web*, serta *back-end* untuk *server*).
2. Membuat diagram alur data (DFD), diagram proses, dan diagram hubungan antar entitas (ERD).
3. Mendesain antarmuka pengguna (*User Interface*) untuk aplikasi *mobile* dan *web*.

3. *Implementation* (Implementasi)

Tujuan: Membuat kode berdasarkan desain sistem.

Aktivitas:

1. Membagi pekerjaan menjadi pengembangan aplikasi *mobile*
2. Membuat fitur aplikasi *mobile* seperti autentikasi, daftar menu, keranjang belanja, dan notifikasi. Dengan modul dashboard, laporan, kelola menu, dan kelola pesanan.
3. Mengintegrasikan *API* untuk menghubungkan aplikasi *mobile* dan database.

**4. *Testing* (Pengujian)**

Tujuan: Memastikan aplikasi berfungsi sesuai dengan kebutuhan dan bebas dari kesalahan.

Aktivitas:

1. Melakukan pengujian unit (*unit testing*) untuk setiap fitur aplikasi.
2. Melakukan pengujian integrasi (*integration testing*) untuk memastikan interaksi antara aplikasi *mobile* dan *back-end* berjalan dengan baik.
3. Melakukan pengujian penerimaan pengguna (*User Acceptance Testing*) untuk memastikan aplikasi memenuhi kebutuhan pelanggan dan *admin*.

**5. *Deployment* (Penerapan)**

Tujuan: Mengimplementasikan aplikasi pada lingkungan produksi.

Aktivitas:

1. Menyiapkan *server* untuk aplikasi dan *database*.
2. Mengunggah aplikasi mobile ke *Google Play Store* dan/atau *App Store*.
3. Memberikan panduan penggunaan kepada pelanggan dan *admin*.

**6. *Maintenance* (Pemeliharaan)**

Tujuan: Memastikan aplikasi tetap berfungsi dengan baik setelah diterapkan.

Aktivitas:

1. Memperbaiki bug atau masalah yang ditemukan setelah aplikasi diluncurkan.
2. Melakukan pembaruan fitur sesuai dengan kebutuhan pengguna yang berkembang.
3. Memastikan *server* tetap stabil dan dapat menangani lalu lintas pengguna.

**1.4 Definisi,Akronim dan Singkatan**

| **Kata Kunci** | **Definisi atau Akronim** |
| --- | --- |
| SKPL | Spesifikasi Kebutuhan Perangkat Lunak atau dalam bahasa Inggris-nya sering juga disebut sebagai *Software Requirements Specification*  ( SRS) dan merupakan spesifikasi perangkat lunak yang akan dikembangkan. |
| UML | UML (*Unified Modeling Language*) adalah sekumpulan diagram yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML dapat digunakan untuk mempermudah pengembangan aplikasi yang berkelanjutan. |
| *Admin* | Merupakan Standar internasional untuk pengembangan dan perancangan perangkat lunak |
| *User* | *Software Requirement Specification* Dokumen ini sama dengan SKPL |

**1.5 Referensi**

Dokumen ini dibuat berdasarkan referensi sebagai berikut:

1. *Recruitment* HMTI UNKHAIR

2. *Requirement Project- Productivity*

3. *Module* dan *backlog*

**2. Gambaran *System***

Sistem pemesanan menu *coffee shop* ini dirancang untuk mempermudah pelanggan, *admin*, dan barista dalam mengelola proses pemesanan dan operasional *coffee shop*. Aplikasi ini berfungsi sebagai solusi *all-in-one* yang dapat diakses melalui perangkat *mobile* dengan antarmuka responsif, memungkinkan setiap pengguna untuk menggunakan fitur yang sesuai dengan perannya.

**A. Aplikasi *Mobile***

1. Autentikasi  
   Sistem ini memungkinkan setiap pengguna, baik pelanggan, *admin*, maupun barista, untuk *login* ke dalam akun mereka melalui fitur autentikasi. Pelanggan dapat mendaftar dengan nama dan nomor telepon untuk membuat akun baru, sementara *admin* dan barista menggunakan kredensial khusus yang telah diberikan oleh manajemen *coffee shop. Logout* juga tersedia untuk menjaga keamanan data pengguna.
2. *Home*  
   Pada aplikasi, pelanggan dapat melihat daftar menu yang tersedia di *coffee shop*. Menu dikelompokkan berdasarkan kategori, seperti minuman panas, dingin, atau makanan ringan. Promosi yang sedang berlangsung juga ditampilkan secara mencolok di bagian atas beranda untuk menarik perhatian pelanggan. Pelanggan dapat langsung memilih menu yang diinginkan dan menambahkannya ke keranjang belanja dengan satu klik.
3. Keranjang  
   Setelah memilih menu, pelanggan dapat mengakses keranjang untuk meninjau item yang telah dipilih. Di sini, pelanggan bisa mengubah jumlah pesanan atau menghapus item yang tidak diinginkan. Saat pelanggan merasa yakin dengan pesanannya, mereka dapat melanjutkan ke proses *checkout* dan memilih metode pembayaran, baik melalui transfer bank, *e-wallet*, atau pembayaran langsung di toko.
4. Pesanan Saya  
   Modul ini memungkinkan pelanggan untuk melacak status pesanan mereka secara r*eal-time.* Status seperti "Dalam Proses", "Siap Diambil", atau "Selesai" akan diperbarui sesuai dengan perkembangan pesanan. Selain itu, pelanggan juga dapat melihat riwayat pesanan mereka untuk memudahkan pemesanan ulang di masa depan.
5. Notifikasi  
   Setiap pembaruan pada status pesanan akan langsung diinformasikan kepada pelanggan melalui *push notification*. Sebagai contoh, pelanggan akan menerima notifikasi saat pesanan diterima, sedang diproses, atau telah siap untuk diambil. Hal ini membantu menjaga pelanggan tetap terinformasi tanpa harus membuka aplikasi secara terus-menerus.
6. Profil  
   Di bagian profil, pelanggan dapat memperbarui informasi akun mereka, seperti nama, alamat pengantaran, atau kata sandi. Pelanggan juga dapat keluar dari akun mereka melalui fitur ini untuk memastikan keamanan data pribadi.
7. *Dashboard*

Untuk *admin* dan barista, aplikasi ini menyediakan *dashboard* yang menampilkan informasi penting secara ringkas. *Dashboard* ini mencakup jumlah total pesanan, pesanan yang sedang diproses, total pendapatan harian, dan grafik penjualan. Data ini membantu manajemen dalam mengambil keputusan operasional.

1. Kelola Menu

*Admin* dapat menambahkan menu baru ke dalam sistem, mengedit informasi menu seperti nama, harga, dan deskripsi, serta menghapus menu yang tidak lagi tersedia. Fitur ini dirancang untuk memberikan fleksibilitas penuh kepada *admin* dalam mengatur menu sesuai dengan kebutuhan *coffee shop.*

1. Kelola Pesanan

Melalui modul ini, dan barista dapat melihat daftar pesanan yang masuk, memperbarui status pesanan, dan mencetak struk pesanan jika diperlukan. Setiap perubahan pada status pesanan secara otomatis akan memperbarui notifikasi pelanggan untuk memastikan transparansi.

1. Laporan Penjualan

Aplikasi menyediakan laporan penjualan yang dapat diakses oleh *admin* dan barista. Laporan ini mencakup data transaksi harian, mingguan, atau bulanan yang dapat diunduh dalam format PDF atau Excel. Dengan laporan ini, manajemen dapat memantau kinerja penjualan dengan mudah.

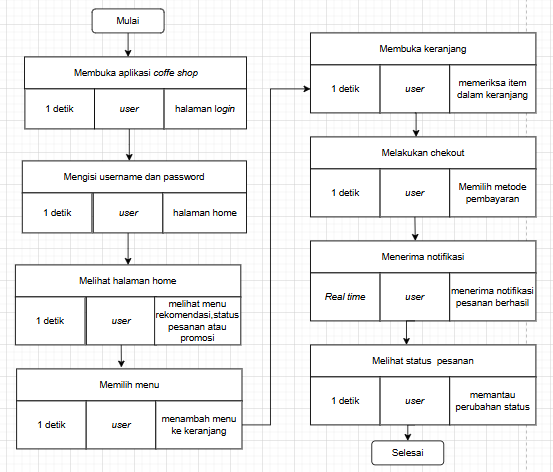
1. Manajemen Akun

*Admin* memiliki akses untuk mengelola akun pengguna lainnya, baik akun *admin* maupun barista. *Admin* dapat menambahkan, menghapus, atau memperbarui informasi akun untuk memastikan setiap pengguna memiliki akses yang sesuai dengan peran mereka.

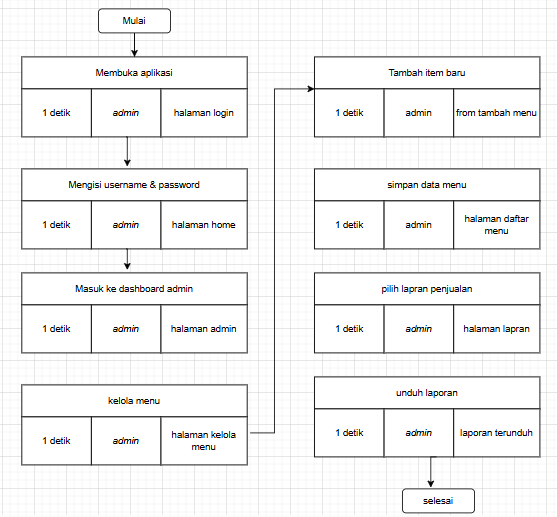
**2.1. Alur Proses**

Berikut alur proses sistem pemesanan menu *coffee shop*:

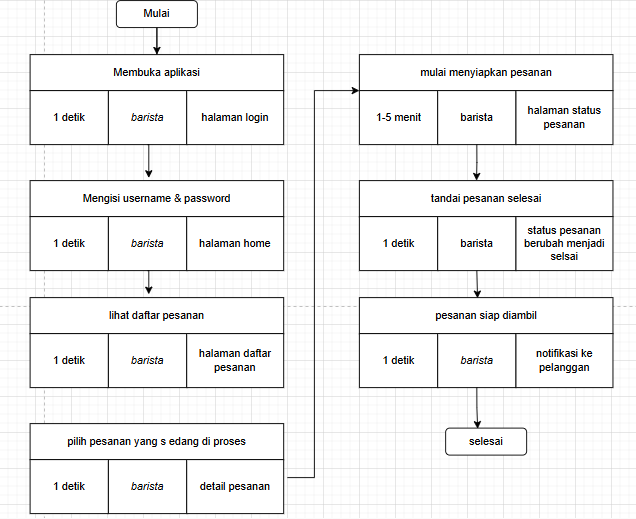
2.1.1 Alur Proses *User*



2.1.2. Alur Proses *Admin*



2.1.3. Alur Proses Barista



**3. Persyaratan *Non*-Fungsional**

**3.1. Persyaratan Infrastruktur**

Berikut persyaratan infrastruktur untuk pemesanan menu *coffee shop*:

| No | Persyaratan | Spesifikasi Minimum | Direkomendasi | *On live* |
| --- | --- | --- | --- | --- |
| 1 | CPU | *Intel core* i-5 | *Intel core* i-7 | - |
| 2 | *Hardware Memory* | 8 GB | 16 GB | - |
| 3 | *Hard Disk Size* | 512 GB | 1 TB | - |
| 4 | *Server* | *Windows Server* 2008 | *Windows Server 2016* | *Windows Serve*r 2019 |
| 5 | *Runtime* | *.NET* 5 | *NET* 6 | - |
| 6 | *Database* | *Microsoft SQL Server* 2022 | *PostgreSQL* 14 | - |

**3.2 Lingkungan Implementasi**

3.2.1 Spesifikasi Aplikasi *Android*

Berikut spesifikasi aplikasi *Android* untuk sistem pemesanan menu *coffee shop:*

| No | Spesifikasi | Deskripsi |
| --- | --- | --- |
| 1 | Sistem Operasi | Min.*Android* Jelly Bean(API Level 17) |
| 2 | Basis Data | *SQLite* |
| 3 | Persyaratan Desain | Responsive min.1080 x 1920 |
| 4 | *Smartphone* | - |
| 5 | Tablet | Portrait |
| 6 | Orientasi | Portrait |
| 7 | Bahasa | Dart |
| 8 | Alat Pengembangan | *Android Studio* |

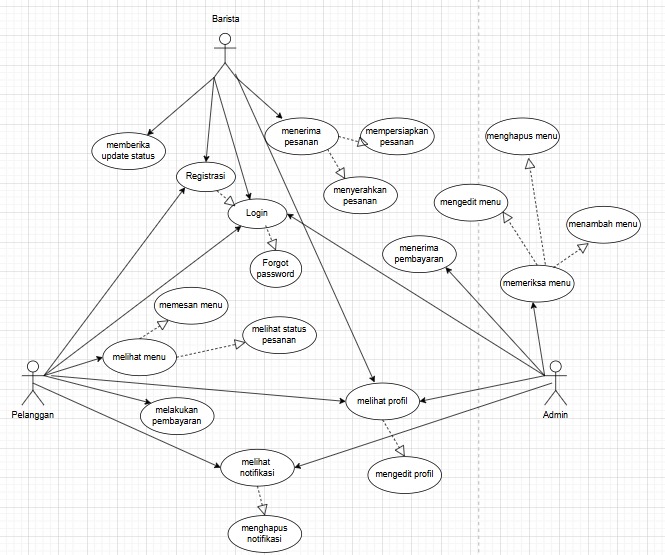
**4. Persyaratan Fungsional**

4.1 Analisa Pengguna Sistem

Berikut analisa pengguna sistem pemesanan menu *coffee shop:*

| Tipe Pengguna | Modul | Detail Aktivitas Pengguna |
| --- | --- | --- |
| *Admin* | *Dashboard* | *Admin* dapat melihat informasi jumlah *order*,jumlah pelanggan, dan jumlah pendapatan |
| *Admin* | *Report* | *Admin* dapat melihat laporan penjualan,stok, dan pendapatan |
| *Admin* | *Management user* | *Admin* dapat menambah,mengubah,dan menghapus data pengguna(pelanggan dan barista) |
| *Admin* | *Search* | *Admin* dapat mencari data pelanggan,produk,dan pesanan |
| Pelanggan | *Dashboard* | Pelanggan dapat melihat menu,menambahkan menu,menambahkan ke keranjang, mengecek status pesanan. |
| Pelanggan | Pelanggan | Pelanggan dapat melihat riwayat pesanan dan melihat detail transaksi |
| Pelanggan | *Search* | Pelanggan dapat mencari menu berdasarkan nama atau kategori |
| Barista | *Dashboard* | Barista dapat melihat daftar menu yang masuk,status pesanan,dan menu yang dipesan |
| Barista | *Report* | Barista dapat melihat laporan,pesanan yang telah selelsai |
| Barista | *Management user* | Barista dapat mengubah pesanan |
| Barista | *Search* | Barista dapat mencari data pesanan berdasarkan nama pelanggan atau id pesanan |

3.3.1 Diagram *Use case*

**

*Gambar 3.1 Use Case Diagram*

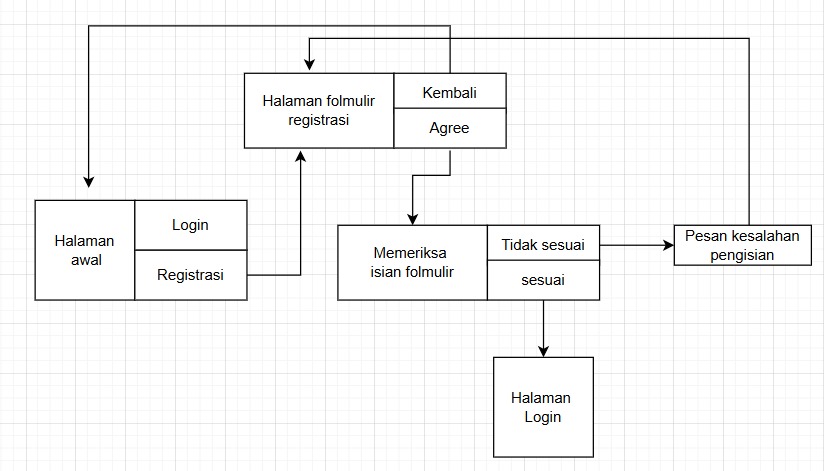
Penjelasan:

Untuk *use case* di gambar diatas terdapat tiga aktor. yang pertama ada *admin* yaitu setelah melakukan *login* admin melakukan 7 aktivitas yaitu, menambah menu,mengedit menu,menghapus menu, menerima pembayaran, melihat profil dan notifikasi yang kedua ada *user*(Pelanggan) setelah *registrasi, login user* melakukan 6 aktivitas yaitu melihat menu, memesan menu,membayar pesanan,melihat riwayat pesanan,melihat notifikasi dan profil yang ketiga ada barista setelah registrasi *login* barista melakukan 5 aktivitas yaitu menerima pesanan,mempersiapkan pesanan,menyerahkan pesanan dan memberikan *update* status dan melihat profil.

4.3 Detail Persyaratan Fungsional

Berikut detail *activity* berdasarkan diagram *use case* yang sudah digambarkan:

4.3.1 *Registrasi*



Gambar 4.1.Diagram *Activity Registrasi*

Deskripsi:

1. Pelanggan atau barista membuka aplikasi, aplikasi menampilkan halaman awal.

2. Pelanggan atau barista memilih pilihan Registrasi.

3. Aplikasi menampilkan halaman formulir registrasi.

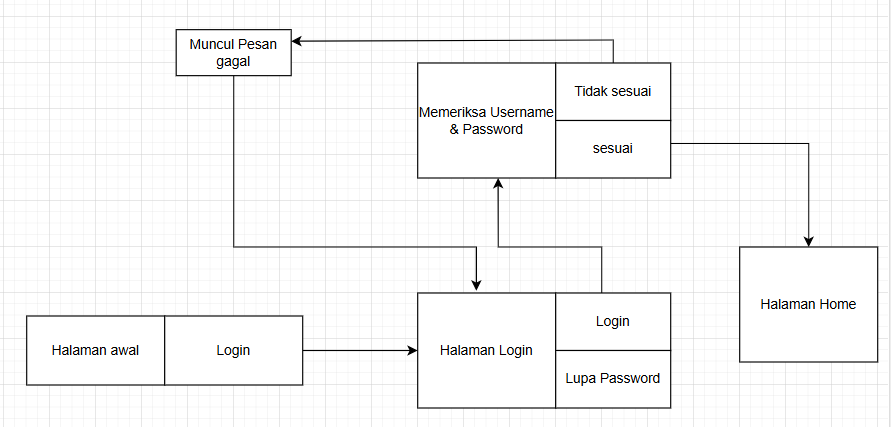
4. Pelanggan atau barista mengisi formulir registrasi yang tersedia. Jika memilih Kembali, maka aplikasi akan kembali ke halaman awal.

5. Pelanggan atau barista memilih pilihan “*Agree* and *register*”, aplikasi memeriksa isian formulir oleh pelanggan.

6. Jika terdapat data yang belum diisi atau data yang tidak sesuai, maka aplikasi akan menampilkan pesan kesalahan dan pelanggan dan barista diminta melengkapi atau mengisi formulir kembali.

7. Jika data sesuai, data akan disimpan ke basis data dan aplikasi akan menampilkan pesan berhasil, lalu Pelanggan atau barista akan menuju halaman login.

4.3.1 *Login*



Gambar 4.2 Diagram *Activity login*

Deskripsi:

1. *Admin*, barista dan pelanggan membuka aplikasi, aplikasi menampilkan halaman awal.

2. *Admin*, barista dan pelanggan memilih pilihan *login*.

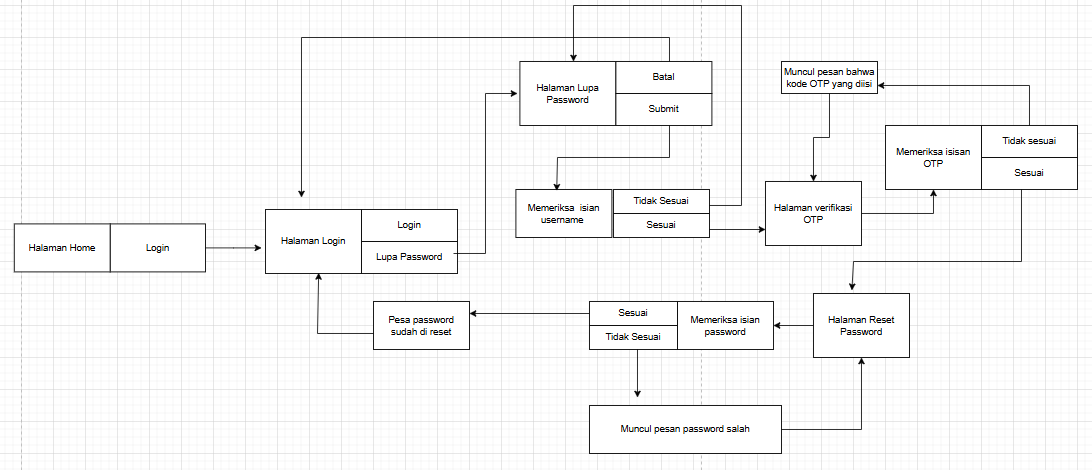
3. *Admin*, barista dan pelanggan akan diarahkan menuju halaman *login.*

4. *Admin*, barista dan pelanggan memilih *login*  dan melengkapkan *username* dan *password*.

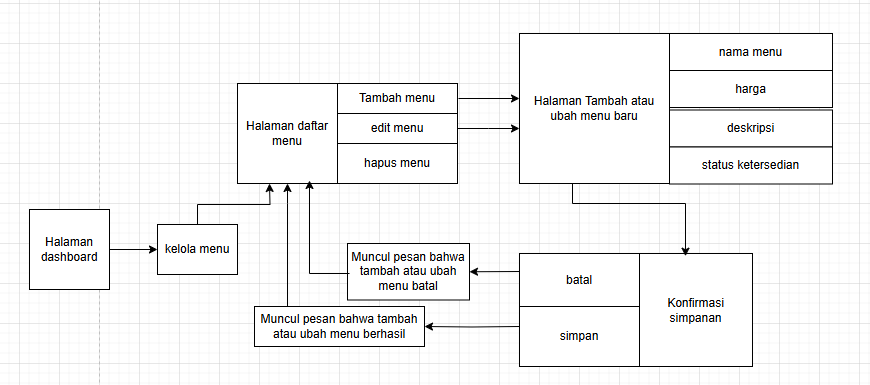
5. Jika terdapat data yang salah, belum ada atau tidak sesuai,maka aplikasi akan menampilkan pesan kesalahan.

6. Jika data sudah sesuai, *admin*, barista dan pelanggan akan diarahkan menuju halaman *home*

4.3.2 *Forgot password*



Gambar 4.3 Diagram *Activity Forgot Password*

4.3.3 *Admin* mengelola menu 

Gambar 4.4 Diagram *Activity Admin* Mengelola menu

Deskripsi:

1. *Admin* membuka aplikasi dan diarahkan ke halaman *dashboard*.

2. *Admin* memilih menu "Kelola Menu." untuk mengatur daftar menu.

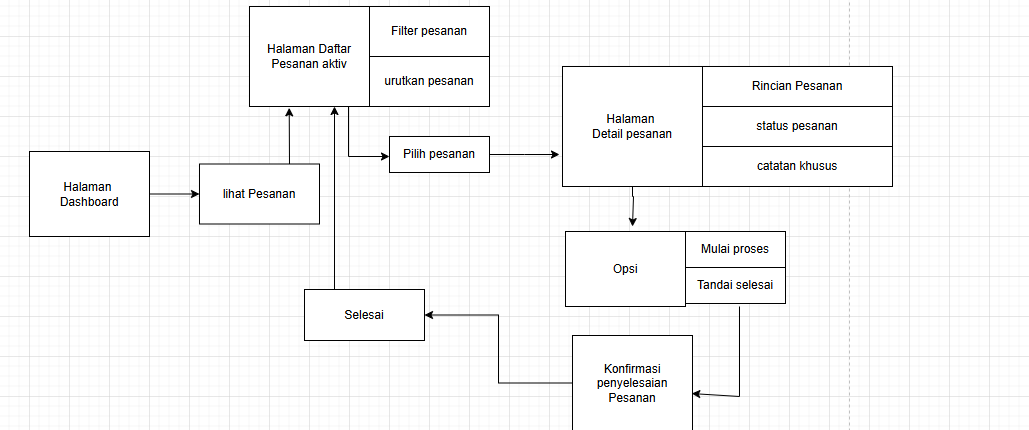
3. *Admin*  melihat daftar menu yang tersedia yaitu *admin* dapat menambah menu (membuat menu baru dengan informasi lengkap), edit menu (mengubah detail menu yang sudah ada ), atau menghapus menu(menghapus menu dari daftar) di halaman ini.

4. Jika menambah atau mengubah menu, *Admin* mengisi informasi menu baru atau mengubah data menu, lalu mengkonfirmasi penyimpanan.

5. Jika memilih "Simpan," muncul notifikasi perubahan disimpan, dan *Admin* kembali ke daftar menu.

6. Jika memilih "Batal," muncul notifikasi perubahan tidak disimpan, dan *Admin* Kembali ke daftar menu.

4.3.4 Barista Melihat Pesanan



Gambar 4.5 Diagram *Activity* Barista Melihat Pesanan

Deskripsi:

1. Barista membuka aplikasi dan langsung diarahkan ke halaman *dashboard*.

2. Barista memilih menu lihat pesanan untuk melihat semua pesan yang masuk.

3. Barista melihat daftar semua pesanan yang statusnya belum selesai dan informasi yang ditampilkan pada daftar pesan.

4. Barista dapat memilih salah satu pesanan untuk melihat detailnya

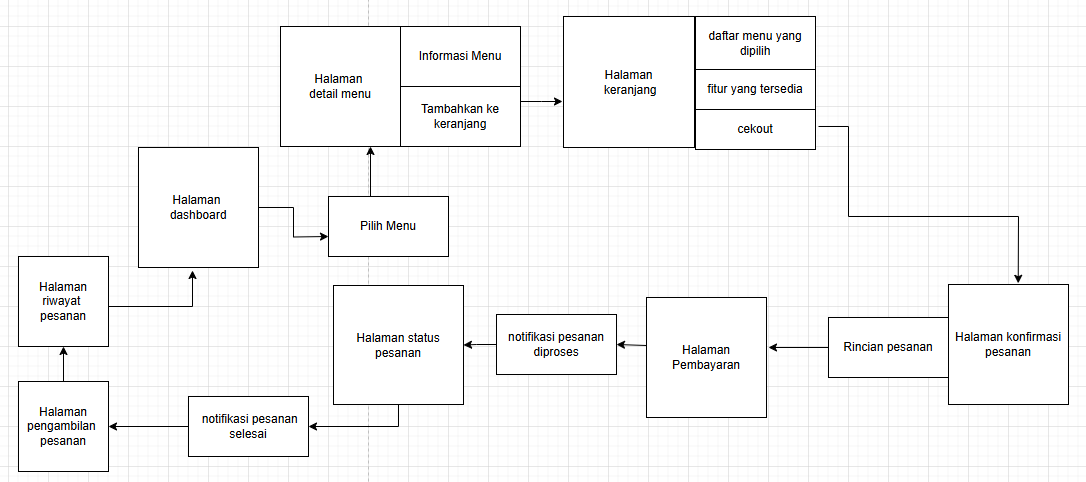
5. Barista melihat informasi lengkap pesanan seperti rincian pesan,status pesanan.

6. Barista memiliki dua opsi yaitu mulai proses dan tandai selesai. jika barista menekan mulai proses maka dia akan mengubah status pesanan menjadi dalam proses, jika menekan tandai selesai maka akan mengubah status menjadi selesai.

7. Jika barista menekan tanda selesai akan muncul dialog konfirmasi, setelah mengkonfirmasi status pesanan diperbarui menjadi selesai, dan pesanan dihapus dari daftar pesanan aktif.

8. Setelah menyelesaikan atau memperbarui status pesanan, barista kembali ke halaman daftar pesanan aktif untuk melihat pesanan berikutnya.

4.3.5 Pelanggan Membuat Pesanan



Gambar 4.6 Diagram *Activity* Pelanggan Membuat Pesanan

Deskripsi:A

1. Pelanggan membuka aplikasi dan diarahkan ke halaman beranda.

2. pelanggan melihat daftar menu dan memilih salah satu menu yang tersedia dan memilih salah satu menu untuk melihat detailnya.

3. pelanggan melihat informasi menu seperti seperti nama,harga,deskripsi, dan status ketersediaan.

4. Pelanggan dapat menambahkan menu ke keranjang dengan memilih opsi tambahkan ke keranjang .

5. Pelanggan dapat meninjau daftar menu yang sudah dipilih,fitur yang tersedia, dan setelah selesai pelanggan memilih opsi *checkout*.

6. Sistem menampilkan rincian pesanan untuk dikonfirmasi seperti daftar item, total harga, pilih metode pembayaran.pelanggan mengkonfirmasi pesanan untuk melanjutkan ke pembayaran.

7. setelah pembayaran berhasil,pesanan dikirim ke barista untuk di proses.

8. Pelanggan menerima notifikasi bahwa pesanan sedang di proses.

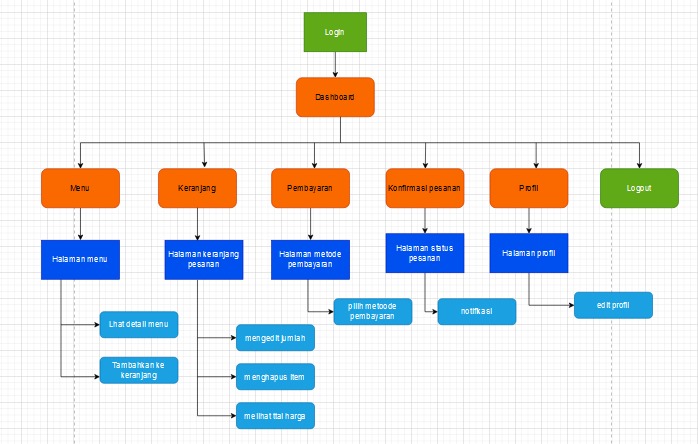
9. setelah itu masuk ke halaman status pesanan pelanggan dapat memantau perkembangan jika statusnya sudah selesai maka muncul notifikasi selesai.

10. Pelanggan diarahkan ke halaman pengambilan pesanan jika *dine-in* atau *takeaway*.

**5. Antarmuka Sistem**

5.1. *Sitemap*

5.1.1. *Mobile*



Gambar 5.1 Antarmuka Sistem *Mobile*

5.5 Antarmuka Sistem Menu *Coffee Shop*

5.2.1. *Authentication*

| Nama  fungsional | *Authentication* |
| --- | --- |
| Nama proses | Melakukan *login* |
|  | |

| Nama  fungsional | *Authentication* |
| --- | --- |
| Nama proses | Melakukan *registrasi* |
|  | |

| Nama  fungsional | *Authentication* |
| --- | --- |
| Nama proses | Melakukan forgot password |
|  | |

| Nama  fungsional | *Authentication* |
| --- | --- |
| Nama proses | Melihat Profil |
|  | |

| Nama fungsional | *Home* |
| --- | --- |
| Nama proses | Melihat welcome *coffee shop* |
|  | |

| Nama  fungsional | Menu |
| --- | --- |
| Nama proses | Melihat daftar menu |
|  | |

| Nama  fungsional | Promo |
| --- | --- |
| Nama Proses | Melihat daftar promo |
|  | |

**6. *Link Timeline :***

[**https://docs.google.com/spreadsheets/d/1Exmqxq4UEKN-TpLNqV11CXD532ZhyYnY/edit?usp=sharing&ouid=114178106286691421840&rtpof=true&sd=true**](https://docs.google.com/spreadsheets/d/1Exmqxq4UEKN-TpLNqV11CXD532ZhyYnY/edit?usp=sharing&ouid=114178106286691421840&rtpof=true&sd=true)