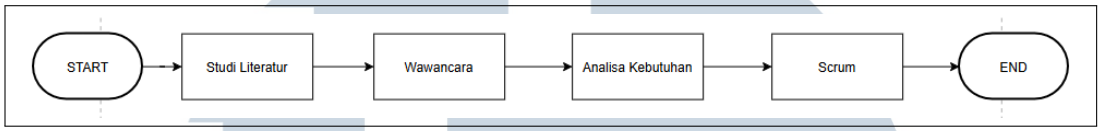
BAB 3
METODOLOGI PENELITIAN



Gambar 3.1. *Alur Metodologi Penelitian*

Gambar 3.1 menunjukkan gambaran umum alur metodologi penelitian. Metodologi penelitian dimulai dengan studi literatur, wawancara, analisa kebutuhan, dan implementasi *scrum*.

3.1 Studi Literatur

Studi literatur dilakukan dengan wawancara, kemudian mengumpulkan, menganalisis, dan mengevaluasi sumber-sumber tertulis seperti buku, jurnal, dan artikel yang relevan dengan topik penelitian. Tujuan dari studi literatur adalah untuk memahami landasan teori dan konsep penelitian sebelumnya yang dapat mendukung penelitian ini.

3.2 Wawancara

Wawancara dilakukan dengan cara berinteraksi langsung dengan narasumber yang memiliki pengetahuan atau pengalaman terkait topik penelitian. Wawancara dilaksanakan secara *online* melalui *platform Google Meet*, dan secara *offline* atau tatap muka. Tujuan wawancara adalah untuk mendapatkan informasi mendalam, terkait kebutuhan, atau data yang tidak dapat diperoleh melalui studi literatur.

3.3 Analisa Kebutuhan

Analisa kebutuhan dilakukan berdasarkan proyek *handbook sphere* tahun 2018. Metode wawancara juga dilakukan untuk menganalisa kebutuhan pada sistem yang akan dikembangkan. Dari hasil wawancara yang sudah dilakukan, sistem akan digunakan untuk mitigasi atau persiapan sebelum terjadinya bencana untuk mengetahui kebutuhan minimum yang harus dipersiapkan. Sehingga relawan tahu

kebutuhan yang harus segera disiapkan. Cakupan luas yang akan dikerjakan adalah Kecamatan Panggarangan karena satu desa memiliki luas yang lumayan besar.

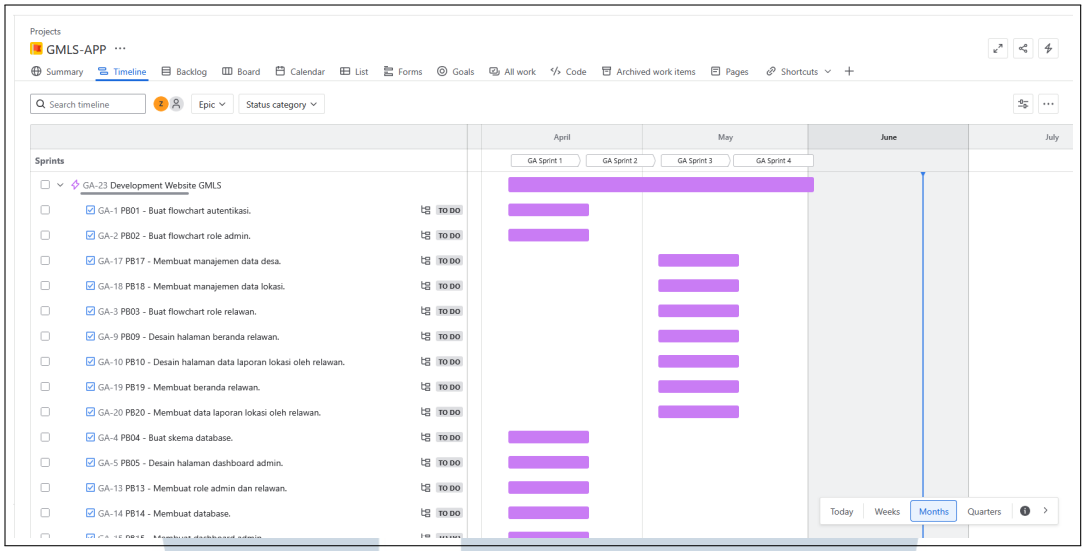
Dengan analisa yang sudah dilakukan dengan studi literatur dan wawancara dapat ditentukan fitur kebutuhan sistem sebagai berikut.

1. *Website* akan digunakan oleh GMLS bersama relawan namun masyarakat tetap dapat mengakses sebagai media informasi.
2. Hak akses *website* dibagi menjadi *admin*, relawan, dan masyarakat umum. Dengan rincian *admin* memiliki hak penuh untuk semua fitur, relawan memiliki hak akses pendataan lokasi pengungsian, dan masyarakat memiliki hak akses informasi umum terkait lokasi pengungsian.
3. *Admin* memiliki tampilan sistem yang berbeda dengan relawan dan masyarakat umum.
4. *Admin* dapat mengakses halaman *dashboard* sebaran lokasi pengungsian, manajemen data relawan, manajemen data desa, dan manajemen data lokasi.
5. Relawan dapat mengakses halaman beranda, data laporan, tambah lokasi pengungsian dan profil.
6. Masyarakat umum hanya dapat mengakses halaman beranda untuk melihat peta sebaran lokasi pengungsian dan kebutuhan minimum pada lokasi pengungsian.
7. Kebutuhan minimum proyek *sphere* pada lokasi pengungsian dihitung berdasarkan luas lokasi pengungsian.

3.4 *Scrum*

3.4.1 *Backlog Grooming*

Backlog grooming dilakukan dengan berdiskusi bersama Ketua GMLS dan juga mengambil dari analisa kebutuhan. Dari hasil diskusi yang sudah dilakukan, ditentukan *timeline* pengembangan aplikasi selama 8 minggu atau 4 *sprint*. Terdapat 24 *backlog* yang harus diselesaikan mencakup perencanaan, desain, pengembangan, dan pengujian. Pada tahap ini juga ditentukan *story point* pada masing - masing *backlog* sebagai estimasi pengerjaan *backlog* dalam satuan jam.



Gambar 3.2. *Timeline*

Gambar 3.2 menunjukkan durasi pengembangan website manajemen bencana yang dimulai dari Bulan April hingga Bulan Mei.

3.4.2 *Product Backlog*

Product Backlog adalah daftar prioritas yang harus dikembangkan. Product Backlog didapat dari proses *Backlog Grooming* yang terdiri dari 24 *product backlog*:

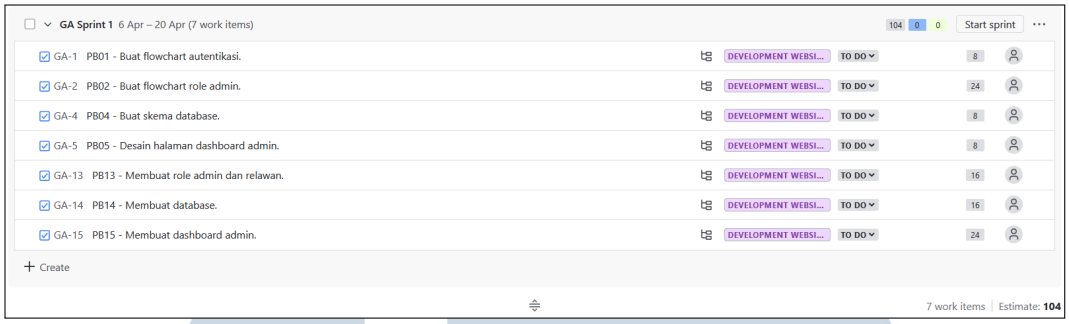
1. PB01 - Buat *flowchart* autentikasi (8 *story point*).
2. PB02 - Buat *flowchart role admin* (24 *story point*).
3. PB03 - Buat *flowchart role* relawan (8 *story point*).
4. PB04 - Buat skema *database* (8 *story point*).
5. PB05 - Desain halaman *dashboard admin* (8 *story point*).
6. PB06 - Desain halaman manajemen data relawan (16 *story point*).
7. PB07 - Desain halaman manajemen data desa (16 *story point*).
8. PB08 - Desain halaman manajemen data lokasi (16 *story point*).
9. PB09 - Desain halaman beranda relawan (8 *story point*).
10. PB10 - Desain halaman data laporan lokasi oleh relawan (8 *story point*).

11. PB11 - Desain halaman tambah lokasi oleh relawan (8 *story point*).
12. PB12 - Desain halaman *profil* relawan (8 *story point*).
13. PB13 - Membuat role *admin* dan *relawan* (16 *story point*).
14. PB14 - Membuat *database* (16 *story point*).
15. PB15 - Membuat *dashboard admin* (24 *story point*).
16. PB16 - Membuat manajemen data relawan (24 *story point*).
17. PB17 - Membuat manajemen data desa (24 *story point*).
18. PB18 - Membuat manajemen data lokasi (32 *story point*).
19. PB19 - Membuat beranda relawan (16 *story point*).
20. PB20 - Membuat data laporan lokasi oleh relawan (16 *story point*).
21. PB21 - Membuat tambah lokasi oleh relawan (16 *story point*).
22. PB22 - Membuat *profil* relawan (16 *story point*).
23. PB23 - Desain halaman *login* dan *register* (8 *story point*).
24. PB24 - Membuat halaman *login* dan *register* (24 *story point*).

3.4.3 *Sprint Planning*

Berdasarkan *backlog grooming* dan *product backlog* yang sudah diketahui. Dilakukan penjadwalan proses *development* selama 4 *sprint* dengan satuan kerja 8 jam 1 hari, maka perencanaannya adalah sebagai berikut.

1. *Sprint 1* (6 April 2025 - 20 April 2025)
   (a) *Sprint Planning*
   Gambar 3.3 menunjukkan daftar *backlog* yang dikerjakan pada *sprint* 1. *Sprint* 1 memiliki total estimasi *story point* sebesar 104. Beberapa tugas yang dikerjakan sebagai prioritas pada *sprint* 1, yaitu:
      i. PB01 - Buat *flowchart* autentikasi (8 *story point*).
      ii. PB02 - Buat *flowchart role admin* (24 *story point*).



Gambar 3.3. *Sprint* 1

iii. PB04 - Buat skema *database* (8 *story point*).

iv. PB05 - Desain halaman *dashboard admin* (8 *story point*).

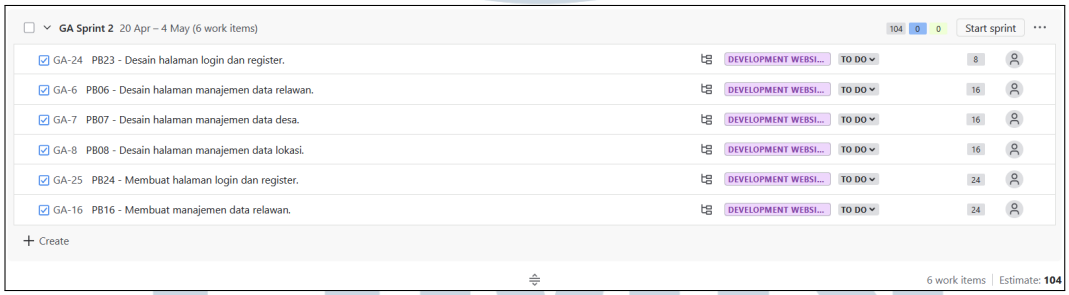
v. PB13 - Membuat role *admin* dan *relawan* (16 *story point*).

vi. PB14 - Membuat *database* (16 *story point*).

vii. PB15 - Membuat *dashboard admin* (24 *story point*).

2. *Sprint 2*

(a) *Sprint Planning*



Gambar 3.4. *Sprint* 2

Gambar 3.4 menunjukkan daftar *backlog* yang dikerjakan pada *sprint* 2. *Sprint* 2 memiliki total estimasi *story point* sebesar 104. Beberapa tugas yang dikerjakan sebagai prioritas pada *sprint* 2, yaitu

i. PB23 - Desain halaman *login* dan *register* (8 *story point*).

ii. SB06 - Desain halaman manajemen data relawan (16 *story point*).

iii. SB07 - Desain halaman manajemen data desa (16 *story point*).

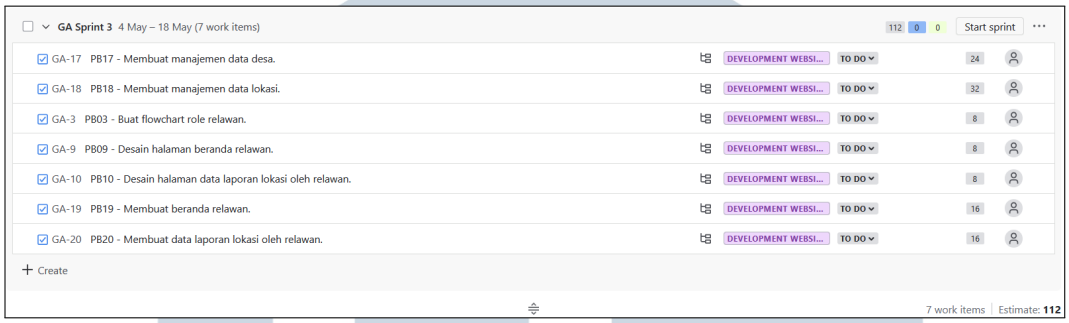
iv. SB08 - Desain halaman manajemen data lokasi (16 *story point*).

v. SB24 - Membuat halaman *login* dan *register* (24 *story point*).

vi. SB16 - Membuat manajemen data relawan (24 *story point*).

3. *Sprint 3*

(a) *Sprint Planning*



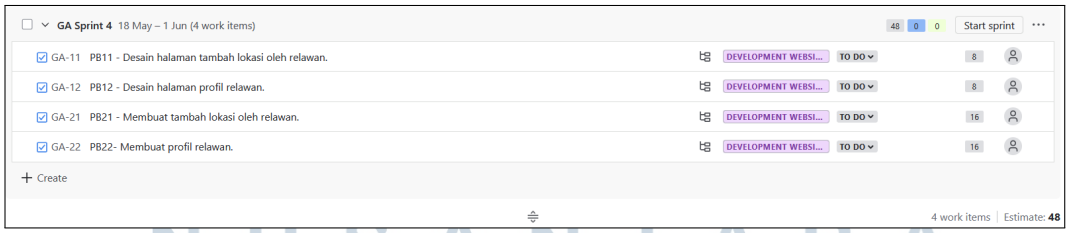
Gambar 3.5. *Sprint* 3

Gambar 3.5 menunjukkan daftar *backlog* yang dikerjakan pada *sprint* 3. *Sprint* 3 memiliki total estimasi *story point* sebesar 112. Beberapa tugas yang dikerjakan sebagai prioritas pada *sprint* 3, yaitu:

i. PB17 - Membuat manajemen data desa (24 *story point*).
ii. PB18 - Membuat manajemen data lokasi (32 *story point*).
iii. PB03 - Buat *flowchart role* relawan (8 *story point*).
iv. PB09 - Desain halaman beranda relawan (8 *story point*).
v. PB10 - Desain halaman data laporan lokasi oleh relawan (8 *story point*).
vi. PB19 - Membuat beranda relawan (16 *story point*).
vii. PB20 - Membuat data laporan lokasi oleh relawan (16 *story point*).

4. *Sprint 4*

(a) *Sprint Planning*



Gambar 3.6. *Sprint* 4

Gambar 3.6 menunjukkan daftar *backlog* yang dikerjakan pada *sprint* 4. *Sprint* 4 memiliki total estimasi *story point* sebesar 48. Beberapa tugas yang dikerjakan sebagai prioritas pada *sprint* 4, yaitu:

i. PB11 - Desain halaman tambah lokasi oleh relawan (8 *story point*).
ii. PB12 - Desain halaman *profil* relawan (8 *story point*).
iii. PB21 - Membuat tambah lokasi oleh relawan (16 *story point*).
iv. PB22 - Membuat *profil* relawan (16 *story point*).

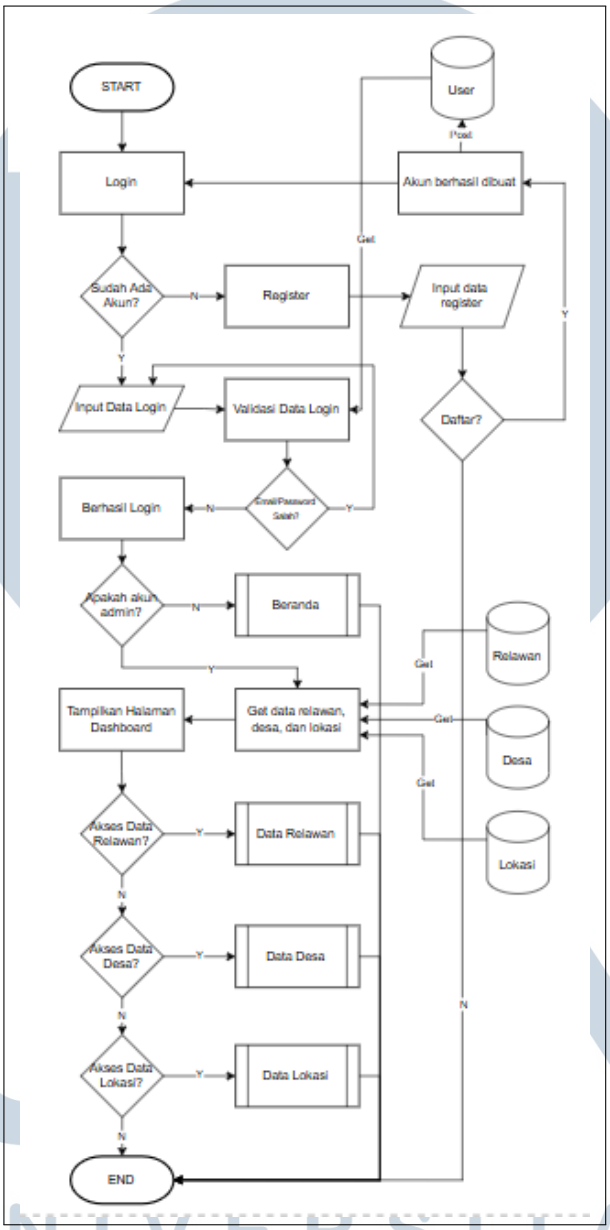
3.4.4 *Daily Scrum*

Daily scrum akan dimulai dari proses perancangan sistem. Perancangan sistem dibuat berdasarkan *requirement website*. Perancangan meliputi *flowchart, entity relationship diagram (ERD), database schema, data flow diagram (DFD)* dan desain.

Setelah melakukan perancangan, dilakukan proses pengembangan yang akan dilakukan dengan menggunakan *framework laravel* dan *database PostgreSQL*. *Library* pendukung seperti *saptie role* juga digunakan untuk mendukung proses pengembangan.

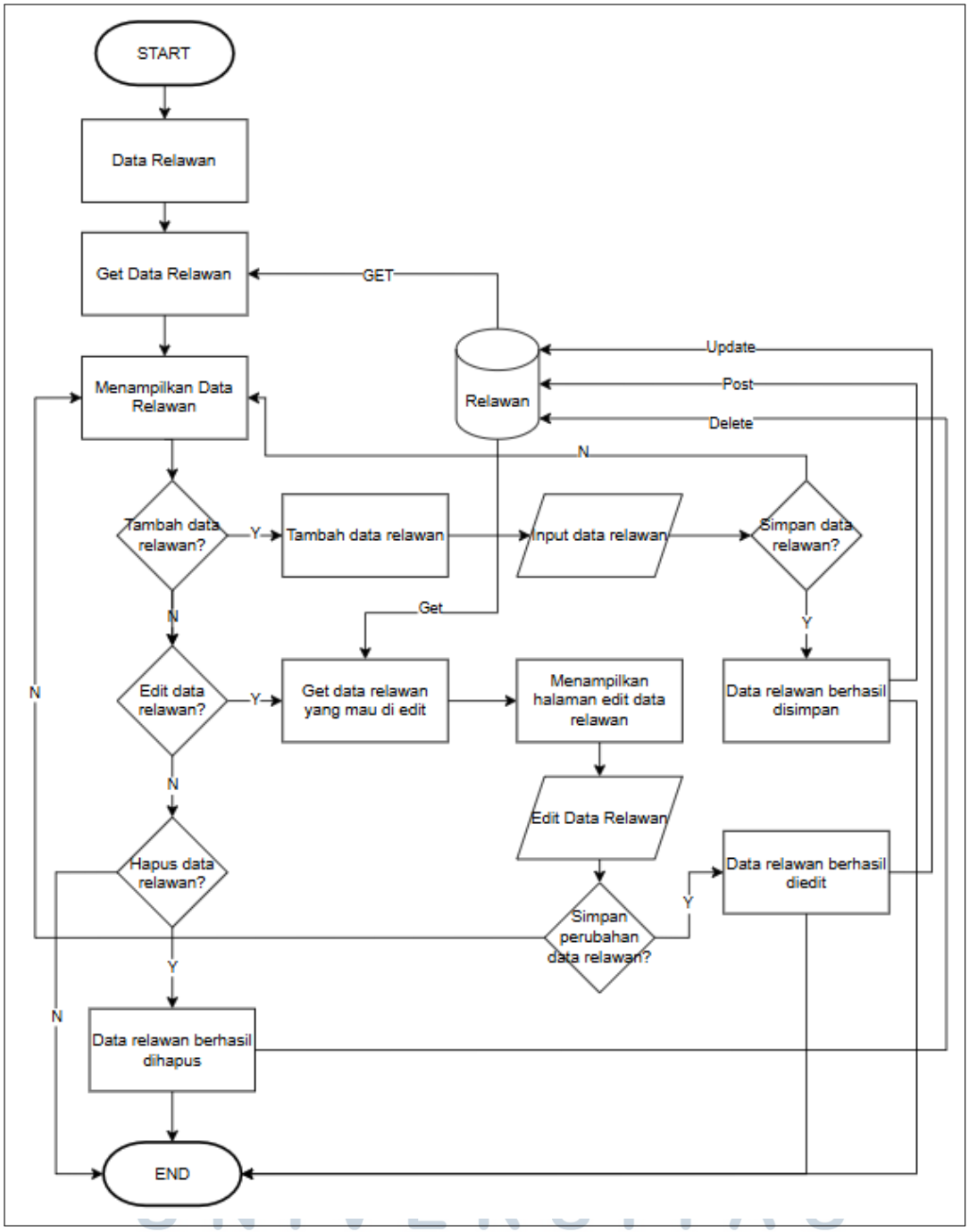
UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

A *Flowchart*



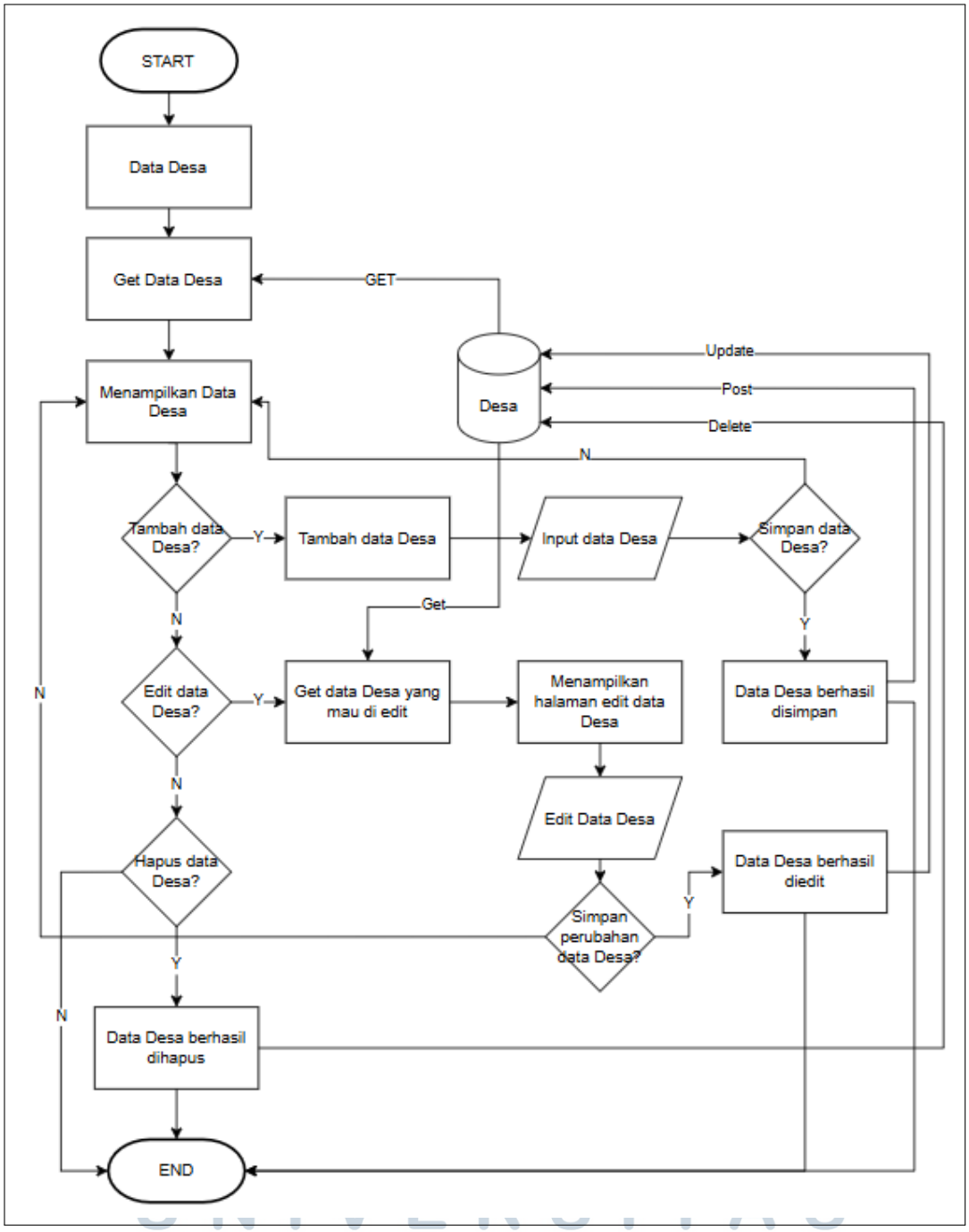
Gambar 3.7. *Flowchart Login* dan *Register*

Gambar 3.7 menjelaskan alur *login* dan *register*. Autentikasi *role* dilakukan saat proses *login*, jika akun adalah seorang *admin* akan dialihkan ke halaman *admin* dan jika relawan akan dialihkan ke halaman relawan. *Register* hanya dilakukan untuk pembuatan akun relawan.



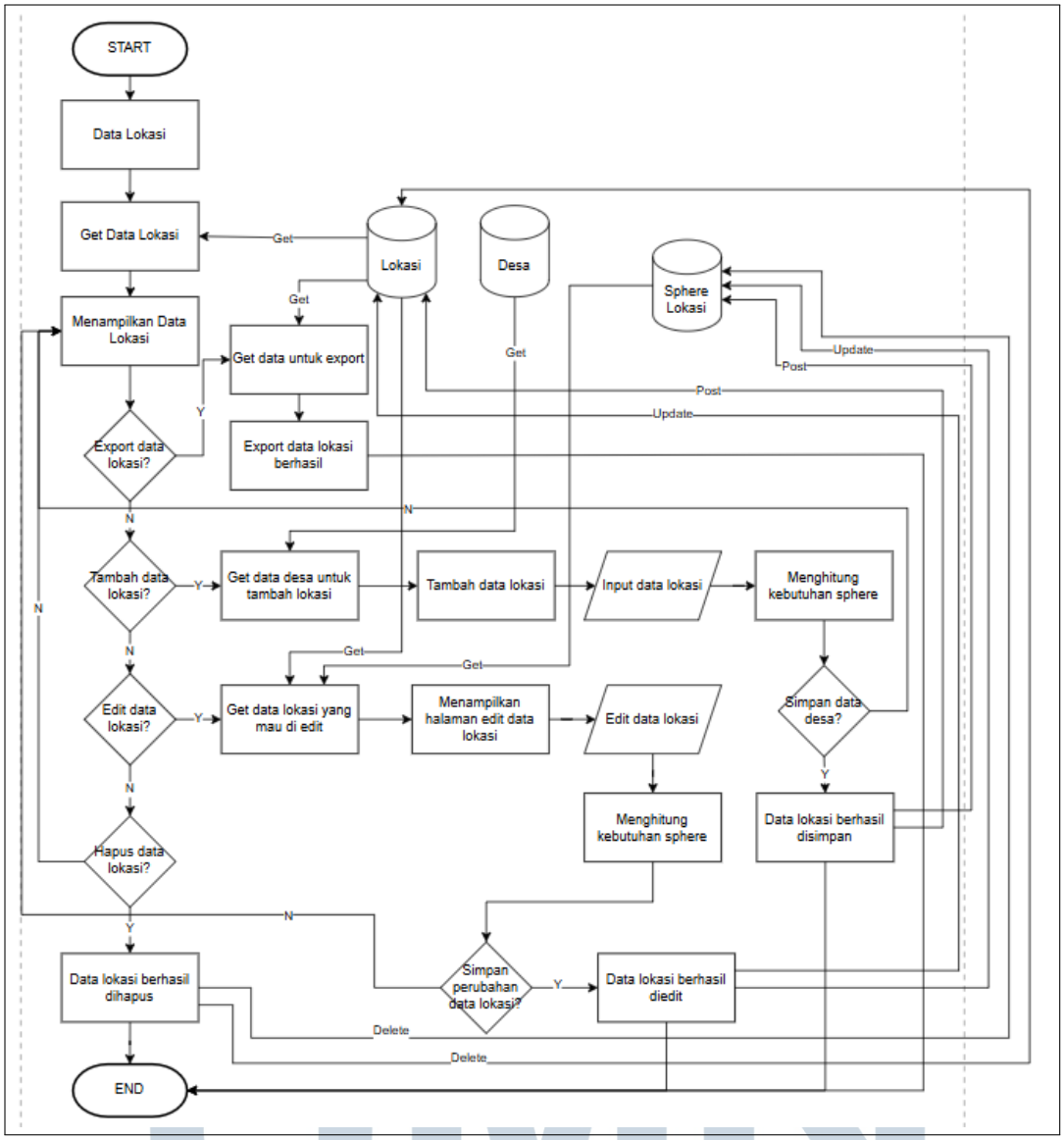
Gambar 3.8. *Flowchart* Data Relawan

Gambar 3.8 menjelaskan fitur yang dimiliki oleh seorang *admin*. *Admin* memiliki hak untuk melakukan manajemen data relawan, mulai dari pembuatan, edit, dan hapus data relawan.



Gambar 3.9. *Flowchart* Data Desa

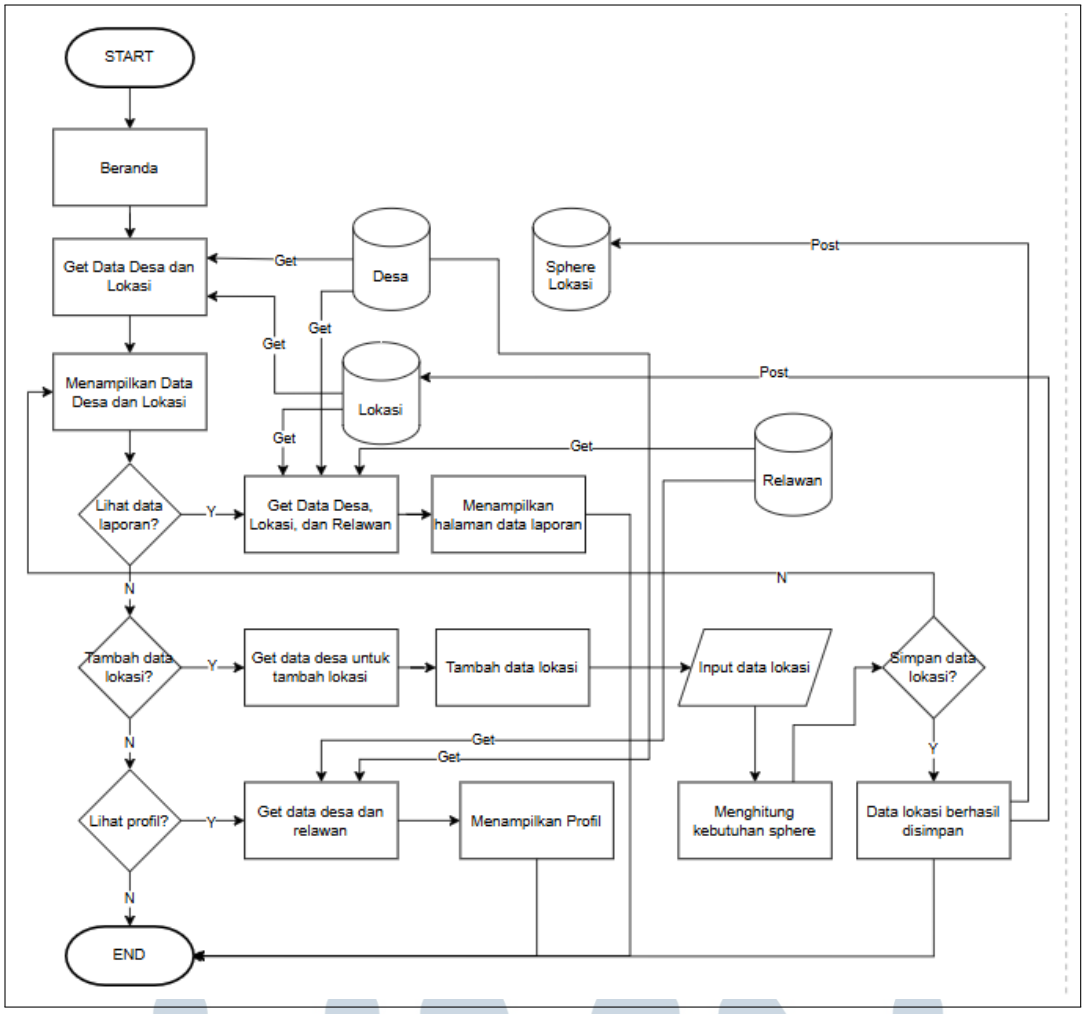
Gambar 3.9 menjelaskan fitur yang dimiliki oleh seorang *admin*. *Admin* memiliki hak untuk melakukan manajemen data desa yang akan menampung data lokasi pengungsian, mulai dari pembuatan, edit, dan hapus data desa.



Gambar 3.10. *Flowchart* Data Lokasi

Gambar 3.10 menjelaskan fitur yang dimiliki oleh seorang *admin*. *Admin* memiliki hak untuk melakukan manajemen data lokasi dan export data lokasi. Pada penambahan data lokasi, sistem juga melakukan perhitungan proyek *sphere*.

UNIVERSITAS MULTIMEDIA NUSANTARA

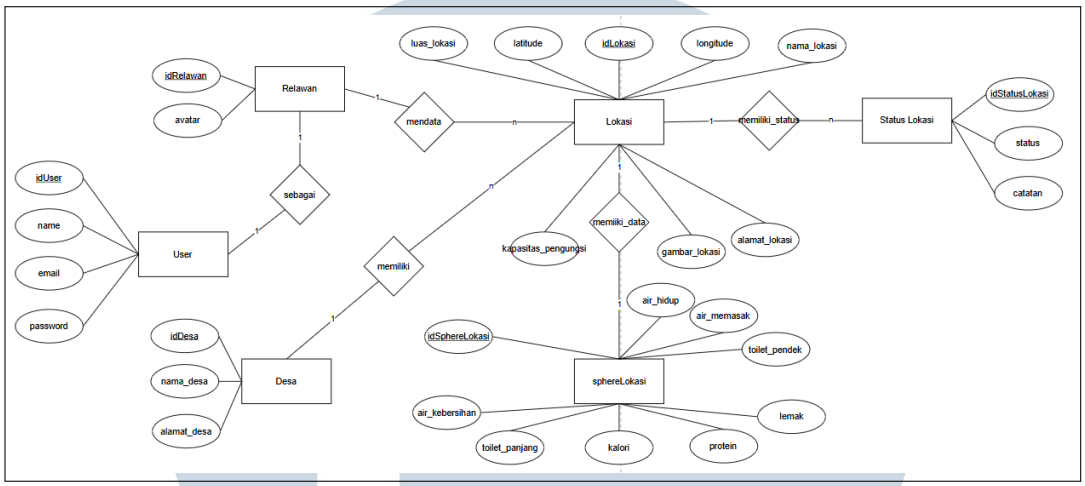


Gambar 3.11. *Flowchart* Beranda

Gambar 3.11 menjelaskan fitur yang dimiliki oleh seorang relawan. Relawan dapat mengakses halaman beranda, halaman data laporan, menambah data lokasi pengungsian, dan halaman *profil*.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

B *Entity Relationship Diagram (ERD)*



Gambar 3.12. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) pada gambar 3.12 menggambarkan struktur basis data untuk sistem mitigasi bencana berbasis proyek *Sphere*. Sistem ini terdiri dari enam entitas utama: *User*, Relawan, Desa, Lokasi, Sphere Lokasi, dan Status Lokasi, yang saling terhubung melalui berbagai hubungan (relasi).

Entitas *User* merepresentasikan pengguna sistem, yang memiliki atribut seperti name, email, dan password. Setiap *user* dapat memiliki satu data sebagai Relawan, yang menyimpan informasi lanjutan seperti avatar dan berelasi satu-satu melalui hubungan memiliki_data. Relawan kemudian memiliki peran penting dalam mendaftarkan data Lokasi, yang merupakan pusat dari sistem. Setiap relawan dapat mendaftarkan lebih dari satu lokasi (relasi *one to many*).

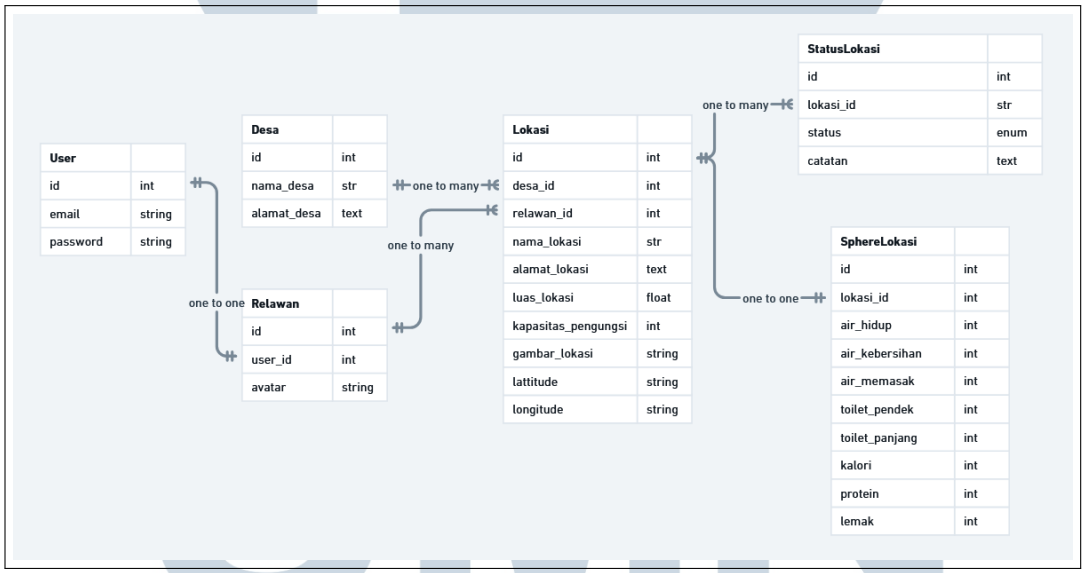
Entitas Lokasi menyimpan berbagai data penting terkait pengungsian, seperti nama dan alamat lokasi, kapasitas pengungsi, ukuran lokasi, serta koordinat geografis (*latitude*, *longitude*). Lokasi juga berelasi dengan entitas Desa melalui relasi berada_di, yang menunjukkan bahwa sebuah lokasi pengungsian terletak dalam satu desa tertentu. Setiap desa dapat memiliki banyak lokasi (relasi *one to many*).

Setiap lokasi memiliki satu data terkait kebutuhan dasar manusia berdasarkan proyek *Sphere*, yang disimpan dalam entitas *Sphere* Lokasi. Relasi ini ditunjukkan dengan hubungan memiliki_data yang bersifat *one to one*. Data dalam Sphere Lokasi meliputi ketersediaan air bersih (air_hidup, air_memasak, air_kebersihan), sanitasi (toilet_pendek, toilet_panjang), serta kebutuhan gizi (kalori, protein, lemak).

Selain itu, Lokasi juga dapat memiliki banyak catatan atau kondisi status terkini yang disimpan dalam entitas Status Lokasi, seperti status kelayakan lokasi atau catatan khusus lainnya. Relasi ini ditunjukkan dengan memiliki_status, yang bersifat *one to many*, mencerminkan bahwa satu lokasi bisa mengalami berbagai status.

Secara keseluruhan, ERD ini membentuk sistem data yang saling terhubung antara pengguna, relawan, lokasi pengungsian, desa, dan kebutuhan dasar berbasis proyek *Sphere*. ERD ini mendukung pengelolaan data yang efektif untuk keperluan monitoring, logistik, dan pengambilan keputusan dalam konteks tanggap darurat bencana.

C *Database Schema*

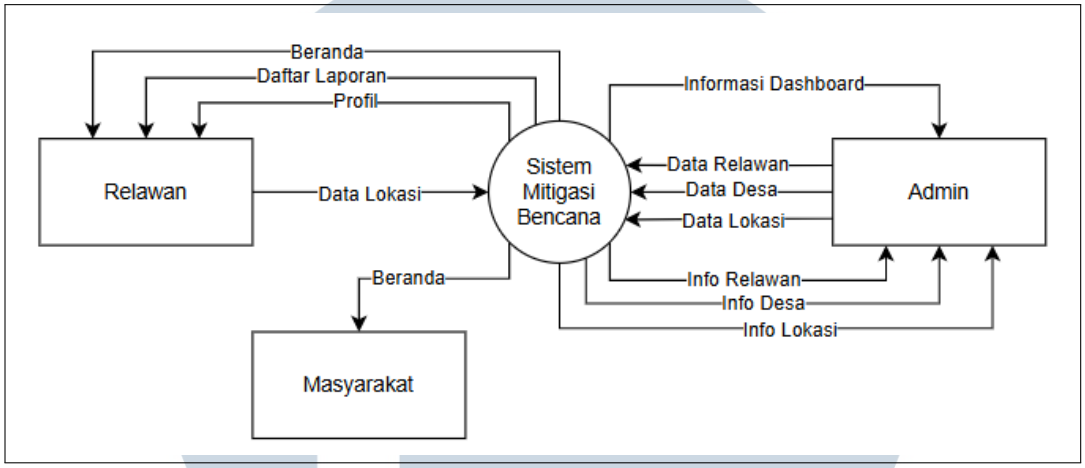


Gambar 3.13. *Database Schema*

Gambar 3.13 menjelaskan relasi tabel *website* manajemen bencana berbasis *project sphere*. Skema *Database* ini terdiri dari tabel *user* yang digunakan untuk menampung akun pengguna termasuk *admin*. Tabel relawan digunakan untuk menyimpan data pengguna sebagai relawan. Tabel desa digunakan untuk menyimpan data desa yang berelasi dengan tabel lokasi. Tabel lokasi untuk menyimpan data lokasi yang diajukan oleh *admin* dan juga relawan berdasarkan desa yang sudah dibuat. Tabel status lokasi berisikan status dari lokasi yang telah diajukan. Kemudian tabel *sphere* lokasi digunakan untuk menyimpan hasil perhitungan standar minimun kebutuhan dari sebuah lokasi.

D *Data Flow Diagram (DFD)*

1. DFD Level 0

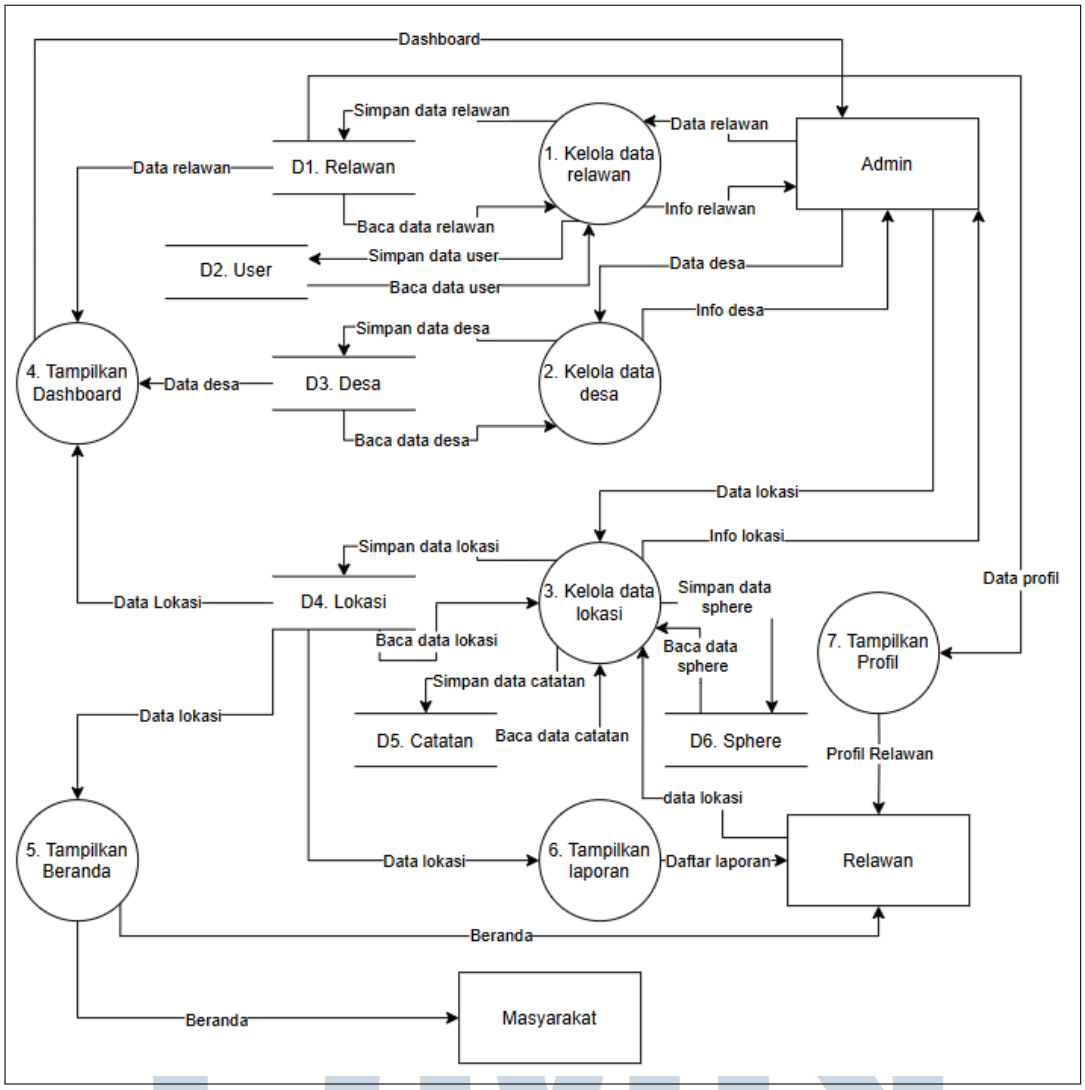


Gambar 3.14. DFD Level 0

Gambar 3.14 menunjukkan *Data Flow Diagram (DFD)* untuk level 0 yang terdiri dari tiga entitas, yaitu *admin*, relawan dan masyarakat. Terdapat satu proses, yaitu *website* mitigasi bencana. *Admin* memiliki *data flow* data relawan, *dashboard*, data lokasi, dan data desa. Relawan memiliki *data flow* lokasi, beranda, laporan lokasi, dan profil. Masyarakat memiliki *data flow* beranda.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

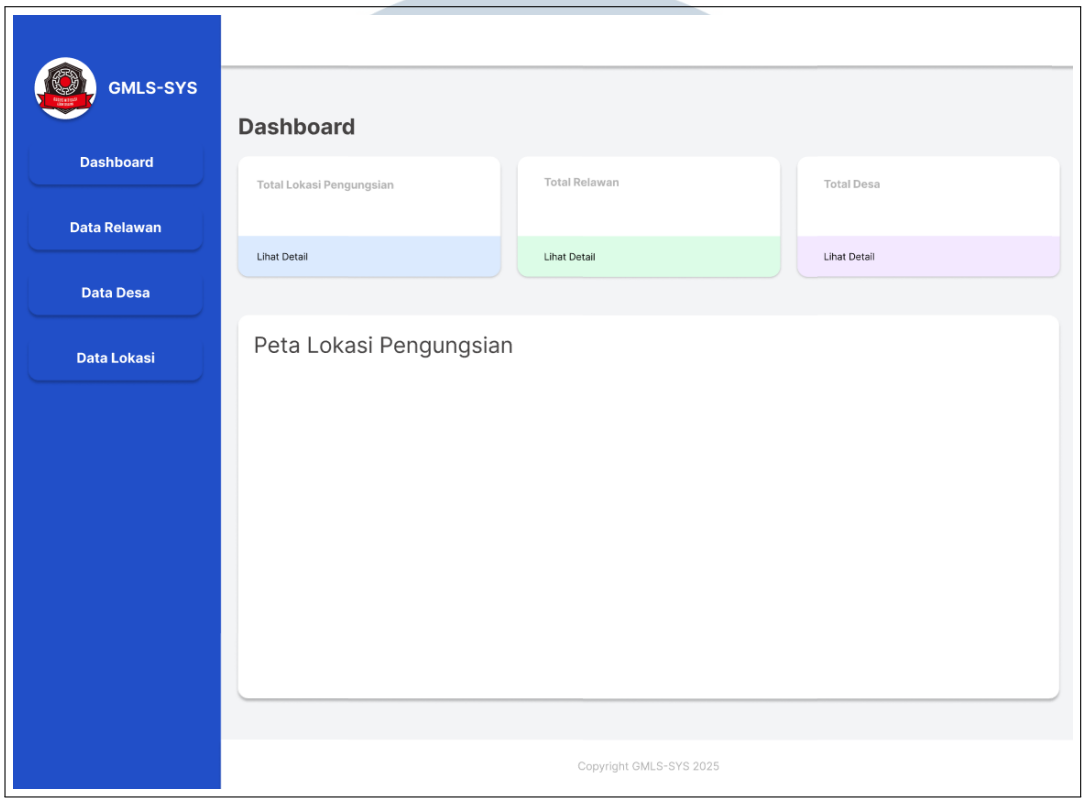
2. DFD Level 1



Gambar 3.15. DFD Level 1

Gambar 3.15 menunjukkan DFD level 1 yang lebih rinci. *Admin* memiliki akses untuk melihat dashboard, serta melakukan manajemen data relawan, desa, dan lokasi. Proses manajemen data oleh admin akan dilakukan penyimpanan ke masing masing *database* yaitu, relawan, desa, dan lokasi. Relawan memiliki akses untuk melihat laporan lokasi, beranda, profil dan menambahkan lokasi. Proses menambahkan lokasi dilakukan pada *database* lokasi. Masyarakat hanya memiliki hak akses menu beranda.

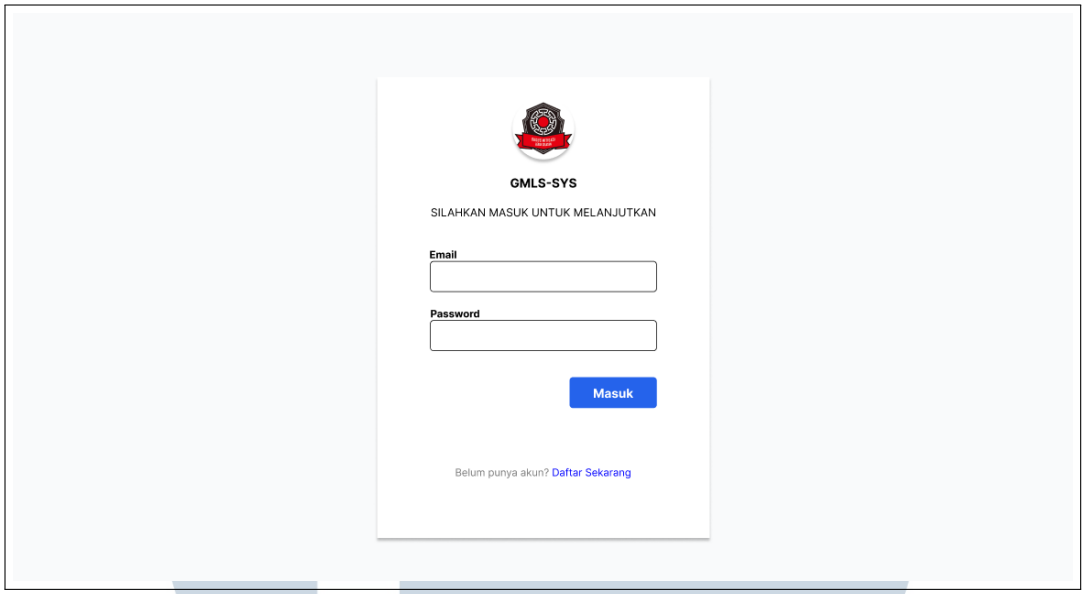
E Desain



Gambar 3.16. Halaman *Dashboard Admin*

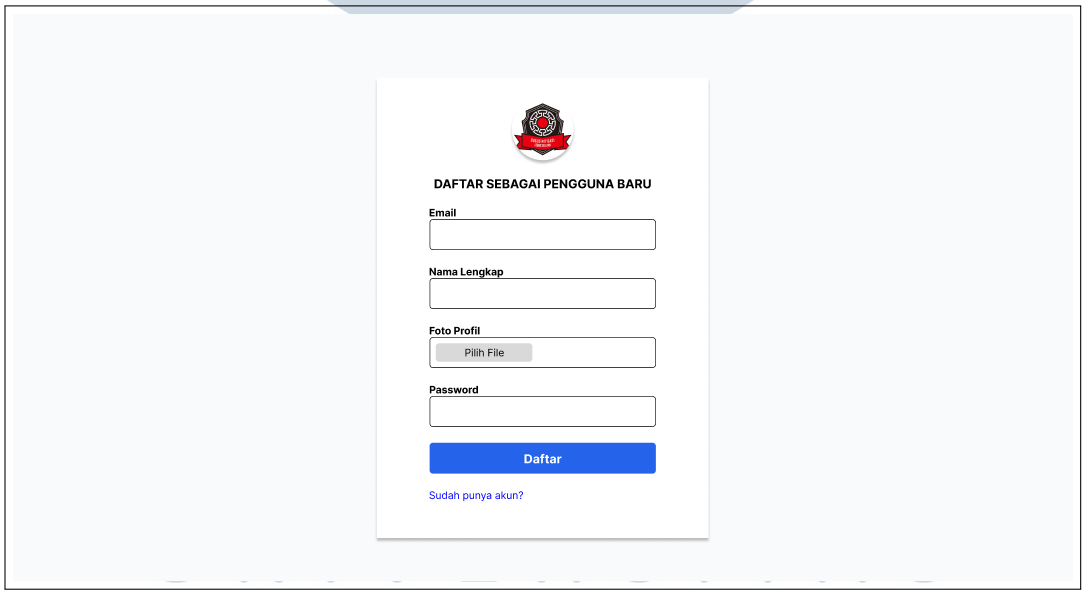
Gambar 3.16 adalah halaman *dashboard admin* guna menampilkan data manajemen. Data yang ditampilkan adalah jumlah relawan, jumlah desa, jumlah lokasi pengungsian, dan sebaran titik lokasi pengungsian yang didaftarkan.

UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.17. Halaman *Login*

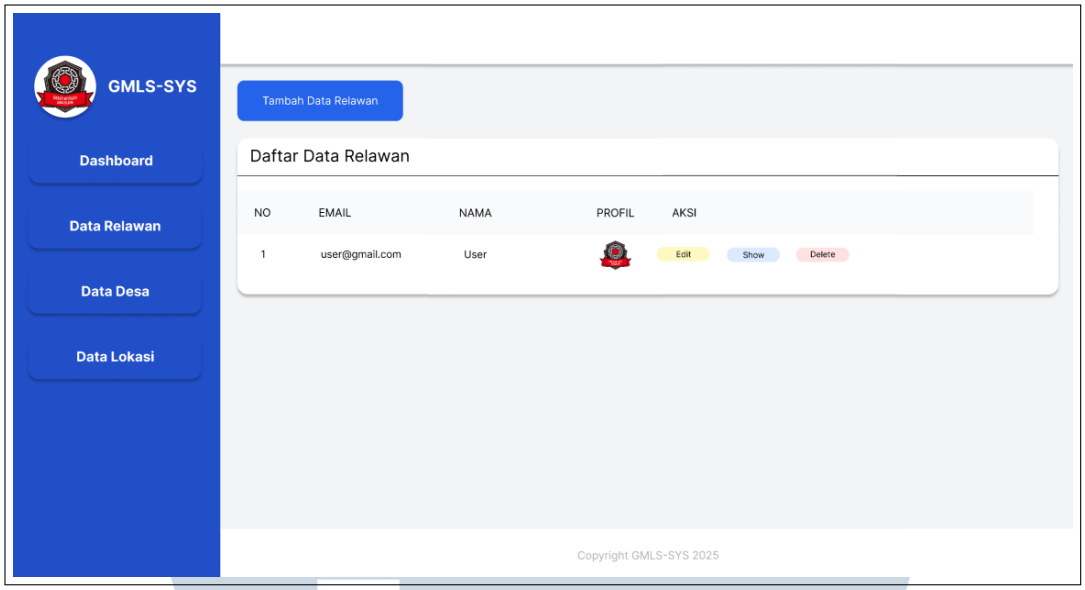
Gambar 3.17 adalah halaman *login*. Jika role akun adalah *admin*, maka pengguna akan dialhkan ke halaman *admin*.



Gambar 3.18. Desain Halaman Register

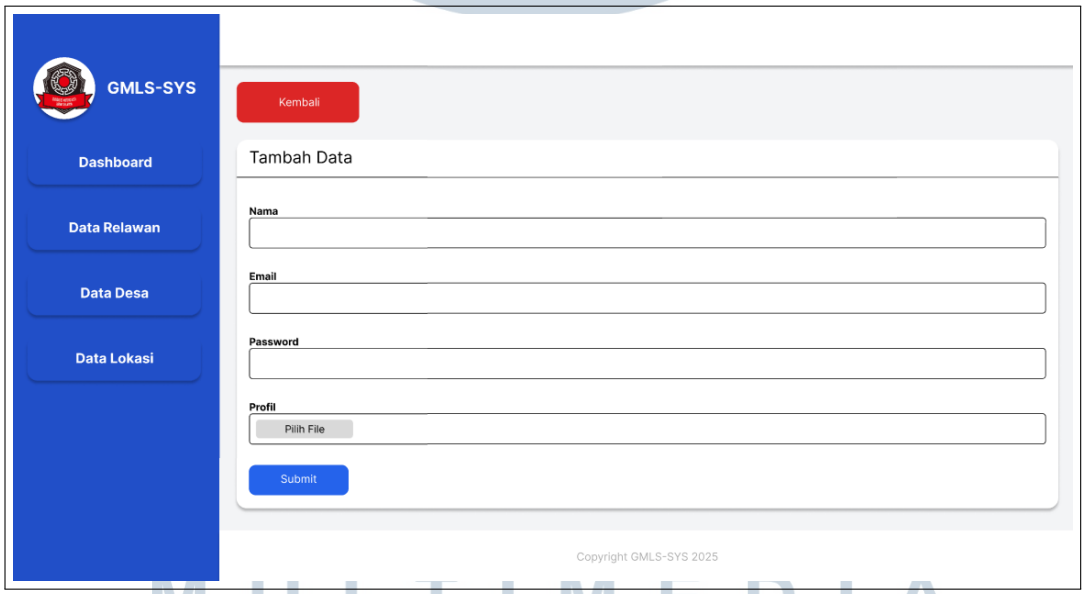
Gambar 3.18 adalah halaman daftar akun. Daftar akun hanya dapat digunakan oleh relawan.

MULTIMEDIA NUSANTARA



Gambar 3.19. Halaman Data Relawan

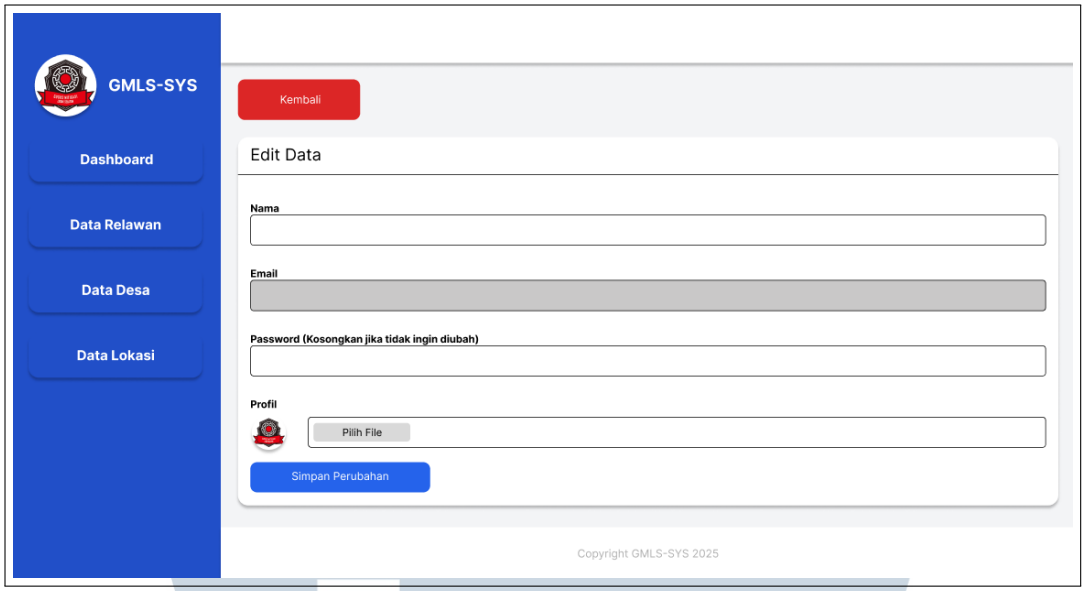
Gambar 3.19 adalah halaman utama fitur manajemen data relawan. *Admin* dapat membuat data relawan baru, melihat *detail* data relawan, mengubah data relawan, dan menghapus data relawan



Gambar 3.20. Halaman Tambah Data Relawan

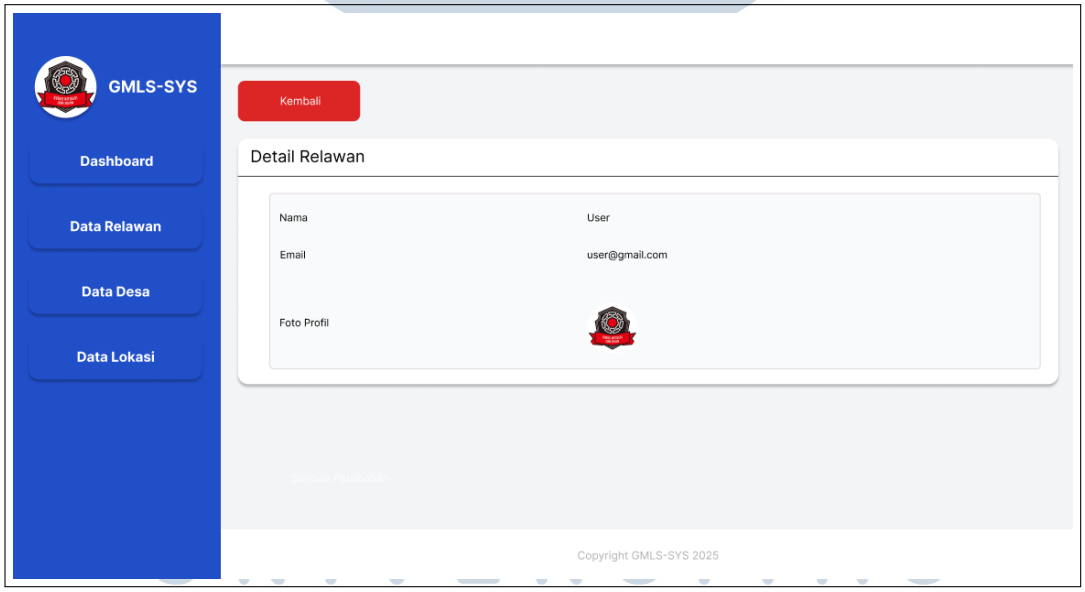
Gambar 3.20 adalah halaman untuk menambahkan data relawan dengan *input* berupa nama, *email*, *password*, dan foto profil.

MULTIMEDIA NUSANTARA



Gambar 3.21. Halaman *Edit* Data Relawan

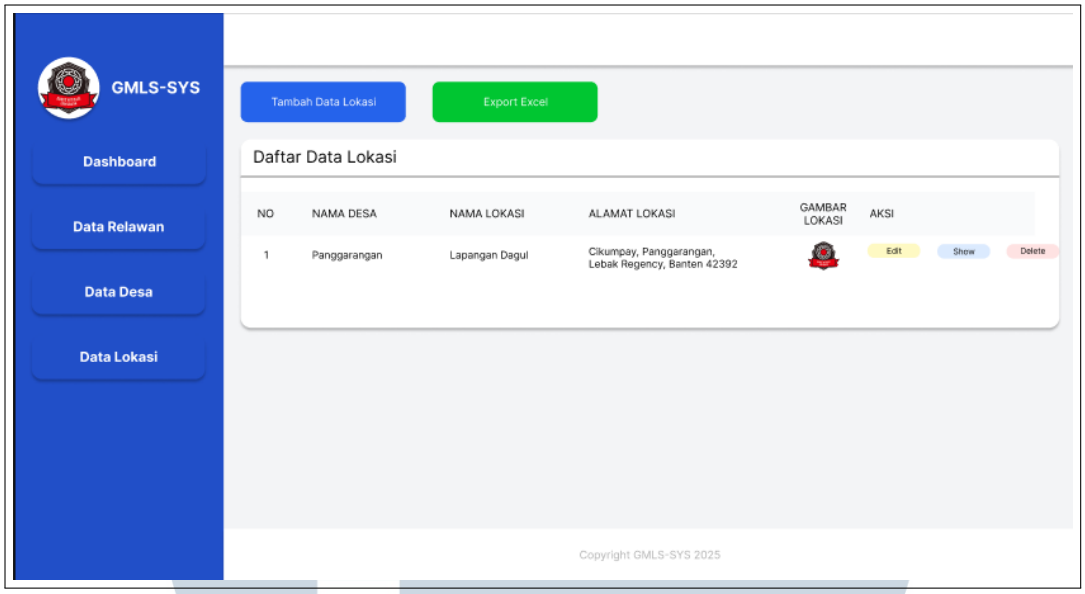
Gambar 3.21 adalah halaman untuk mengubah data relawan dengan *input* berupa nama, *email*, *password*, dan foto profil.



Gambar 3.22. Halaman *Detail* Data Relawan

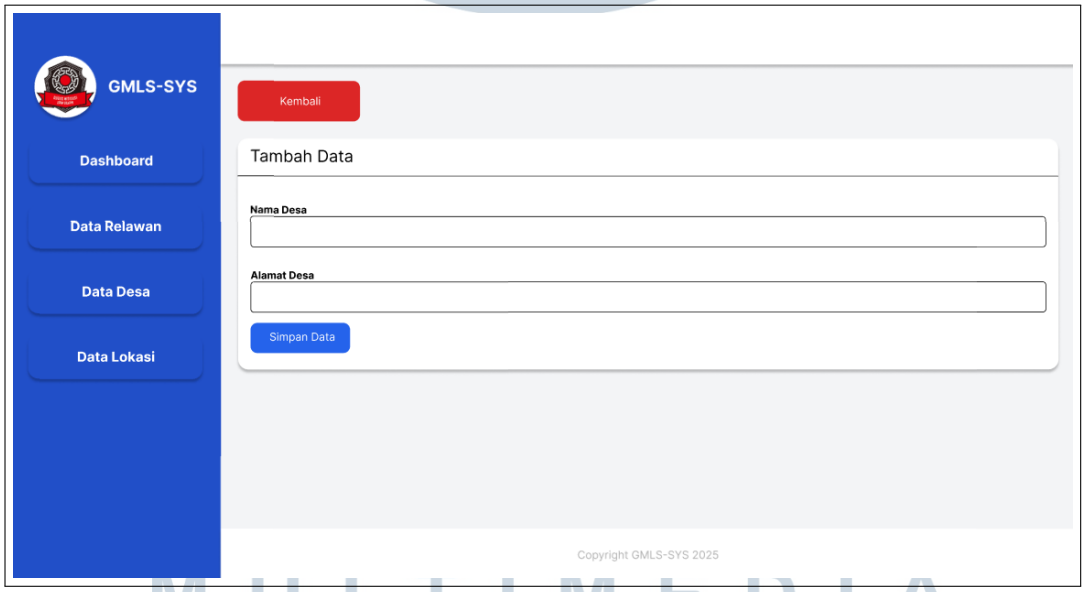
Gambar 3.22 adalah halaman untuk melihat detail data relawan.

MULTIMEDIA NUSANTARA



Gambar 3.23. Halaman Data Desa

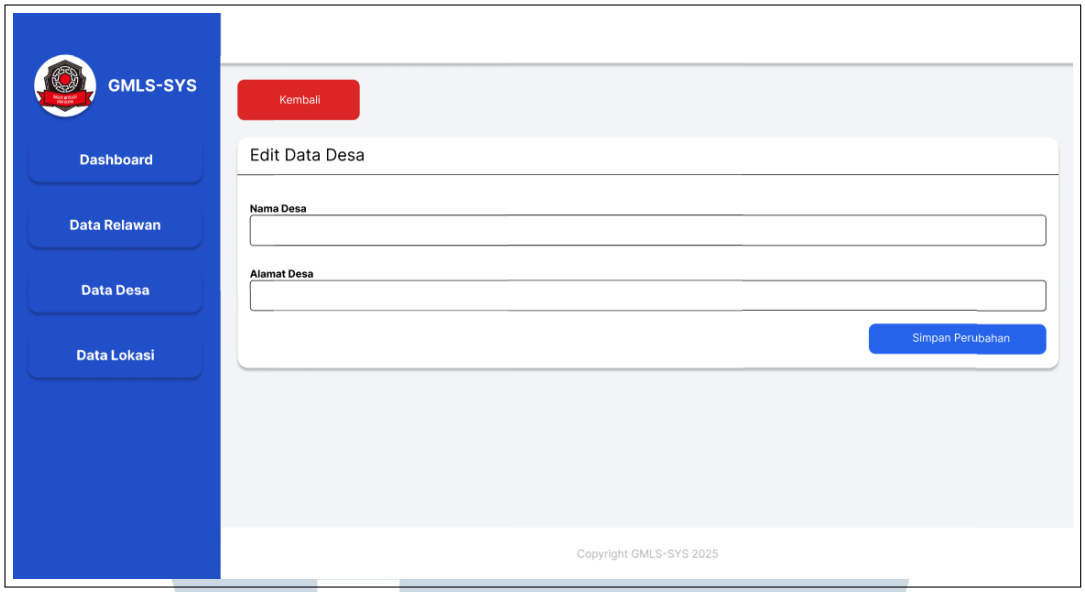
Gambar 3.23 adalah halaman utama fitur manajemen data desa. *Admin* dapat membuat data desa baru, melihat *detail* data desa, mengubah data desa, menghapus data desa, dan *export* data excel.



Gambar 3.24. Halaman Tambah Data Desa

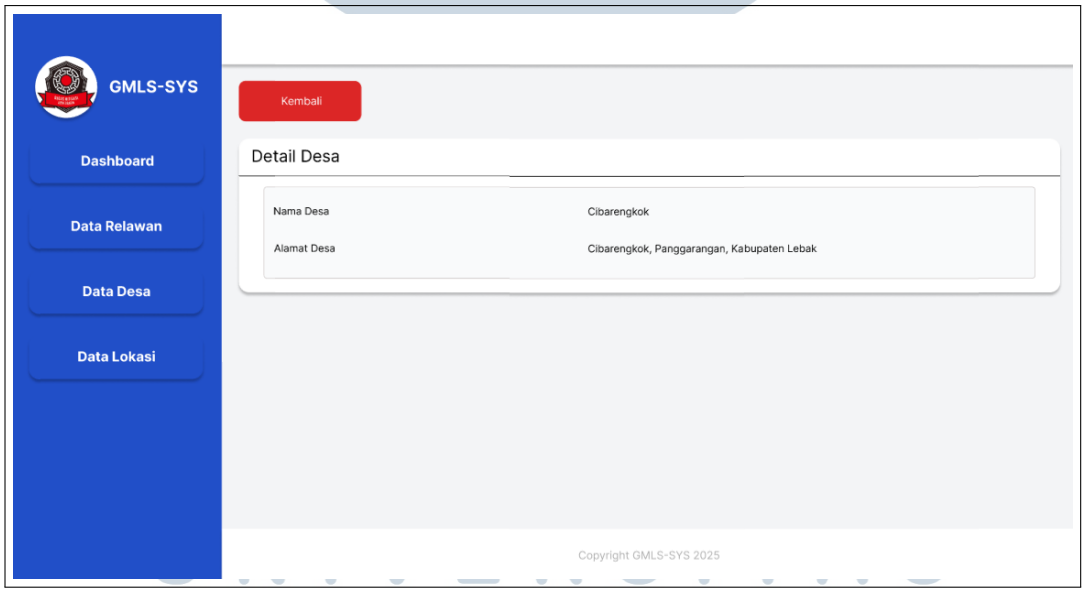
Gambar 3.24 adalah halaman untuk menambahkan data desa dengan *input* berupa nama desa dan alamat desa.

MULTIMEDIA NUSANTARA



Gambar 3.25. Halaman *Edit* Data Desa

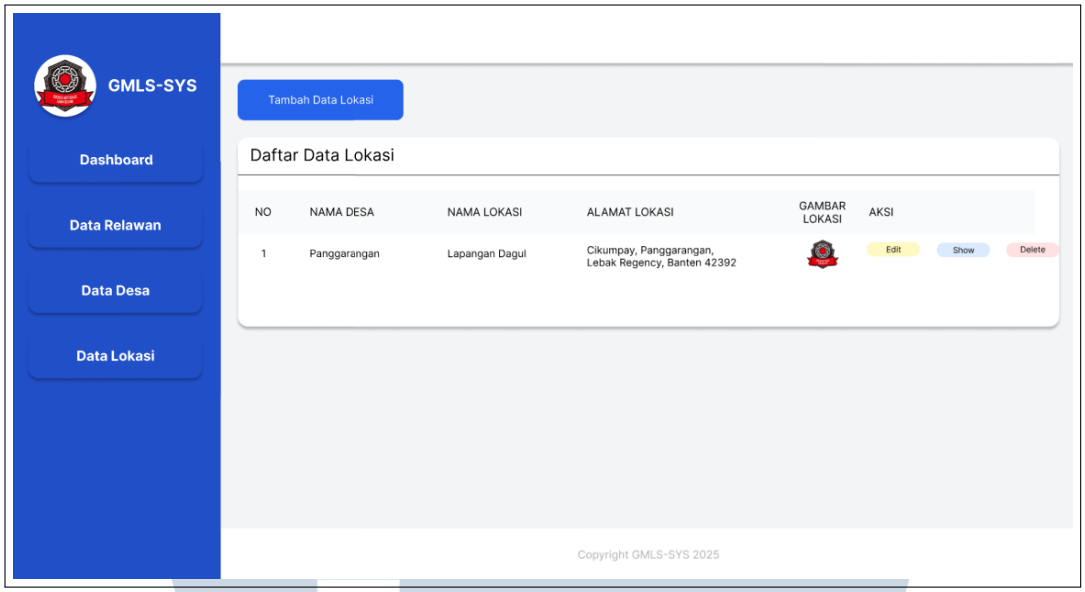
Gambar 3.25 adalah halaman untuk mengubah data desa dengan *input* berupa nama desa dan alamat desa.



Gambar 3.26. Halaman *Detail* Data Desa

Gambar 3.26 adalah halaman untuk melihat detail data desa berupa nama desa dan alamat desa.

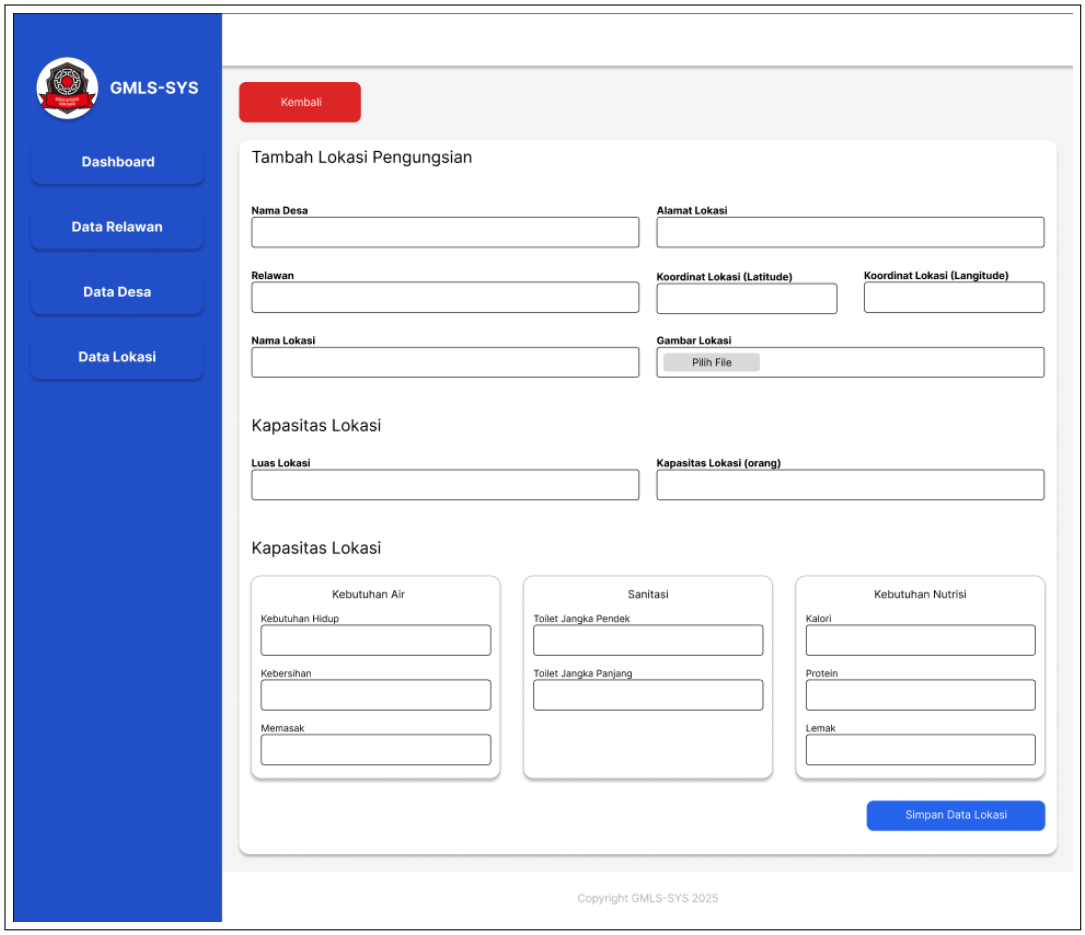
Halaman Data Lokasi



Gambar 3.27. Halaman Data Lokasi

Gambar 3.27 adalah halaman utama fitur manajemen data lokasi. *Admin* dapat membuat data lokasi baru, melihat *detail* data lokasi, mengubah data lokasi, dan menghapus data lokasi.

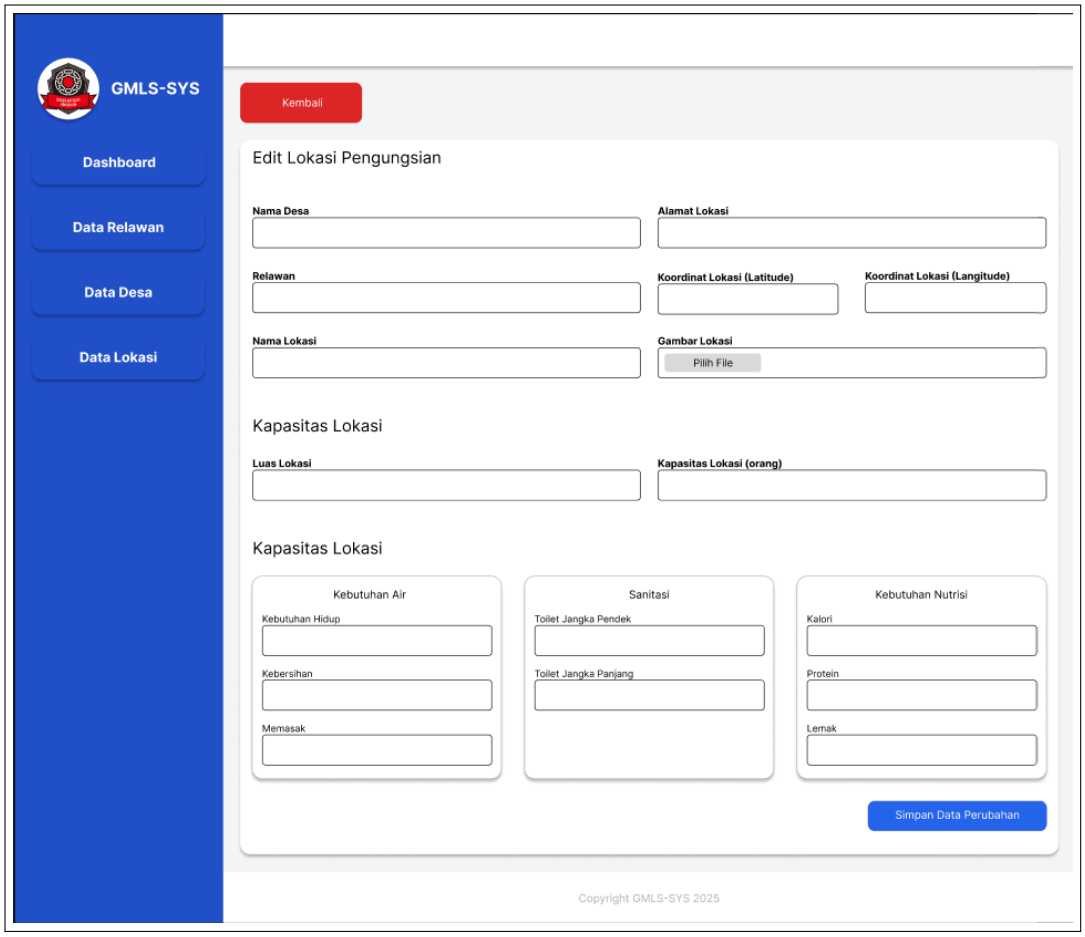
UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.28. Halaman Tambah Data Lokasi

Gambar 3.28 adalah halaman untuk menambahkan data lokasi dengan *input* berupa nama desa, alamat lokasi, relawan, koordinat lokasi, nama lokasi, gambar lokasi, dan luas lokasi. Berdasarkan *input* luas lokasi akan ditampilkan kapasistas lokasi, dan data kebutuhan proyek *sphere*.

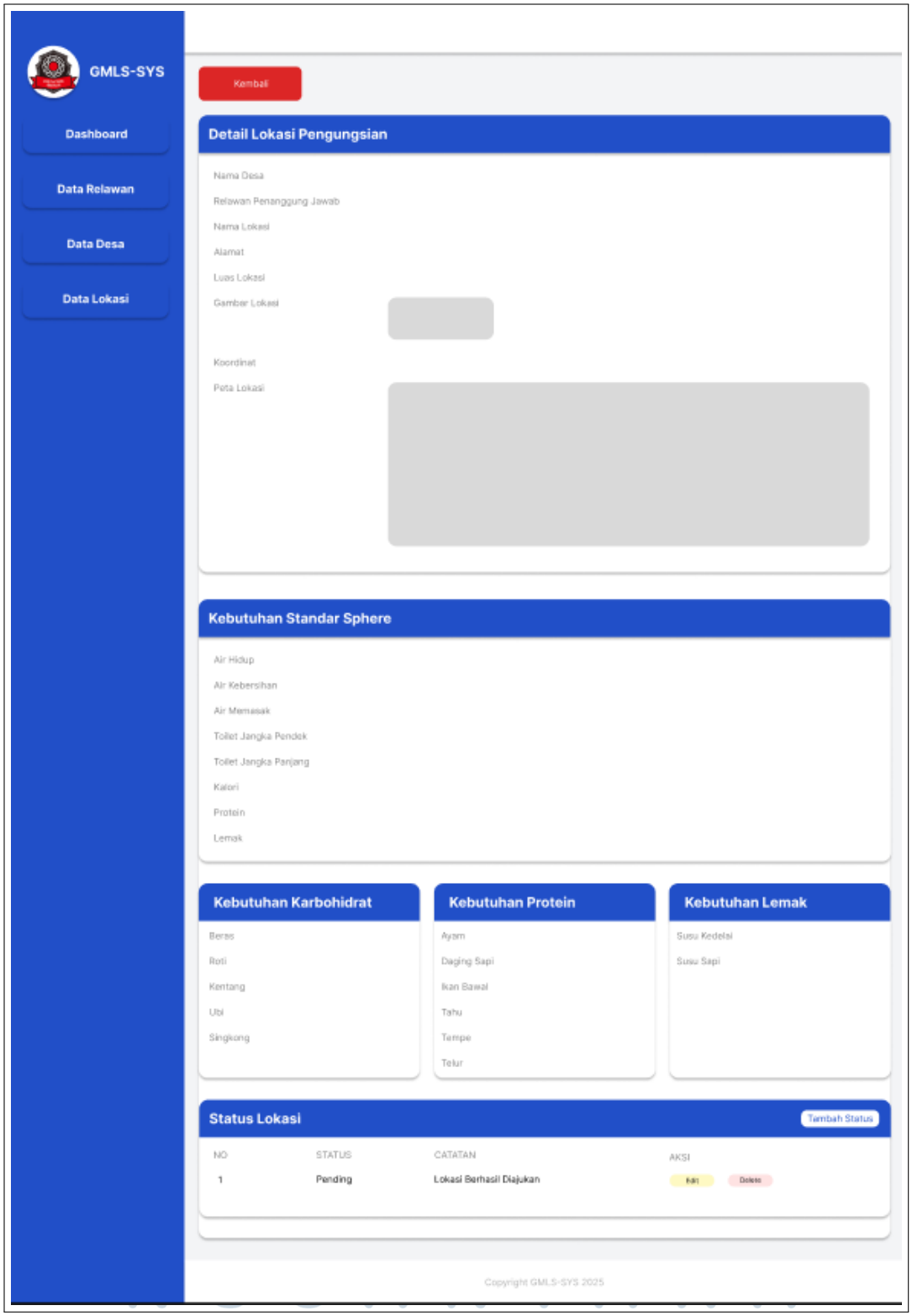
UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.29. Halaman Ubah Data Lokasi

Gambar 3.29 adalah halaman untuk mengubah data lokasi dengan *input* berupa nama desa, alamat lokasi, relawan, koordinat lokasi, nama lokasi, gambar lokasi, dan luas lokasi. Berdasarkan *input* luas lokasi akan ditampilkan kapasistas lokasi, dan data kebutuhan proyek *sphere*.

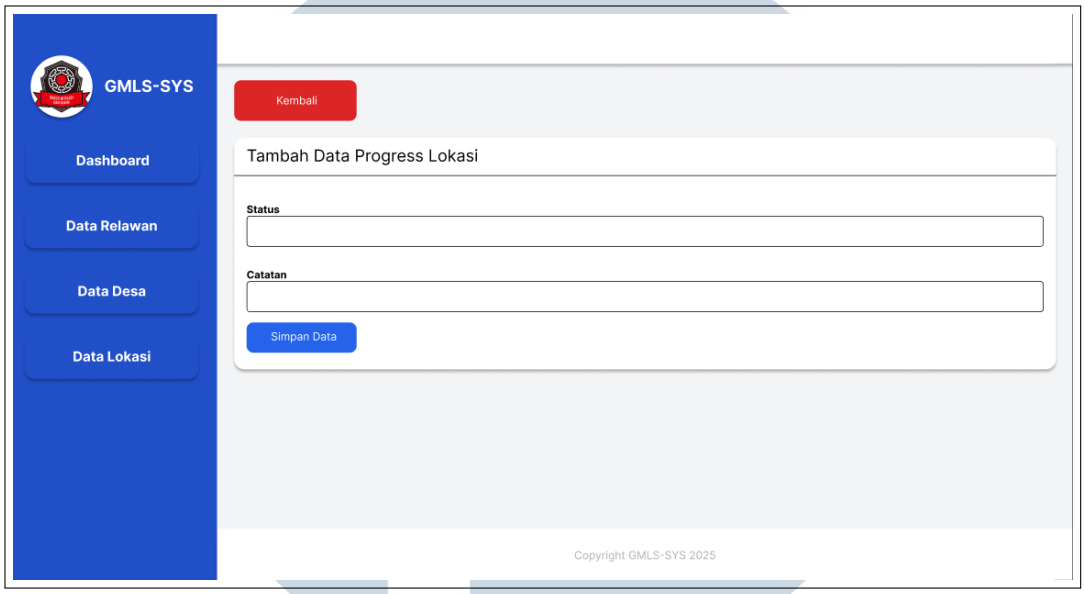
UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.30. Halaman *Detail* Data Lokasi

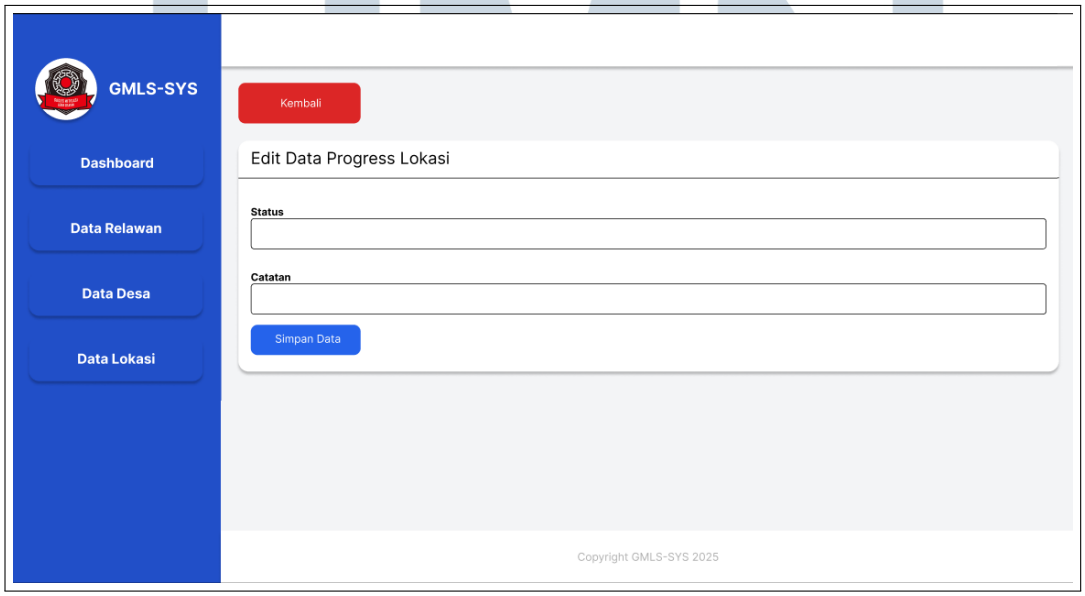
Gambar 3.30 adalah halaman untuk melihat *detail* data lokasi. Informasi yang ditampilkan berupa nama desa, relawan penanggung jawab, nama lokasi,

alamat, gambar lokasi, koordinat lokasi, peta lokasi, data kebutuhan proyek *sphere* dan konversi kebutuhan nutrisi. Pada halaman ini juga *admin* dapat menambahkan status lokasi, *edit* status lokasi, dan hapus status lokasi.



Gambar 3.31. Halaman Tambah Status Lokasi

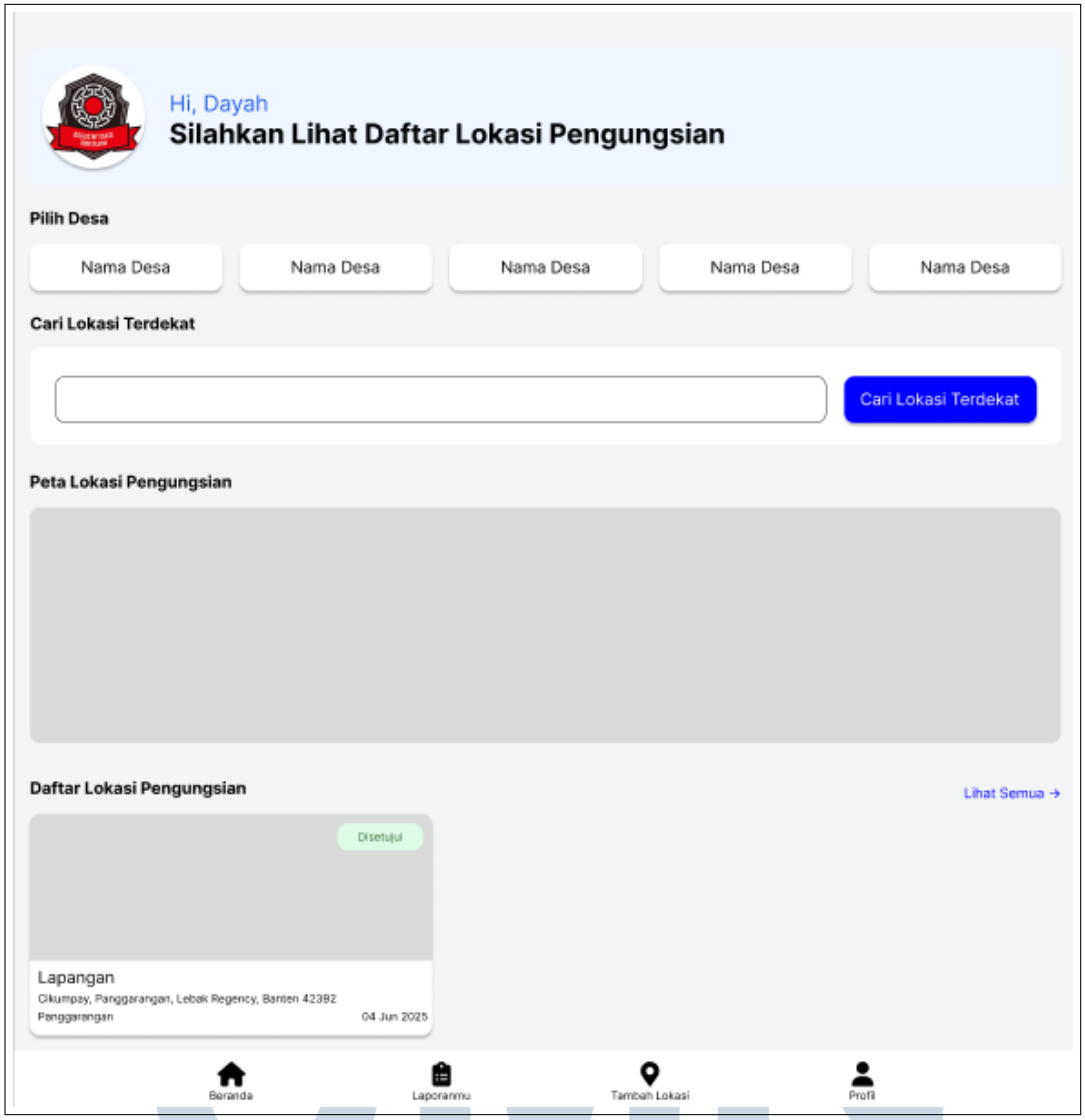
Gambar 3.31 adalah halaman untuk menambahkan status lokasi, secara *default* status lokasi adalah *pending*, kemudian dapat ditambahkan status *approved* dan *rejected*



Gambar 3.32. Halaman Ubah Data Lokasi

Gambar 3.32 adalah halaman untuk mengubah status lokasi yang sudah di

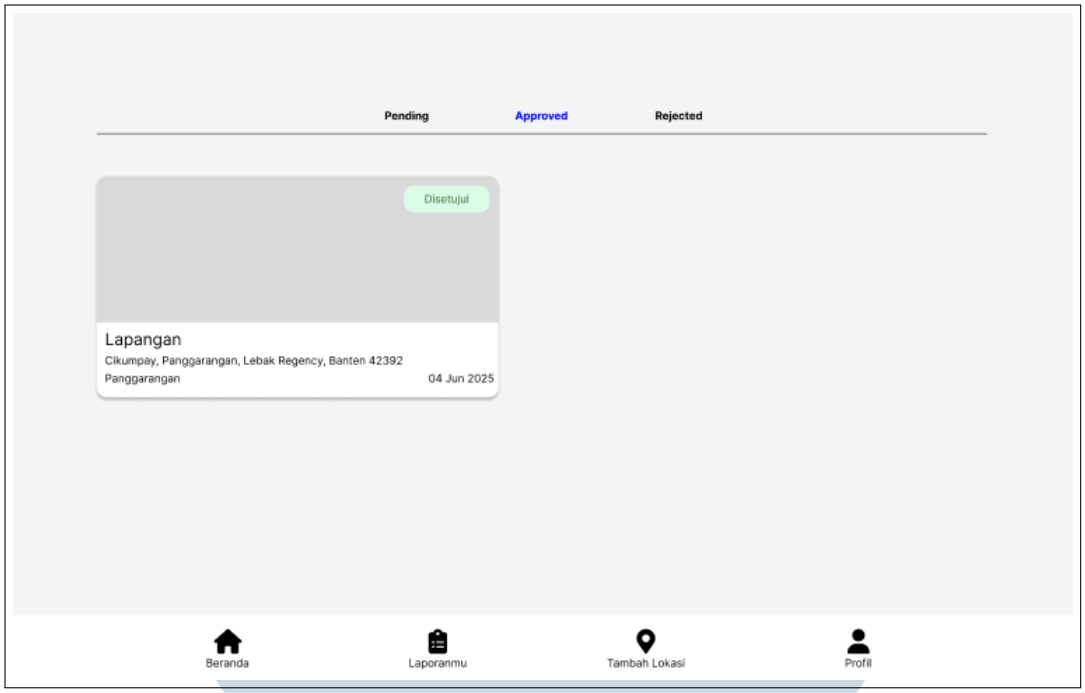
input.



Gambar 3.33. Desain Halaman Beranda Relawan

Gambar 3.33 adalah halaman beranda relawan. Beranda menampilkan informasi data desa dan data lokasi pengugnsian.

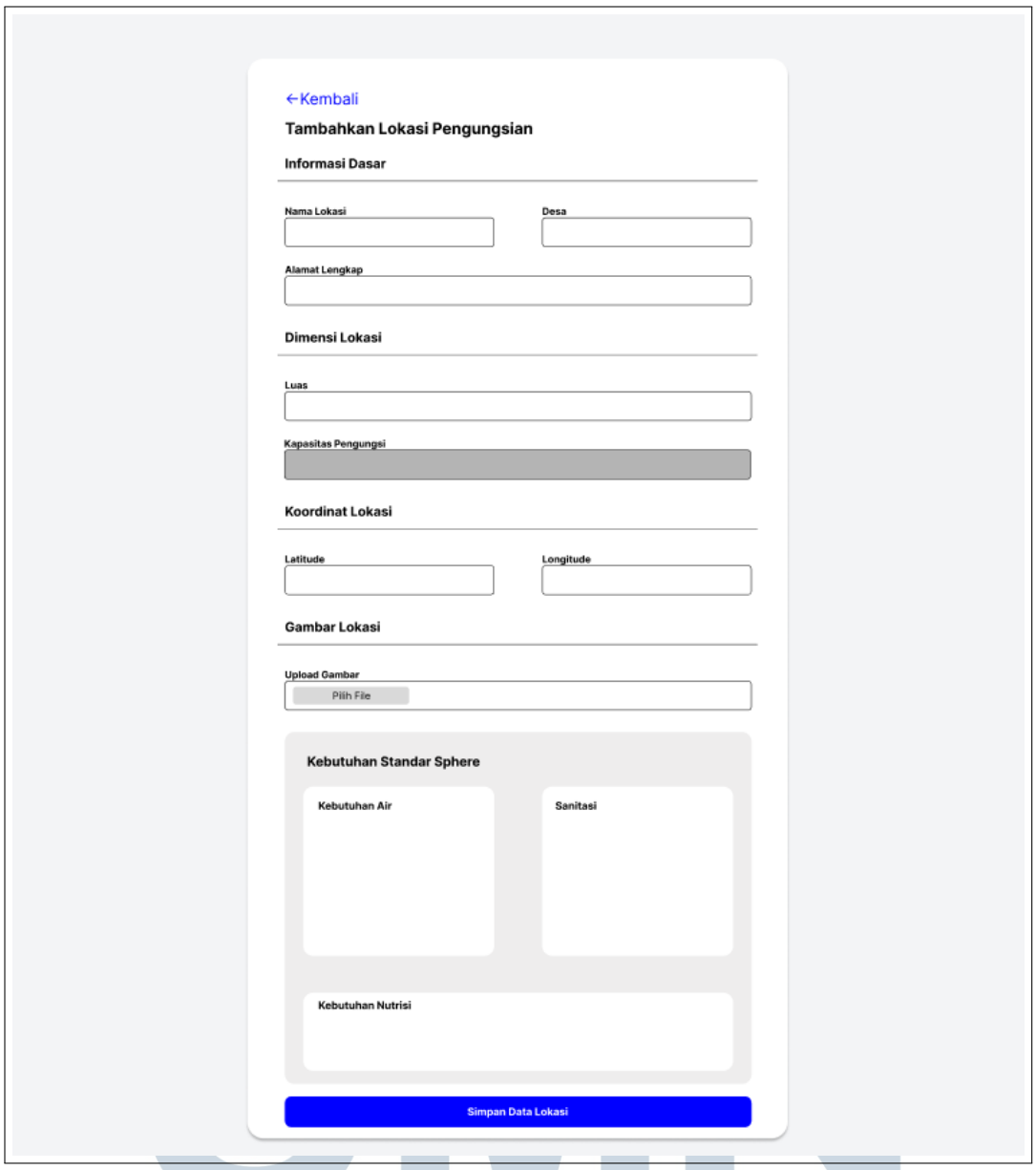
UNIVERSITAS MULTIMEDIA NUSANTARA



Gambar 3.34. Desain Halaman Data Laporan Lokasi Oleh Relawan

Gambar 3.34 adalah halaman laporan lokasi pengungsian oleh relawan. Halaman ini menampilkan daftar lokasi pengungsian yang diajukan oleh relawan. Data lokasi tampil berdasarkan status lokasi.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.35. Desain Halaman Tambah Lokasi Oleh Relawan

Gambar 3.35 adalah halaman tambah data lokasi pengungsian oleh relawan. Halaman ini secara otomatis menghitung kebutuhan standar proyek *sphere* seperti pada halaman *admin* berdasarkan *input* luas lokasi.

3.4.5 *Sprint Review*

Tahap Evaluasi terhadap pengembangan atau *sprint review* akan dilakukan bersama dengan *product owner* atau Ketua GMLS. Proses ini akan dilakukan pada masa akhir *sprint* yang dilakukan setiap dua minggu sekali, maka akan ada empat

kali proses sprint review. *Sprint review* akan dilakukan secara *online* melalui media google meet.

3.4.6 *Restropective*

Tahap Evaluasi terhadap proses kerja atau *restropective* akan dilakukan secara mandiri tanpa *product owner*. Proses ini akan dilakukan pada masa akhir *sprint* setelah proses *sprint review* dilakukan.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA