

BAB 3

METODE PENELITIAN

Penelitian ini berkaitan dengan rancang-bangun. Proses penelitian dimulai dengan mempelajari dasar masalah, menggambarkan prosedur, menemukan sumber masalah, dan akhirnya merancang dan mengembangkan suatu sistem yang bertujuan untuk mengurangi atau menghilangkan masalah yang ada. Untuk merancang sistem informasi manajemen dokumen di Ormawa Universitas Jenderal Achmad Yani Yogyakarta, berikut adalah bahan, alat, metode pengembangan sistem, dan tahapan penelitian.

3.1 BAHAN DAN ALAT PENELITIAN

Dalam penelitian ini, bahan dan alat yang digunakan meliputi:

1. Data administrasi surat-menyurat Ormawa Universitas Jenderal Achmad Yani Yogyakarta, termasuk dokumen yang telah tersimpan dalam *hard disk* dan Google Drive.
2. Informasi mengenai proses bisnis yang berjalan dalam pengelolaan surat-menyurat di Ormawa Universitas Jenderal Achmad Yani Yogyakarta.

Alat yang digunakan dalam penelitian ini adalah komputer dengan spesifikasi yang memadai untuk menjalankan sistem operasi dan perangkat lunak pengembangan, serta memiliki koneksi Internet. Sistem operasi dan aplikasi yang digunakan dalam pengembangan aplikasi ini adalah:

1. Prosesor: Intel Core i5.
2. Sistem Operasi: Windows 11.
3. RAM: 12 GB.
4. Penyimpanan: HDD 1 TB dan 256 GB SSD.
5. Visual Studio Code versi 1.99.
6. Gitlab versi 17.10 untuk kontrol kode sumber,
7. Postman untuk pengujian API versi 11.41.4. jika diperlukan.
8. Browser (Google Chrome versi 145, Mozilla Firefox versi 137.0.1) untuk pengujian tampilan antarmuka aplikasi.

9. Basisdata: MySQL versi 9.0 sebagai sistem manajemen basis data (DBMS).

3.2 JALAN PENELITIAN

Metode Agile digunakan dalam penelitian ini karena sistem penomoran dokumen pada setiap organisasi kemahasiswaan (Ormawa) belum memiliki format yang baku. Dengan pendekatan *agile*, pengembangan aplikasi dapat dilakukan secara adaptif sehingga memungkinkan perawatan (*maintenance*) dan pembaruan sistem secara fleksibel jika terjadi perubahan standar penomoran di masa mendatang. Tahapan-tahapan pengembangan sistem:

1. Perencanaan
 - a. Identifikasi kebutuhan *user*, terutama pengurus Ormawa dan pihak berwenang. Menentukan kebutuhan *user* terkait pengelolaan dokumen.
 - b. Menentukan ruang lingkup sistem, seperti otomatisasi penomoran, kontrol akses, *dashboard*, dan surat digital.
 - c. Menyusun *backlog* yang berisi daftar fitur prioritas berdasarkan kebutuhan *user*.
2. Desain (Perancangan Awal)
 - a. Merancang struktur basis data untuk mendukung otomatisasi penomoran dokumen.
 - b. Mendesain antarmuka pengguna yang *user friendly* dan mudah digunakan.
 - c. Menentukan aspek keamanan dan aksesibilitas sistem.
3. Implementasi
 1. *Sprint* 1 - Pengembangan Fitur Utama:
 - a. Penomoran otomatis.
 - b. Penyimpanan dokumen digital
 - c. Autentikasi *user* dan hak akses.
 - d. Uji coba awal untuk memastikan fungsi berjalan baik.
 2. *Sprint Review* dan *Feedback*:
 - a. Mendiskusikan hasil *sprint* dengan pihak ormawa.

- b. Mencatat masukan dan identifikasi kekurangan atau perubahan kebutuhan.
- 3. *Sprint 2* – Pengembangan Fitur Lanjutan:
 - a. *Dashboard* laporan bulanan (statistik surat masuk/keluar).
 - b. Pengiriman surat digital antar Ormawa.
 - c. Melakukan pengujian internal dan perbaikan berdasarkan masukan sebelumnya.
- 4. *Sprint 3* – Penyempurnaan dan Integrasi:
 - a. Optimasi performa dan antarmuka *user*.
- 4. Pengujian
 - a. Melakukan uji coba fungsionalitas sistem dengan *user* Ormawa Universitas Jenderal Achmad Yani Yogyakarta.
 - b. Mengevaluasi efektivitas sistem.
- 5. Pemeliharaan dan Perbaikan
 - a. Mengumpulkan umpan balik *user* secara berkala.
 - b. Melakukan perbaikan dan pembaruan sistem berdasarkan hasil pengujian dan masukan *user*.

3.3 MODEL INTERAKSI

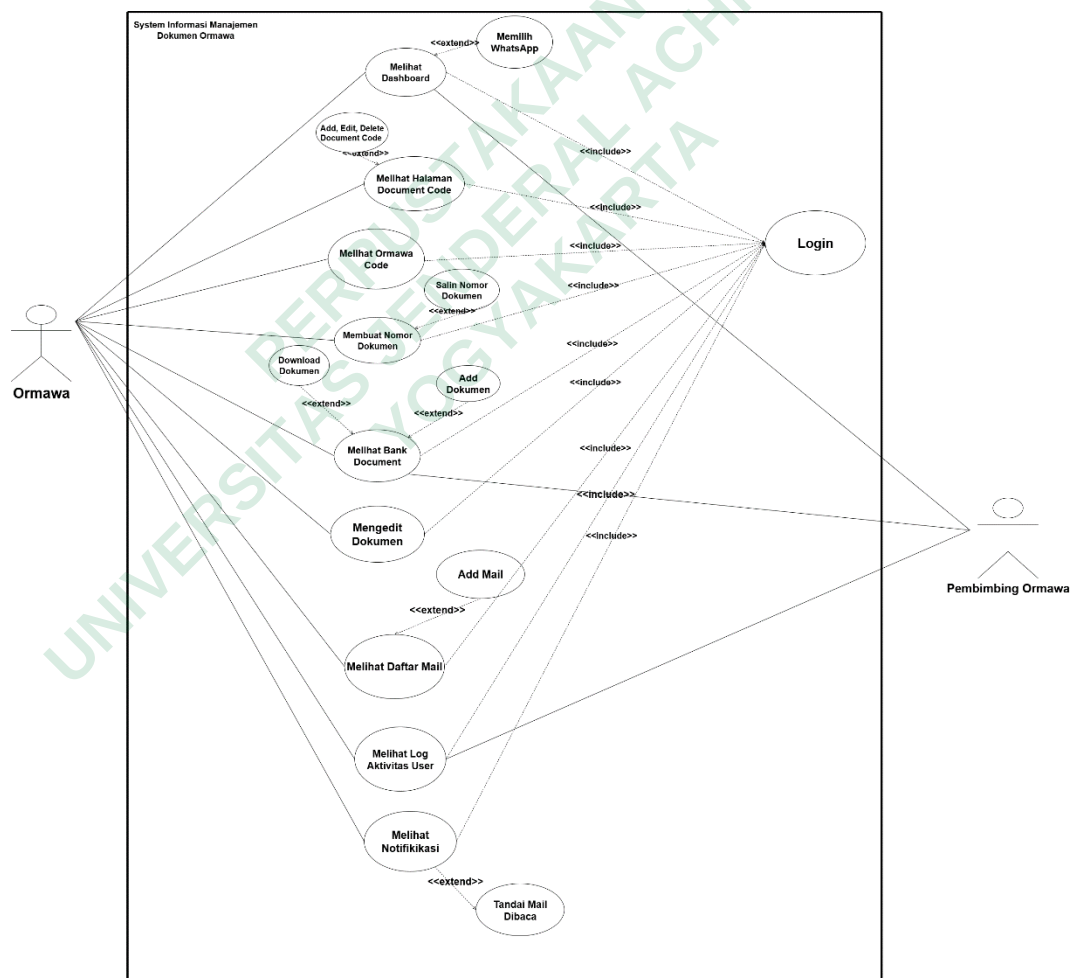
Model interaksi digunakan untuk menggambarkan bagaimana *user* berinteraksi dengan sistem, serta bagaimana alur aktivitas dalam sistem berlangsung. Dalam penelitian ini, model interaksi yang digunakan terdiri dari *Use Case Diagram* dan *Activity Diagram*. *Use Case Diagram* digunakan untuk menunjukkan hubungan antara aktor dengan fungsi-fungsi yang tersedia dalam sistem, sedangkan *Activity Diagram* digunakan untuk menggambarkan alur aktivitas atau proses bisnis yang terjadi dalam sistem secara lebih rinci dan berurutan. Kedua model ini membantu dalam memahami dan merancang sistem sesuai dengan kebutuhan *user*.

3.3.1 *Use Case Diagram*

Use Case Diagram ini menggambarkan alur interaksi dalam Sistem Informasi Manajemen Dokumen Ormawa. Terdapat dua aktor utama, yaitu

Ormawa sebagai *user* utama dan Pembimbing Ormawa sebagai *user* pendukung. Ormawa memiliki akses terhadap berbagai fitur seperti *login*, melihat *dashboard*, mengelola kode dokumen, membuat nomor dokumen, melihat dan mengubah bank dokumen, mengakses daftar surat, log aktivitas, notifikasi, hingga berbagi melalui WhatsApp. Seluruh aktivitas tersebut bergantung pada proses *login* terlebih dahulu. Sementara itu, Pembimbing Ormawa hanya memiliki akses untuk melihat notifikasi sebagai bentuk monitoring kegiatan. Diagram ini menunjukkan bagaimana sistem mendukung pengelolaan dokumen secara terstruktur dan aman berdasarkan peran masing-masing *user*.

Use Case Diagram pada Sistem Informasi Manajemen Dokumen Ormawa Unjaya dapat dilihat pada gambar 3.1.



Gambar 3.1 *Use Case Diagram*

Tabel 3.1 menjelaskan berbagai aktivitas yang dilakukan oleh aktor dalam sistem. Aktor utama dalam sistem ini adalah Ormawa, yang memiliki peran dalam mengelola dokumen organisasi mahasiswa, termasuk membuat, melihat, mengubah dokumen, serta menerima notifikasi. Selain itu, terdapat aktor pendukung yaitu Pembimbing Ormawa, yang berperan dalam memverifikasi dan mengawasi dokumen yang dikelola oleh Ormawa.

Tabel 3.1 Penjelasan *Use Case Diagram*

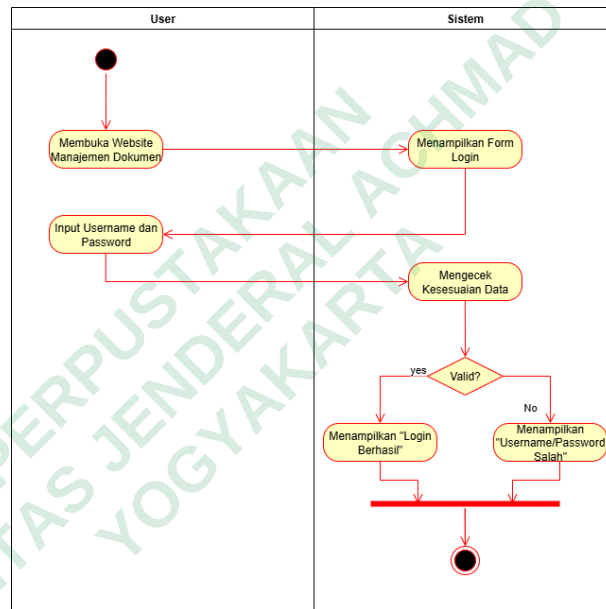
Aktor	<i>Use Case Diagram</i>	Deskripsi
Ormawa, Pembimbing Ormawa	<i>Login</i>	Proses masuk ke sistem oleh Ormawa maupun Pembimbing Ormawa. Wajib dilakukan untuk mengakses fitur lainnya.
Ormawa, Pembimbing Ormawa	Melihat <i>Dashboard</i>	Melihat informasi umum seperti statistik dokumen, aktivitas terakhir, jumlah dokumen, dan menuju halaman whatsapp.
Ormawa	Melihat Halaman Dokumen Kode	Melihat daftar kode dokumen yang tersedia, dan dapat menambahkan, <i>mengedit</i> dan menghapus <i>code document</i> .
Ormawa	Melihat Ormawa Kode	Melihat kode identifikasi untuk masing-masing Ormawa.
Ormawa	Membuat Nomor Dokumen	Membuat nomor unik untuk dokumen baru.
Ormawa, Pembimbing Ormawa	Melihat <i>Document Archives</i>	Melihat arsip dokumen yang telah disimpan dan dapat menambahkan, <i>mengedit</i> dan mendownload dokumen.
Ormawa	<i>Mengedit Dokumen</i>	Mengubah isi dokumen yang sudah ada.
Ormawa	Melihat Daftar <i>Mail</i>	Melihat daftar surat masuk dan keluar, dan dapat mengirim pesan.
Ormawa, Pembimbing Ormawa	Melihat Log Aktivitas <i>User</i>	Melihat riwayat aktivitas <i>user</i> sistem.
Ormawa	Melihat Notifikasi	Melihat pemberitahuan dari sistem ketika ada pesan masuk.

3.3.2 Activity Diagram

Berikut *Activity Diagram* yang digunakan pada Sistem Informasi manajemen Dokumen Ormawa.

1. Activity Diagram Login

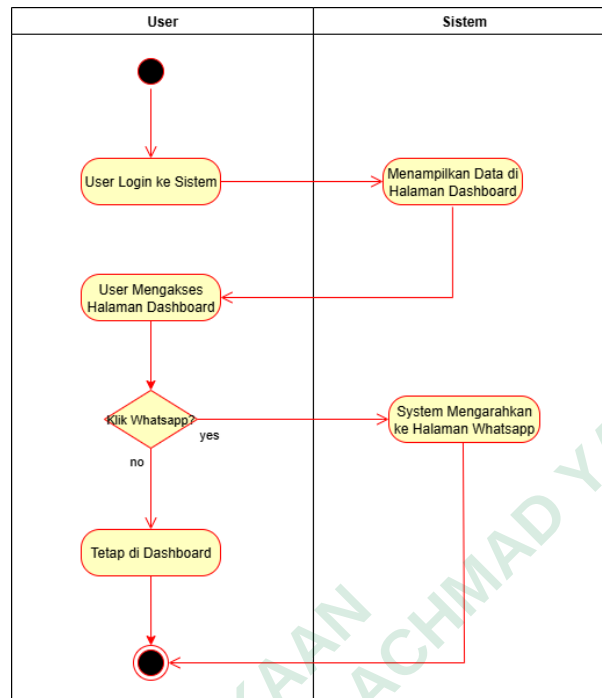
Diagram 3.2 menunjukkan alur proses *login* pada sistem manajemen dokumen. *User* membuka website, lalu sistem menampilkan form *login*. Setelah *user* mengisi *username* dan *password*, sistem memeriksa kecocokan data. Jika data valid, sistem menampilkan pesan "*Login Berhasil*", jika tidak, ditampilkan pesan "*Username/Password Salah*".



Gambar 3.2 Activity Diagram Login

2. Activity Diagram Dashboard

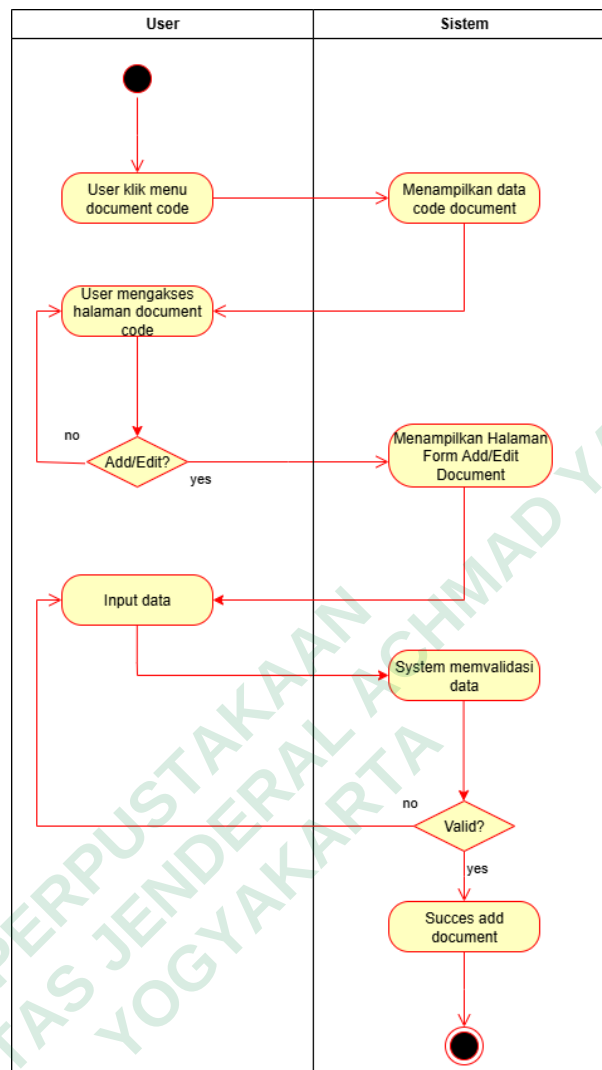
Diagram 3.3 menggambarkan alur saat *user* mengakses halaman *dashboard* setelah *login*. Setelah berhasil *login*, sistem menampilkan data di *dashboard*. Jika *user* mengklik tombol WhatsApp, sistem akan mengarahkan ke halaman WhatsApp. Jika tidak, *user* tetap berada di halaman *dashboard*.



Gambar 3.3 Activity Diagram Dashboard

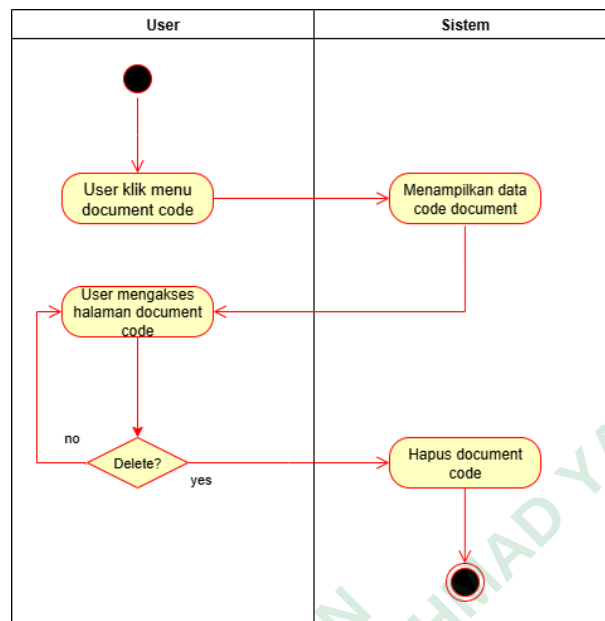
3. Activity Diagram Document Code

Diagram 3.4 menggambarkan proses *user* dalam menambahkan atau mengubah kode dokumen. Setelah *user* memilih menu *Document Code* dan mengakses halaman tersebut, sistem menampilkan data yang ada. Jika *user* memilih untuk menambah atau mengubah, sistem menampilkan form, lalu *user* mengisi data. Sistem kemudian memvalidasi *input*. Jika data valid, proses diselesaikan dengan notifikasi “*Success add document*”, jika tidak, kembali ke form *input* data.



Gambar 3.4 Activity Diagram Document Code

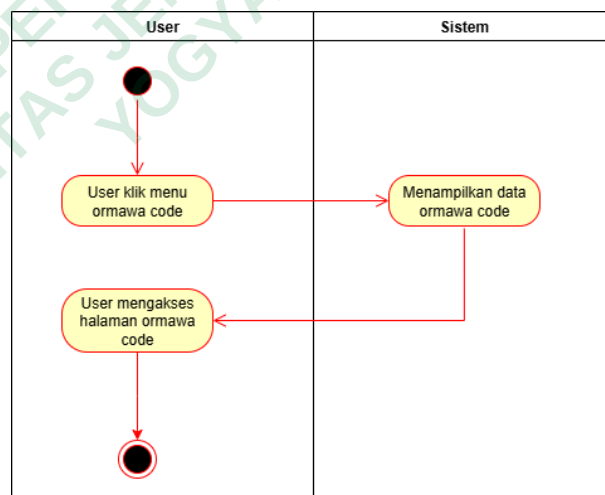
Diagram 3.5 menunjukkan proses penghapusan kode dokumen. Setelah *user* membuka menu dan halaman *document code*, sistem menampilkan data. Jika *user* memilih untuk menghapus, maka sistem akan langsung menjalankan proses penghapusan kode dokumen.



Gambar 3.5 Activity Diagram Delete Document Code

4. Activity Diagram Ormawa Code

Diagram 3.6 menampilkan proses ketika *user* membuka menu Ormawa Code, lalu sistem menampilkan data dan *user* mengakses halaman tersebut.

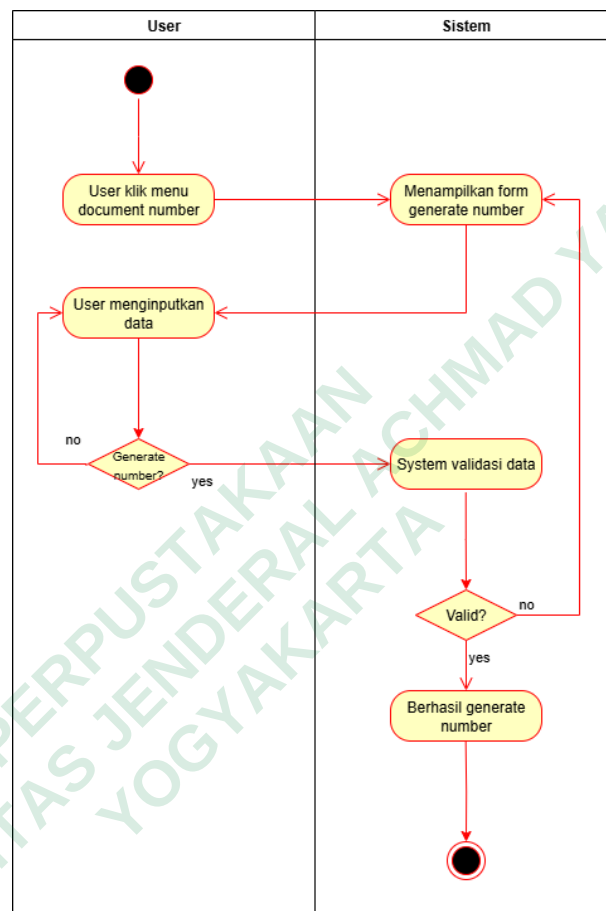


Gambar 3.6 Activity Diagram Ormawa Code

5. Activity Diagram Generate number

Diagram 3.7 menampilkan proses *user* dalam menghasilkan nomor dokumen. *User* mengklik menu *Document Number*, kemudian sistem menampilkan form *generate number*. Setelah *user* mengisi data dan

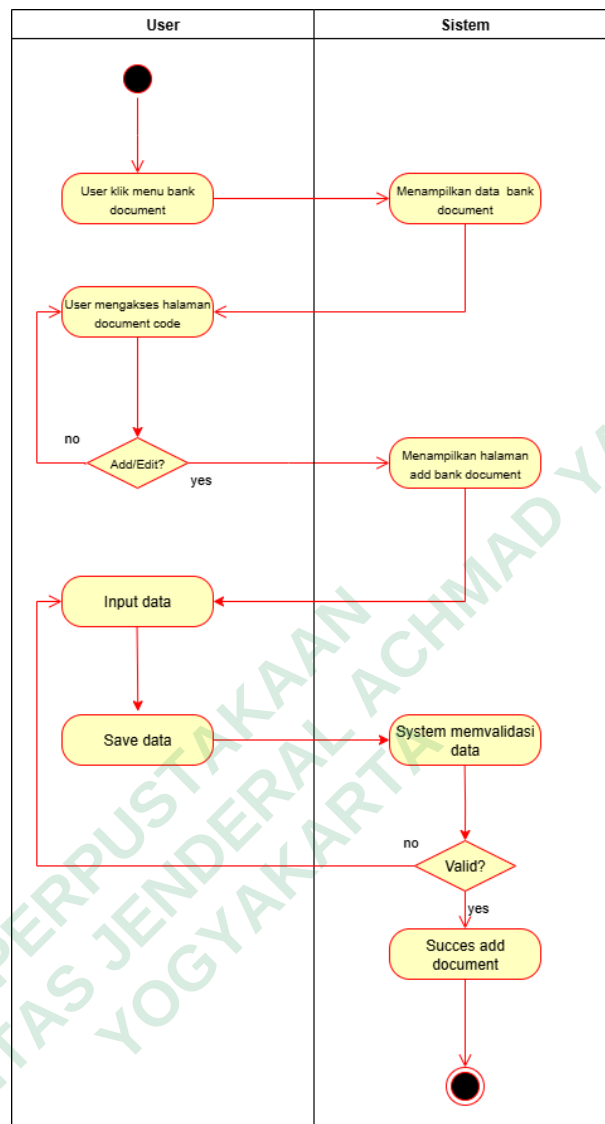
memilih untuk *generate*, sistem memvalidasi data tersebut. Jika valid, sistem akan menampilkan *number* yang berhasil di *generate*, jika tidak valid maka akan dikembalikan ke form *input*.



Gambar 3.7 Activity Diagram Generate Number

6. Activity Diagram Document Archive

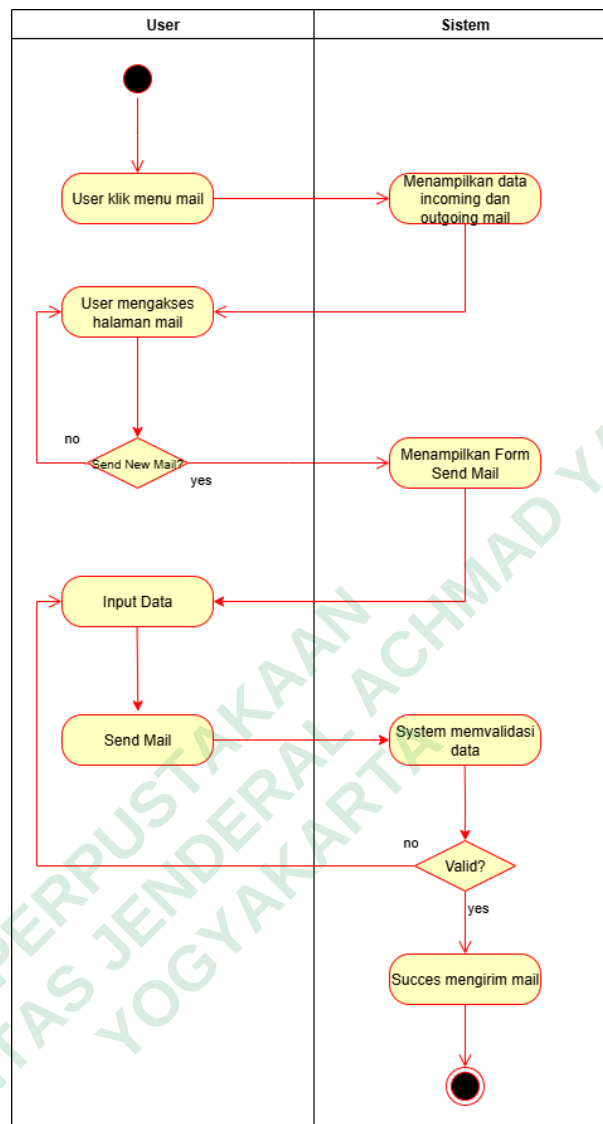
Diagram 3.8 menampilkan proses saat *user* menambah atau mengubah data pada menu *Document Archive*. Setelah mengklik menu tersebut, sistem menampilkan data *document archive*. Jika *user* memilih untuk *add* atau *edit*, maka sistem akan menampilkan form isian. *User* mengisi dan menyimpan data, lalu sistem memvalidasi. Jika valid, sistem menampilkan pesan bahwa dokumen berhasil ditambahkan.



Gambar 3.8 Activity Diagram Document Archive

7. Activity Diagram Mail

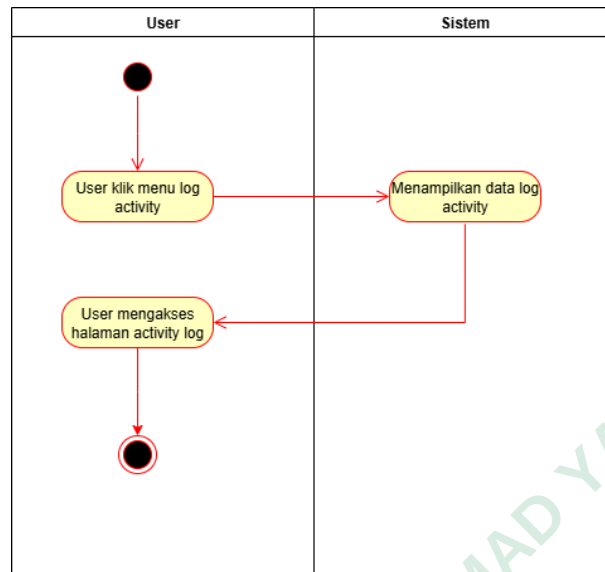
Diagram 3.9 menampilkan proses pengiriman surat melalui sistem. *User* mengklik menu *Mail*, lalu sistem menampilkan data surat masuk dan keluar. Jika *user* memilih untuk mengirim surat baru, sistem menampilkan form pengisian. Setelah data diisi dan dikirim, sistem memvalidasi data. Jika valid, sistem menampilkan pesan bahwa surat berhasil dikirim.



Gambar 3.9 Activity Diagram Mail

8. Activity Diagram Activity log

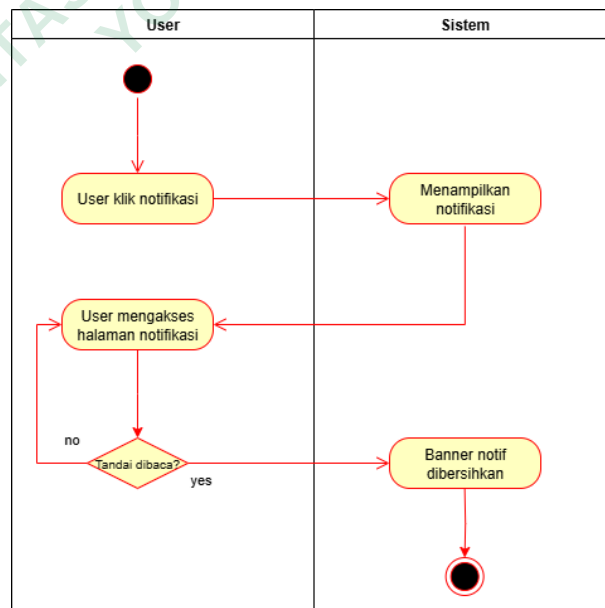
Diagram 3.10 menampilkan alur ketika *user* mengakses data aktivitas. *User* mengklik menu *Activity log*, kemudian sistem menampilkan data riwayat pengguna. Selanjutnya, *user* diarahkan ke halaman *Activity log* untuk melihat detail aktivitas yang telah dicatat oleh sistem.



Gambar 3.10 Activity Diagram Activity log

9. Activity Diagram Notification

Diagram 3.11 menampilkan alur ketika *user* membuka notifikasi. *User* mengklik menu *Notification*, lalu sistem menampilkan daftar notifikasi yang tersedia. Jika *user* menandai notifikasi sebagai telah dibaca, maka sistem akan membersihkan *banner* notifikasi. Jika tidak, *user* tetap berada di halaman notifikasi.



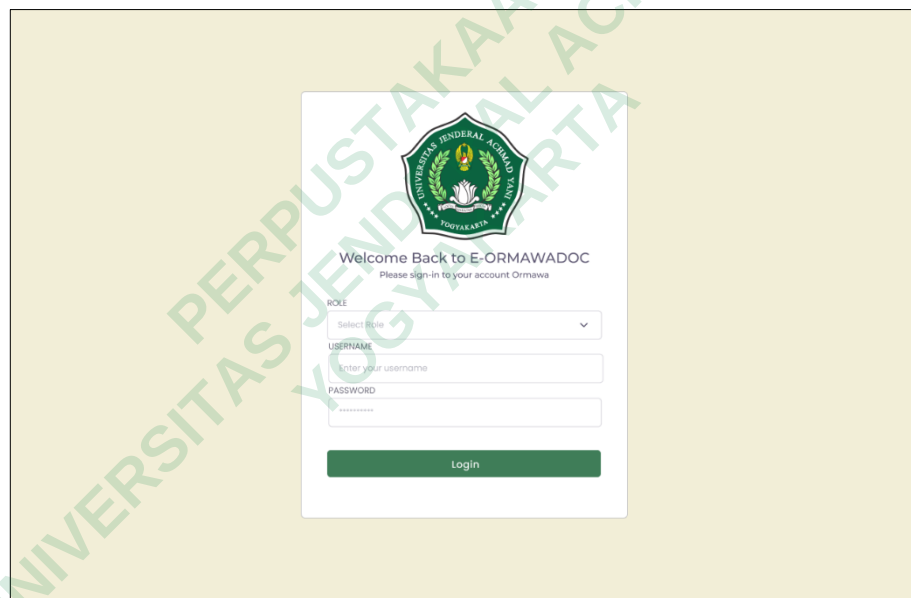
Gambar 3.11 Activity Diagram Notifications

3.4 DESAIN ANTARMUKA

Berikut merupakan desain antarmuka yang penulis gunakan dalam pengembangan sistem, yang telah disesuaikan berdasarkan kebutuhan dan permintaan pengguna untuk memastikan sesuai dengan kebutuhan penggunaan dan *user friendly*.

3.4.1 Desain Halaman *Login*

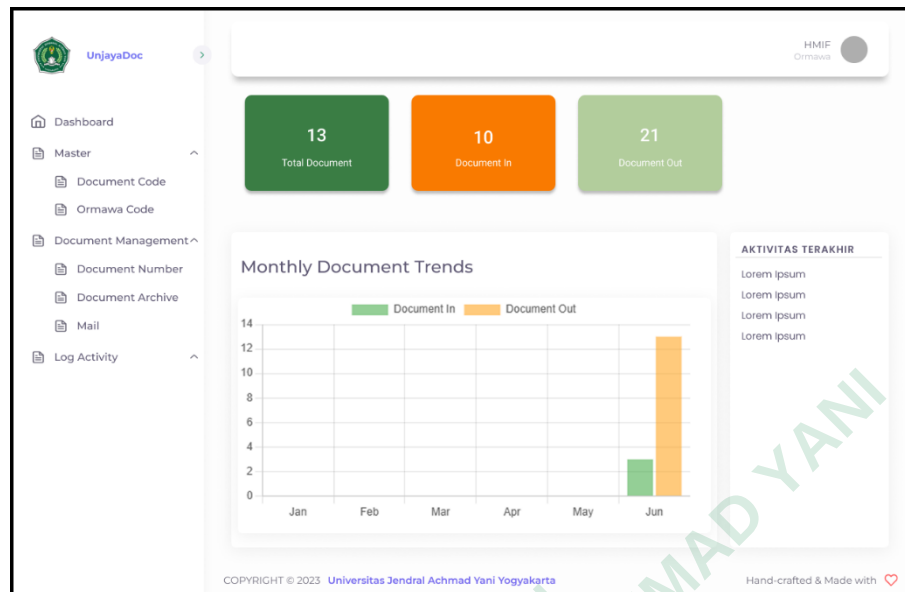
Gambar 3.12 merupakan form *login* untuk akun Ormawa dan Pembimbing. *User* wajib *login* menggunakan akun yang telah diberikan oleh admin. Sebelum *login*, *user* harus memilih peran (Ormawa atau Pembimbing), lalu memasukkan *username* dan *password* dengan benar. Jika informasi yang dimasukkan salah, sistem akan menampilkan pesan kesalahan dan proses *login* akan gagal.



Gambar 3.12 Desain Halaman *Login*

3.4.2 Desain Halaman *Dashboard*

Gambar 3.13 menunjukkan halaman *dashboard* yang menampilkan ringkasan informasi, seperti jumlah dokumen masuk dan keluar, total dokumen, serta aktivitas terakhir yang dilakukan *user*.



Gambar 3.13 Desain Halaman *Dashboard*

3.4.3 Desain Halaman *Document Code*

Gambar 3.14 merupakan desain halaman *Document Code* yang menampilkan daftar kode dokumen yang sedang berlaku. Pada halaman ini tersedia tombol *Add*, *Edit*, dan *Delete* untuk menambahkan, mengubah, atau menghapus kode dokumen sesuai kebutuhan.

UnjayaDoc

HMIF Ormawa

Document Code | Document Code

Add Code Document

Show 10

Search

NO	CODE	DESCRIPTION	STATUS	ACTIONS
1	004	Kode Untuk Surat Tugas	Active	
2	C	Kode Untuk Surat Tugas	Active	
3	007	Kode Untuk Surat Tugas	Active	
4	A.1	Kode Untuk Surat Tugas	Active	
5	B	Kode Untuk Surat Tugas	Active	
6	009	Kode Untuk Surat Tugas	Active	
7	005	Kode Untuk Surat Tugas	Active	
8	003	Kode Untuk Surat Tugas	Active	
9	002	Kode Untuk Surat Tugas	Active	

Showing 1 to 10 of 100 entries

1 2 3 4 5 6 7

COPYRIGHT © 2023 Universitas Jendral Achmad Yani Yogyakarta

Hand-crafted & Made with

Gambar 3.14 Desain Halaman *Document Code*

Gambar 3.15 adalah desain halaman *Add Document Code*, di mana *user* dengan *role* Ormawa dapat *input* data kode dokumen baru. Form yang tersedia harus diisi sesuai dengan format yang telah ditentukan sebelum disimpan ke sistem.

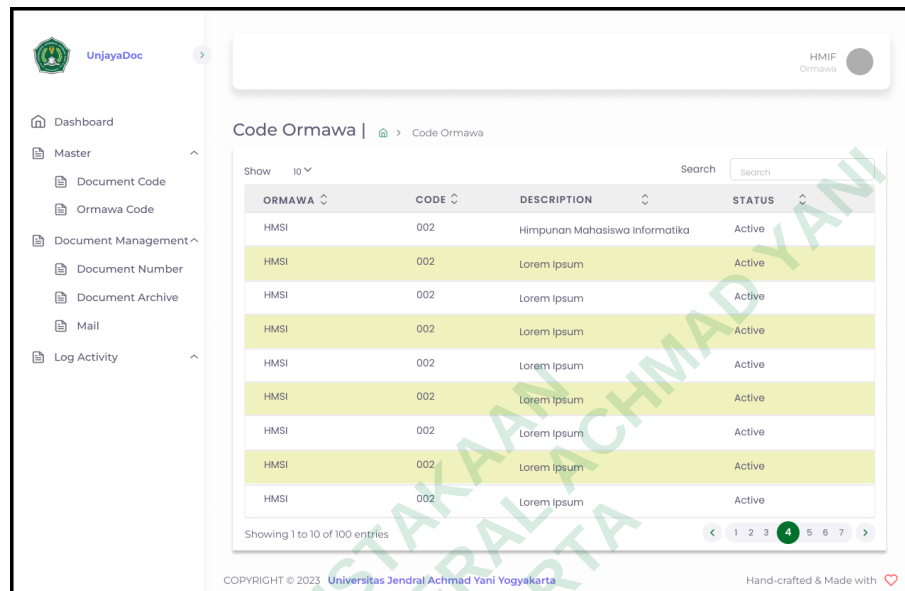
Gambar 3.15 Desain Halaman *Add Document Code*

Gambar 3.16 adalah desain halaman *Edit Document Code*, dimana *user* Ormawa dapat mengubah kode dokumen yang sudah ada jika diperlukan. *User* cukup menginputkan data kode dokumen yang ingin diperbarui melalui form.

Gambar 3.16 Desain Halaman *Edit Document Code*

3.4.4 Desain Halaman Ormawa Code

Gambar 3.17 adalah desain halaman Ormawa Code yang menampilkan informasi berupa kode dan nama Ormawa, baik yang berstatus aktif maupun tidak aktif. Data ditampilkan secara terstruktur untuk memudahkan pengelolaan.



Gambar 3.17 Desain Halaman Ormawa Code

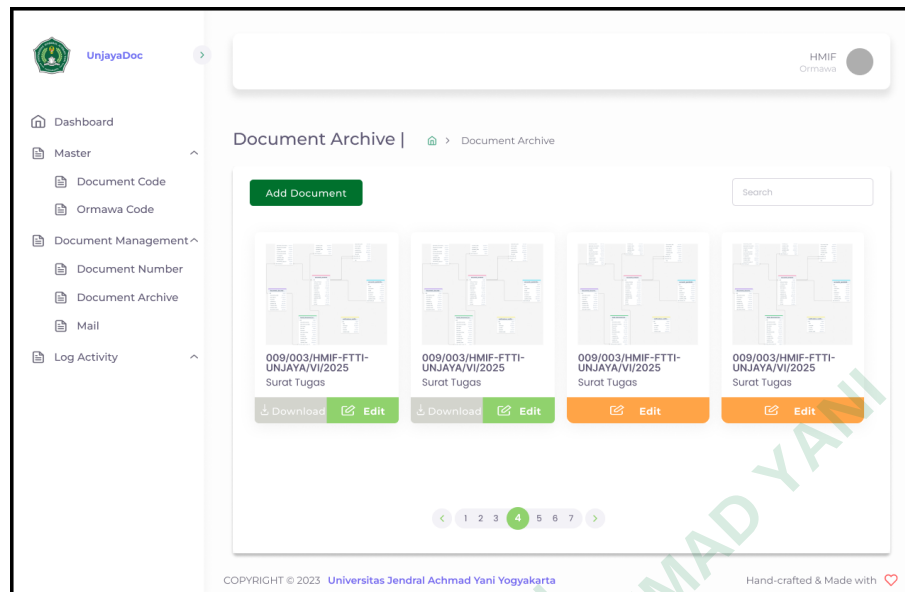
3.4.5 Desain Halaman Document Number

Gambar 3.18 adalah desain halaman Document Numbering yang digunakan untuk membuat nomor dokumen. User input data yang diperlukan, kemudian sistem akan membaca nomor dokumen terakhir dan menghasilkan nomor baru secara otomatis. Nomor dokumen yang berhasil dibuat akan langsung ditampilkan di halaman.

Gambar 3.18 Desain Halaman *Document Number*

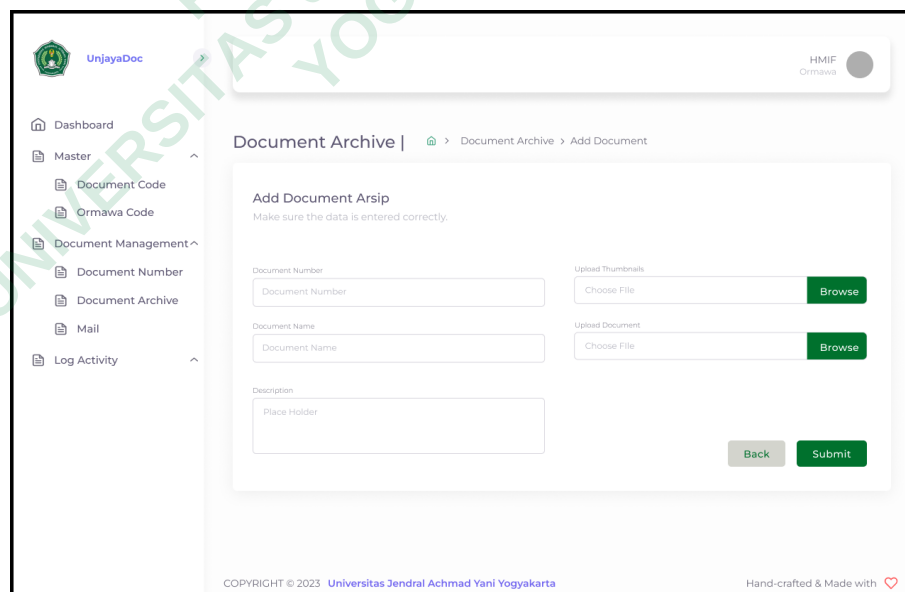
3.4.6 Desain Halaman *Document Archive*

Gambar 3.19 merupakan halaman *Document Archive*, tampilan yang menampilkan semua dokumen yang telah diunggah ke dalam sistem UnjayaDoc. Dokumen-dokumen ini secara otomatis masuk ke halaman arsip setelah nomor dokumen dibuat. Setiap dokumen ditampilkan dalam bentuk kartu (*card*) yang berisi informasi nomor dokumen, nama dokumen, serta status kelengkapan file. Jika dokumen belum memiliki file atau data, maka kartu akan ditampilkan tanpa thumbnail dan tombol *edit* berwarna kuning. Sebaliknya, jika dokumen sudah lengkap dengan file yang diunggah, maka akan muncul thumbnail dari halaman pertama dokumen, tombol *edit* akan berubah menjadi hijau, dan tersedia tombol *download* untuk mengunduh file. Di bagian atas halaman, terdapat tombol *add document* berwarna hijau yang memungkinkan *user* untuk menambahkan dokumen baru tanpa harus melalui proses pembuatan nomor dokumen terlebih dahulu. Selain itu, tersedia juga kolom pencarian di sisi kanan untuk memudahkan pencarian dokumen, serta fitur *pagination* di bagian bawah untuk navigasi antar halaman dokumen.



Gambar 3.19 Desain Halaman *Document Archive*

Gambar 3.20 adalah halaman *Add Document*, *user* dapat menambahkan arsip dokumen baru dengan mengisi form yang terdiri dari *document number*, nama dokumen, deskripsi, serta mengunggah file *thumbnail* dan file pdf. Halaman ini digunakan ketika *user* ingin menambahkan dokumen secara manual tanpa melalui proses *generate* nomor.



Gambar 3.20 Desain Halaman *Add Document Archive*

Gambar 3.21 adalah desain halaman *Edit Document* digunakan untuk memperbarui informasi atau mengganti file dokumen yang sudah ada. Isi form-nya sama dengan halaman *add*, namun informasi dokumen sudah terisi akan ditampilkan secara otomatis. *User* dapat memperbarui nama, deskripsi, serta mengganti file jika diperlukan.

The screenshot shows the 'Edit Document' interface. On the left is a sidebar with a menu: Dashboard, Master (Document Code, Ormawa Code), Document Management (Document Number, Document Archive, Mail, Log Activity). The main area is titled 'Document Archive | Edit Document'. The form 'Edit Document Arsip' contains:

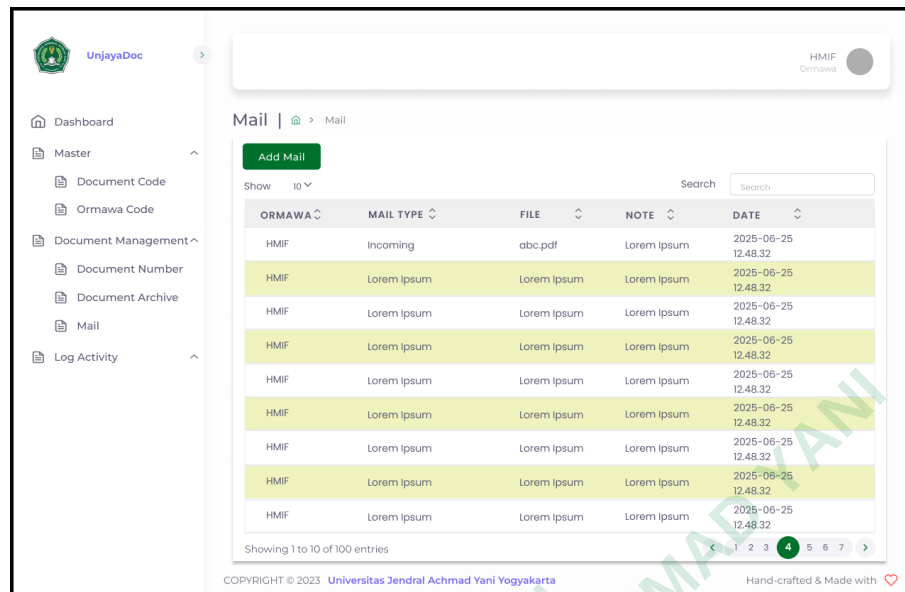
- Document Number: A text input field.
- Document Name: A text input field.
- Description: A text area with a placeholder 'Place Holder'.
- Upload Thumbnails: A 'Choose File' button and a green 'Browse' button.
- Upload Documents: A 'Choose File' button and a green 'Browse' button.
- At the bottom right: 'Back' and 'Submit' buttons.

 The footer of the page reads 'COPYRIGHT © 2023 Universitas Jendral Achmad Yani Yogyakarta' and 'Hand-crafted & Made with ❤️'.

Gambar 3.21 Desain Halaman *Edit Document Archive*

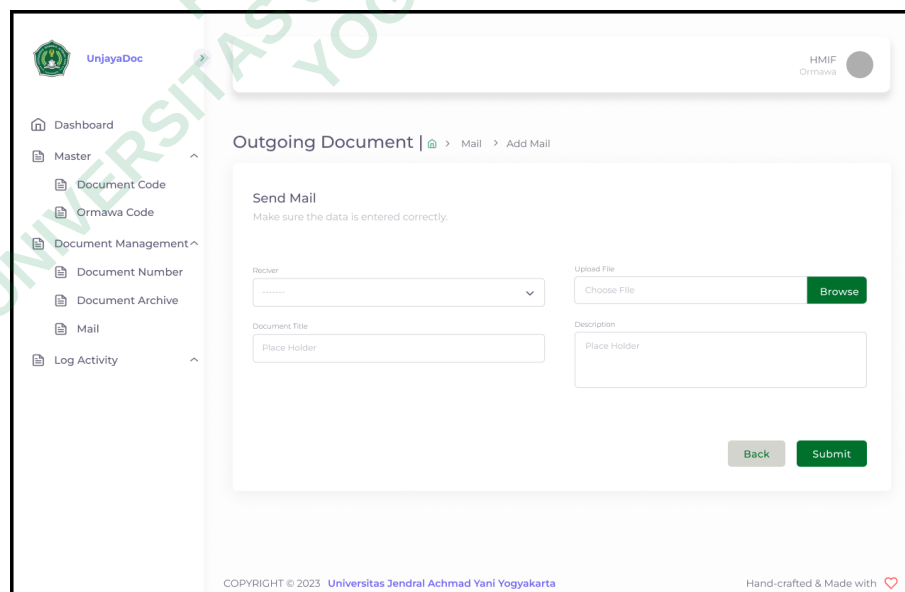
3.4.7 Desain Halaman *Mail*

Halaman *Mail* pada gambar 3.22 menampilkan daftar pesan masuk dan keluar antar Ormawa dalam sistem UnjayaDoc. Setiap baris dalam tabel menunjukkan informasi seperti nama Ormawa, jenis pesan (*mail type*), file yang dilampirkan, catatan, dan tanggal pengiriman. Terdapat tombol *add mail* di bagian atas untuk mengirimkan pesan baru ke Ormawa lain. Fitur pencarian dan *pagination* juga disediakan untuk mempermudah navigasi dan pencarian data pesan.



Gambar 3.22 Desain Halaman *Mail*

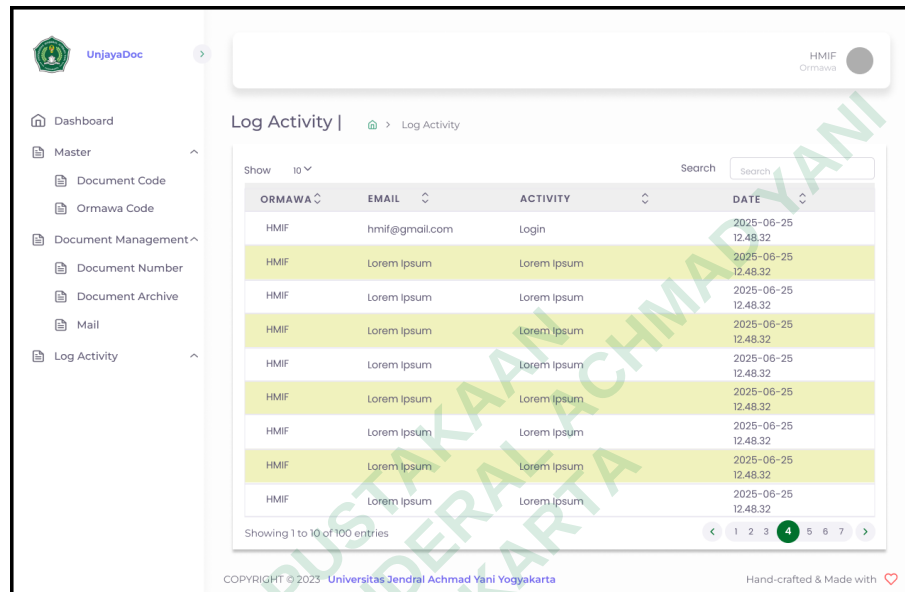
Halaman *Add Mail* pada gambar 3.23 merupakan tampilan untuk mengirim pesan antar Ormawa dalam sistem UnjayaDoc. Pada halaman ini, *user* memilih Ormawa tujuan sebagai penerima, lalu *input* data. Setelah semua data terisi dengan benar, *user* dapat menekan tombol Submit untuk mengirim pesan. Pesan yang terkirim akan secara otomatis ditampilkan sebagai laporan pada halaman *mail*.



Gambar 3.23 Desain Halaman *Add Mail*

3.4.8 Desain Halaman *Activity log*

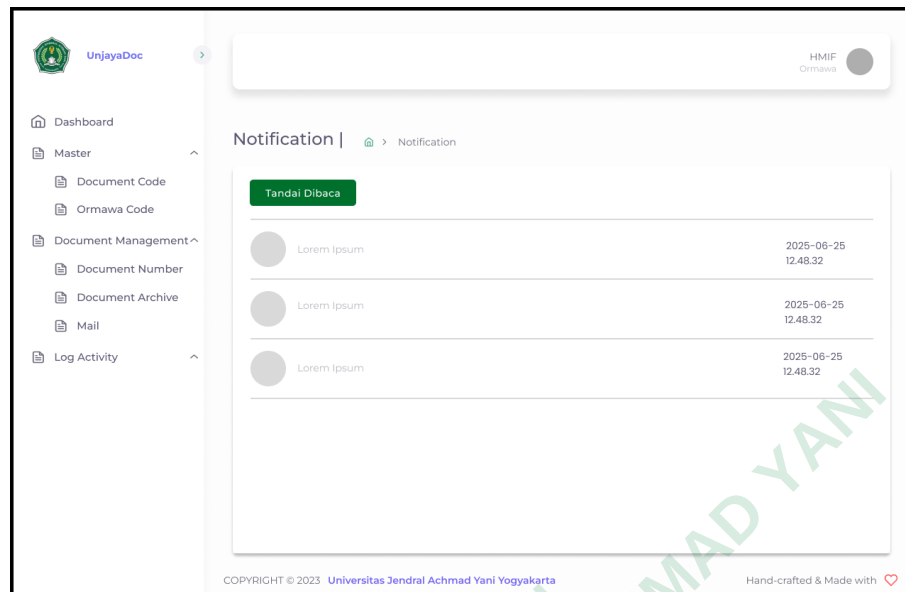
Halaman *Activity log* pada gambar 3.24 menampilkan riwayat aktivitas setiap *user* saat menggunakan sistem UnjayaDoc. Setiap log mencatat tindakan yang dilakukan, waktu kejadian, serta informasi *user*, sehingga memudahkan pemantauan pengguna di sistem.



Gambar 3.24 Desain Halaman *Activity log*

3.4.9 Desain Halaman *Notification*

Halaman *Notification* pada gambar 3.25 menampilkan pemberitahuan kepada *user* Ormawa ketika mereka menerima dokumen dari Ormawa lain. Setiap notifikasi berisi informasi singkat mengenai dokumen yang diterima, sehingga memudahkan *user* untuk mengetahui adanya pesan baru tanpa harus membuka halaman dokumen secara langsung.



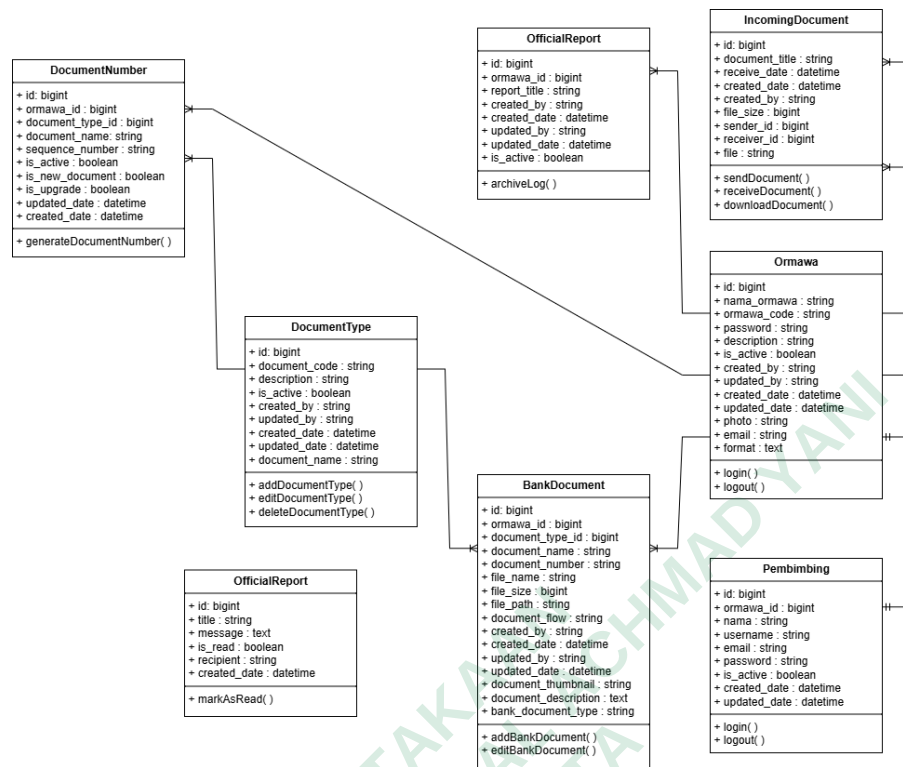
Gambar 3.25 Desain Halaman *Notification*

3.5 DESAIN DATABASE

Berikut merupakan rancangan basis data yang digunakan dalam sistem, yang dirancang untuk mendukung pengelolaan data secara terstruktur, sesuai dengan kebutuhan fungsional sistem dan alur informasi yang dibutuhkan oleh pengguna.

3.5.1 Class Diagram

Gambar 3.26 merupakan *class diagram* yang digunakan dalam sistem ini. Diagram ini menggambarkan struktur kelas, atribut, metode, serta hubungan antar kelas yang merepresentasikan logika dan fungsi utama dari sistem secara menyeluruh.



Gambar 3.26 Class Diagram

3.5.2 Struktur Tabel

Struktur tabel berikut menunjukkan rincian kolom, tipe data, serta atribut penting lainnya yang digunakan dalam setiap tabel di basis data sistem ini. Perancangan struktur ini disesuaikan untuk mendukung kebutuhan fungsional dan integritas data dalam aplikasi:

1. Tabel Ormawa

Tabel 3.2 merupakan tabel Ormawa digunakan untuk menyimpan data terkait organisasi mahasiswa yang terdaftar dalam sistem. Setiap baris dalam tabel ini merepresentasikan satu akun Ormawa lengkap dengan informasi profil, status, dan riwayat pencatatannya. Berikut rincian atributnya:

Tabel 3.2 Tabel Ormawa

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK

2	Nama_ormawa	Varchar(255)	
3	Ormawa_code	Varchar(100)	
4	Password	Varchar(255)	
5	Description	Longtext	
6	Is_active	Tinyit(1)	
7	Created_by	Varchar(100)	
8	Update_by	Varchar(100)	
9	Created_date	Datetime(6)	
10	Update_date	Datetime (6)	
11	Photo	Varchar(255)	
12	Email	Varchar(255)	
13	Format	Varchar(255)	

2. Tabel *Document Type*

Tabel 3.3 merupakan tabel *Document Type* menyimpan data jenis-jenis dokumen dalam sistem. Memiliki *primary key* pada kolom *Id* sebagai identitas unik tiap dokumen. Tabel ini mencatat informasi kode, nama, deskripsi, status aktif, serta data pembuat dan pembaruan.

Tabel 3.3 Tabel *Document Type*

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Document_code	Varchar(255)	
3	Description	Longtext	
4	Is_active	Tinyit(1)	
5	Created_by	Varchar(100)	
6	Update_by	Varchar(100)	
7	Created_date	Datetime	
8	Update_date	Datetime	
9	Document_name	Varchar(255)	

3. Tabel *Document Number*

Tabel 3.4 merupakan tabel *Document Number* digunakan untuk menyimpan informasi terkait penomoran dokumen yang dihasilkan dalam sistem. Tabel ini memiliki *primary key* pada kolom *Id* sebagai identitas unik setiap nomor dokumen. Dua kolom penting dalam tabel ini adalah *foreign key*, yaitu *Document_type_id* dan *Ormawa_code_id*, yang masing-masing merujuk ke tabel *Ormawa*. Relasi ini memungkinkan sistem menghubungkan nomor dokumen dengan jenis dokumen dan organisasi yang mengeluarkannya.

Tabel 3.4 Tabel *Document Number*

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Document_name	Varchar(255)	
3	Generate_document_number	Varchar(255)	
4	Sequence_number	Varchar(255)	
5	Is_active	Tinyit(1)	
6	Is_new_document	Tinyit(1)	
7	Document_version	Varchar(100)	
8	Is_upgrade	Tinyit(1)	
9	Update_date	Datetime(6)	
10	Created_date	Datetime(6)	
11	Document_type_id	Bigint(20)	
12	Ormawa_code_id	Bigint(20)	FK

4. Tabel *Bank Document*

Tabel Bank 3.5 merupakan tabel *Document* digunakan untuk menu *Document Archive* berfungsi untuk menyimpan arsip dokumen yang telah diunggah ke sistem. Setiap dokumen memiliki *primary key* pada kolom *Id* sebagai identitas unik. Tabel ini juga memiliki dua *foreign key*, yaitu *Document_type_id* dan *Ormawa_id*, yang masing-masing

terhubung ke tabel *Document Type* dan *Ormawa*, sehingga memungkinkan pelacakan jenis dan asal dokumen.

Tabel 3.5 Tabel *Bank Document*

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Document_name	Varchar(255)	
3	Document_number	Varchar(255)	
4	File_name	Varchar(255)	
5	File_size	Bigint(20)	
6	File_path	Varchar(500)	
7	Document_flow	Varchar(100)	
8	Is_active	Tinyint(1)	
9	Created_by	Varchar(100)	
10	Created_date	Datetime(6)	
11	Update_by	Varchar(100)	
12	Update_Date	Datetime(6)	
13	Document_type_id	Bigint(20)	FK
14	Ormawa_id	Bigint(20)	FK
15	Document_thumbnail	Varchar(100)	
16	Document_description	Longtext	
17	Bankdocument_type	Varchar(100)	

5. Tabel *Incoming Document*

Tabel 3.6 merupakan tabel *Incoming Document* digunakan untuk menu *Mail* berfungsi untuk menyimpan data dokumen yang masuk ke sistem. Tabel ini memiliki *primary key* pada kolom *Id* sebagai penanda unik setiap dokumen. Terdapat dua *foreign key*, yaitu *Sender_id* dan *Reciver_id*, yang menunjukkan siapa pengirim dan penerima dokumen. Selain itu, tabel ini mencatat judul dokumen, tanggal diterima, deskripsi, ukuran file, nama file, serta informasi siapa yang menginput dan kapan dokumen tersebut dimasukkan ke sistem.

Tabel 3.6 Tabel *Incoming Document*

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Document_title	Varchar(255)	
3	Recived_date	Datetime(6)	
4	Description	Longtext	
5	Created_by	Varchar(100)	
6	Created_date	Datetime(6)	
7	File_size	Bigint(20)	
8	Sender_id	Bigint(20)	FK
9	Reciver_id	Bigint(20)	FK
10	File	Varchar(100)	

6. Tabel *Official Report*

Tabel 3.7 merupakan tabel *Official Report* digunakan untuk halaman *Activity log* berfungsi untuk menyimpan log aktivitas yang dilakukan oleh Ormawa dalam sistem. Tabel ini memiliki *primary key* pada kolom *Id* sebagai identitas unik setiap log. Kolom *Ormawa_id* berperan sebagai *foreign key* yang menghubungkan aktivitas tersebut dengan Ormawa yang terkait.

Tabel 3.7 Tabel *Official Report*

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Report_title	Varchar(255)	
3	Created_by	Varchar(100)	
4	Created_date	Datetime(6)	
5	Update_by	Varchar(100)	
6	Update_Date	Datetime(6)	
7	Is_active	Tinyit(6)	
8	Ormawa_id	Bigint(20)	FK

7. Tabel *Notification*

Tabel 3.8 merupakan tabel *Notification* digunakan untuk menyimpan data notifikasi yang dikirimkan kepada *user* dalam sistem. Tabel ini memiliki *primary key* pada kolom *Id* sebagai identitas unik setiap notifikasi. Informasi yang disimpan mencakup judul, isi pesan, penerima, status apakah notifikasi sudah dibaca atau belum, serta waktu notifikasi dibuat. Tabel ini membantu dalam menyampaikan informasi atau pemberitahuan penting kepada *user* secara terstruktur.

Tabel 3.8 Tabel *Notification*

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Title	Varchar(255)	
3	Message	Longtext	
4	Is_read	Tinyit(6)	
5	Recipient	Varchar(100)	
6	Created_date	Datetime(6)	

8. Tabel Pembimbing Ormawa

Tabel 3.9 merupakan tabel Pembimbing Ormawa digunakan untuk menyimpan data akun para pembimbing yang terhubung dengan Ormawa. Tabel ini memiliki *primary key* pada kolom *Id* sebagai identitas unik setiap pembimbing. Kolom *Ormawa_id* sebagai *foreign key* yang menghubungkan pembimbing dengan Ormawa yang dibimbing.

Tabel 3.9 Tabel Pembimbing Ormawa

No	Atribut	Jenis	Keterangan
1	Id	Bigint(20)	PK
2	Name	Varchar(255)	
3	Username	Varchar(100)	
4	Email	Varchar(254)	

No	Atribut	Jenis	Keterangan
5	Password	Varchar(255)	
6	Is_active	Tinyint(1)	
7	Created_date	Datetime(6)	
8	Updated_date	Datetime(6)	
9	Ormawa_id	Bigint(6)	FK

PERPUSTAKAAN
UNIVERSITAS JENDERAL ACHMAD YANI
YOGYAKARTA