**Analyst**

**Aplikasi To-Do List berbasis Web**

**Anggota Kelompok:**

1. Muhammad Hasan Nurrahman (Analyst)
2. Naufal Muhammad Yusuf (Backend)
3. Syahdan Arifat (Frontend)

**Latar Belakang Proyek**

Dalam era digital saat ini, manajemen waktu dan produktivitas menjadi hal yang sangat penting, terutama bagi pelajar dan pekerja. Oleh karena itu, kami memutuskan untuk membuat aplikasi to-do list berbasis web yang dapat membantu pengguna dalam mengatur dan mengingat tugas-tugas mereka. Aplikasi ini memungkinkan pengguna untuk menambahkan tugas dengan kategori tertentu, menetapkan tenggat waktu, serta mendapatkan notifikasi sebelum deadline agar tidak melewatkan pekerjaan penting. Selain itu, kami juga ingin memastikan bahwa setiap tugas yang dibuat tetap tersimpan di akun pengguna, sehingga mereka bisa mengaksesnya kembali setelah login. Dengan tampilan yang simpel dan fitur yang interaktif seperti modal input tugas dan update data, saya berharap aplikasi ini dapat memberikan pengalaman yang nyaman dan efisien bagi pengguna dalam mengelola aktivitas sehari-hari.

**Tujuan**

Tujuan dari aplikasi to-do list ini adalah untuk membantu pengguna mengelola tugas mereka dengan lebih efisien melalui fitur pencatatan tugas, pengkategorian, pengingat sebelum deadline, serta penyimpanan data berbasis akun agar bisa diakses kapan saja. Dengan desain yang simpel dan interaktif, aplikasi ini diharapkan dapat meningkatkan produktivitas pengguna dalam menyelesaikan tugas sehari-hari.

**Dokumen Kebutuhan**

**Kebutuhan Fungsional**

* Pengguna dapat membuat akun agar tugas tersimpan.
* Pengguna dapat membuat tugas.
* Pengguna dapat mengedit tugas.
* Pengguna dapat menghapus tugas.
* Pengguna dapat mengategorikan tugas-tugas (sekolah, pekerjaan, pribadi).
* Pengguna akan mendapatkan notifikasi sederhana jika tugas tinggal 1 hari, 1 jam, 10 menit sebelum tenggat waktu, dan saat tugas menyentuh tenggat waktu.

**Kebutuhan Non-Fungsional**

* Website yang dapat diakses melalui perangkat Desktop.

**Spesifikasi Teknis**

**Teknologi yang digunakan**

* **Frontend:** HTML, CSS, Javascript.
* **Backend:** PHP.
* **Database:** MySQL.
* **Server:** XAMPP.
* **Browser:** Google Chrome.
* **Kode Edtor:** Visual Studio Code.

**API yang dibutuhkan**

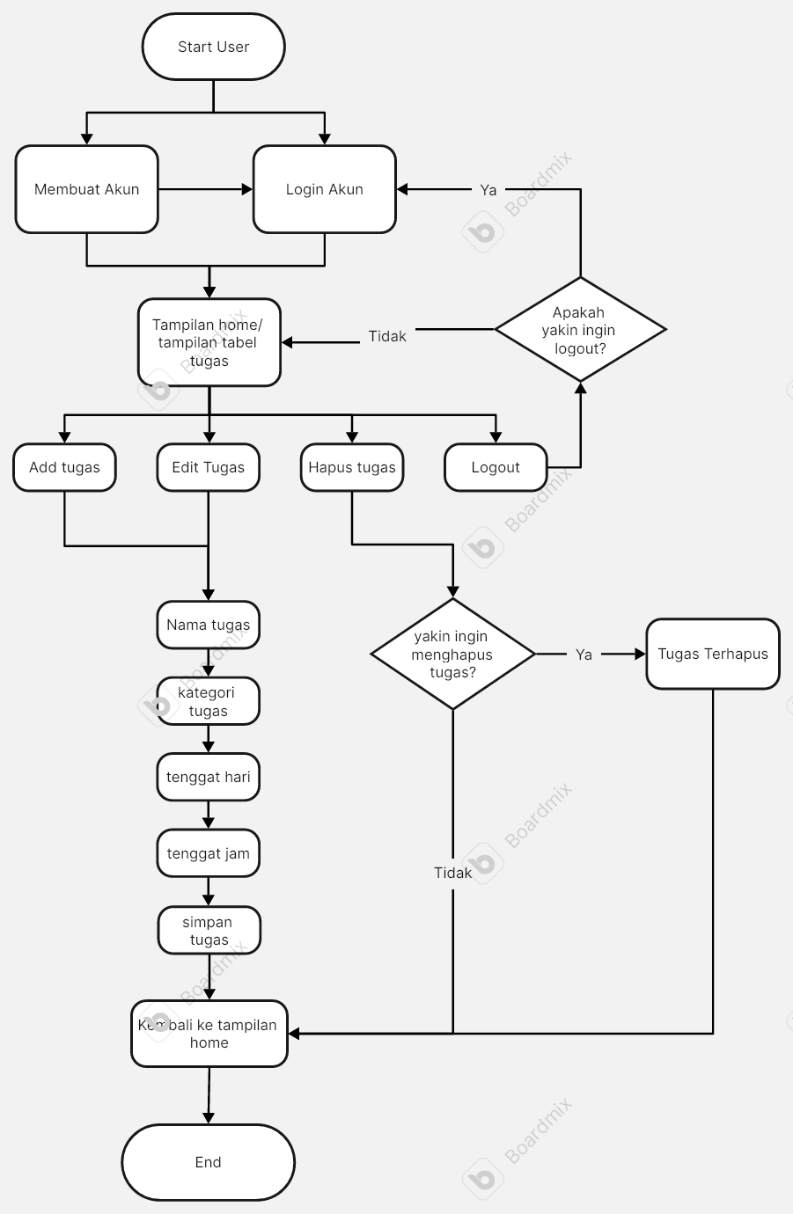
* API untuk membuat sebuah akun.
* API untuk memverifikasi username dan password untuk proses login.
* API untuk membuat tugas.
* API untuk mengedit tugas.
* API untuk menghapus tugas.

**3. Use Case and Workflow**

**Use case:**

**Aktor:**

1. User



1. User login ke sebuah akun -> User diarahkan ke tampilan tabel tugas -> User dapat membuat atau mengedit sebuah tugas -> User mengisi/mengubah isi dari tugas -> Tugas tersimpan.

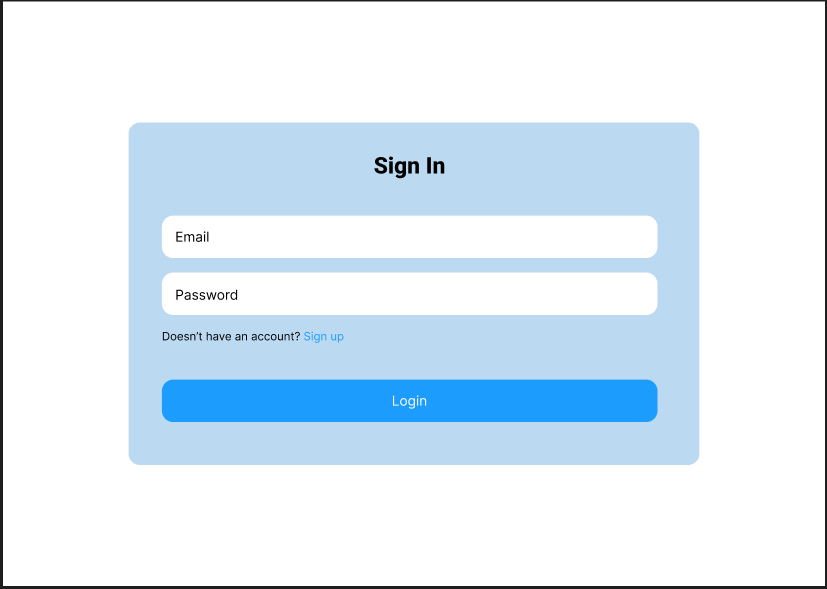
**Workflow proses membuat dan mengedit tugas**

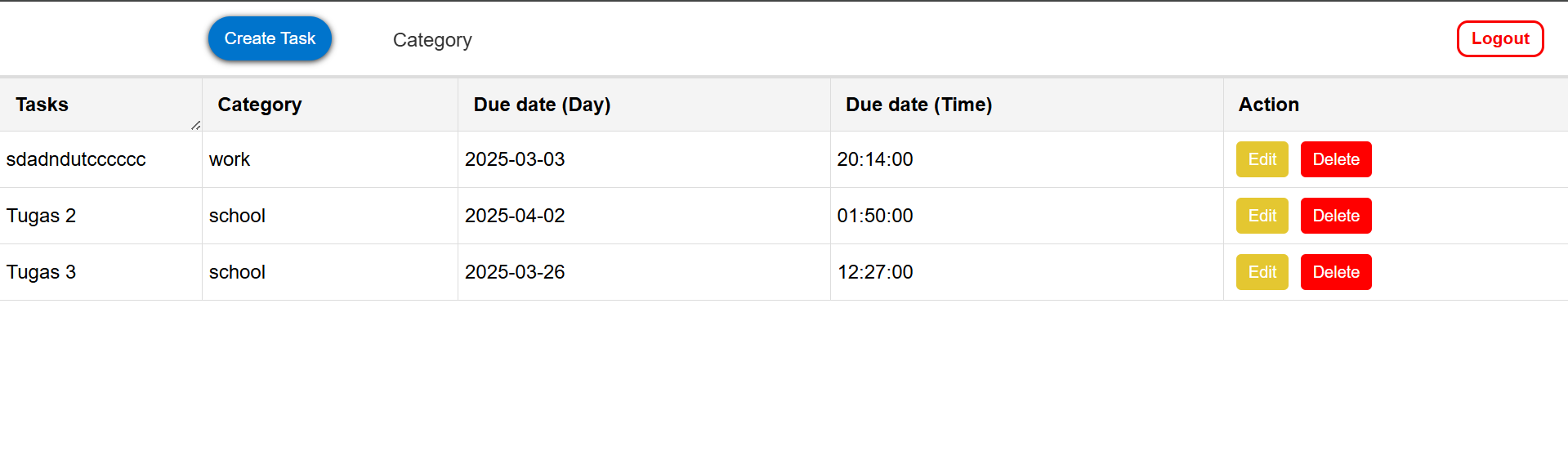
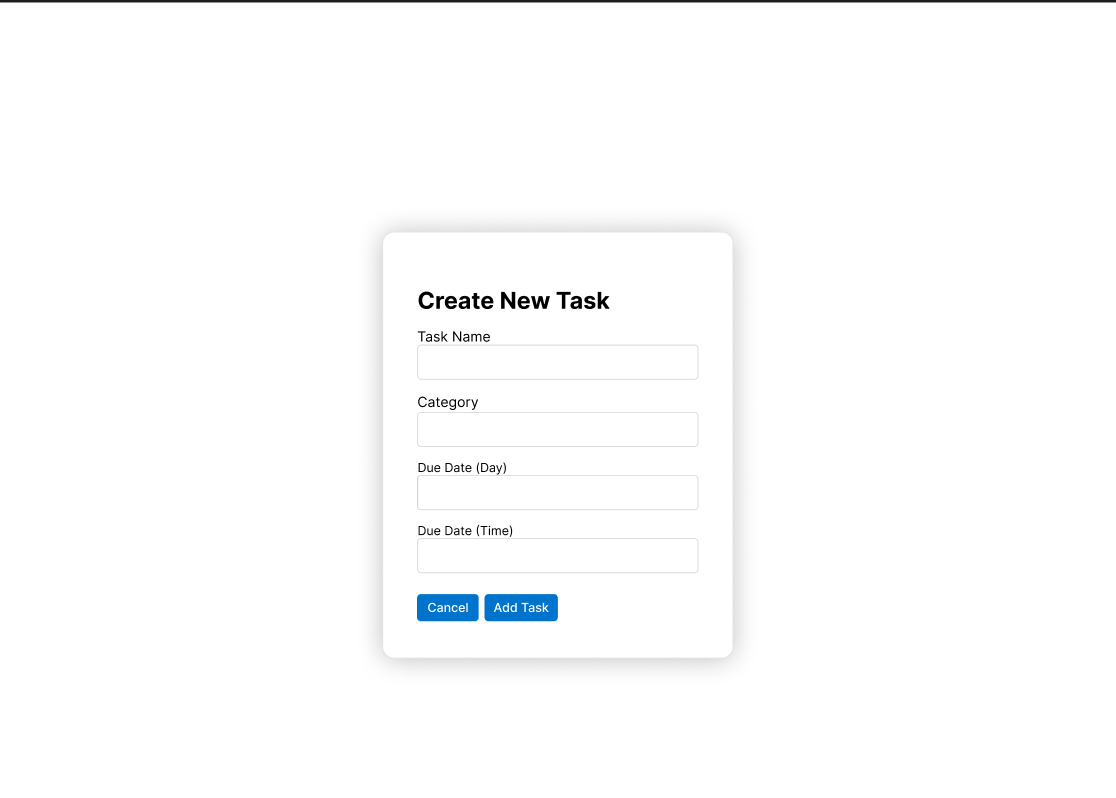
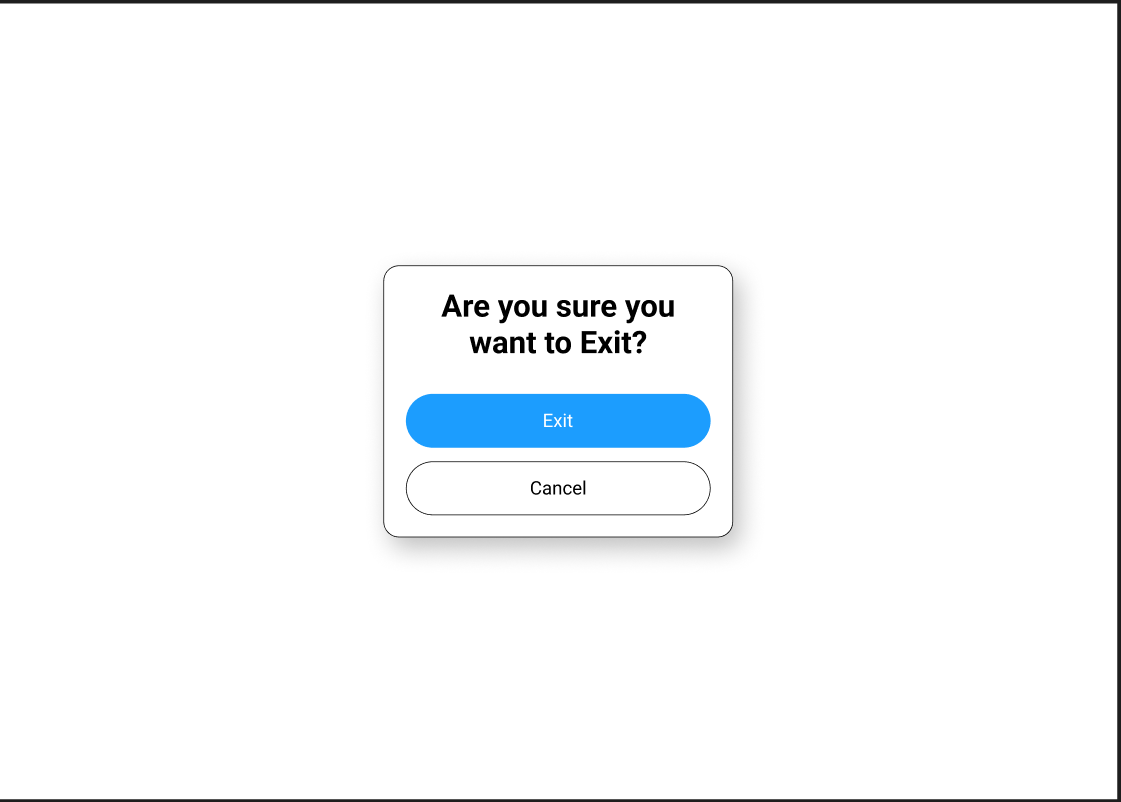
1. Pengguna login ke sebuah akun.
2. Pengguna menekan tombol “Create task”.
3. Pengguna mengisi beberapa pertanyaan.
4. Tugas tersimpan.

**4. Mockup (Desain Tampilan)**

* Halaman utama dengan Tabel berisi kumpulan tugas.
* Halaman Login dan Registrasi Akun.
* Modal Popup untuk membuat, mengedit, dan logout.

**Desain MOCKUP**

**Sign In**

Sign Up   

1. **Dokumen Rencana Proyek**

**Tahapan Pengembangan**

1. **Analisis Kebutuhan:** Identifikasi fitur yang diperlukan.
2. **Desain Sistem:** Pembuatan UI dan arsitektur sistem.
3. **Pengembangan:** Implementasi backend dan frontend.
4. **Pengujian:** Uji coba untuk memastikan website berjalan dengan baik.
5. **Implementasi:** Peluncuran website untuk publik.
6. **Pemeliharaan:** Perbaikan bug dan pembaruan fitur.

**Alat yang digunakan**

1. XAMPP

**6. Dokumen Pengujian**

**Pengujian Keamanan**

* Validasi input untuk mencegah SQL Injection.
* Enkripsi Password User.

**Pengujian Frontend**

* Pengujian di beberapa aplikasi browser.

**Pengujian Fungsional**

* Simulasi pembuatan Tugas.
* Pengujian fitur Edit pada Tugas.
* Pengujian fitur Hapus pada Tugas.
* Pengujian fitur Logout pada Akun.
* Simulasi Registrasi dan Login pada Akun.

1. **Panduan Pengguna**

**Untuk Pengguna:**

1. Buka website To-Do List
2. Buat/Login Akun.
3. Tekan tombol “Create Task” untuk membuat Tugas
4. Isi beberapa input mengenai Tugas
5. Tugas berhasil disimpan dan ditampilkan di halaman Home.
6. **Rancang Database**

**Entitas dan Relasi dalam ERD**

Entitas:

1. **Tb\_list:** Tabel ini berfungsi untuk menyimpan tugas tugas yang dibuat oleh pengguna.

* User (user\_id)

Kolom ini berfungsi untuk menyimpan ID pengguna agar tabel bisa menampilkan tugas sesuai dengan akun yang telah dimasukkan oleh pengguna.

* ID (id)

Kolom ini berfungsi untuk menampilkan nomor tugas yang telah dibuat oleh pengguna. Urutan nomornya hanya bisa dilihat di MySQL.

* Nama Tugas (task\_name)

Kolom ini berfungsi untuk menyimpan kumpulan nama-nama tugas yang telah dibuat oleh pengguna.

* Kategori (category)

Kolom ini berfungsi untuk menyimpan kategori tugas yang telah dipilih oleh pengguna, kami telah menyediakan tiga opsi kategori untuk tugas (Sekolah, Pekerjaan, dan Pribadi).

* Tenggat hari (due\_date\_day)

Kolom ini berfungsi untuk menyimpan Tenggat hari dari tugas yang waktunya telah disesuaikan oleh pengguna. Nantinya kolom ini juga akan berfungsi untuk menampilkan notifikasi ketika waktu telah mendekati deadline.

* Tenggat jam (due\_date\_time)

Kolom ini berfungsi untuk menyimpan Tenggat jam dari tugas yang waktunya telah disesuaikan oleh pengguna. Sama seperti kolom tenggat hari, nantinya kolom ini akan disambungkan dengan kolom tenggat hari agar notifikasi bisa muncul ketika waktu mendekati deadline.

* Tanggal pembuatan (created\_at)

Kolom ini berfungsi untuk menyimpan tanggal pembuatan. Kolom ini tidak akan berfungsi jika dilihat dari pengguna karena pengguna tidak akan bisa melihatnya dan hanya Developer yang dapat melihatnya melalui tabel MySQL.

1. **Tb\_signup:** Tabel ini berfungsi untuk menyimpan akun-akun yang telah dibuat oleh pengguna.

* ID (id)

Kolom ini berfungsi untuk menyimpan urutan nomor akun yang dapat menjadi acuan system untuk membedakan antara sesame pengguna keetika membuat tugas.

* Nama (username)

Kolom ini berfungsi untuk menyimpan username pengguna yang dapat digunakan ketika nantinya ia akan login kedalam akun.

* Email (email)

Kolom ini berfungsi untuk menyimpan email pengguna.

* Password (password)

Kolom ini berfungsi untuk menyimpan Password pengguna ke dalam tabel. Siapapun yang mengakses tabel ini tidak akan bisa melihat kumpulan password pengguna karena sudah ter Enkripsi.

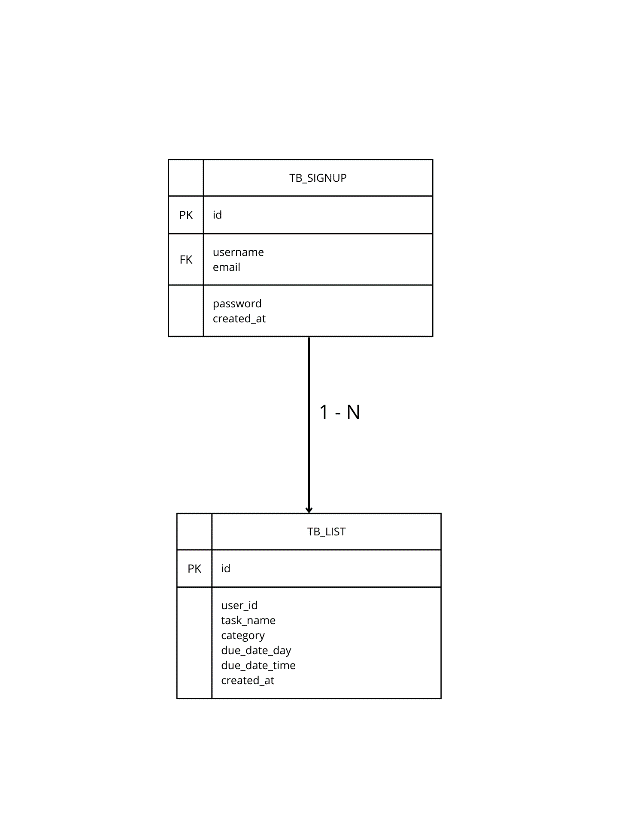
* Tanggal pembuatan (created\_at)

Kolom ini berfungsi untuk menyimpan tanggal pembuatan. Kolom ini tidak akan berfungsi jika dilihat dari pengguna karena pengguna tidak akan bisa melihatnya dan hanya Developer yang dapat melihatnya melalui tabel MySQL.

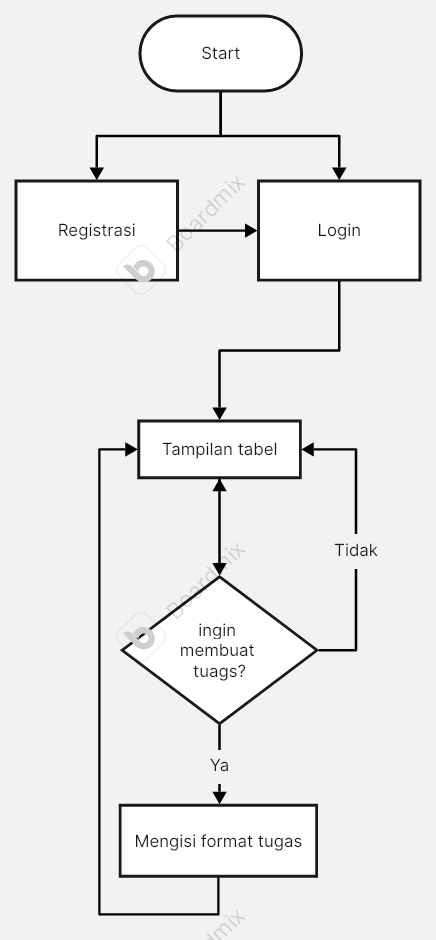
Relasi:

* Akun (1) -> (N) Tugas: Satu akun bisa membuat banyak tugas.

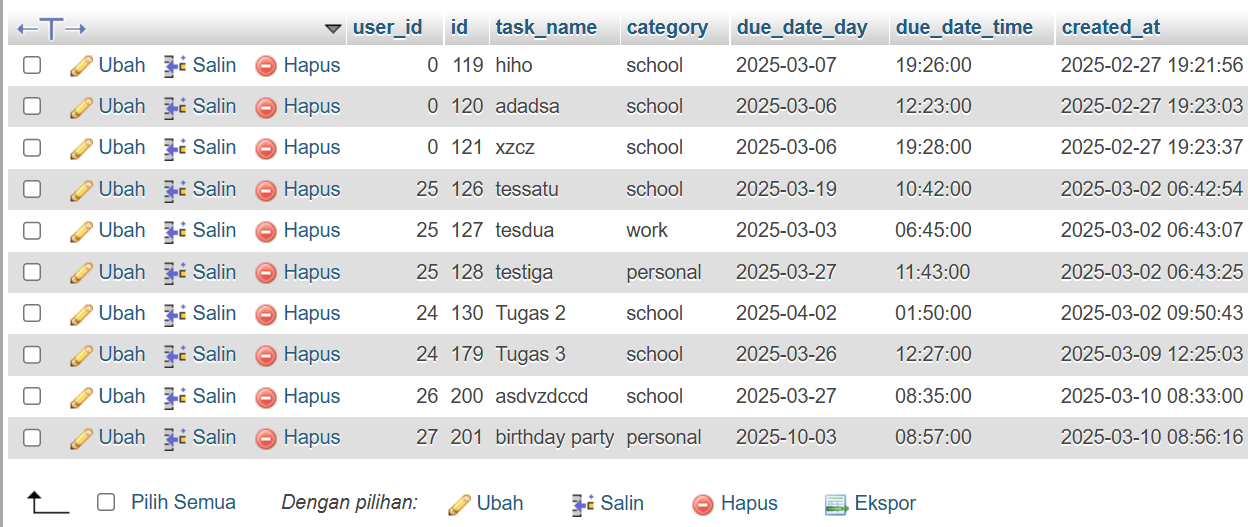
**DESAIN ERD**

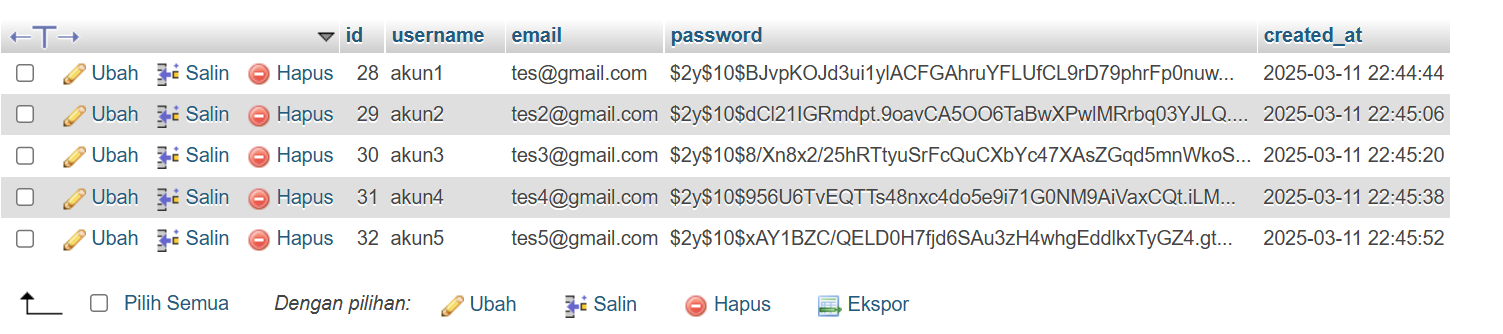
****

**DESAIN FLOWCHART**

****

**DESAIN DATABASE**

****

****

1. **Rancang Atribut Frontend dan Analyst**

**Frontend:**

* Struktur HTML untuk Halaman Login, Registrasi, Home, dan Modal.
* Desain CSS dan untuk Desain.
* Implementasi JavaScript untuk interaksi pengguna.

**Analyst:**

* Analisis kebutuhan pengguna.
* Penyusunan dokumen spesifikasi teknis.
* Pengembangan skenario pengujian.
* Dokumentasi hasil pengujian dan rekomendasi peningkatan.