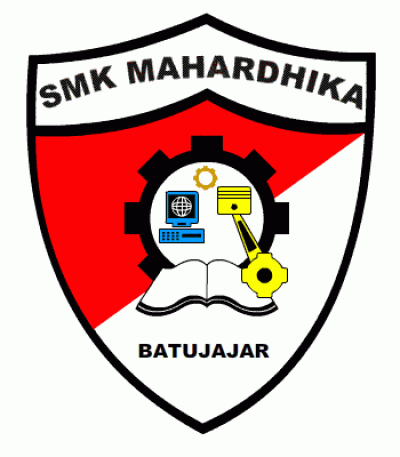
**LAPORAN PENGERJAAN**

**UJI KOMPETENSI KEJURUAN**



**KOMPETENSI KEAHLIAN : Rekayasa Perangkat Lunak**

**KELAS : XII RPL 2**

**NAMA LENGKAP : Ardita**

**NISN :**

**SMK MAHARDHIKA**

**TAHUN PELAJARAN 2024-2025**

**LAPORAN KEGIATAN UJI KOMPETENSI**

**KEJURUAN SMK MAHARDHIKA BATUJAJAR**

**TAHUN PELAJARAN 2024-2025**

KOMPETENSI KEAHLIAN : Rekayasa Perangkat Lunak

KELAS : XII RPL 2

NAMA LENGKAP : Ardita

NIS : 2223.10.395

PAKET SOAL : 2

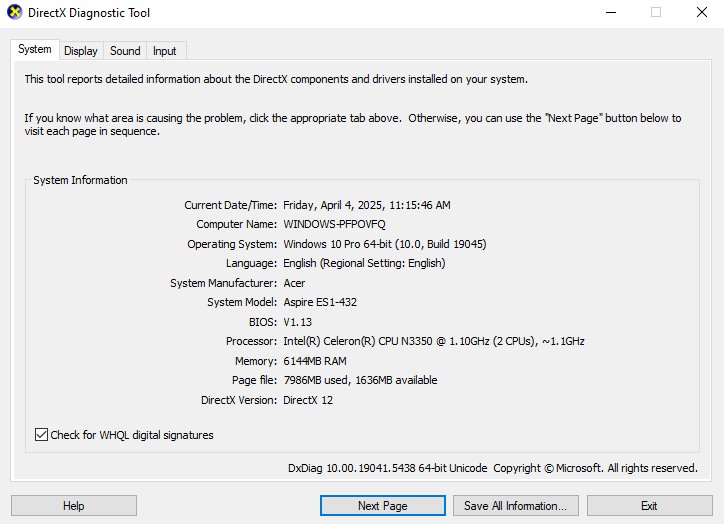
JENIS APLIKASI : Website

ALOKASI WAKTU : 8 jam

# 1. PENJADWALAN KEGIATAN

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kegiatan | Minggu 1 | | | Minggu 2 | | | Minggu 3 | | | | Minggu 4 | | | | Minggu 5 | | | | | Minggu 6 | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | | 11 | 12 | 13 | | 14 | 15 | 16 | 17 | | 18 | 19 | 20 |
| Analisis Soal |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Perancangan Sistem |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Perancangan DB |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Perancangan UI/UX |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Implementasi DB |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Implementasi UI/UX |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Implementasi Program |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Finalisasi Program |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Pengujian Program |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Hosting + Github |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Finalisasi Dokumentasi User Manual |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Finalisasi Projek UKK |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |
| Dokumentasi |  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  |  | |  |  |  |

# 2. PERSIAPAN PERANGKAT KERAS



# 3. PERSIAPAN PERANGKAT LUNAK

Pembuatan aplikasi *to-do list* berbasis *website* ini membutuhkan perangkat lunak yang dijalankan untuk aplikasi yaitu xampp, visual studio code, dan chrome. Serta aplikasi tambahan untuk pembuatan aplikasi menggunakan astah dan balsamiq. Berikut penjelasan mengenai aplikasi aplikasi tersebut.

1. *Website*

*Website* adalah kumpulan halaman *web* yang saling terhubung dan seluruh file saling terkait. *Web* terdiri dari *page* atau halaman dan kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas dengan halaman-halaman terkait berada di bawahnya. Biasanya, setiap halaman di bawah *homepage* (*child page*) berisi *hyperlink* ke halaman lain dalam *web* (Gregorius, 2000). [3]

1. *Website* merupakan fasilitas internet yang menghubungkan dokumen dalam lingkup lokal maupun jarak jauh. Dokumen pada *website* disebut dengan *web page* dan *link* dalam *website* memungkinkan pengguna bisa berpindah dari satu *page* ke *page* lain (*hypertext*), baik antara *page* yang disimpan dalam server yang sama maupun server di seluruh dunia. Halaman dapat diakses dan dibaca melalui *browser* seperti Google Chrome, Mozilla Firefox, dan lainnya (Hakim Lukmanul, 2004).
2. *Website* adalah halaman in*form*asi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet. *Website* merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara, dan animasi sehingga menarik untuk dikunjungi (Sholechul Azis, 2013).



Logo Website

1. HTML

HTML adalah kependekan dari *Hypertext Markup Language* yang merupakan sebuah bahasa *markup*. HTML adalah kode untuk membuat struktur halaman suatu *website* yang menarik, saling terhubung satu dengan yang lainnya, dan yang pasti dapat diakses melalui internet.

Awalnya HTML ditemukan oleh Tim Berners-Lee pada tahun 1991. HTML adalah solusi untuk membantu ilmuwan dalam mengakses dokumen, namun kini html semakin berkembang pesat di dunia pemrograman *web*.

Bagi pemula yang ingin memulai belajar pemrograman *web* dan bingung belajar bahasa markup apa, HTML adalah jawabannya. HTML dianggap mudah dipahami oleh orang awam karena tag dasar html mudah dibaca dan diimplementasikan. [4]



Logo HTML

1. CSS

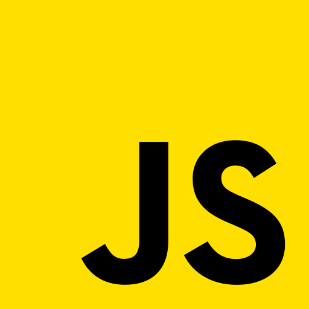
*Cascading Style Sheet* (disingkat CSS) adalah bahasa lembar gaya yang digunakan sebagai penentu presentasi dan gaya dokumen yang ditulis dalam bahasa markup seperti HTML dan XML. CSS termasuk ke dalam teknologi dasar *World Wide Web*, bersama dengan HTML dan *JavaScript*.



Logo CSS

1. *JavaScript*

*JavaScript* adalah bahasa pemrograman yang digunakan *developer* untuk membuat halaman *web* yang interaktif. Dari menyegarkan umpan media sosial hingga menampilkan animasi dan peta interaktif, fungsi *JavaScript* dapat meningkatkan pengalaman pengguna situs *web*. Sebagai bahasa skrip sisi *klien*, *JavaScript* adalah salah satu teknologi inti dari *World Wide Web.* Misalnya, saat menjelajah internet, kapan pun Anda melihat carousel gambar, menu tarik-turun klik untuk menampilkan, atau warna elemen yang berubah secara dinamis di halaman *web*, Anda melihat efek *JavaScript.*



Logo JavaScript

1. PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *open-source* yang umumnya digunakan untuk membangun aplikasi *web* dinamis dan interaktif. PHP dapat dijalankan pada server *web* dan dikombinasikan dengan HTML, CSS, dan *JavaScript* untuk membuat halaman *web* yang dinamis.

Saat ini, PHP sangat populer di kalangan *web developer* karena mudah dipelajari dan memiliki kemampuan yang cukup kuat. PHP juga mendukung banyak jenis *database*, seperti MySQL, PostgreSQL, dan Oracle sehingga memungkinkan pengembang untuk membuat aplikasi *web* yang lebih kompleks dan fungsional.



Logo PHP

1. Bootstrap

Bootstrap adalah salah satu *framework front-end development* yang populer untuk mengembangkan aplikasi *web* *responsive* dan *mobile-first*. Framework ini dibangun berbasis HTML, CSS, dan *JavaScript*. Komponen-komponen dasarnya dibangun menggunakan kombinasi HTML untuk struktur, CSS untuk *style* dan *layout* dan *JavaScript* untuk interaktif. Bootstrap memanfaatkan *jQuery*, sebuah *library* *JavaScript* populer, untuk menangani interaksi dan animasi.

Bootstrap awalnya dikembangkan oleh Mark Otto dan Jacob Thornton di Twitter sebagai proyek internal. Mereka merilis Bootstrap secara publik pada tahun 2011 sebagai proyek *open source*, itulah mengapa nama awal Bootstrap adalah Twitter *Blueprint.*

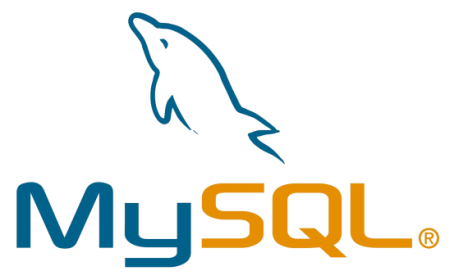
Tujuan dari *Bootstrap* adalah untuk menyederhanakan pengembangan *web* sehingga *developer* tidak perlu menulis kode CSS dari nol, melainkan bisa menggunakan kumpulan template CSS dan *JavaScript* yang sudah dibuat untuk komponen antarmuka. [5]



Gambar 3. 1 Logo Bootstrap

1. MySQL

MySQL merupakan sistem manajemen *database* yang bersifat open-source yang menggunakan perintah dasar atau bahasa pemrograman yang berupa structured query language (SQL) yang cukup populer di dunia teknologi yang penyusun gunakan untuk menyimpan *database* untuk sistem in*form*asi stok dan inventaris.



Gambar 3. 2 Logo MySQL

1. *Activity Diagram*

*Activity Diagram*, dalam bahasa Indonesia diagram aktivitas, yaitu *diagram* yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity Diagram* merupakan pengembangan dari *Use Case* yang memiliki alur aktivitas. [6]

1. *Use Case Diagram*

Use case diagram adalah satu dari berbagai jenis diagram UML (*Unified Modelling Language*) yang menggambarkan hubungan interaksi antara sistem dan aktor. *Use Case* dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya. [7]

1. *Class Diagram*

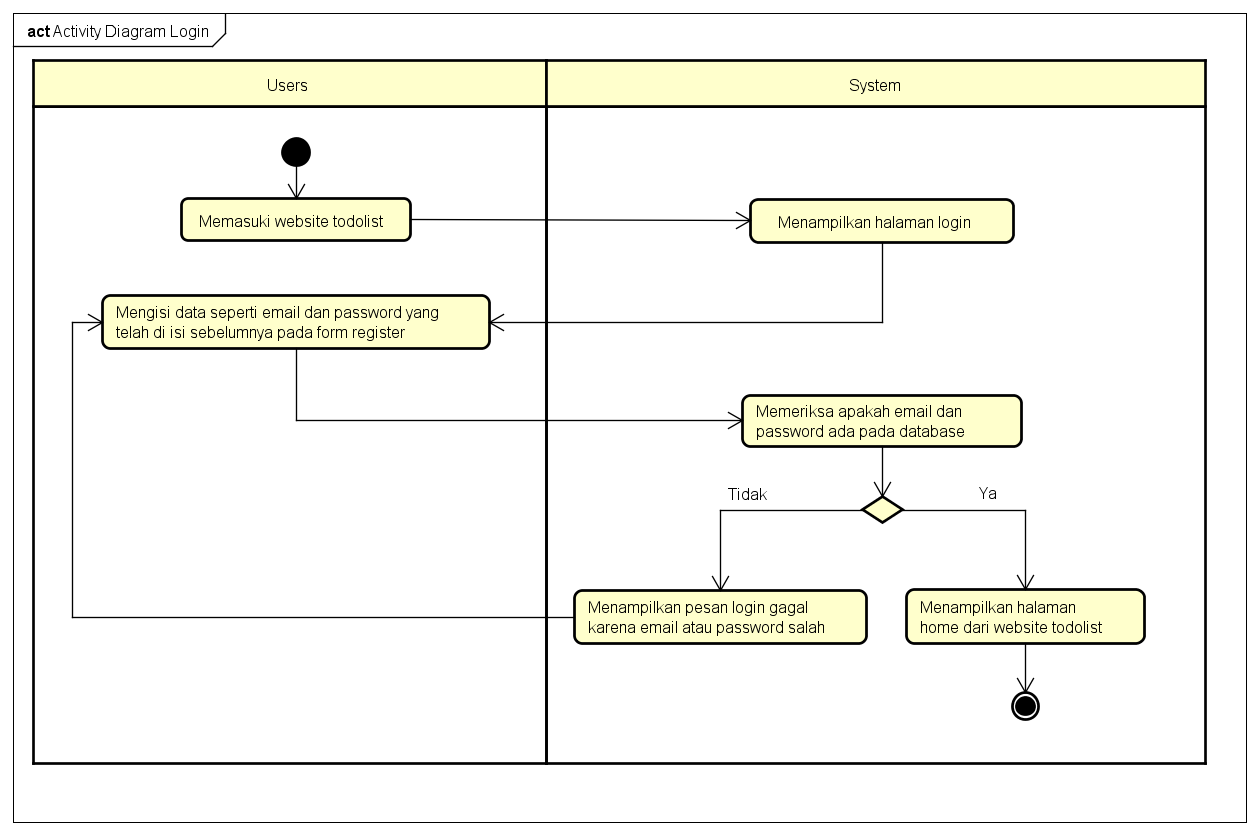
*Class diagram* atau diagram kelas adalah salah satu jenis diagram struktur pada UML yang menggambarkan dengan jelas struktur serta deskripsi *class*, atribut, metode, dan hubungan dari setiap objek. Ia bersifat statis, dalam artian diagram kelas bukan menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, melainkan menjelaskan hubungan apa yang terjadi.

1. SQL *Designer*

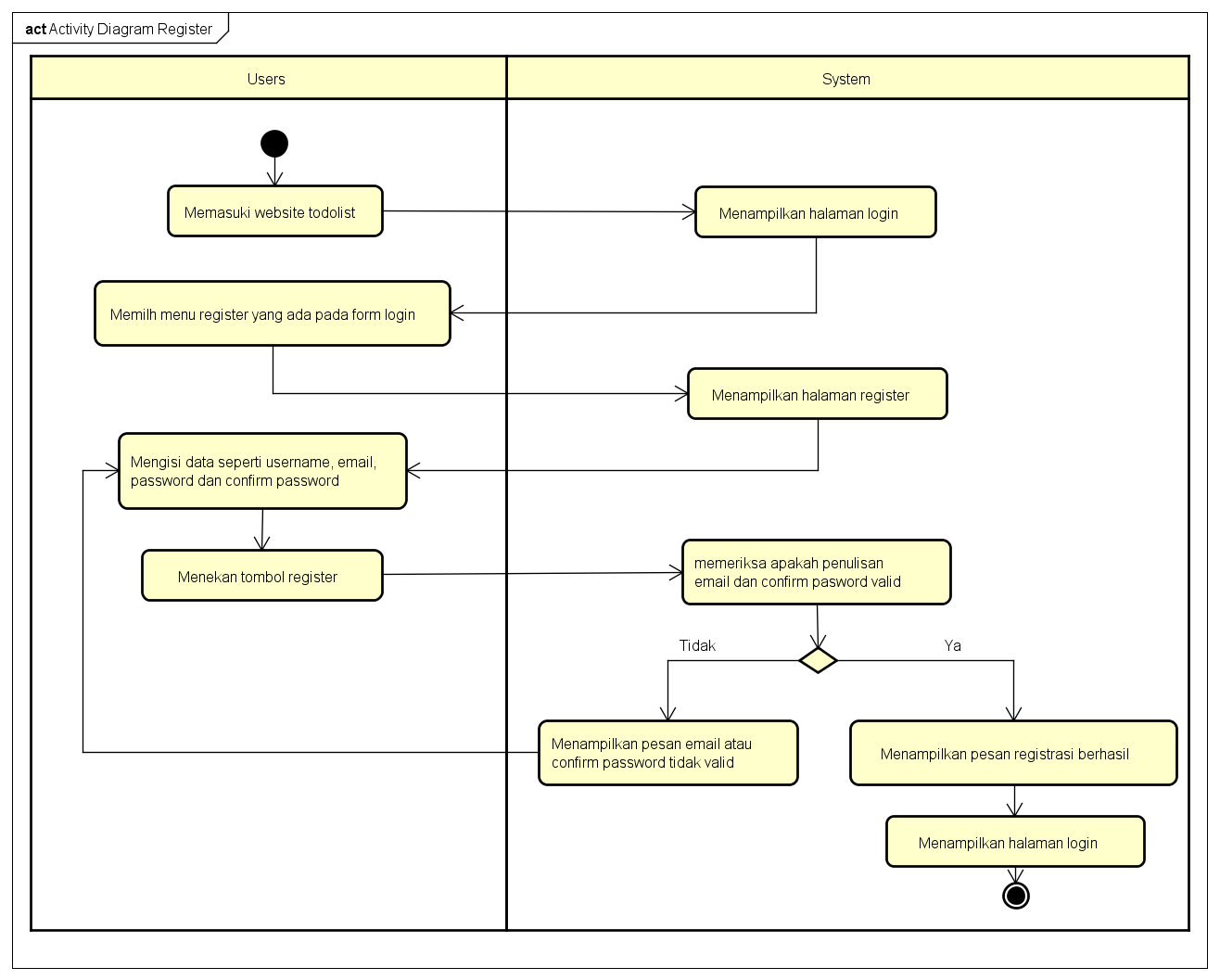
SQL *Designer* digunakan untuk menjelajahi dan membuat kueri SQL untuk titik pemeriksaan basis data dan variabel Tabel DB. Dengan menggunakan SQL *Designer*, Anda juga dapat melihat nama tabel dan kolom yang akan digunakan titik pemeriksaan atau variabel, serta membuat kueri secara visual.

# 4. PERANCANGAN ALGORITMA APLIKASI

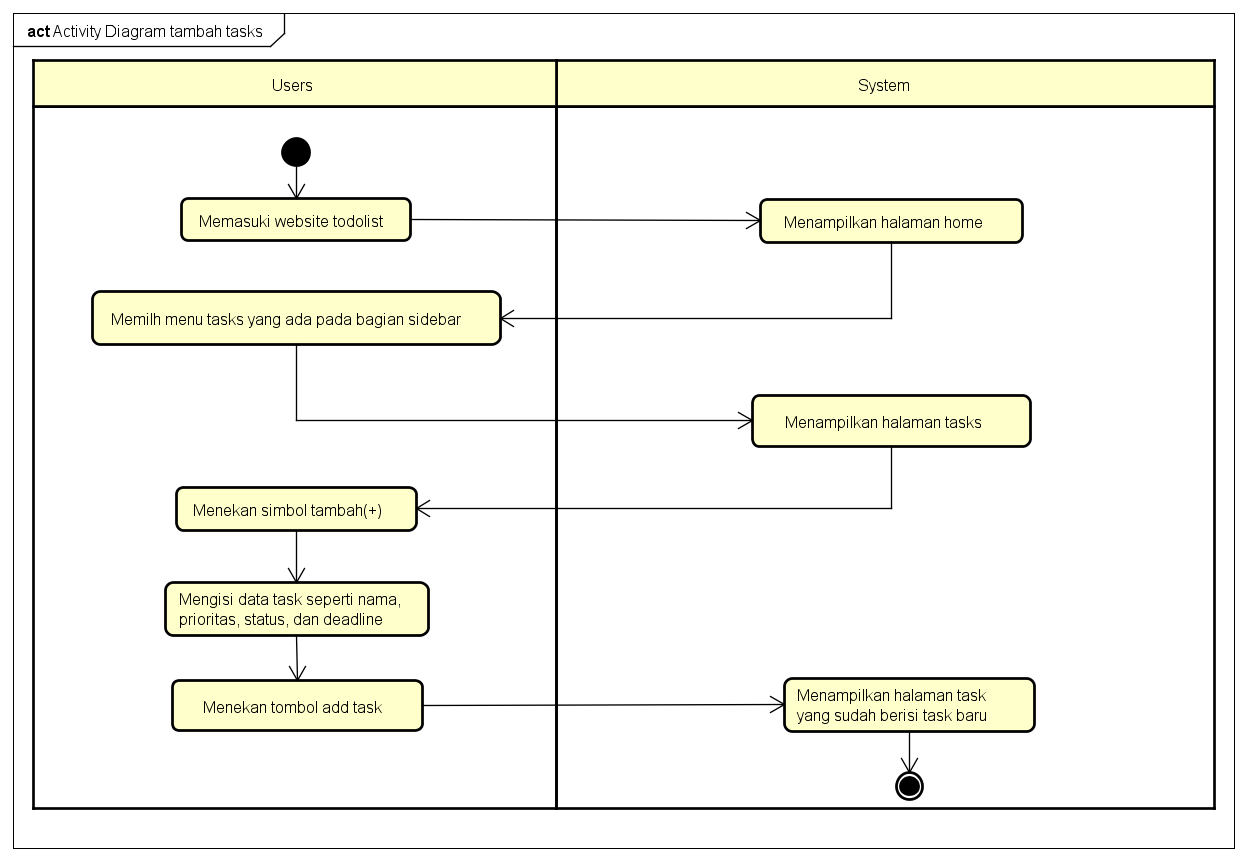
*Activity Diagram Login*



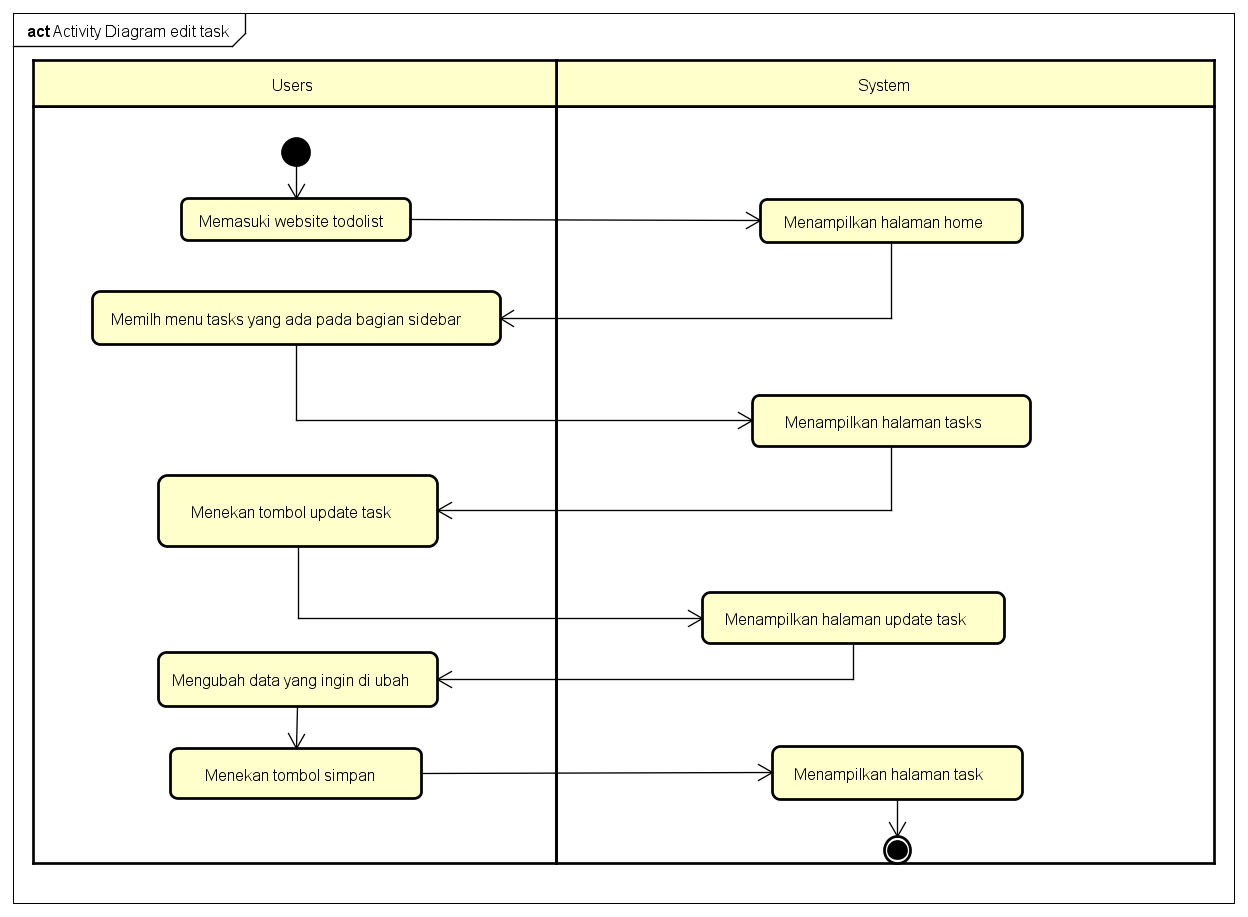
*Activity Diagram Register*



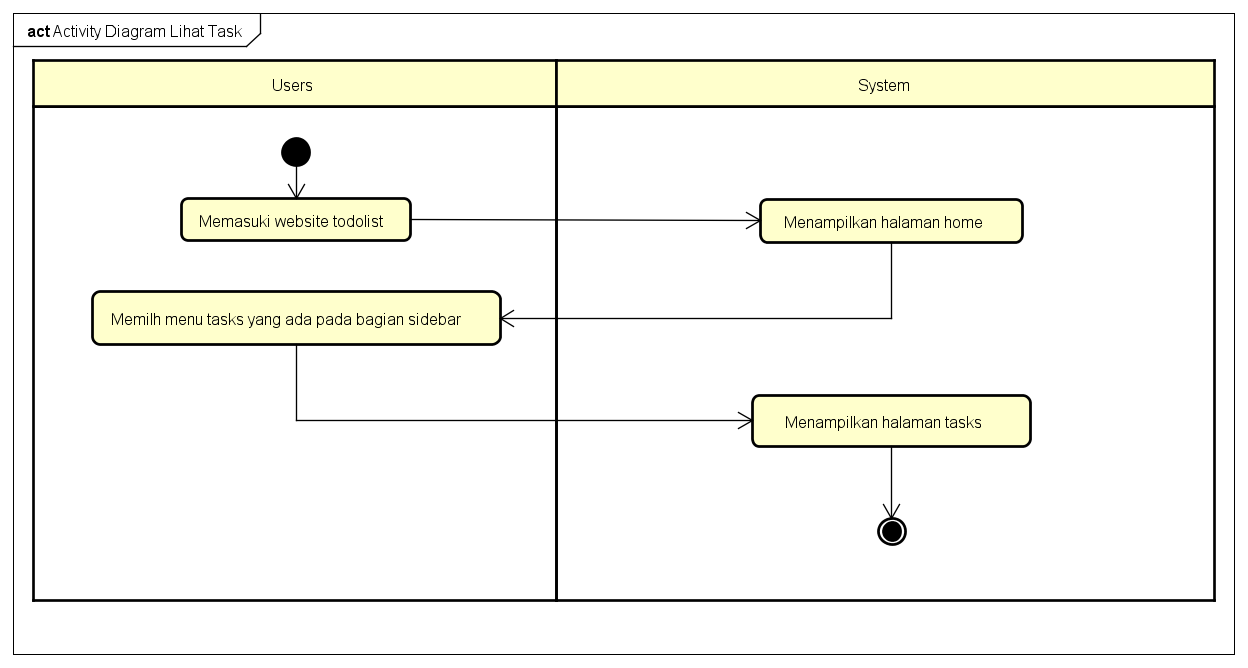
*Activity Diagram* Tambah *Task*



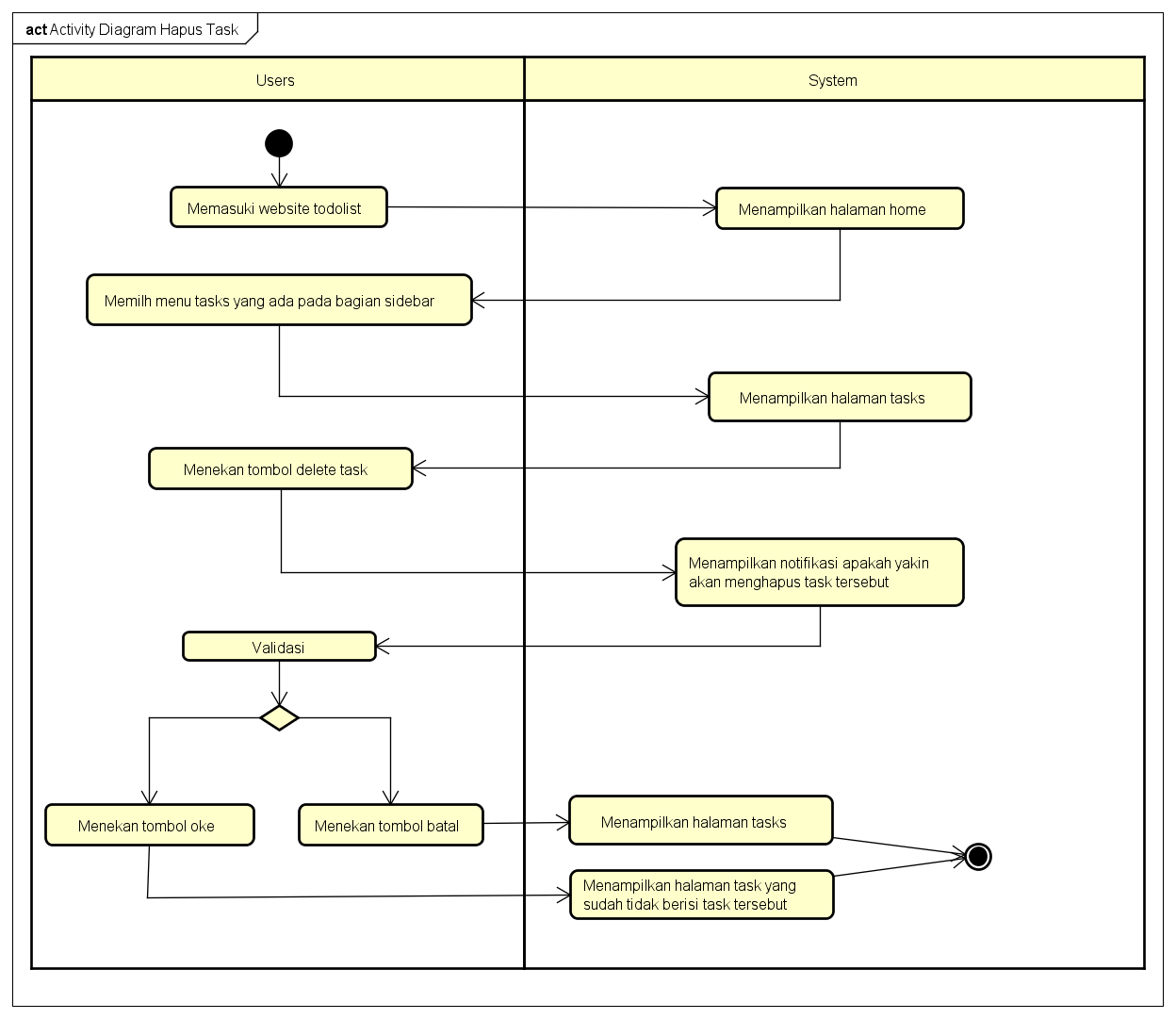
*Activity Diagram* Edit *Task*



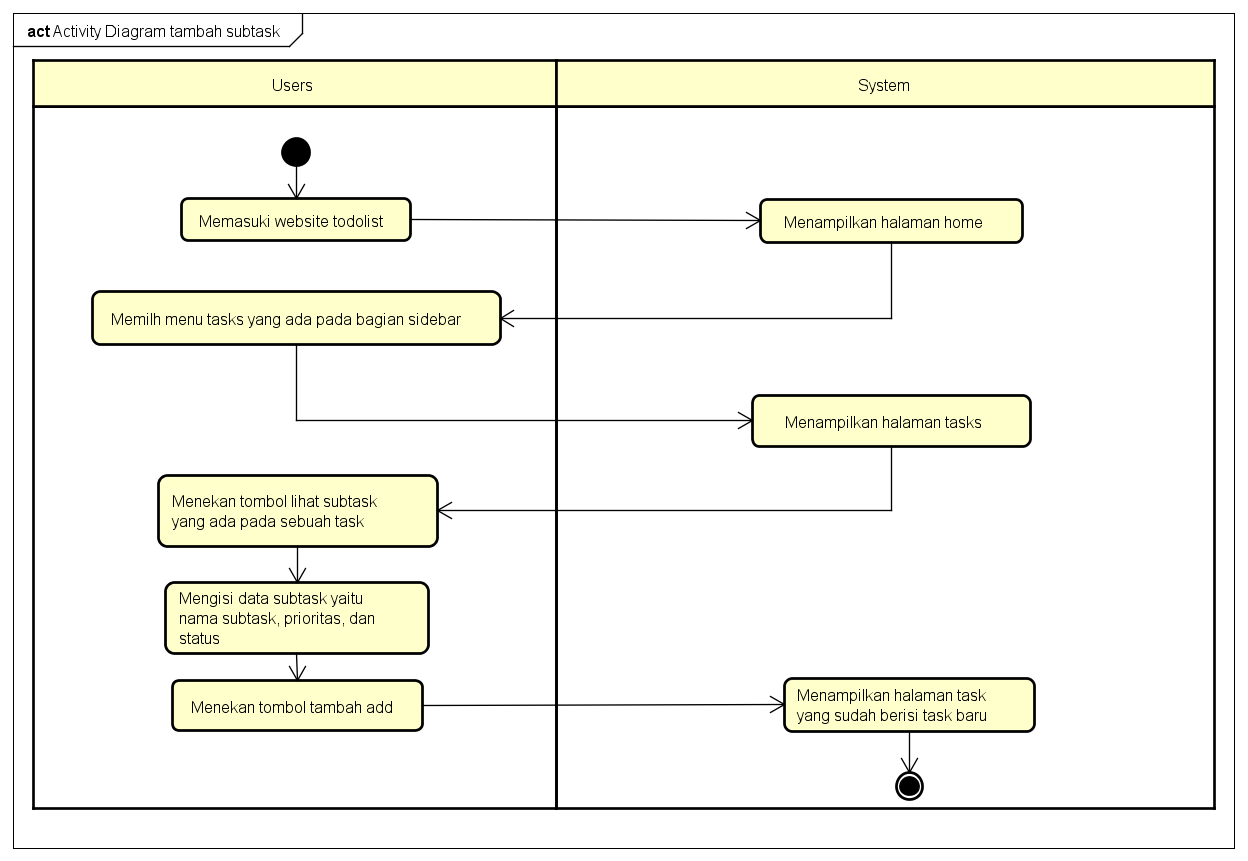
*Activity Diagram* Lihat *Task*



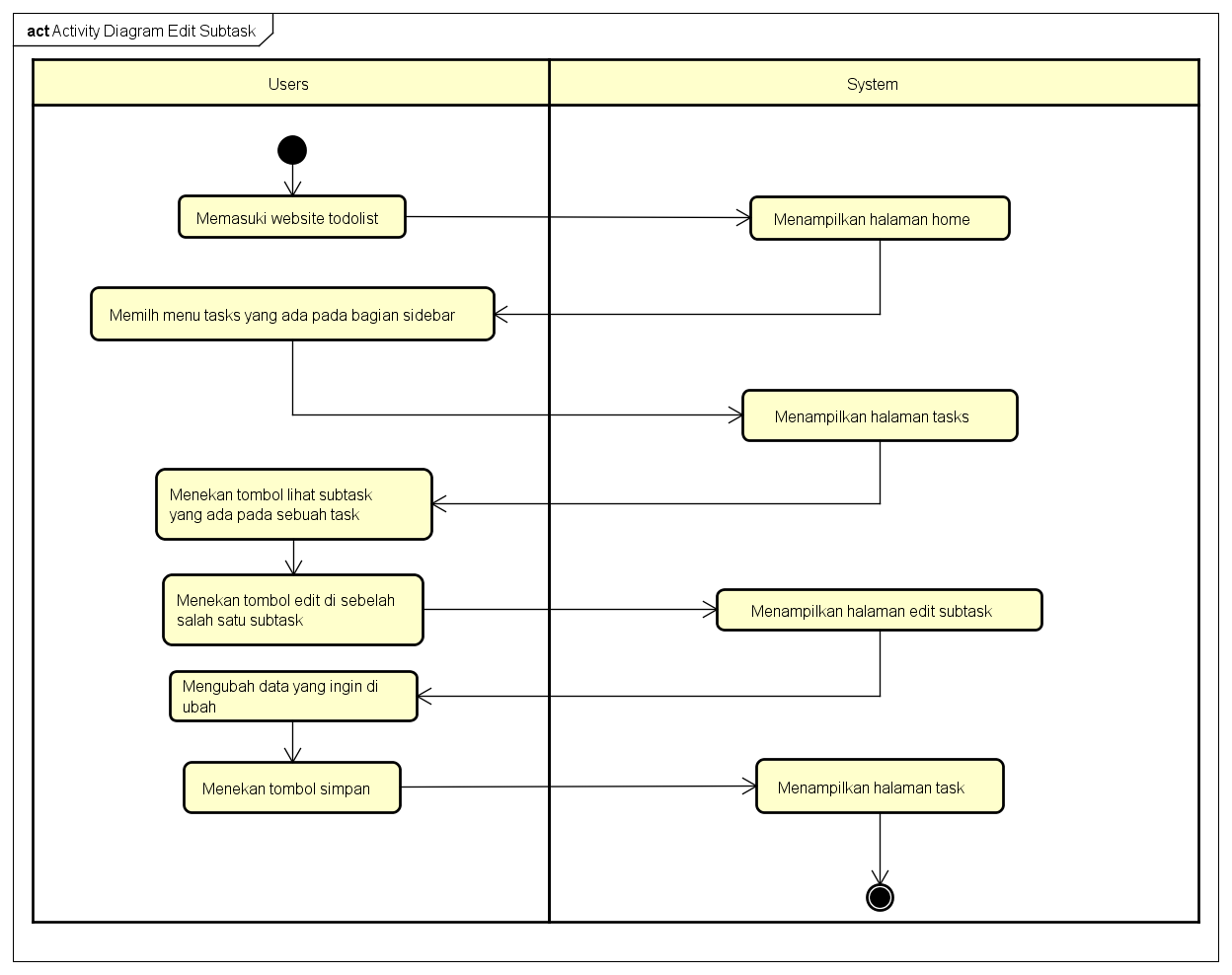
*Activity Diagram* Hapus *Task*



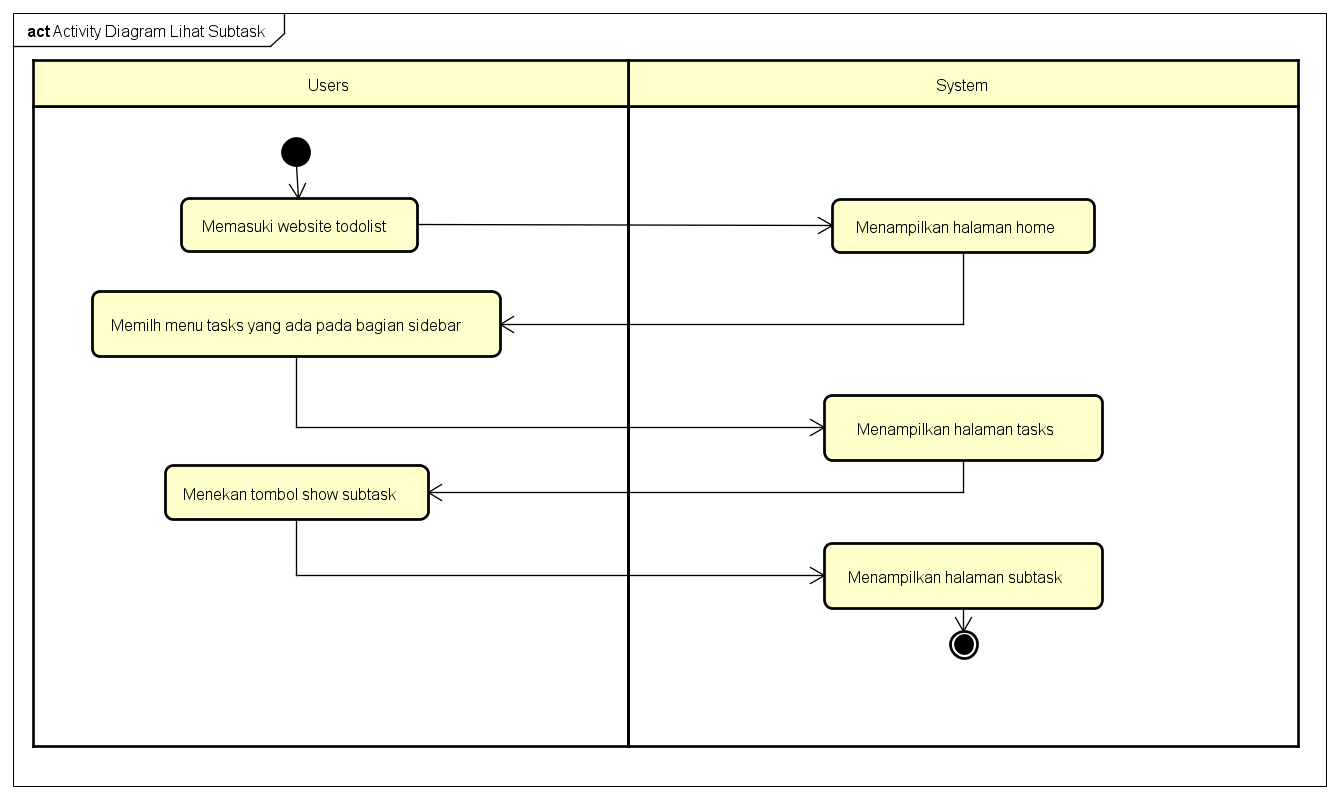
*Activity Diagram* Tambah *Subtask*



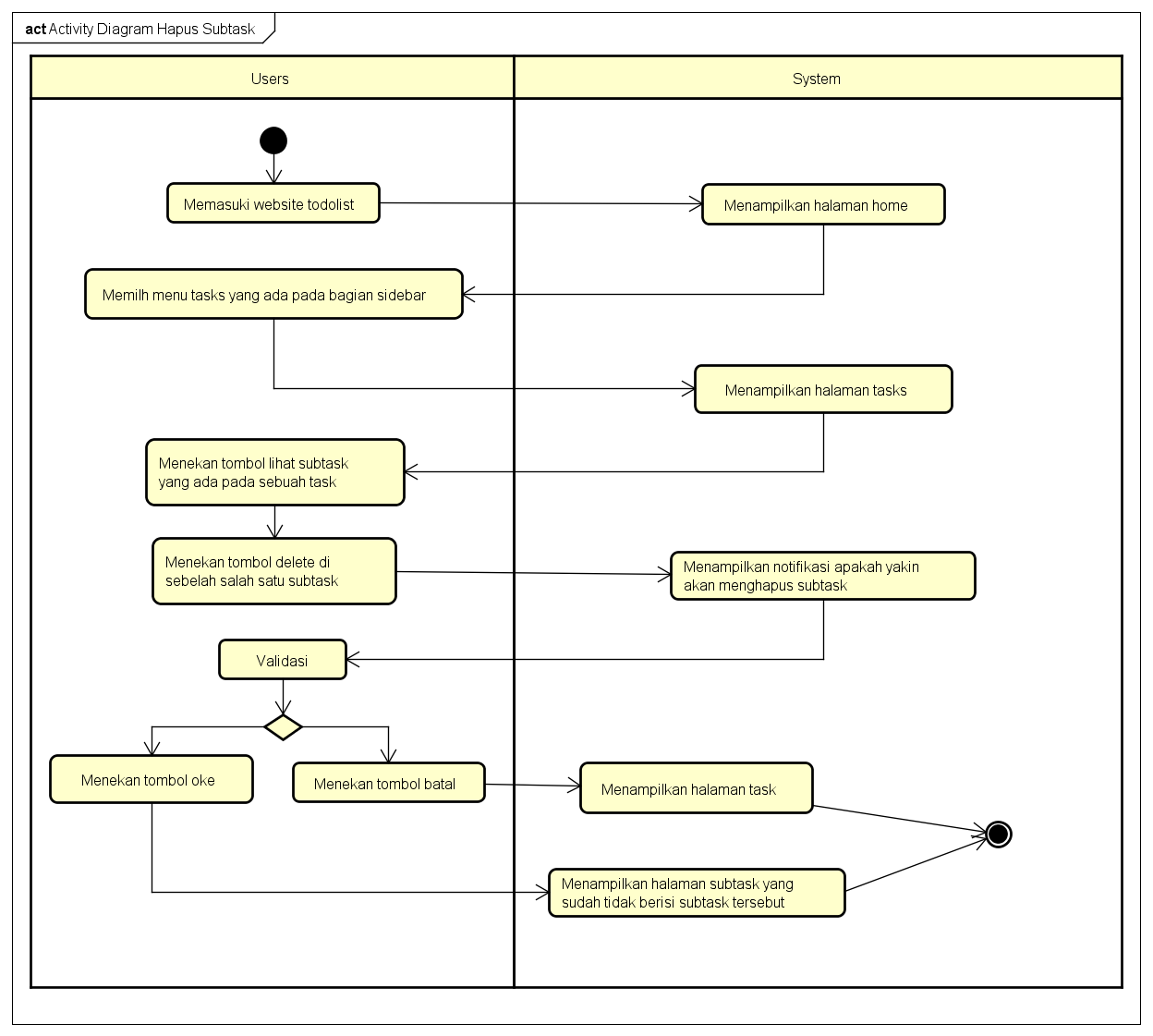
*Activity Diagram* Edit *Subtask*



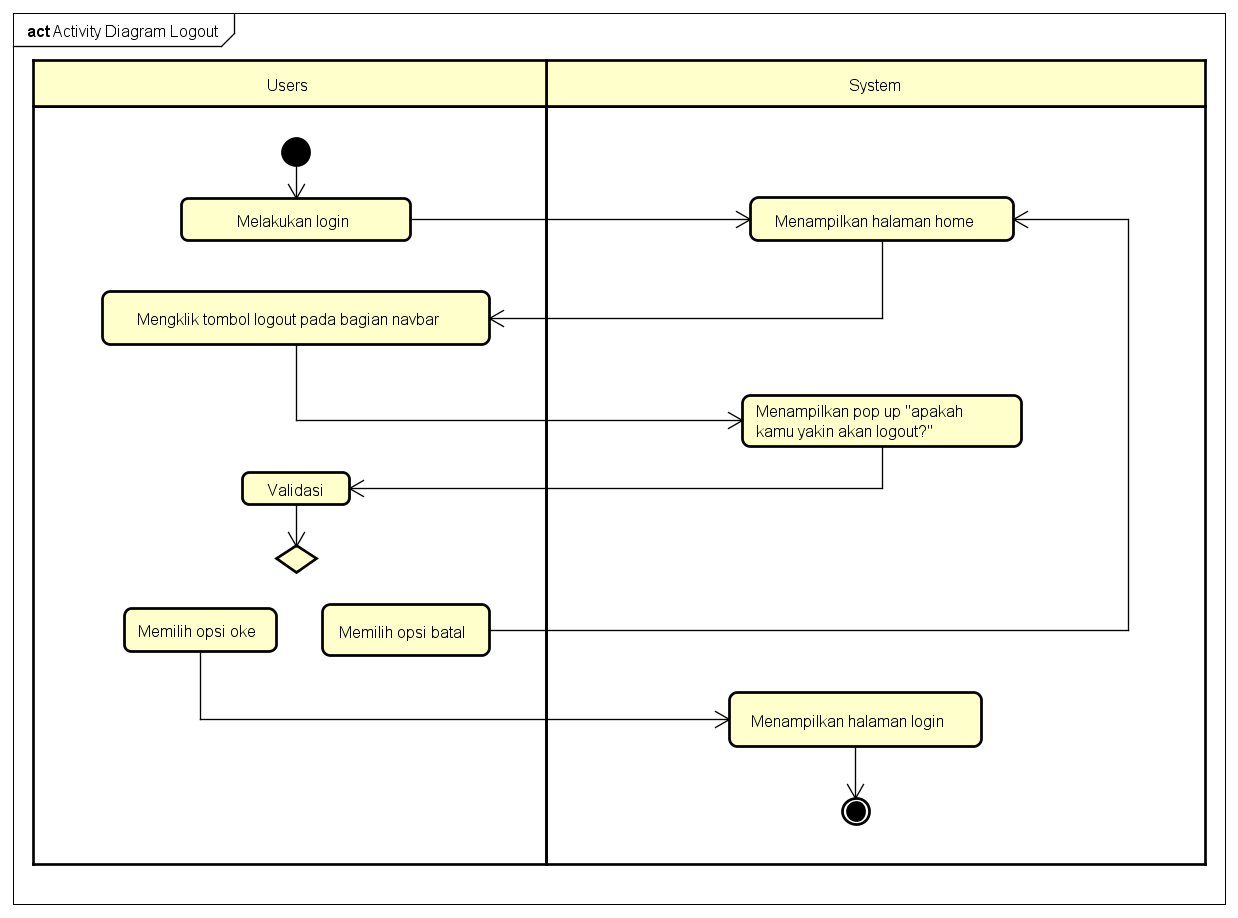
*Activity Diagram* Lihat Sub*task*



*Activity Diagram* Hapus *Subtask*

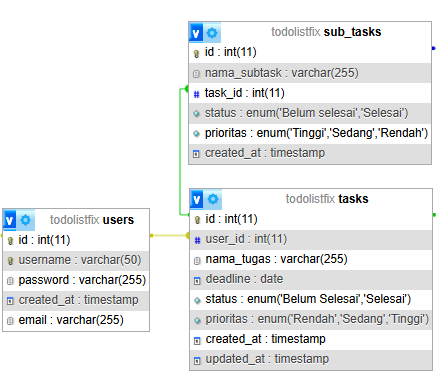


# *Activity Diagram Logout*

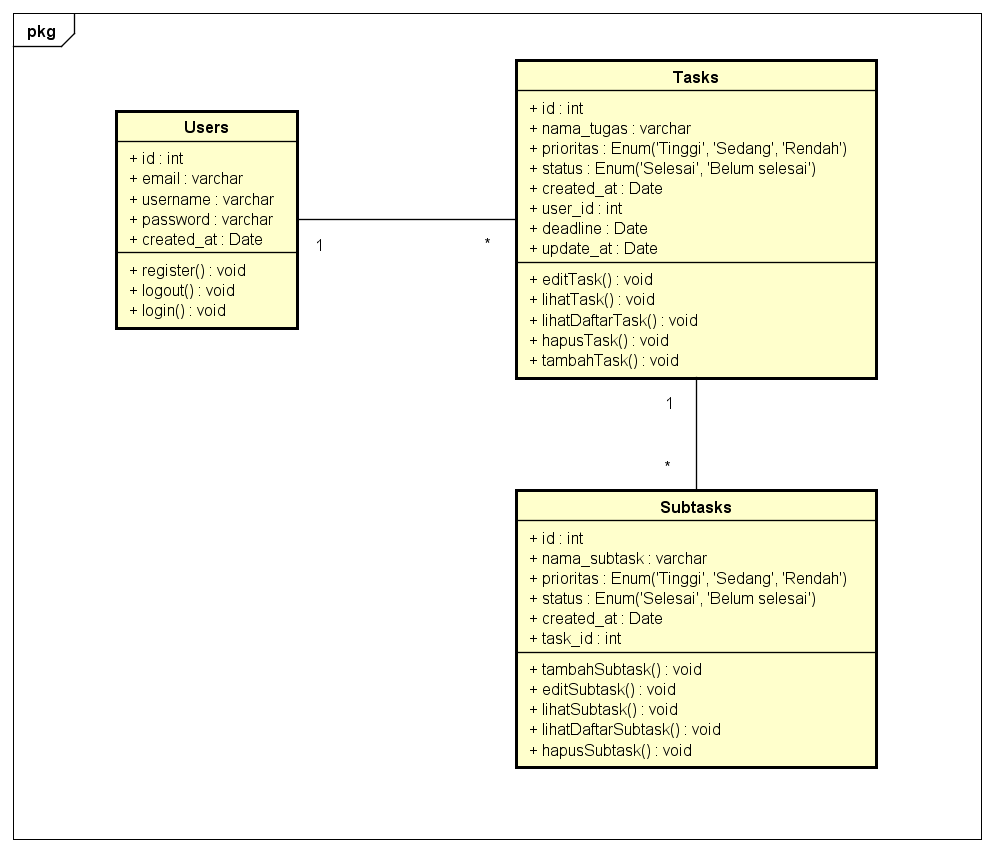


# 5. PERANCANGAN DESAIN BASIS DATA

***Desainer***



***Class Diagram***



# 6. DUMMY DATA

**Tabel User**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | Username | Password | created\_at | Email |
| 1 | Ardita | MD5 | 2025-02-05 | arditaa@gmail.com |
| 2 | Marcell | MD5 | 2025-02-05 | marcell@gmail.com |
| 3 | Farell | MD5 | 2025-02-05 | farell@gmail.com |
| 4 | Galih | MD5 | 2025-02-05 | galih@gmail.com |
| 5 | Dimas | MD5 | 2025-02-05 | dimas@gmail.com |
| 6 | Rizki | MD5 | 2025-02-05 | Rizki@gmail.com |
| 7 | Nurdin | MD5 | 2025-02-05 | nurdin@gmail.com |
| 8 | Radit | MD5 | 2025-02-05 | radit@gmail.com |
| 9 | Adit | MD5 | 2025-02-05 | adit@gmail.com |
| 10 | Alka | MD5 | 2025-02-05 | alka@gmail.com |

**Tabel Task**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| id | user\_id | nama\_tugas | deadline | status | prioritas | Created\_at | Update\_at |
| 1 | 1 | Belajar PHP | 2025-04-20 | Belum Selesai | Rendah | 2025-04-15  08:00:00 | 2025-04-15  08:05:00 |
| 2 | 1 | Belajar CSS | 2025-04-21 | Belum Selesai | Sedang | 2025-04-15  08:05:00 | 2025-04-14  08:10:00 |
| 3 | 2 | Belajar Bootstrap | 2025-04-22 | Belum Selesai | Tinggi | 2025-04-14  08:10:00 | 2025-04-14  08:15:00 |
| 4 | 2 | Belajar HTML | 2025-04-23 | Belum Selesai | Tinggi | 2025-04-14  08:15:00 | 2025-04-13  08:20:00 |
| 5 | 3 | Belajar JavaScript | 2025-04-24 | Belum Selesai | Sedang | 2025-04-13  08:20:00 | 2025-04-13  08:25:00 |
| 6 | 3 | Kerjakan PR | 2025-04-25 | Selesai | Rendah | 2025-04-13  08:25:00 | 2025-04-12  08:30:00 |
| 7 | 4 | Belanja | 2025-04-26 | Selesai | Sedang | 2025-04-12  08:30:00 | 2025-04-12  08:35:00 |
| 8 | 4 | Baca Buku | 2025-04-27 | Selesai | Tinggi | 2025-04-12  08:35:00 | 2025-04-11  08:40:00 |
| 9 | 5 | Belajar Bahasa Inggris | 2025-04-28 | Selesai | Tinggi | 2025-04-11  08:40:00 | 2025-04-11  08:45:00 |
| 10 | 5 | Olahraga Pagi | 2025-04-28 | Selesai | Rendah | 2025-04-11  08:45:00 | 2025-04-15  08:50:00 |

**Tabel Subtask**

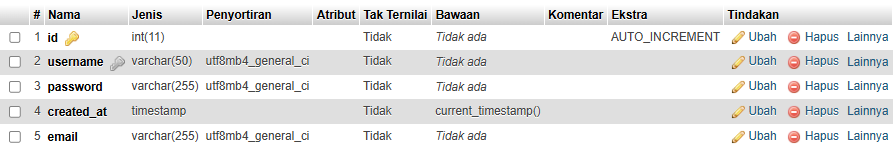
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | Nama\_subtask | task\_id | status | prioritas | created\_at |
| 1 | Download Materi 1 | 1 | Belum Selesai | Belum Selesai | 2025-04-15  08:06:00 |
| 2 | Baca Bab 1 | 2 | Belum Selesai | Belum Selesai | 2025-04-14  08:11:00 |
| 3 | Praktik materi 1 | 3 | Belum Selesai | Belum Selesai | 2025-04-14  08:16:00 |
| 4 | Cari Referensi | 4 | Belum Selesai | Belum Selesai | 2025-04-13  08:21:00 |
| 5 | Review kode | 5 | Belum Selesai | Belum Selesai | 2025-04-13  08:26:00 |
| 6 | Rapikan Tugas | 6 | Selesai | Selesai | 2025-04-12  08:31:00 |
| 7 | Belanja Kebutuhan dapur | 7 | Selesai | Selesai | 2025-04-12  08:36:00 |
| 8 | Baca buku sejarah | 8 | Selesai | Selesai | 2025-04-11  08:41:00 |
| 9 | Download materi 4 | 9 | Selesai | Selesai | 2025-04-11  08:46:00 |
| 10 | Menonton video contoh peregangan | 10 | 2025-04-28 | Selesai | 2025-04-15  08:51:00 |

**Tabel History**

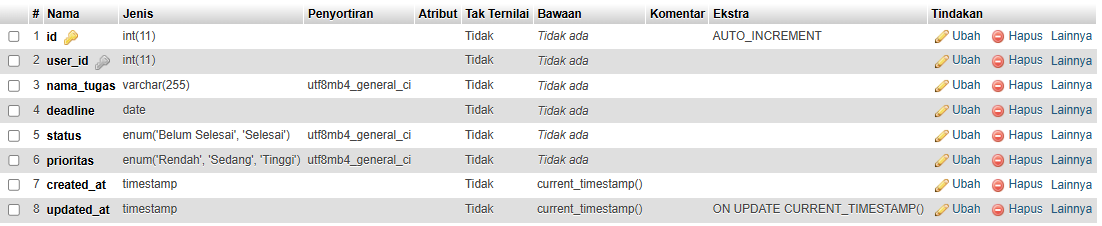
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| id | user\_id | Task\_id | Action\_type | Action\_details | Created\_at | Update\_at |
| 1 | 1 | 1 | Create | Membuat rangkuman materi 1 | 2025-04-15  08:00:00 | 2025-04-15  08:06:00 |
| 2 | 1 | 2 | create | Membuat rangkuman bab 1 | 2025-04-15  08:05:00 | 2025-04-14  08:11:00 |
| 3 | 2 | 3 | complete | Menandai tugas selesai | 2025-04-14  08:10:00 | 2025-04-14  08:16:00 |
| 4 | 2 | 4 | complete | Menandai tugas selesai | 2025-04-14  08:15:00 | 2025-04-13  08:21:00 |
| 5 | 3 | 5 | create | Membuat review kode yang telah dipelajari | 2025-04-13  08:20:00 | 2025-04-13  08:26:00 |
| 6 | 3 | 6 | delete | Menghapus tugas yang tidak diperlukan | 2025-04-13  08:25:00 | 2025-04-12  08:31:00 |
| 7 | 4 | 7 | create | Membuat daftar atau list belanja | 2025-04-12  08:30:00 | 2025-04-12  08:36:00 |
| 8 | 4 | 8 | complete | Menandai tugas selesai | 2025-04-12  08:35:00 | 2025-04-11  08:41:00 |
| 9 | 5 | 9 | Complete | Menandai tugas selesai | 2025-04-11  08:40:00 | 2025-04-11  08:46:00 |
| 10 | 5 | 10 | complete | Menandai tugas selesai | 2025-04-11  08:45:00 | 2025-04-15  08:51:00 |

**7. PEMBUATAN BASIS DATA (MySQL Code)**

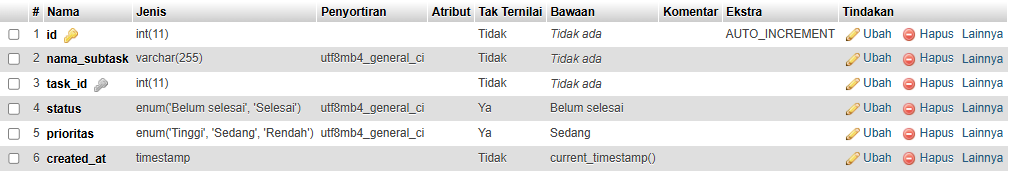
1. ***User***

****

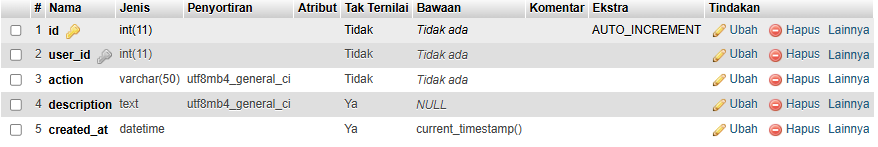
1. ***Task***

****

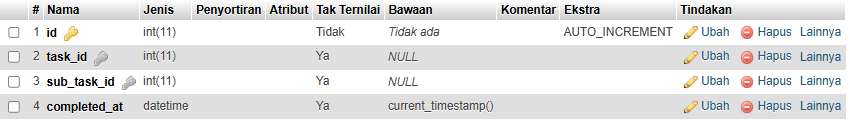
1. ***Subtask***

****

1. ***History***

****

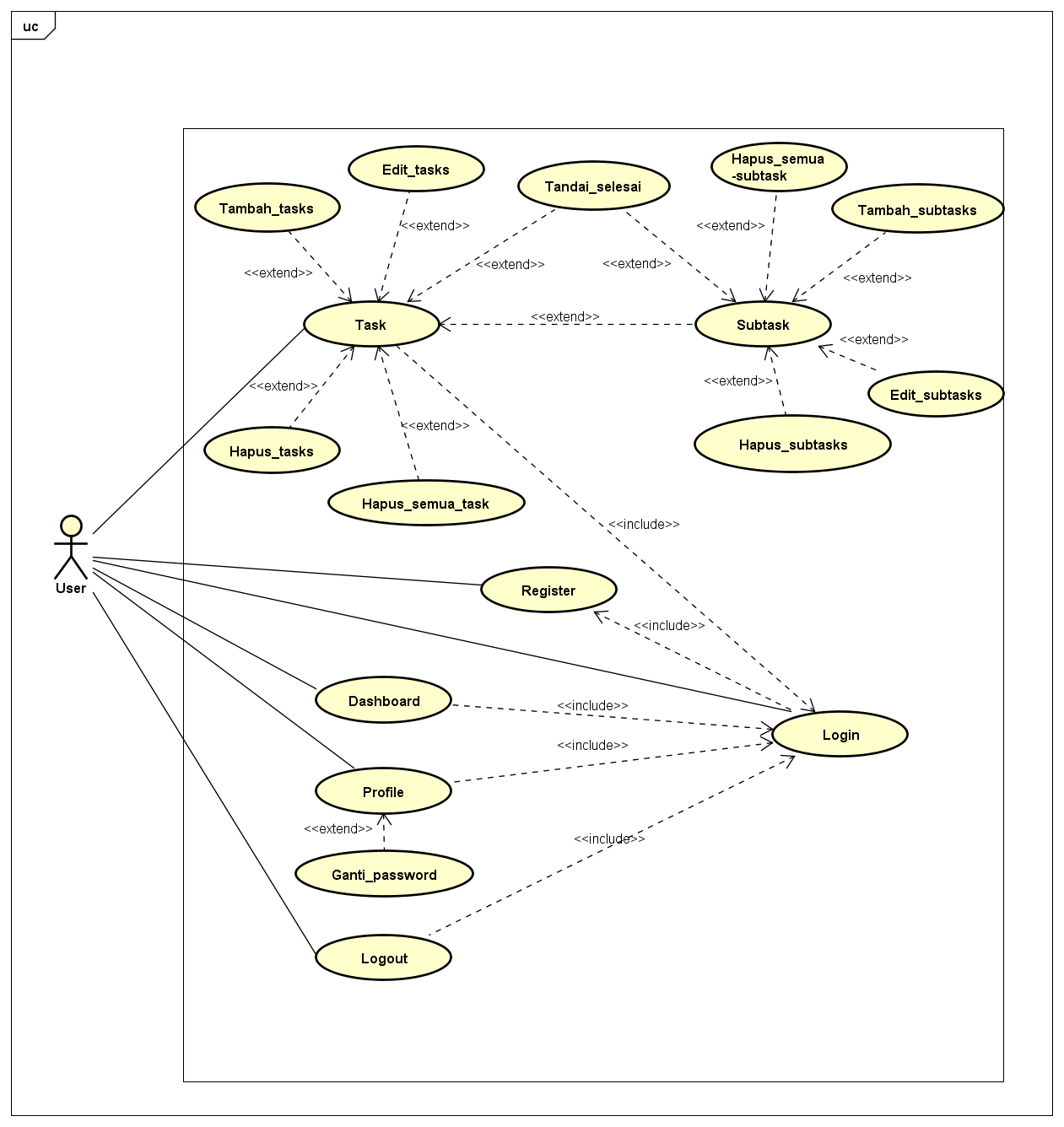
1. ***Completed Task***

****

**8. PERANCANGAN TAMPILAN ANTARMUKA (UI) *WEBSITE***

**Aplikasi Web *To-Do List***

1. ***Use Case Diagram***

****

1. ***Use Case Scenario***
2. ***Register***

Aktor : Pengguna

Deskripsi : Pengguna mendaftar untuk membuat akun baru

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Memilih opsi *register* | Menampikan *form* *register* |
| Mengisi *username*, *email*, *password*, dan *confirm password* | Melakukan verifikasi apakah penulisan *email* sudah valid dan sudah terdaftar atau belum, serta memeriksa kesesuaian antara *password* dan *confirm password* |
|  | Jika *email* belum terdatar serta *password* dan *confirm password* sudah sesuai maka akun akan berhasil dibuat |
| Melakukan proses *login* |  |
|  |  |

1. ***Login***

Aktor : Pengguna

Deskripsi : Pengguna masuk ke akun dengan menggunakan *email* dan *password*

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Membuka situs web | Menampikan *form* *Login* |
| Mengisi *email* dan *password* | Melakukan verifikasi *email* dan *password* sudah terdaftar atau belum |
|  | Jika valid, akan menampilkan halaman *home* |

1. ***Logout***

Aktor : Pengguna

Deskripsi : Pengguna keluar dari akun *website*

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Menekan button *logout* pada bagian navbar website | Menampikan notifikasi konfirmasi *logout* |
| Memilih opsi oke | Menampilkan halaman *login* |

1. **Tambah *Task***

Aktor : Pengguna

Deskripsi : Pengguna dapat menambahkan *task* utama ke dalam daftar *to-do list*

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Melakukan *login* ke sistem |  |
| Memasuki halaman *home* *to-do list* |  |
| Mengklik tombol tambah *task* |  |
|  | Menampilkan *form* tambah *task* |
| Mengisi data seperti nama *task*, prioritas, dan *deadline* |  |
| Mengklik tombol simpan *task* |  |
|  | Menyimpan data ke tabel *task* dan menampilkan *task* baru di *card* |

1. **Edit *Task***

Aktor : Pengguna

Deskripsi : Pengguna dapat mengubah nama, tanggal, atau prioritas *task*.

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Mengklik opsi edit *task* |  |
|  | Menampilkan *form* edit *task* |
| Melakukan perubahan data pada *task* |  |
| Mengklik tombol simpan | Memperbaharui data di *database* dan menampilkan hasil perubahan data yang baru pada halaman |

1. **Hapus** ***Task***

Aktor : Pengguna

Deskripsi :Pengguna dapat menghapus *task* dan memindahkannya ke arsip terlebih dahulu.

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Mengklik opsi hapus *task* |  |
|  | Menampilkan *pop up* “apakah yakin akan menghapus *task*?” |
| Mengklik opsi “oke” | Memperbaharui data di *database* dan menampilkan hasil perubahan data yang baru pada halaman |

1. **Tambah *Subtask***

Aktor : Pengguna

Deskripsi : Pengguna dapat menambahkan sub-*task* ke dalam *task* tertentu.

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Pengguna klik tombol "Lihat Sub*task*" pada *card* *task*. | Sistem menampilkan daftar sub-*task* dan *form* tambah sub-*task*. |
| Pengguna mengisi nama sub-*task* dan klik "Tambah" | Sistem menyimpan sub-*task* ke tabel sub\_*task* dan menampilkannya. |

1. **Edit *Subtask***

Aktor : Pengguna

Deskripsi : Pengguna dapat memperbarui nama atau status *sub*-*task*.

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Mengklik opsi edit pada sebuah sub*task* | Menampilkan halaman edit sub*task* |
| Melakukan perubahan data pada sub*task* |  |
| Mengklik tombol simpan | Sistem menyimpan perubahan ke *database* dan memperbarui tampilan. |

1. **Hapus *Subtask***

Aktor : Pengguna

Deskripsi : Pengguna dapat menghapus *sub*-*task* dari dalam *task*

|  |  |
| --- | --- |
| Pengguna | Sistem |
| Mengklik opsi hapus pada sebuah sub*task* | Menampilkan *pop up* “yakin ingin menghapus?” |
| Mengklik opsi oke | Sistem menyimpan perubahan ke *database* dan memperbarui tampilan. |

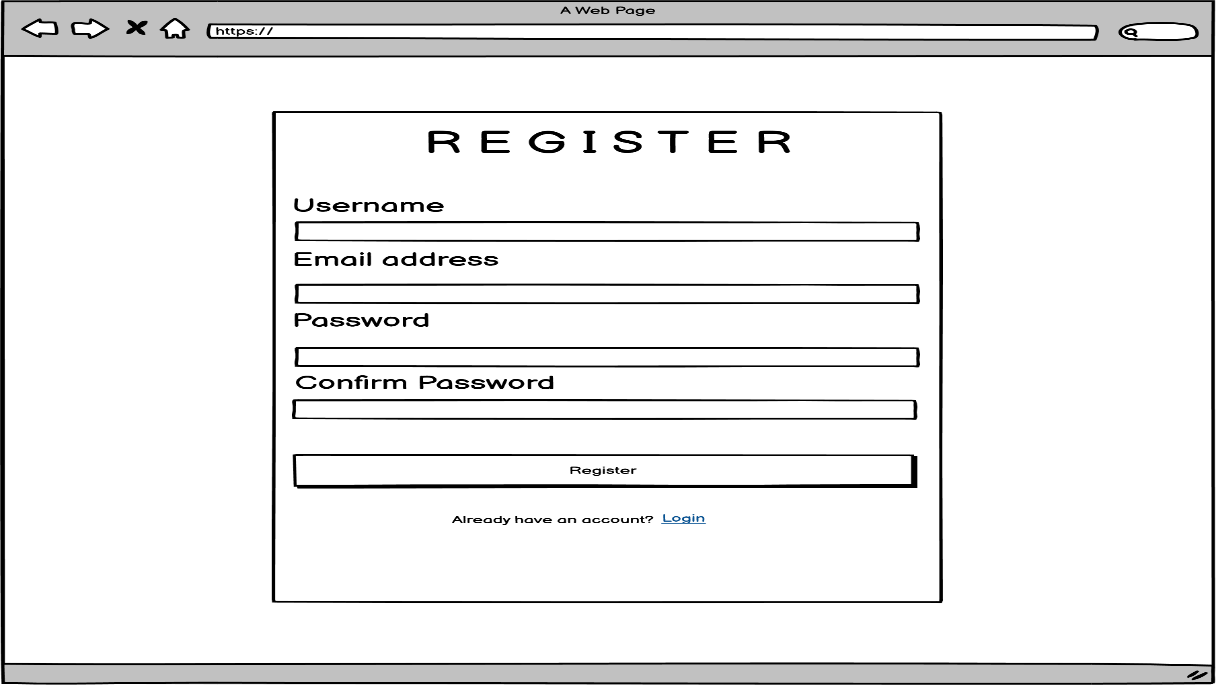
1. **Sketsa atau *Mockup* Aplikasi Website *To-Do List***

**Berikut ini sketsa atau *mockup* yang telah dibuat penyusun dalam perancangan aplikasi**

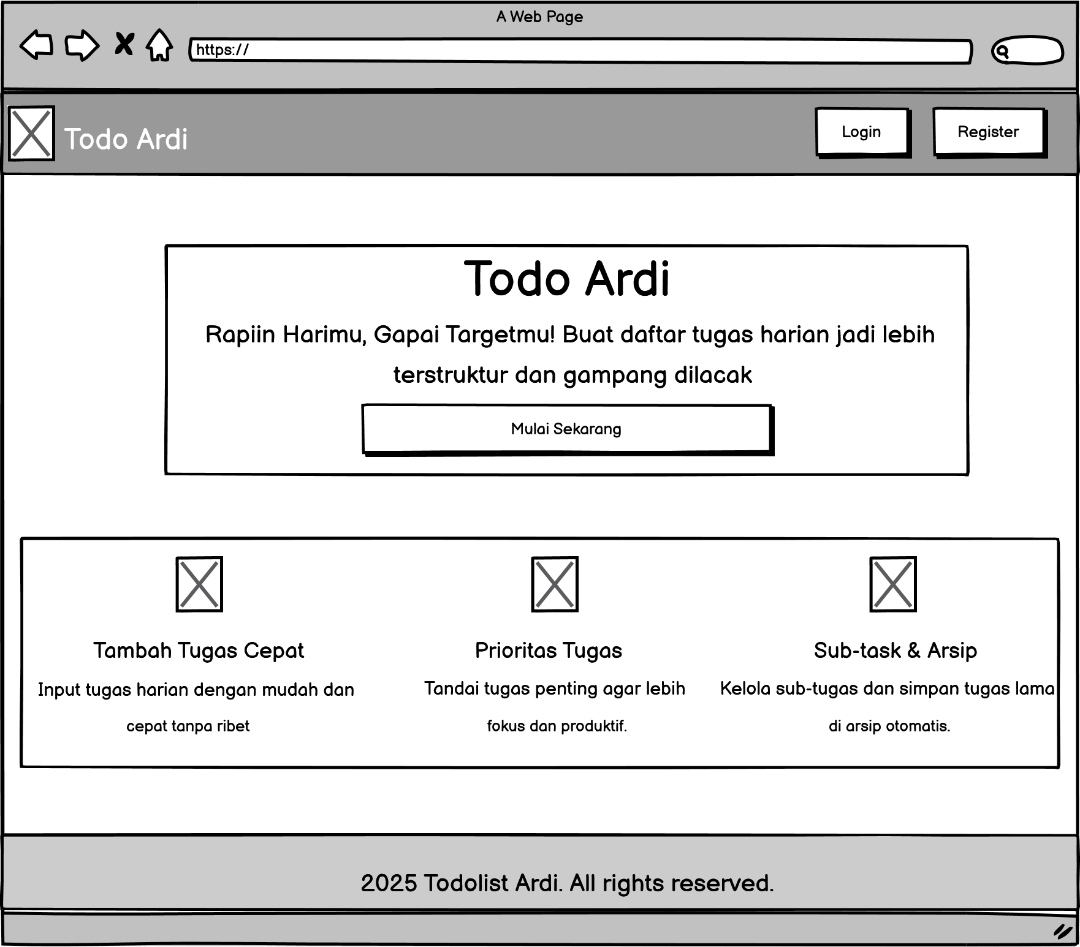
1. ***Login***

****

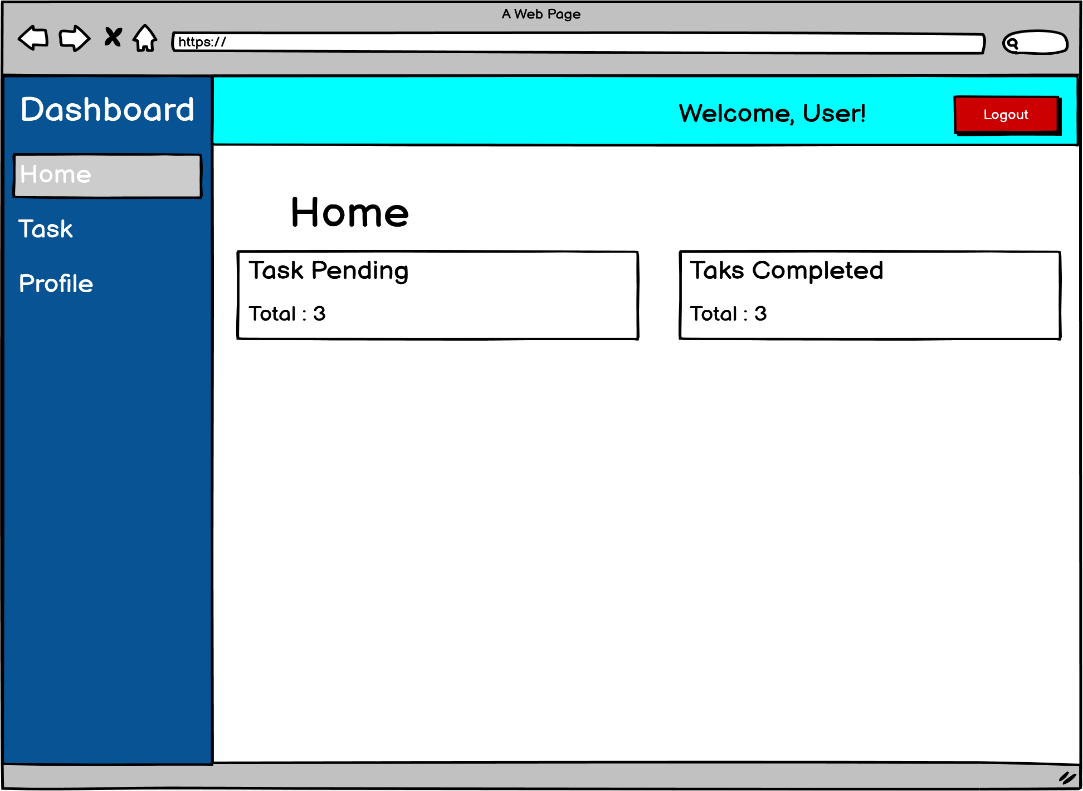
1. ***Register***

****

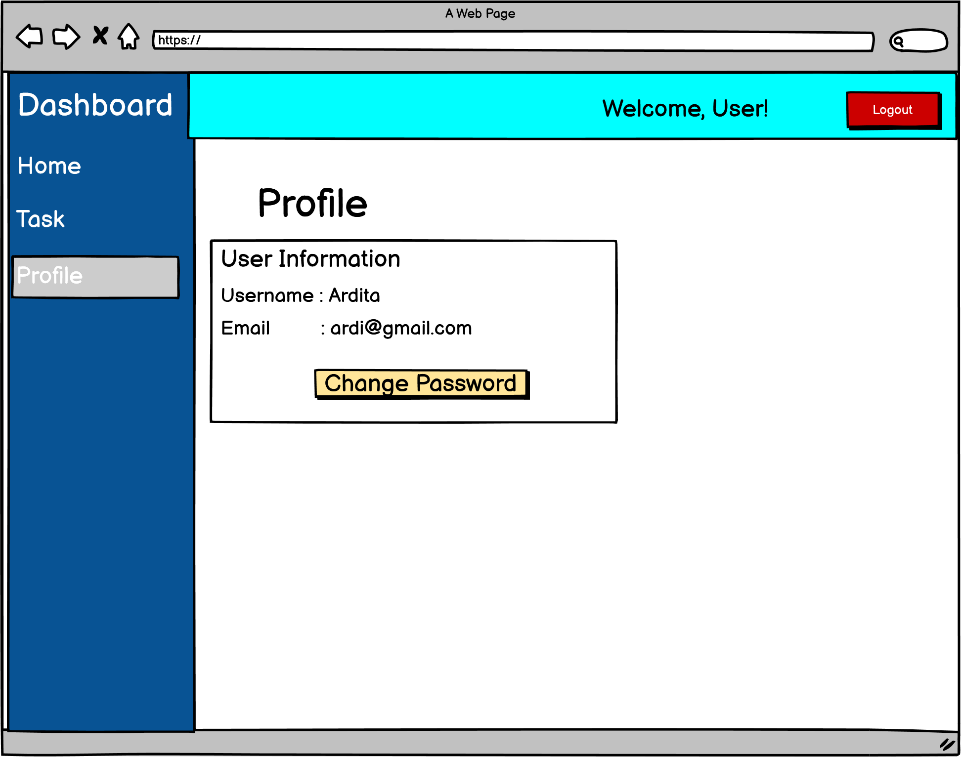
1. ***Landing Page***

****

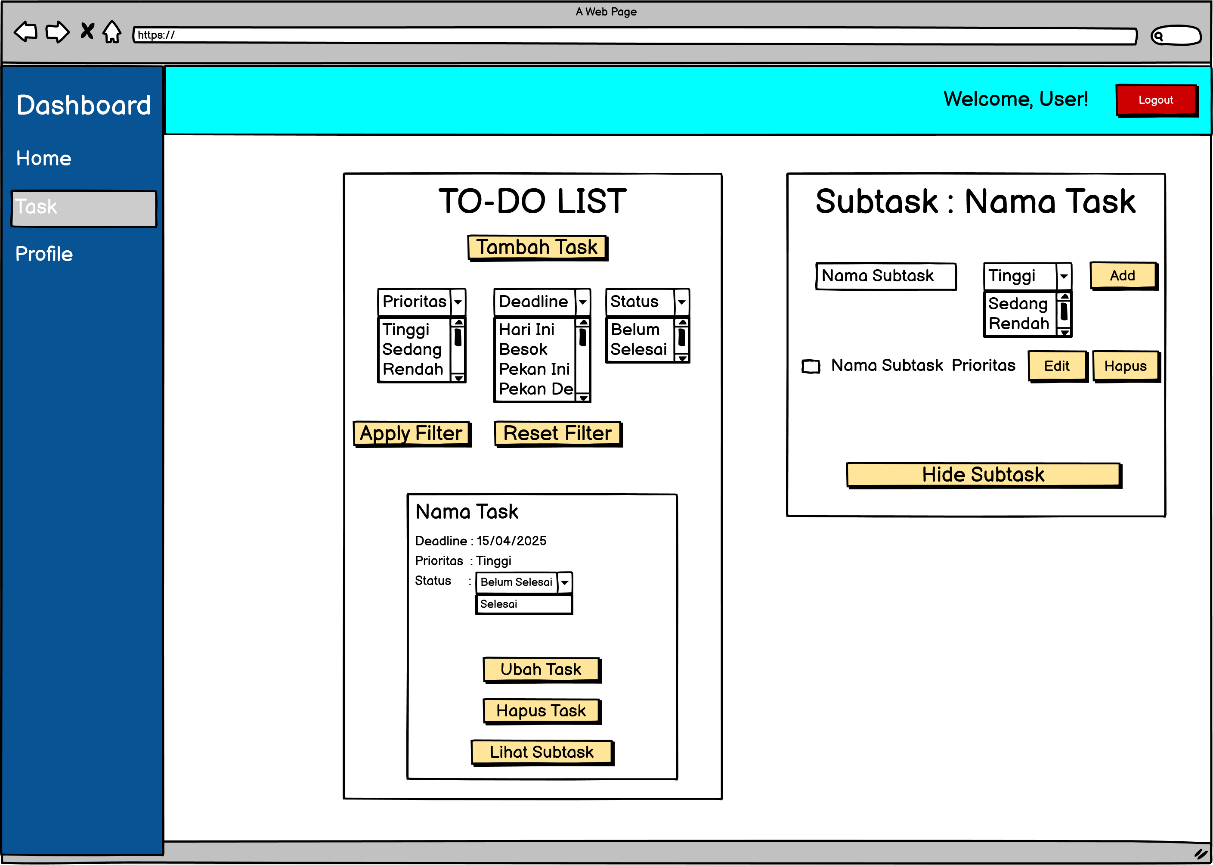
1. ***Home***

****

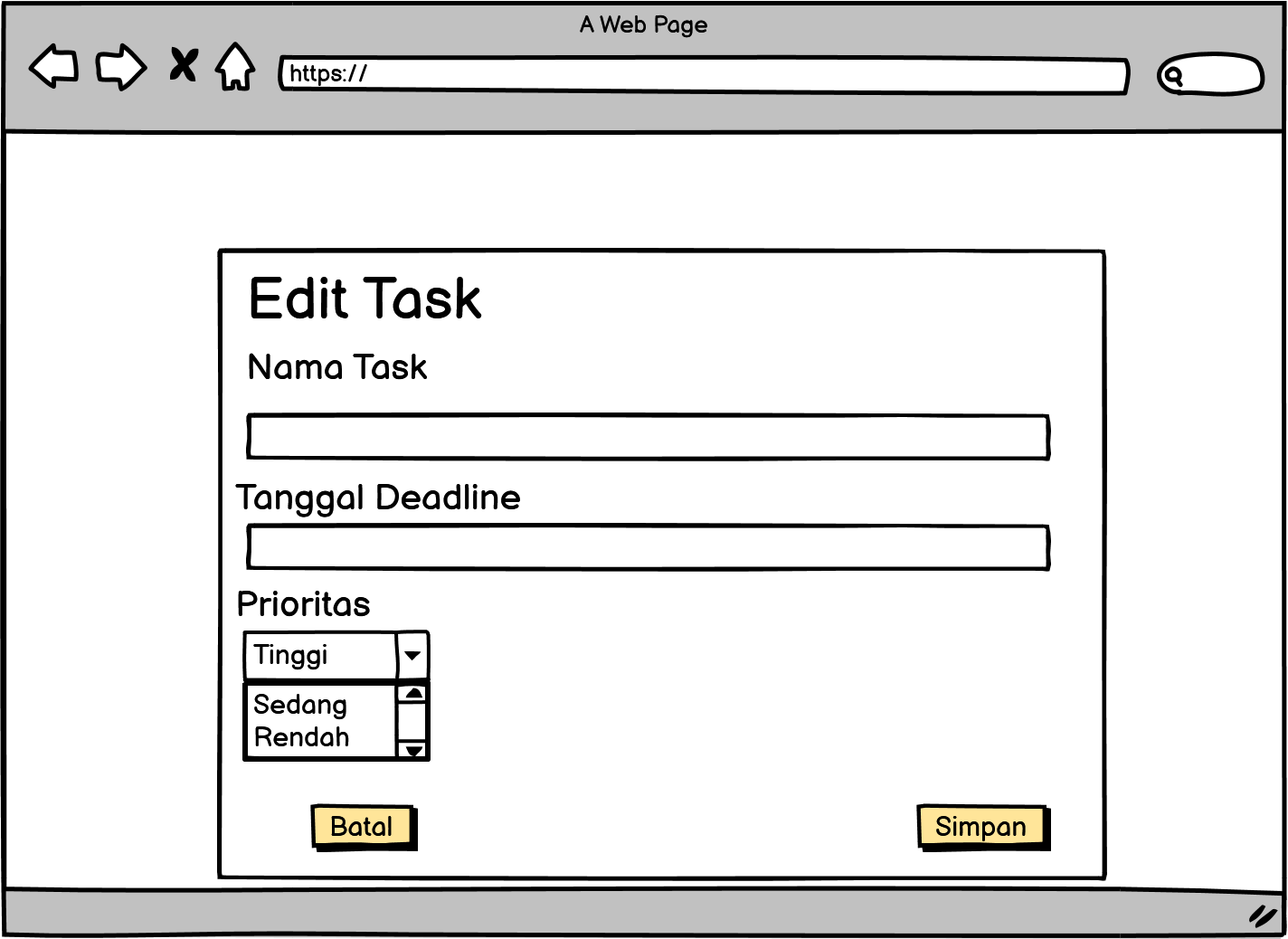
1. ***Profile***



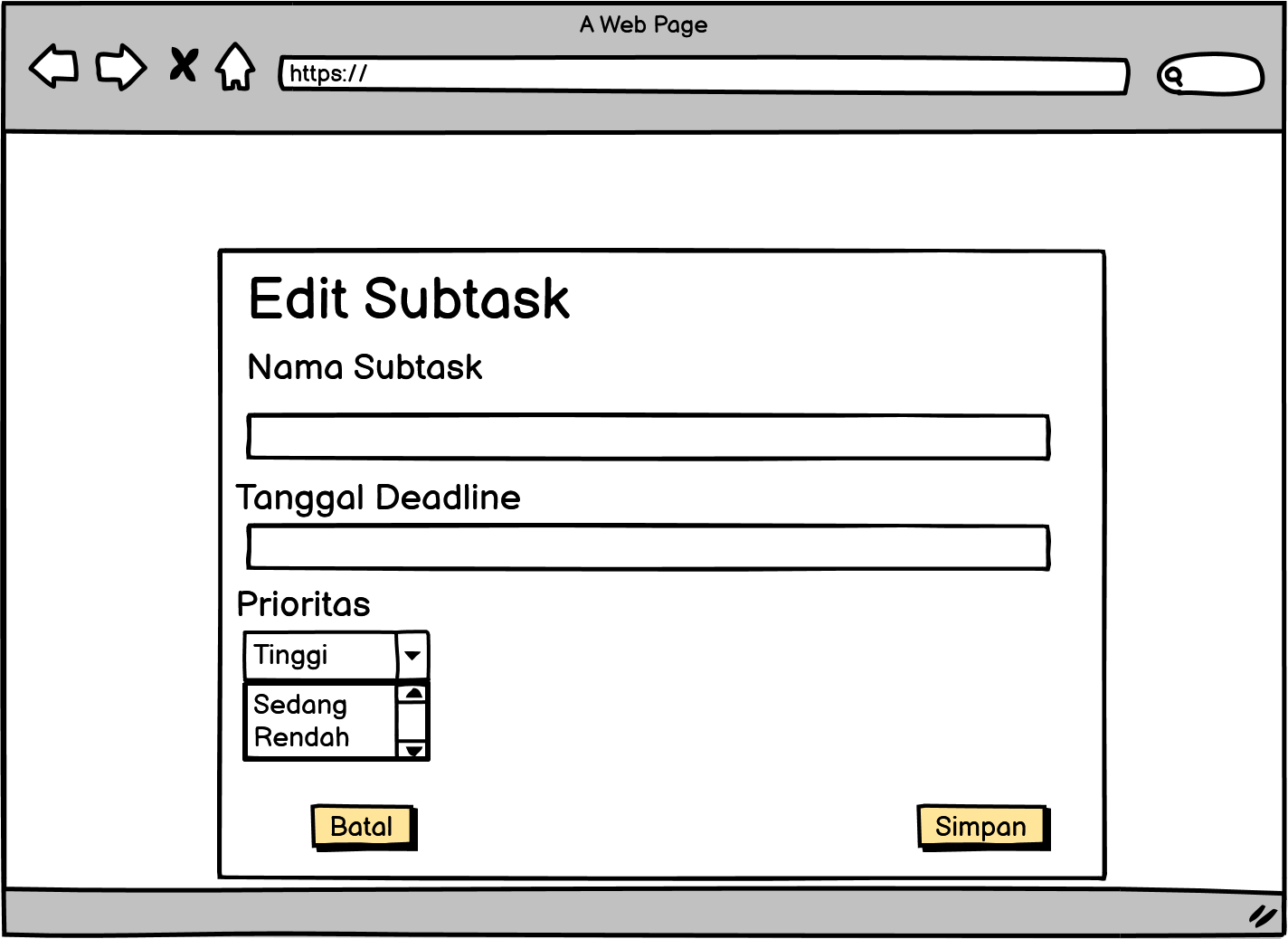
1. ***Task***

****

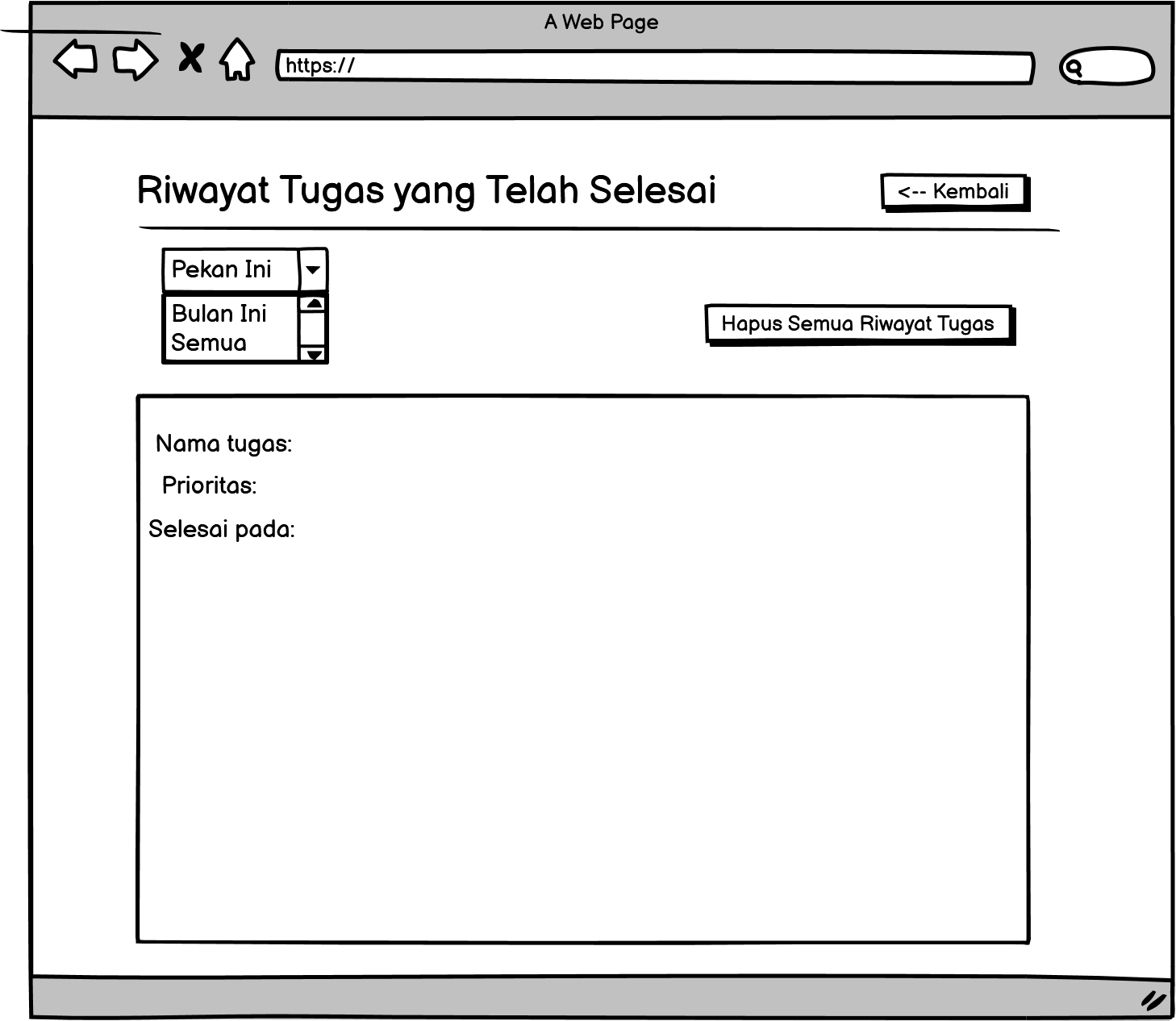
1. **Edit *Task***

****

1. **Edit *Subtask***

****

1. **Lihat History**

****

**9. UJI COBA (BLACK BOX TESTING)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Module Name** | **Pre-Condition** | **Test Case** | **Test Skenario** | **Expected Result** | **Actual Result** |
| 1 | Login | User telah melakukan proses registrasi akun | *1.Field Email* dan *password* tidak dimasukkan kemudian menekan tombol “Login” | 1. Membuka website dengan menggunakan domain : **<http://arditodo.my.id/>**  Menampilkan beranda awal aplikasi  Klik menu “Login”.  2. Menampilkan form login  3. Mengosongkan *field* *email* dan *password*  4. Klik tombol “Login” | menampilkan pesan  “Harap isi bidang ini” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\login1.PNG=** |
|  |  |  | *2.Field* *email* dan *password* yang dimasukkan salah kemudian menekan tombol “login” | 1. Membuka website dengan menggunakan domain : **<http://arditodo.my.id/>**  Menampilkan beranda awal aplikasi  Klik menu “Login”.  2. Menampilkan form login  3.Mengisi data yang tidak sesuai dengan yang ada di database  4.Klik tombol “Login” | menampilkan pesan  “Login gagal. Akun tidak ditemukan.” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\login2.PNG** |
|  |  |  | *3.Field* *email* dan *password* yang di masukkan benar, menekan tombol “Login” | 1. Membuka website dengan menggunakan domain : **<http://arditodo.my.id/>**  Menampilkan beranda awal aplikasi  Klik menu “Login”.  2. Menampilkan form login  3.Mengisi form dengan data yang benar | Menampilkan halaman home | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\login3.PNG** |
| 2 | Register | User belum melakukan registrasi akun | *1.Field* *Username*, *email*, *password* dan *confirm password* tidak diisi kemudian Menekan tombol “Register” | 1.memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengosongkan *field* *Username*, *email*, *password* dan *confirm password*  4. Menekan tombol “Register” | Menampilkan pesan “Harap isi bidang ini” |  |
|  |  |  | *2.Field* *email*. *Password* , dan *Confirm password* diisi, dan mengosongkan *username*, kemudian klik tombol Register | 1. memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengosongkan *field* *Username*, dan mengisi *email*, *password* dan *confirm password*  4. Menekan tombol “Register” | Sistem akan menolak dan menampilkan pesan  “Harap isi bidang ini” |  |
|  |  |  | *3.Field* *username*, *password*, dan *confirm password* diisi, dan mengosongkan *email*, kemudian klik tombol “Register” | 1. memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengosongkan *field* *email*, dan mengisi *username*, *password*, dan *confirm password*  4. Menekan tombol “Register” | Sistem akan menolak dan menampilkan pesan  “Harap isi bidang ini” |  |
|  |  |  | *4.Field* *username*, *email*(tidak valid), *password*, dan *confirm password* diisi, kemudian klik tombol “Register” | 1.memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengisi *field* *username*, *email* (tidak valid) , *password*, dan *confirm password*  4. Menekan tombol “Register” | Menampilkan pesan “Sertakan ‘@’ pada alamat *email*. ‘  Ardigmail.com’ tidak memiliki ‘@’.” |  |
|  |  |  | *5.Field* *username*, *email* dan *confirm password* diisi, dan mengosongkan *password* , kemudian klik tombol “Register” | 1.memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengisi *field* *username*, *email*, dan confirm, dan mengosongkan *password*  4. Menekan tombol “Register” | Menampilkan pesan “Harap isi bidang ini” |  |
|  |  |  | *6.Field* *username*, *email* dan *password* diisi, dan mengosongkan *confirm password* , kemudian klik tombol “Register” | 1.memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengisi *field* *username*, *email*, dan *password*, dan mengosongkan *confirm password*  4. Menekan tombol “Register” | Menampilkan pesan “Harap isi bidang ini” |  |
|  |  |  | *7.Field* *username*, *email*, *password* dan *confirm password*(tidak sesuai diisi),  kemudian klik tombol Register | 1.memilih opsi “Register” pada form login.  2. Menampilkan halaman register  3. Mengisi *field* *username*, *email*, dan *password*, dan *confirm password*(tidak sesuai)  4. Menekan tombol “Register” | menampilkan pesan “*Password* dan konfirmasi *password* tidak cocok” |  |
| 3 | Tambah Task | User sudah terdaftar dan masuk ke websiter | 1.Field nama task, deadline, dan prioritas tidak di isi | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Tambah Task”.  3. Menampilkan form tambah task  4. Mengosongkan nama dan deadline  5. Menekan tombol “Simpan Task” | Menampilkan pesan “Harap isi bidang ini” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian tambah task 1.PNG** |
|  |  |  | 2.Field deadline dan prioritas diisi dan field nama dikosongkan | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Tambah Task”.  3. Menampilkan form tambah task  4. Mengisi deadline dan prioritas  5. Menekan tombol “Simpan Task” | Menampilkan pesan “Harap isi di bidang ini.” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian tambah task 2.PNG** |
|  |  |  | 3.Field nama dan deadline diisi dan field prioritas dikosongkan | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Tambah Task”.  3. Menampilkan form tambah task  4. Mengisi nama dan deadline  5. Menekan tombol “Simpan Task” | Menampilkan pesan “Harap isi di bidang ini.” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian tambah task 3.PNG** |
|  |  |  | 4.Field nama dan prioritas diisi dan deadline dikosongkan | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Tambah Task”.  3. Menampilkan form tambah task  4. Mengisi nama dan prioritas | Menampilkan pesan “Harap isi di bidang ini.” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian tambah task 4.PNG** |
|  |  |  | 5.Field nama, deadline dan prioritas di isi | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Tambah Task”.  3. Menampilkan form tambah task  4. Mengisi nama, deadline dan prioritas  5. Menekan tombol “Simpan Task” | Menampilkan halaman task yang sudah berisi task yang baru dibuat | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian tambah task 5.2.PNGD:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian tambah task 5.1.PNG** |
| 4 | Edit Task | User sudah terdaftar dan masuk ke aplikasi serta sudah menambahkan task | 1.Field nama tugas, dan deadline,tidak di isi | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Edit Task” yang ada pada sebuah task.  3.Menampilkan form edit task  4.Mengosongkan nama tugas, dan deadline  5. Menekan tombol “Simpan” | Menampilkan pesan “Harap isi bidang ini” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian edit task 1.PNG** |
|  |  |  | 2.Field deadline dan prioritas diisi dan field nama dikosongkan | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Edit Task” yang ada pada sebuah task.  3.Menampilkan form edit task  4.Mengisi deadline dan prioritas  5. Menekan tombol “Simpan” |  | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian edit task 2.PNG** |
|  |  |  | 3.Field nama dan prioritas diisi dan deadline dikosongkan | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Tambah Task”.  3. Menampilkan form tambah task  4. Mengisi nama dan prioritas | Menampilkan pesan “Harap isi di bidang ini.” | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian edit task 3.PNG** |
|  |  |  | 4.Field nama, deadline dan prioritas di isi | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Edit Task” yang ada pada sebuah task.  3.Menampilkan form edit task  4.Mengosongkan nama, deadline dan prioritas  5. Menekan tombol “Simpan” |  | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian edit task 4.PNG**  D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian edit task 4.1.PNG |
| 5. | Hapus Task | User sudah terdaftar dan masuk ke aplikasi serta sudah menambahkan task | 1.Memilih untuk tetap menghapus task | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Hapus Task” yang ada pada sebuah task.  3.Menampilkan notifikasi konfirmasi hapus task  4.Memilih opsi “Oke”  5. Menampilkan halaman task yang sudah tidak berisi task tersebut | Menampilkan halaman task yang sudah tidak berisi task tersebut | D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian hapus task 3.PNGD:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian hapus task 2.PNG**D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian hapus task 1.PNG** |
|  |  |  | 2.Memilih untuk batal menghapus task | 1.Klik menu “Task”  2. Menampilkan halaman task, klik tombol “Hapus Task” yang ada pada sebuah task.  3.Menampilkan notifikasi konfirmasi hapus task  4.Memilih opsi “Batal”  5. Menampilkan halaman task yang masih berisi task tersebut | Menampilkan halaman task yang masih berisi task tersebut | **D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian hapus task 2.2.PNG**  D:\Ardita 12 RPL 2 Semester 2\ArditaUjikom\pengujian hapus task 2.3.PNG |

**10. REVISI**

**11. PENENTUAN DOMAIN DAN HOSTING**

[**https://www.rumahweb.com/**](https://www.rumahweb.com/)

**12. UPLOAD WEB KE HOSTING**

**<http://arditodo.my.id/>**

**13. UPLOAD WEB KE GITHUB**

[**https://github.com/ArdiHalal/todolist**](https://github.com/ArdiHalal/todolist)

**KESIMPULAN**

Pembuatan aplikasi Todolist ini bertujuan untuk membantu pengguna dalam mengatur dan mengelola daftar tugas secara efisien. Aplikasi ini dirancang dengan fitur-fitur utama seperti sistem *login* dan registrasi, CRUD (Create, Read, Update, Delete) untuk tugas dan sub-tugas, sistem arsip sebelum penghapusan permanen, serta tampilan antarmuka yang rapi dan dominan menggunakan Bootstrap dengan nuansa dark mode yang tidak suram. Penggunaan struktur data yang terorganisir dengan lima tabel utama termasuk *task*s dan sub*task*s memungkinkan pengelolaan tugas menjadi lebih fleksibel dan sistematis. Dengan dukungan desain yang soft, elegan, dan profesional, aplikasi ini diharapkan dapat menjadi alat yang efektif dalam meningkatkan produktivitas dan manajemen waktu pengguna.