BAB I
PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi informasi sekarang ini berkembang sangat pesat. Dengan adanya perkembangan teknologi ini, kinerja manusia bisa menjadi lebih efektif dan efisien. Perkembangan teknologi ini terjadi di hampir semua aspek kehidupan, salah satunya adalah dalam bidang komunikasi. Saat ini, semua informasi yang dibutuhkan bisa didapatkan melalui media ponsel. Hal ini karena fungsi ponsel yang bisa digunakan di mana saja dan kapan saja. Penggunaan ponsel lebih praktis, karena ukurannya yang kecil dan mudah dibawa membuat user bisa mencari informasi yang diinginkan dengan cepat dan mudah (Desan, dkk., 2020).

Kota Palopo adalah bagian dari wilayah Luwu Raya, yang terdiri dari beberapa kabupaten, seperti Kabupaten Luwu Utara, Luwu Timur, Luwu, dan Kota Palopo itu sendiri. Jika dilihat dari sisi sejarah, awalnya wilayah Luwu Raya adalah satu wilayah *admin*istratif, yaitu Kabupaten Luwu yang luasnya hampir mencapai 30 persen dari total luas Provinsi Sulawesi Selatan (termasuk juga Sulawesi Barat), yaitu sekitar 28,47 persen. Karena adanya tuntutan dan keinginan masyarakat untuk membangun daerahnya ke arah yang lebih maju, pada tahun 1999, wilayah ini terpecah menjadi dua kabupaten, yaitu Kabupaten Luwu dan Kabupaten Luwu Utara. Pada tahun 2002, sesuai dengan Undang-Undang No. 11 Tahun 2002, Kota Palopo secara resmi dibentuk sebagai pecahan dari Kabupaten Luwu. Kemudian, pada tahun 2003, terjadi pemekaran di Luwu Utara, dan terakhir terbentuklah Kabupaten Luwu Timur sebagai pecahannya (Yunus, 2023).

Kendaraan bermotor merupakan salah satu alat transportasi yang paling banyak digunakan masyarakat Kota Palopo. Hal ini menyebabkan kebutuhan akan layanan tambal ban juga meningkat. Tambal ban merupakan salah satu layanan yang diperlukan untuk mengatasi ban bocor atau rusak. Kebutuhan akan layanan tambal ban di Kota Palopo dapat dilihat dari banyaknya tambal ban yang tersebar di berbagai lokasi di kota ini. Namun, masih banyak masyarakat yang kesulitan untuk menemukan tambal ban terdekat, terutama di saat kondisi darurat.

Tukang tambal ban atau teknisi tambal ban adalah profesi yang sangat

dibutuhkan oleh para pengendara motor, khususnya saat mereka mengalami ban bocor di jalan. Ketika ban motor tiba-tiba kempes, pengendara biasanya akan langsung mencari tukang tambal ban terdekat di sepanjang jalan yang dilewati. Jika tempat tambal bannya jauh, pengendara akan terpaksa mendorong motornya sampai menemukan papan tanda tambal ban. Semakin jauh jarak tambal bannya, semakin lelah pula pengendara untuk mendorong motornya.

Atas dasar itu, penulis berinisiatif untuk membuat sebuah *Website* yang diberi nama tambalin. *Website* ini bertujuan untuk memudahkan para pengendara motor dalam mencari lokasiteknisi tambal ban terdekat dengan bantuan *Google Maps API*. Di sisi lain, teknisi tambal ban juga dapat melihat lokasi pengendara yang bannya bocor melalui panggilan yang dilakukan menggunakan *Google Maps API.*

Google Maps API adalah sekumpulan fungsi pemrograman yang disediakan oleh *Google Maps* agar dapat diintegrasikan ke dalam *Website.* Contoh sederhananya, jika ingin membuat Sistem Informasi Geografis (*SIG*), kita bisa memanfaatkan *Google Maps API* tanpa perlu membuat peta sendiri, cukup menggunakan *Google Maps* dan memanggil fungsi-fungsi yang dibutuhkan, seperti menampilkan peta, menempatkan marker, dan lain-lain.

Berdasarkan uraian di atas, penulis mengangkat topik ‘Perancangan *Website* Tambalin Berbasis *Web* untuk Mencari lokasiTeknisi Tambal Ban di Kota Palopo’. Diharapkan, *Website* ini dapat membantu pengendara motor dalam mencari lokasi teknisi tambal ban terdekat.

1.2 Rumusan Masalah

Berangkat dari latar belakang yang sudah dibahas sebelumnya, rumusan masalah dalam penulisan ini adalah bagaimana caranya merancang dan membangun *Website* Tambalin untuk mempermudah pencarian lokasi tambal ban di Kota Palopo?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk merancang *Website* pencarian lokasi teknisi tambal ban sehingga mempermudah para pengendara jika mengalami kebocoran ban pada kendaraannya.

1.4 Manfaat Penelitian

Manfaat yang dapat diambil dari perancangan *Website* Tambalin berbasis *android* adalah:

1. Bagi Penulis Dengan mangadakan penelitian ini dan pembuatan sistem ini, maka penulis mempunyai kesempatan untuk mengetahui permasalahan yang sering dihadapi oleh pengendara motor.
2. Bagi Almamater Hasil penelitian ini dapat digunakan sebagai referensi untuk penelitian berikutnya untuk bidang yang sama dan dapat diharapkan dapat menambah pengetahuan serta dapat dijadikan sebagai studi perbandingan di masa yang akan datang.
3. Bagi Masyarakat Untuk memberikan solusi mudah untuk mencari dan memanggil teknisi tambal ban melalui *smartphone Android*.

1.5 Batasan Penelitian

Adapun batasan dari penelitian ini adalah:

1. Pembangunan sistem berbasis *Website* menggunakan *software visual studio code* dengan bahasa pemrograman *Java Script, HTML*, *PHP* dan basis data *MySQL*.
2. Jasa tambal ban tersedia pada lingkup area cakupan daerah Kota Palopo, Sulawesi Selatan.
3. Informasi yang disediakan khusus tambal ban sepeda motor yang ada pada kota palopo.
4. Perancangan *UML* menggunakan *draw.io* dan Rancangan *Interface*nya Menggunakan *Software Balsamiq.*
5. *Website* Tambalin hanya memberikan informasi penyediaan jasa tambal ban tanpa mengelola transaksi antara pelanggan dan penyedia jasa

BAB II
TINJAUAN PUSTAKA

2.1 Kajian Teori

Kajian Teori adalah penegasan landasan teori yang dipilih penulis dalam penelitian ini. Teori-teori digunakan dalam penelitian ini dijelaskan telah lengkap pada sub-sub pokok bahasan berikut.

1. Rancang Bangun

Rancang bangun adalah proses menggambarkan, merencanakan, dan menyusun beberapa elemen yang terpisah menjadi satu kesatuan yang lengkap dan berfungsi dengan baik. Jadi, bisa dibilang rancang bangun adalah kegiatan yang menerjemahkan hasil analisis menjadi paket perangkat lunak, lalu membangun sistem baru atau memperbaiki sistem yang sudah ada (abdul rauf, 2021).

Rancang bangun juga bisa dipahami sebagai penyusunan dan perencanaan yang melibatkan berbagai elemen menjadi satu kesatuan yang utuh dan bisa berfungsi. Jadi, rancang bangun berarti menerjemahkan hasil analisis ke dalam perangkat lunak untuk menciptakan sistem baru atau memperbaiki yang sudah ada (Gunawan, 2021).

Berdasarkan penjelasan di atas, rancang bangun dapat dimaknai sebagai proses menciptakan sesuatu dari awal. Proses ini dimulai dengan memikirkan apa yang dibutuhkan dan diinginkan oleh pengguna, lalu dilanjutkan dengan merancang desain atau gambaran umum mengenai cara kerja dari sesuatu tersebut.

2. *Website*

Menurut Nuh, (2022) *Website* adalah kumpulan halaman web yang saling terhubung dan bisa diakses oleh publik, semuanya berbagi satu nama domain. *Website* bisa dibuat dan dikelola oleh individu, kelompok, bisnis, atau organisasi untuk berbagai tujuan. Namun, memiliki sebuah situs saja belum tentu memberikan manfaat maksimal jika tidak dikelola dengan baik. Isi *Website* perlu diperbarui secara rutin dan berkelanjutan agar kontennya menjadi lebih komunikatif dan informatif. Aktivitas tersebut sebenarnya bisa dilakukan oleh tim internal, termasuk dalam hal penambahan fitur atau fungsi baru sesuai kebutuhan. Berdasarkan sifatnya, *Website* dapat dibedakan menjadi dua jenis, yaitu:"

a. *Website* Dinamis adalah jenis *Website* yang kontennya bisa diperbarui kapan saja. Contoh *Website* dinamis antara lain Wikipedia, Detik, Tokobagus, dan blog terkait internet marketing. Yang membuat *Website* ini bersifat dinamis adalah adanya *Content Management System* (*CMS*). Dengan *CMS*, siapa saja yang memiliki akses sebagai *admin*istrator dapat dengan mudah memperbarui kontennya.
b. *Website* Statis adalah *Website* yang jarang sekali diubah karena tidak membutuhkan pembaruan yang sering. Contohnya adalah *Website* profil perusahaan (*company profile*) dan *Website* profil organisasi. *Website* statis biasanya hanya berfungsi sebagai brosur atau kartu nama digital perusahaan.

3. Tambal Ban

Tambal, menurut Kamus Besar Bahasa Indonesia, artinya adalah memperbaiki sesuatu, seperti rumah atau ban, tapi tidak seluruhnya, hanya bagian yang rusak saja. Sementara itu, ban adalah benda bulat dari karet yang dipasang di roda, seperti di motor atau sepeda. Jadi, tambal ban itu kegiatan memperbaiki ban yang bocor, hanya di bagian yang rusak, bukan seluruhnya" (Pribadi & Istiqomah, 2021).

4. *Global Positioning System (GPS)*

GPS itu adalah sistem navigasi yang memanfaatkan satelit untuk memberikan informasi tentang posisi, kecepatan, dan waktu di hampir semua tempat di bumi, kapan saja, dan dalam kondisi cuaca apa pun. Awalnya, *GPS* ini hanya digunakan untuk kepentingan militer, tapi sejak tahun 1980, masyarakat sipil sudah bisa menggunakan *GPS* ini secara gratis dan bebas. Ada lima fungsi dan kegunaan *GPS* yang bisa kita bagi seperti berikut:

a. *GPS* untuk Militer Membantu mendukung sistem pertahanan militer dan digunakan untuk memantau pergerakan musuh saat terjadi perang.
b. *GPS* untuk Navigasi Membantu berkendara dalam menentukan kemana arah dan tujuan perjalan yang bisa di lihat dalam *GPS*.
c. *GPS* untuk Sistem Informasi Geografis Digunakan untuk keperluan sistem informasi geografis, seperti pembuatan peta, mengukur jarak perbatasan, atau dijadikan referensi pengukuran suatu wilayah.
d. *GPS* untuk Sistem Pelacakan Kendaraan Sistem pelacakan adalah alat penampil dan penerima sinyal berbeda lokasi. Contohnya bsa mengetahui lokasi

kendaraan yang hilang dengan melihat titik koordinat yang di hasilkan dari alat terpasang di kendaraan.

e. *GPS* untuk Pemantau Gempa Tengologi *GPS* yang ditingkatkan menghasilkan tingkat ketelitian dan keakuratan yang tinggi sehingga *GPS* dimanfaatkan untuk memantau pergerakan tanah dibumi, bisa dilihat pada Gambar 1.



Gambar 1. *GPS* Untuk Pemantau Gempa
Sumber: *GPS* untuk Pemantau Gempa oleh Rahmah Muthia (2022).

5. Sistem Informasi Geografi (*SIG*)

Sistem Informasi Geografis (*SIG*) itu adalah sistem yang fokus pada informasi tentang geografi. Istilah 'geografis' ini berkaitan dengan hal-hal yang berhubungan dengan ruang atau lokasi. Seringkali, istilah ini dipakai secara bergantian dengan istilah lain, sampai muncul istilah baru yaitu geospasial. Ketiga istilah ini punya makna yang sama dalam konteks *SIG*. Ketika kita bilang 'geografis,' itu merujuk pada masalah yang terkait dengan permukaan bumi, baik dua atau tiga dimensi. Sedangkan 'informasi geografis' berarti informasi tentang tempat-tempat di permukaan bumi, termasuk pengetahuan tentang posisi objek di bumi dan keterangan lain yang ada di permukaan bumi yang sudah diketahui posisinya (Alnast, 2021)

6. *Google Application Programming Interface* (*API*)

API itu bisa dibilang sebagai kode program yang berfungsi sebagai penghubung antara *Website* yang kita buat dengan berbagai fungsi yang tersedia. Contohnya, Google *API* adalah kode program yang bisa kita tambahkan ke *Website* kita untuk mengakses atau menggunakan fitur yang disediakan oleh Google.

Misalnya, kita bisa menambahkan fitur *Google Map* ke *Website* kita. *API* juga mencakup fungsi-fungsi pemrograman yang disediakan oleh *Website* atau layanan supaya layanan tersebut bisa terintegrasi dengan *Website* yang kita buat" (Wahyuni, & Sopiandi, 2020).

Jadi, *Google Maps API* adalah fungsi-fungsi pemrograman dari Google Maps yang memungkinkan kita mengintegrasikan *Google Maps* ke dalam *Website* yang kita bangun. Contoh sederhana, jika kita mau membuat sistem informasi geografis untuk kampus di Jogja, dengan menggunakan *Google Maps API*, kita bisa bikin *GIS* tanpa perlu repot memikirkan peta Jogja. Kita tinggal pakai *Google Maps* dan memanggil fungsi-fungsi yang dibutuhkan, seperti menampilkan peta, menempatkan marker, dan lain-lain. *Google Maps API* sendiri menyediakan banyak sekali fungsi, berikut adalah pembagiannya :

c. Google maps *Java Script API* v3 Ini adalah *Google Maps API* yang akan penulis pelajari. embel embel *Java Script* menunjukan *API* ini tersedia dalam bahasa *Java Script.*
d. *Google Maps API Webservice* Layanan *Google Maps API* untuk fungsi lanjutan seperti *direction, Geocoding, Distance Matrix API* dan *elevation API*.
e. *Google Place API* Ini adalah *API* wajib yang harus dikuasai untuk membuat *Website Location based service*.

Tiga *API* diatas adalah *API* yang paling penting untuk penulis pelajari, sebenarnya masih ada beberapa *API* lain seperti *Google Earth API* dan *Google Maps Image API* namun untuk saat ini *API* ini tidak penulis pelajari.

Website jenis apa yang bisa dibuat dengan *Google Maps API*? Semua *Website* yang membutuhkan layanan peta. *Website* tidak terbatas pada *Website,* namun juga bisa *Website* desktop ataupun *Website* di smartphone bahkan *Website* di embeded sistem misalkan *Website* navigasi di mobil. Berikut ini adalah gambaran ide yang bisa diterapkan diterapkan

a. Penulis ingin membuat *Website review* Kuliner, *user* bisa memasukan nama rumah makan, merating kualitas makanannya dan menginput lokasinya di peta.
b. Penulis bisa membuat *Website GIS* berbasis *Desktop* dan dengan mudah mengintegrasikan dengan *Google Maps API* untuk memvisualisasikan data, misalkan visualisasi peta kemiskinan di propinsi X.

c. Penulis bisa membuat *Website* di *Mobile* (*Android*) untuk menampilkan informasi hotel diurutkan berdasarkan lokasi terdekat dengan penulis dan petunjuk jalan menuju Hotel tersebut.
d. Penulis bisa membuat *Website tracking* taksi sehingga manajer perusahaan taksi bisa melihat secara *realtime* posisi setiap taksi yang beroperasi. Dan ide ide lain yang masih berhubungan dengan lokasi, tempat atau peta bisa memanfaatkan *Google Maps API* untuk meralisasikannya.

7. *Visual Studio Code (VS Code)*

Visual Studio Code (VS Code) adalah editor kode yang dikembangkan oleh *Microsoft*. Editor ini dirancang agar memudahkan para pengembang perangkat lunak dengan berbagai fitur yang lengkap, tapi tetap ringan. *VS Code* bisa digunakan untuk berbagai bahasa pemrograman dan alat pengembangan, sehingga jadi salah satu alat yang paling populer di kalangan komunitas pemrograman" (Kusuma, 2024).

8. *MySQL*

MySQL adalah perangkat lunak yang termasuk dalam sistem manajemen basis data SQL atau yang biasa kita sebut *DBMS*. *MySQL* ini bersifat multithread dan bisa digunakan oleh banyak pengguna secara bersamaan, dengan sekitar 6 juta orang yang menyimpannya di seluruh benua. *MySQL* ada di bawah lisensi *GPL* atau *General Public License*, tetapi bisa juga dijual secara komersial bagi pengguna yang tidak cocok dengan ketentuan lisensi tersebut" (Siregar, dkk., 2024).

9. *HTML*

Menurut Nasution, (2022) *HTML* atau *Hypertext Markup Language* adalah bahasa yang digunakan untuk menulis halaman web. Fungsi utamanya adalah memberi perintah kepada browser untuk mengatur tampilan menggunakan berbagai tag yang ada di *HTML*. Dengan *HTML*, pengguna bisa membuat dan menyusun bagian paragraf, heading, tautan, atau blockquote untuk halaman web dan aplikasi. *HTML* juga merupakan bahasa yang digunakan untuk menggambarkan struktur sebuah halaman web dan mempublikasikan dokumen secara online" (Agung Noviantoro, dkk., 2022).

Dari penjelasan di atas, penulis menyimpulkan bahwa *HTML* adalah bahasa pemrograman standar untuk menampilkan halaman web

10. *PHP*

Menurut Sandria, (2022) *PHP* adalah singkatan dari *Hypertext Preprocessor* yaitu suatu bahasa scripting tingkat tinggi yang dipasang pada dokumen *HTML*. Secara dominan, sintaks dalam *PHP* mirip dengan bahasa C, *Java* dan *Perl*, namun pada *PHP* ada beberapa fungsi yang lebih spesifik. Sedangkan tujuan utama dari penggunaan bahasa ini adalah untuk memungkinkan perancang *web* yang dinamis dan dapat bekerja secara otomatis.

Dari pengertian diatas penulis menyimpulkan bahwa *PHP* merupakan bahasa pemrograman untuk pembuatan *Website* yang ditanamkan dalam *HTML*

11. *Javascript*

Javascript adalah bahasa pemrograman tingkat tinggi yang sangat penting dalam pengembangan aplikasi web modern. Diciptakan oleh Brendan Eich pada tahun 1995, *Javascript* awalnya dibuat untuk menambahkan interaktivitas dan dinamika pada halaman web. Bahasa ini dijalankan di sisi klien, artinya kode *Javascript* langsung dijalankan di browser pengguna. Ini memungkinkan pengalaman pengguna yang responsif dan interaktif tanpa perlu memuat ulang seluruh halaman (Wali, dkk., 2023).

Salah satu hal utama dari *Javascript* adalah kemampuannya untuk berinteraksi dengan *Document Object Model* (*DOM*). *DOM* itu adalah representasi struktur halaman web, di mana elemen seperti teks, gambar, dan form diatur dalam hierarki. Dengan menggunakan *Javascript*, pengembang bisa memanipulasi elemen-elemen ini secara langsung, sehingga mereka bisa menambah, menghapus, atau mengubah konten dan gaya elemen di halaman *web* secara dinamis. Ini memungkinkan pengembang untuk menciptakan berbagai fitur interaktif, seperti validasi formulir, animasi, dan reaksi terhadap aksi pengguna, seperti klik dan hover.

Javascript tidak hanya berfungsi untuk menciptakan interaksi di sisi klien, *API* juga sekarang bisa digunakan di sisi server. Dengan munculnya *Node.js*, pengembang bisa memakai *Javascript* untuk menangani pemrograman sisi server. *Node.js* adalah lingkungan runtime yang memungkinkan *Javascript* dijalankan di luar browser. Dengan *Node.js*, pengembang bisa membangun aplikasi server yang mampu menangani permintaan *HTTP*, mengelola basis data, dan file. Keberadaan *Node.js* membuka peluang baru bagi pengembang untuk menggunakan satu bahasa

pemrograman di seluruh bagian aplikasi, dari *frontend* hingga *backend*.

Javascript juga dikenal karena dukungannya terhadap berbagai pustaka dan *framework* yang meningkatkan produktivitas pengembangan. Pustaka seperti *jQuery* memudahkan manipulasi *DOM* dan penanganan acara, sementara *framework seperti React, Angular, dan Vue.js* memberikan struktur yang lebih kuat untuk membangun aplikasi *web* yang kompleks. *Framework* ini memudahkan pengembang untuk menciptakan aplikasi yang lebih terorganisir, mengelola kode dengan lebih baik, dan meningkatkan pengalaman pengguna dengan antarmuka yang lebih responsif.

Kelebihan *Javascript* adalah kemampuannya berfungsi di berbagai *platform* dan perangkat. Tidak seperti banyak bahasa pemrograman lain, *Javascript* tidak memerlukan proses kompilasi untuk dijalankan kode-nya langsung ditafsirkan oleh browser. Ini membuatnya sangat cepat dalam eksekusi, ideal untuk aplikasi web interaktif yang memerlukan respon cepat terhadap interaksi pengguna. Selain itu, *Javascript* memiliki komunitas besar dan aktif, dengan banyak sumber daya, dokumentasi, dan alat untuk membantu pengembang menyelesaikan masalah dan memperluas fungsionalitas.

Javascript terus berkembang, dengan pembaruan rutin yang menghadirkan fitur baru dan perbaikan. Versi terbaru dari *Javascript*, yang dikenal sebagai *ECMAScript*, memperkenalkan banyak inovasi yang meningkatkan kemampuan bahasa ini. Beberapa fitur modern termasuk pemrograman *asynchronous* dengan promises dan *async/await*, pengenalan kelas, dan modul, semua ini membuat pengembangan dengan *Javascript* menjadi lebih efisien dan terorganisir.

Secara keseluruhan, *Javascript* adalah bahasa pemrograman yang sangat kuat dan fleksibel, yang tidak hanya penting dalam pengembangan web, *API* juga semakin populer di berbagai aplikasi perangkat lunak. Dengan kemudahan penggunaan, dukungan luas, dan kemampuan untuk menciptakan aplikasi yang kaya dan interaktif, *Javascript* akan terus menjadi salah satu bahasa pemrograman utama di dunia pengembangan perangkat lunak.

12. *Unified Modeling Language* (*UML*)

Analisis dan perancangan pemodelan untuk sistem ini dilakukan dengan menggunakan pola *UML*. *UML* berfungsi untuk membantu dalam mendeskripsikan

dan merancang sistem perangkat lunak, terutama yang dibangun dengan pemrograman berorientasi objek. Penggunaan *UML* ini tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya, *UML* paling banyak digunakan dalam metodologi berorientasi objek (Nistrina & Sahidah, 2022).

Ada beberapa jenis diagram dalam Unified Modeling Language (*UML*), salah satunya adalah use case diagram yang berfungsi untuk mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibangun, serta dapat menggambarkan fungsi-fungsi yang ada dalam sistem informasi tersebut (Nistrina & Sahidah, 2022).

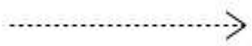
Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem dalam perangkat lunak. Perbedaan antara use case diagram dan activity diagram adalah use case diagram menunjukkan bagaimana aktor berinteraksi dengan sistem untuk melakukan aktivitas, sedangkan activity diagram menggambarkan aktivitas sistem itu sendiri, bukan apa yang dilakukan oleh aktor (Nistrina, 2022).

Sebagai alat bantu pemodelan dalam pengembangan sistem, *UML* memiliki beberapa diagram yang dapat membantu pengembang dalam mengkomunikasikan sistem yang akan dibuat, seperti use case diagram, activity diagram, class diagram, dan sequence diagram.

a. *Diagram Use Case*

Usecase atau *diagram usecase* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat (abdul rauf, 2021).

Tabel 1. *Diagram Usecase*

Simbol	Keterangan
	*Usecase:* Deskripsi dari aksi-aksi yang ditampilkan oleh sistem yang menghasilkan hasil yang terukur bagi seorang aktor.
	Aktor: Menyebutkan kumpulan peran yang dimainkan oleh pengguna saat berinteraksi dengan *use case.*
	Include: Menyatakan bahwa use case sumber secara eksplisit terlibat.
	*Accociaton:* Merupakan apa yang menyanmbungkan dengan objek satu dengan objek lainnya.
	*Extand:* Menspesifikasi kalau *usecase* target meluaskan perilaku dari *usecase* sumber pada titik yang diberikan.
	system Boundary**:** Adalah batasan antara sistem dan aktor, biasanya digambarkan dengan bentuk persegi. Semua use case harus

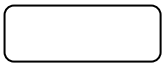
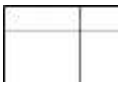
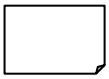
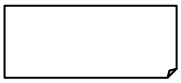
Simbol	Keterangan
	berada di luar batasan sistem ini.

Sumber: Willyana, (2022).

b. Diagram Activity

Sebuah *diagram activity* menggambarkan perilaku dinamis dari sistem atau bagian dari sistem melalui aliran kontrol antara tindakan yang sistem lakukan (abdul rauf, 2021).

Tabel 2. *Diagram Activity*

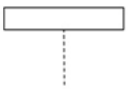
Simbol	Keterangan
	Lingkaran, penuh memulai proses dalam suatu *diagram* aktivitas
	Mata banteng, Akhir dari proses.
	Segiempah panjang. kejadian, aktifitas, atau pemicu.
	Berlian, Sebuah cabang.
	Tabel, suatu *file* komputer dari mana data bisa dibaca atau di rekam selama kejadian bisnis.
	Dokumen, menunjukkan dokumen sumber atau laporan.
	Garis putus-putus, alur informasi antara kejadian.
	Garis tidak putus, urutan dari satu kejadian atau aktifitas ke yang berikutnya.
	Catatan, memberikan acuan bagi pembaca pada *diagram* atau dokumen lain untuk memperinci.

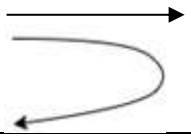
Sumber: Saputra dan Fahriza (2021).

c. *Diagram sequence*

Sequence diagram merupakan *UML* yang menggambarkan interaksi antar objek di dalam dan disekitar sistem, termasuk pengguna, *display*, dan sebagainya berupa *message* yang digambarkan terhadap waktu (Syarif, 2020).

Tabel 3. *Diagram Sequence*

Simbol	Keterangan
	Aktor*:* Seorang yang berinteraksi dengan sistem yang sedang kita kembangkan.
	Objek: Menambahkan objek baru pada *diagram*.
	Aktifitas: Menggambarkan langkah-langkah dalam aliran kerja

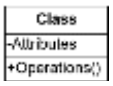
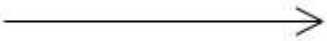
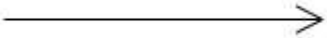
Simbol	Keterangan
	Pesan: Menggambarkan Pesan antara dua objek Pengulangan: Menggambarkan Pesan yang menuju dirinya sendiri

Sumber: Saputra dan Fahriza (2021).

d. *Diagram Class*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sebuah sistem (Syarif 2020)

Tabel 4. *Diagram Class*

Simbol	Keterangan
	Kelas: Pada struktur sistem.
	Antarmuka/*interface*. Sama dengan konsep *interface* dalam pemograman berorientasi objek dalam.
	Relasi. Antar kelas dengan makna umum, asosiasi biasanya yang juga disertai dengan *multiplicity.*
	Asosiasi berarah. Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.*
	*Generalisasi.* Relasi antar kelas dengan makna *generalisasi-generalisasi-spesialisasi* (umum khusus).
	Keberuntungan. Relasi antar kelas dengan makna keberuntungan antar kelas.
	*Agresasi.* Relasi antar kelas dengan makna semua bagian (*whole-part*).

Sumber: Julianto dan Setiawan. (2019).

2.2 Hasil Penelitian yang Relevan

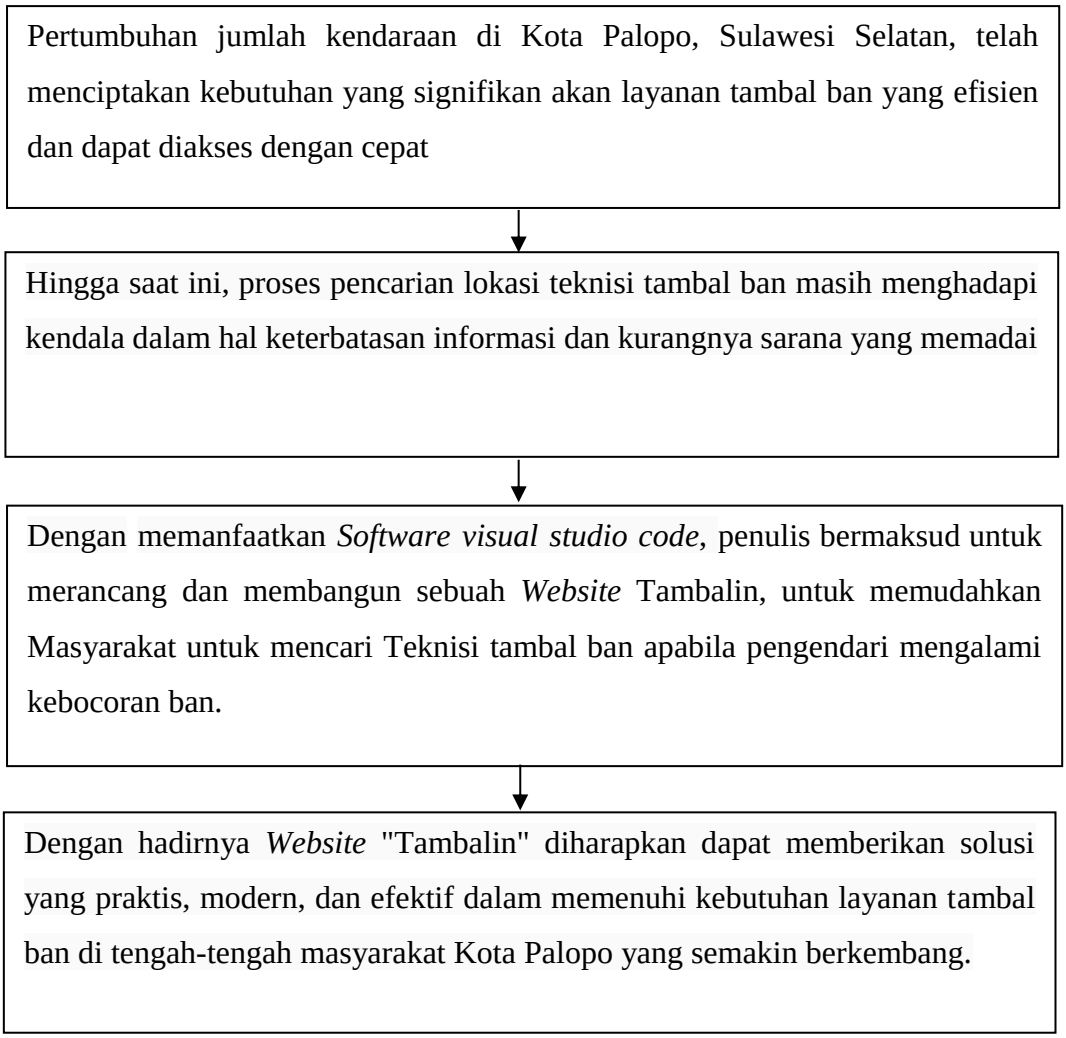
Hasil penelitian yang relavan merupakan penelitian-penelitian terdahulu yang bahan rujukan dari referansi. Penelitian terdahulu merupakan acuan dalam pembuatan Skripsi ini, sehingga sumber referensi diantaranya:

1. Rina, Agung (2015). Dengan judul Rancang Bangun *Website* Pencarian Lokasi Tambal Ban Berbasis *Web*. Penelitian ini membahas tentang rancang bangun *Website* pencarian lokasi tambal ban berbasis web. *Website* ini dibangun untuk memudahkan pengguna dalam mencari lokasi tambal ban di kota Surabaya. *Website* ini menggunakan *Google Maps API* untuk menampilkan data lokasi tambal ban.

2. Arief, dkk., (2016). Dengan judul Rancang Bangun *Website Go*-Ban untuk Mencari lokasiTeknisi Tambal Ban Menggunakan Software Delphi dan *Google Maps API*. Penelitian ini membahas tentang rancang bangun *Website Go*-Ban untuk mencari lokasiteknisi tambal ban. *Website* ini menggunakan *software Delphi* dan *Google Maps API*. *Website* ini dapat digunakan untuk mencari lokasi teknisi tambal ban terdekat dan memanggil teknisi tambal ban untuk datang ke lokasi pengguna.
3. Ilham, dkk., (2020). Dengan judul Rancang Bangun *Website* Tambal Ban Berbasis Android Menggunakan *Google Maps API.* Penelitian ini membahas tentang rancang bangun *Website* tambal ban berbasis Android menggunakan *Google Maps API. Website* ini dapat digunakan untuk mencari lokasi tambal ban terdekat, melihat informasi tentang tambal ban, dan memesan jasa tambal ban.

2.3 Kerangka Pikir

Website Tambalin, dikembangkan menggunakan bahasa pemrograman *Javascript* dan menggunakan *Software Visual studio code* untuk membuat *website* Tambalin, bertujuan memberikan solusi praktis bagi pemilik kendaraan di Kota Palopo dalam menemukan teknisi tambal ban. *Website* ini menggunakan *database MySQL* untuk menyimpan informasi teknisi, serta data pengguna. Fitur utama *Website* melibatkan pencarian teknisi berdasarkan lokasi, pemanggilan langsung melalui platform, dan sistem penilaian dengan ulasan pengguna. Arsitektur *website* mencakup antarmuka pengguna yang responsif dan sistem manajemen yang terintegrasi. Keamanan data dan otentikasi dua faktor diimplementasikan untuk melindungi informasi sensitif. Dengan fokus pada penggunaan teknologi *Javascript* dan *MySQL*, pengembangan *website* ini diharapkan dapat memberikan layanan tambal ban yang efektif dan efisien bagi pengguna di Kota Palopo. Untuk itu memperjelas permasalahan yang di sajikan, maka kerangka dan pikir akan di uraikan sebagai berikut.



Gambar 2. Kerangka Pikir.