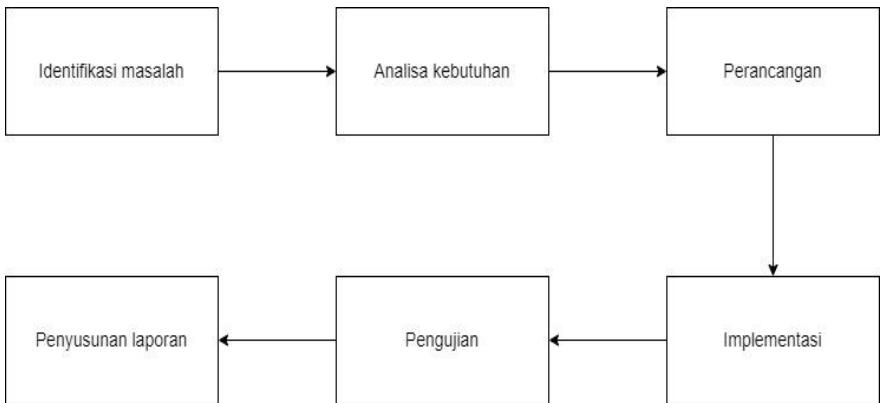


BAB III

METODOLOGI PENELITIAN

3.1. Tahapan penelitian

Berikut adalah alur dari metode penelitian proses Rancang Bangun Sistem Informasi Komunitas Motor Last Limitd Surabaya Berbasis Web.



Gambar 3. 1 Alur Penelitian

Metode pengumpulan data yang akan digunakan pada awalnya untuk membangun suatu sistem adalah dengan mengidentifikasi data yang akan dikumpulkan dari unit motor melalui identifikasi, dilanjutkan dengan penelitian kepustakaan dengan menggunakan referensi teori-teori yang terkait dengan kasus, pengumpulan data melalui observasi, dan analisis data menggunakan

wawancara.setelah itu dilanjutkan dengan merancang sistem dengan menggunakan *Usecase*, implementasi sistem dan yang terakhir pengujian sistem menggunakan *blackbox testing*.

3.2. Identifikasi Masalah

Langkah pertama yang dilakukan adalah mengidentifikasi masalah, yaitu suatu proses yang bertujuan untuk memahami permasalahan yang terdapat pada sistem. Proses identifikasi dilakukan di lokasi penelitian, yaitu komunitas motor last limit Surabaya, dengan menggunakan teknik pengumpulan data seperti observasi dan konsultasi dengan pihak terkait.

3.3. Analisa Kebutuhan

Setelah melakukan tahapan identifikasi masalah maka selanjutnya penulis menganalisa berbagai kebutuhan yang digunakan untuk membuat suatu sistem. Input pada Analisis kebutuhan yaitu pengumpulan data dari referensi yang telah tersedia di dalam buku atau jurnal.

3.3.1 Kebutuhan Fungsional

Mengenai persyaratan fungsional, tujuan utamanya adalah untuk memiliki pemahaman yang jelas tentang proses yang terlibat dalam sistem yang akan dijalankan. Persyaratan ini memiliki poin

tentang informasi yang dihasilkan oleh sistem. Berikut ini adalah berbagai kebutuhan fungsional.

1. Kebutuhan *User*
 - a. *User* dapat melihat jadwal *event* yang akan dilaksanakan pada hari yang akan datang
 - b. *User* dapat melihat hasil dokumentasi setelah acara *event*
 - c. *User* dapat notifikasi *event* yang akan diselenggarakan
 - d. *User* dapat melihat anggota yang aktif ikut acara *event*
 - e. *User* dapat *login* dan *logout website*

2. Kebutuhan Admin
 - a. Admin dapat mengupload *event* dan dokumentasi
 - b. Admin dapat memberi notifikasi *event*
 - c. Admin dapat menginput anggota yang aktif
 - d. Admin dapat *login* dan *logout* pada *website*

3.3.2 Kebutuhan Non fungsional

Tujuan dari persyaratan non-fungsional ini adalah untuk mengidentifikasi berbagai persyaratan sistem. Kebutuhan non-fungsional sama beragamnya dengan kebutuhan itu sendiri. Berikut ini adalah beberapa kebutuhan non-fungsional.

1. Perangkat lunak
 - a. Koneksi internet

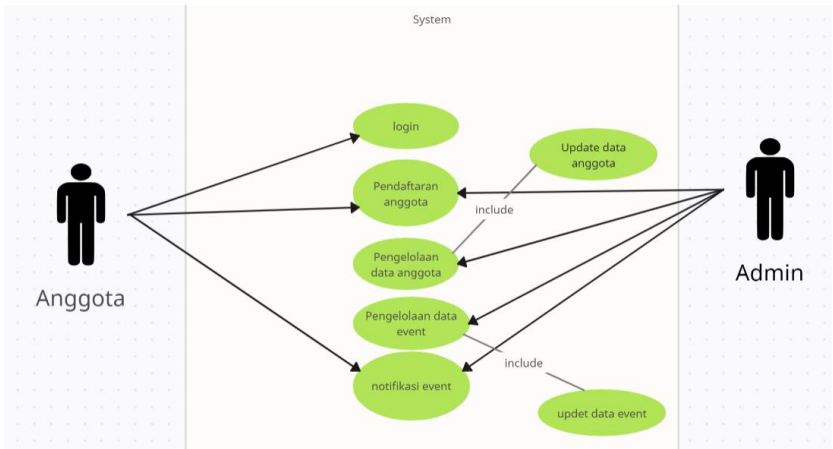
- b. PhpMyAdmin yang akan digunakan untuk *database*
 - c. Adobe visual studio code untuk Pembangunan kode program
2. Kebutuhan perangkat keras
- a. Komputer atau laptop dengan minimal spesifikasi RAM 4GB dan ROM 250GB
 - b. Handphone android atau ios dengan spesifikasi minimal RAM 2GB dan ROM 16GB

3.4. Perancangan sistem

Dalam proses pembuatan sistem ini, penulis membuat situs web komunitas motor kerangka yang berfungsi sebagai model untuk sistem web yang akan dibuat menggunakan data yang dikumpulkan sebelumnya. Alat untuk mengoptimalkan komunitas motor berbasis situs web ini menggunakan *Star UML*.

3.4.1 Usecase diagram

Proses perancangan ini di gambarkan menggunakan Use case *diagram* yang terdapat.

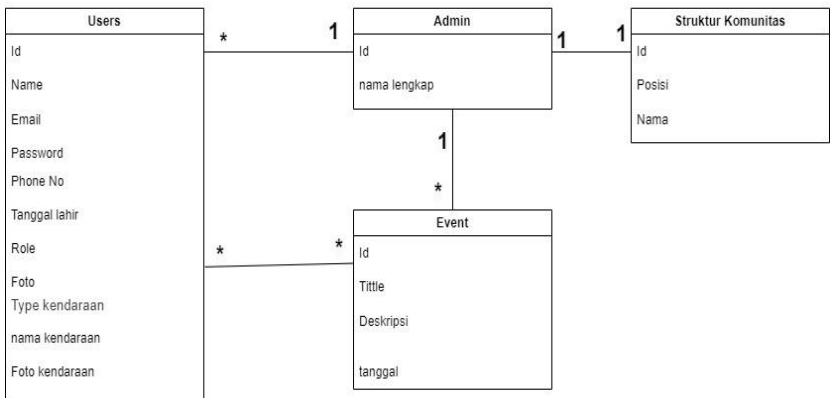


Gambar 3. 2 Usecase Diagram

Gambar diatas menggambarkan *Usecase diagram* dari Sistem Informasi komunitas Last Limitd yang berbasis *web*, pada usecase di atas terdapat tiga entitas yaitu Admin, Anggota, dan server atau sistem, terdapat juga atribut dalam usecase tersebut. Pertama-tama entitas Anggota masuk ke *login* dan memasukkan identitas anggota, setelah itu sistem memberikan akses ke anggota untuk dapat masuk home atau beranda sistem. Admin dapat *login* kemudian memasukan berbagai data anggota dan data kendaraan anggota. Lalu sistem *website* memasukkan informasi informasi dari admin ke seluruh anggota *website* dan adanya fitur sistem pengingat atau disebut notification.

3.4.2 Class Diagram

Diagram Kelas untuk menggambarkan hubungan antar kelas yang sering berhubungan dalam suatu sistem. Gambar 4.1 mengilustrasikan hubungan tersebut. composite one to many antara *users* dengan *admin* , relasi composite one to many antara *admin* dengan *users*, relasi one to many antara *admin* dengan struktur data, relasi one to many antara *admin* dengan *event*.



Gambar 3. 3 Class Diagram

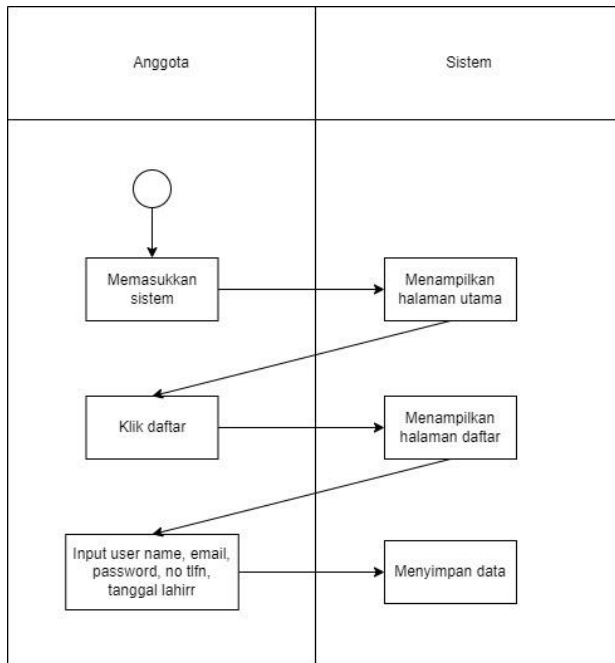
3.5 Activity Diagram

pada *activity diagram* ini menampilkan rancangan untuk *website*, yang ada dibawah ini.

3.5.1 Registrasi

Dalam register diagram aktivitas, pengguna akan berinteraksi dengan sistem dengan memasukkan nama

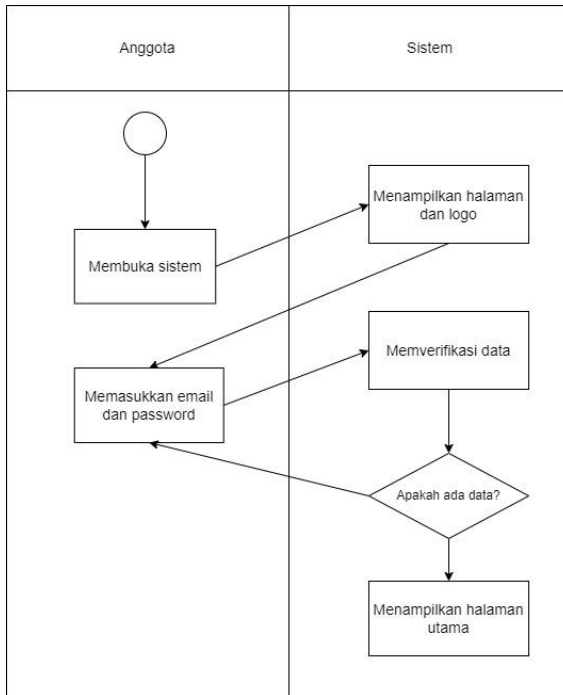
lengkap, alamat email, kata sandi, dan nomor telepon seperti gambar 3.4



Gambar 3. 4 Register

3.5.2 Login

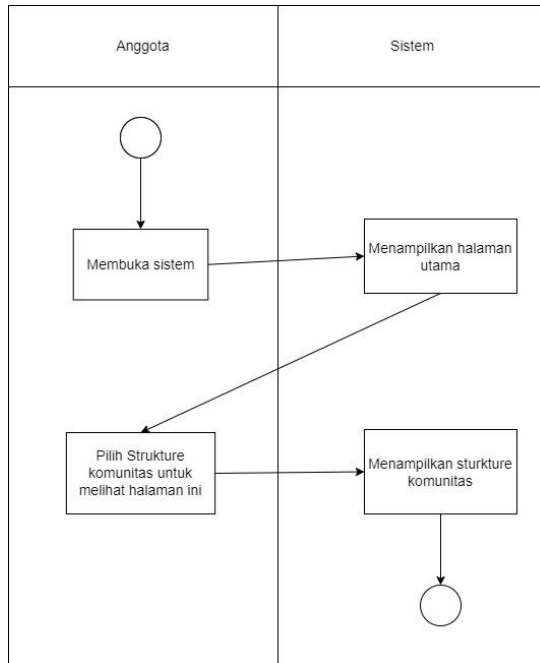
Dalam diagram aktivitas login, pengguna akan berinteraksi dengan sistem untuk memasukkan alamat email dan kata sandi mereka. Sistem kemudian akan memverifikasi apakah data yang dimasukkan sesuai dengan contoh atau tidak. 3.5



Gambar 3. 5 Login

3.5.3 Melihat Struktire komunitas

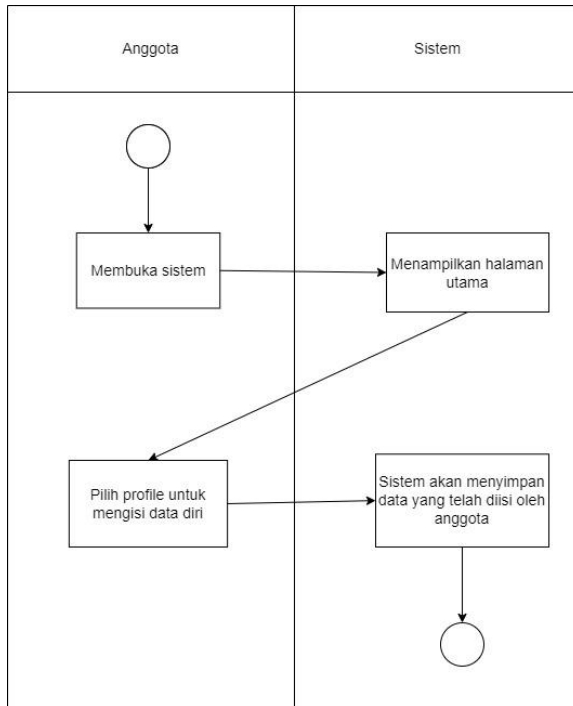
Pada *activity diagram* melihat struktur komunitas, *user* akan berinteraksi dengan sistem, maka akan menampilkan halaman stuktire komunitasseperti pada gambar 3.6



Gambar 3. 6 Melihat Strukture Komunitas

3.5.4 Melihat Data Diri User

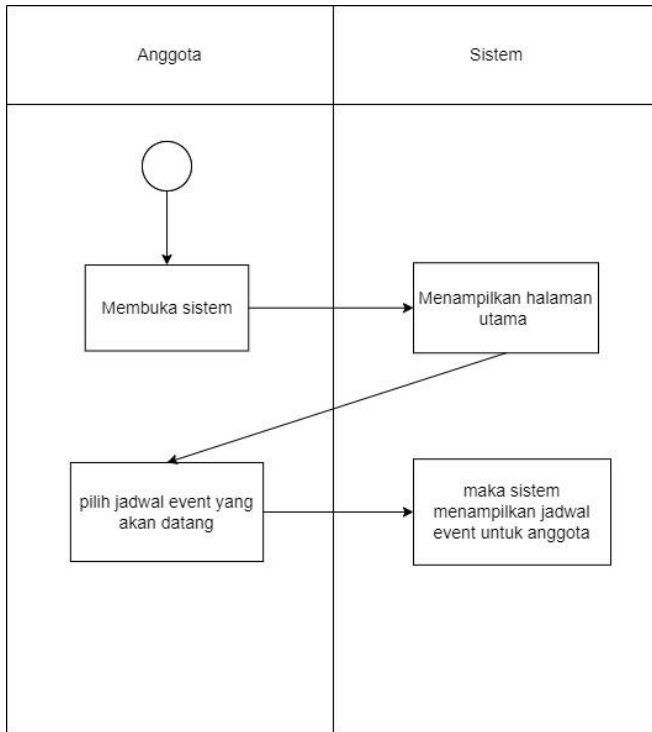
Pada *activity diagram* melihat dan memasukkan data diri, *user* akan berinteraksi dengan sistem untuk menginputkan Nama, Email, No.telephone, Tanggal lahir, Jenis kendaraan, Nama kendaraan, dan Foto kendaraan seperti pada gambar 3.7



Gambar 3. 7 Melihat Data Diri *User*

3.5.5 Melihat Jadwal Event

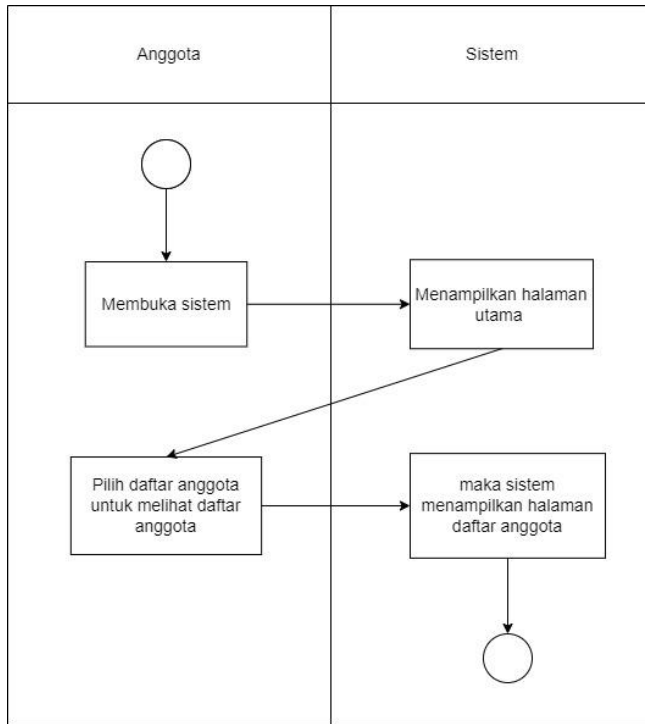
Pada *activity diagram* melihat jadwal *event*, *user* akan berinteraksi dengan sistem. Jika admin yang masuk maka jadwal *event* yang terbaru seperti pada gambar 3.8



Gambar 3. 8 Melihat Jadwal Event

3.5.6 Melihat Daftar Anggota

Pada *activity diagram* ini untuk cara melihat daftar anggota, *user* akan berinteraksi dengan sistem seperti pada gambar 3.9



Gambar 3. 9 Melihat Daftar Anggota

3.6. Implementasi

Pada tahapan penulis melakukan implementasi rancangan ke dalam bentuk *website* atau sistem informasi, PHP digunakan dalam pengembangan sistem ini. dan berbagai perangkat lunak seperti *adobe visual studio code*, *xampp*, dan browser.

Sistem ini menampilkan informasi pada platform *website* tentang jadwal *event*, dokumentasi, anggota yang aktif dan notifikasi. Data dinput oleh admin dan setiap pengguna memperoleh informasi.

3.7. Penguji sistem

Pada langkah ini, setiap aktivitas yang digunakan dianalisis untuk melakukan evaluasi, mengidentifikasi masalah pada sistem yang dimaksud, dan kemudian memperbaikinya. Pengujian sistem perangkat dengan tujuan mengamati output sistem yang dibuat melalui pengujian blackbox.

3.8. Penyusunan Laporan

Selama proses penulisan, penulis mencoba merangkum hasil penelitian yang telah dilakukan secara jujur sehingga dihasilkan narasi berbasis naska..