

LAPORAN UJIAN TENGAH SEMESTER
“TUGASKU : SISTEM MANAJEMEN TUGAS”

Laporan ini disusun untuk memenuhi Tugas Ujian Tengah Semester Mata Kuliah

Pemrograman Berbasis Web

Dosen Pengampu : H. Bagja Nugraha, S.T., M.Kom.



Kelas 4C

Disusun oleh Kelompok 4:

Dwi Septian	(2310631250048)
Fachry Firdaus Avicenna	(2310631250014)
Galih Yusuf Ghifari	(2310631250059)
Jaizah Lutfiyah Azzahra	(2310631250021)
Shafa Aulia Nadhira	(2310631250032)

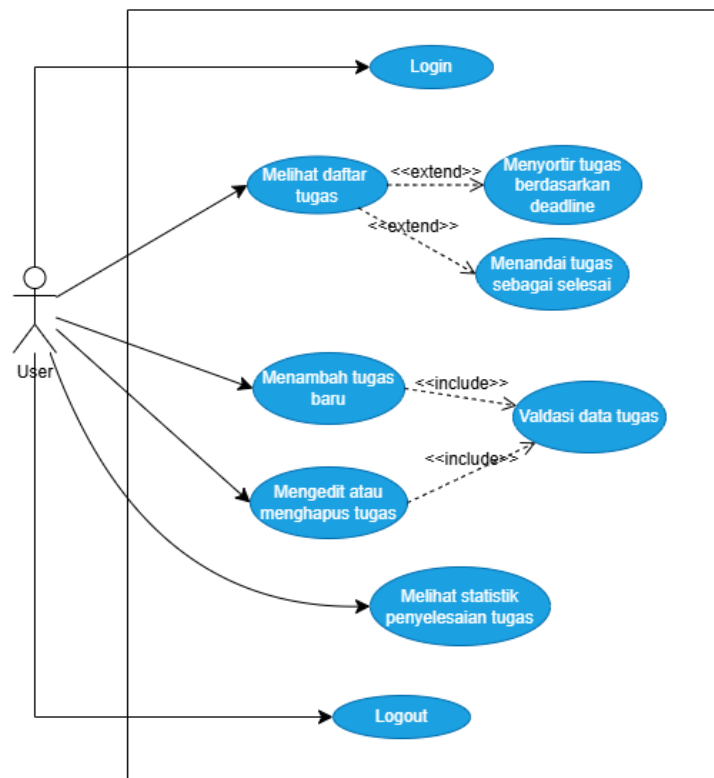
PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SINGAPERBANGSA KARAWANG
2025

DAFTAR ISI

DAFTAR ISI	i
DIAGRAM UML	1
1.1 Use case	1
1.2 Activity Diagram	2
1.3 Sequence Diagram	8
STRUKTUR DATABASE	14
ENTITY RELATIONSHIP DIAGRAM (ERD)	15
RANCANGAN WEBSITE	16

DIAGRAM UML

1.1 Use case



Pada diagram di atas, menunjukkan Use Case Diagram untuk sistem manajemen tugas mahasiswa berbasis web. Diagram ini menggambarkan berbagai interaksi antara pengguna (Mahasiswa) dan sistem dalam mengelola tugas mereka. Berikut adalah penjelasan dari setiap elemen dalam diagram:

a. Aktor

User (Mahasiswa) : Pengguna utama yang berinteraksi dengan sistem untuk mengelola tugas mereka.

b. Use Case

1. Login → Mahasiswa harus masuk ke sistem dengan kredensial mereka.
2. <<include>> Verifikasi kredensial → Sistem akan memverifikasi kredensial sebelum memberikan akses.
3. Melihat daftar tugas → Mahasiswa dapat melihat semua tugas yang telah mereka tambahkan.
4. <<extend>> Menyortir tugas berdasarkan deadline → Mahasiswa dapat menyortir daftar tugas berdasarkan tenggat waktu.
5. <<extend>> Menandai tugas sebagai selesai → Mahasiswa dapat menandai tugas yang sudah selesai.

6. Menambah tugas baru → Mahasiswa dapat memasukkan tugas baru dengan detail seperti nama, deadline, dan deskripsi.
7. <<include>> Validasi data tugas → Sistem akan memvalidasi data sebelum menyimpan tugas.
8. Mengedit atau menghapus tugas → Mahasiswa dapat mengubah atau menghapus tugas yang sudah dimasukkan.
9. <<include>> Validasi data tugas → Data tugas yang diedit akan divalidasi sebelum diperbarui.
10. Melihat statistik penyelesaian tugas** → Mahasiswa dapat melihat ringkasan tugas yang telah mereka selesaikan.
11. Logout → Mahasiswa dapat keluar dari sistem setelah selesai menggunakan aplikasi.

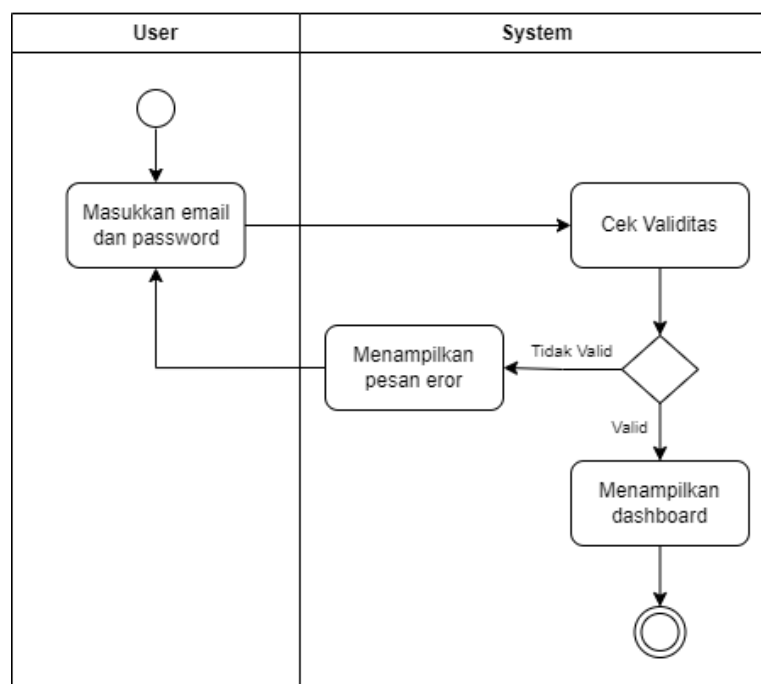
c. Hubungan Use Case

1. <<include>> → Menunjukkan bahwa suatu use case selalu melibatkan use case lain. Misalnya, setiap kali pengguna login, sistem harus memverifikasi kredensial.
2. <<extend>> → Menunjukkan bahwa suatu use case dapat diperluas jika diperlukan, seperti opsi untuk menyortir tugas atau menandai tugas sebagai selesai setelah melihat daftar tugas.

Secara keseluruhan, diagram ini menunjukkan bagaimana mahasiswa dapat mengelola tugas mereka dalam sistem, mulai dari login, menambah, mengedit, menyortir, hingga menyelesaikan tugas.

1.2 Activity Diagram

1. Activity Diagram Login



Pada diagram di atas, menggambarkan alur proses login ke dalam sistem manajemen tugas mahasiswa berbasis web dengan keterlibatan dua aktor utama, yaitu User (Mahasiswa) dan System. Diagram ini menggunakan format Activity Diagram untuk memodelkan alur aktivitas yang terjadi dalam proses login ke sistem. Prosesnya sebagai berikut:

1) Masukkan email dan password

Aktivitas dimulai saat pengguna memasukkan email dan password pada form login yang tersedia di halaman sistem.

2) Cek validitas

Sistem akan melakukan pengecekan terhadap kredensial yang dimasukkan, mencocokkannya dengan data yang tersimpan di database.

3) Keputusan validitas

- Sistem akan menentukan apakah email dan password yang dimasukkan valid atau tidak.
- Jika valid, proses berlanjut ke langkah berikutnya.
- Jika tidak valid, sistem menampilkan pesan error dan meminta pengguna untuk memasukkan ulang kredensial.

4) Menampilkan pesan error (Jika tidak valid)

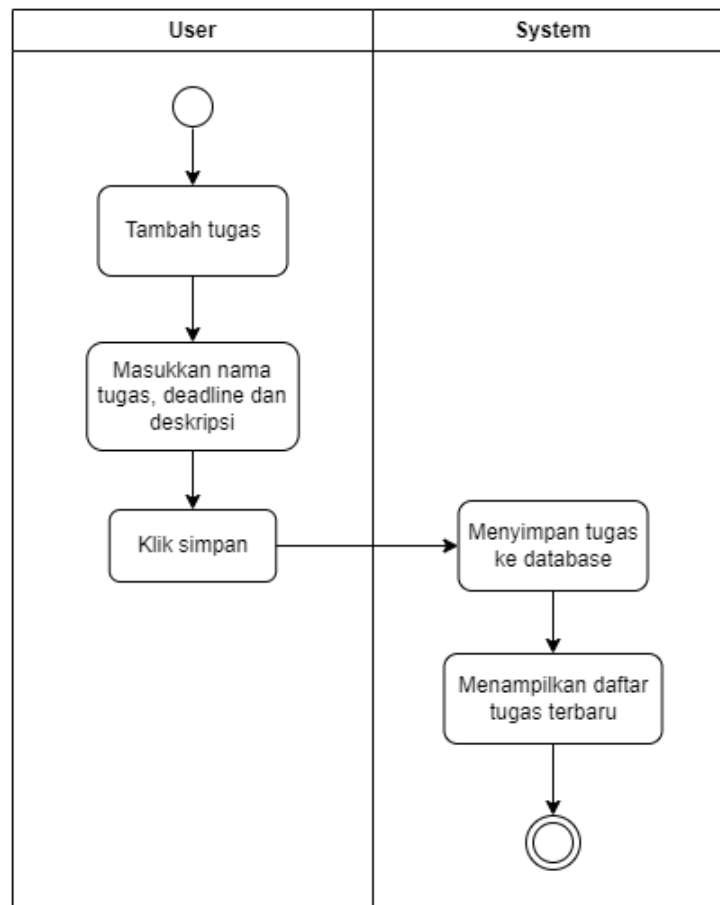
- Jika kredensial tidak valid, sistem akan menampilkan pesan error kepada pengguna, menginformasikan bahwa login gagal.
- Pengguna dapat mencoba memasukkan kembali email dan password yang benar.

5) Menampilkan dashboard (Jika valid)

- Jika kredensial valid, sistem akan mengizinkan pengguna masuk dan menampilkan dashboard utama dari sistem manajemen tugas.
- Pengguna dapat mulai menggunakan fitur-fitur yang tersedia dalam sistem.

Dengan demikian, diagram ini menunjukkan bagaimana pengguna berinteraksi dengan sistem untuk melakukan login, serta bagaimana sistem menangani validasi kredensial sebelum memberikan akses ke dalam sistem.

2. Activity Diagram Menambah Tugas



Pada diagram di atas, menggambarkan alur proses penambahan tugas dalam sistem manajemen tugas mahasiswa berbasis web dengan keterlibatan dua aktor utama, yaitu User (Mahasiswa) dan System. Diagram ini menggunakan format Activity Diagram untuk memodelkan alur aktivitas yang terjadi dalam proses penambahan tugas. Prosesnya sebagai berikut:

1) Tambah tugas

Aktivitas dimulai saat pengguna memilih opsi untuk menambahkan tugas baru dalam sistem.

2) Masukkan nama tugas, deadline, dan deskripsi

Pengguna mengisi informasi tugas yang terdiri dari nama tugas, tanggal deadline, dan deskripsi tugas sebagai detail tambahan.

3) Klik simpan

Setelah semua informasi tugas diisi, pengguna menekan tombol "Simpan" untuk menyimpan tugas ke dalam sistem.

4) Menyimpan tugas ke database

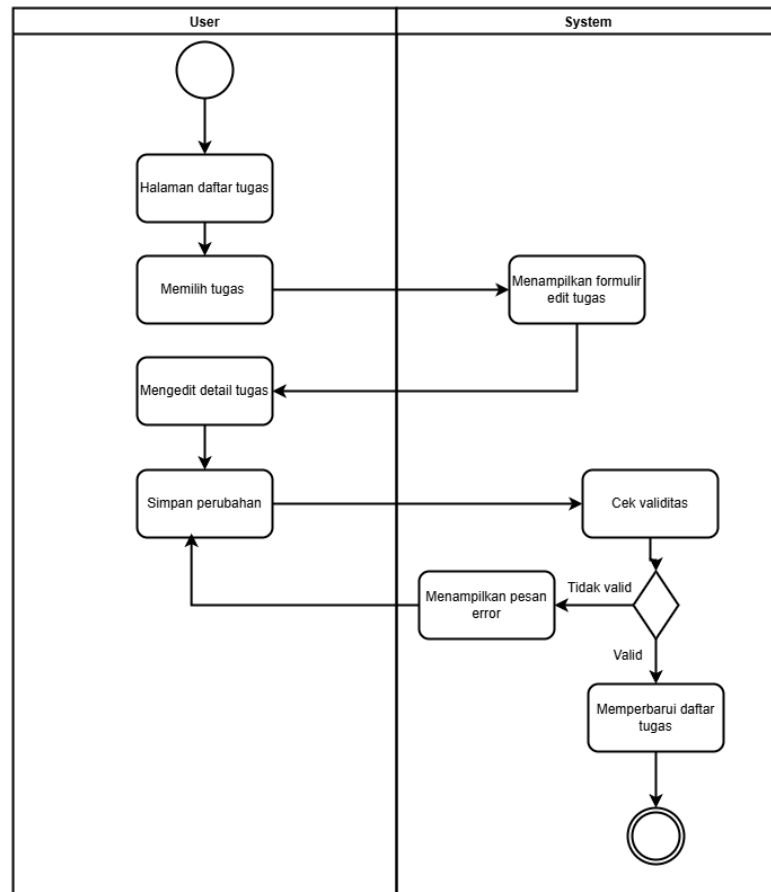
Sistem menerima data tugas yang dimasukkan oleh pengguna dan menyimpannya ke dalam database.

5) Menampilkan daftar tugas terbaru

Setelah tugas berhasil disimpan, sistem akan memperbarui dan menampilkan daftar tugas terbaru kepada pengguna.

Diagram ini menunjukkan bagaimana pengguna dapat menambahkan tugas ke dalam sistem dengan langkah-langkah yang sederhana, serta bagaimana sistem menangani penyimpanan data dan memperbarui daftar tugas secara otomatis.

3. Activity Diagram Mengedit Tugas



Pada diagram di atas, menggambarkan alur proses pengeditan tugas dalam sistem manajemen tugas berbasis web dengan keterlibatan dua aktor utama, yaitu User (Pengguna) dan System (Sistem). Diagram ini menggunakan format Activity Diagram untuk memodelkan alur aktivitas yang terjadi dalam proses pengeditan tugas. Prosesnya sebagai berikut :

1) Halaman daftar tugas

Aktivitas dimulai saat pengguna berada di halaman daftar tugas dalam sistem.

2) Memilih tugas

Pengguna memilih tugas yang ingin diedit dari daftar tugas yang tersedia.

3) Menampilkan formulir edit tugas

Sistem menerima permintaan pengguna dan menampilkan formulir edit tugas yang memungkinkan pengguna untuk mengubah detail tugas.

4) Mengedit detail tugas

Pengguna melakukan perubahan pada detail tugas, seperti mengubah nama, deskripsi, atau deadline tugas.

5) Simpan perubahan

Setelah selesai mengedit, pengguna menekan tombol "Simpan" untuk menyimpan perubahan yang telah dilakukan.

6) Cek validitas

Sistem melakukan validasi terhadap data yang telah dimasukkan oleh pengguna untuk memastikan bahwa data yang diedit sesuai dengan aturan yang berlaku.

7) Menampilkan pesan error (jika tidak valid)

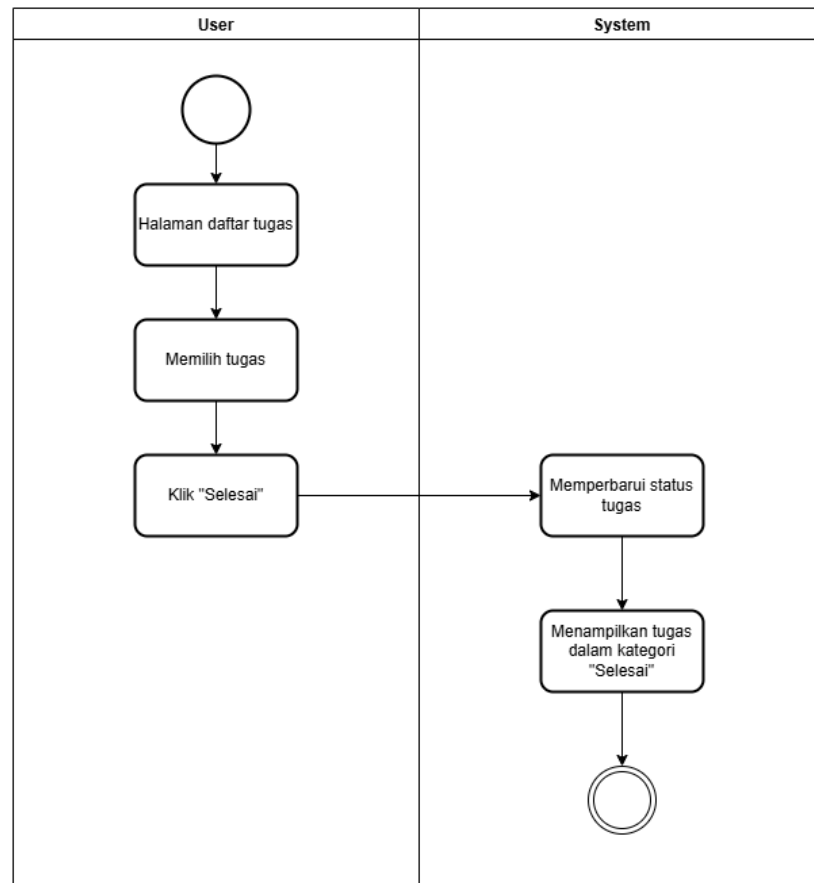
Jika data yang dimasukkan tidak valid, sistem akan menampilkan pesan error kepada pengguna, dan pengguna harus memperbaiki kesalahan sebelum melanjutkan.

8) Memperbarui daftar tugas (jika valid)

Jika data yang dimasukkan valid, sistem akan memperbarui daftar tugas dengan informasi terbaru yang telah diedit oleh pengguna.

Diagram ini menunjukkan bagaimana pengguna dapat mengedit tugas yang sudah ada dalam sistem dengan langkah-langkah yang sistematis, serta bagaimana sistem menangani validasi data dan memperbarui daftar tugas setelah perubahan berhasil disimpan.

4. Activity Diagram Menandai Tugas Selesai



Pada diagram di atas, menggambarkan alur proses menandai tugas selesai dalam sistem manajemen tugas berbasis web dengan keterlibatan dua aktor utama, yaitu User (Pengguna) dan System (Sistem). Diagram ini menggunakan format Activity Diagram untuk memodelkan alur aktivitas yang terjadi dalam proses menyelesaikan tugas. Prosesnya sebagai berikut :

1) Halaman daftar tugas

Aktivitas dimulai saat pengguna berada di halaman daftar tugas dalam sistem.

2) Memilih tugas

Pengguna memilih tugas yang ingin ditandai sebagai selesai dari daftar tugas yang tersedia.

3) Klik "Selesai"

Pengguna menekan tombol "Selesai" untuk menandai bahwa tugas tersebut telah diselesaikan.

4) Memperbarui status tugas

Sistem menerima permintaan pengguna dan memperbarui status tugas menjadi "Selesai" dalam database.

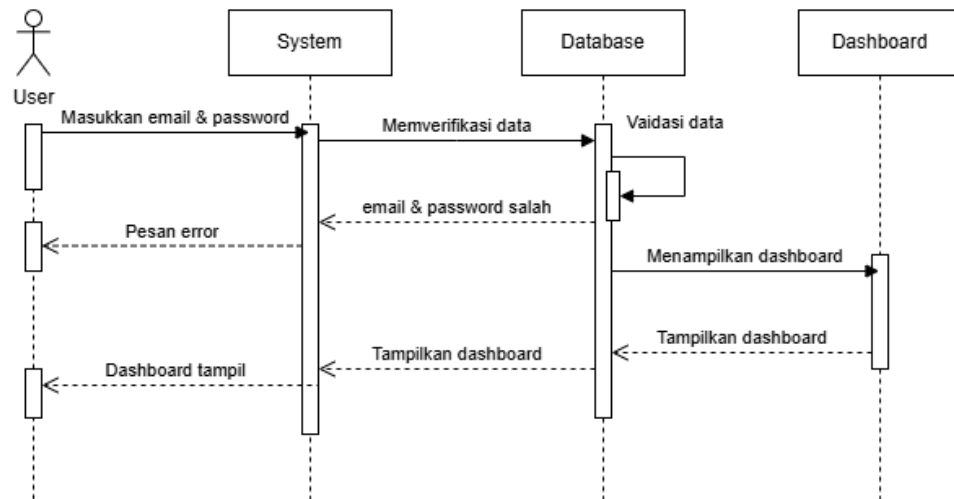
5) Menampilkan tugas dalam kategori "Selesai"

Setelah status diperbarui, sistem akan memindahkan tugas tersebut ke dalam kategori tugas yang telah selesai dan menampilkannya kepada pengguna.

Diagram ini menunjukkan bagaimana pengguna dapat menandai tugas sebagai selesai dalam sistem dengan langkah-langkah yang sederhana, serta bagaimana sistem menangani pembaruan status tugas secara otomatis.

1.3 Sequence Diagram

1. Sequence Diagram Login



Pada diagram di atas, menggambarkan alur proses login dalam sistem berbasis web dengan keterlibatan beberapa komponen utama, yaitu User (Pengguna), System (Sistem), Database dan Dashboard. Diagram ini menggunakan format Sequence Diagram untuk memodelkan interaksi antar komponen dalam proses login pengguna. Prosesnya sebagai berikut:

1) Masukkan email & password

Pengguna memasukkan email dan password pada form login di sistem.

2) Memverifikasi data

Sistem mengirimkan data login ke database untuk diverifikasi.

3) Validasi data

Database melakukan validasi terhadap email dan password yang dikirimkan.

4) Jika email & password salah

- Jika kombinasi email dan password tidak sesuai, database mengirimkan respons ke sistem bahwa data login salah.
- Sistem kemudian menampilkan ****pesan error**** kepada pengguna untuk mencoba kembali.

5) Jika email & password benar

- Jika data login benar, database mengonfirmasi bahwa pengguna valid.

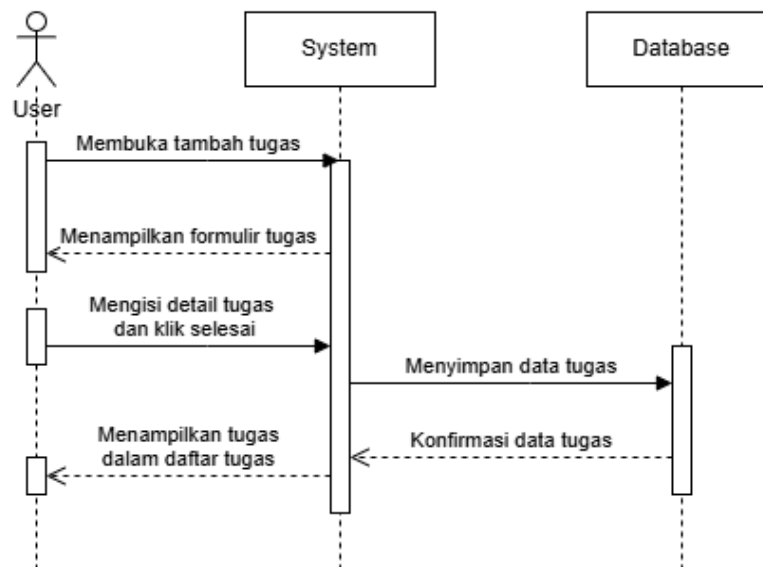
- Database mengirimkan perintah ke Dashboard untuk ditampilkan.

6) Menampilkan dashboard

- Sistem menerima respons dari database dan mengarahkan pengguna ke halaman Dashboard.
- Dashboard kemudian ditampilkan kepada pengguna.

Diagram ini menunjukkan bagaimana sistem menangani proses login dengan validasi data dari database, memberikan pesan kesalahan jika login gagal, dan menampilkan dashboard jika login berhasil.

2. Sequence Diagram Menambah Tugas



Pada diagram di atas, menggambarkan alur proses penambahan tugas dalam sistem manajemen tugas berbasis web dengan keterlibatan tiga komponen utama, yaitu User (Pengguna), System (Sistem), dan Database. Diagram ini menggunakan format Sequence Diagram untuk memodelkan interaksi antar komponen dalam proses penambahan tugas. Prosesnya sebagai berikut :

1) Membuka tambah tugas

Pengguna memilih opsi untuk menambahkan tugas baru dalam sistem.

2) Menampilkan formulir tugas

Sistem menampilkan formulir untuk memasukkan detail tugas yang akan ditambahkan.

3) Mengisi detail tugas dan klik selesai

- Pengguna mengisi informasi tugas, seperti nama tugas, deskripsi, dan deadline.
- Setelah selesai, pengguna menekan tombol "Selesai" untuk menyimpan tugas.

4) Menyimpan data tugas

Sistem menerima data tugas yang dimasukkan oleh pengguna dan mengirimkannya ke database untuk disimpan.

5) Konfirmasi data tugas

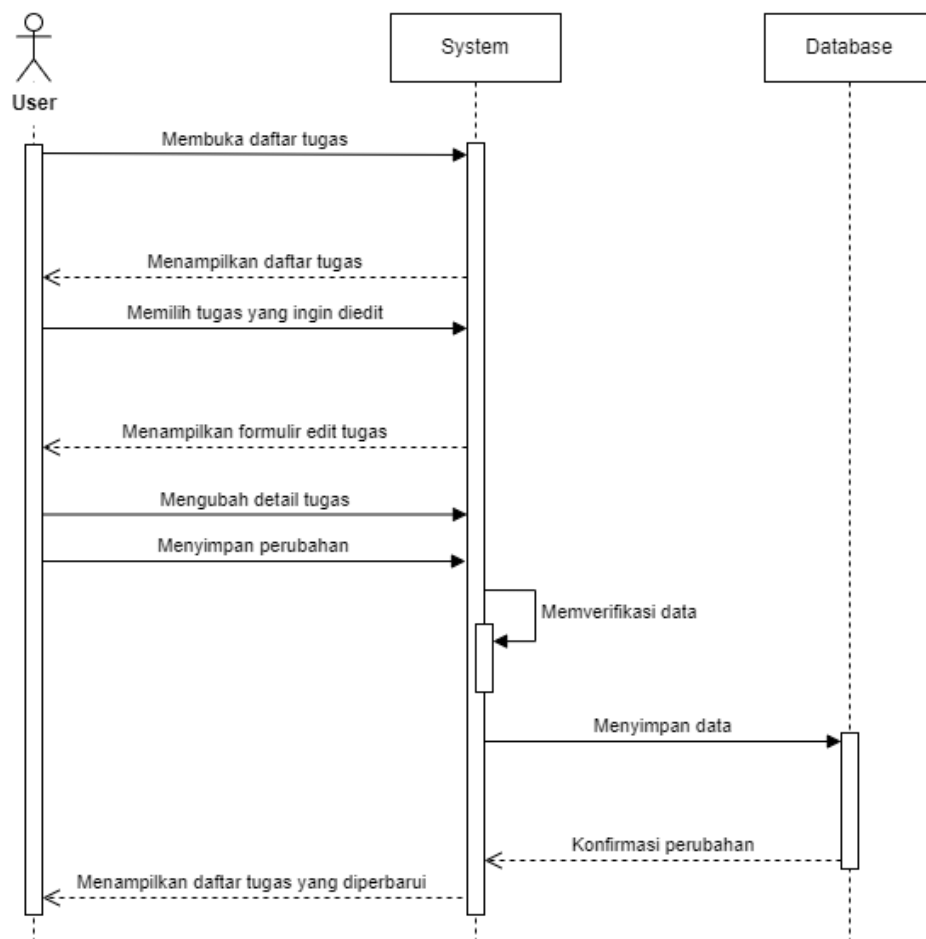
Database memproses penyimpanan tugas dan mengonfirmasi bahwa data tugas telah berhasil disimpan.

6) Menampilkan tugas dalam daftar tugas

Sistem menerima konfirmasi dari database dan memperbarui daftar tugas untuk menampilkan tugas yang baru ditambahkan kepada pengguna.

Diagram ini menunjukkan bagaimana pengguna dapat menambahkan tugas ke dalam sistem dengan langkah-langkah yang sistematis, serta bagaimana sistem menangani penyimpanan data dan memperbarui daftar tugas secara otomatis.

3. Sequence Diagram Mengedit Tugas



Pada diagram di atas, menggambarkan alur proses pengeditan tugas dalam sistem manajemen tugas berbasis web dengan keterlibatan dua aktor utama, yaitu User (Mahasiswa) dan System. Diagram ini menggunakan format Sequence Diagram untuk memodelkan alur interaksi antara pengguna, sistem, dan database dalam proses pengeditan tugas. Prosesnya sebagai berikut:

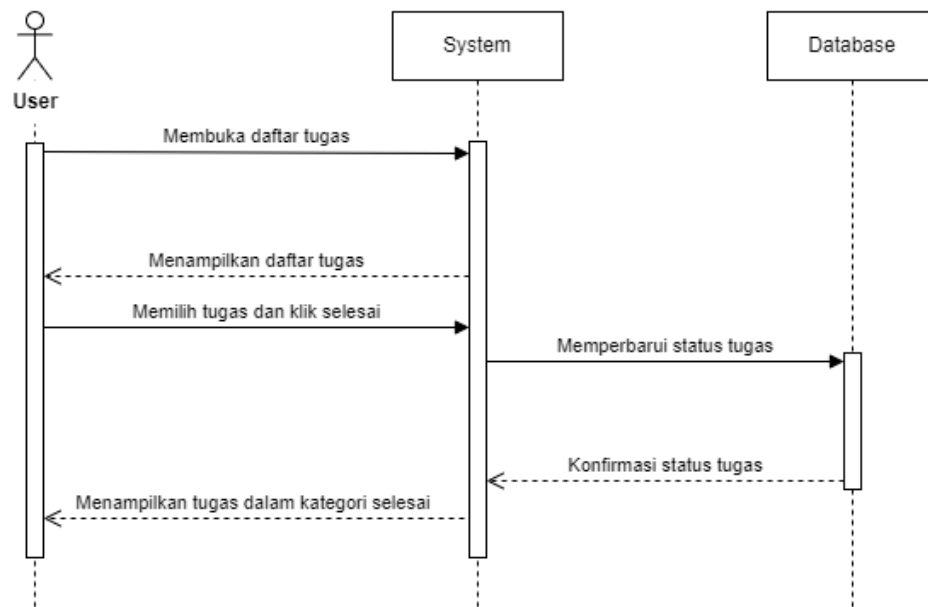
1) Membuka daftar tugas

Aktivitas dimulai saat pengguna mengakses sistem dan membuka daftar tugas yang tersedia.

- 2) Menampilkan daftar tugas
Sistem akan mengambil daftar tugas dari database dan menampilkannya kepada pengguna.
- 3) Memilih tugas yang ingin diedit
Pengguna memilih salah satu tugas dari daftar yang ingin diperbarui.
- 4) Menampilkan formulir edit tugas
Sistem menampilkan formulir yang berisi detail tugas yang dipilih agar pengguna dapat mengedit informasi yang diperlukan.
- 5) Mengubah detail tugas
Pengguna melakukan perubahan pada informasi tugas, seperti nama tugas, deadline, atau deskripsi.
- 6) Menyimpan perubahan
Setelah selesai mengedit, pengguna menekan tombol "Simpan" untuk memperbarui tugas dalam sistem.
- 7) Memverifikasi data
Sistem akan melakukan validasi terhadap data yang telah diubah sebelum menyimpan ke dalam database.
- 8) Menyimpan data ke dalam database
Jika data valid, sistem akan mengirimkan perubahan ke database untuk diperbarui.
- 9) Konfirmasi perubahan
Setelah berhasil menyimpan perubahan, sistem mengirimkan konfirmasi kepada pengguna.
- 10) Menampilkan daftar tugas yang diperbarui
Sistem akan memperbarui tampilan daftar tugas sehingga pengguna dapat melihat hasil perubahan yang telah dilakukan.

Diagram ini menunjukkan bagaimana pengguna dapat melakukan perubahan pada tugas yang telah ada, bagaimana sistem menangani validasi dan penyimpanan data, serta bagaimana daftar tugas diperbarui secara otomatis setelah perubahan berhasil dilakukan.

4. Sequence Diagram Menandai Tugas Selesai



Pada diagram di atas, menggambarkan alur proses menandai tugas selesai dalam sistem manajemen tugas berbasis web dengan keterlibatan dua aktor utama, yaitu User (Mahasiswa) dan System. Diagram ini menggunakan format Sequence Diagram untuk memodelkan interaksi antara pengguna, sistem, dan database dalam proses penyelesaian tugas. Prosesnya sebagai berikut :

1) Membuka daftar tugas

Aktivitas dimulai saat pengguna mengakses sistem dan membuka daftar tugas yang tersedia.

2) Menampilkan daftar tugas

Sistem mengambil daftar tugas dari database dan menampilkannya kepada pengguna.

3) Memilih tugas dan klik selesai

Pengguna memilih salah satu tugas dari daftar yang ingin ditandai sebagai selesai dengan menekan tombol "Selesai".

4) Memperbarui status tugas

Sistem menerima perintah dari pengguna dan memperbarui status tugas dalam database menjadi "Selesai".

5) Konfirmasi status tugas

Setelah status tugas berhasil diperbarui dalam database, sistem mengonfirmasi bahwa perubahan telah dilakukan.

6) Menampilkan tugas dalam kategori selesai

Sistem memperbarui tampilan daftar tugas dengan memindahkan tugas yang telah selesai ke dalam kategori "Selesai".

Diagram ini menunjukkan bagaimana pengguna dapat menandai tugas sebagai selesai, bagaimana sistem menangani pembaruan status dalam database, serta bagaimana daftar tugas diperbarui agar pengguna dapat melihat perubahan secara langsung.

STRUKTUR DATABASE

1. Tabel User

No	Kolom	Tipe Data	Primary Key	Null	Auto Increment
1	id_user	int (11)	Yes	Not Null	Yes
2	nama	varchar (100)	No	Not Null	No
3	email	varchar (50)	No	Not Null	No
4	username	varchar (20)	No	Not Null	No
5	password	varchar (255)	No	Not Null	No
6	dibuat_pada	datetime	No	Not Null	No
7	diperbarui_pada	datetime	No	Not Null	No

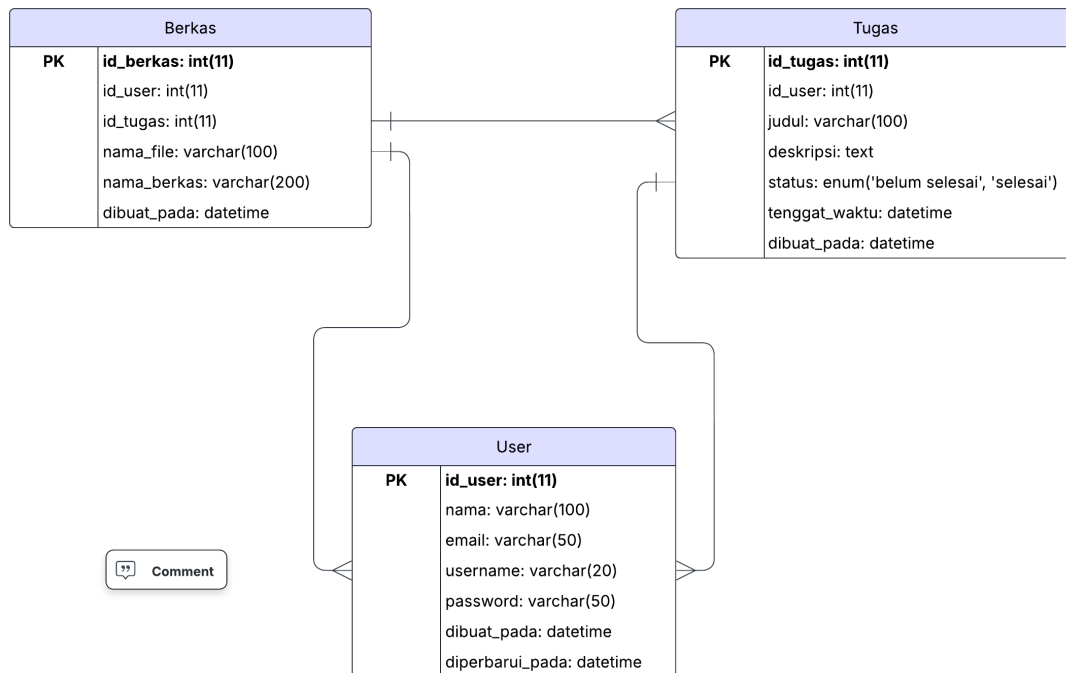
2. Tabel Tugas (Menyimpan data Tugas)

No	Kolom	Tipe Data	Primary Key	Null	Auto Increment
1	id_tugas	int (11)	Yes	Not Null	Yes
2	id_user	int (11)	No	Not Null	No
3	judul	varchar (100)	No	Not Null	No
4	deskripsi	text	No	Yes	No
5	status	ENUM ('belum selesai', 'selesai')	No	Not Null	No
6	tenggat_waktu	datetime	No	Not Null	No
7	dibuat_pada	datetime	No	Not Null	No

3. Tabel Berkas (menyimpan file yang diunggah oleh user)

No	Kolom	Tipe Data	Primary Key	Null	Auto Increment
1	id_berkas	int(11)	Yes	Not Null	Yes
2	id_user	int(11)	No	Not Null	No
3	id_tugas	varchar(50)	No	Not Null	No
4	nama_file	varchar(100)	No	Not Null	No
5	nama_berkas	varchar (200)	No	Not Null	No
6	dibuat_pada	datetime	No	Not Null	No

ENTITY RELATIONSHIP DIAGRAM (ERD)



1. Tabel User

- ID unik (id_user)
- Nama, email, username, password
- Waktu dibuat dan diperbarui

2. Tabel Tugas

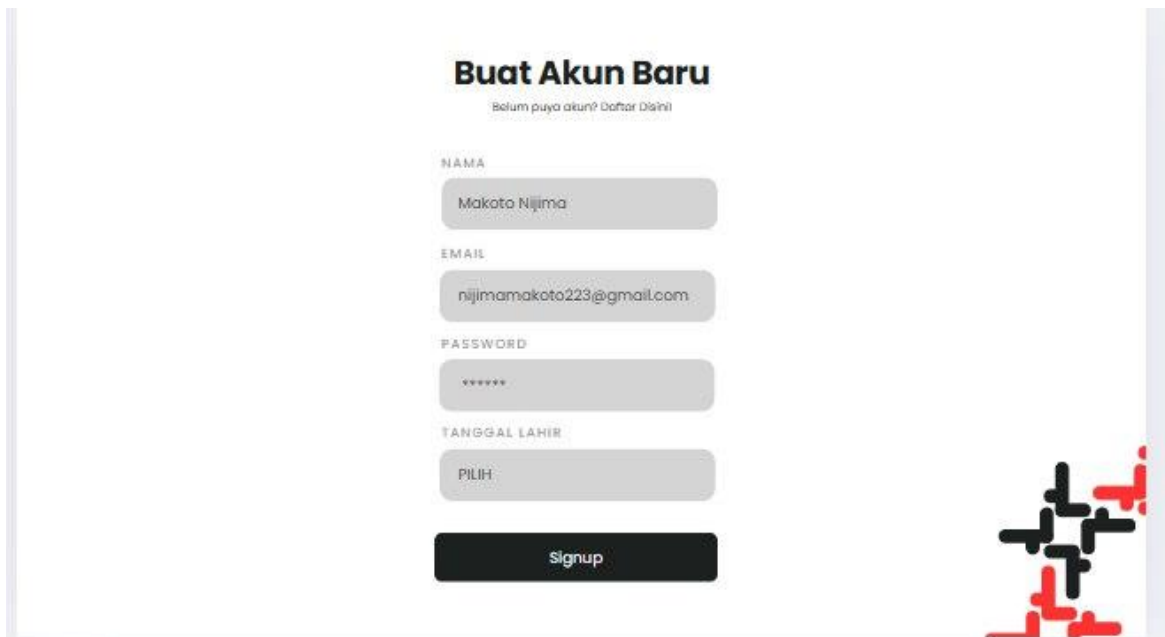
- Setiap tugas dibuat oleh satu pengguna (relasi ke user lewat id_user)
 - Tapi satu pengguna bisa membuat banyak tugas (relasi satu ke banyak: 1 user → banyak tugas)
- Isi data tugas mencakup: Judul, deskripsi, status (belum selesai atau selesai), tenggat waktu, dan waktu dibuat

3. Tabel Berkas

- Satu berkas terkait dengan satu tugas (id_tugas)
- Satu berkas juga terkait dengan satu pengguna (id_user). Tapi sebaliknya, satu tugas bisa punya banyak berkas, dan satu user juga bisa mengunggah banyak berkas

RANCANGAN WEBSITE

1. Halaman Buat Akun Baru



Buat Akun Baru
Belum punya akun? Daftar Disini!

NAMA
Makoto Nijima

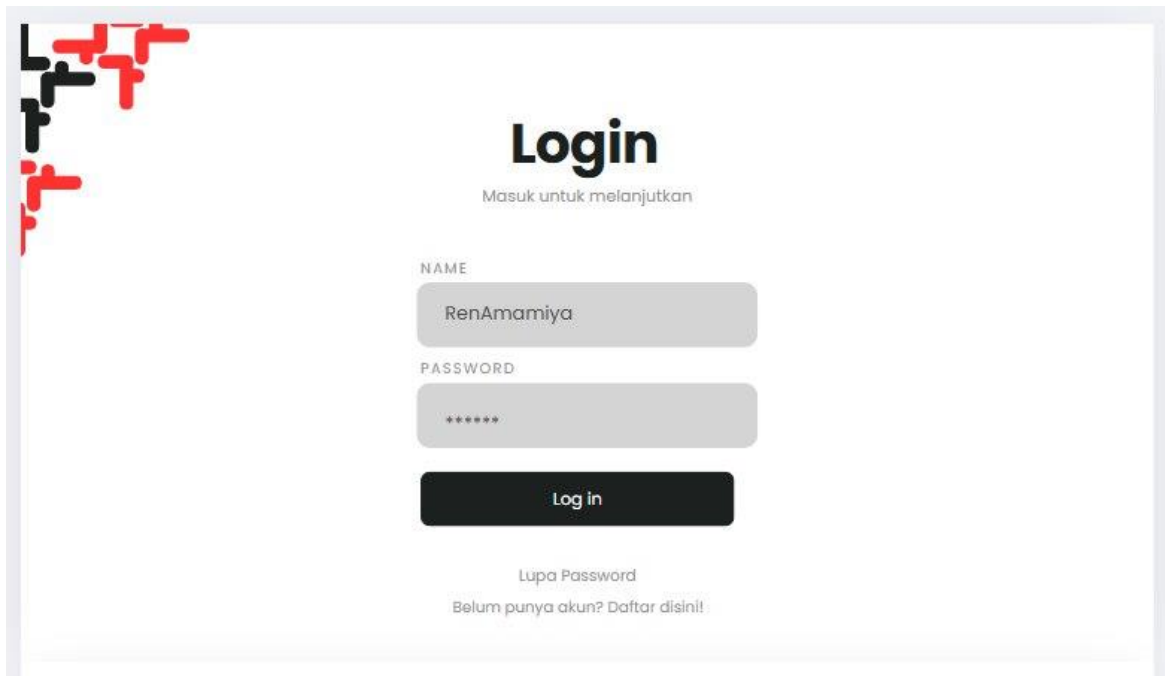
EMAIL
nijimamakoto223@gmail.com

PASSWORD

TANGGAL LAHIR
PILIH

Signup

2. Halaman Login



Login
Masuk untuk melanjutkan

NAME
RenAmamiya

PASSWORD

Log in

Lupa Password
Belum punya akun? Daftar disini!

3. Halaman Utama



4. Halaman Tambah Tugas

The screenshot shows the 'Tambah Tugas' (Add Task) page. At the top, there is a navigation bar with a hamburger menu icon on the left, a search bar with the placeholder text 'Cari tugas kamu' and a magnifying glass icon, and a user profile section on the right displaying the location 'Kyoto, Jepang', a phone icon, an email icon, and the username '@RenAmamiya'. The main content area is titled 'Tambah Tugas' in a bold, black, sans-serif font. Below the title, there are four input fields: 'Mata Kuliah' (Course), 'Deskripsi' (Description), 'Deadline Tugas' (Task Deadline), and 'File'. The 'File' field is a larger area with a placeholder text 'Seret dan Letakan file yang ingin anda tambahkan disini' (Drag and drop the file you want to add here). At the bottom of the form is a large red button with the text 'TAMBAHKAN TUGAS' in white.

5. Halaman Edit Tugas

Edit Tugas

Kyoto, Jepang

@RenAmamiya

Mata Kuliah

Deskripsi

Deadline Tugas

File

Seret dan Letakan file yang ingin anda tambahkan disini

EDIT TUGAS

TANDAI SELESAI

6. Halaman Daftar Tugas

Tugasku

Home

Daftar Tugas

Akun dan Settings

Logout

kritik & saran

Cari tugas kamu

Kyoto, Jepang

@RenAmamiya

+ TUGAS

Pemrograman
Membuat game pakai bahasa java.
EDIT TUGAS Deadline 11 April 2025

Pemrograman Objek
Membuat game menggunakan bahasa java.
EDIT TUGAS Deadline 10 Juni 2025

Matematika Dasar
Tugas matriks.
EDIT TUGAS Deadline 11 Maret 2025

Ekonomi
Belajar Saham
EDIT TUGAS Deadline 18 April 2025

Kewarganegaraan
Membuat makalah
EDIT TUGAS Deadline 2 Februari 2025 **SELESAI**

Rekayasa perangkat lunak
Testing Software yang telah dibuat.
EDIT TUGAS Deadline 5 April 2025

7. Halaman Rincian Tugas



8. Halaman Profile

