

BAB I PENDAHULUAN

1.1 Latar Belakang

Seiring dengan berbagai bentuk mobilitas yang dihadapi dengan berbagai bentuk transportasi, kendaraan semakin diperlukan setiap hari yang berlalu di Indonesia. Secara indikatif, kendaraan berharga pada tingkat mobilitas individu, produktivitas, dan bahkan gaya hidup. Namun, nilai harus diselaraskan dengan kapasitas finansial individu, dan oleh karena itu, banyak individu membeli kendaraan yang sudah dimiliki sebelumnya. Data dari Gaikindo menunjukkan bahwa, kendaraan bekas bahkan lebih mendominasi di kota-kota Jawa yang mahal seperti Jakarta, Medan, dan Surabaya.

Melakukan pembayaran secara cicilan adalah pilihan yang diambil banyak individu karena banyak kendaraan dibeli dengan kredit. Seorang pembeli tetap harus menghadapi banyak sistem kredit yang terkadang tidak perlu kompleks dan pekerjaan administratif di usia pemula. Akhirnya, dalam hal memberikan penghasilan, mereka membutuhkan sistem dalam hal layanan dan mata uang yang dapat dan proporsional terhadap kendaraan yang akan dikredit.

Sebagian besar layanan simulasi kredit yang ada tidak membangun estimasi simulasi kredit pada profil pengguna dan hanya menggunakan sistem bunga *flat rate*. Hal ini membuat estimasi simulasi kredit yang pengguna terima tidak mencerminkan akurasi estimasi yang menyeluruh. Maka dari itu, dibutuhkan sistem yang banding estimasi simulasi kredit yang lebih adaptif dan lebih nyata dengan mengestimasi penghasilan dan juga jenis pekerjaan pengguna.

Berkembangnya teknologi informasi, terutama perkembangan aplikasi berbasis *web*, mendorong perkembangan sistem yang interaktif, responsif, dan mudah diakses. Aplikasi berbasis *web* lebih mudah diakses, tidak memerlukan instalasi, dan dapat digunakan di berbagai jenis perangkat. Maka dari itu konstruktor mengusulkan aplikasi berbasis *web* simulasi kredit mobil bekas menggunakan algoritma logika *rule-based*. Algoritma ini akan mengestimasi simulasi kredit dengan mengestimasi penghasilan pengguna menggunakan logika *if-else* yang dapat dengan mudah disesuaikan dan dikembangkan.

Aplikasi ini memberikan calon pembeli estimasi yang lebih akurat tentang cicilan yang perlu dibayar dengan mempertimbangkan harga mobil, uang muka, tenor, penghasilan, dan tipe pekerjaan. Oleh karena itu, penulis mengusulkan pembuatan aplikasi Simulasi Kredit Mobil Bekas. Penulis berharap aplikasi ini dapat membantu pengguna dalam mempertimbangkan pembelian kendaraan bekas secara kredit.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah penelitian ini adalah sebagai berikut:

1. Bagaimana merancang dan membangun sistem simulasi kredit mobil bekas yang dapat memberikan hasil perhitungan cicilan secara cepat, akurat, dan transparan?
2. Bagaimana mengimplementasikan algoritma *rule-based* pada sistem simulasi kredit untuk merepresentasikan aturan-aturan kredit, seperti uang muka, tenor pinjaman, bunga, dan umur kendaraan?

3. Bagaimana hasil pengujian sistem simulasi kredit mobil bekas berbasis *rule-based* dalam membantu konsumen maupun *dealer* dalam pengambilan keputusan kredit?

1.3 Batasan Masalah

Agar penelitian lebih terarah, maka batasan masalah ditetapkan sebagai berikut:

1. Sistem hanya digunakan untuk **simulasi kredit mobil bekas**.
2. Perhitungan dibatasi pada variabel utama: **harga mobil, DP, tenor, bunga, dan umur kendaraan**.
3. Algoritma yang digunakan adalah *rule-based* sederhana berbasis logika *IF-THEN*.
4. Data simulasi bersifat contoh/studi kasus, **tidak terintegrasi langsung dengan leasing atau bank**.
5. Aplikasi dibangun berbasis **web sederhana menggunakan PHP dan MySQL**.
6. Hasil simulasi hanya sebagai **alat bantu keputusan**, bukan keputusan final kredit.
7. **Aplikasi dibangun menggunakan bahasa pemrograman PHP dengan framework *Native PHP* (tanpa framework modern)**.
 - a. **Versi PHP yang digunakan: PHP 8.x**
 - b. **Basis data menggunakan MySQL versi 8.0 melalui XAMPP (phpMyAdmin)**.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah merancang serta mengembangkan sebuah aplikasi yang dapat memberikan simulasi kredit kendaraan bekas dengan tingkat akurasi dan relevansi yang sesuai dengan kondisi finansial pengguna. Secara khusus, penelitian ini bertujuan untuk:

1. Merancang dan membangun sistem simulasi kredit mobil bekas yang mampu menghitung cicilan, bunga, dan total pembayaran secara otomatis.
2. Mengimplementasikan algoritma rule-based untuk mempresentasikan aturan-aturan kredit yang berlaku.
3. Menguji dan mengevaluasi kinerja sistem simulasi kredit mobil bekas berbasis *rule-based* sebagai alat bantu pengambilan keputusan.
4. Memberikan hasil simulasi berupa estimasi cicilan bulanan yang realistis, sehingga calon pembeli dapat mempertimbangkan secara matang kemampuan finansial mereka sebelum mengambil keputusan membeli mobil secara kredit.

1.5 Manfaat Penelitian

Penelitian ini diharapkan memberikan kontribusi baik secara praktis maupun akademik, dengan manfaat sebagai berikut :

1. **Bagi calon pembeli mobil bekas**, aplikasi membantu merencanakan pembelian secara kredit dengan simulasi cicilan yang sesuai kondisi finansial, sehingga mengurangi risiko *overbudgeting* maupun gagal bayar.
2. **Bagi lembaga pembiayaan dan dealer**, sistem dapat berfungsi sebagai *pre-screening tool* yang mempercepat proses penawaran dan meningkatkan efektivitas komunikasi dengan konsumen.

3. **Bagi pengembang sistem dan mahasiswa**, penelitian ini menjadi studi kasus penerapan logika *rule-based* dan model *waterfall* dalam pengembangan aplikasi sederhana, serta dapat dijadikan referensi untuk proyek serupa.
4. **Bagi akademisi**, penelitian ini menambah literatur di bidang aplikasi *web*, khususnya simulasi kredit, sekaligus menunjukkan penerapan teknologi informasi dalam menyelesaikan masalah nyata masyarakat.

1.6 Sistematika Penulisan

Skripsi ini disusun secara terstruktur ke dalam lima bab pokok, di mana setiap bab saling berkaitan secara logis. Adapun penjelasan mengenai sistematika penulisan adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini merupakan bagian pendahuluan yang memaparkan latar belakang permasalahan sebagai dasar dilaksanakannya penelitian. Di dalamnya disajikan rumusan masalah yang menjadi fokus utama, batasan penelitian agar tetap terarah dan realistis, serta tujuan yang ingin dicapai melalui pengembangan aplikasi. Selain itu, bab ini juga menguraikan manfaat penelitian baik secara praktis maupun akademis, serta menyajikan sistematika penulisan sebagai pedoman bagi pembaca dalam memahami keseluruhan isi laporan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi kajian teori yang menjadi pijakan dalam pengembangan aplikasi simulasi kredit mobil bekas. Beberapa teori yang diuraikan mencakup konsep sistem dan sistem informasi, definisi serta jenis-jenis

kredit kendaraan bermotor, algoritma logika *rule-based*, dan teknologi *web* yang digunakan dalam perancangan aplikasi. Selain itu, bab ini juga memuat tinjauan penelitian terdahulu yang relevan sebagai bahan perbandingan sekaligus penguat dalam penyusunan sistem.

BAB III METODOLOGI PENELITIAN

Bab ini menguraikan metode serta pendekatan yang diterapkan dalam proses perancangan dan pengembangan aplikasi. Model rekayasa perangkat lunak yang digunakan adalah *waterfall*, dengan tahapan meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan. Selain itu, bab ini juga memaparkan waktu dan lokasi penelitian, alat serta bahan yang dipakai, serta perancangan basis data dan struktur sistem yang digambarkan melalui diagram (*ERD dan flowchart*). Dengan demikian, bab ini menjadi bagian utama yang menjelaskan bagaimana proses pembangunan aplikasi dilakukan secara terstruktur.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil implementasi dari aplikasi simulasi kredit mobil bekas yang telah dirancang. Ditampilkan antarmuka pengguna (*user interface*), hasil input maupun output simulasi, serta mekanisme perhitungan dalam sistem yang berjalan berdasarkan logika *rule-based*. Selanjutnya, hasil implementasi dianalisis dan dibandingkan dengan tujuan penelitian untuk menilai sejauh mana aplikasi mampu menjawab rumusan masalah pada bab sebelumnya. Selain itu, bab ini juga membahas keunggulan, keterbatasan, serta kendala yang muncul selama proses pengembangan dan penerapan sistem.

BAB V PENUTUP

Bab ini menjadi penutup dari keseluruhan skripsi, yang berisi rangkuman kesimpulan berdasarkan hasil penelitian baik dari aspek teknis maupun konseptual. Kesimpulan diperoleh dari analisis yang telah dilakukan pada bab sebelumnya. Selain itu, bab ini juga memuat saran untuk pengembangan ke depan, mencakup peningkatan fitur, peningkatan akurasi simulasi, serta peluang penerapan metode lain di masa mendatang, sehingga aplikasi dapat lebih optimal dan memberikan manfaat yang lebih luas.

BAB II TINJAUAN PUSTAKA

2.1 Pengertian Sistem

Sistem merupakan suatu kesatuan elemen yang saling berinteraksi serta bekerja sama dalam rangka mencapai tujuan tertentu. Menurut Jogiyanto (2005), sistem dapat diartikan sebagai kumpulan elemen yang saling terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Elemen-elemen tersebut dapat berupa manusia, perangkat keras, perangkat lunak, prosedur, maupun data yang berfungsi sebagai satu kesatuan. Suatu sistem juga memiliki batasan yang membedakannya dari lingkungan luar, serta meliputi komponen masukan (*input*), proses, dan keluaran (*output*) yang saling berkaitan.

Dalam penelitian ini, sistem yang dimaksud adalah rangkaian komponen perangkat lunak yang dirancang untuk mensimulasikan perhitungan kredit mobil bekas. Sistem menerima *input* berupa harga mobil, uang muka, penghasilan, dan jenis pekerjaan calon pembeli. Data tersebut kemudian diproses dengan menggunakan algoritma berbasis aturan (*rule-based*) untuk menentukan besaran bunga serta cicilan, sehingga menghasilkan *output* berupa estimasi angsuran yang harus dibayarkan.

2.1.1 Karakteristik Sistem

Berdasarkan teori sistem, karakteristik utama suatu sistem meliputi :

1. Komponen sistem – Bagian-bagian yang bekerja sama dalam suatu kesatuan, seperti modul input data, modul perhitungan bunga, dan modul *output* simulasi.

2. Batas sistem – Memisahkan sistem dari lingkungannya, misalnya aplikasi simulasi hanya memproses data kredit mobil bekas, tidak untuk objek lain.
3. Lingkungan luar – Faktor eksternal yang mempengaruhi kinerja sistem, seperti regulasi pembiayaan atau tren pasar mobil bekas.
4. Penghubung – Media interaksi antara komponen sistem atau antar sistem dengan pengguna. Dalam aplikasi ini, penghubungnya adalah antarmuka *web*.
5. Tujuan sistem – Sistem dirancang untuk membantu pengguna menentukan simulasi cicilan yang sesuai kemampuan finansial.

2.1.2 Sistem Informasi

Sistem informasi merupakan integrasi dari manusia, perangkat keras, perangkat lunak, jaringan, serta sumber data yang berfungsi untuk mengumpulkan, mengelola, dan menyajikan informasi dalam rangka mendukung proses pengambilan keputusan. Menurut Laudon (2018), sistem informasi memiliki tiga fungsi utama, yaitu:

1. Mengumpulkan data dari berbagai sumber (*input*).
2. Mengolah data menjadi informasi yang bermanfaat (*process*).
3. Menyajikan informasi kepada pihak yang membutuhkan (*output*).

Dalam konteks penelitian ini, sistem informasi berbasis *web* dimanfaatkan sebagai sarana untuk membantu calon pembeli dalam melakukan simulasi kredit mobil bekas, dengan mempertimbangkan faktor penghasilan dan jenis pekerjaan.

2.2 Kredit Kendaraan Bermotor

Kredit kendaraan bermotor merupakan salah satu bentuk fasilitas pembiayaan yang disediakan oleh bank maupun lembaga pembiayaan, yang memungkinkan konsumen memperoleh kendaraan dengan cara pembayaran secara angsuran dalam jangka waktu tertentu. Pada skema ini, konsumen diwajibkan untuk membayar uang muka (*down payment*) di awal, kemudian melunasi sisa harga kendaraan melalui cicilan bulanan yang terdiri atas pokok pinjaman dan bunga.

Menurut Otoritas Jasa Keuangan (OJK, 2021), mekanisme pembiayaan kendaraan di Indonesia telah diatur guna melindungi kepentingan konsumen serta menjaga stabilitas lembaga pembiayaan. Salah satu ketentuan yang diberlakukan adalah kewajiban pembayaran uang muka minimal sebesar 30% untuk pembelian mobil penumpang.

2.2.1 Jenis Perhitungan Kredit

1. *Flat* – Suku bunga dihitung dari jumlah pokok pinjaman awal dan nilainya tetap setiap bulan.
2. Efektif – Suku bunga dihitung dari sisa pokok pinjaman, sehingga nominal bunga menurun setiap bulan.
3. Anuitas – Cicilan per bulan tetap, tetapi porsi bunga dan pokok berubah setiap bulan.

2.2.2 Faktor Yang Mempengaruhi Kredit

1. Harga kendaraan – Semakin tinggi harga, semakin besar cicilan.
2. Uang muka (DP) – DP yang besar mengurangi jumlah pinjaman.
3. Tenor – Semakin panjang tenor, cicilan perbulan semakin kecil bunga total lebih besar.

4. Suku bunga – Menentukan besarnya beban tambahan di luar pokok pinjaman.
5. Profil pemohon – Pendapatan, pekerjaan, dan riwayat kredit memengaruhi penentuan bunga.

Dalam penelitian ini, bunga kredit akan disesuaikan berdasarkan penghasilan dan pekerjaan menggunakan algoritma *rule-based*.

2.3 Algoritma Rule-Based

Algoritma *rule-based* merupakan metode pengambilan keputusan yang bekerja dengan menggunakan aturan-aturan logis (*rules*) yang telah didefinisikan sebelumnya. Aturan ini biasanya berbentuk pernyataan *if-then*, di mana suatu kondisi tertentu akan menghasilkan aksi tertentu. Misalnya:

Jika penghasilan $< 8.000.000$ dan pekerjaan = “karyawan swasta”

Maka bunga kredit = 6,5%

Jika penghasilan $> 10.000.000$ dan pekerjaan = “PNS”

Maka bunga kredit = 4,5%

Dengan pendekatan ini, sistem akan mengeksekusi aturan secara berurutan hingga ditentukan kondisi yang sesuai dengan data yang diberikan pengguna. Sederhananya, *rule-based system* dapat dianggap sebagai representasi dari “pengetahuan pakar” yang dituangkan dalam bentuk aturan formal. Perhitungan keuangan menggunakan rumus dasar kredit sebagai berikut:

1. Pinjaman (P)

Pinjaman = Harga Mobil – DP

2. Total Bunga (B)

Total Bunga = Pinjaman \times Suku Bunga \times Lama Kredit

3. Total Bayar (T)

$$\text{Total Bayar} = \text{Pinjaman} + \text{Total Bunga}$$

4. Cicilan per Bulan (C)

$$\text{Cicilan per Bulan} = \text{Total Bayar} \div (\text{Lama Kredit} \times 12)$$

2.3.1 Sejarah Dan Perkembangan

Metode *rule-based* mulai digunakan sejak tahun 1970-an ketika sistem pakar (*expert system*) dikembangkan untuk membantu proses pengambilan keputusan di bidang medis dan teknik. Salah satu sistem terkenal adalah MYCIN, sebuah sistem pakar untuk diagnosis penyakit menular yang menggunakan aturan *if-then*. Keberhasilan MYCIN menjadi tonggak penting karena membuktikan bahwa aturan logis dapat digunakan untuk meniru cara berfikir manusia dalam menyelesaikan masalah kompleks.

Seiring perkembangan teknologi, metode *rule-based* tidak hanya digunakan dalam bidang medis, tetapi juga dalam bidang keuangan, perbankan, pendidikan, hingga rekomendasi produk. Hingga saat ini, *rule-based system* masih digunakan karena sifatnya yang sederhana, transparan, dan mudah dipahami.

2.3.2 Komponen Utama

Sebuah sistem *rule-based* biasanya terdiri dari tiga komponen utama:

1. *Knowledge Base* – Merupakan kumpulan aturan (*rules*) yang berisi pengetahuan atau logika bisnis.
2. *Inference Engine* – Bagian yang berfungsi mengevaluasi kondisi dan menentukan aturan mana yang berlaku berdasarkan input yang diberikan.
3. *Working Memory* – Tempat menyimpan data sementara yang digunakan saat proses penalaran berlangsung.

Dalam penelitian ini, *knowledge base* berisi aturan mengenai penentuan bunga kredit berdasarkan penghasilan dan pekerjaan calon pembeli. *Inference engine* akan memproses data tersebut, dan *working memory* menyimpan data *input* sebelum dihasilkan *output* simulasi kredit.

2.3.3 Kelebihan Algoritma *Rule-Based*

1. Sederhana dan transparan – Keputusan yang dihasilkan dapat dijelaskan dengan mudah kepada pengguna.
2. Cepat diimplementasikan – Tidak memerlukan metode yang rumit atau data pelatihan (training data).
3. Mudah dimodifikasi – Aturan dapat diperbarui atau ditambahkan sesuai kebutuhan bisnis.
4. Cocok untuk permasalahan dengan aturan jelas – Misalnya simulasi kredit berdasarkan kategori pendapatan dan pekerjaan.

2.3.4 Kelemahan Algoritma *Rule-Based*

1. Tidak fleksibel jika aturan terlalu banyak – Jika jumlah aturan bertambah besar, sistem menjadi sulit dikelola.
2. Kurang adaptif – Tidak dapat belajar dari data baru secara otomatis seperti halnya *machine learning*.
3. Bias tergantung aturan – Jika aturan tidak lengkap atau tidak tepat, keputusan bisa kurang akurat.

2.3.5 Relevansi Dengan Penelitian

Dalam penelitian ini, algoritma *rule-based* dipilih karena permasalahan yang dihadapi bersifat relatif sederhana, yaitu menentukan besaran bunga kredit berdasarkan profil penghasilan dan jenis pekerjaan calon pembeli mobil bekas.

Pendekatan lain seperti *machine learning* maupun *fuzzy logic* tidak diperlukan, sebab kebutuhan utama hanyalah logika keputusan yang jelas serta mudah dijelaskan kepada pengguna.

Melalui penerapan metode *rule-based*, sistem mampu menghasilkan simulasi kredit secara cepat, akurat, dan mudah dipahami oleh calon pembeli. Hal ini sesuai dengan tujuan utama aplikasi, yakni membantu pengguna dalam menentukan pilihan mobil yang sesuai dengan kemampuan finansialnya.

2.4 Teknologi Web

Teknologi *web* merupakan sekumpulan perangkat lunak, protokol, serta bahasa pemrograman yang mendukung pengembangan aplikasi berbasis internet. Menurut Kadir (2014), aplikasi web dapat didefinisikan sebagai perangkat lunak yang berjalan di atas jaringan komputer, khususnya internet, dan dapat diakses melalui peramban (*browser*).

Dibandingkan dengan aplikasi berbasis desktop, aplikasi *web* memiliki beberapa keunggulan, antara lain tingkat aksesibilitas yang lebih tinggi, tidak memerlukan proses instalasi pada perangkat pengguna, serta kemudahan dalam melakukan pembaruan. Atas dasar keunggulan tersebut, aplikasi simulasi kredit mobil bekas dalam penelitian ini dikembangkan dengan memanfaatkan *platform web* sehingga dapat diakses oleh siapa pun, kapan pun, dan di mana pun.

2.4.1 Arsitektur Client-Server

Aplikasi web umumnya menggunakan arsitektur *client-server*.

1. *Client* adalah perangkat pengguna yang mengakses aplikasi melalui browser, misalnya Google Chrome atau Mozilla Firefox, *Client* hanya menampilkan antarmuka aplikasi.

2. Server adalah mesin yang menjalankan logika aplikasi dan mengelola basis data. Server bertugas menerima permintaan dari *client*, memprosesnya, lalu mengirimkan hasil kembali ke *client*.
3. Database menyimpan informasi penting, misalnya data harga mobil, data simulasi kredit, serta input pengguna.

Dalam penelitian ini, pengguna (*client*) akan mengisi data berupa harga mobil, DP, penghasilan, dan pekerjaan. Server kemudian memproses data tersebut menggunakan algoritma *rule-based*, dan hasil simulasi kredit dikirim kembali kepada client dalam bentuk output cicilan bulanan.

2.4.2 Bahasa Pemrograman

Bahasa pemrograman yang digunakan dalam aplikasi ini adalah:

1. PHP (*Hypertext Preprocessor*) – Bahasa pemrograman sisi server yang populer, bersifat *open-source*, dan mudah digunakan untuk pengembangan aplikasi *web* dinamis.
2. HTML (*Hypertext Markup Language*) – Digunakan untuk membuat struktur dasar halaman *web*.
3. CSS (*Cascading Style Sheets*) – Berfungsi mengatur tampilan halaman agar lebih menarik dan *responsif*.
4. JavaScript – Menambah interaktivitas pada halaman, seperti validasi *input* sebelum dikirim ke server.

2.4.3 Database Management System (DBMS)

Aplikasi ini menggunakan MySQL sebagai sistem manajemen basis data. MySQL dipilih karena:

1. Bersifat *open-source* dan banyak digunakan di dunia akademik maupun industri.
2. Mendukung struktur data relasional yang sesuai untuk menyimpan data pengguna dan data simulasi.
3. Mudah diintegrasikan dengan PHP.

2.4.4 Framework Pendukung

Untuk mempercepat pengembangan, *framework* PHP seperti Laravel atau CodeIgniter dapat digunakan. Namun, karena penelitian ini menekankan pada kesederhanaan implementasi, maka penggunaan PHP dasar dianggap sudah cukup.

2.4.5 Kelebihan Aplikasi Web

1. Akses multi-platform – Dapat diakses dari komputer, laptop, maupun *smartphone*.
2. Tidak memerlukan instalasi – Pengguna hanya perlu membuka *browser*.
3. Pembaruan mudah – Jika ada perubahan sistem, cukup dilakukan di server.
4. Biaya distribusi rendah – Tidak ada biaya untuk mengirimkan aplikasi ke pengguna.
5. Akses jarak jauh – Aplikasi bisa digunakan kapan saja selama ada koneksi internet.

Dengan pertimbangan tersebut, pemilihan teknologi *web* sangat sesuai untuk penelitian ini. Aplikasi simulasi kredit mobil bekas yang berbasis *web* akan lebih mudah diakses oleh calon pembeli dan dapat membantu mereka mengambil keputusan finansial dengan cepat.

2.5 PHP

PHP merupakan singkatan dari PHP (*Hypertext Preprocessor*). PHP merupakan bahasa pemrograman script yang diletakkan dalam server yang biasa digunakan untuk membuat aplikasi *web* yang bersifat dinamis (Winanjar, J., & Susanti, D. 2021).

Hypertext Preprocessor (PHP) merupakan *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi server. Fungsi utama PHP dalam membangun *website* adalah untuk melakukan pengolahan data pada database. Data *website* akan dimasukkan ke database, di-edit, dihapus, dan ditampilkan pada *website* yang diatur oleh PHP. PHP merupakan bahasa pemrograman universal untuk penanganan pembuatan dan pengembangan sebuah *website* dan bisa digunakan bersamaan dengan HTML (Resman, K. I. K., Gunadnya, I. B. P., & Budisanjaya, I. P. G. 2021).

PHP (*Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menterjemahkan basis kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server side* yang ditambahkan ke HTML (Hari Utami, 2022).

2.6 HTML

HTML adalah suatu bahasa pemrograman yang digunakan untuk pembuatan halaman *website* agar dapat menampilkan berbagai informasi baik tulisan maupun gambar pada sebuah *web browser*. Saat ini bahasa HTML masih terus dikembangkan. Hal ini dikarenakan pengguna internet semakin hari semakin berkembang pesat. Oleh karena itu bahasa HTML harus ditingkatkan lagi agar bisa

menciptakan halaman *web* yang lebih berkualitas (Noviantoro, A., Silviana, A. B., Fitriani, R. R., & Permatasari, H. P. 2022).

2.7 CSS

CSS (*Cascading Style Sheets*) adalah sebuah bahasa pemrograman yang digunakan untuk mengatur tampilan atau desain dari halaman *website* agar lebih menarik dan terstruktur. Dengan CSS, pengembang *web* dapat mengatur berbagai aspek visual seperti warna, ukuran teks, jenis *font*, tata letak, jarak antar elemen, hingga efek animasi. CSS bekerja berdampingan dengan HTML, di mana HTML berfungsi untuk membuat struktur konten, sedangkan CSS berfungsi untuk mempercantik tampilan dari konten tersebut pada *web browser*. Saat ini, CSS telah berkembang hingga versi terbaru (CSS3) yang memiliki banyak fitur modern untuk menciptakan desain *web* yang responsif dan interaktif. Perkembangan CSS sangat penting karena kebutuhan pengguna internet yang semakin meningkat terhadap tampilan *web* yang lebih dinamis dan *user-friendly* (Noviantoro, A., Silviana, A. B., Fitriani, R. R., & Permatasari, H. P. 2022).

2.8 XAMPP

XAMPP merupakan *software server apache* dimana dalam XAMPP yang telah tersedia *database server* seperti MySQL dan PHP *programming*. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada *Windows* dan *Linux*. Keuntungan lain yang didapatkan adalah hanya dengan melakukan instalasi cukup satu kali kemudian didalamnya tersedia MySQL, *apache web server*, *database server* PHP. (Mawaddah & Fauzi, 2018)

2.9 MySQL

My Structured Query Language (MySQL) adalah suatu sistem basis data *relation* atau *Relational Database Managemnt System* (RDBMS) yang mampu bekerja secara cepat dan mudah digunakan. MySQL juga merupakan program pengakses *database* yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *multiuser* (Resman, K. I. K., Gunadnya, I.B. P., & Budisanjaya, I. P. G. 2021).

MySQL adalah salah satu jenis *database server* yang sangat populer, hal ini disebabkan karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*-nya. MySQL bersifat *Open Source, Software* ini dilengkapi dengan *Source code* (kode yang dipakai untuk membuat MySQL) (Winanjar, J., & Susanti, D. 2021).

2.10 JavaScript

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman *website* menjadi lebih interaktif dan dinamis. Jika HTML berfungsi untuk membangun struktur konten dan CSS untuk mengatur tampilan, maka JavaScript memberikan kemampuan pada *website* untuk merespons tindakan pengguna, seperti menampilkan pesan, memvalidasi form, membuat animasi, hingga mengelola data secara *real-time*. Dengan JavaScript, halaman web tidak hanya bersifat statis, tetapi dapat berinteraksi secara langsung dengan pengguna sehingga memberikan pengalaman yang lebih menarik. Seiring perkembangan teknologi, JavaScript terus mengalami peningkatan fitur dan didukung oleh berbagai *framework* maupun *library* seperti React, Angular, dan Vue, yang semakin memudahkan pengembang

dalam membangun aplikasi *web modern* (Noviantoro, A., Silviana, A. B., Fitriani, R. R., & Permatasari, H. P. 2022).

2.11 Penelitian Terdahulu

Beberapa penelitian yang relevan antar lain:

1. Arifin (2020) – Mengembangkan aplikasi simulasi kredit kendaraan bermotor berbasis Android dengan metode bunga *flat*. Aplikasi ini membantu pengguna menghitung cicilan bulanan, namun tidak mempertimbangkan profil penghasilan.
2. Suryanto (2019) – Menerapkan logika *fuzzy* dalam sistem penentuan kelayakan kredit konsumen. Penelitian ini unggul dalam menilai tingkat risiko, tetapi metode *fuzzy* relatif kompleks untuk pengguna awam.
3. Putra (2021) – Merancang sistem simulasi kredit rumah berbasis *web* menggunakan PHP dan MySQL. Fokus penelitian pada properti, bukan kendaraan, namun menunjukkan efektivitas aplikasi *web* dalam perhitungan kredit.
4. Rahmawati (2018) – Menggunakan sistem *rule-based* untuk seleksi penerimaan beasiswa. Hal ini membuktikan fleksibilitas metode *rule-based* dalam berbagai bidang.
5. Andriansyah (2022) – Membangun sistem rekomendasi pembelian motor bekas berbasis *web* dengan algoritma *if-else* sederhana. Persamaannya adalah pada kendaraan bekas, perbedaannya pada aspek perhitungan cicilan.
6. Hidayat (2021) – Membuat simulasi kredit kendaraan bermotor berbasis *web* dengan sistem bunga *flat*. Kelemahannya, perhitungan tidak memperhitungkan kemampuan finansial pengguna.

2.12 *Flowchart*

Flowchart atau diagram alir merupakan suatu representasi visual yang digunakan untuk menggambarkan urutan proses, langkah-langkah logika, atau alur kerja dalam penyelesaian suatu masalah. *Flowchart* menyajikan struktur algoritma secara grafis melalui simbol-simbol standar seperti terminator, proses, *input/output*, *decision*, *connector*, dan preparation. Dengan visualisasi ini, alur logika menjadi lebih mudah dipahami dibandingkan hanya menggunakan deskripsi teks.

Flowchart berfungsi sebagai alat bantu yang sangat penting dalam analisis sistem, pemrograman, dan perancangan prosedur kerja. *Flowchart* membantu menggambarkan hubungan antar langkah secara terstruktur, sehingga mempermudah pengembang maupun pengguna untuk memahami cara kerja sebuah sistem. Selain itu, *flowchart* juga membantu mengidentifikasi potensi kesalahan logika lebih awal sebelum program diimplementasikan, sehingga menghemat waktu dan biaya dalam proses pengembangan.

Pembuatan *flowchart* biasanya dilakukan pada tahap analisis atau perancangan, sebelum proses coding dimulai. *Flowchart* dapat digunakan untuk berbagai tujuan, seperti mendeskripsikan alur sistem manual, proses bisnis, algoritma komputasi, hingga prosedur operasional standar. Dengan adanya *flowchart*, komunikasi antara analis sistem, programmer, dan pengguna menjadi lebih efektif karena setiap langkah digambarkan secara jelas dan tidak menimbulkan penafsiran ganda.

2.12.1 Tujuan dan Fungsi *Flowchart*

Tujuan *Flowchart*

1. Mempermudah pemahaman alur proses atau algoritma. Dengan simbol yang sederhana dan *visual*, *flowchart* membuat logika kompleks menjadi lebih mudah dibaca.
2. Menyederhanakan komunikasi antara perancang sistem, programmer, dan pengguna. *Flowchart* menjadi media komunikasi yang jelas sebelum implementasi dilakukan.
3. Menjadi dasar penyusunan program komputer. *Flowchart* membantu programmer dalam menerjemahkan alur logika ke dalam kode program.
4. Menyediakan dokumentasi visual untuk pengembangan sistem. Dokumentasi ini memudahkan pemeliharaan (*maintenance*) dan pengembangan lanjutan.



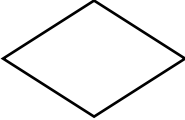

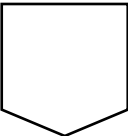
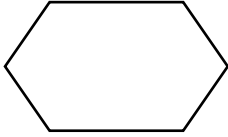
Fungsi *Flowchart*

1. Menjelaskan alur kerja proses secara sistematis. *Flowchart* menyusun langkah-langkah secara berurutan sehingga proses dapat diikuti dengan jelas.
2. Mendeteksi kesalahan logika lebih awal. Karena alur terlihat secara keseluruhan, kesalahan dapat ditemukan sebelum tahap implementasi.
3. Mempermudah dokumentasi sistem atau algoritma. *Flowchart* menjadi dokumen pendukung yang memberikan gambaran struktur logika program.
4. Meningkatkan efisiensi kerja. Setiap langkah sudah dirancang secara jelas, sehingga meminimalkan kesalahan dan pengulangan pekerjaan.

5. Membantu proses *debugging* dan evaluasi sistem. *Flowchart* memudahkan pelacakan alur jika terjadi kesalahan program.
6. Sebagai pedoman pembuatan prosedur kerja. Banyak organisasi menggunakan *flowchart* untuk menggambarkan proses bisnis.

2.12.2 Simbol Simbol Standar *Flowchart*

Tabel 2.1 Simbol Simbol *Flowchart*

Simbol	Nama	Fungsi
	Terminator (<i>Start/End</i>)	Menunjukkan awal atau akhir dari proses.
	Proses	Menunjukkan langkah perhitungan atau operasi.
	<i>Decision</i> (Keputusan)	Menunjukkan percabangan logika “Ya/Tidak” atau “Benar/Salah”.
	<i>Input/Output (I/O)</i>	Menunjukkan proses pemasukan atau keluaran data
	<i>Connector / Flowline</i>	Menunjukkan arah aliran proses
	<i>Preparation</i>	Menandakan inisialisasi atau persiapan sebelum proses utama

2.13 ERD

Entity Relationship Diagram (ERD) adalah model pemodelan data yang digunakan untuk mendeskripsikan struktur logis dari sebuah sistem atau database secara grafis. *ERD* memetakan entitas yang relevan, atribut yang dimiliki entitas,

serta hubungan antar entitas tersebut. Menurut Chen (1976), *ERD* berfungsi sebagai alat untuk memodelkan data konseptual yang mendukung kebutuhan sistem secara akurat dan terstruktur sebelum diimplementasikan ke dalam database fisik. Dengan *ERD*, perancang sistem dapat menggambarkan kebutuhan data, hubungan antar data, serta aturan bisnis (*business rules*) yang berlaku pada sistem.

2.13.1 Tujuan *ERD*

ERD digunakan dengan beberapa tujuan utama, antara lain:

1. Mengidentifikasi data yang diperlukan dalam sistem.
2. Menentukan hubungan antar data berdasarkan proses bisnis.
3. Menjadi dasar perancangan database yang efektif, konsisten, dan bebas duplikasi.
4. Mempermudah komunikasi antara analis sistem, pengembang, dan pemangku kepentingan.
5. Menjadi blueprint dari kebutuhan data sebelum proses implementasi pada DBMS.

2.13.2 Komponen *ERD*

Berikut komponen dasar dalam *ERD* modern :

1. *Entity*

Entitas adalah objek nyata atau konsep yang datanya ingin disimpan dalam sistem.

Contoh entitas dalam aplikasi simulasi kredit mobil:

- I. *User*
- II. *Mobil*
- III. *Admin*
- IV. *Simulasi*

Ditampilkan dalam bentuk persegi panjang.

2. *Attribute*

Atribut adalah karakteristik atau informasi yang melekat pada suatu entitas.

Contoh atribut entitas mobil:

- I. id_mobil
- II. nama_mobil
- III. harga_mobil
- IV. tahun

Atribut khusus:

- I. *Primary Key (PK)* → atribut unik identitas entitas
- II. *Foreign Key (FK)* → atribut yang merujuk pada entitas lain

3. *Relationship*

Relationship menggambarkan hubungan antar entitas dalam sistem.

Terdapat beberapa jenis relasi:

- I. *One to One (1:1)*
- II. *One to Many (1:M)*
- III. *Many to Many (M:N)*

Contoh:

- I. Satu user dapat melakukan banyak simulasi (1:M).
- II. Satu mobil dapat digunakan pada banyak simulasi (1:M).
- III. Relationship ditampilkan dalam bentuk garis dengan simbol.

2.13.3 Kardinalitas

Kardinalitas adalah batasan jumlah kemunculan suatu entitas yang dapat berhubungan dengan entitas lainnya. Jenis kardinalitas antara lain:

- I. *Mandatory One*
- II. *Optional One*
- III. *Mandatory Many*
- IV. *Optional Many*

Kardinalitas membantu memahami aturan bisnis sistem secara mendalam.

2.13.4 Manfaat ERD

- I. Memudahkan perancang sistem memahami struktur data.
- II. Mengurangi redundansi data.
- III. Memastikan database tersusun sesuai kebutuhan proses bisnis.
- IV. Menjadi acuan pembentukan tabel pada database fisik.
- V. Menjaga integritas dan konsistensi data.

2.14 Diagram Konteks

Diagram Konteks adalah diagram level tertinggi (Level 0) dari *Data Flow Diagram* (DFD) yang menggambarkan keseluruhan sistem sebagai satu proses tunggal. Diagram ini menunjukkan interaksi antara sistem dengan lingkungan luarnya melalui aliran data. Diagram konteks digunakan untuk memberikan gambaran makro mengenai ruang lingkup sistem sebelum dianalisis lebih detail ke dalam DFD level 1 dan seterusnya.

2.14.1 Fungsi Diagram Konteks

Menjelaskan batasan sistem.

- I. Menggambarkan entitas luar yang berinteraksi dengan sistem.
- II. Menjelaskan aliran data masuk dan keluar dari sistem.
- III. Menjadi fondasi untuk pemodelan DFD level berikutnya.
- IV. Mempermudah komunikasi konsep awal sistem kepada *stakeholder*.

2.14.2 Komponen Diagram Konteks

1. Sistem

Digambarkan sebagai sebuah lingkaran atau kotak besar yang mewakili seluruh proses dalam sistem.

Contoh:

Sistem Simulasi Kredit Mobil

2. *External Entity*

Entity luar adalah pihak yang berada di luar sistem namun berinteraksi langsung dengan sistem.

Contoh pada aplikasi simulasi kredit:

- I. *User*
- II. Admin

Ditampilkan sebagai kotak persegi panjang.

3. *Data Flow*

Menunjukkan informasi yang mengalir antara entitas luar dan sistem.

Contoh:

- I. *User* mengirim data simulasi → Sistem
- II. Sistem mengirim hasil simulasi → *User*
- III. Admin mengirim data mobil → Sistem

Ditampilkan sebagai panah.

2.14.3 Aturan Penyusunan Diagram Konteks

- I. Harus terdapat satu proses utama saja.
- II. Tidak boleh terdapat data store.
- III. Semua aliran data harus memiliki arah dan label yang jelas.

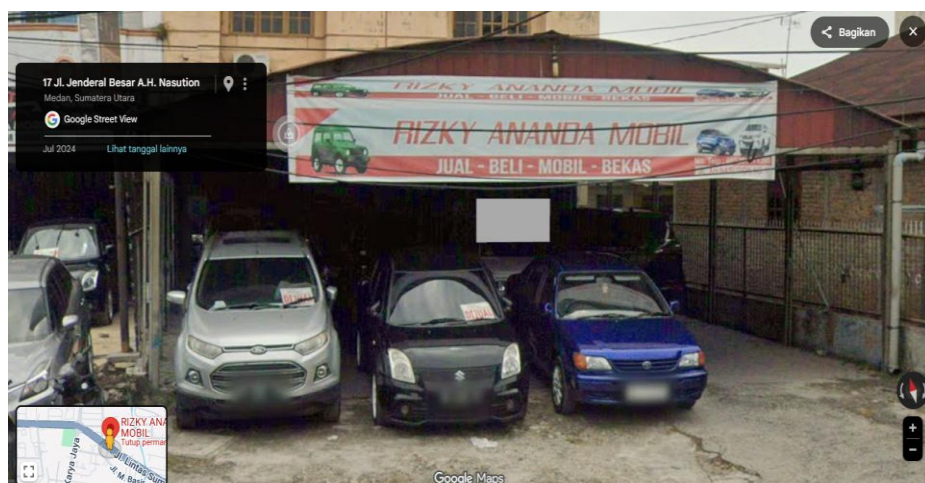
- IV. Semua entitas luar harus terhubung ke proses utama.
- V. Tidak boleh ada loop di dalam proses.

2.14.4 Keunggulan Diagram Konteks

- I. Menyederhanakan pemahaman awal sistem.
- II. Memisahkan sistem dari lingkungan eksternal.
- III. Menjadi dasar pemodelan DFD level 1.
- IV. Cocok untuk komunikasi dengan pihak non-teknis.

2.15 Profil Singkat Rizky Ananda Mobil

Rizky Ananda Mobil adalah *showroom* mobil bekas yang berfokus pada penyediaan kendaraan berkualitas dengan harga kompetitif. Berpengalaman dalam bidang jual beli mobil, *showroom* ini menjadi salah satu pilihan masyarakat Medan dan sekitarnya untuk mendapatkan mobil bekas yang layak pakai, aman, dan sesuai kebutuhan. Dengan lokasi yang mudah dijangkau, Rizky Ananda Mobil menghadirkan berbagai merek dan tipe mobil, mulai dari *city car* yang praktis, MPV untuk kebutuhan keluarga, hingga SUV yang tangguh.



Gambar 2.1 *Showroom* Rizky Ananda Mobil

Layanan Utama

- i. **Jual Mobil Bekas** – Menyediakan berbagai pilihan mobil dengan kondisi terawat dan siap pakai.
- ii. **Beli Mobil Bekas** – Membeli mobil lama pelanggan dengan harga yang sesuai dengan kondisi kendaraan.
- iii. **Tukar Tambah Mobil** – Memudahkan pelanggan yang ingin menukar mobil lama dengan unit lain.
- iv. **Konsultasi & Informasi** – Memberikan penjelasan dan rekomendasi kepada pelanggan untuk menemukan mobil yang sesuai dengan kebutuhan dan anggaran.

Kelebihan Rizky Ananda Mobil

- i. Lokasi strategis di pusat keramaian Medan.
- ii. Stok mobil bervariasi dengan kondisi terjaga.
- iii. Proses jual beli yang transparan dan aman.
- iv. Pelayanan ramah dan komunikatif.
- v. Harga kompetitif dengan negosiasi *fleksibel*.