

BAB 3

METODE PENELITIAN

Penelitian ini merupakan hasil dari rancangan serta pengembangan sistem yang didasarkan pada pemahaman yang mendalam terhadap permasalahan yang ditemukan, analisis proses, identifikasi sumber masalah, lalu perancangan dan implementasi aplikasi dengan tujuan mengurangi dan mengatasi permasalahan. Hal-hal yang diperlukan dalam pengembangan aplikasi pemesanan layanan tata rias Novy Salon akan dijabarkan sebagai berikut.

3.1 BAHAN DAN ALAT PENELITIAN

Bahan penelitian yaitu daftar inventaris salon, jenis-jenis treatment dan rias pengantin yang ada pada Novy Salon, harga setiap treatment dan paket rias pengantin, dan buku laporan keuangan serta data-data dari setiap asisten salon yang bekerja di Novy Salon.

Alat adalah perangkat yang digunakan pada penelitian yang mencakup komputer atau laptop sesuai spesifikasi yang memadai untuk mengoperasikan sistem komputer dan *software* pengembangan, dan koneksi internet.

Berikut adalah sistem operasi serta *software* untuk pengembangan dan perancangan aplikasi ini yaitu:

1. *Processor*: AMD Ryzen 5 5500U with Radeon Graphics
2. *RAM*: 8.00 GB
3. *Sytem Type*: 64-bit operating system, x64-based processor
4. *Operation System*: Windows 11 Home Single Language Ver 23H2
5. *Webserver*: Xampp Ver 3.3.0
6. *Bahasa Pemrograman*: Python Ver 3.9.12
7. *Framework*: Django Ver 4.1.4
8. *Database engine*: MySQL Ver 10.4.28
9. *Text Editor*: Visual Studio Code Ver p1.89.1
10. *Manajemen backlog*: Trello

3.2 JALAN PENELITIAN

Pengembangan aplikasi ini menggunakan model pengembangan Agile. Beberapa alasan menggunakan model Agile dikarenakan salah satu kelebihan model Agile yaitu tidak bergantung pada sumber daya yang besar (Yusril et al., 2021) sehingga sesuai dengan aplikasi yang akan dikembangkan dimana bukan sistem dengan skala besar. Alasan lainnya yaitu pengelolaan perubahan dalam pengembangan aplikasi dapat dengan cepat diatasi (Yusril et al., 2021) sehingga menambah kualitas dari aplikasi.

Pada metodologi Agile terdapat tahapan-tahapan yang diterapkan pada penelitian ini, diantaranya:

1. Tahap Analisis (*Plan*)

Dalam tahap ini dilakukan analisis pada proses pemesanan yang sebelumnya manual diubah menjadi sistem yang berbasis komputer. Proses analisis melibatkan pengumpulan data *user* dengan sistem wawancara secara langsung kepada pemilik salon agar memperoleh informasi yang *real* dan *valid*. Setelah memperoleh kebutuhan maka selanjutnya memetakan solusi serta fitur-fitur yang akan dirancang dan dikembangkan sesuai dengan fungsi yang diinginkan.

2. Tahap Perancangan (*Design*)

Dalam tahap ini, dilakukan perancangan aplikasi sesuai kebutuhan *user*. Fitur-fitur akan diterjemahkan menjadi sebuah desain yang mudah dipahami dan sesuai dengan kebutuhan. Perancangan yang dilakukan meliputi perancangan *database*, tampilan *user* dan juga admin.

3. Tahap Implementasi (*Develop*)

Tahap Implementasi yaitu desain yang sebelumnya telah dirancang akan ditranslasikan melalui penulisan kode atau bahasa pemrograman spesifik yang dipahami dan dapat diterjemahkan mesin komputer. Perancangan aplikasi ini menggunakan *framework* Django sehingga menggunakan bahasa pemrograman Python.

4. Tahap Pengujian (*Test*)

Dalam tahap pengujian ini, aplikasi yang telah dirancang akan diuji menggunakan metode pengujian *black box* serta *User Acceptance Test* (UAT).

Menurut Achmad & Yulfitri (2020) pengujian *black box* merupakan jenis pengujian berfokus terhadap pengujian fungsional sistem tanpa memperhatikan struktur internal dan logika kode yang digunakan. Untuk melakukan pengujian dengan *black box* akan disiapkan *test case* dan formulir penilaian. Pengujian akan dilakukan oleh Pakar (2 orang), Admin (1 orang), dan Asisten (1 orang) dan mengisikan hasilnya pada formulir penilaian yang telah disiapkan. User Acceptance Test (UAT) yaitu pengujian yang dilakukan pengguna dalam hal menguji terkait fungsionalitas aplikasi berjalan dengan sesuai spesifikasinya (Wahyudi et al., 2023).

5. Tahap *Deploy*

Pada tahap *deploy* akan melibatkan pemilihan penyedia *hosting web*, konfigurasi server, penyiapan *database*, migrasi data, pengaturan keamanan, peluncuran sistem, dan pemantauan produksi. Dengan *hosting online*, sistem dapat diakses oleh pengguna, meningkatkan efisiensi dan pengalaman pelanggan dalam pemesanan layanan salon.

Dalam model pengembangan Agile, pengembangan aplikasi dilakukan dalam beberapa sprint. Setiap sprint mencakup seluruh tahapan dari analisis, perancangan, implementasi hingga pengujian fungsionalitas. Sprint adalah iterasi pendek yang biasanya berlangsung selama 1 hingga 4 minggu, di mana fokus pada penyelesaian sejumlah fitur atau peningkatan tertentu yang telah diidentifikasi dalam backlog.

- Sprint 1: Halaman Admin

Pada Sprint 1, fokus pengembangan adalah pada pembuatan dan penyelesaian fitur dashboard admin, booking admin dan halaman admin lainnya. Dashboard ini diharapkan mampu memudahkan admin dalam mengelola berbagai aspek aplikasi seperti manajemen jadwal salon, detail pemesanan, dan pengaturan tanggal buka atau tutup tertentu. Fitur utama yang akan dikembangkan meliputi tampilan statistik pemesanan serta manajemen data jadwal. Sprint ini juga mencakup *review* dan penyesuaian untuk memastikan semua kebutuhan fungsional dan tampilan antarmuka sesuai dengan standar *user experience* (UX) yang baik.

- Sprint 2: Halaman Pelanggan

Pada Sprint 2, fokus utama adalah mengembangkan fitur reservasi pelanggan yang memungkinkan pengguna untuk memesan layanan tata rias melalui aplikasi. Fitur ini mencakup pemilihan layanan, pemilihan jadwal, dan konfirmasi pemesanan. Pengguna dapat dengan mudah memilih jenis layanan sesuai ketersediaan tanggal, lalu melakukan pemesanan secara *real-time*. Selain itu, sistem reservasi ini juga diintegrasikan dengan notifikasi untuk mengingatkan admin tentang jadwal yang telah pelanggan pesan.

3.3 ANALISIS KEBUTUHAN

Analisis dimulai dengan mengidentifikasi kebutuhan seluruh sistem yang akan diimplementasikan menjadi sebuah aplikasi. Dalam menganalisis sebuah sistem, ada tiga langkah dasar yang perlu dilakukan, yakni:

1. Identifikasi masalah

Berdasarkan pengumpulan data yang dilakukan di Novy Salon ditemukan permasalahan yaitu belum tersedianya sistem informasi yang melayani pemesanan pelanggan serta pelaporan dan *database* penjualan belum optimal.

2. Pemecahan masalah

Dalam memecahkan masalah yang dialami Novy Salon maka diusulkan mengembangkan sebuah aplikasi agar dapat mendukung, membantu, mempermudah serta meningkatkan sistem pelayanan di Novy Salon.

3. *Software Requirement*

Beberapa fitur yang akan dirancang dan diimplementasikan agar dapat mengatasi masalah yang ada yaitu, sebagai berikut:

- Fitur pemesanan layanan salon yang dapat mempermudah pelanggan dalam memesan layanan salon secara *online*, menyediakan informasi tentang layanan salon yang tersedia, harga, dan jadwal.
- Fitur reservasi jadwal layanan tata rias untuk mempermudah pelanggan dalam melakukan reservasi jadwal dan informasi jadwal yang tersedia.
- Fitur pengelolaan data pemesanan mempermudah pengelolaan data pemesanan, status pemesanan, dan histori pemesanan.

- Fitur pengelolaan data transaksi mempermudah pengelolaan data transaksi, termasuk status transaksi, dan histori transaksi.
- Fitur pengelolaan data asisten salon mempermudah pengelolaan data asisten, performa asisten dan unduh data transaksi terkait asisten.
- Fitur pengelolaan data pelanggan mempermudah pengelolaan data pelanggan, termasuk data kontak, alamat, dan histori pemesanan.
- Fitur pengelolaan data layanan mempermudah pengelolaan data layanan, termasuk data kategori layanan, harga, dan status.
- Fitur kelola *profile* untuk mengubah data *profile* dan juga melakukan perubahan pada kata sandi.
- Fitur notifikasi untuk mempermudah admin dalam mengetahui reservasi pelanggan.

3.4 UNIFIED MODELLING LANGUAGE (UML)

Pada tahap perancangan UML, terdapat tiga diagram utama yang akan dirancang yaitu *Use Case Diagram*, *Activity Diagram*, serta *Class Diagram*. *Use Case Diagram* akan memvisualisasikan relasi pengguna (aktor) terhadap sistem perihal menyelesaikan suatu tugas atau skenario tertentu. *Activity Diagram* akan menggambarkan urutan aktivitas atau proses yang terjadi dalam sistem. *Class Diagram* akan menjelaskan struktur statis *database* dari aplikasi yang akan dikembangkan dengan menunjukkan kelas-kelas yang ada beserta atribut dan metodenya. Diagram ini akan membantu dalam memahami hubungan antara entitas-entitas utama dalam sistem.

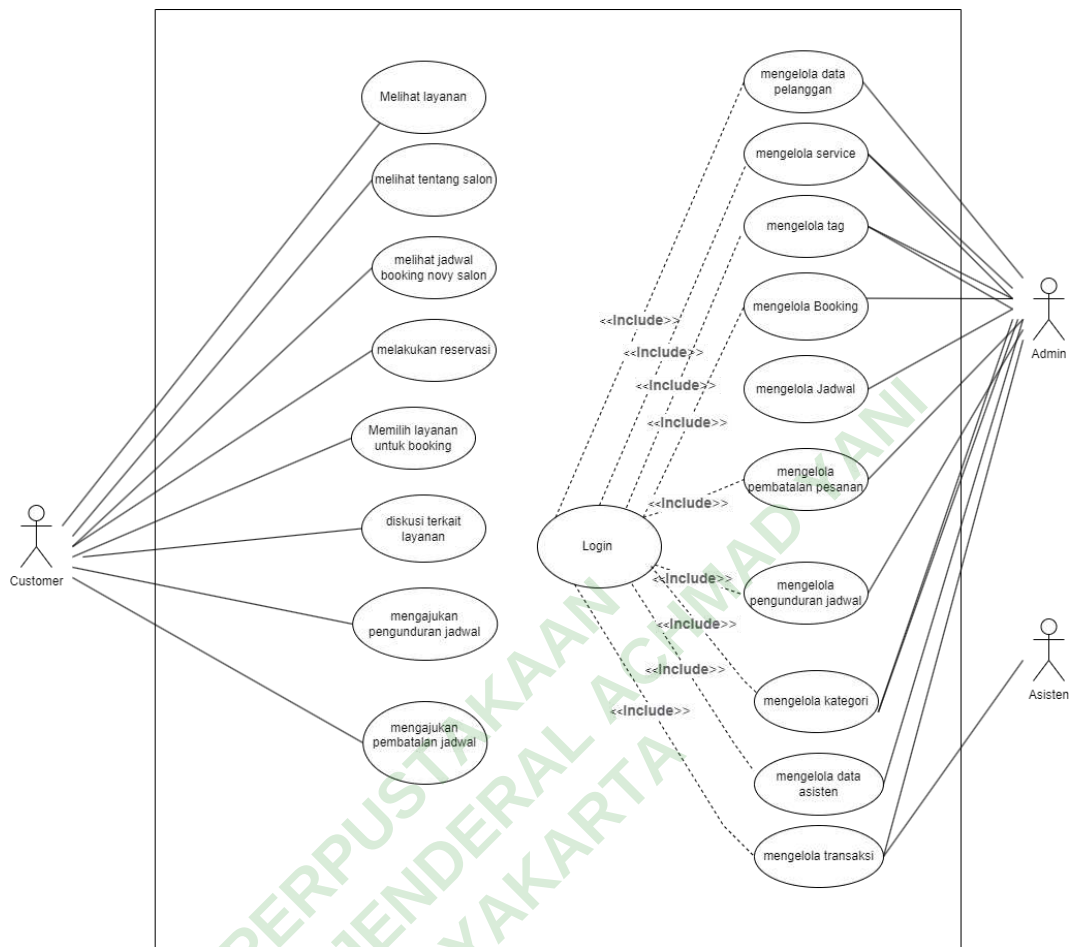
3.4.1 Use Case Diagram

Perancangan *Use Case Diagram* bertujuan untuk memvisualisasikan hubungan antar pengguna dalam aplikasi. Tabel 3.1 berikut merupakan penjabaran pengertian *use case*.

Tabel 3.1 Pengertian *Use Case*

No	<i>Use Case</i>	Keterangan
1	<i>Login</i>	Merupakan proses admin atau asisten mengakses halaman sistem.
2	Mengelola data kategori	Proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu kategori.
3	Mengelola data layanan	Proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu layanan.
4	Mengelola data tag	Melakukan proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu tag.
5	Mengelola data pemesanan	Proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu <i>booking</i> .
6	Mengelola data transaksi	Merupakan proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu transaksi.
7	Mengelola jadwal salon	Merupakan proses tambah dan <i>update</i> jadwal di kalender yang akan dapat dilihat pelanggan.
8	Mengelola data pelanggan	Merupakan proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu pelanggan.
9	Mengelola data asisten salon	Merupakan proses <i>create</i> , <i>update</i> , dan <i>delete</i> di menu asisten salon.
10	Melakukan reservasi jadwal salon	Merupakan proses reservasi jadwal sesuai dengan tanggal yang tersedia.
11	Mengelola notifikasi reservasi	Merupakan proses kelola notifikasi <i>booking</i> dan menampilkan detail <i>booking</i> .

Berdasarkan Tabel 3.1 maka hak akses pengguna berbeda -beda. Pada aplikasi yang akan dikembangkan terdapat 3 aktor yaitu pemilik salon atau admin, asisten salon dan pelanggan. Masing-masing aktor memiliki akses seperti di Gambar 3.1.

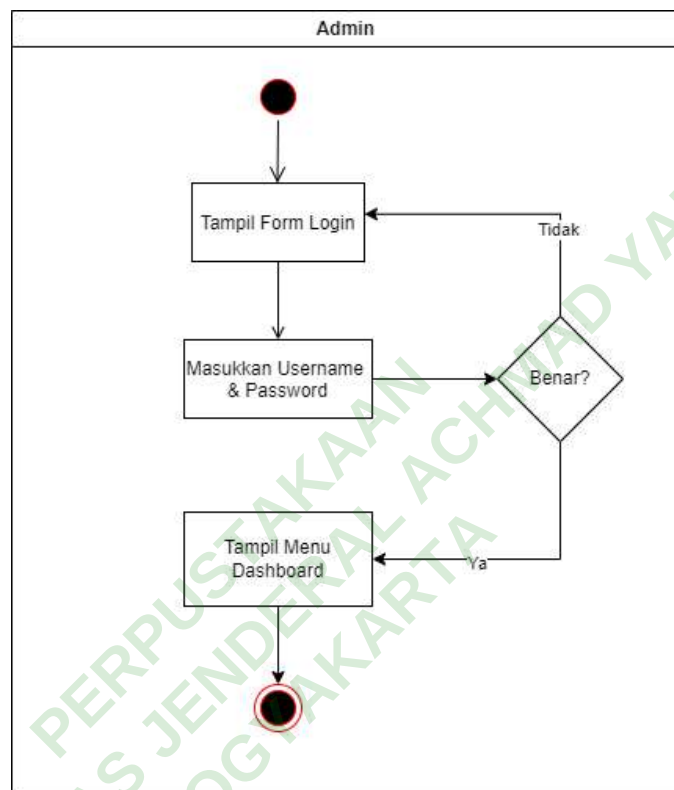


Gambar 3.1 Use case diagram

Menurut Gambar 3.1 *customer* atau pelanggan memiliki akses untuk melihat layanan, tentang salon, jadwal yang tersedia di novy salon, melakukan reservasi jadwal, dan melakukan pengajuan pengunduran jadwal atau pembatalan jadwal. Admin atau pemilik salon memiliki semua akses sistem yang ada yaitu mengelola data kategori, tag, layanan, pemesanan, transaksi, pelanggan, dan asisten. Selain itu, asisten salon memiliki hak akses pada fitur transaksi yang dimana hal ini bertujuan untuk mengetahui performa asisten setiap periode waktu tertentu serta perkembangan bisnis.

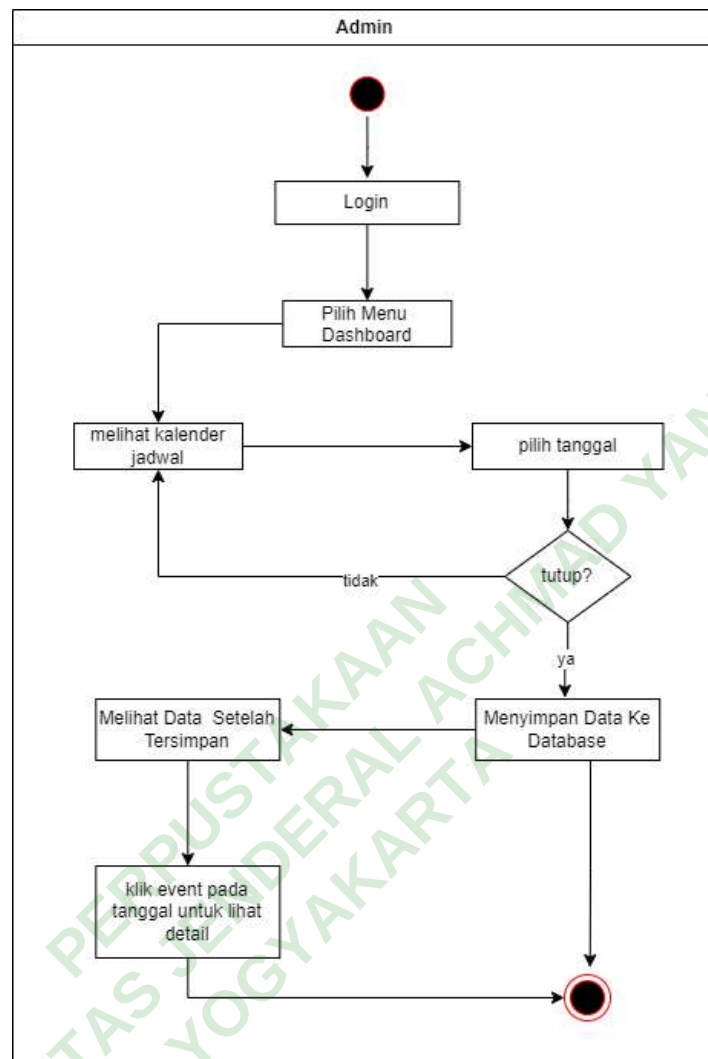
3.4.2 Activity Diagram

Activity diagram akan menjelaskan runtutan aktivitas setiap proses dalam sistem. Aktivitas-aktivitas tersebut memiliki awal sampai akhir dari proses aktivitas.



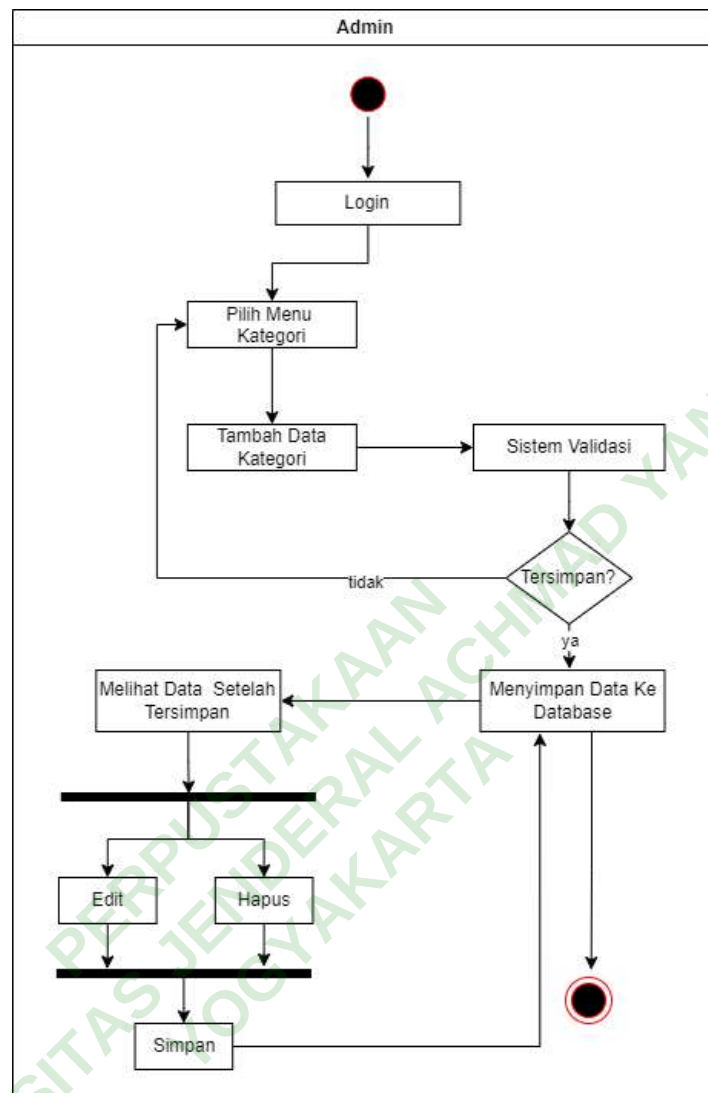
Gambar 3.2 Proses *login* admin

Pada *activity* proses *login* seperti terlihat pada Gambar 3.2 menjelaskan alur admin sehingga dapat mengakses atau masuk kedalam sistem. Admin akan menginputkan *username* serta *password* yang terdaftar lalu melakukan *login*. Apabila terdapat kesalahan saat penginputan *username* atau *password* maka akan tampil pesan *error* dan tidak dapat mengakses sistem. Namun, jika *username* serta *password* yang diinputkan valid maka berhasil melakukan login dan diarahkan ke halaman dashboard.



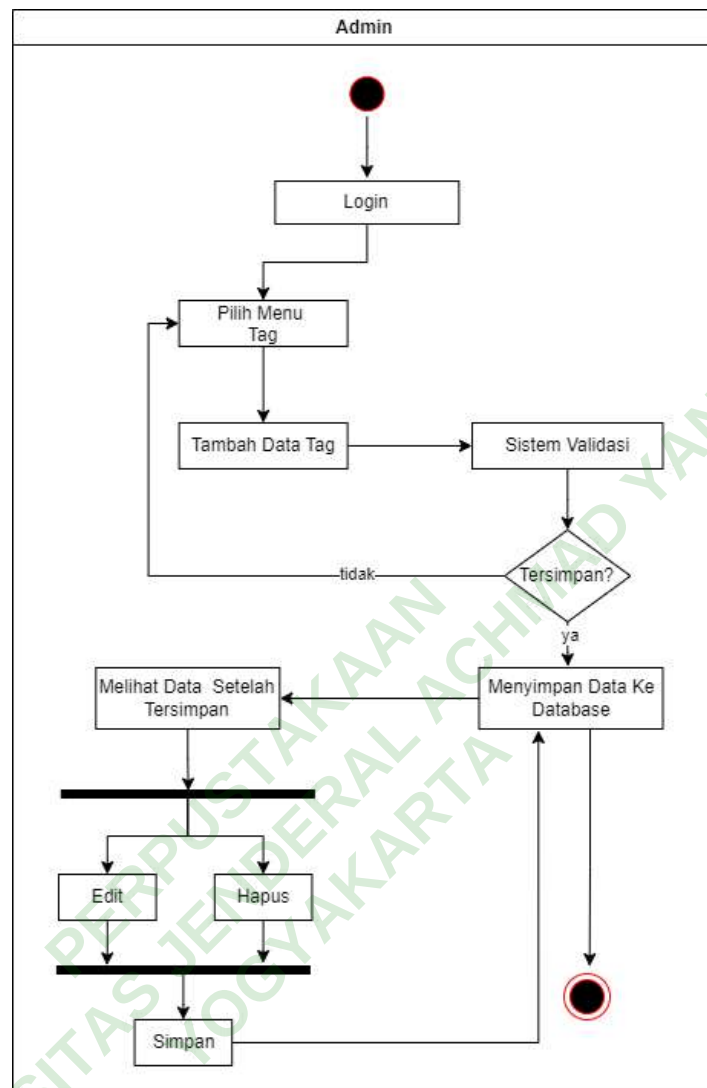
Gambar 3.3 Proses kelola jadwal

Pada *activity* proses kelola jadwal seperti Gambar 3.3 terlihat bahwa setelah proses autentikasi admin berhasil maka akan menampilkan halaman *dashboard*. Halaman *dashboard* digunakan untuk mengelola jadwal. Dalam halaman *dashboard* terdapat sebuah kalender yang berisi jadwal dengan beberapa keterangan. Admin dapat melakukan proses perubahan ketersediaan jadwal dan melihat detail jadwal pada tanggal tertentu.



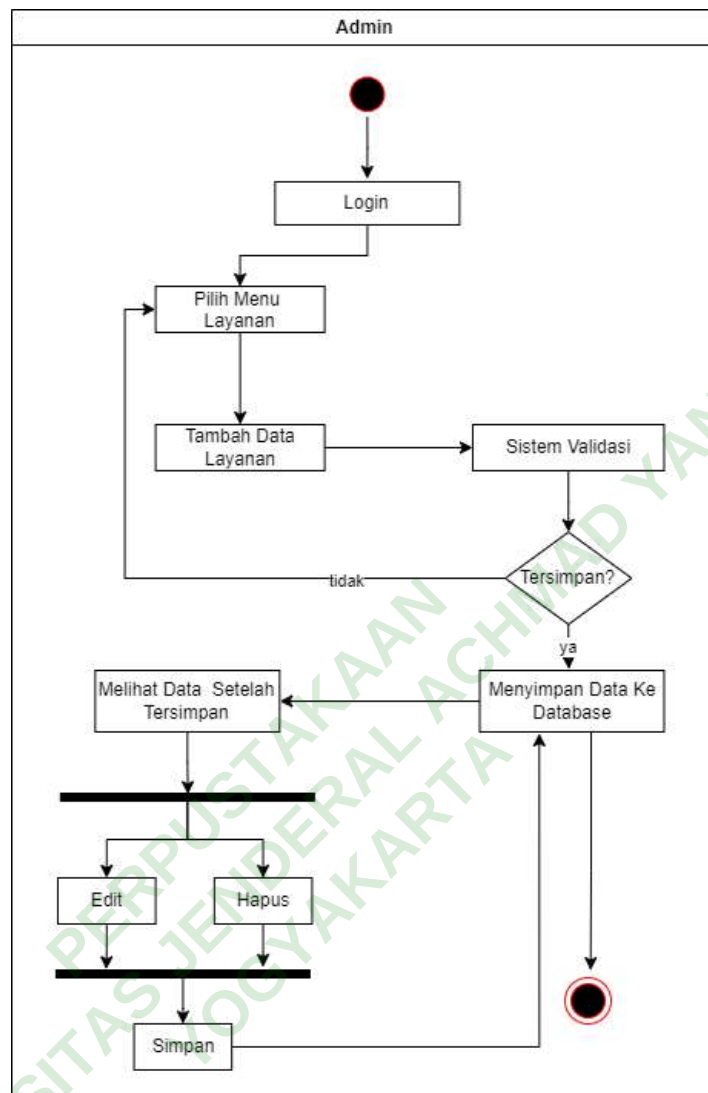
Gambar 3.4 Proses kelola kategori

Sebelum melakukan kelola kategori, admin harus terlebih dahulu *login* yang terlihat di Gambar 3.4. Proses kelola kategori bertujuan untuk menampilkan list kategori, melakukan edit kategori, menambah kategori dan menghapus kategori tertentu. Sebelum melakukan penyimpanan proses tersebut maka sistem akan memvalidasi data yang diinputkan jika benar maka akan menampilkan perubahan atau penambahan yang dilakukan.



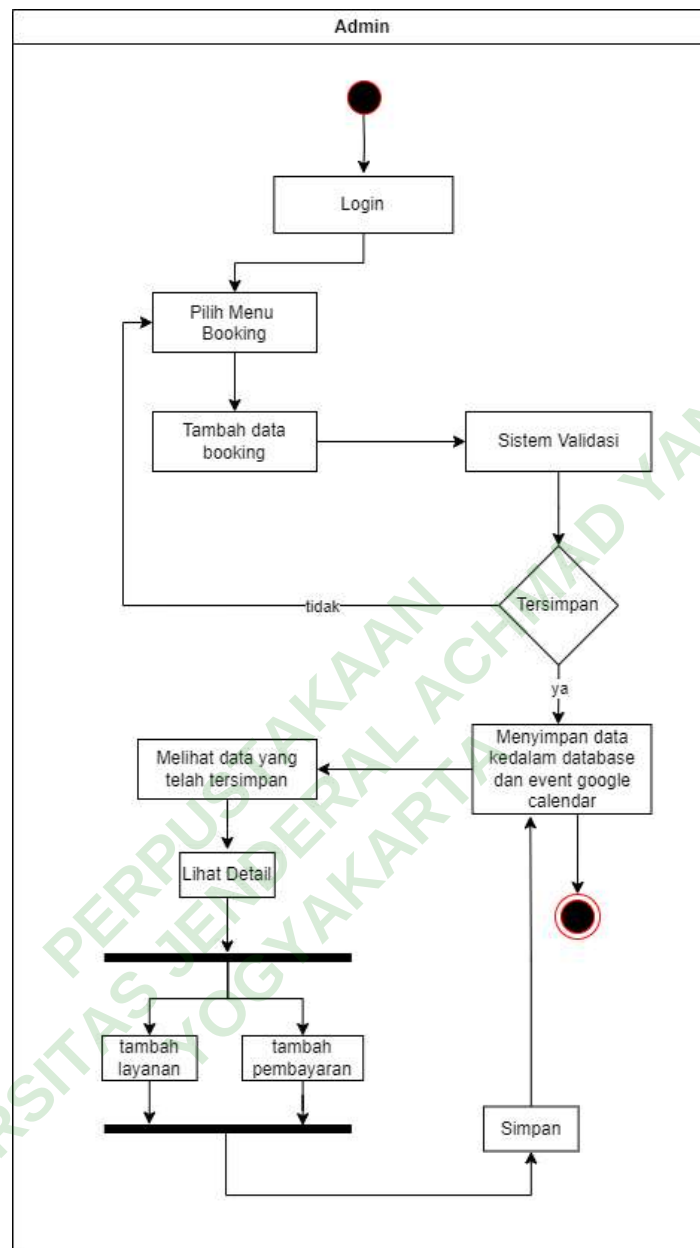
Gambar 3.5 Proses kelola tag

Sebelum melakukan kelola tag layanan maka admin harus berhasil melalui autentikasi *login* seperti Gambar 3.5. Proses kelola tag bertujuan untuk menampilkan list tag, melakukan edit tag, menambah tag dan menghapus tag tertentu. Sebelum melakukan penyimpanan proses tersebut maka sistem akan memvalidasi data yang diinputkan jika benar maka akan menampilkan perubahan atau penambahan yang dilakukan. Dan jika gagal maka akan kembali menampilkan list tag tanpa perubahan.



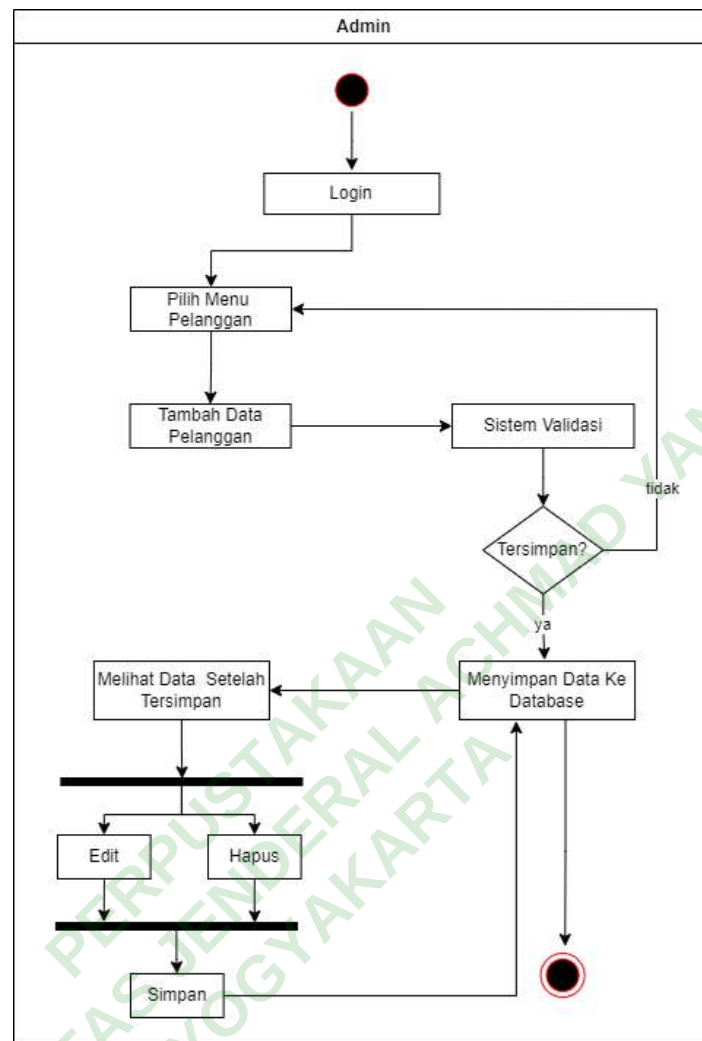
Gambar 3.6 Proses kelola layanan

Untuk mengakses halaman layanan maka perlu melalui proses harus berhasil melalui autentikasi *login* terlihat di Gambar 3.6. Proses kelola layanan bertujuan untuk menampilkan list layanan, melakukan edit layanan, menambah layanan dan menghapus layanan tertentu. Sebelum melakukan penyimpanan proses tersebut maka sistem akan memvalidasi data yang diinputkan jika benar maka akan menampilkan perubahan atau penambahan yang dilakukan. Dan jika gagal maka akan kembali menampilkan list layanan tanpa perubahan.



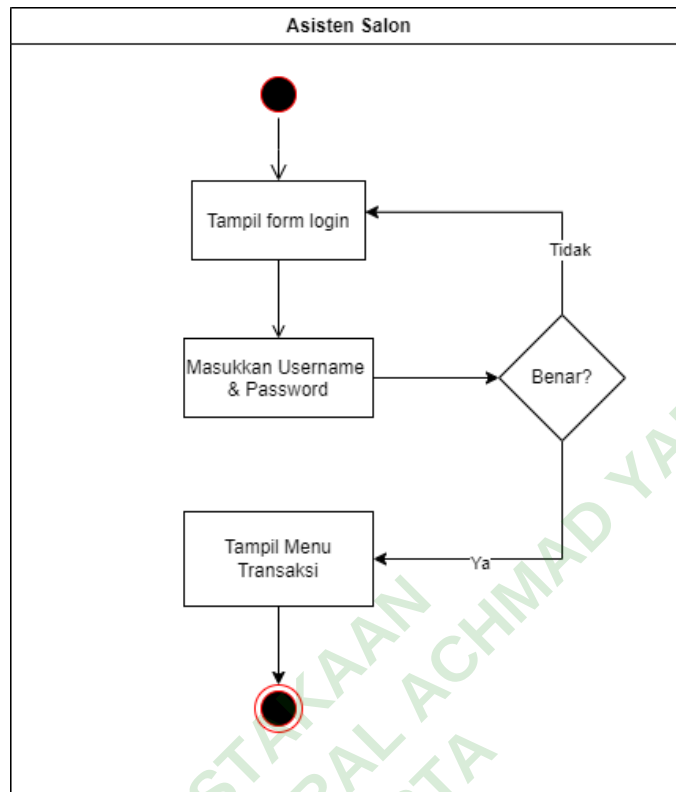
Gambar 3.7 Proses kelola *booking*

Untuk mengakses halaman *booking* maka perlu melalui proses harus berhasil melalui autentikasi *login* terlihat di Gambar 3.7. Proses kelola halaman *booking* bertujuan untuk menampilkan riwayat *booking*, melakukan penambahan *booking*, melakukan *edit booking*. Jika berhasil menambah *booking* maka dapat melihat detail yang mencakup proses *edit*, tambah dan hapus layanan tertentu yang di telah *booking*. Namun jika proses penambahan *booking* gagal maka akan kembali ke menampilkan list *booking* tanpa perubahan.



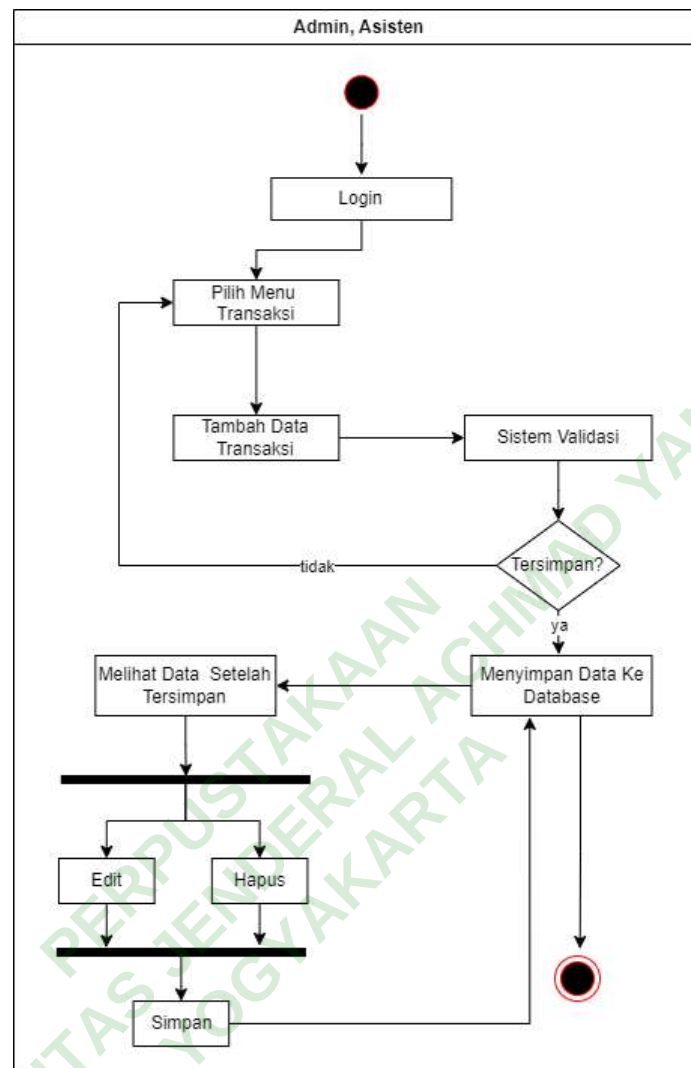
Gambar 3.8 Proses kelola data pelanggan

Pengelolaan data pelanggan harus melalui proses harus berhasil melalui autentikasi *login* seperti pada Gambar 3.8. Pada menu pelanggan akan menampilkan list data pelanggan yang telah berhasil melakukan *booking*. Data tersebut dapat dikelola seperti menambahkan data pelanggan baru, melakukan edit pelanggan dan menghapus data pelanggan. Apabila proses tersebut berhasil dilakukan maka akan kembali ke halaman pelanggan dengan perubahan yang dilakukan. Namun, apabila gagal maka akan menampilkan halaman pelanggan tanpa perubahan.



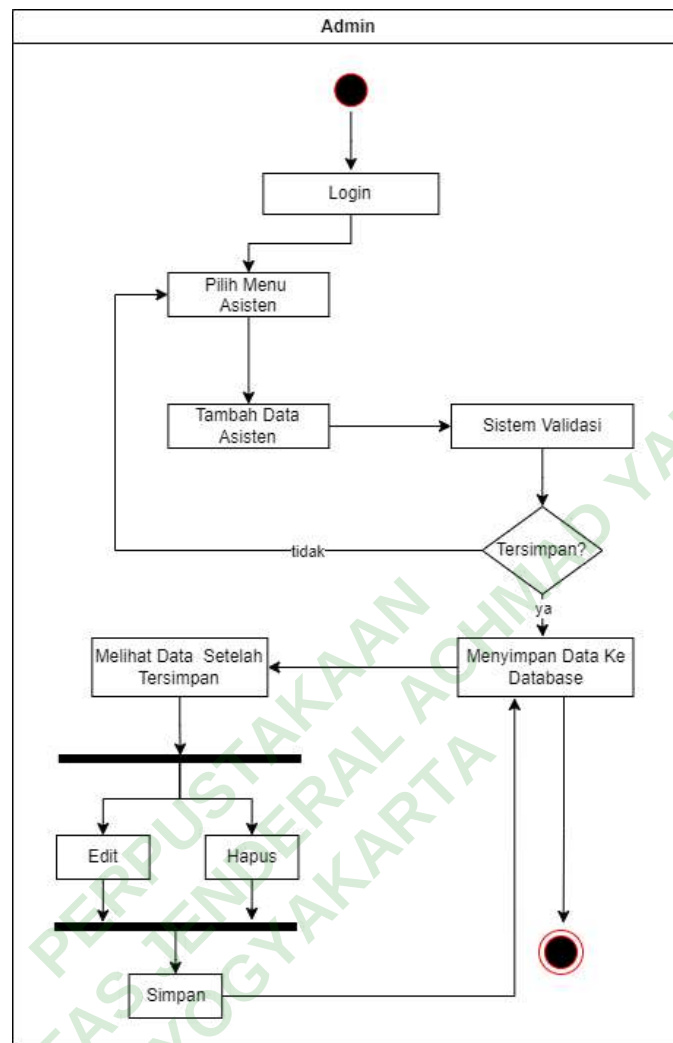
Gambar 3.9 Proses *login* asisten

Pada *activity* proses *login* seperti terlihat pada Gambar 3.9 menjelaskan alur asisten agar dapat mengakses atau masuk kedalam sistem. Asisten akan menginputkan *username* serta *password* yang terdaftar kemudian melakukan *login*. Apabila terdapat kesalahan saat penginputan *username* atau *password* maka akan tampil pesan *error* dan tidak dapat mengakses halaman sistem. Namun, jika *username* serta *password* yang diinput *valid* maka berhasil melakukan *login* serta akan menampilkan halaman transaksi.



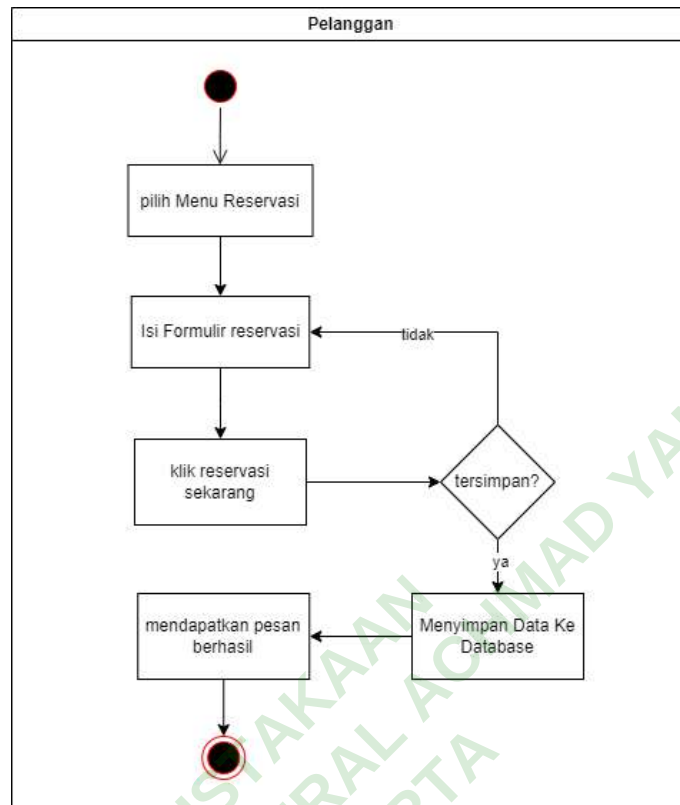
Gambar 3.10 Proses kelola data transaksi

Proses kelola data transaksi harus melalui proses autentikasi *login* seperti Gambar 3.10. Menu transaksi akan menampilkan list transaksi yang berhasil disimpan. Pada menu tersebut admin maupun asisten salon dapat mengelola data transaksi seperti menambahkan data transaksi baru, melakukan edit transaksi dan menghapus data transaksi. Apabila proses tersebut berhasil dilakukan maka akan kembali ke halaman list transaksi dengan perubahan yang dilakukan. Namun, apabila gagal maka akan menampilkan halaman transaksi tanpa perubahan.



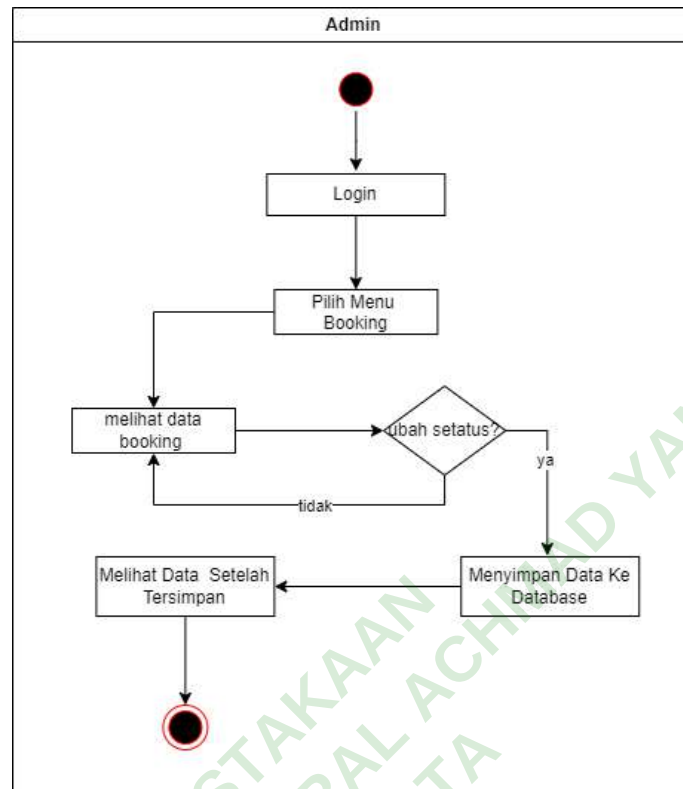
Gambar 3.11 Proses kelola data asisten

Proses kelola data asisten harus melalui proses autentikasi *login* seperti Gambar 3.11. Menu asisten bertujuan untuk menampilkan list asisten salon saat ini. Pada menu ini admin berhak melakukan penambahan data asisten baru, ubah data asisten, dan menghapus data asisten tertentu. Menu asisten juga memiliki apabila proses tersebut berhasil maka admin dapat melihat detail asisten. Jika proses gagal maka akan kembali menampilkan list data asisten



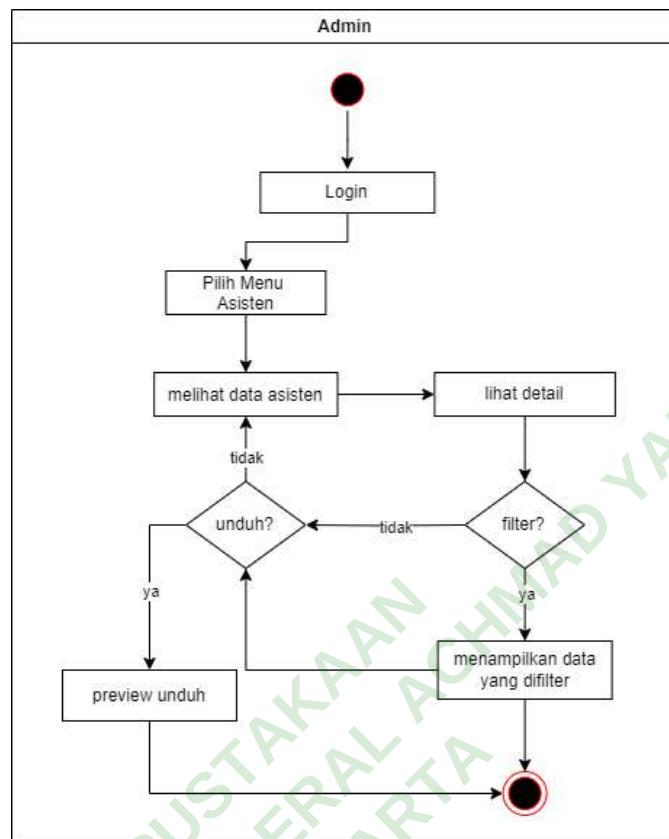
Gambar 3.12 Proses kelola data reservasi pelanggan

Pengelolaan data reservasi dimulai dengan pelanggan terlebih dahulu memilih menu reservasi pada tampilan sistem seperti terlihat pada Gambar 3.12. Setelah itu, maka sistem akan menampilkan kalender dengan informasi jadwal atau tanggal yang tersedia serta formulir reservasi. Pelanggan harus mengisi formulir reservasi apabila ingin melakukan reservasi sesuai dengan tanggal yang tersedia. Jika formulir telah dilengkapi maka pelanggan dapat melakukan *submit*. Apabila data yang diisikan sesuai maka proses *submit* berhasil dilakukan dan pesanan akan diproses oleh admin. Namun, jika terdapat kesalahan data atau tidak sesuai dan melakukan *submit* maka akan menampilkan halaman reservasi kembali.



Gambar 3.13 Proses kelola pembatalan jadwal

Pada proses kelola pembatalan jadwal, admin menerima pembatalan pesanan seperti terlihat pada Gambar 3.13. Setelah itu, admin melakukan *login* ke sistem, apabila berhasil akan menampilkan menu booking dan jika gagal maka melakukan *login* kembali. Setelah sistem menampilkan halaman list *booking* maka admin melakukan perubahan pada status *booking* menjadi batal. Apabila proses perubahan berhasil maka akan diarahkan kembali ke halaman list *booking*.

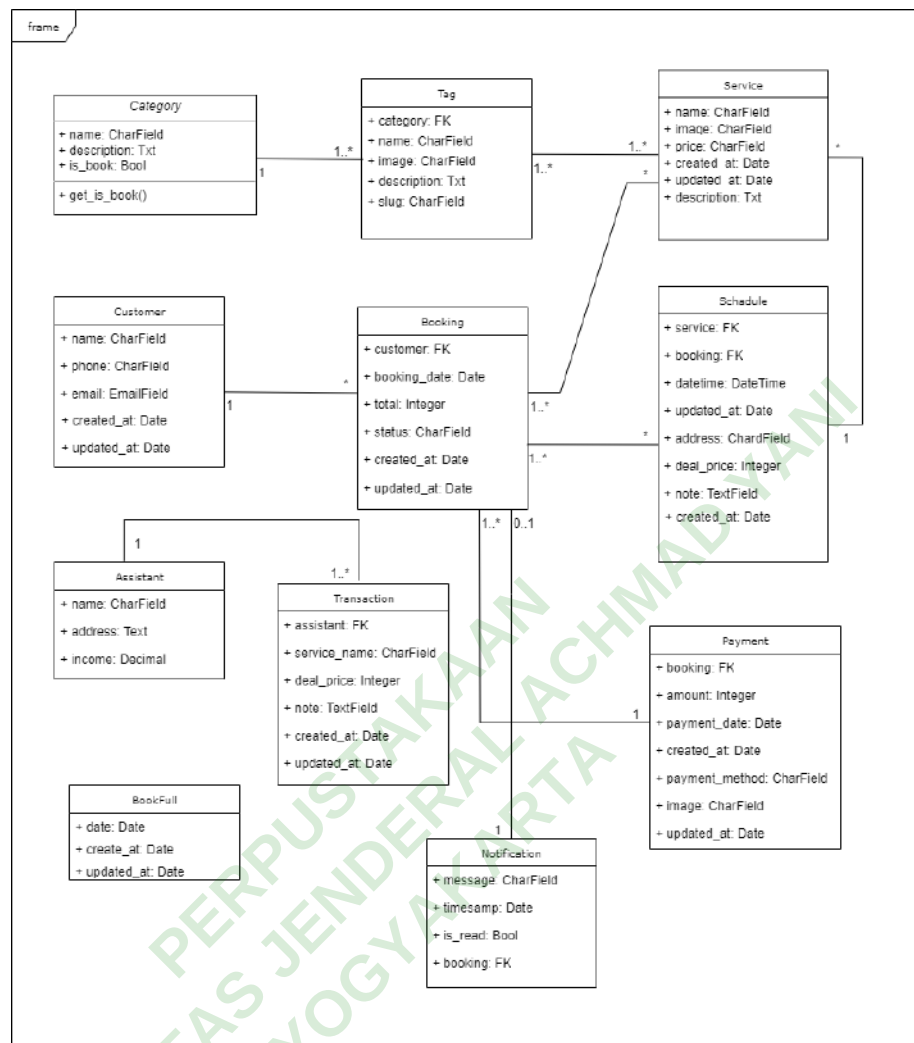


Gambar 3.14 Proses kelola detail asisten

Dalam proses kelola detail asisten salon, admin akan mendapatkan dapat melakukan *filter* data transaksi terkait asisten dalam rentang waktu tertentu seperti terlihat pada Gambar 3.14. Data yang di *filter* akan ditampilkan dalam tabel lalu admin juga dapat melakukan unduh data tersebut.

3.4.3 Class Diagram

Perancangan *class diagram* untuk menjelaskan hubungan objek-objek pada aplikasi yang akan dikembangkan dengan menunjukkan kelas-kelas, atribut dan metodenya. Hal ini akan mempermudah dalam proses implementasi sistem. Rancangan *class diagram* terlihat di Gambar 3.15.



Gambar 3.15 Class diagram

Class diagram digunakan dalam proses implementasi sistem pada pengembangan basis data yang akan digunakan. Rincian dan penjelasan terkait class diagram dapat dicermati pada Tabel 3.2.

Tabel 3.2 Penjelasan Class Diagram

No	Nama Class	Keterangan
1	Category	Class category yaitu class yang akan digunakan dalam pengelolaan kategori layanan salon.
2	Tag	Sebuah class untuk pengelolaan data tag yang memiliki relasi

		dengan <i>class</i> kategori.
3	<i>Service</i>	<i>Class</i> yang digunakan untuk pengelolaan data layanan salon dan memiliki relasi dengan <i>class</i> tag.
4	<i>Customer</i>	<i>Class customer</i> digunakan untuk pengelolaan data pelanggan dan memiliki relasi dengan <i>class</i> booking.
5	<i>Booking</i>	<i>Class booking</i> digunakan dalam pengelolaan data <i>booking</i> dengan relasi pada <i>class</i> service, payment dan notification.
6	<i>Schedule</i>	<i>Class schedule</i> merupakan <i>class</i> yang terbentuk dari relasi booking dan service untuk mengelola data pemesanan jadwal dan layanan.
7	<i>Assistant</i>	<i>Class assistant</i> digunakan untuk mengelola data asisten salon dan memiliki relasi dengan <i>class</i> transaction.
8	<i>Transaction</i>	<i>Class transaction</i> merupakan <i>class</i> yang digunakan dalam pengelolaan data transaksi.
9	<i>Payment</i>	<i>Class payment</i> yaitu <i>class</i> yang bertujuan untuk mengelola data pembayaran reservasi dan berelasi dengan <i>class</i> booking.
10	<i>Notification</i>	<i>Class notification</i> digunakan untuk mengelola data pesanan yang masuk ke admin dalam bentuk pesan notifikasi.
11	<i>Book Full</i>	<i>Class book full</i> digunakan untuk menyimpan tanggal tertentu sebagai informasi jadwal operasi salon.

3.5 PERANCANGAN DATABASE

Dalam mendesain *database*, perlu ditentukan batasan-batasan kolom, baris dan tipe data yang akan digunakan sesuai dengan keperluan. Berdasarkan

perancangan *class diagram* maka diperoleh rancangan *database* untuk implementasi aplikasi ini.

3.5.1 Tabel Kategori

Tabel Kategori yaitu tabel yang akan menampung kategori setiap layanan yang ada di Novy Salon. Tabel Kategori mempunyai beberapa kolom dengan *category_id* sebagai *primary key*. Struktur dan kegunaan kolom pada Tabel kategori seperti ditunjukkan pada Tabel 3.3.

Tabel 3.3 Struktur Tabel Kategori

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
Category_id (PK)	bigint	None	No	20	Menampung id dari kategori
name	varchar	None	No	225	Nama kategori produk
is_book	boolean	False	No	1	untuk penanda bahwa kategorinya itu bisa di <i>booking</i>
description	longtext	None	Yes		Berisi penjelasan dari kategori

3.5.2 Tabel Layanan

Tabel Layanan yaitu tabel yang digunakan untuk menampung layanan yang ada di Novy Salon. Tabel Layanan mempunyai beberapa kolom dengan *service_id* sebagai *primary key*. Struktur Tabel layanan seperti ditunjukkan pada Tabel 3.4

Tabel 3.4 Struktur Tabel Layanan

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
Service_id (PK)	bigint	None	No	20	Menampung id Service yang tersedia
tag_id (FK)	int	None	No	11	berelasi dengan Tabel tag
name	varchar	None	No	225	Nama layanan
image	varchar	None		225	Untuk menampung gambar

slug	varchar	None	No	225	Menampung url dari id atau nama layanan
description	longtext	None	Yes		Berisi penjelasan dari layanan
created_at	timestamp	None	Yes		Tanggal kategori ditambahkan
updated_at	timestamp	None	Yes		Tanggal kategori terakhir kali diubah

3.5.3 Tabel Tags

Tabel *Tags* adalah tabel yang digunakan menampung tag yang terkait dengan kategori layanan. Tabel *Tags* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *tags* seperti ditunjukkan pada Tabel 3.5.

Tabel 3.5 Struktur Tabel *Tags*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	int	None	No	11	Menampung id tag yang tersedia
Name	varchar	None	No	225	Nama tag
Slug	varchar	None	No	225	Menampung url dari id atau name tag
category_id (FK)	bigint	None	No	20	menjadi foreign key dengan Tabel categories
image	varchar	None		225	Untuk menampung tag gambar
description	longtext	None	Yes		Berisi penjelasan dari Tag

3.5.4 Tabel Customers

Tabel *Customers* yaitu tabel yang digunakan menampung informasi tentang pelanggan yang berhasil melakukan *booking*. Tabel *Customers* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *customers* seperti ditunjukkan pada Tabel 3.6.

Tabel 3.6 Struktur Tabel *Customers*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	int	None	No	11	Menampung id pelanggan
name	varchar	None	No	225	Untuk menampung nama pelanggan
no_telp	int	None	No	15	untuk menampung no telp pelanggan
email	varchar	None	Yes	225	untuk menampung email pelanggan
created_at	timestamp	None	Yes		Tanggal data pelanggan ditambahkan
updated_at	timestamp	None	Yes		Tanggal data pelanggan terakhir kali diubah

3.5.5 Tabel *Bookings*

Tabel *Bookings* yaitu tabel untuk menampung informasi *booking* pelanggan yang memiliki keterkaitan dengan jadwal. Tabel *Bookings* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *bookings* seperti ditunjukkan pada Tabel 3.7. Tabel *booking* akan berelasi dengan tabel *service*, tabel *payment* dan juga tabel *customer*. Hal ini dikarenakan pada formulir *booking* akan menyimpan data *customer*, data pembayaran dan data layanan yang di *booking*.

Tabel 3.7 Struktur Tabel *Bookings*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	bigint	None	No	20	Menampung id <i>bookings</i>
booking_date	date	None	No		Untuk menampung tanggal <i>booking</i>
customer_id (FK)	int	None	No	11	untuk memanggil atribut dari Tabel <i>customer</i>
total	int	None	No		untuk menampung total tagihan

status	enum	belum lunas	No		untuk menampung status <i>booking</i> (lunas, belum lunas, batal)
updated_at	timestamp	None	Yes		Tanggal <i>booking</i> terakhir kali diubah

3.5.6 Tabel *Payments*

Tabel *Payments* yaitu tabel untuk menampung informasi pembayaran dari setiap *booking* yang berhasil dilakukan pelanggan. Tabel *Payments* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *payments* seperti ditunjukkan pada Tabel 3.8.

Tabel 3.8 Struktur Tabel *Payments*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	bigint	None	No	20	Menampung id dari <i>payment</i>
booking_id (FK)	bigint	None	No	20	relasi terhadap Tabel <i>bookings</i>
amount_paid	int	None	No		jumlah yang dibayar
payment_date	timestamp	None	No		tanggal pembayaran
payment_method	enum	transfer	No		untuk menampung metode pembayaran

3.5.7 Tabel *Schedules*

Tabel *Schedules* yaitu tabel yang digunakan untuk menampung informasi jadwal dari *booking* yang terkait dengan layanan yang di *booking*. Tabel *Schedules* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *schedules* seperti ditunjukkan pada Tabel 3.9.

Tabel 3.9 Struktur Tabel *Schedules*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	int	None	No	11	Menampung id dari jadwal
Service_id (FK)	bigint	None	No	20	relasi terhadap Tabel <i>service</i>
booking_id (FK)	bigint	None	No	20	relasi terhadap Tabel <i>booking</i>
address	longtext	None	No		untuk menampung alamat pelaksanaan layanan
datetime	date	None	No		untuk menampung tanggal pelaksanaan layanan
deal_price	int	None	No		untuk menampung harga deal tiap layanan yang dipesan
Note	longtext	None	Yes		untuk menampung catatan layanan yang dipesan
created_at	timestamp	None	Yes		Tanggal kategori ditambahkan
updated_at	timestamp	None	Yes		Tanggal kategori terakhir kali diubah

3.5.8 Tabel *Assistants*

Tabel *Assistants* adalah tabel yang digunakan menampung informasi asisten salon. Tabel *Assistant* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *assistants* seperti ditunjukkan pada Tabel 3.10.

Tabel 3.10 Struktur Tabel *Assistants*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	int	None	No	11	Menampung id dari asisten
name	varchar	None	No	50	Menampung nama asisten salon
address	longtext	None	No		untuk menampung alamat asisten
income	decimal	None	Yes		untuk menampung pendapatan asisten

3.5.9 Tabel *Transactions*

Tabel *Transactions* yaitu tabel yang digunakan menampung informasi transaksi yang ada di salon sesuai dengan asisten salon yang mengerjakan. Tabel *transactions* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *transactions* seperti ditunjukkan pada Tabel 3.11.

Tabel 3.11 Struktur Tabel *Transactions*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	int	None	No	11	Menampung id dari <i>transactions</i>
assistant_id (FK)	int	None	No	11	relasi terhadap Tabel <i>assistants</i>
Service_name	varchar	None	No	100	Menampung nama layanan
Deal_price	Int	None	No		Menampung harga deal setiap layanan
created_at	timestamp	None	Yes		Tanggal transaksi ditambahkan
updated_at	timestamp	None	Yes		Tanggal transaksi terakhir kali diubah

3.5.10 Tabel *Book Full*

Tabel *book full* merupakan tabel yang akan menampung informasi terkait jadwal tertentu sebagai informasi kepada customer bahwa tanggal tersebut tersedia atau tidak. Tabel *book full* mempunyai beberapa kolom dengan id sebagai *primary key*. Struktur Tabel *book full* seperti ditunjukkan pada Tabel 3.12.

Tabel 3.12 Struktur Tabel *Book Full*

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	int	None	No	11	Menampung id dari <i>book full</i>
date	date	None	No		Menampung tanggal
created_at	timestamp	None	Yes		Tanggal <i>book full</i> ditambahkan
updated_at	timestamp	None	Yes		Tanggal <i>book full</i> terakhir kali diperbaharui

3.5.11 Tabel Notifikasi

Tabel notifikasi yaitu tabel yang digunakan untuk menampung informasi notifikasi reservasi layanan yang ada di salon. Saat pelanggan berhasil melakukan reservasi maka di bagian admin akan muncul notifikasi. Tabel notifikasi mempunyai berelasi dengan *booking* sehingga *booking_id* menjadi *foreign key* yang akan menampung id booking dari reservasi pelanggan. Struktur Tabel notifikasi seperti ditunjukkan pada Tabel 3.13.

Tabel 3.13 Struktur Tabel Notifikasi

Nama Kolom	Tipe Data	Default	Null	Lebar	Kegunaan
id (PK)	bigint	None	No	20	Menampung id dari <i>notifications</i>
booking_id (FK)	bigint	None	No	20	relasi terhadap Tabel <i>booking</i>
message	varchar	None	No	255	Menampung pesan

					notifikasi
Is_read	tinyint	0	No	1	Menampung status notifikasi apakah sudah dibaca atau tidak
timestamp	timestamp	None	No		Tanggal notifikasi ditambahkan

3.6 PERANCANGAN DESAIN ANTARMUKA PENGGUNA

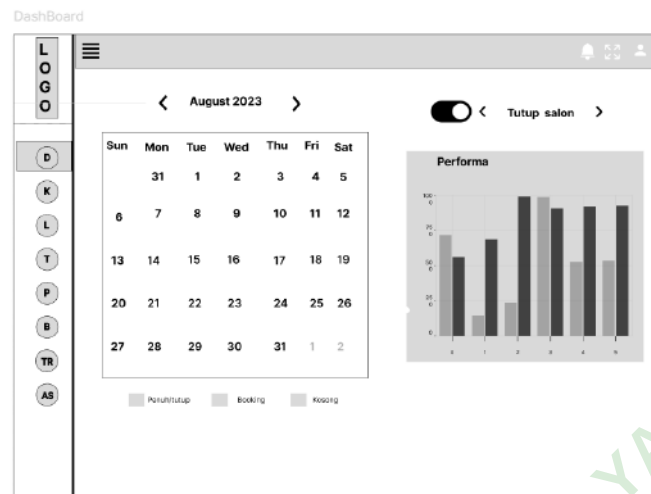
3.6.1 Halaman Login

Tujuan halaman *login* yaitu memungkinkan admin atau asisten salon mengakses halaman aplikasi. Pada halaman ini, admin atau asisten perlu memasukkan kombinasi *username* serta *password valid* yang sesuai dengan data terdaftar dalam basis data, sebagai syarat untuk halaman utama. Detail desain halaman *login* tersebut dicermati dalam Gambar 3.16.

Gambar 3.16 Halaman login

3.6.2 Halaman Dashboard

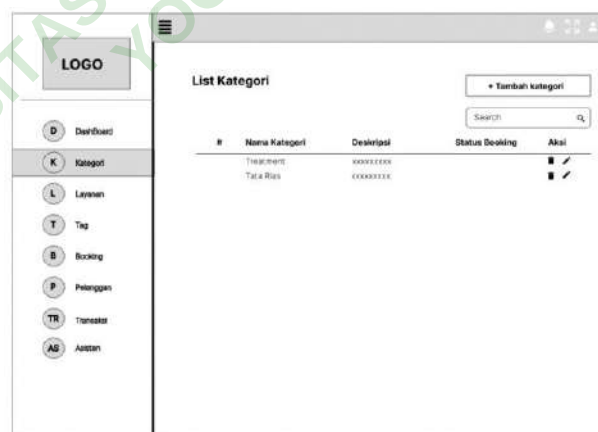
Halaman ini bertujuan untuk memungkinkan admin mengelola jadwal salon dan waktu salon beroperasi. Di halaman *dashboard*, admin harus memilih tanggal tertentu yang kemudian akan menampilkan detail *booking* pada tanggal tersebut. Kalender tanggal akan berubah warna sesuai kondisi yang disesuaikan berdasarkan apakah tanggal tersebut sudah penuh atau masih kosong. Rancangan halaman *dashboard* terlihat di Gambar 3.17



Gambar 3.17 Halaman dashboard admin

3.6.3 Halaman Kategori

Tujuan halaman kategori yaitu memungkinkan admin mengelola data kategori. Halaman kategori ini bertujuan untuk mengelompokkan kategori tag layanan salon. Pada halaman ini, admin dapat menambah kategori, mengedit kategori serta menghapus kategori. Apabila ingin mencari kategori tertentu dapat menginputkan kata kunci pada kolom pencarian. Detail desain halaman kategori dapat dicermati dalam Gambar 3.18.

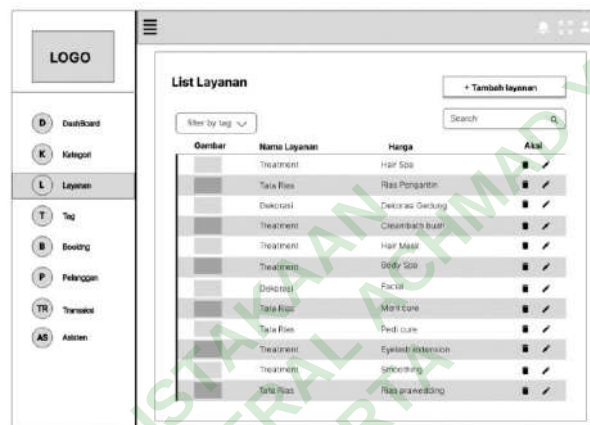


Gambar 3.18 Halaman kategori

3.6.4 Halaman Layanan

Tujuan halaman layanan yaitu memungkinkan admin mengelola data layanan salon. Halaman ini bertujuan untuk mengelompokkan layanan berdasarkan

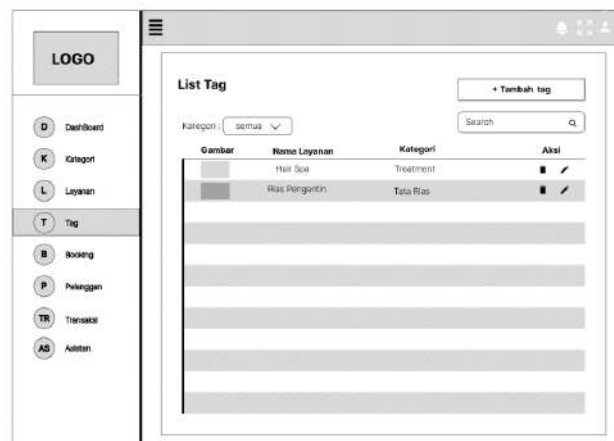
tag, dimana satu tag dapat memiliki banyak layanan. Admin dapat melakukan pengelolaan data dengan menambah layanan, mengedit layanan serta menghapus layanan. Apabila ingin mencari layanan tertentu dapat dilakukan dengan menginputkan kata kunci pada kolom pencarian. Halaman ini juga menyediakan *filter by tag* yang akan memudahkan admin dalam mengetahui jenis layanan apa saja yang terkait dengan tag tersebut. Detail desain halaman layanan dapat dicermati dalam Gambar 3.19.



Gambar 3.19 Halaman layanan

3.6.5 Halaman Tag

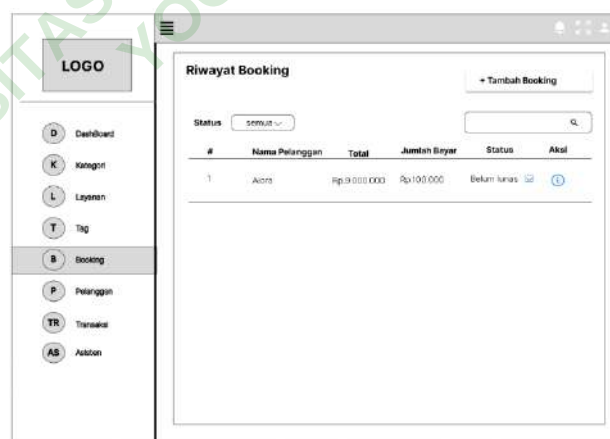
Tujuan halaman tag yaitu memungkinkan admin mengelola data pada tag. Satu kategori dapat memiliki banyak tag sehingga pada halaman tag terdapat *filter* kategori untuk memudahkan admin dalam mengetahui kategori tag. Dalam hal pengelolaan admin dapat menambah tag, mengedit tag serta menghapus tag. Apabila ingin mencari tag tertentu dapat dilakukan dengan menginputkan kata kunci pada kolom pencarian. Detail desain tampilan halaman tag dapat dicermati dalam Gambar 3.20.



Gambar 3.20 Halaman tag

3.6.6 Halaman Booking

Tujuan halaman *booking* yaitu memungkinkan admin mengelola data pesanan pelanggan. Pengelolaan data pada *booking* meliputi penambahan *booking* baru, melakukan filter *booking* berdasarkan status booking. Pada menu aksi terdapat ikon (i) yang apabila di klik akan diarahkan ke halaman detail *booking*. Apabila ingin mencari *booking* tertentu dapat dilakukan dengan menginputkan kata kunci pada kolom pencarian. Detail desain tampilan halaman *booking* terlihat di Gambar 3.21.

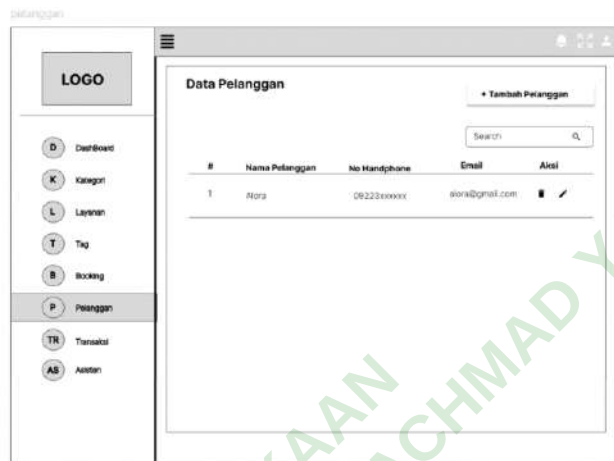


Gambar 3.21 Halaman booking

3.6.7 Halaman Pelanggan

Tujuan halaman pelanggan yaitu untuk mengelola data pelanggan. Pengelolaan data pada menu pelanggan meliputi penambahan pelanggan baru,

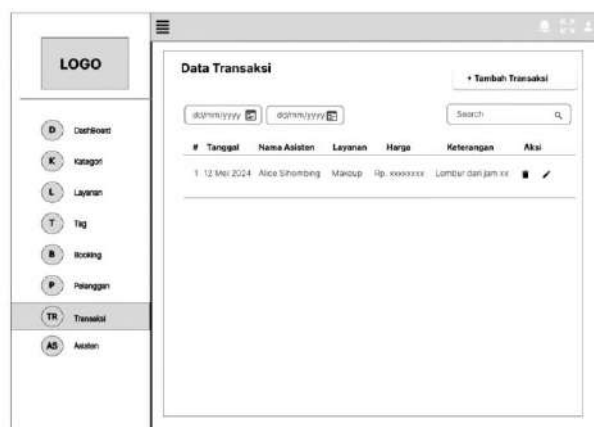
melakukan pengeditan data pelanggan dan menghapus data pelanggan tertentu. Apabila ingin mencari data pelanggan tertentu dapat dilakukan dengan menginputkan kata kunci pada kolom pencarian. Detail desain tampilan halaman pelanggan terlihat di Gambar 3.22.



Gambar 3.22 Halaman data pelanggan

3.6.8 Halaman Transaksi

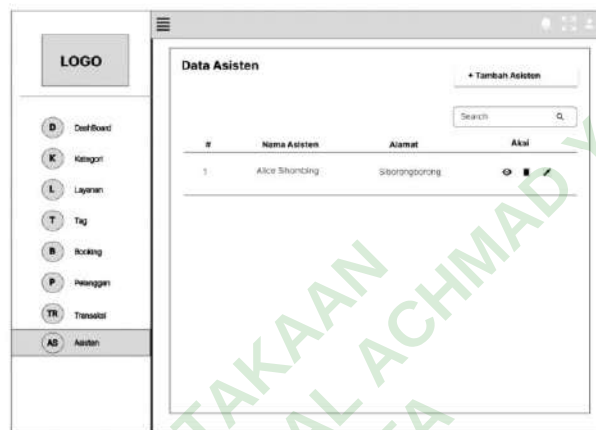
Halaman ini digunakan dalam hal pengelolaan data transaksi. Pengelolaan data diantaranya penambahan transaksi baru, melakukan pengeditan data dan menghapus data transaksi tertentu. Apabila ingin mencari data transaksi tertentu dapat dilakukan dengan menginputkan kata kunci pada kolom pencarian. Detail desain tampilan halaman transaksi dicermati di Gambar 3.23.



Gambar 3.23 Halaman data transaksi

3.6.9 Halaman Asisten

Halaman asisten digunakan untuk mengelola data asisten salon. Pengelolaan data meliputi penambahan data asisten baru, melakukan pengeditan data dan menghapus data asisten tertentu. Apabila ingin mencari data asisten tertentu dapat dilakukan dengan menginputkan kata kunci pada kolom pencarian. Detail desain tampilan halaman asisten dicermati di Gambar 3.24.



Gambar 3.24 Halaman data asisten

3.6.10 Halaman Beranda

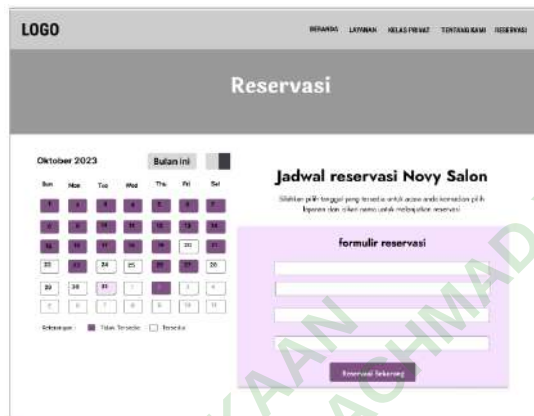
Beranda bertujuan untuk menampilkan informasi tentang salon. Dalam halaman ini terdapat ringkasan dari beberapa menu yang ada agar informasi dapat diperoleh dengan mudah dan cepat. Untuk tampilan detail dari ringkasan menu dapat diakses pada menu navigasi. Rancangan halaman beranda yang akan digunakan dapat dicermati dalam Gambar 3.25.



Gambar 3.25 Halaman beranda

3.6.11 Halaman Reservasi

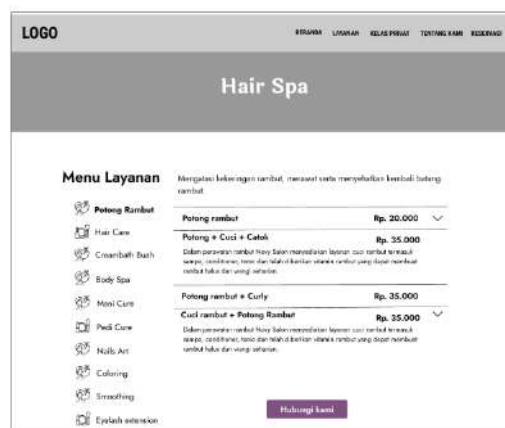
Halaman Reservasi bertujuan untuk menampilkan jadwal yang penuh dan kosong yang dapat di pesan oleh pelanggan. Pada halaman ini, pelanggan harus memilih tanggal yang masih tersedia dan mengisi formulir reservasi yang akan disediakan. Rancangan halaman reservasi ditunjukkan oleh Gambar 3.26.



Gambar 3.26 Halaman reservasi

3.6.12 Halaman LayananPelanggan

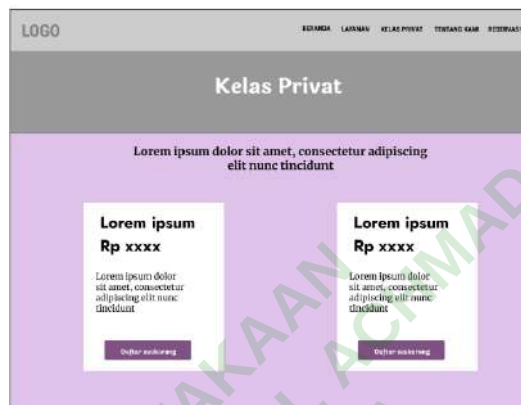
Halaman user layanan akan menampilkan daftar layanan salon. Pada halaman ini pengguna dapat mengakses deskripsi layanan dan juga gambar terkait layanan. Apabila ingin mengajukan pertanyaan maka pelanggan dapat menekan tombol hubungi kami yang akan langsung diarahkan halaman wa admin salon. Tampilan detail halaman layanan seperti pada Gambar 3.27



Gambar 3.27 Halaman layanan

3.6.13 Halaman Kelas Privat

Halaman kelas privat bertujuan untuk menampilkan daftar kelas privat salon. Pada halaman ini, pelanggan dapat mengakses detail privat serta ketentuan mengikuti kelas privat. Apabila pelanggan tertarik untuk mengikuti kelas privat dapat menekan tombol daftar sekarang. Detail tampilan kelas privat terlihat pada Gambar 3.28.



Gambar 3.28 Halaman kelas privat

3.6.14 Halaman Tentang Novy

Halaman tentang Novy merupakan tampilan halaman informasi tentang Novy salon. Dalam halaman ini terdapat sejumlah informasi yang dapat diakses pelanggan meliputi pemilik salon, serta informasi alamat salon. Tampilan desain halaman tentang Novy dapat dicermati di Gambar 3.29



Gambar 3.29 Halaman tentang Novy

3.7 RENCANA PENGUJIAN (*TESTING APLIKASI*)

Pada tahap pengujian, aplikasi akan diuji dengan metode *black box* serta *User Acceptance Test*. *Black box testing* akan dilakukan oleh Pakar (2 orang) dan Admin (1 orang) serta mengisi hasilnya pada formulir penilaian yang telah disiapkan. Salah satu test case (kasus uji) yang akan dilakukan dapat dicermati pada Tabel 3.14 dan untuk selengkapnya terdapat pada bagian lampiran.

Tabel 3.14 Kasus Uji Pada Fitur *Booking*

Fungsi yang di uji	Kasus uji	Hasil yang diharapkan	Hasil uji	
			Sesuai	Tidak Sesuai
Data booking	Tekan menu booking	Berhasil menampilkan halaman booking		
Tambah booking	Tekan tombol tambah booking	Berhasil menampilkan halaman tambah booking		
	Mengisi data formulir booking yang sesuai	Menampilkan pesan validasi jika terdapat kesalahan input		
Tambah booking	Tekan tombol tambah layanan	Menampilkan formulir layanan		
	Tekan ikon hapus pada layanan	Menghapus formulir layanan		
Detail booking	Tekan ikon detail	Menampilkan detail data booking		
Tambah layanan booking	Tekan tombol tambah layanan di halaman detail booking, lalu isi formulir dan tekan tombol simpan	Menampilkan modal tambah data layanan dan menyimpan data.		
Edit layanan booking	Tekan ikon edit layanan di halaman detail booking, lalu isi data formulir dan tekan tombol simpan	Berhasil menampilkan modal edit layanan dan menyimpan perubahan data		

Hapus data layanan	Tekan ikon hapus pada data layanan di halaman detail booking	Menampilkan pesan konfirmasi penghapusan data		
	Tekan ok pada pesan konfirmasi	Berhasil menghapus data dari tabel detail layanan booking		
	Tekan cancel pada pesan konfirmasi	Kembali ke halaman detail booking tanpa penghapusan data		
Tambah pembayaran	Tekan tombol tambah pembayaran pada halaman detail booking, lalu isi formulir dan tekan tombol simpan	Berhasil menampilkan modal tambah pembayaran dan menyimpan data		
Edit pembayaran	Tekan ikon edit pada kolom aksi tabel pembayaran di halaman detail booking dan lakukan perubahan data lalu tekan tombol simpan	Berhasil menampilkan modal edit pembayaran dan menyimpan perubahan data.		

Pengujian *User Acceptance Test* yaitu pengujian paling akhir yang akan dilakukan sebelum merilis aplikasi. Pengujian ini agar menyakinkan pengguna apakah aplikasi sesuai dan berfungsi dengan baik. Untuk mengukur hasil pengujian digunakan skala likert pada Tabel 3.15.

Tabel 3.15 Skala Likert

Skala Jawaban	Keterangan	Skor	Persentase
SS	Sangat Setuju	5	80% - 100%
S	Setuju	4	60% - 79%
KS	Kurang Setuju	3	40% - 59%
TS	Tidak Setuju	2	20% - 39%
STS	Sangat Tidak Setuju	1	0% - 19%

Salah satu pertanyaan yang akan digunakan dalam pengujian User Acceptance Test terlihat Tabel 3.16 selengkapnya pada bagian pembahasan. Frekuensi merupakan jumlah jawaban yang diterima berdasarkan skala jawaban.

1. Apakah aplikasi pemesanan layanan tata rias dapat mempermudah dalam pengelolaan jadwal reservasi di Novy salon?

Tabel 3.16 Pertanyaan Pengujian UAT

Skala Jawaban	Skor	Frekuensi	Persentase
SS	5		
S	4		
KS	3		
TS	2		
STS	1		