DAFTAR TABEL

Tabel 3. 1 Tabel bobot 40
Tabel 3. 2 Tabel data 40
Tabel 3. 3 Tabel faq 41
Tabel 3. 4 Tabel konfigurasi 41
Tabel 3. 5 Tabel kriteria 42
Tabel 3. 6 Tabel merek 42
Tabel 3. 7 Tabel mobil 43
Tabel 3. 8 Tabel users 44
Tabel 3. 9 Tabel wishlist 45
Tabel 3. 10 Tabel kriteria 58
Tabel 3. 11 Tabel Bobot 59
Tabel 3. 12 Tabel kecocokan 60
Tabel 3. 13 Tabel pengujian 65
Tabel 4. 1 Tabel data model 69
Tabel 4. 2 Tabel controller 69
Tabel 4. 3 Tabel Pengujian Sistem 80
Tabel 4. 4 Mobil Honda 84
tabel 4. 5 Sub-Kriteria 85
tabel 4. 6 Hasil Deskripsi 86

ABSTRAK

Saat ini E-Commerce sebagai sebuah portal yang mampu menunjang perniagaan, dalam kasus ini adalah rajamobil.com. adapun selain sebagai penunjang e-commerce memiliki sebuah problem yang mana pengunjung situs tidak disuguhkan pada e-commerce yang mampu memilih mobil dengan cepet, tepat, dan efisien. Sehingga dalam proses pembelian yang terjadi terdapat beberapa kendala dalam menentukan pilihan mobil yang sesuai dengan kriteria yang dikehendaki. Munculculnya sistem pendukung keputusan (SPK) bertujuan untuk memudahkan pengguna dalam pembelian mobil dengan dilengkapi dengan metode simple additive weighting yang mampu menampilkan pilihan mobil dengan merek dan jenis yang sama namun dengan harga yang berbeda dari banyaknya data mobil dari beberapa penjual berbeda yang bisa menjadi pertimbangan pembeli seperti tahun keluaran, jarak tempuh, kondisi mesin, kondisi fisik dan lain-lain. Demi memenuhi kebutuhan pembeli, dirancanglah sistem pendukung keputusan rekomendasi Pembelian mobil dengan metode SAW yang dapat mengatur bobot sesuai dengan titik berat kriteria yang diunggulkan sendiri.
Keyword: mobil, e-commerce, Sistem Pendukung Keputusan, SAW.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Mobil merupakan salah satu alat transportasi yang diidamkan banyak orang. Selain kenyamannan yang ditawarkan, mobil juga dapat memuat banyak penumpang dan banyak barang, sehingga memilih mobil sebagai alat transportasi sehari–hari menjadi hal yang mutlak bagi kebanyakan masyarakat di era modern ini. Untuk memiliki dan membeli mobil, tidaklah sesusah dulu sebelum adanya internet, dimana harus mendatangi setiap shorum mobil atau mendatangi pameran mobil jika ingin melihat model dan harga mobil terbaru maupun mobil bekas. Sekarang pembeli cukup mengakses portal informasi jual beli mobil melalui situs–situs *e-commerce* yang terkenal di Indonesia seperti OLX, Rajamobil.com, Bukalapak dan *e-commerce* lainnya. Dengan adanya informasi mobil yang selalu *update* memudahkan pembeli untuk mencari mobil idaman.

Selain memberikan kemudahan, munculnya *e-commerce* sebagai portal untuk melihat informasi mobil yang akan dibeli juga menimbulkan masalah baru. Banyaknya data mobil yang dijual oleh *e-commerce* tersebut membuat pengguna kesulitan dalam memilih mobil mana yang cocok untuk dibeli. Untuk perbedaan secara umum seperti jenis mobil maupun merek dari mobil, pengguna cukup terbantu dengan fitur kategori yang memisahkan jenis mobil dan merek mobil juga terdapat fitur pencarian untuk mendapatkan mobil dengan kata kunci yang diinginkan. Namun berbeda ketika pembeli dihadapkan dengan pilihan pada mobil dengan merek dan jenis yang sama namun dengan harga yang berbeda dari data mobil yang ditampilkan dari beberapa penjual yang berbeda. Karena mobil tidak hanya melihat aspek merek dan jenis saja, namun beberapa kriteria juga bisa menjadi pertimbangan pembeli seperti tahun keluaran, jarak tempuh, kondisi mesin, kondisi fisik dan lain – lain.

Rekomendasi mobil sangat dibutuhkan untuk membantu pembeli memilih mobil dari banyaknya mobil yang dijual dengan membandingkan beberapa aspek seperti harga, jarak tempuh, tahun keluaran, kondisi fisik dan beberapa aspek

lainnya. Salah satu metode yang digunakan untuk rekomendasi adalah Simple additive weighting (SAW). Penelitian ini menggunakan metode Simple additive weighting dikarenakan metode ini efektif merekomendasikan suatu permasalahan yang cukup kompleks, data dalam jumlah banyak, dan mampu mengkalkulasi banyak kriteria pertimbangan.

Berdasarkan latar belakang masalah yang telah dijabarkan diatas, maka judul yang diangkat dalam Tugas Akhir ini adalah ”**Sistem Pendukung Keputusan Rekomendasi Pembelian Mobil Menggunakan Metode Simple Additive Weighting (Saw) (Studi kasus: E-commerce Rajamobil.com)**”.

1.2 Perumusan Masalah

Dari uraian latar belakang di atas, maka dapat dibuat perumusan masalah yaitu antara lain:

1. Apakah metode simple additive weighting dapat digunakan untuk merekomendasikan mobil yang sesuai kriteria?
2. Bagaimana data dapat terupdate dari situs rajamobil.com?

1.3 Pembatasan Masalah

Pada perancangan Tugas Akhir ini, Batasan masalah yang telah dibuat ialah sebagai berikut:

1. Objek penelitian dibatasi hanya pada *e-commerce* dengan jenis *classifield*, yaitu di Rajamobil.com.
2. Parameter yang digunakan untuk mengambil keputusan yang tepat dibatasi pada 7 kriteria, yaitu merk mobil, harga mobil, sepesifikasi mesin, transmisi mobil, jarak tempuh mobil, deskripsi mobil, dan tahun mobil. Dan Merk mobil sebagai alternatif
3. Sistem pendukung keputusan menggunakan metode *Simple Additive Wieghting*, penentuan nilai bobot preferensi ditentukan oleh pengguna, bobot sub-kriteria menggunakan metode *Range Order Centroid*.
4. Dirancang dan diimplementasikan dalam bentuk halaman web menggunakan bahasa pemograman PHP dan MySQL untuk penyimpanan database.

1.4 Tujuan Penelitian

Berikut ini merupakan tujuan yang akan dicapai dalam penelitian tugas akhir ini.

1. Untuk mendapatkan data barang online dari situs Rajamobil.com.
2. Untuk merancang sistem rekomendasi mobil dengan metode SAW.

1.5 Manfaat Penelitian

Penulisan skripsi ini diharapkan mampu membantu para konsumen khususnya pembeli mobil untuk dapat menemukan mobil idaman saat melakukan pencarian pada situs *e-commerce* dengan menerapkan metode SAW yang dapat digunakan untuk merekomendasikan mobil terbaik sesuai dengan kriteria yang diinginkan oleh pembeli.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan pada pembuatan Tugas Akhir ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bagian ini berisi latar belakang dari masalah, Perumusan masalah, Pembatasan Masalah, Tujuan, Manfaat dan Sistematika.

BAB II TINJAUAN PUSTAKA

Memuat penjelasan tentang konsep dan prinsip dasar yang diperlukan untuk memecahkan masalah Tugas Akhir dan untuk merumuskan hipotesis apabila memang diperlukan dari berbagai referensi yang dijadikan landasan pada kegiatan penelitian yang dilakukan.

BAB III ANALISA DAN PERANCANGAN

Pada bagian ini berisi analisa yang dari kebutuhan sistem yang akan dibuat dan juga terdapat Perancangan dari sistem yang digambarkan dalam bentuk desain sesuai dengan analisa kebutuhan yang telah didapat pada saat pengumpulan data.

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bagian ini menjelaskan implementasi dari sistem yang telah dibuat, serta menjelaskan tentang jalannya ujicoba dari sistem serta analisa dari hasil yang telah dicapai dari sistem yang telah dibuat.

BAB V PENUTUP

Pada bagian ini berisi Kesimpulan dan Saran. Kesimpulan berasal dari hasil analisis dan merupakan pernyataan singkat, jelas dan tepat tentang apa yang diperoleh / dapat dibuktikan / dijabarkan dari hipotesis. Saran memuat berbagai usulan/pendapat yang sebaiknya di susun oleh penelitian sejenis.

BAB II
TINJAUAN PUSTAKA

Pada bagian ini membahas mengenai tinjauan pustaka yang relevan serta mendukung dalam penyusunan Tugas Akhir. Tinjauan pustaka ditulis secara lengkap dan urut sejalan dengan permasalahan yang dihadapi. Tinjauan pustaka digunakan sebagai referensi dan tolak ukur dalam perancangan sistem. Adapun tujuannya adalah sebagai dasar untuk memahami dalam meneliti sistem yang berjalan dan mengambarkan sistem baru yang akan dibuat. Selain itu juga digunakan sebagai pedoman dalam perancangan sistem yang akan dibuat.

2.1 E-Commerce

E-Commerce merupakan bagian dari e-business. Secara umum, seorang konsumen yang akan berbelanja online melalui internet memerlukan teknologi atau infrastruktur internet untuk mencari tahu tentang toko online atau *webstore*. Di sisi penjual atau penyedia jasa E-Commerce dapat melakukan pengumpulan informasi, misalnya data mengenai konsumen dimana seorang konsumen biasanya diminta untuk menjadi member terlebih dahulu sebelum melakukan transaksi lebih lanjut (Laudon & Traver, 2013). Pertama kali, tentunya seorang konsumen memerlukan informasi mengenai toko online mana yang menjual barang yang akan dibelinya dengan memanfaatkan teknologi *search engine* yang ada (Lee, 2014). Setelah toko online ditemukan, konsumen dapat langsung melihat-lihat barang yang akan dibelinya melalui fasilitas *web catalog* yang disediakan oleh situs E-Commerce. *Web catalog* ini juga berlaku sebagai sarana promosi barang (termasuk harga promosi dan diskon) bagi penjual. Beberapa situs E-Commerce menyediakan layanan untuk bagi konsumen untuk bernegosiasi harga dengan penjual, namun beberapa situs ada pula yang menerapkan harga *fixed* yang tidak dapat dinegosiasikan (Li, dkk. 2011).

E-Commerce secara strategis dapat berperan sebagai diferensiator yang dapat membentuk daya saing perusahaan melalui sejumlah keunikan baik produk maupun sistem pelayanannya. Usaha bisnis perniagaan tidak lagi bergantung

kepada ruang dan waktu dengan segmen pasar yang terbatas. E-commerce merupakan satu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang terjadi secara elektronik (Fingar, dkk. 2000). Kehadiran sistem E-Commerce membuat berbagai perusahaan mikro, kecil maupun menengah dapat memasuki pasar global secara mudah dan bahkan bersifat real-time. Semua pelaku bisnis mudah menjalin hubungan suatu bisnis dengan bisnis lainnya, dan memiliki hubungan langsung dengan konsumen (H. Li & Hong, 2013).

Sistem E-Commerce sebagai suatu bentuk kemajuan teknologi informasi telah membawa sejumlah perubahan, di antaranya menurunkan biaya interaksi antara pembeli dan penjual, interaksi menjadi lebih mudah tanpa batasan waktu dan tempat, lebih banyak alternatif dan mempermudah promosi, peluang memperluas pangsa pasar tanpa harus memiliki modal dan investasi yang besar, transparansi bisnis dan kemudahan memberikan pelayanan kepada konsumen atau pelanggan (Bernadi, 2013).

(Cordiaz & Prahasto, 2013) Secara umum, membedakan bisnis proses dari e-commerce menjadi 5 bentuk berbeda. Diantaranya adalah sebagai berikut:

1. *Classifieds/Listing/*Iklan Baris

Ini adalah model bisnis e-commerce paling sederhana yang cocok digunakan di negara-negara berkembang. Dua kriteria yang biasa diusung model bisnis ini:

- Website yang bersangkutan tidak memfasilitasi kegiatan transaksi online
- Penjual individual dapat menjual barang kapan saja, dimana saja secara gratis.

Adapun Metode transaksi yang paling sering digunakan di situs iklan baris ialah metode *cash on delivery* atau COD.

2. *Marketplace* C2C (*Customer to Customer*)

Ini adalah model bisnis dimana website yang bersangkutan tidak hanya membantu mempromosikan barang dagangan saja, tapi juga memfasilitasi

transaksi uang secara online. Berikut ialah indikator utama bagi sebuah website *marketplace*:

- Seluruh transaksi online harus difasilitasi oleh website yang bersangkutan
- Bisa digunakan oleh penjual individual
- Kegiatan jual beli di website marketplace harus menggunakan fasilitas transaksi online seperti layanan *escrow* atau rekening pihak ketiga untuk menjamin keamanan transaksi. Penjual hanya akan menerima uang pembayaran setelah barang diterima oleh pembeli. Selama barang belum sampai, uang akan disimpan di rekening pihak ketiga. Apabila transaksi gagal, maka uang akan dikembalikan ke tangan pembeli.

3. *Shopping* mall

Model bisnis ini mirip sekali dengan marketplace, tapi penjual yang bisa berjualan disana haruslah penjual atau brand ternama karena proses verifikasi yang ketat. Satu-satunya situs online shopping mall yang beroperasi di Indonesia ialah Blibli.

4. Toko *online* B2C (Business to Consumer)

Model bisnis ini cukup sederhana, yakni sebuah toko online dengan alamat wjebsite (domain) sendiri dimana penjual memiliki stok produk dan menjualnya secara online kepada pembeli. Beberapa contoh toko online di Indonesia ialah Bhinneka, Lazada Indonesia, BerryBenka.

5. Toko online di media sosial

Banyak penjual di Indonesia yang menggunakan situs media sosial seperti Facebook dan Instagram untuk mempromosikan barang dagangan mereka. Uniknya lagi, sudah ada pemain-pemain lokal yang membantu penjual berjualan di situs Facebook yakni Onigi dan LakuBgt. Ada juga *startup* yang mengumpulkan seluruh penjual di Instagram ke dalam satu website yakni Shopious (yuliani, 2014).

Merujuk penelitian sebelumnya, seorang konsumen yang akan berbelanja online melalui internet memerlukan teknologi atau infrastruktur internet untuk mencari tahu tentang toko online atau *webstore*. secara strategis E-Commerce

dapat berperan sebagai pembeda yang dapat membentuk daya saing perusahaan melalui sejumlah keunikan baik produk maupun sistem pelayanannya. Dan usaha bisnis perniagaan tidak lagi bergantung kepada ruang dan waktu dengan segmen pasar yang terbatas. Sehingga melalui media E-Commerce para *customer* dimudahkan dalam melakukan transaksi jual beli.

Adapun jenis e-commerce yang akan diterapkan pada sistem yaitu jenis *classifield.* Sebagaimana sistem yang dibuat memiliki kriteria seperti jenis *classifield* yang telah dijelaskan di atas. Seperti halnya website yang disediakan ialah gratis tanpa dipungut biaya, selain itu sistem tidak menyediakan transaksi bagi *customer* dikarenakan nilai harga barang begitu besar. Sehingga alangkah baiknya apabila transaksi dilakukan secara COD.

2.2 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) atau *Decision Support System* (DSS) adalah sebuah sistem yang mampu memberikan kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah dengan kondisi semi terstruktur dan tak terstruktur. Sistem ini digunakan untuk membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak seorangpun tahu secara pasti bagaimana keputusan seharusnya dibuat (Turban & Aronson, 2001).

Decision Support System didefinisikan sebagai sistem komputer yang mampu memberikan kemampuan baik kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah semi terstruktur. Secara khusus, DSS didefinisikan sebagai sebuah sistem yang mendukung kerja seorang manajer dalam memecahkan masalah semi terstruktur dengan cara memberikan informasi ataupun usulan menuju pada keputusan tertentu (Kusrini, 2007).

Sistem Pendukung Keputusan atau *Decision Support System*, secara umum di definisikan sebagai sebuah sistem yang mampu memberikan kemampuan baik kemampuan pemecahan masalah maupun kemampuan pengkomunikasian untuk masalah semi terstruktur. Secara khusus, DSS didefinisikan sebagai sebuah sistem yang mendukung kerja seorang manajer maupun sekelompok manajer dalam

memecahkan masalah semi terstruktur dengan cara memberi informasi ataupun usulan menuju pada keputusan tertentu (Setiaji, 2013).

2.2.1 Karakteristik Sistem Pendukung Keputusan

(Sprague & Watson, 1993) mendefinisikan Sistem Pendukung Keputusan (SPK) sebagai sistem yang memiliki lima karakteristik utama yaitu:

1. Sistem yang berbasis komputer.
2. Dipergunakan untuk membantu para pengambil keputusan
3. Untuk memecahkan masalah-masalah rumit yang mustahil dilakukan dengan kalkulasi manual
4. Melalui cara simulasi yang interaktif
5. data dan model analisis sebagai komponen utama.

2.2.2 Komponen Sistem Pendukung Keputusan

Secara umum Sistem Pendukung Keputusan dibangun oleh tiga komponen besar yaitu database Management, Model Base dan Software System/User Interface. Komponen SPK tersebut dapat digambarkan seperti gambar di bawah ini:

a) Database Management

Merupakan subsistem data yang terorganisasi dalam suatu basis data. Data yang merupakan suatu sistem pendukung keputusan dapat berasal dari luar maupun dalam lingkungan. Untuk keperluan SPK, diperlukan data yang relevan dengan permasalahan yang hendak dipecahkan melalui simulasi.

b) Model Base

Merupakan suatu model yang merepresentasikan permasalahan kedalam format kuantitatif (model matematika sebagai contohnya) sebagai dasar simulasi atau pengambilan keputusan, termasuk didalamnya tujuan dari permaslahan (objektif), komponen-komponen terkait, batasan-batasan yang ada (*constraints*), dan hal-hal terkait lainnya. Model *Base* memungkinkan pengambil keputusan menganalisa secara utuh dengan mengembangkan dan membandingkan solusi alternatif.

c) *User Interfase* / Pengelolaan Dialog

Terkadang disebut sebagai subsistem dialog, merupakan penggabungan antara dua komponen sebelumnya yaitu *Database Management* dan *Model Base* yang disatukan dalam komponen ketiga (*user interface*), setelah sebelumnya dipresentasikan dalam bentuk model yang dimengerti computer. *User Interface* menampilkan keluaran sistem bagi pemakai dan menerima masukan dari pemakai kedalam Sistem Pendukung Keputusan (Sprague et.al, 1993).

2.3 Metode SAW

2.3.1 Pengertian Metode SAW

Metode Simple Additive Weighting (SAW) sering juga dikenal istilah metode penjumlahan terbobot. Konsep dasar metode SAW adalah mencari penjumlahan terbobot dari rating kinerja pada setiap alternatif pada semua atribut (Pahlevy, 2010). Metode SAW membutuhkan proses normalisasi matriks keputusan X ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada (Kusumadewi, dkk. 2006).

Metode SAW dapat membantu dalam pengambilan keputusan suatu kasus, akan tetapi perhitungan menggunakan metode SAW ini hanya menghasilkan nilai terbesar yag akan terpilih sebagai alternatif yang terbaik. Perhitungan akan sesuai dengan metode ini apabila alternatif yang terpilih sesuai dengan kriteria.

2.3.2 Kelebihan Metode SAW

Kelebihan dari metode simple additive weighting dibanding dengan model pengambil keputusan lainnya terletak pada kemampuannya untuk melakukan penilaian secara lebih tepat karena didasarkan pada nilai kriteria dan bobot preferensi yang sudah ditentukan, selain itu SAW juga dapat menyeleksi alternatif terbaik dari sejumlah alternatif yang ada karena adanya proses perangkingan setelah menentukan bobot untuk setiap atribut (Kusumadewi dkk.,2006).

2.3.3 Langkah-langkah Penggunanaan Metode SAW

Tahapan-tahapan dalam metode SAW dijabarkan menjadi beberapa poin seperti berikut ini:

1. Menentukan alternatif (kandidat), yaitu Ai.
2. Menentukan kriteria yang akan dijadikan acuan dalam pengambilan keputusan, yaitu Cj.
3. Menentukan bobot preferensi atau tingkat kepentingan (W) setiap kriteria. W = [W1, W2, W3, …. , Wj] (1)
4. Membuat tabel rating kecocokan dari setiap alternatif pada setiap kriteria.
5. Membuat matrik keputusan X yang dibentuk dari tabel rating kecocokan dari setiap alternatif pada setiap kriteria. Nilai x setiap alternatif (Ai) pada setiap kriteria (Cj) yang sudah ditentukan, dimana, i=1,2,…m dan j=1,2,…n.

$$x = \begin{bmatrix} x_{11} & \cdots & x_{1j} \\ \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} \end{bmatrix} \quad (2)$$

6. Melakukan normalisasi matrik keputusan X dengan cara menghitung nilai rating kinerja ternomalisasi (rij) dari alternatif Ai pada kriteria Cj.

$$r_{ij} = \begin{cases} \dfrac{x_{ij}}{Max_i\, x_{ij}}, & jika\ j\ atribut\ benefit \\ \dfrac{Min_i x_{ij}}{x_{ij}}, & jika\ j\ atribut\ cost \end{cases} \quad (3)$$

   Keterangan:

   a. Dikatakan kriteria keuntungan apabila nilai xij memberikan keuntungan bagi pengambil keputusan, sebaliknya kriteria biaya apabila xij menimbulkan biaya bagi pengambil keputusan.
   b. Apabila berupa kriteria keuntungan maka nilai xij dibagi dengan nilai Maxi(xij) dari setiap kolom, sedangkan untuk kriteria biaya, nilai Mini(xij) dari setiap kolom dibagi dengan nilai xij.

7. Hasil dari nilai rating kinerja ternomalisasi (rij) membentuk matrik ternormalisasi (R)

$$V_i = \sum_{j=1}^{n} W_j r_{ij} \quad (4)$$

8. Hasil akhir nilai preferensi (Vi) diperoleh dari penjumlahan dari perkalian elemen baris matrik ternormalisasi (R) dengan bobot preferensi (W) yang bersesuaian eleman kolom matrik (W) (Eniyati, 2011).

2.3.4 Penerapan Metode SAW

Dalam menentukan penerimaan beasiswa, banyak sekali kriteria-kriteria yang harus dimiliki oleh individu sebagai syarat dalam mendapatkan beasiswa. Masing-masing sekolah pasti memiliki Kriteria-kriteria untuk menentukan siapa yang akan terpilih untuk menerima beasiswa. Pembagian beasiswa dilakukan oleh beberapa lembaga untuk membantu seseorang yang kurang mampu ataupun berprestasi selama menempuh studinya. Untuk membantu penentuan dalam menetapkan seseorang yang layak menerima beasiswa maka dibutuhkan sebuah sistem pendukung keputusan. Pada penelitian ini akan diangkat suatu kasus yaitu mencari alternative terbaik bedasarkan Kriteria-kriteria yang telah ditentukan dengan menggunakan metode SAW atau Simple Additive Weighting. Penelitian dilakukan dengan mencari nilai bobot untuk setiap atribut, kemudian dilakukan proses perankingan yang akan menentukan alternatif yang optimal, yaitu siswa terbaik (Eniyati, 2011).

Memilih alat kontrasepsi bukan merupakan hal yang mudah karena efek yang berdampak terhadap tubuh tidak akan diketahui selama belum menggunakannya. Selain itu tidak ada alat kontrasepsi yang selalu cocok bagi semua orang karena situasi dan kondisi tubuh dari setiap individu selalu berbeda. Dalam sistem pendukung keputusan ini dapat memberikan keputusan alternatif alat kontrasepsi yang nantinya bisa dijadikan sebagai acuan untuk memilih alat kontrasepsi. Sistem ini dikembangkan menggunakan PHP sebagai bahasa pemogramannya, MySql sebagai database server dan Apache sebagai web server (Kamaludin, 2012).

Dalam penelitian penulis membuat sebuah sistem informasi pengambilan keputusan dengan metode SAW. Yang mana dibatasi dengan 7 kriteria,

diantaranya: merk mobil, harga mobil, sepesifikasi mesin, transmisi mobil, jarak tempuh mobil, deskripsi mobil, dan tahun mobil. Dalam sistem pendukung keputusan ini dapat memberikan keputusan alternatif mobil yang sesuai dengan kriteria yang terbobot.

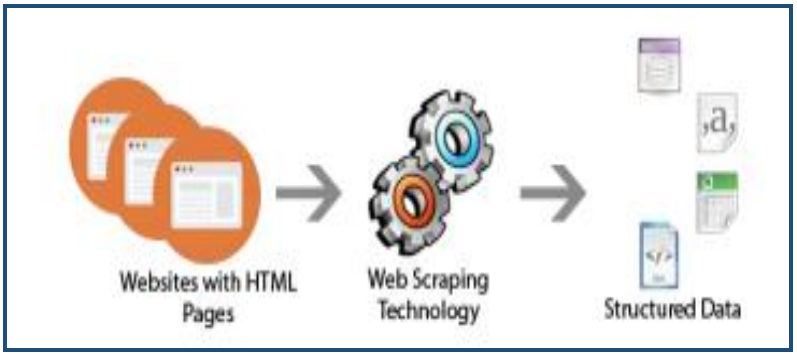
2.4 Web Scrapping

Web scraping adalah cara yang dilakukan untuk mendapatkan sebagian atau seluruh data dari sebuah halaman *world wide web* pada protokol *Hypertext Transfer Protocol* (HTTP). *Web Scraping* juga dikenal dengan isitilah *web harvesting.* Cara ini sangat bermanfaat bagi siapa saja yang membutuhkan data *internet* dalam jumlah banyak, *web scraping* jauh lebih efektif jika dibandingkan memasukkan data dengan cara mengetik, atau *copy-paste*.

Web scraping dapat digunakan untuk berbagai macam keperluan, mulai dari kebutuhan pribadi hingga kebutuhan persuhaan, beberapa manfaat penggunaan *web scraping* adalah:

- *Market research, web scraping* dapat membantu perusahaan untuk mendapatkan data yang bersifat *analytic*, untuk kebutuhan analisa pasar.
- Mendapatkan kontak info, web scraping digunakan untuk mendapat informasi kontak pelanggan, misal *e-mail* dan nomor telepon untuk kebutuhan perusahaan.
- Mendapatkan harga dari berbagai situs *E-commerce*, analisa harga barang menjadi pertimbangan bagi kebutuhan perusahaan, dengan mendapatkan informasi harga barang, perusahaan dapat menetapkan harga pada produk serupa.
- Digunakan oleh mesin telusur untuk melakukan pengindeksan halaman *website* dan menelusuri kata kunci tertentu, contoh Google, Bing, Yahoo.
- *Web scraping* juga diterapkan pada hampir semua *browser* untuk menginspeksi elemen HTML pada suatu halaman *website* dengan model *Data Objek Model* Tree (DOM-Tree). *Web scraping* dapat dilakukan dengan berbagai macam cara, beberapa teknik *web scraping* yang dikenal ialah *text-grepping* atau *text-mining*, HTML *parser*, HTML *Data Objek Model* atau

dengan menggunakan software khusus untuk melakukan *scraping*. Ilustrasi proses pengambilan data, dengan metode *web scraping* dapat dilihat pada Gambar 2.1 berikut.



Gambar 2. 1 Ilustrasi Mekanisme *Web scraping*

Pada penelitian kali ini, penulis akan menggunakan teknik HTML parser dengan bantuan *library simple HTML DOM* untuk mendapatkan data dari situs *Ecommerce*.

2.5 Pemrograman Berorientasi Objek

Pemrograman berorientasi objek (*Object Oriented Programming* atau disingkat OOP) adalah paradigma pemrograman yang berorientasikan kepada objek yang merupakan suatu metode dalam pembuatan program, dengan tujuan untuk menyelesaikan kompleksnya berbagai masalah program yang terus meningkat. Objek adalah entitas yang memiliki atribut, karakter dan disertai kondisi (Downing & Covington, 1992).

Ide dasar pada OOP adalah mengkombinasikan data dan fungsi untuk mengakses data menjadi sebuah kesatuan unit yang dikenal dengan nama objek. Objek adalah struktur data yang terdiri dari bidang data dan metode bersama dengan interaksi mereka untuk merancang aplikasi dan program komputer. Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya.

Pemrograman berorientasi objek bekerja dengan baik ketika dibarengi dengan *Objek-Oriented Analysis And Design Process* (OOAD). Jika membuat

program berorientasi objek tanpa OOAD, seperti membangun rumah tanpa terlebih dahulu penganalisis apa saja yang dibutuhkan oleh rumah itu, tanpa perencanaan, tanpa blue-print, tanpa menganalisis ruangan apa saja yang diperlukan, beberapa besar rumah yang akan dibangun dan sebagainya.

Konsep Dasar Pemrograman Berorientasi Objek

1. Kelas (class)

   Kelas (class) merupakan penggambaran satu set objek yang memiliki atribut yang sama. Kelas mirip dengan tipe data ada pemrograman non objek, akan tetapi lebih komprehensif karena terdapat struktur sekaligus karakteristiknya. Kelas baru dapat dibentuk lebih spesifik dari kelas ada umumnya.kelas merupakan jantung dalam pemrograman berorientasi objek.

2. Objek (Object)

   Objek merupakan teknik dalam menyelesaikan masalah yang kerap muncul dalam pengembangan perangkat lunak. Teknik ini merupakan teknik yang efektif dalam menemukan cara yang tepat dalam membangun sistem dan menjadi metode yang paling banyak dipakai oleh para pengembang perangkat lunak. Orientasi objek merupakan teknik pemodelan sistem riil yang berbasis objek.

3. Abstaksi (Abstraction)

   Kemampuan sebuah program untuk melewati aspek informasi yang diolah adalah kemampuan untuk fokus pada inti permasalahan. Setiap objek dalam sistem melayani berbagai model dari pelaku abstrak yang dapat melakukan kerja, laporan dan perubahan serta berkomunikasi dengan objek lain dalam sistem, tanpa harus menampakkan kelebihan diterapkan.

4. Enkapsulasi (pembungkus)

   Pembungkusan merupakan penggabungan potongan-potongan informasi dan perilaku-perilaku spesifik yang bekerja pada informasi tersebut, kemudian mengemasnya menjadi sesuatu yang disebut objek (Nugroho, 2005).

2.6 Framework

Framework adalah komponen pemrograman yang siap *re-use* (bisa digunakan ulang) kapan saja, sehingga *programmer* tidak harus membuat skrip yang sama untuk tugas yang sama. Misalkan *programmer* ingin halaman-halaman web menampilkan data dengan paginasi (*paging*) halaman, *framework* telah menyediakan fungsi *paging* tersebut sedangkan *programmer* cukup menggunkan fungsi tersebut pada saat *coding*, tetapi tentu dengan kaidah-kaidah yang ditetapkan oleh masing – masing *framework* (Kasman, 2015).

Adapun *framework* yang digunakan adalah Codeigniter. Codeigniter merupakan sebuah *framework* php yang bersifat open source dan menggunakan metode MVC (*Model*, *View*, *Controller*). Codeigniter bersifat *free* alias tidak berbayar jika anda menggunakannya. *framework* codeigniter di buat dengan tujuan sama seperti *framework* lainnya yaitu untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuat nya dari awal (Alfarabi Hadi, 2016).

BAB III
ANALISA DAN PERANCANGAN

Pada bagian ini membahas mengenai analisa dan perancangan yang mendukung dalam penyusunan Tugas Akhir. Analisa dan perancangan dilakukan dengan metode pengumpulan data dengan melihat referensi-referensi penelitian terdahulu dan menggunakan metode *waterfall*. Sedangkan untuk perancangannya menggunakan pemrograman berbasis objek atau sering disebut dengan OOP. Adapun tujuannya adalah sebagai dasar pembuatan sistem yang baru.

3.1 Metode Penelitian

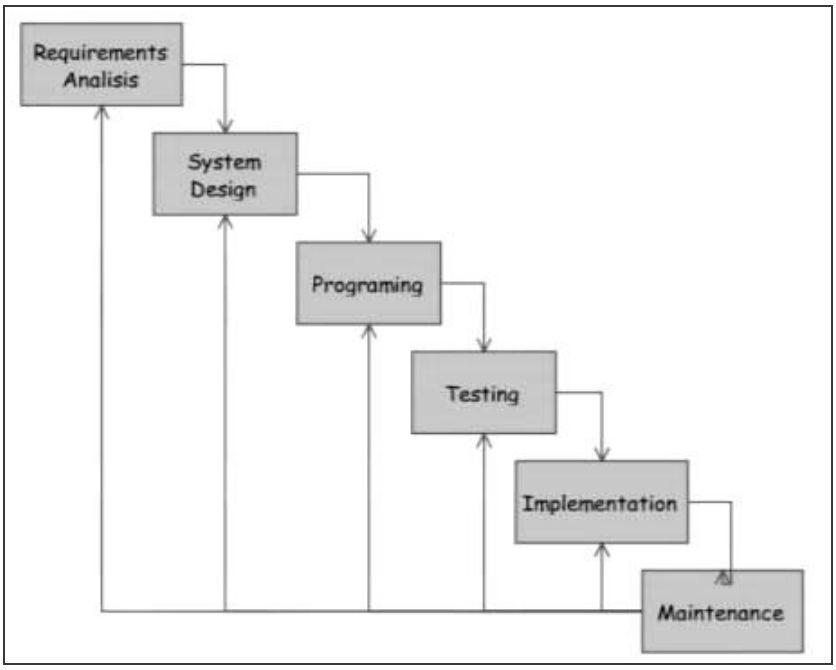
Metodologi penelitian dapat diartikan sebagai suatu langkah-langkah atau cara-cara dalam meneliti suatu objek. Berikut adalah metode yang digunakan dalam penelitian ini:

3.1.1 Tahap Pertama Pengumpulan Data

Metode pengumpulan data yang digunakan adalah studi pustaka, dimana penulis mengumpulkan data yang dilakukan dengan cara meneliti literatur-literatur atau referensi yang mendukung dalam penyusunan tugas akhir ini. Penulis membaca dan mempelajari bagaimana mengembangkan sistem melalui buku maupun internet, seperti yang telah diuraikan pada BAB II .

3.1.2 Tahap Kedua Pengembangan Perangkat Lunak

Pada tahapan ini penulis melakukan tahapan-tahapan dalam pembuatan "Sistem Pendukung Keputusan Rekomendasi Pembelian Mobil Menggunakan Metode Simple Additive Weighting (Saw) (Studi kasus: E-commerce Mobil Indonesia)" dengan menggunakan metode *waterfall* yaitu sebagai berikut :



Gambar 3. 1 Metode *Waterfall* (Air Terjun)

Gambar 3.1 merupakan penjelasan dari *System Development Life Cycle* (SDLC) dengan Metode *Waterfall* seperti pada gambar di atas:

1. Analisa Kebutuhan

   Pada tahap ini penulis menganalisa kebutuhan secara keseluruhan. Baik itu kebutuhan secara teknis maupun *non*-teknis dan juga memahami cara kerja dari sistem sebelumnya.

   Adapun kebutuhan-kebutuhan yang digunakan untuk menunjang pmbuatan sistem diantaranya sebagai berikut:

   - 1 unit PC
   - Web browser yang digunakan untuk media berselancar dalam pengoperasian system
   - Xampp sebagai web server
   - Sublime text yang digunakn sebagai pengetikan *coding*
   - Kebutuhsn non teknis mengenai sistem sebelumnya yang telah diuraikan pada gambar 3.2

2. Desain Sistem

   Pada tahap ini setelah mengetahui apa saja yang dibutuhkan oleh sistem maka penulis melanjutkan pada proses desain dari sistem aplikasi itu sendiri yaitu meliputi desain tampilan atau *User Interface*, *Flow Chart* dan diagram-diagram yang lain serta termasuk di dalamnya yaitu desain *Database*.

3. Pemrograman

   Tahapan ini merupakan tahapan yang dilakukan penulis selanjutnya setelah proses desain. Pada tahapan ini desain yang telah dibuat sebelumnya akan dituangkan dalam bentuk kode pemrograman dengan menggunakan *software* sublime text.

4. Pengujian

   Pada tahap ini dapat dikatakan sebagai tahap uji coba yang telah dibuat, penulis dapat menilai dan mengetahui sejauh mana sistem informasi telah bisa memperlihatkan cara kerja secara sepesifik dan apakah semua fungsi berjalan sesuai rencana tanpa adanya kesalahan yang harus diperbaiki lagi.

5. Implementasi

   Pada tahap ini bisa dikatakan sistem telah siap diimplementasikan dan digunakan pada khalayak umum.

6. Pemeliharaan

   Setelah dilakukan implementasi yang perlu dilakukan penulis adalah untuk tetap menjaga kesetabilan dan kelancaran dari penggunaan dari sitem informasi yang telah dibuat agar dapat mengatasi segala sesuatu yang tidak di inginkan baik itu dari kesalahan sistem itu sendiri atau faktor luar.

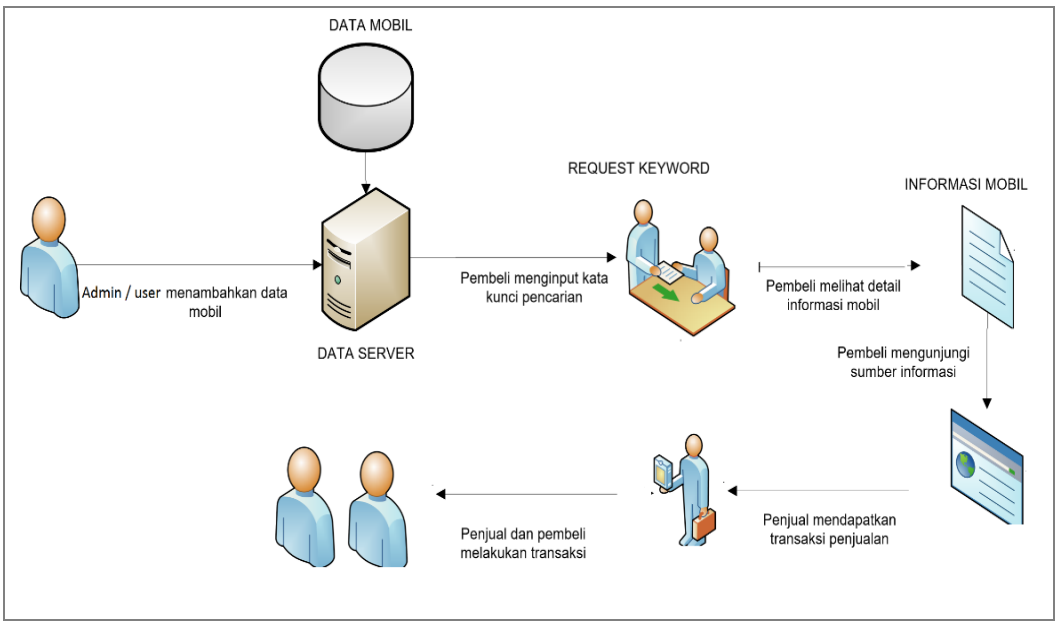
3.2 Deskripsi Sistem

Tugas akhir ini meneliti mengenai bagaimana menerapkan metode SAW dalam pengembangan sistem e-commerce pada penjualan mobil di Rajamobil.com untuk mendapatkan rekomendasi mobil terbaik bagi pembeli. Dengan demikian pembeli akan merasa sangat terbantu untuk mencari mobil idaman yang diinginkan, juga membantu pembeli agar terhindar dari kasus penyalahgunaan *e-commerce* oleh pihak yang tidak bertanggung jawab.

Berikut ini adalah bisnis proses dari sistem yang sedang berjalan dan bisnis proses baru yang akan ditawarkan untuk memperbaiki sistem yang sudah ada.

3.2.1 Bisnis proses yang sedang berjalan.

Pada gambar 3.2 berikut ini menampilkan bisnis proses yang sedang berjalan saat ini, atau yang banyak diterapkan pada *e-commerce* kebanyakan di indonesia. Dan yang nantinya akan diperbaiki dengan bisnis proses baru yang sesuai pada sistem yang akan dibuat dengan dilegkapi metode SAW.



Gambar 3. 2 Bisnis proses sistem yang sedang berjalan

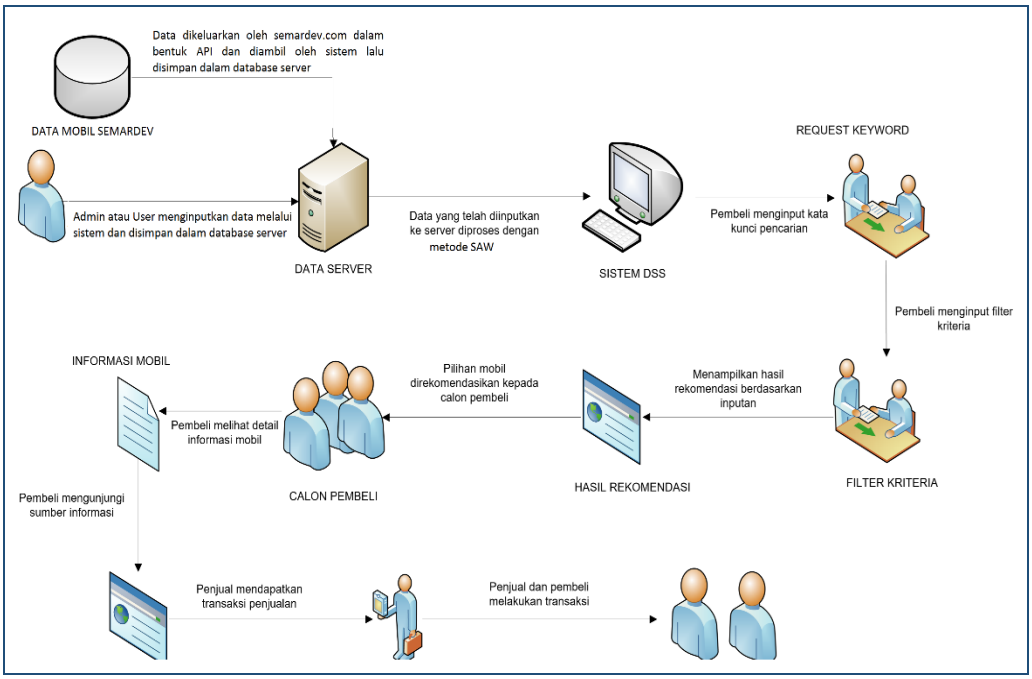
Pada gambar 3.2 di atas menunjukkan bisnis proses lama atau bisnis proses yang sedang berjalan saat ini dimana berikut ialah alur bisnisnya.

1. Data mobil diinputkan oleh admin atau oleh user sebagai penjual mobil dan disimpan kedalam database server.
2. Lalu pembeli mobil yang ingin mencari mobil cukup menginputkan kata kunci pencarian sesuai dengan mobil yang diinginkan.
3. Sistem menampilkan informasi mobil sesuai dengan kata kunci yang diinpukan pembeli.
4. Pembeli mengunjungi atau melihat sumber informasi atau detail informasi dari mobil tersebut untuk menghubungi penjual untuk negosiasi.

5. Selanjutnya penjual mendapatkan transaksi penjualan dan bersiap melakukan COD atau *Cash On Delivery.*
6. Penjual dan pembeli melakukan COD dimana mereka bertemu dan melakukan transaksi jual beli mobil sesuai dengan kesepakatan sebelumnya.
7. Penjual mendapatkan uang dari hasil penjualan mobil dan pembeli mendapatkan mobil yang diinginkan.
8. Jika penjual telah melakukan transaksi dan mobil sudah terjual, maka penjual dapat mengubah status mobil menjadi terjual agar tidak bisa dibeli oleh pembeli lain.

3.2.2 Bisnis proses yang ditawarkan

Untuk lebih memahami mengenai Tugas Akhir yang akan dibuat oleh penulis, berikut akan dijabarkan melalui bisnis proses dari sistem yang akan dibuat yang ditunjukan pada gambar 3.3 berikut yang akan menunjukan bagaimana nantinya alur dari sistem yang akan dikembangkan oleh penulis dalam penelitian tugas akhir ini:



Gambar 3. 3 Bisnis proses yang ditawarkan

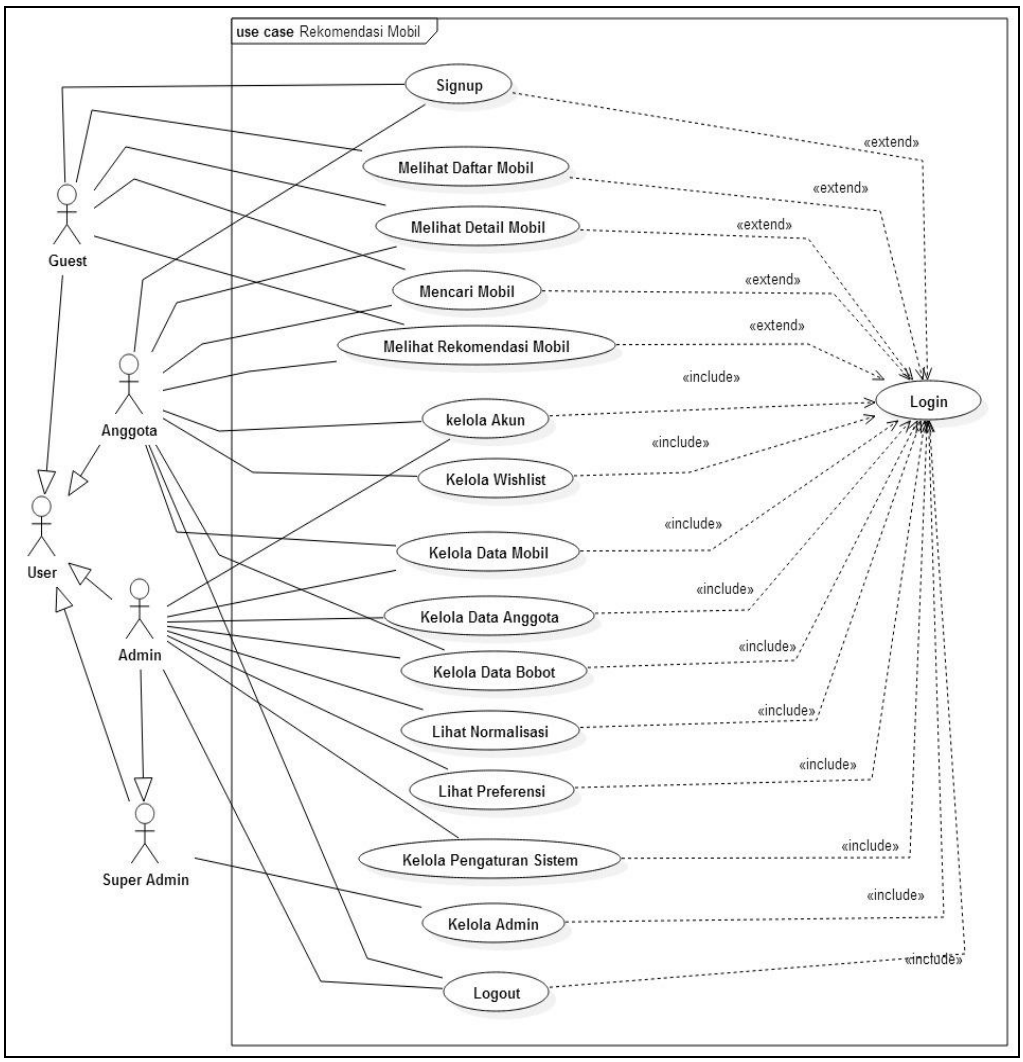
Pada gambar 3.3 di atas menampilkan bisnis proses yang ditawarkan. Dan berikut penjelasan dari setiap alurnya:

1. Untuk bisa menampilkan data mobil, sistem terlebih dahulu harus memiliki data dimana data tersebut yang nantinya akan diolah. Data dari sistem pada dasarnya diperoleh menggunakan HTML DOM lalu disimpan ke dalam database server. Selain itu dari inputan admin maupun user yang ingin menjual mobil kepada pembeli melalui menu yang sudah disediakan, lalu data tersebut akan disimpan pada *database* server.
2. Data yang telah diinputkan tersebut selanjutnya akan diolah sedemikian rupa dengan menggunakan metode SAW untuk mendapatkan data baru berupa data rekomendasi yang akan ditampilkan melalui sistem DSS.
3. Selanjutnya pembeli dapat mencari mobil dengan menginputkan kata kunci sesuai dengan kriteria mobil yang diinginkan dengan melakukan pengisian form kriteria.
4. Sistem akan menampilkan rekomendasi sesuai dengan data yang telah diinputkan oleh pembeli dan ditampilkan dalam bentuk daftar mobil yang direkomendasikan melalui sistem.
5. Pembeli melihat detail mobil yang direkomendasikan, kemudian pembeli dapat mengunjungi sumber informasi baik itu dari contact person penjual, sosial media dan sebagainya untuk melakukan negosiasi dengan pemilik mobil.
6. Setelah sepakat dengan negosiasi harga, pembeli dan penjual melakukan COD atau *Cash On Delivery* dimana selanjutnya penjual bertemu dengan pembeli dan menunjukan mobil yang akan dijual, selanjutnya melakukan transaksi hingga transaksi selesai.
7. Jika penjual telah melakukan transaksi dan mobil sudah terjual, maka penjual dapat mengubah status mobil menjadi terjual agar tidak bisa dibeli oleh pembeli lain.

3.3 Diagram Perancangan

3.3.1 Use Case Diagram

Pada gambar 3.4 berikut ini merupakan diagram *use case* yang akan mempermudah dalam memahami peran dari setiap aktor dalam menggunakan sistem yang akan dibuat oleh penulis.



Gambar 3. 4 Use Case Diagram

Pada gambar 3.4 di atas ialah *usecase* yang digunakan pada sistem untuk menggambarkan fungsi-fungsi apa saja yang dapat diakses oleh setiap aktor yang menggunakan sistem. Pada *usecase* diagram tersebut yang memiliki empat aktor yang menggunakan sistem. Berikut ini merupakan peran dari masing masing aktor.

1. **Guest**

   Aktor ini merupakan pengunjung dari website dan belum termasuk bagian dari sistem. Tidak banyak fungsi yang dapat diakses oleh aktor ini. Fungsi yang dapat di akses antara lain yaitu dapat melakukan *login*, dapat melakukan signup, dapat melihat daftar mobil, melihat detail mobil, mencari mobil dan melihat rekomendasi mobil.

2. **Anggota**

   Aktor ini merupakan pengguna yang telah terdaftar dalam sistem dimana banyak sekali fungsi yang dapat diakses oleh aktor ini. Untuk menjadi anggota pengunjung haruslah mendaftar terlebih dahulu dan stelah terdaftar langsung melakukan *login* pada halaman *login* untuk bisa memasuki sistem.

   Fungsi-fungsi yang dapat diakses oleh aktor ini ialah yaitu *login*, *logout*, melihat daftar mobil, melihat detail mobil, mencari mobil, melihat rekomendasi mobil, mengelola data mobil, mengelola data bobot, mengelola wishlist.

3. **Admin**

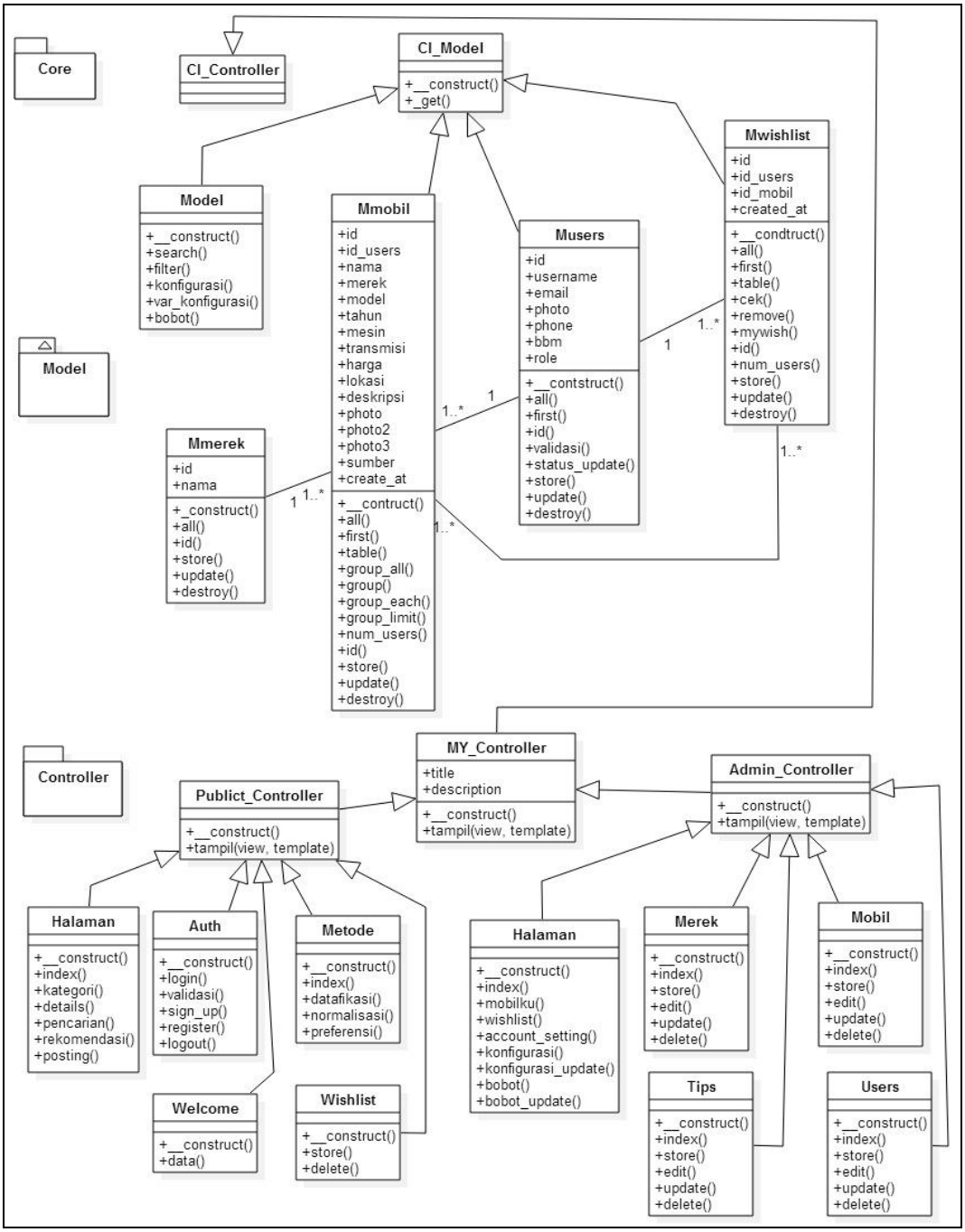
   Admin merupakan aktor yang dapat mengakses fungsi paling banyak dimana hampir semua fungsi dapat diakses oleh admin. Fungsi tersebut ialah *login*, *logout*, melihat daftar mobil, melihat detail mobil, mencari mobil, melihat rekomendasi mobil, mengelola data mobil, mengelola data bobot, mengelola wishlist, melihat normalisasi, melihat preferensi kelola pengaturan sistem.

4. **Super Admin**

   Aktor ini merupakan generalisasi dari aktor admin. Aktor ini dapat mengakses semua fungsi dari sistem dimana semua fungsi tersebut yaitu semua fungsi yang dapat diakses oleh aktor Admin. Yang membedakan yaitu fungsi tambahan yang hanya bisa diakses oleh aktor Super Admin yaitu kelola admin seperti menambah, dan menghapus admin.

3.3.2 Class Diagram

Pada gambar merupakan *class* diagram yang digunakan dalam penelitian tugas akhir dari penulis. Diagram ini digunakan untuk menampilkan kelas – kelas atau paket – paket di dalam sistem dan relasi antar mereka



Gambar 3. 5 *Class diagram*

Pada gambar 3.5 merupakan *class diagram* yang memiliki beberapa kelas yang saling berhubungan satu dengan yang lainnya. Pada setiap kelas memiliki nama kelas, atribut dan operasi (fungsi) yang dapat dilakukan oleh kelas tersebut. Dalam konsep CI, kelas yang terdapat pada gambar merupakan kelas *model* yang bertugas mengelola basis data. Untuk memanipulasi data pada *database* tidak perlu memanggil setiap tabel pada *database* cukup menggunakan *model* sebagai penghubung langsung ke *database*. Untuk menghubungkan satu model dengan model yang lain digunakan konsep *relationship* dimana model dapat dihubungkan *one to one*, *one to many* dan *many to many* tergantung kebutuhan.

Pada *class* model ditulis fungsi – fungsi yang berkaitan dengan database baik itu pengambilan data atau manipulasi data. Tujuannya dibuat fungsi tersebut agar pada controller nantinya lebih mudah membuat *query* tanpa menulis ulang, namun dapat dilakukan dengan memanggil nama fungsi yang ada pada model yang berkaitan dengan *database* yang ingin digunakan.

3.3.3 Sequence Diagram

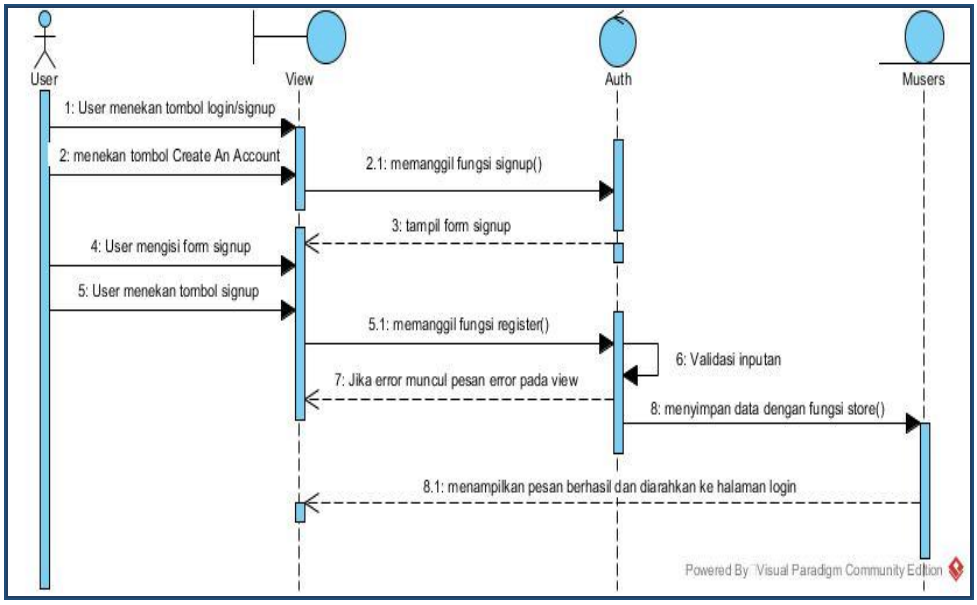
Berikut ini terdabat sequence diagram yang mempermudah memahami pembuatan sistem dalam memahami alur dari setiap fungsi yang dibuat dalam sistem dan juga sebagai acuan dalam pembuatan setiap fungsi dalam sistem. Dalam *sequence diagram* untuk Sistem Pendukung Keputusan Rekomendasi Mobil Menggunakan Metode Simple Additive Weighting (Saw) (Studi kasus: E-commerce Rajamobil.com) terbagi dalam beberapa *sequence diagram* yaitu proses *signup*, melihat daftar mobil, melihat detail mobil, mencari mobil, melihat rekomendasi mobil, kelola akun, kelola *wishlist*, kelola data mobil, kelola data anggota, kelola data bobot, lihat normalisasi, lihat preferensi, kelola pengaturan system, kelola admin dan masih banyak lagi. Untuk lebih jelasnya berikut ini adalah gambaran dari beberapa *sequence diagram* yang terbagi berdasarkan hak aksesnya sebagai berikut:

1. ***Sequence Diagram* semua aktor.**

   Sequence diagram yang dapat diakses oleh semua user yang terdapat pada sistem.

a. *Sequence Diagram sign up*

Berikut merupakan *sequence diagram* untuk registrasi pada sistem informasi rekomendasi mobil.

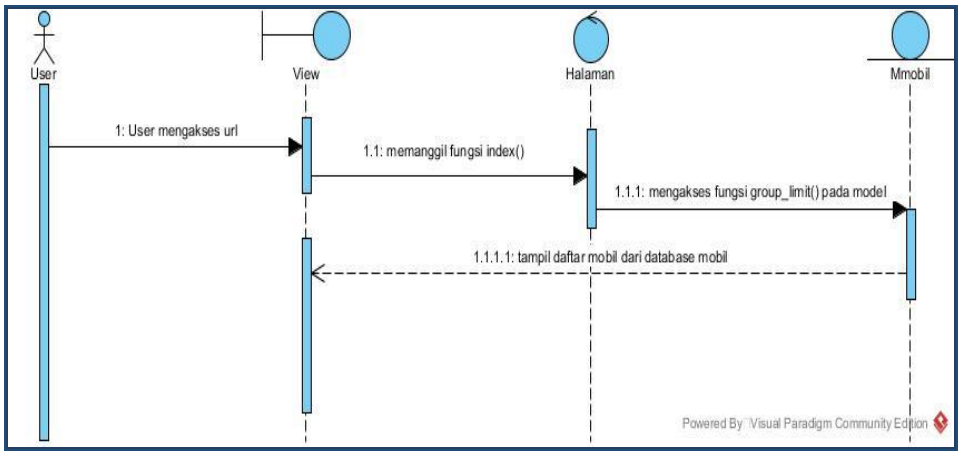


Gambar 3. 6 Sequence Diagram Register

Pada gambar 3.6 diatas terdapat aktor *guest* yang dapat melakukan pendaftaran ke sistem diawali dengan aktor memilih menu *login/logout* dan menekan tombol *create an account* kemudian akan memanggil fungsi *signup()* pada *AuthController* setelah itu akan menampilkan *form signup* pada *view*, proses selanjutnya *guest* mengisi *form register* dan memilih tombol *signup* kemudian data akan dikirim ke *AuthController* pada fungsi *register(),* pada *AuthController* akan dilakukan validasi *input* jika validasi gagal makan akan menampilkan *Alert error* pada *view,* jika validasi berhasil maka akan mengjalankan fungsi *store()* jika berhasil maka data akan disimpan melalui model *muser* bersamaan dengan itu akan ditampilkan *form login* dan *Alert Success* pada *view*.

b. *Sequence Diagram* melihat daftar mobil

Berikut merupakan *sequence diagram* melihat daftar mobil yang terdapat pada sistem informasi rekomendasi mobil.

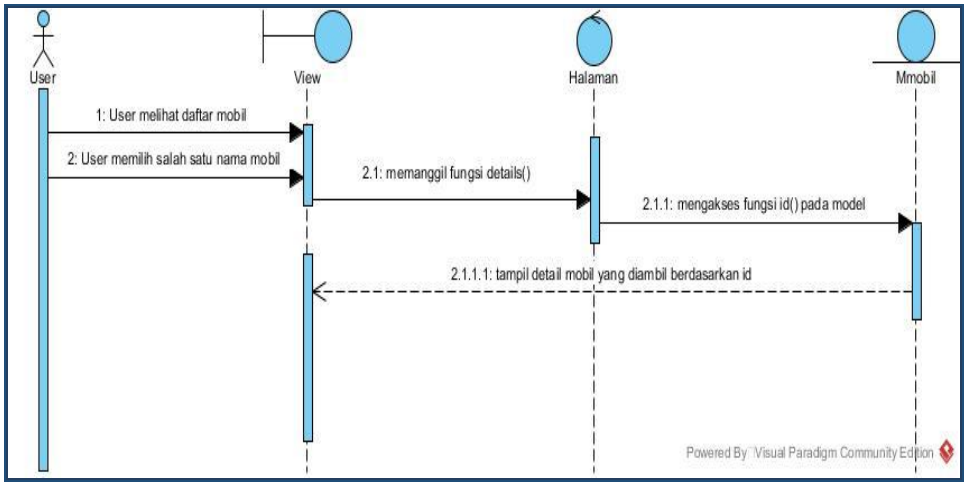


Gambar 3. 7 *Sequence Diagram* melihat daftar mobil

Pada gamabar 3.7 diatas terdapat aktor *guest* yang akan melihat daftar mobil, *sequence diagram* diawali dengan aktor *guest* mengakses url pada *view* kemudian akan memanggil fungsi *index()* ke *model halaman* dan menjalankan fungsi *group_limit()* ke model mobil. Selanjutnya Data daftar mobil akan ditampilkan ke *view*.

c. *Sequence Diagram* melihat detail mobil

Berikut merupakan *sequence diagram* melihat detail mobil yang terdapat pada sistem informasi rekomendasi mobil.



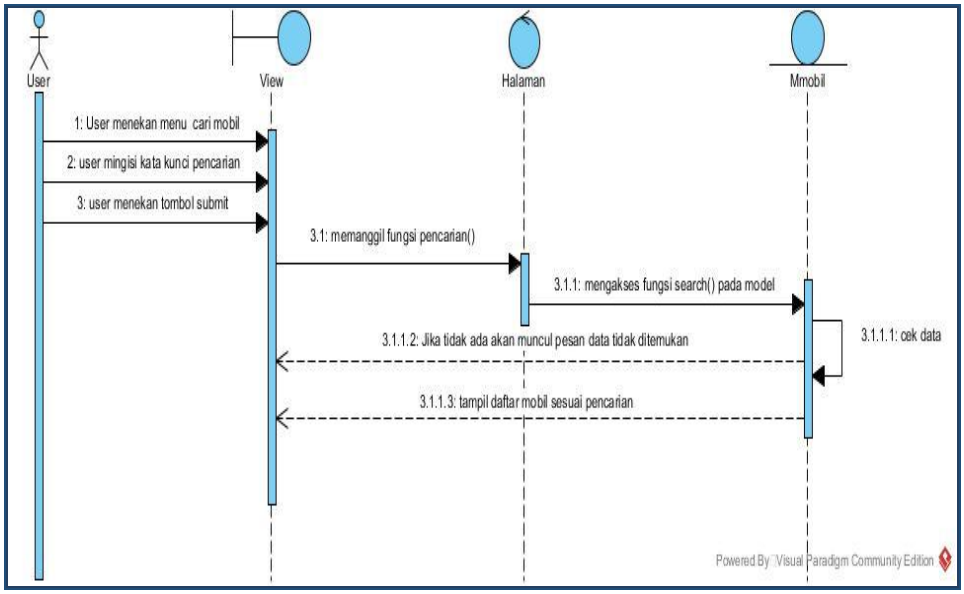
Gambar 3. 8 *Sequence Diagram* melihat detail mobil

Pada gamabar 3.8 diatas terdapat aktor *guest* yang akan melihat detail mobil, *sequence diagram* diawali dengan aktor *guest* melihat daftar mobil dan memilih salah satu mibil pada *view* kemudian akan memanggil fungsi

database() ke *model halaman* dan menjalankan fungsi *id()* ke model mobil. Selanjutnya Data detail mobil akan ditampilkan ke *view* yang diambil berdasarkan id.

d. Sequence Diagram mencari mobil

Berikut merupakan *sequence diagram* mencari mobil yang terdapat pada sistem informasi rekomendasi mobil.

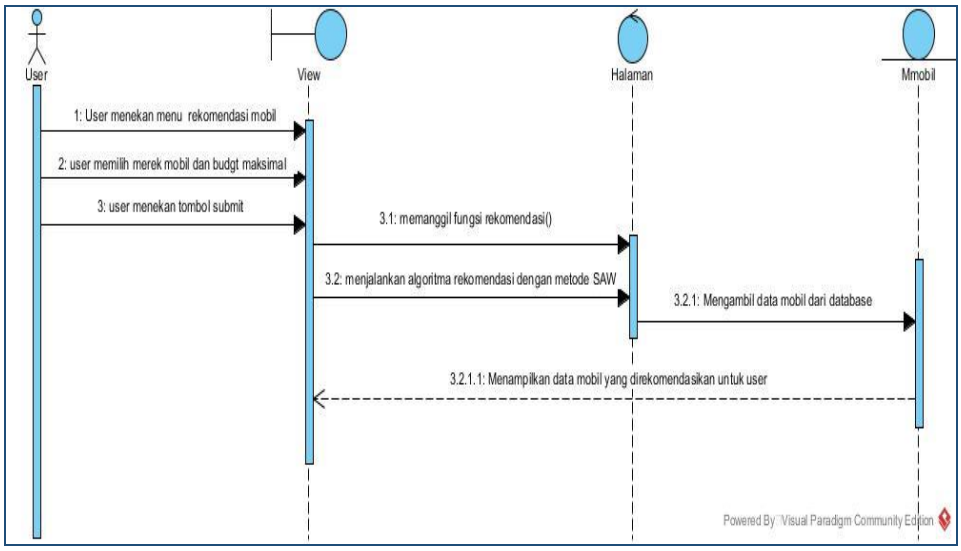


Gambar 3. 9 Sequence Diagram mencari mobil

Pada gamabar 3.9 diatas terdapat aktor *guest* yang akan mencari mobil, *sequence diagram* diawali dengan aktor *guest* menekan menu cari mobil, mengisi kata kunci, dan menekan tombol submit pada *view* kemudian akan memanggil fungsi *pencarian()* ke *model halaman* dan menjalankan fungsi *search()* ke model mobil. Selanjutnya mengecek data akan *input* jika tidak ada akan menampilkan *Alert error* pada *view*, jika validasi berhasil maka dengan itu akan ditampilkan daftar mobil berdasarkan itu dengan *Alert Success* pada *view.*

e. Sequence Diagram melihat rekomendasi mobil

Berikut merupakan *sequence diagram* melihat rekomendasi mobil yang terdapat pada sistem informasi rekomendasi mobil.



Gambar 3. 10 Sequence Diagram melihat rekomendasi mobil

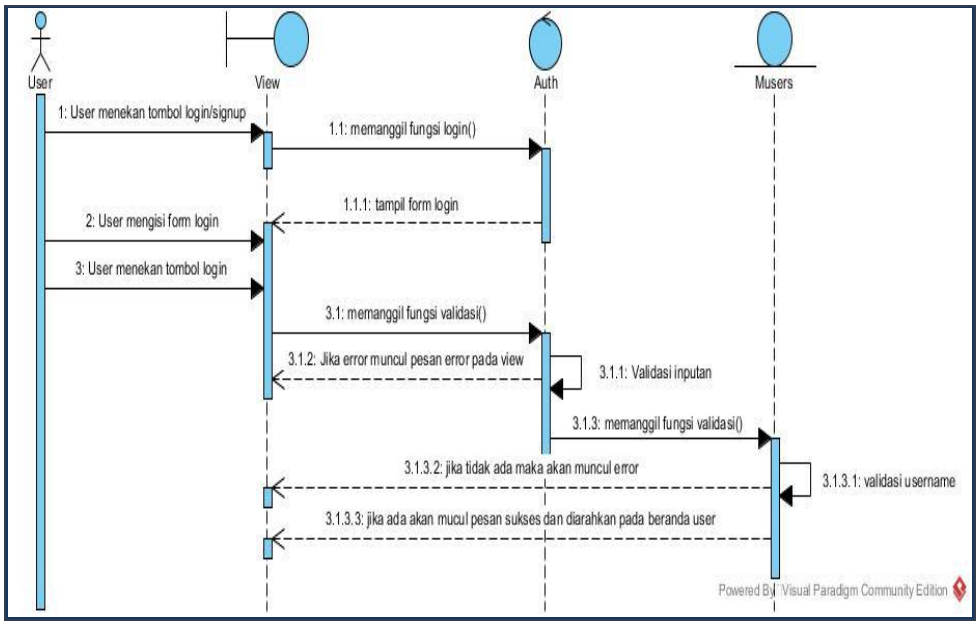
Pada gamabar 3.10 diatas terdapat aktor *guest* yang melihat rekomendasi mobil, *sequence diagram* diawali dengan aktor *guest* menekan menu rekomendasi mobil, memilih merk mobil dan budget maks, dan menekan tombol submit pada *view* kemudian akan memanggil fungsi *rekomendasi()* dan menjalankan algoritma dengan metode SAW ke *model halaman* lemudian mengambil data mobil dari database ke model mobil. Selanjutnya daftar mobil dari hasil rekomendasi ditampilkan pada *view.*

2. *Sequence Diagram* yang diakses oleh user, admin, super admin.

Sequence diagram yang dapat diakses oleh user yang telah terdaftar, admin, dan juga oleh super admin adalah sebagai berikut:

a. Sequence Diagram login

Berikut merupakan *sequence diagram* untuk *login* pada sistem informasi rekomendasi mobil.

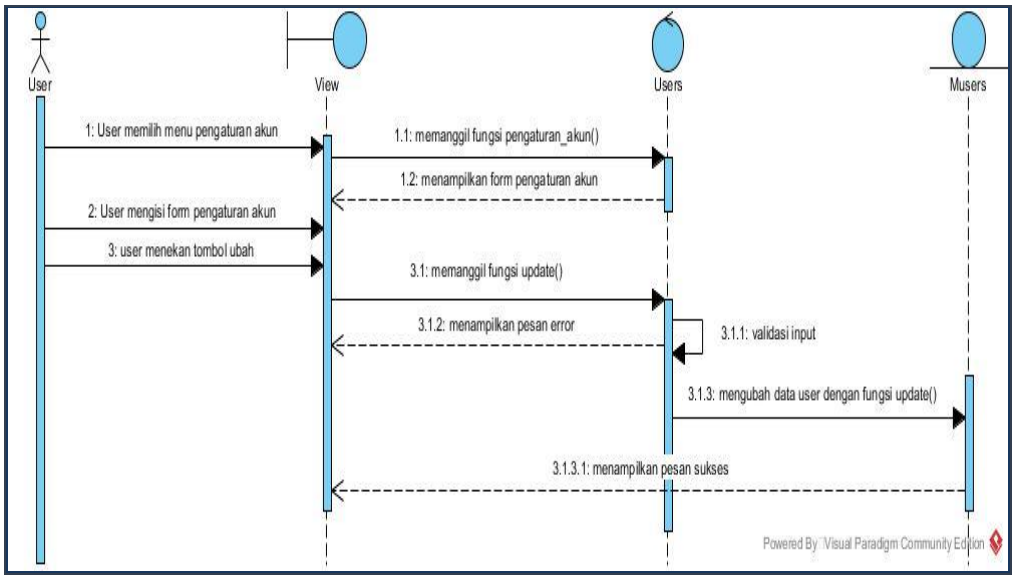


Gambar 3. 11 Sequence *Diagram Login*

Pada gambar 3.11 diatas terdapat aktor *user* yang dapat *login* ke sistem. Sequence *diagram login* diawali degan aktor memilih menu *Login/signup* pada *view* kemudain akan memanggil fungsi *login()* pada *AuthController* akan menampilkan *form login*, kemudaian *user* mengisi *form login* dan menekan tombol *login* setelah itu data akan dikirim ke *AuthController* pada fungsi *validasi()*. Pada *AuthController* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi fungsi *validasi()* pada model *Musers* kemudian akan menampilkan Beranda pada *view* beserta *Alert Success*.

b. Sequence Diagram pengaturan akun

Berikut merupakan *sequence diagram* pengaturan akun pada sistem informasi rekomendasi mobil.

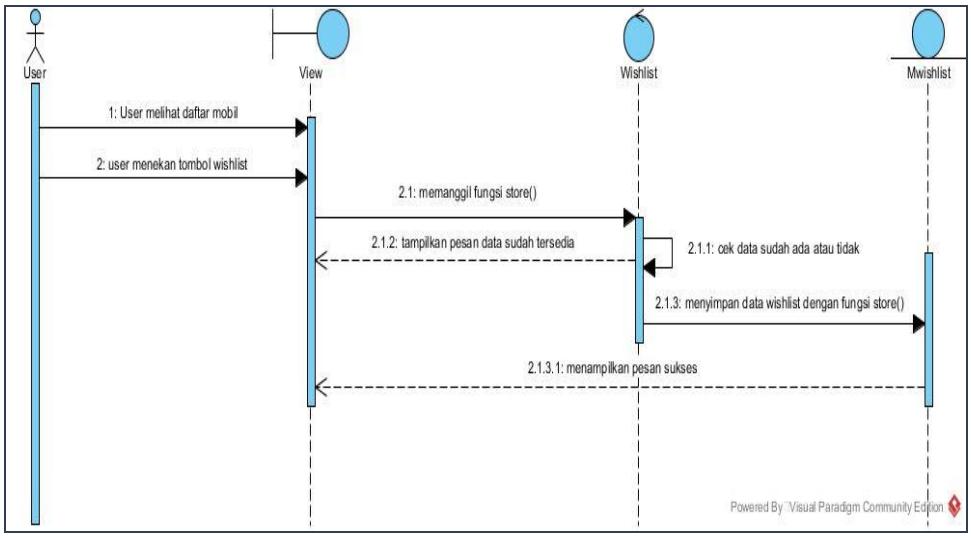


Gambar 3. 12 *Sequence Diagram* pengaturan akun

Pada gambar 3.12 diatas terdapat aktor *user* yang melakukan pengaturan akun ke sistem. *Sequence diagram* pengaturan akun diawali degan aktor memilih menu pengaturan akun pada *view* kemudain akan memanggil fungsi *pengaturan_akun()* pada *AuthController* akan menampilkan *form* pengaturan akun, kemudaian *user* mengisi *form login* dan menekan tombol *submit* setelah itu data akan dikirim ke *model user* pada fungsi *update()*. Pada *model user* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi fungsi *update()* pada model *Musers* kemudian akan menampilkan *Alert Success* pada view.

c. *Sequence Diagram* menambah *wishlist*

Berikut merupakan *sequence diagram* menambah *wishlist* pada sistem informasi rekomendasi mobil.

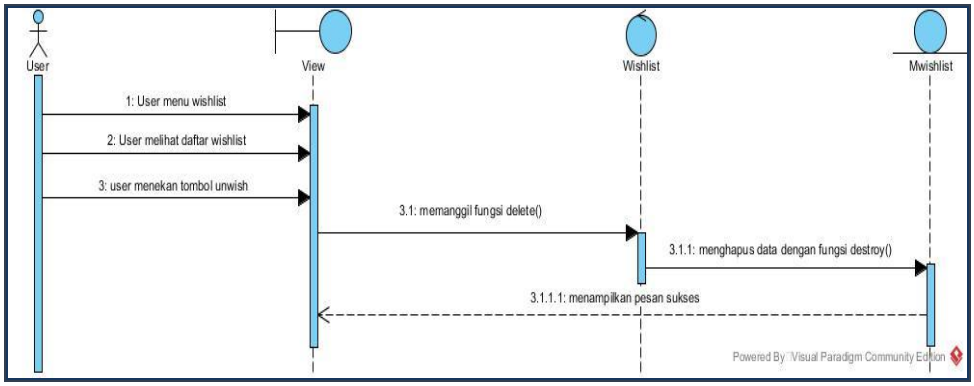


Gambar 3. 13 *Sequence Diagram* menambah *wishlist*

Pada gambar 3.13 diatas terdapat aktor *user* yang ingin menambah *wishlist* ke sistem. *Sequence diagram* menambah *wishlist* diawali degan *user* melihat daftar mobil dan menekan tombol *wishlist* pada *view* kemudain akan memanggil fungsi *store()* pada model *wishlist.* Pada model *wishlist* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi fungsi *store()* pada model *Musers* kemudian akan menampilkan *Alert Success* pada view.

d. Sequence Diagram menghapus *wishlist*

Berikut merupakan *sequence diagram* menghaapus *wishlist* pada sistem informasi rekomendasi mobil.

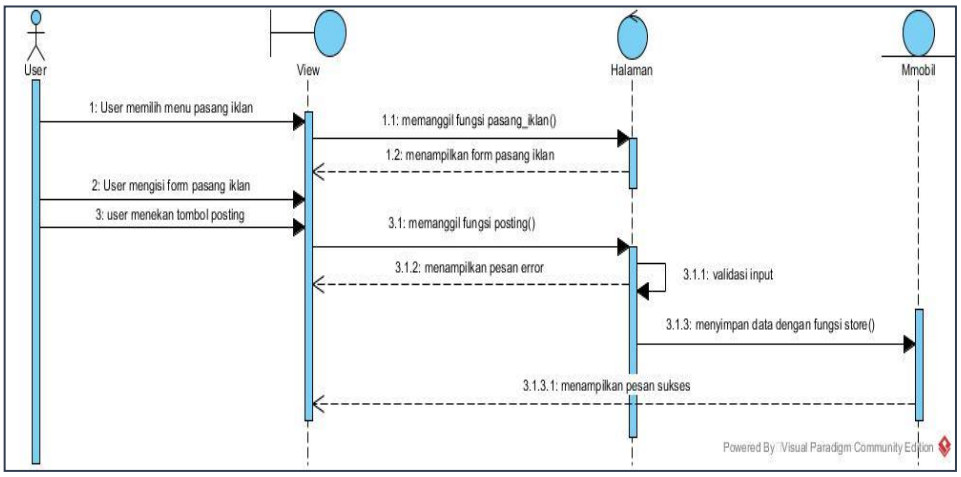


Gambar 3. 14 *Sequence Diagram* menghapus *wishlist*

Pada gambar 3.14 diatas terdapat aktor *user* yang ingin menghapus *wishlist* ke sistem. *Sequence diagram* menghapus *wishlist* diawali degan *user* menekan tombol *wishlist,* melihat daftar *wishlist* kemudian menekan *unwish* pada *view* selanjutnya akan memanggil fungsi *delete()* pada model *wishlist.* Kemudian data akan dihapus dengan menjalankan fungsi *destroy()* pada model *Mwishlist* kemudian akan menampilkan *Alert Success* pada view.

e. *Sequence Diagram* menambah mobil

Berikut merupakan *sequence diagram* menambah mobil pada sistem informasi rekomendasi mobil.



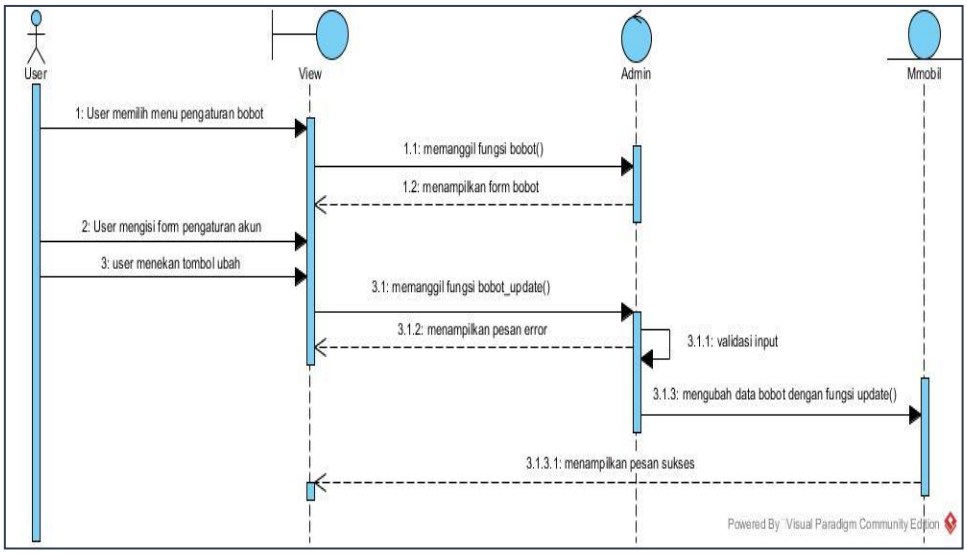
Gambar 3. 15 *Sequence Diagram* menambah mobil

Pada gambar 3.15 diatas terdapat aktor *user* yang ingin menambah mobil ke sistem. *Sequence diagram* menambah mobil diawali degan *user* memilih menu pasang iklan kemudian akan menjalankan fungsi *pasang_iklan()* pada *view* kemudain akan memanggil fungsi *pengaturan_akun()* pada model halaman akan menampilkan *form* pasang iklan, kemudaian *user* mengisi *form login* dan menekan tombol *posting* setelah itu data akan dikirim ke model *user* pada fungsi *posting()*. Pada *model user* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi

fungsi *store()* pada model *Musers* kemudian akan menampilkan *Alert Success* pada view.

f. *Sequence Diagram* pengaturan bobot

Berikut merupakan *sequence* diagram pengaturan bobot pada sistem informasi rekomendasi mobil.



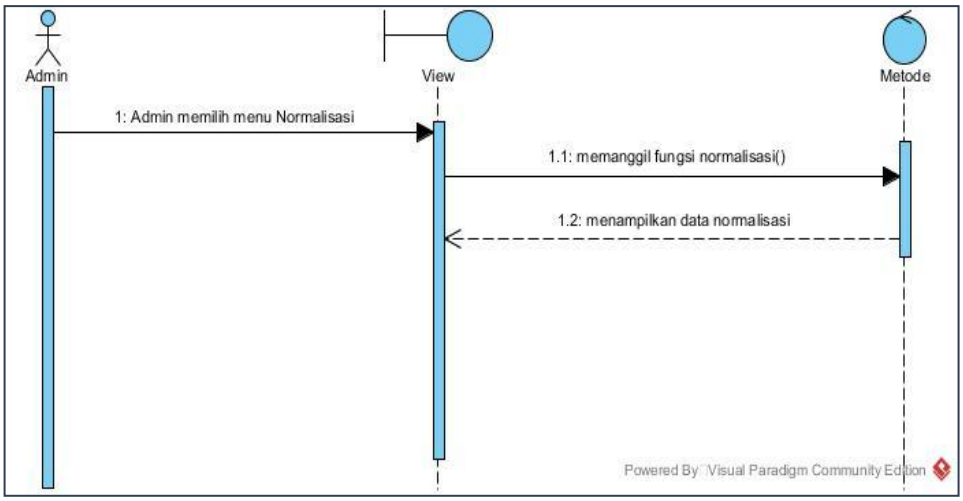
Gambar 3. 16 *Sequence Diagram* pengaturan bobot

Pada gambar 3.16 di atas terdapat aktor *user* yang melakukan pengaturan bobot ke sistem. *Sequence diagram* pengaturan bobot diawali degan *user* memilih menu pengaturan bobot pada *view* kemudain akan memanggil fungsi *bobot()* pada *model admin* akan menampilkan *form* pengaturan bobot, kemudaian *user* mengisi *form bobot* dan menekan tombol ubah setelah itu data akan dikirim ke *model admin* pada fungsi *bobot_updater()*. Pada *model admin* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi fungsi *update()* pada model *mmobil* kemudian akan menampilkan *Alert Success* pada view.

3. *Sequence Diagram* yang diakses oleh admin, super admin.

a. Sequence Diagram melihat normalisasi

Berikut merupakan *sequence diagram* melihat normalisasi yang terdapat pada sistem informasi rekomendasi mobil.

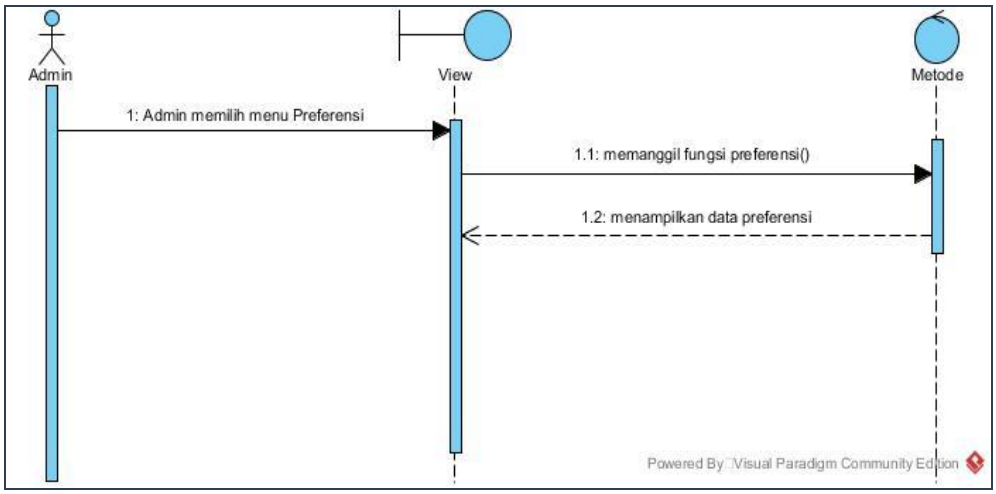


Gambar 3. 17 *Sequence Diagram* melihat normalisasi

Pada gamabar 3.17 di atas terdapat aktor *admin* yang melihat normalisasi, *sequence diagram* diawali dengan aktor *admin* menekan menu normalisasi mobil pada *view* kemudian akan memanggil fungsi *normalisasi().* Selanjutnya daftar normalisasi akan ditampilkan pada *view.*

b. Sequence Diagram melihat preferensi

Berikut merupakan *sequence diagram* melihat preferensi yang terdapat pada sistem informasi rekomendasi mobil.

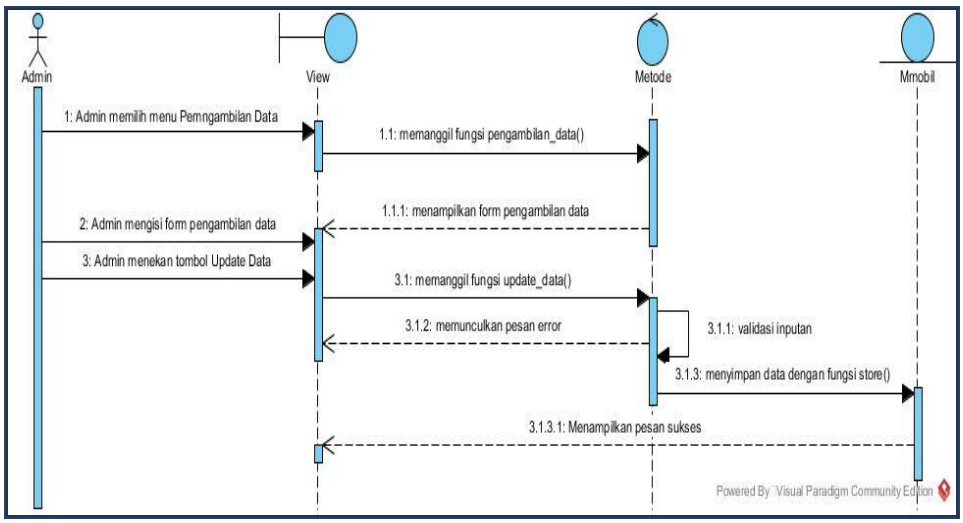


Gambar 3. 18 *Sequence Diagram* melihat preferensi

Pada gamabar 3.18 di atas terdapat aktor *admin* yang melihat preferensi, *sequence diagram* diawali dengan aktor *admin* menekan menu preferensi mobil pada *view* kemudian akan memanggil fungsi *preferensi()*. Selanjutnya daftar normalisasi akan ditampilkan pada *view.*

c. *Sequence Diagram* pengambilan data

Berikut merupakan *sequence diagram* pengambilan data yang terdapat pada sistem informasi rekomendasi mobil.

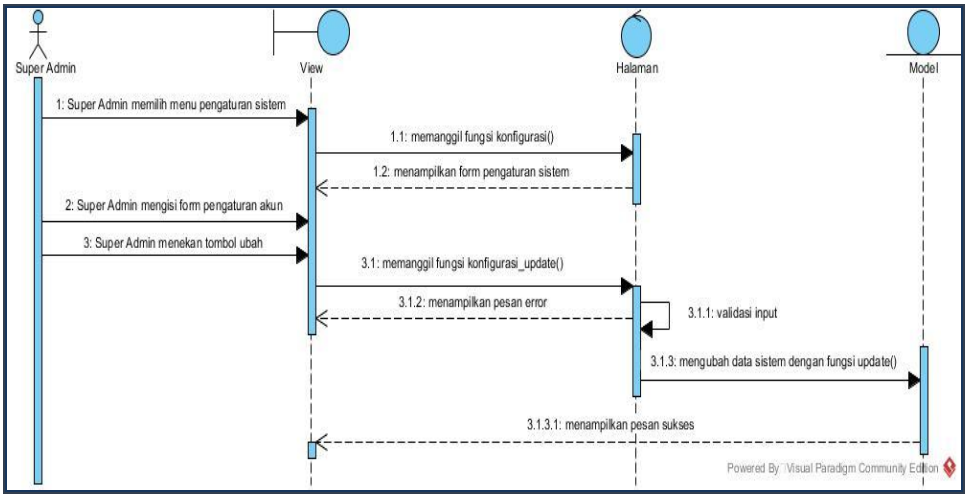


Gambar 3. 19 *Sequence Diagram* pengambilan data

Pada gamabar 3.19 di atas terdapat aktor *admin* yang melakukan pengambilan data, *sequence diagram* diawali dengan aktor *admin* menekan menu pengambilan data pada *view* kemudian akan memanggil fungsi *pengambilan_data()* pada *model metode* akan menampilkan *form* pengambilan data, kemudaian *admin* mengisi *form bobot* dan menekan tombol update data setelah itu data akan dikirim ke *model metode* pada fungsi *bobot_data()*. Pada *model metode* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi fungsi *update()* pada model *mmobil* kemudian akan menampilkan *Alert Success* pada view.

d. Sequence Diagram pengaturan sistem

Berikut merupakan *sequence diagram* pengaturan sistem pada sistem informasi rekomendasi mobil.

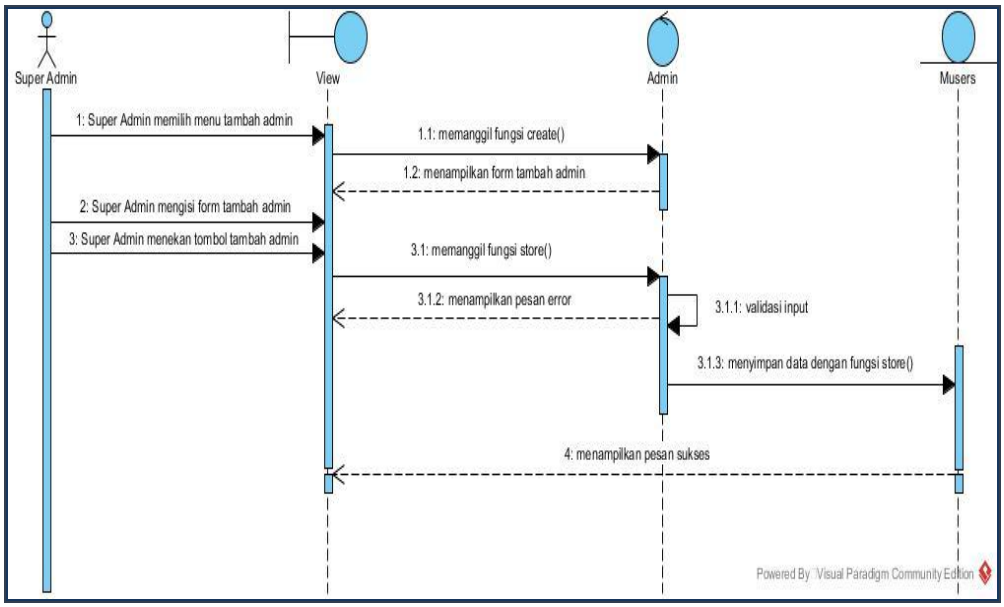


Gambar 3. 20 *Sequence Diagram* pengaturan sistem

Pada gambar 3.20 di atas terdapat aktor *admin* yang melakukan pengaturan bobot ke sistem. *Sequence diagram* pengaturan sistem diawali degan *admin* memilih menu pengaturan sistem pada *view* kemudain akan memanggil fungsi *konfigurasi()* pada *model halaman* akan menampilkan *form* pengaturan sistem, kemudaian *admin* mengisi *form system* dan menekan tombol ubah setelah itu data akan dikirim ke *model halaman* pada fungsi *konfigurasi_update()*. Pada *model halaman* akan di lakukan validasi jika validasi gagal akan menampilkan *Alert error* pada *view* jika validasi berhasil akan maka akan mengeksekusi fungsi *update()* pada model *mmobil* kemudian akan menampilkan *Alert Success* pada view.

4 *Sequence Diagram* yang diakses oleh super admin.

Gambar berikut ini merupakan squence diagram menambah admin yang digunakan untung menggambarkan alur dari proses tambah admin pada sistem.



Gambar 3. 21 *Sequence Diagram* menambah admin

Pada gambar 3.21 di atas terdapat aktor *admin* yang melihat preferensi, awalnya aktor suuper admin yang hendak menambah admin. Dimulai dengan super admin menekan tombol tambah admin pada view kemudian akan memanggil fungsi *create()* pada *controller Admin*, lalu akan menampilkan *form* tambah admin, admin kemudian mengisi *form* lalu menekan tombol tambah admin. Setelah itu akan mengakses fungsi *store()* pada controller Admin, sebelum disimpan terdapat validasi input dimana jika gagal akan menampilkan pesan error dan jika benar akan lanjut untuk menyimpan data pada database users dengan menggunakan fungsi *store()* pada model Musers setelah itu akan ditampilkan pesan sukses.

3.3.4 Desain Tabel Basis Data

1. Tabel bobot

   Tabel 3.1 berikut ini merupakan tabel bobot yang digunakan untuk menyimpan pengaturan bobot preferensi dari setiap user.

Tabel 3. 1 Tabel bobot

bobot

	Field	Type	Extra
P	id	int(11)	Auto Increment
	id_users	int(11)	
	b1	double	
	b2	double	
	b3	double	
	b4	double	
	b5	double	
	b6	double	

Tabel bobot di atas memiliki beberapa kolom diantaranya ialah kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel bobot dengan tipe integer dan panjang data 11 serta auto increment dimana id akan bertambah secara otomatis. Terdapat kolom id_users demham tipe data integer dan panjang data 11 yang digunakan sebagai foreign key dari tabel users dan kolom b1, b2, b3, b4, b5, b6 dengan tipe data double yang digunakan untuk menyimpan data pengaturan bobot dari setiap user.

2. Tabel data

Pada Tabel 3.2 berikut ini merupakan tabel data yang digunakan untuk menyimpan data nilai dari setiap alternatif (data mobil).

Tabel 3. 2 Tabel data

data

	Field	Type	Extra
P	id	int(11)	Auto Increment
	id_mobil	int(11)	
	c1	double	
	c2	double	
	c3	double	
	c4	double	
	c5	double	
	c6	double	

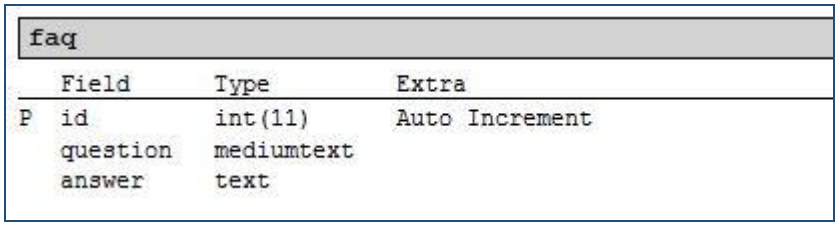
Tabel bobot di atas memiliki beberapa kolom diantaranya ialah kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel data dengan tipe integer dan panjang data 11 serta auto increment dimana id akan bertambah

secara otomatis. Terdapat kolom id_mobil dengan tipe data integer dan panjang data 11 yang digunakan sebagai foreign key dari tabel mobil dan kolom c1, c2, c3, c4, c5, c6 dengan tipe data double yang digunakan untuk menyimpan data nilai dari setiap alternatif (mobil).

3. Tabel faq

Tabel 3.3 berikut merupakan tabel faq yang digunakan untuk menyimpan data pertanyaan serta jawanban dari pertanyaan yang paling sering ditanyakan oleh user.

Tabel 3. 3 Tabel faq



faq

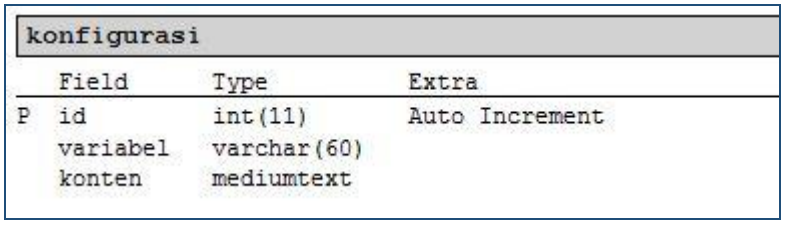
	Field	Type	Extra
P	id	int(11)	Auto Increment
	question	mediumtext	
	answer	text	

Pada tabel faq diatas terdapat kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel faq dengan tipe integer dan panjang data 11 serta auto increment dimana id akan bertambah secara otomatis. Terdapat kolom question yang digunakan untuk menyimpan data pertanyaan. Terdapat kolom answer yang digunakan untuk menyimpan data jawaban dari setiap pertanyaan.

4. Tabel konfigurasi

Tabel 3.4 berikut ini merupakan tabel konfigurasi dimana tabel tersebut digunakan untuk menyimpan data pengturan dari sistem.

Tabel 3. 4 Tabel konfigurasi



konfigurasi

	Field	Type	Extra
P	id	int(11)	Auto Increment
	variabel	varchar(60)	
	konten	mediumtext	

Pada tabel konfigurasi terdapat kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel konfigurasi dengan tipe integer serta auto

increment dimana id akan bertambah secara otomatis. Terdapat kolom variabel dengan tipe data varchar dan dengan panjang nilai 60 yang digunakan untuk menyimpan data umum terkait sistem. Terdapat kolom konten dengan tipe data mediumtext yang digunakan untuk menyimpan data deklarasi setiap kolom variabel.

5. Tabel kriteria

Tabel 3.5 berikut ini merupakan tabel kriteria yang digunakan untuk menyimpan data kriteria yang akan menjadi pembanding dalam rekomendasi mobil.

Tabel 3. 5 Tabel kriteria

kriteria

	Field	Type	Extra
P	id	int(11)	Auto Increment
	nama	varchar(90)	

Pada tabel kriteria terdapat kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel kriteria dengan tipe integer serta auto increment dimana id akan bertambah secara otomatis. Terdapat kolom nama yang digunakan untk menyimpan data nama dari kriteria dengan tipe varchar dengan panjang nilai 90.

6. Tabel merek

Tabel 3.6 berikut ini merupakan tabel merek yang digunakan untuk menyimpan data merek mobil yang akan direkomendasikan.

Tabel 3. 6 Tabel merek

merek

	Field	Type	Extra
P	id	int(11)	Auto Increment
	nama	varchar(100)	

Pada tabel merek terdapat kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel merek dengan tipe integer serta auto increment dimana id akan bertambah secara otomatis. Terdapat kolom nama bertipe data varchar dengan panjang 100 yang digunakan untuk menyimpan data nama mobil.

7. Tabel mobil

Tabel 3.7 berikut ini merupakan tabel mobil yang digunakan untuk menyimpan data mobil yang akan direkomendasikan.

Tabel 3. 7 Tabel mobil

mobil			
	Field	Type	Extra
P	id	int(11)	Auto Increment
	id_users	int(11)	
	nama	varchar(100)	
	merek	varchar(20)	
	model	varchar(50)	
	tahun	int(4)	
	mesin	int(5)	
	transmisi	varchar(5)	
	jarak_tempuh	int(10)	
	harga	int(20)	
	lokasi	varchar(60)	
	deskripsi	text	
	photo	varchar(200)	
	photo2	varchar(200)	
	photo3	varchar(200)	
	sumber	varchar(100)	
	created_at	timestamp ON UPDATE CURRENT_TIMESTAMP	

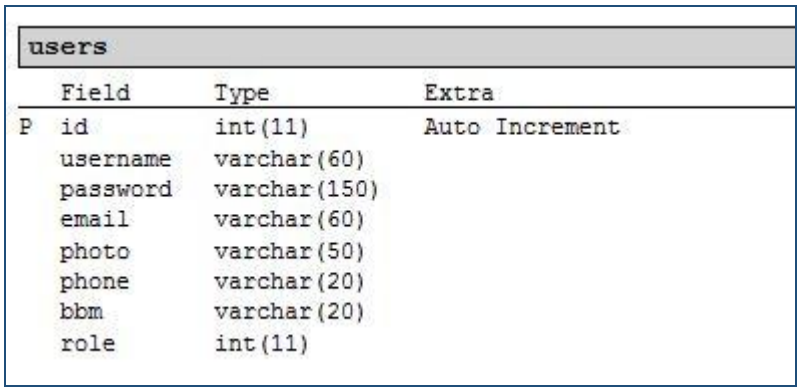
Pada tabel mobil terdapat banyak kolom diantaranya ialah kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel mobil dengan tipe integer serta auto increment dimana id akan bertambah secara otomatis, terdapat kolom id_users sebagai foreign key dari tabel user, terdapat kolom nama untuk menyimpan nama mobil, terdapat kolom merek untuk menyimpan merek dari mobil, terdapat kolom tahun untuk menyimpan data tahun pembuatan mobil, terdapat kolom mesin sebagai penampung kapasitas dari mesin, terdapat kolom transmisi untuk menyimpan data transmisi mobil, terdapat kolom jarak_tempuh untuk menyimpan seberapa jauh mobil digunakan, terdapat kolom harga untuk menyimpan harga, terdapat kolom lokasi untuk menyimpan data lokasi dari mobil, terdapat kolom deskripsi

untuk menyimpan data deskripsi mobil, terdapat kolom photo untuk menyimpan data foto, terdapat kolom sumber untuk menyimpan data kapan dan dimana dan created_at untuk menyimpan tanggal kapan mobil diunggah kedalam sistem.

8. Tabel users

Tabel 3.9 berikut ini merupakan tabel user yang digunakan untuk menyimpan data users.

Tabel 3. 8 Tabel users



users

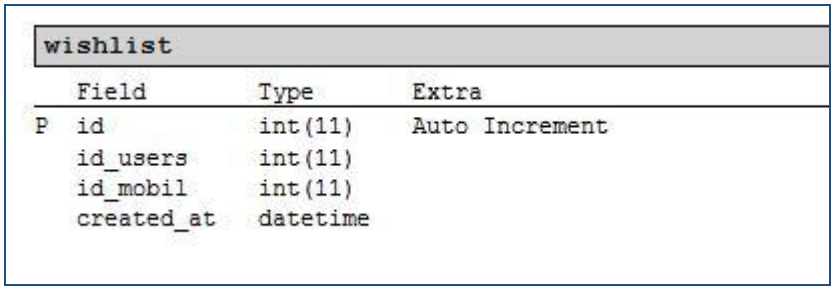
	Field	Type	Extra
P	id	int(11)	Auto Increment
	username	varchar(60)	
	password	varchar(150)	
	email	varchar(60)	
	photo	varchar(50)	
	phone	varchar(20)	
	bbm	varchar(20)	
	role	int(11)	

Pada tabel users di atas ialah kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel users dengan tipe integer serta auto increment dimana id akan bertambah secara otomatis, terdapat kolom username password dan email untuk menyimpan username, password dan email untuk login. Terdapat kolom photo untuk menyimpan foto dari user, terdapat kolom phone untuk nomor terlpon dan bbm untuk pin bbm user serta terdapat kolom role untuk menyimpan data level dari user.

9. Tabel wishlist

Tabel 3.10 berikut merupakan tabel wishlist yang digunakan untuk menyimpan data mobil yang ingin dibeli kemudian hari atau bisa dibilang daftar mobil favorit dari user.

Tabel 3. 9 Tabel wishlist



wishlist

	Field	Type	Extra
P	id	int(11)	Auto Increment
	id_users	int(11)	
	id_mobil	int(11)	
	created_at	datetime	

Pada tabel users di atas ialah kolom id yang berfungsi sebagai kode unik untuk setiap data dari tabel wishliast dengan tipe integer serta auto increment dimana id akan bertambah secara otomatis, terdapat kolom id_users sebagai foreign key dari tabel users, terdapat kolom id_mobil merupakan foreign key dari tabel mobil dan created_at untuk menyimpan tanggal kapan mobil diunggah kedalam sistem.

3.4 Perancangan Antarmuka

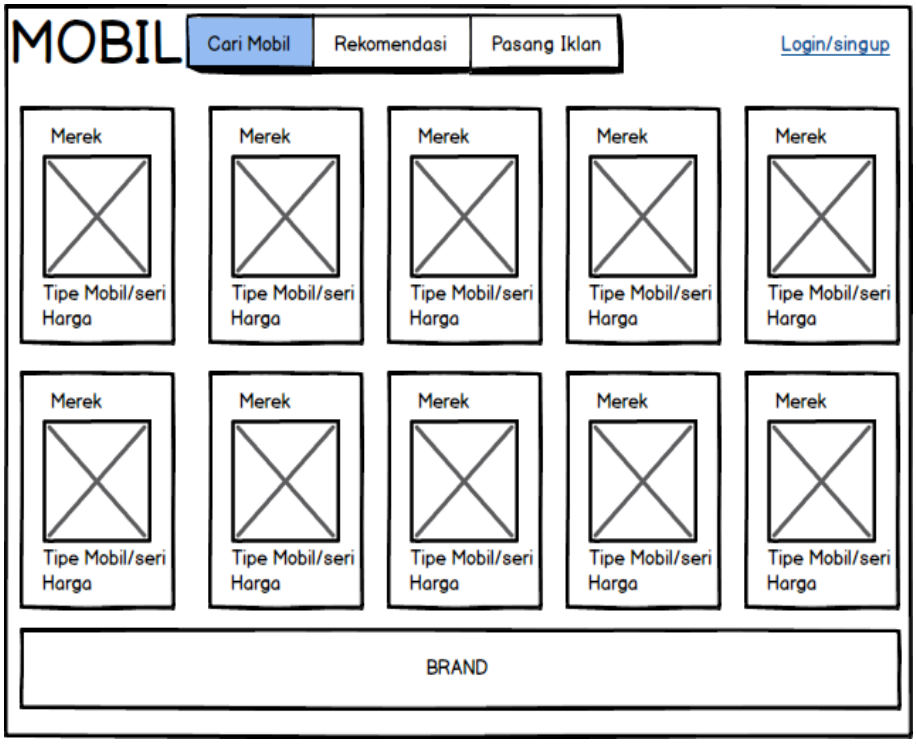
Berikut ini adalah rancangan antarmuka sistem. Dalam sistem ini terdapat dua antar muka yaitu antarmuka admin dan antarmuka user.

3.4.1 Antarmuka User

Berikut ini adalah perancangan antarmukayang dibuat dan di sesuaikan untuk kebutuhan *user*.

1. Halaman Awal

   Pada gambar berikut merupakan halaman utama ketika user mengakses sistem ini dan akan melakukan login/singup.

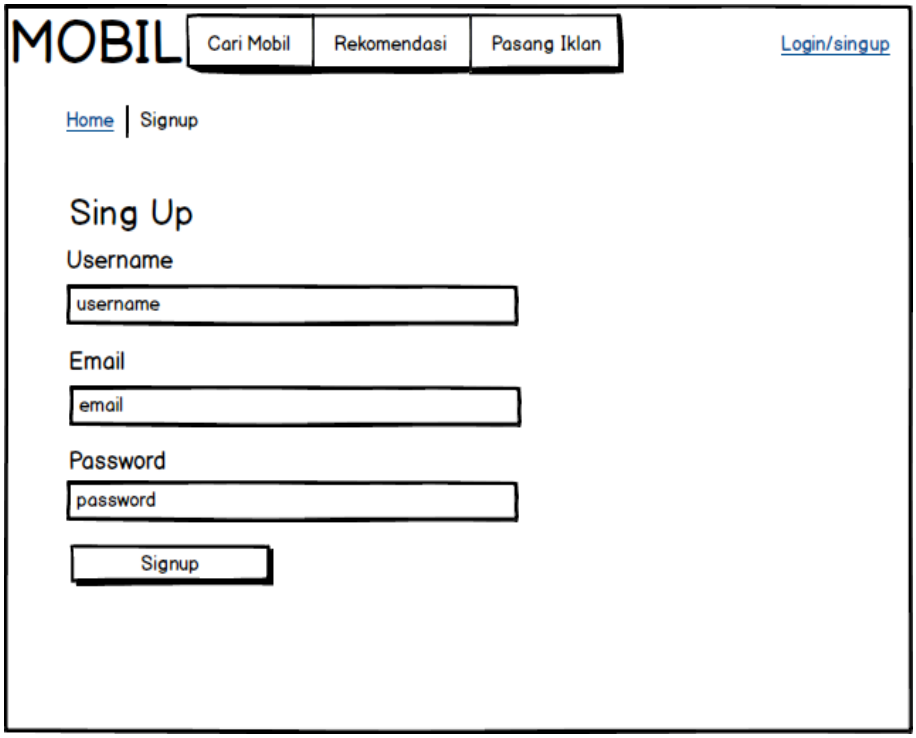


Gambar 3. 22 Halaman awal

Pada gambar 3.22 merupakan halaman awal yang terdapat menu cari mobil yang di gunakan untuk mencari mobil berdasarkan merek mobil dan serinya, pada menu rekomendasi digunakan untuk mencari iklan mobil berdasarkan harga, dan tahun, pada bagian tengah terdapat iklan iklan yang sudah diinputkan.

2. Halaman *Sign up*

Desain *interface* ini adalah halaman singup yang digunakan untuk melakukan pendaftaran ke sistem. Halaman ini di akses melalui menu create acount pada menu login/singup.

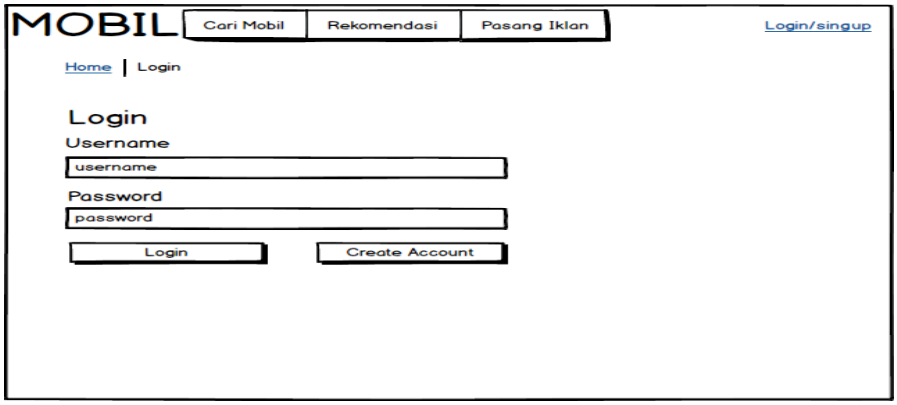


Gambar 3. 23 Singup

Gambar 3.23 merupakan interface Untuk melakukan pendafataran ke sistem, user di haruskan mengisi form yang ada. Form tersebut meluputi username, email dan password dan memilih tombol singup untuk mengirim data – data tersebut ke dalam database.

3. Halaman *Login*

Desain *interface* ini adalah halaman *interface* login yang digunakan untuk masuk ke sistem dengan cara memilih menu login pada bagian kanan atas di halaman utama

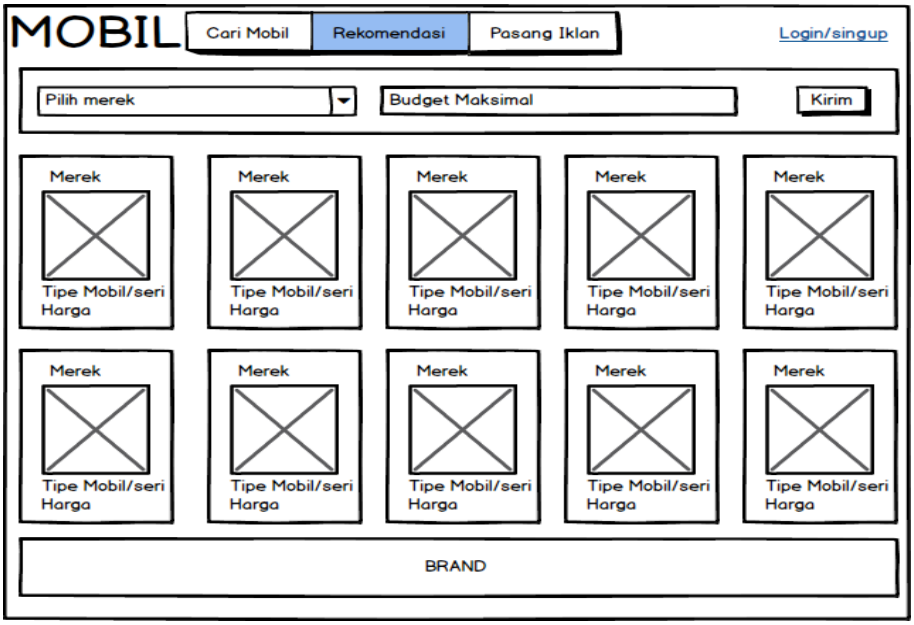


Gambar 3. 24 Halaman Login

Gambar 3.24 merupakan interface Pada halaman login yang terdapat form username, form password, tombol login dan tombol Create Account. Tombol login digunakan untuk masuk ke sistem jika sudah memmpunyai username beserta password, dan create accout digunakan untuk masuk kehalaman singup yang nantinya ada data username dan password akan di gunakan untuk login kesistem.

4. Halaman Rekomendasi

Interface ini merupakan *interface* rekomendasi yang di akases melalui menu pada halaman utama.

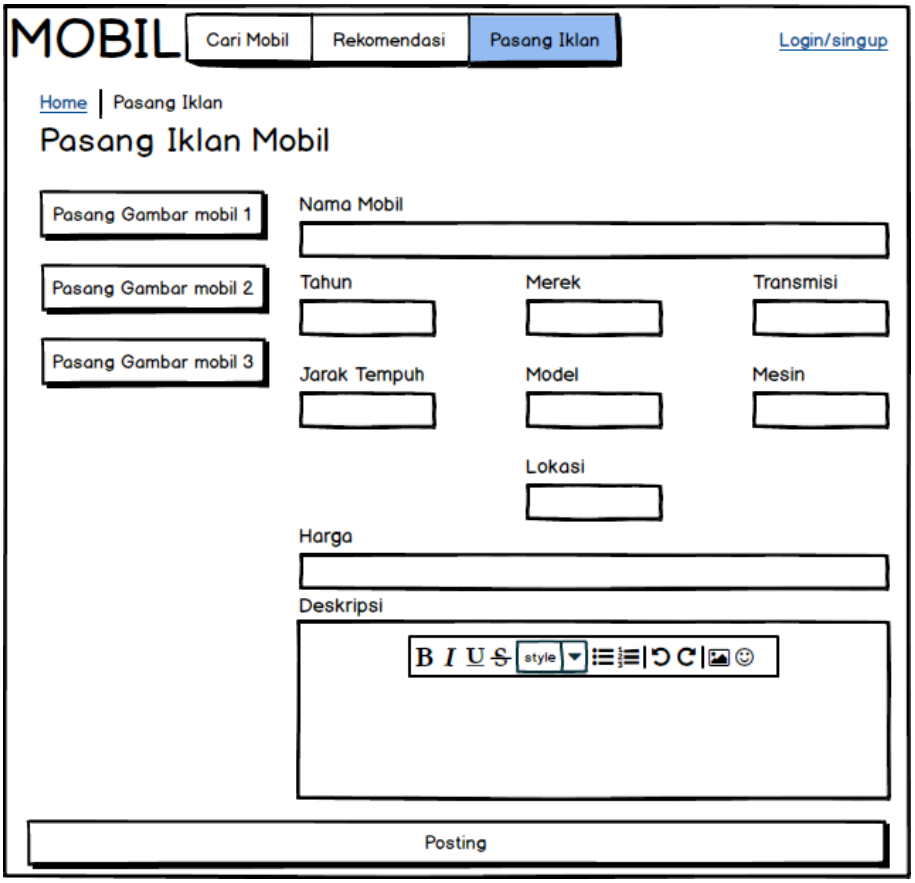


Gambar 3. 25 Rekomendasi

Gambar 3.25 merupakan *interface* Pada halaman rekomendasi yang terdapat menu pilih merek dan *budget* maksimal yang berfungsi untuk mensortir daftar mobil sesuai merek dan harganya, dan terdapat tombol kirim untuk memulai pencarian ikalan mobil yang telah diinputkan.

5. Halaman Pasang iklan

Pada desai *interface* ini dapat di akses melalui menu Pasang iklan yang terdpat pada halaman utama.



Gambar 3. 26 Halaman pasang iklan

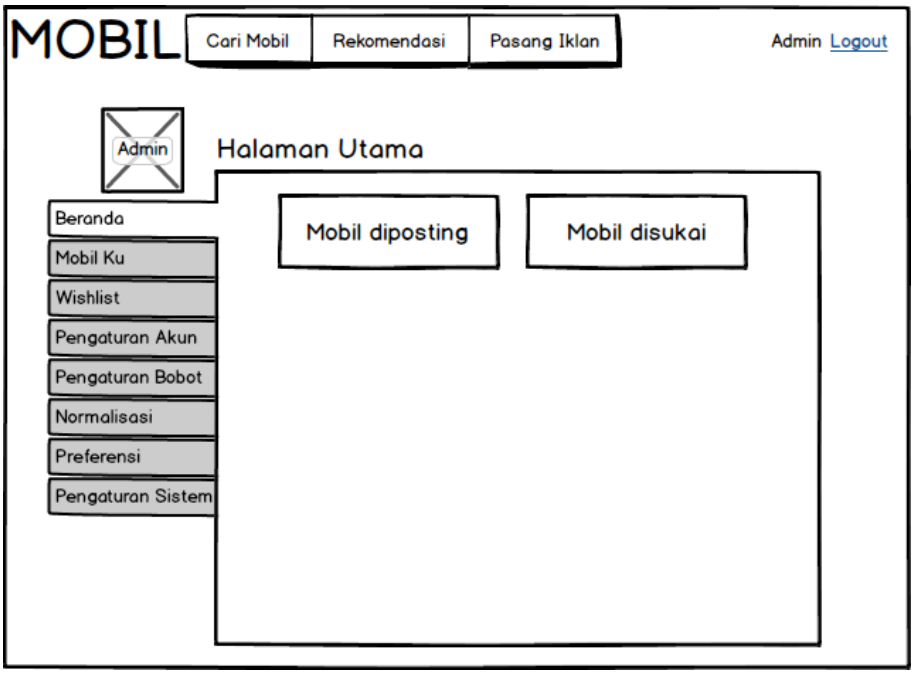
Gambar 3.26 merupakan *interface* Pada halaman pasang iklan yang digunakan utntuk memasang iklan pada sistem dan terdapat beberapa *form* yang harus isi untuk memasang iklan seperti nama mobil, tahun keluaran mobil, transmisim jarak tempuh, model mesin, lokasi, harga dan deskirpsinya, pada bagian kiri terdapat menu untuk menginputkan gambar atau foto mobil yang akan dijual dan terdapat tombol posting untuk mengupload iklan ke sistem.

3.4.2 Antarmuka admin.

Berikut ini adalah perancangan antarmukayang dibuat dan di sesuaikan untuk kebutuhan admin sebagai pengelola sistem.

1. Halaman Beranda Admin

   Interface ini merupakan halaman beranda yang akan muncul ketika admin melakukan *login* ke sistem dengan memilih menu login pada halan utama dengan menginputkan *username* dan *password* dengan level admin.

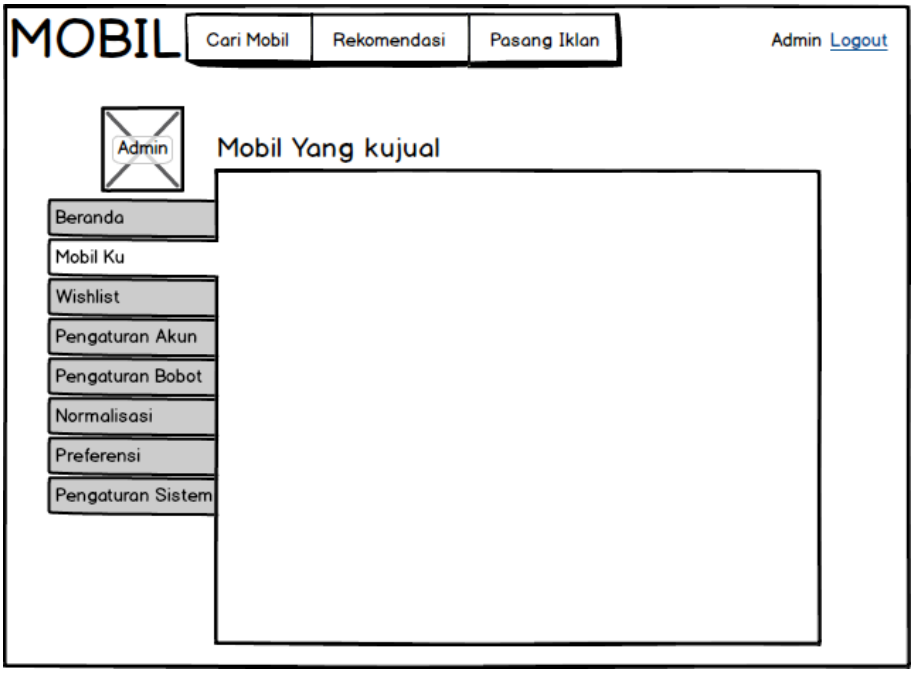


Gambar 3. 27 Halaman utama admin

Gambar 3.27 merupakan *interface* Pada halaman utama admin, yang terdapat Pada halaman ini akan terlihat jumlah mobil yang telah di posting dan mboil yang di sukai pada bagian kanan atas akan ada nama *user* dan tombol *logout* yang akan digunakan untuk keuar dari sistem admin dan akan diarahkan ke halaman utama.

2. Tampilan Mobil yang dijual

   Interface ini merupakan *interface* yang menampilkan daftar mobil – mobil yang akan dijual.

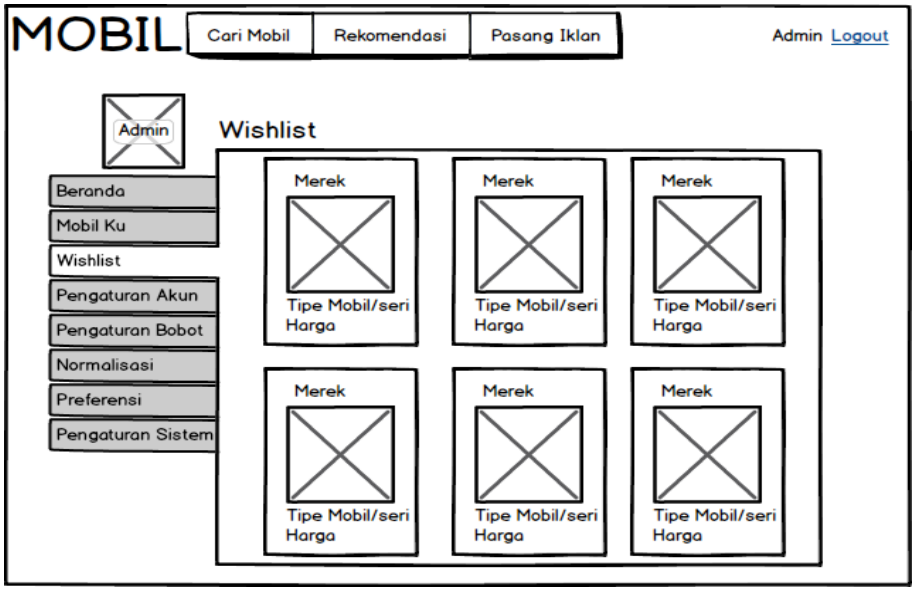


Gambar 3. 28 Mobil yang ku jual

Gambar 3.28 merupakan *interface* Pada halaman mobil yang dijual oleh admin, Pada halaman ini akan terlihat mobil-mobil yang telah di jual oleh admin.

3. Halaman *Wishlist*

Interface ini merupakan *interface* yang menampilkan daftar mobil – mobil yang telah menjadi favorit.

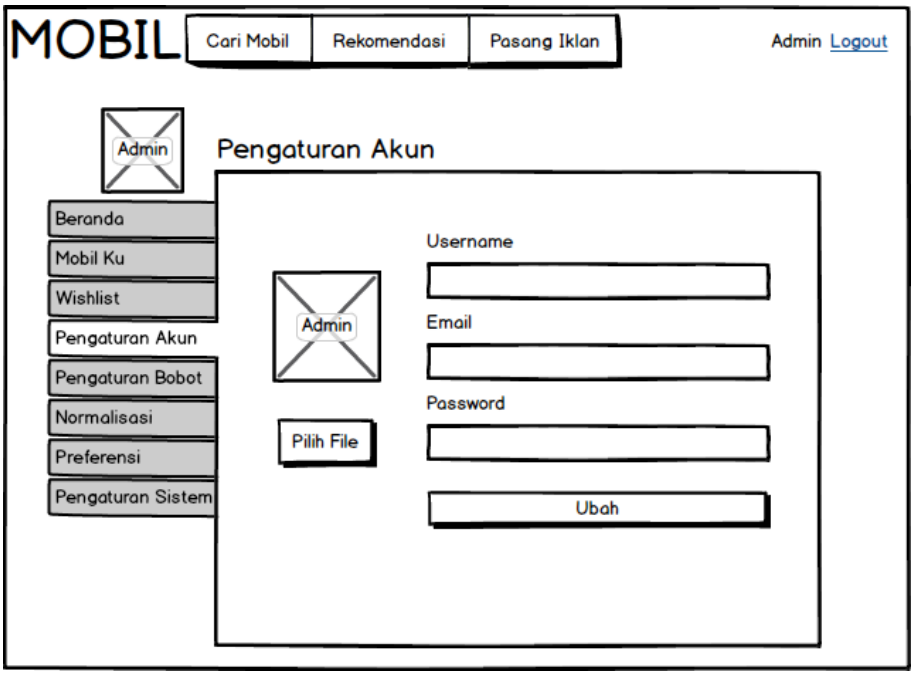


Gambar 3. 29 *Wishlist*

Pada gambar 3.29 merupakan *Interface* yang dapat di akses dengan cara memilih menu *wishlist* yang ada pada sebelah kiri, pada *interfice* ini menampilkan mobil-mobil yang sudah ditandai sebagai mobil favorit.

4. Halaman Pengaturan akun

Interface ini digunakan untuk mengatur akun admin yang dapat diakses dengan cara memilih menu pengaturan akun pada menu bagian kiri.

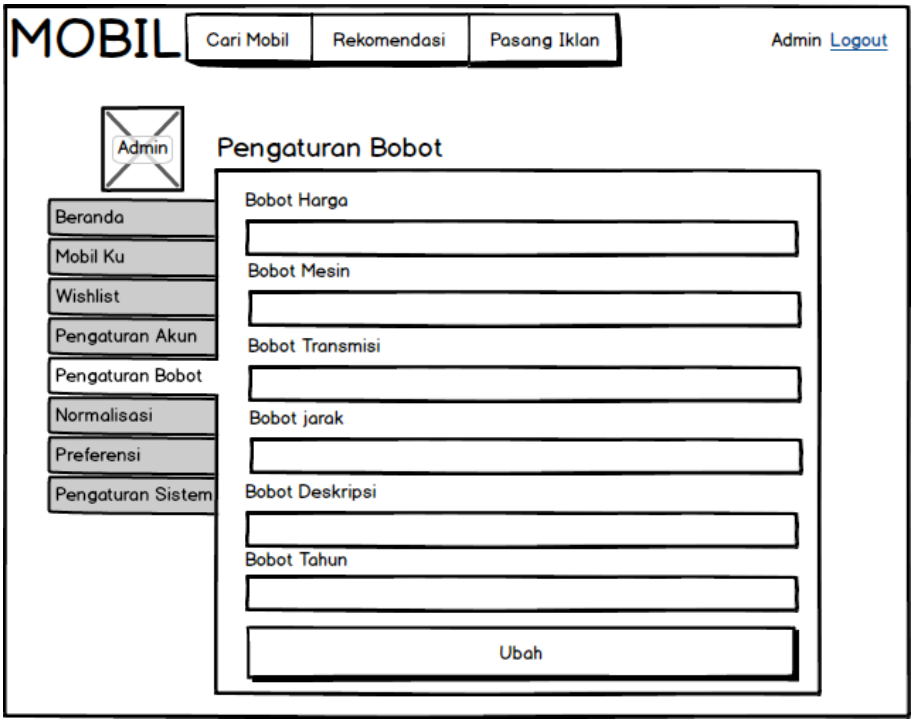


Gambar 3. 30 Pengaturan akun

Pada gambar 3.30 merupakan *Interface* yang terdapat menu pengaturan akun untuk mengganti foto profil admin, merubah username, merubah alamat *email* dan merubah *password* . terdapat juga tombol ubah yang digunakan untuk mengubah data -data admin yang telah di inputkan sebelumnya.

4. Halaman Pengaturan bobot

Interface ini digunakan untuk mengatur bobot yang dapat diakses dengan cara memilih menu pengaturan bobot pada menu bagian kiri.

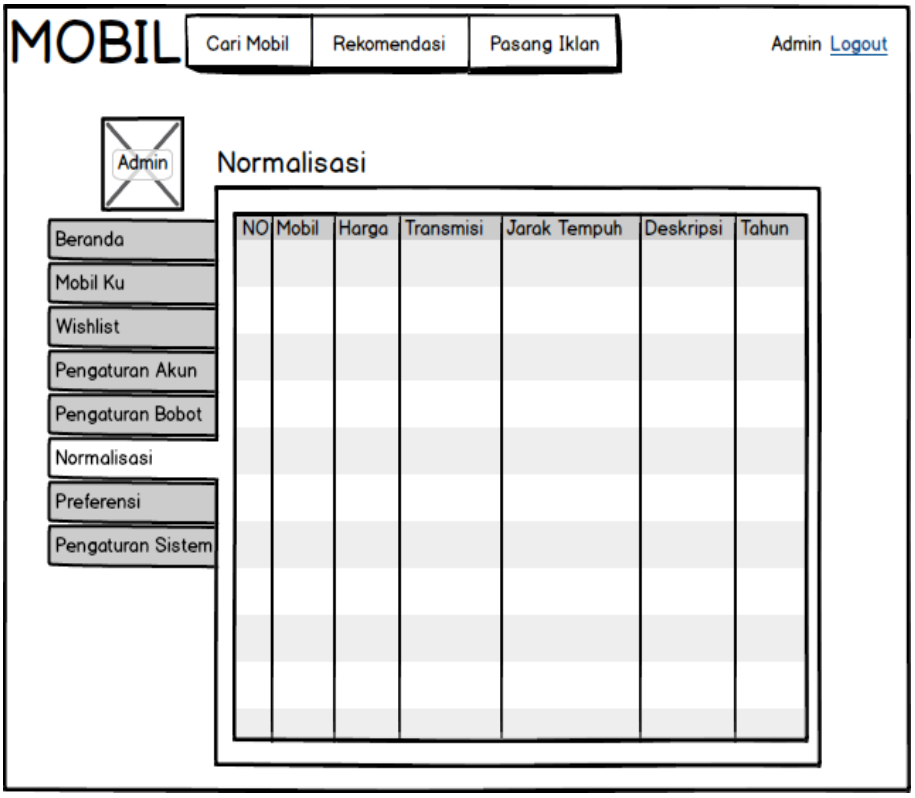


Gambar 3. 31 Pengaturan bobot

Pada gambar 3.31 merupakan *Interface* yang digunakan untuk pengaturan bobot, seperti halnya lebih mempreriotaskan kriteria yang akan dicari dalam melakukan rekomendasi mobil. Dan disarakan total dari bobot untuk totolnya adalah 100%.

5. Halaman Normalisasi

   Interface ini digunakan untuk melihat hasil normalisasi dari metode SAW yang dapat diakses dengan cara memilih menu pengaturan bobot pada menu bagian kiri.

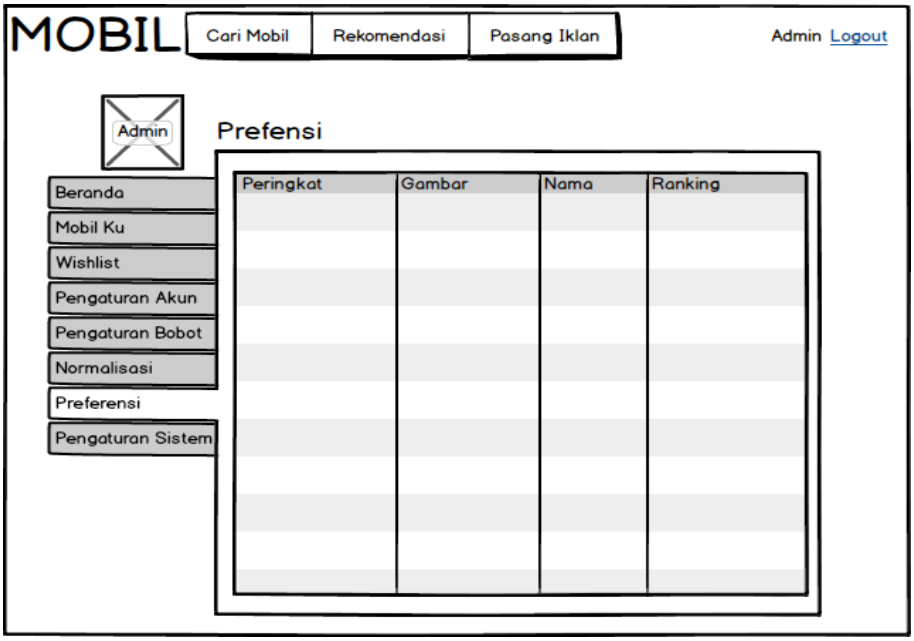


Gambar 3. 32 Normalisasi

Pada gambar 3.32 merupakan *Interface* yang digunakan untuk menampilkan normalisasi. Normalisasi menampilkan data-data yang diperoleh dari hasil *web scrapping*. Adapun data yang ditampilkannya nanti adalah sesuai waktu pengambilan data.

6. Halaman Prefensi

   Interface ini digunakan untuk melihat hasil preferensi dari metode SAW yang dapat diakses dengan cara memilih menu pengaturan bobot pada menu bagian kiri.

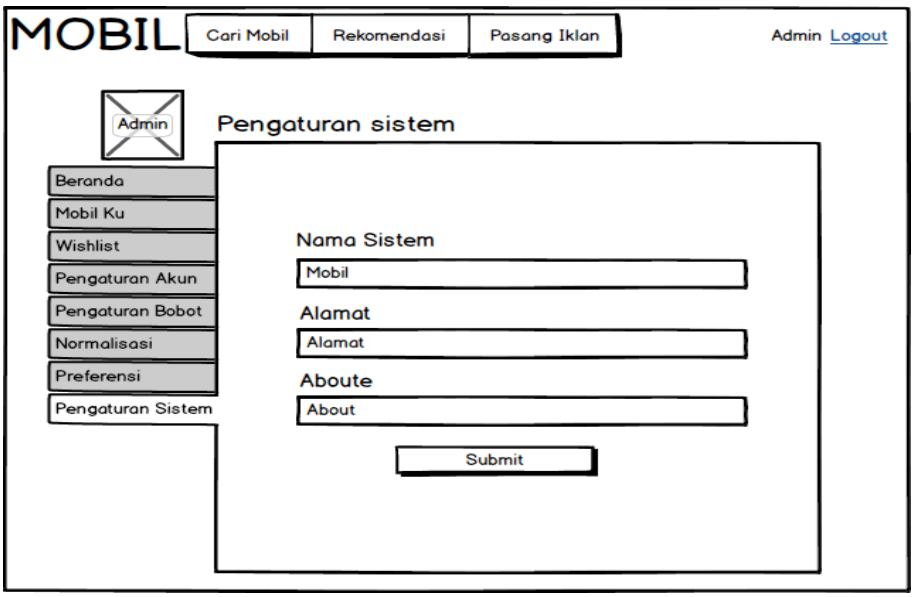


Gambar 3. 33 Perfensi

Pada gambar 3.33 mrupakan *Interface* yang digunakan untuk melihat hasil preferensi peringkat berdasarkan kategori-kategori dari metode SAW.

7. Halaman Pengaturan sistem

Interface ini merupakan interface yang digunakan untyk mengatur sistem yang dapat diakses melalui menu pada sebelah kiri



Gambar 3. 34 Pengaturan sistem

Pada gambar 3.34 merupakan *interface* yang digunakan untuk pengatruan sistem. Pada *interface* ini terdapat menu atau *form* yang di gunakan untuk menginputkan nama sistem, alamatnya dan tentang sistem tersebut atau perusahaanya, dan memilih submit untuk mengubah pengaturannya.

3.5 Scrapping Data

Data didapatkan dari situs *E-commerce* rajamobil.com, untuk lebih spesifik, data yang akan diolah didapatkan dari hasil pencarian situs rajamobil. Setiap mobil akan dimasukkan ke dalam *database*, berdasarkan nomor halaman data dapat dimasukkan oleh admin. Berikut merupakan *coding* dari proses *scrapping*:

```
echo '<a href='.site_url("metode/datafikasi").'>Ubah ke
    Numerik</a>';
date_default_timezone_set("Asia/Jakarta");
$page = $_GET['page'];
$url=
"https://www.rajamobil.com/jual/mobil/bekas?page=".$page;
$html = file_get_html($url);
 $ret = $html->find('.list-mobil');
$no = 0;
 foreach ($ret as $get) {
.$link = 'http://www.rajamobil.com'.$get->find('a', 0)->href;
.$merek = $get->find('.firstTitle', 0)->plaintext;
.$model = $get->find('.secondTitle', 0)->plaintext;
.$name = $merek." ".$model;
.$tahun = filter_var(substr($model,FILTER_SANITIZE_NUMBER_INT);
.$harga = filter_var($get->find('.thirdTitle', 0)
.>plaintext,FILTER_SANITIZE_NUMBER_INT);
.$ul = $get->find('ul.umum-spek');
.foreach($ul as $a){
.$mesin = $a->find('li',1)->plaintext;
.$transmisi = $a->find('li',2)->plaintext;
.$jarak_tempuh = filter_var($a->find('li',3)
.>plaintext,FILTER_SANITIZE_NUMBER_INT);}
.$url = file_get_html($link);
.$det = $url->find('.nop');
.foreach ($det as $details) {
.$gambar = $details->find('li img',1)->src;
.$gambar2 = $details->find('li img',3)->src;
.$gambar3 = $details->find('li img',5)->src;
```

```
. $content = $details->find('.iklan .h-deskripsi',0)
.>plaintext;
.$lokasi = $details->find('.lokasi .kota',0)->title;
.$created_at = date('Y-m-d H:i:s');
.$slug = strtolower(url_title($name));
.$array1 = array(
.'harga' => $harga,);
.$array2 = array(
.'id_users'=> $this->session->userdata('id'),
.'nama' => $name,
.'merek' => $merek,
.'model' => $model,
.'tahun' => $tahun,
.'mesin' => $mesin,
.'transmisi' => $transmisi,
.'harga' => $harga,
.'jarak_tempuh' => $jarak_tempuh,
.'lokasi' => $lokasi,
.'deskripsi' => $content,
.'lokasi' => $lokasi,
.'photo' => $gambar,
.'photo2' => $gambar2,
.'photo3' => $gambar3,
.'sumber' => $link,
.'created_at' => $created_at);
.$cek = $this->db->where('nama',$name)->get('mobil');
.if ($cek->num_rows() > 0){
.$this->db->where('id',$cek->row()->id)
.>update('mobil',$array1);
.echo '<span style="color: red">Mobil '. $name.' Sudah
.Ada</span></br>';
.} else {
.$this->db->insert('mobil',$array2);
.echo '<span style="color: green">Berhasil menambahkan
.'.$name.' </span></br>';}
.}
```

Penjelasan:

- Baris 1 dan 2 merupakan hasil dari *scprapping* yang kemudian akan dijadikan ke numerik.
- Baris 3 sampai 7 merupakan perintah untuk mendapatkan data mobil bekas dari rajamobil.com
- Baris 10 merupakan alamat *URl* pencarian situs rajamobil.com.

- Baris 11 sampai 35 merupakan perintah untuk mendapatkan nilai kriteria dari merek hingga deskripsi pada situs rajamobil.com, dan di simpan dalam bentuk variabel nama-nama kriteria.
- Baris 36 – 53 merupakan penyimpanan variabel-variabel yang didapatkan kedalam bentuk array, supaya lebih mudah untuk dimasukkan kedalam *database*.
- Baris 54 sampai 61 merupakan proses *update* jika mobil telah ada didalam *database* dan *insert* jika mobil belum ada ke dalam *database*.
- Baris 62 merupakan pesan berhasil bahwa mobil telah berhasil ditambahkan.

3.6 Perhitungan SAW

3.6.1 Himpunan Variable Kriteria

Himpunan kriteria yang terdapat pada sistem pendukung keputusan dengan metode SAW adalah sebagai berikut:

Tabel 3. 10 Tabel kriteria

Kode Kriteria	Ketentuan Kriteria
C1	Harga mobil
C2	Mesin mobil
C3	Transmisi mobil
C4	Jarak mobil
C5	Deskripsi Mobil
C6	Tahun mobil

3.6.2 Menentukan Bobot Default

Pada tabel 3.11 merupakan tabel yang menentukan bobot sebagai *default* yaitu bejumlah 1. Adapun untuk tipe dari kriteria dibedakan menjasi dua, diantaranya cost dan benefit. Apabila nilai dari kriteria bernilai tinggi tapi merugikan bagi pembeli maka disebut cost, dan sebaliknya apabila nilai dari kriteria tinggi dan menguntungkan maka disebut benefit.

Tabel 3. 11 Tabel Bobot

Kode Kriteria	Tipe	Bobot
C1	Cost	0,30
C2	Benefit	0,20
C3	Benefit	0,15
C4	Cost	0,15
C5	Benefit	0,10
C6	Benefit	0,10

3.6.3 Tabel Kecocokan

Tabel 3.12 merupakan tabel kecocokan yang menjelaskan dari masing-masing kriteria tersebut akan ditentukan sub-sub kreiterianya. Pada sub kriteria terbagi menjadi beberapa bilangan-bilangan fuzzy. Dari masing-masing bobot tersebut, dibuat suatu variabel yang akan dirubah ke dalam bilangan fuzzy dengan rumus yaitu variabel ken/banyaknya sub kriteria-1.

Contohnya : terdapat sub kriteria 5, maka dicari bobot sub kriterianya.

0/(5-1)=0/4=0 → dikenfersikan menjadi 0,01 agar nilai yang keluar tidak menghasilkan *error*.

1/(5-1)=1/4=0,25

2/(5-1)=2/4=0,5

3(5-1)=3/4=0,75

4/(5-1)=4/4=1

Adapun Penentuan nilai bobot ditentukan pengamatan terhadap situs *E-commerce*, dan wawancara terhadap beberapa pengguna *E-commerce,* atau ditentukan oleh pengguna. Adapun nilai bobot preferensi yang diterapkan secara *default* pada sistem pendukung keputusan ini yaitu, harga 30%, mesin 20%, transmisi 15%, jarak 15%, deskripsi 10%,dan tahun 10%. Pada sistem nilai bobot preferensi sepenuhnya dapat diatur oleh pengguna yang telah menjadi member.

Tabel 3. 12 Tabel kecocokan

Kode Kriteria	**Sub-Kriteria**	**Range**	**Bobot**
C1	Sangat murah	0 dan <=50.000.000	0,01
	Murah	>50.000.000 dan <=100.000.000	025
	Sedang	>100.000.000 dan <=200.000.000	0,5
	Mahal	>200.000.000 dan <=300.000.000	0,75
	Sangat mahal	<=500.000.000	1
C2	Kecil	0-1000	0,01
	Sedang	1000-2000	0,5
	Besar	>=2000	1
C3	Manual	M/T	0,01
	Automatic	A/T	1
C4	Sangat dekat	0-2000	0,01
	Dekat	2000-10.000	0,25
	Terjangkau	10.000-50.000	0,5
	Jauh	50.000-100.000	0,75
	Sangat jauh	>100.000	1
C5	Tidak jelas	0-50	0,01
	Kurang Jelas	50-200	0,3
	Jelas	200-1000	0,6
	Sangat jelas	>=2000	1
C6	Lama	0-20	0,01
	Dekade	20-40	0,3
	5 tahun terakhir	40-60	0,6
	Tahun ini	60-80	1

3.6.4 Perhitungan Normalisasi

Berikut merupakan gambaran dari algoritma perhitungan normalisasi

```
public function normalisasi()
    {
        $max = $this->db->select_max('c2','maxC2') //mesin
            ->select_max('c3','maxC3') //transmisi
            ->select_max('c5','maxC5') //deskripsi
            ->select_max('c6','maxC6') //tahun
            ->join('mobil','mobil.id = data.id_mobil')
            ->limit(50)
            ->where('data.id <',50)
            ->get('data');
        $min = $this->db->select_min('c1','minC1') //harga
            ->select_min('c4','minC4') //jarak_tempuh
            ->join('mobil','mobil.id = data.id_mobil')
            ->limit(50)
            ->where('data.id <',50)
            ->get('data');
        $this->data['max'] = $max->row_array();
        $this->data['min'] = $min->row_array();
        $this->data['kriteria']          =          $this->db-
>join('mobil','mobil.id = data.id_mobil')
            ->limit(50)
            ->where('data.id <',50)
            ->get('data');
    }
<?php
                $no = 0;
                foreach($kriteria->result_array() as $dt2) {
                    echo "<tr>
             <td>".$dt2['id_mobil']."</td>
             <td>".$this->db->where('id',$dt2['id_mobil'])-
>get('mobil')->row()->nama."</td>
             <td>".round($min['minC1']/$dt2['c1'],6)."</td>
             <td>".round($dt2['c2']/$max['maxC2'],6)."</td>
             <td>".round($dt2['c3']/$max['maxC3'],6)."</td>
             <td>".round($min['minC4']/$dt2['c4'],6)."</td>
```

```
            <td>".round($dt2['c5']/$max['maxC5'],6)."</td>
            <td>".round($dt2['c6']/$max['maxC6'],6)."</td>
      </tr>";
                }
                ?>
```

Penjelasan :

1. Double Array max[ ] nilai maksimum dari semua nilai alternatif pada kriteria ke n.
2. Double Array min[ ] nilai minimum dari semua nilai alternatif pada kriteria ke n.
3. B merupakan variabel untuk baris, k variabel untuk kolom.
4. Mengubah nilai data mentah dan memasukkan kedalam database berdasarkan kondisi nilai.
5. Melakukan perulangan untuk mendapatkan nilai kriteria disetiap alternatif.
6. Kemudian, hasil dari penjumlahan tersebut disimpan pada variabel array max[ ] untuk benefit dan min[ ] untuk cost.
7. Menampilkan nilai normalisasi matriks setiap alternatif.

3.6.5 Mencari nilai preferensi dari seiap kriteria

Berikut merupakan gambaran dari algoritma perhitungan normalisasi

```
public function preferensi()
    {
        $w        =        $this->model->bobot($this->session-
>userdata('id'));
        foreach($w as $x){
            $total   =   $x->b1+$x->b2+$x->b3+$x->b4+$x->b5+$x-
>b6;
            $this->data['b1'] = $x->b1/$total;
            $this->data['b2'] = $x->b2/$total;
            $this->data['b3'] = $x->b3/$total;
            $this->data['b4'] = $x->b4/$total;
            $this->data['b5'] = $x->b5/$total;
```

```
            $this->data['b6'] = $x->b6/$total;
        }

        $max = $this->db->select_max('c2','maxC2') //mesin
            ->select_max('c3','maxC3') //transmisi
            ->select_max('c5','maxC5') //deskripsi
            ->select_max('c6','maxC6') //tahun
            ->join('mobil','mobil.id = data.id_mobil')
            ->limit(50)
            ->where('data.id <=',50)
            ->get('data');

        $min = $this->db->select_min('c1','minC1') //harga
            ->select_min('c4','minC4') //jarak_tempuh
            ->join('mobil','mobil.id = data.id_mobil')
            ->limit(50)
            ->where('data.id <=',50)
            ->get('data');

        $this->data['max'] = $max->row_array();
        $this->data['min'] = $min->row_array();

        $this->data['ranking']          =          $this->db-
>join('mobil','mobil.id = data.id_mobil')
            ->limit(50)
            ->where('data.id <=',50)
            ->get('data');
        }
<?php
                $no = 1;
                foreach ($ranking->result_array() as $dt3) {
                    $info       =       $this->db->where('id',
$dt3['id_mobil'])->get('mobil');
                    $pref  =  round((($min['minC1']/$dt3['c1'])
* $b1) +
                        (($dt3['c2']/$max['maxC2']) * $b2) +
                        (($dt3['c3']/$max['maxC3']) * $b3) +
```

```
                        (($min['minC4']/$dt3['c4']) * $b4) +
                        (($dt3['c5']/$max['maxC5']) * $b5) +
                        (($dt3['c6']/$max['maxC6']) * $b6)
                        , 6);
                    echo " <tr>
                          <td><a
href='".site_url('halaman/details/'.$info->row()->id)."'><img
src='".$info->row()->photo."'                    width='100'
height='110'></a></td>
                          <td>" . $info->row()->nama . "</td>
                          <td>" . $pref . "</td>
                        </tr> ";
                    $no++;
                }
                ?>
```

Penjelasan :

1. perhitungan nilai preferensi data barang, lalu menjumlahkannya untuk mendapatkan nilai preferensi setiap alternatif.
2. kriteria[ ][ ] merupakan nilai kriteria alternatif dari setiap kriteria barang.
3. D[ ][ ] merupakan nilai perhitungan criteria[ ][ ] dikalikan dengan bobot
4. v[ ] merupakan jumlah nilai preferensi setiap alternatif
5. bobot merupakan nilai bobot yang ditentukan oleh pemberi keputusan
6. i merupakan variabel untuk baris, j variabel untuk kolom.
7. Melakukan proses perkalian antara nilai criteria ternormalisasi, dengan nilai bobot yang telah ditentukan pendukung keputusan.
8. Menjumlahkan nilai preferensi setiap alternative disemua kriteria.
9. Mengurutkan data dari terbesar hingga terkecil

3.7 Perancangan Tabel Pengujian Sistem

Pada tabel 3.13 akan diberikan gambaran mengenai rancangan pengujian dari sistem setelah berhasil diimplementasikan. Pengujian yang akan dilakukan dengan menggunakan metode *blackbox testing* dimana setiap fungsi akan diuji,

apakah fungsi tersebut telah sesuai dengan harapan atau belum. Untuk lebih jelasnya berikut rancangan pengujian sistem rekomendasi pembelian mobil.

Tabel 3. 13 Tabel pengujian

Fungsi yang diuji (Penguji/Aktor)	Sekenario Uji	Keterangan
Registrasi (*User*)	Menampilkan eror apabila data yang diisi tidak sesuai dengn data yang diminta	
	Menampilkan pesan eror apabila data yang dimasukkan sudah pernah dugunakan seperti *email* dan *username*.	
	Memunculkan pesan sukses apabila registrasi berhasil dilakukan	
Login (Super Admin, *Admin*,*User*)	Menampilkan pesan *error* apabila *user* memasukkan kombinasi *username* atau *email* dan *password* yang tidak sesuai.	
	Mengatar ke beranda sesuai dengan role *user*.	
Mengunggah kiriman data mobil/ pasang iklan (*User*)	*User* yang dapat memasang iklan adalah user yang sudah mempunyai akun	
	Menampilkan pesan sukses apabila iklan berhasil diunggah	
	Menampilkan pesan eror apabila kolom yang seharusnya diisi belum diisi	
	Mobil akan tampil setelah diunggah	
Mengubah data mobil yang diunggah (*User*, Admin, Super Admin)	Menampilkan data mobil pada form edit data mobil	
	Menampilkan pesan error apabila form tidak diisi atau tidak sesuai	
	Menampilkan pesan sukses ababila data mobil berhasil diubah	
Menghapus iklan mobil atau data mobil (User, Admin, Super Admin)	Data mobil akan hilang setelah mobil tersebut dihapus	
Mengubah status mobil yang sudah terjual (*User*, Admin, Super Admin)	Mobil yang sudah terjual akan berubah statusnya menjadi terjual	
	Akan menampilkan status mobil terjual pada	

Fungsi yang diuji (Penguji/Aktor)	Sekenario Uji	Keterangan
Registrasi (*User*)	Menampilkan eror apabila data yang diisi tidak sesuai dengn data yang diminta	
	Menampilkan pesan eror apabila data yang dimasukkan sudah pernah dugunakan seperti *email* dan *username*.	
	Memunculkan pesan sukses apabila registrasi berhasil dilakukan	
Login (Super Admin, *Admin*,*User*)	Menampilkan pesan *error* apabila *user* memasukkan kombinasi *username* atau *email* dan *password* yang tidak sesuai.	
	Mengatar ke beranda sesuai dengan role *user*.	
	detail mobil sehingga user mengetahui bahwa mobil telah terjual	
Pencarian mobil (*User*)	Akan memunculkan form pencarian abalika menu pencarian mobil ditekan	
	Menampilkan error apabila kolom inputan belum terisi	
	Menampilkan hasil pencarian mobil sesuai dengan katakunci *user*	
Melihat rekomendasi mobil (*User*)	Menampilkan form rekomendai apabila menu rekomendasi mobil ditekan	
	Akan memunculkan error apabila kolom intnputan tidak diisi atau tidak sesuai	
	Akan memunculkan daftar rekomendasi mobil dengan tepat sesuai data yang diinputkan user pada form rekomendasi mobil	
Pengaturan Akun (*User*, Admin, Super Admin)	User dapat mengubah data akun sesai dengan inputan user pada form pengaturan akun	
	Memunculkan error apabila username telah ada sebelumnya	
	Memunculkan error apabila password lama tidak sesuai	

Fungsi yang diuji (Penguji/Aktor)	Sekenario Uji	Keterangan
Registrasi (*User*)	Menampilkan eror apabila data yang diisi tidak sesuai dengn data yang diminta	
	Menampilkan pesan eror apabila data yang dimasukkan sudah pernah dugunakan seperti *email* dan *username*.	
	Memunculkan pesan sukses apabila registrasi berhasil dilakukan	
Login (Super Admin, *Admin*,*User*)	Menampilkan pesan *error* apabila *user* memasukkan kombinasi *username* atau *email* dan *password* yang tidak sesuai.	
	Mengatar ke beranda sesuai dengan role *user*.	
	Memunculkan pesan sukses apabila data berhasil diubah	
Melihat daftar mobil (*User*)	Menampilan daftar mobil yang tersedia atau mobil yang dijual oleh pengguna.	
Menambahkan *wishlist* (*User*)	Akan memasukan data mobil pada halaman wishlist dari *user*	
Menghapus *wishlist* (*User*)	Data mobil akan hilang dari halaman *wihlist user* apabila telah dihapus	

BAB IV
IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bagian ini membahas mengenai hasil gambaran dari rancangan sistem yang telah dibuat. Adapun sistem yang telah dibuat dan diuji setiap fungsi yang ada. Sehingga sistem dinyatakan berhasil dan siap untuk diimplementasikan pada khalayak umum.

4.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap lanjutan dari tahap perancangan sistem, dimana semua perancangan yang terlah dibuat baik berupa perancangan diagram sistem maupun perancangan *user interface* akan dituangkan dalam bentuk kode program. Proses ini biasa disebut sebagai prsoses *koding*. Proses koding itu sendiri memakan waktu yang cukup lama karena *programmer* harus membuat semua fungsi dari aplikasi sesuai dengan perancangan yang diharapkann sehingga aplikasi yang dibuat memenuhi kriteria aplikasi yang baik. Karena aplikasi yang baik adalah aplikasi yang dibuat atau diimplementasikan sesuai dengan perancangannya. Apalah guna perancangan sustem apabila saat diimplementasikan tidak sesui degan perancangan sistem yang telah dibuat.

4.1.1 Implementasi Codeigniter

Pada bab sebelumnya telah disampaikan bahwa sistem akan dibangun dengan menggunakan *framework* Laravel. Berikut ini spesifikasi dan konfigurasi dalam pembangunan sistem.

1. Versi Codeigniter

Framework Codeigniter yang digunakan dalam pembangunan sistem ini adalah Codeigniter versi 3.1

2. *Package*

a. Image Resize

Package ini digunakan untuk mengurangi ukuran gambar atau dapat disebut *package* yang digunakan untuk mengkompres dan mengubah ukuran gambar.

b. Image Manipulation

Package ini digunakan untuk melakukan manipulasi gambar serta pemrosesan data gambar pada sistem.

c. HTML Scapper

Package ini digunakan untuk membantu dalam melaksanakan proses pengambilan atau *scarpping* data dari rajamobil.com.

d. JQuery Datatables

Package ini digunakan untuk menampilkan data dalam bentuk tabel serta dengan fungsi yang lebih lengkap seperti *sorting*, *pagination* dan kolom pencarian.

3. *Model*

Model meupakan *class* yang terdisri dari beberapa fungsi yang digunakan untuk berinteraksi dengan database. Untuk lebih jelasnya pada tabel 3.1 berikut akan dijelaskan model apa saja yang digunakan.

Tabel 4. 1 Tabel data model

Nama Model	Fungsi
Model	Model utama dan didalamnya terdapat fungsi yang berkaitan dengan algoritma dari sistem
Mmobil	Manajemen data pada tabel mobil yang ada pada *database*
Musers	Manajemen data pada tabel users yang ada pada *database*
Mwishlist	Manajemen data pada tabel wishlist yang ada pada *database*
Mmerek	Manajemen data pada tabel merek yang ada pada *database*

4. *Controller*

Dalam konsep MVC (*Model View Controller*), *controller* memiliki peran sebagai penghubung antara *model* yang berkaitan langsung dengan *database* dengan *view* yaitu halaman tampilan antarmuka untuk pengguna.

Tabel 4. 2 Tabel controller

Nama *Controller*	Fungsi
My_Controller	Digunakan sebagai controller utama atau super class controller, yang selalu digunakan pada semua controller lain

Public_Controller	Digunakan untuk semua aktifitas yang tidak memerlukan autentikasi user.
Halaman	Untuk menjalankan semua fungsi yang bertugas menampilkan halaman utama
Auth	Mengakomodir fungsi – fungsi yang berkaitan dengan autentikasi user pada sistem.
Metode	Mengakomodir fungsi – fungsi yang berkaitan dengan pengolahan data dengan menggunakan metode SAW
Wishlist	Mengakomodir fungsi – fungsi yang berkaitan dengan data wishlist
Welcome	Berkaitan dengan tampilan utama sistem
Admin_Controller	Supe class dari controller admin yang diakses semua controller yang berkaitan dengan admin
Halaman	Berkaitan dengan semua fungsi yang mengatur tampilan halaman admin
Merek	Berkaitan dengan CRUD dari data merek mobil
Mobil	Berkaitan dengan CRUD dari data mobil
Tips	Berkaitan dengan fungsi yang menampilkan tips
Users	Berkaitan dengan semua fungsi yang mengelola data user

5. Routing

Pada *framework* Codeigniter *routing* adalah proses dimana suatu item dapat sampai ke tujuan dari satu lokasi ke lokasi lain.

```
$route['default_controller'] = 'halaman';
$route['login'] = 'auth/login';
$route['signup'] = 'auth/signup';
$route['logout'] = 'auth/logout';
$route['posting'] = 'halaman/posting';
$route['merek/(:any)'] = 'halaman/kategori/$1';
$route['mobil/(:any)'] = 'halaman/details/$1';
$route['pencarian'] = 'halaman/pencarian';
```

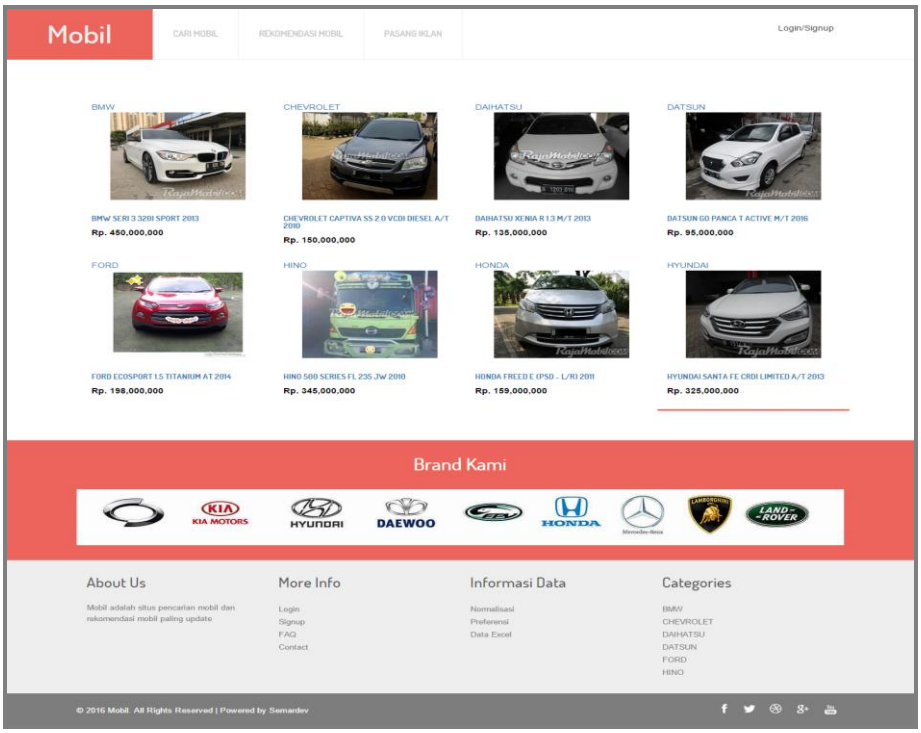
```
$route['rekomendasi'] = 'halaman/rekomendasi';
$route['account/dashboard'] = 'admin/halaman/index';
$route['account/mobilku'] = 'admin/halaman/mobilku';
$route['account/wishlist'] = 'admin/halaman/wishlist';
$route['account/setting'] = 'admin/halaman/account_setting';
$route['system/setting'] = 'admin/halaman/konfigurasi';
$route['system/weight'] = 'admin/halaman/bobot';
$route['faq'] = 'halaman/faq';
$route['404_override'] = '';
$route['translate_uri_dashes'] = TRUE;
```

4.1.2 Implemtasi Antarmuka

Berikut ini adalah beberapa tampilan hasil implementasi antarmuka dari sistem yang ditampilkan melalui *browser*:

1. Halaman beranda utama

   Gambar 4.1 berikut merupakan tampilan halaman beranda utama saat pertama kali sistem dibuka dan halaman tersebut dapat diakses oleh semua pengguna sistem rekomendasi pembelian mobil.

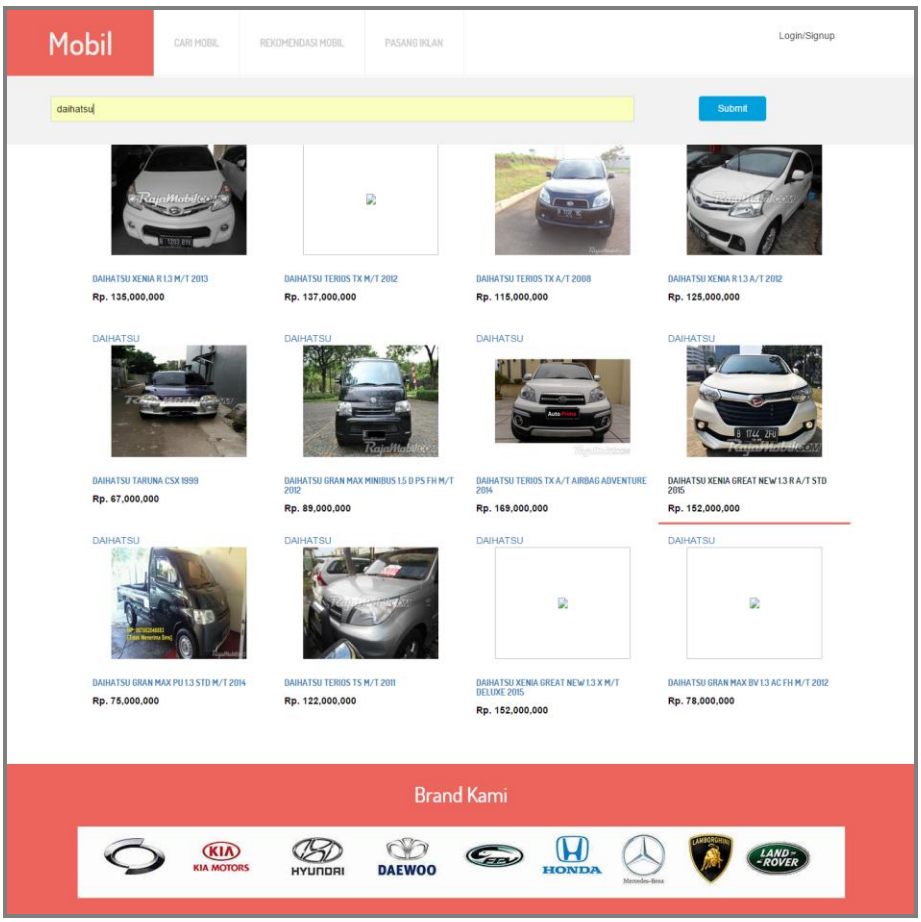


Gambar 4. 1 Hamaman beranda

Pada halaman beranda di atas terdapat beberapa menu yaitu cari mobil, rekomendasi mobil, pasang iklan dan *login* dan *signup*. Selain itu terdapat banyak data iklan dari mobil yang ditampilkan sehingga penggua dapat melihat mobil mana yang akan dibeli.

2. Halaman cari mobil

Gambar 4.2 merupakan tampilan halaman cari mobil pengguna yang ada pada sistem informasi rekomendasi pembelian mobil.

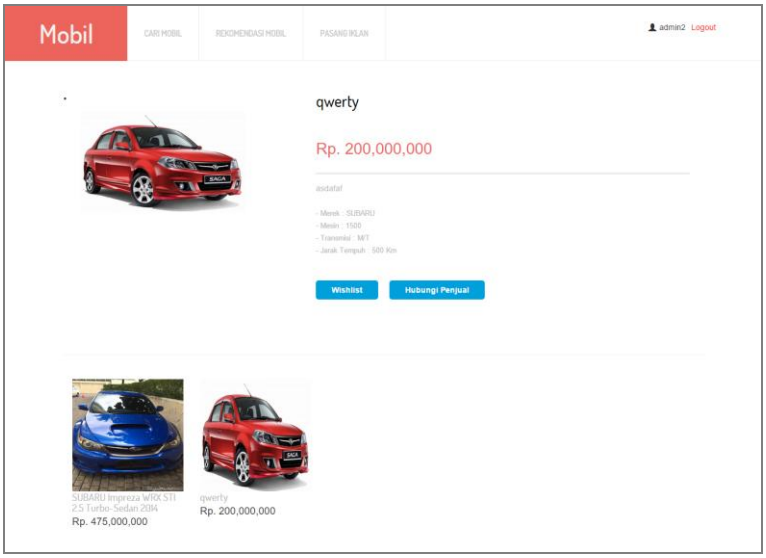


Gambar 4. 2 Halaman cari mobil

Untuk mengakses halaman di atas terlebih dahulu user harus memilih menu cari mobil, kemudian akan muncul form inputan untuk memasukan kata kunci yang diinginkan lalu menekan tombol submit untuk memuai pencarian. Setelah beberapa saat akan munvul hasil pencarian sesuai dengan kata kunci yang ditulis oleh pengguna.

3. Detail Mobil

Gambar 4.3 merupakan tampilan halaman detail mobil pengguna yang ada pada sistem informasi rekomendasi pembelian mobil.

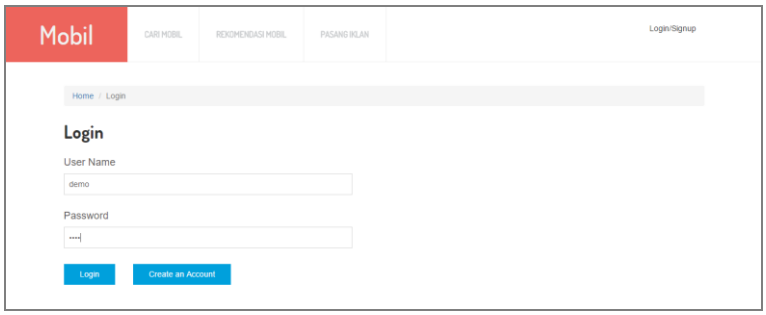


Gambar 4. 3 Halaman Detail Mobil

Gambar di atas menampilkan detail dari mobil. Untuk menampilkan halaman detail mobil pengguna terlebih dahulu harus melihat daftar mobil pada beranda atau tempat lain, lalu menekan nama atau foto dari mobil untuk menuju halaman detail dari mobil. Pada halaman ini menggambarkan spesifikasi dari mobil yang dijual, dan terdapat menu untuk menghubungi penjual dan memasukan mobil sebagai mobil favorit.

4. Halaman Login

Gambar 4.4 merupakan tampilan halaman login pengguna yang ada pada sistem informasi rekomendasi pembelian mobil.

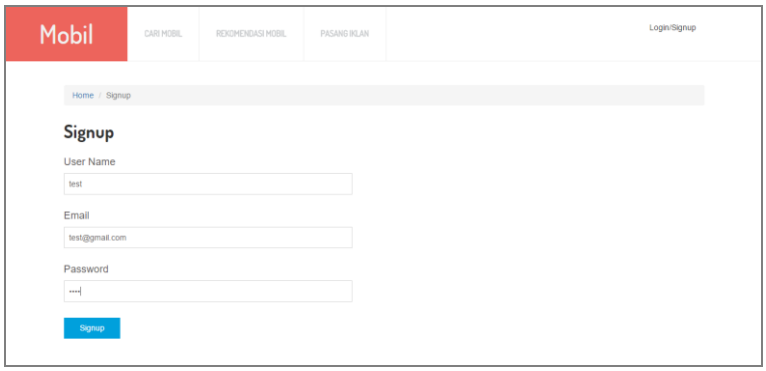


Gambar 4. 4 Halaman Login

Pada halaman *login* di atas dibuat untuk aktifitas autentikasi user yang ingin menggunakan sistem. Pada halaman tersebut terdapat form login dimana dalam form tersebut terdapat kolom inputan username dan password. Selain itu terdapat tombil login dan tombol lain yang berfungsi untuk mengarahkan pengguna ke halaman registrasi.

5. Halaman Registrasi

Gambar 4.5 merupakan tampilan halaman registrasi pengguna yang ada pada sistem informasi rekomendasi pembelian mobil.

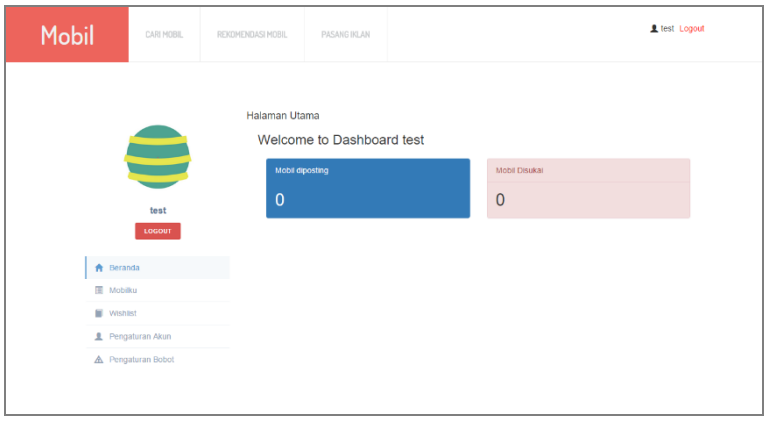


Gambar 4. 5 Halaman *Registrasi*

Pada tampilan registrasi di atas terdapat *form* yang dapat digunakan oleh user untuk menuliskan data diri *user*, dimana user harus melengkapi *form* inputan kemudian menekan tombol signup untuk melakukan registrasi atau pendaftaran.

6. Halaman utama user

Gambar 4.6 merupakan tampilan halaman utama user yang ada pada sistem informasi rekomendasi pembelian mobil.

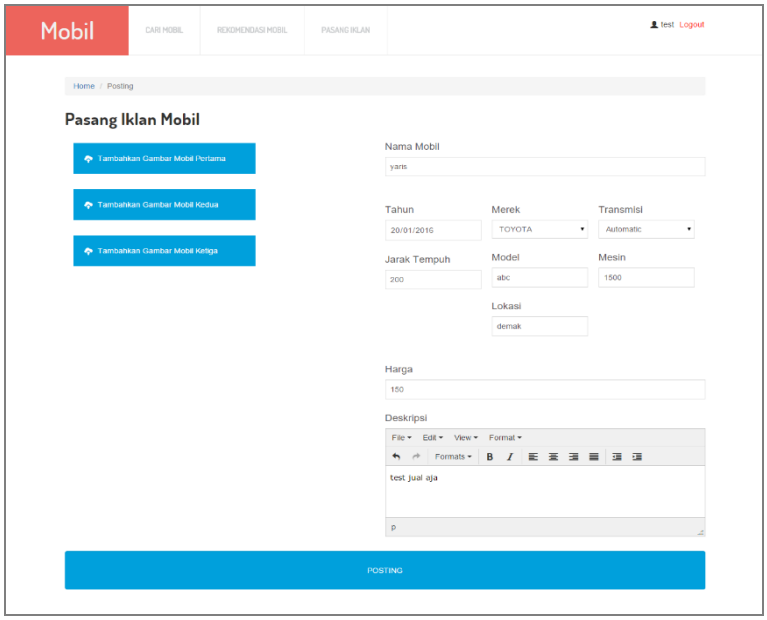


Gambar 4. 6 Halaman Utama *user*

Halaman tersebut memiliki beberapa fungsi untuk melihat jumlah daftar mobil yang telah diposting dan disukai oleh user lain yang melihatnya pada halaman utama sistem pada bagian kiri terdapat slide bar yang berisi menu – menu seperti beranda, mobilku, whistlst, pengaturan akun dan pengaturan bobot. Pada bagian kanan atas terdapat tombol logout untuk keluar dari sistem.

7. Pasang iklan

Gambar 4.7 merupakan tampilan halaman pasang ikalan yang ada pada sistem informasi rekomendasi pembelian mobil.

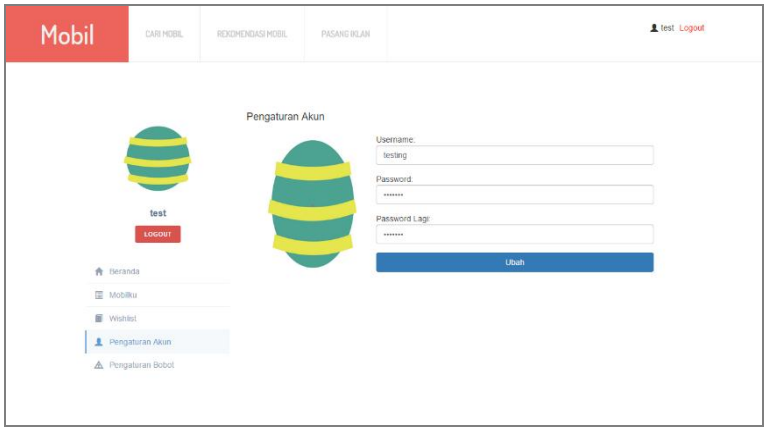


Gambar 4. 7 Halaman pasang iklan

Halaman ini halaman yang diguanakn untuk memasang iklan, pada bagian kanan terdapat form yang harus diisi ketika *user* akan memposting iklan dan user dapat menginputkan deskripsi dengan mengisi form deskripsi. Pada bagian kiri terapat form untuk menginputkan foto dari mobil yang akan dijual, foto yang diinputkan sebanyak 3 foto.

8. Pengaturan akun

Gamabar 4.8 merupakan tampilan halaman pengaturan akun yang ada pada sistem informasi rekomendasi pembelian mobil.

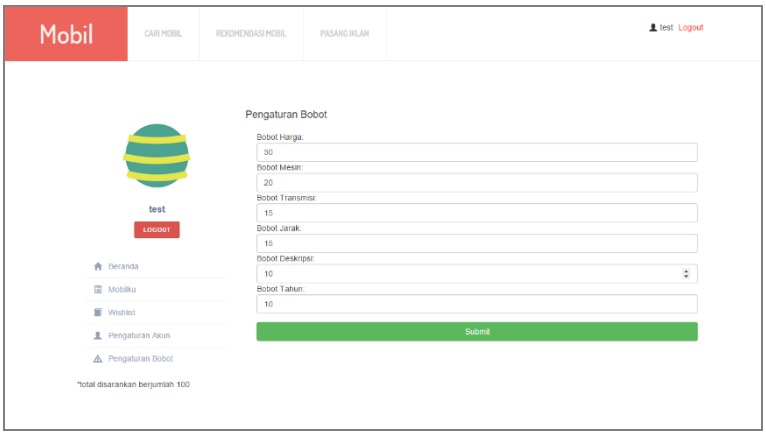


Gambar 4. 8 Pengaturan akun

Halaman tersbut digunakan untuk mengatur akun user seperti merubah foto profil, username dan password. Pada saat penggatian password dapat langsung mengganti password yang baru dan setelah diganti dapat menekan tombol ubah.

9. Pengaturan Bobot

Gamabar 4.9 merupakan tampilan halaman pengaturan bobot yang ada pada sistem informasi rekomendasi pembelian mobil.

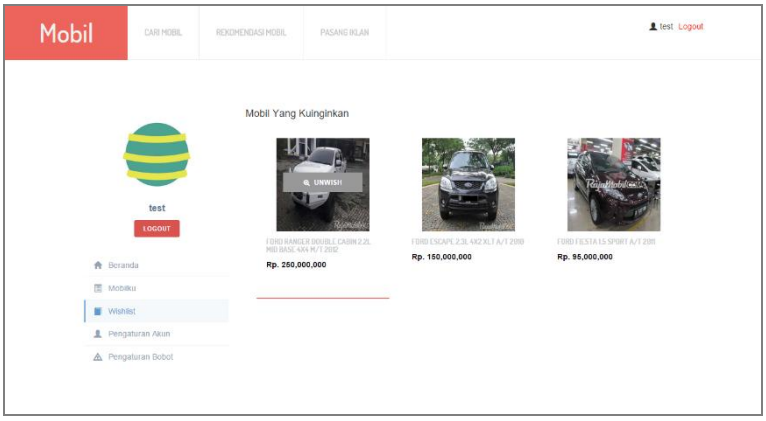


Gambar 4. 9 Pengaturan bobot

Halaman tersebut merupakan halaman yang digunakan untuk pengaturan bobot dari mobil, pengatuaran tersebut memiliki jumlh total nilai adalah 100 sehingga user dapat menentukan bagian mana yang memilii nilai lebih dan bagian mana yang nilainya kurang.

10. *Wishlistt*

Gamabar 4.10 merupakan tampilan halaman *wistlist* yang ada pada sistem informasi rekomendasi pembelian mobil.

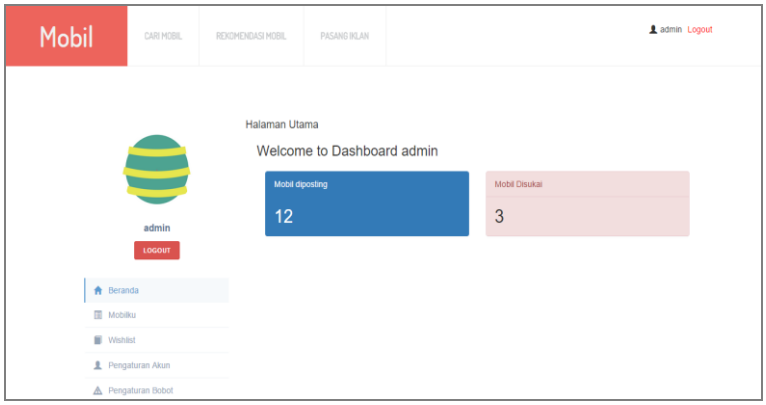


Gambar 4. 10 Tampilan *wishlist*

Halaman tersebut merupakan halaman *wishlist*, iklan yang muncul dalam whistlist adalah iklan yang telah dimasukkan ke daftar whistlist pada bagian gambar tersbut terdapat fungi untuk *unwish* .

11. Halaman Utama Admin

Gambar 4.11 berikut merupakan halaman beranda dari admin pada sistem informasi rekomendasi pembelian mobil.



Gambar 4. 11 Halaman utama admin

Pada gambar diatas menunjukan beranda dari admin. Tidak banyak yang ditampilkan pada halaman ini, dimana hanya terdapat ucapan selamat datang dan dua tampilan yang menunjukan jumlah total dari data yang terdapat pada database sistem yaitu jumlah mobil yang diposting dan jumlah mobil yang disukai oleh pengguna.

12. Halaman manajemen pengguna

Gambar 4.12 berikut merupakan halaman manajemen pengguna pada sistem informasi rekomendasi pembelian mobil.

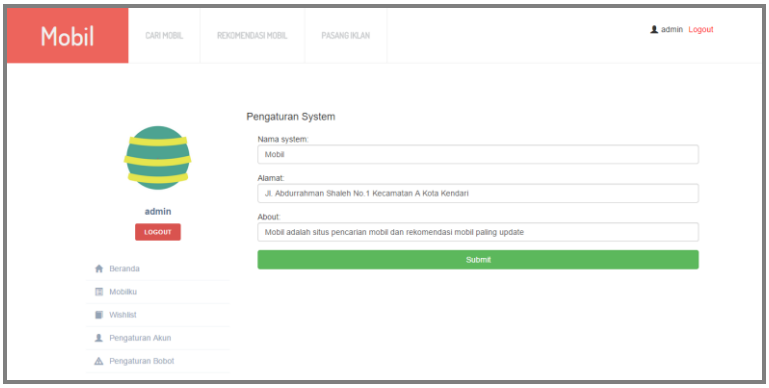
Gambar 4. 12 Halaman manajemen pengguna

Pada halaman manajemen pengguna di atas menampilkan sebuah tabel yang didalamnya terdapat data pengguna yang telah mendaftar pada sistem. Admin

dapat menggunakan halaman ini untuk melakukan manajemen pengguna seperti melakukan pengubahan data pengguna dan lainnya.

13. Pengaturan Sistem

Gambar 4.13 berikut merupakan halaman pengaturan sistem pada sistem informasi rekomendasi pembelian mobil.

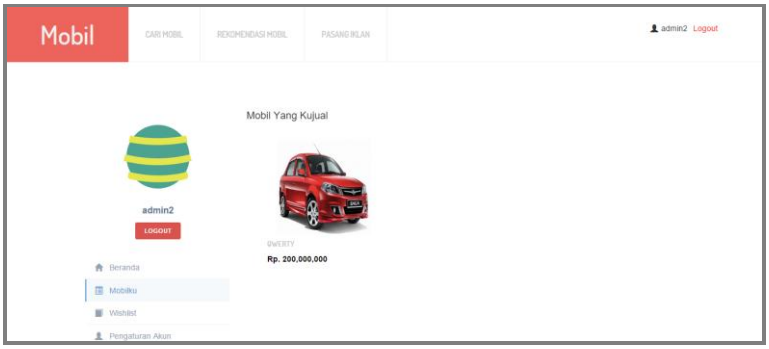


Gambar 4. 13 Pengaturan akun

Pada gambar di atas menunjukan sebuah halaman yang digunakan oleh admin untuk melakukan pengatuan sistem. Dimana ini mempermudah admin mengubah nama dari aplikasi ini, alamat dari perusaan dan profil perusahaan tanpa harus melakukan pemrograman.

14. Daftar mobil yang kujual.

Gambar 4.14 berikut merupakan tampilan dari daftar mobil yang dijual oleh user pada sistem informasi rekomendasi pembelian mobil.



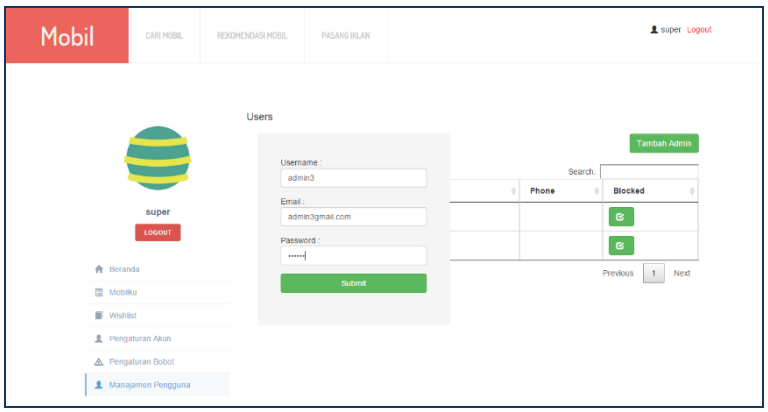
Gambar 4. 14 Daftar mobil *user*

Pada gambar diatas menunjukan tampilan dari daftar mobil yang dijual oleh user. Pada halaman tersebut hanya menampilkan daftar mobil yang dijual

oleh user yang melakukan login saat ini dimana tidak terdapat data dari mobil yang dijual oleh user lain.

15. Beranda Super Admin

Gambar 4.15 berikut merupakan halaman tampilan beranda super admin pada sistem informasi rekomendasi pembelian mobil.



Gambar 4. 15 Beranda super admin

Pada gambar di atas menampilkan beranda dari super damin dimana pada beranda tersebut terdapat sebuah tabel yang berisikan daftar *user* yang ada pada sistem. Pada halaman ini berfungsi sebagai halaman dimana super admin bisa mengangakan atau mengganti level *user* menjadi admin.

4.2 Pengujian Sistem

Berikut ini merupakan hasil pengujian sistem yang telah dilakukan dengan menggunakan metode pengujian *blackbox testing* dimana pengujian dilakukan saat sistem telah selesai dibuat. Pengujian ini dilakukan terhadap fungsi – fungsi yang dapat dilakukan sistem telah sesuai atau belum dengan yang diharapkan. Misalnya apabila inputan yang diinginkan apakah output yang diharapkan telah sesuai dengan inputan.

Tabel 4. 3 Tabel Pengujian Sistem

Fungsi yang diuji (Penguji/Aktor)	Sekenario Uji	Keterangan
Registrasi (*User*)	Menampilkan eror apabila data yang diisi tidak sesuai dengn data yang diminta	Sesuai

	Menampilkan pesan eror apabila data yang dimasukkan sudah pernah dugunakan seperti *email* dan *username*.	Sesuai
	Memunculkan pesan sukses apabila registrasi berhasil dilakukan	Sesuai
Login (Super Admin, *Admin*,*User*)	Menampilkan pesan *error* apabila *user* memasukkan kombinasi *username* atau *email* dan *password* yang tidak sesuai.	Sesuai
	Mengatar ke beranda sesuai dengan role *user*.	Sesuai
Mengunggah kiriman data mobil/ pasang iklan (*User*)	*User* yang dapat memasang iklan adalah user yang sudah mempunyai akun	Sesuai
	Menampilkan pesan sukses apabila iklan berhasil diunggah	Sesuai
	Menampilkan pesan eror apabila kolom yang seharusnya diisi belum diisi	Sesuai
	Mobil akan tampil setelah diunggah	Sesuai
Mengubah data mobil yang diunggah (*User*, Admin, Super Admin)	Menampilkan data mobil pada form edit data mobil	Sesuai
	Menampilkan pesan error apabila form tidak diisi atau tidak sesuai	Sesuai
	Menampilkan pesan sukses ababila data mobil berhasil diubah	Sesuai
Menghapus iklan mobil atau data mobil (User, Admin, Super Admin)	Data mobil akan hilang setelah mobil tersebut dihapus	Sesuai
Mengubah status mobil yang sudah terjual (*User*, Admin, Super Admin)	Mobil yang sudah terjual akan berubah statusnya menjadi terjual	Sesuai
	Akan menampilkan status mobil terjual pada detail mobil sehingga user mengetahui bahwa mobil telah terjual	Sesuai
Pencarian mobil (*User*)	Akan memunculkan form pencarian abalika menu pencarian mobil ditekan	Sesuai
	Menampilkan error apabila kolom inputan belum terisi	Sesuai
	Menampilkan hasil pencarian mobil sesuai	Sesuai

Fungsi yang diuji (Penguji/Aktor)	Sekenario Uji	Keterangan
Registrasi (*User*)	Menampilkan eror apabila data yang diisi tidak sesuai dengn data yang diminta	Sesuai
	Menampilkan pesan eror apabila data yang dimasukkan sudah pernah dugunakan seperti *email* dan *username*.	Sesuai
	Memunculkan pesan sukses apabila registrasi berhasil dilakukan	Sesuai
Login (Super Admin, *Admin*,*User*)	Menampilkan pesan *error* apabila *user* memasukkan kombinasi *username* atau *email* dan *password* yang tidak sesuai.	Sesuai
	Mengatar ke beranda sesuai dengan role *user*.	Sesuai
	dengan katakunci *user*	
Melihat rekomendasi mobil (*User*)	Menampilkan form rekomendai apabila menu rekomendasi mobil ditekan	Sesuai
	Akan memunculkan error apabila kolom intnputan tidak diisi atau tidak sesuai	Sesuai
	Akan memunculkan daftar rekomendasi mobil dengan tepat sesuai data yang diinputkan user pada form rekomendasi mobil	Sesuai
Pengaturan Akun (*User*, Admin, Super Admin)	User dapat mengubah data akun sesai dengan inputan user pada form pengaturan akun	Sesuai
	Memunculkan error apabila username telah ada sebelumnya	Sesuai
	Memunculkan error apabila password lama tidak sesuai	Sesuai
	Memunculkan pesan sukses apabila data berhasil diubah	Sesuai
Melihat daftar mobil (*User*)	Menampilan daftar mobil yang tersedia atau mobil yang dijual oleh pengguna.	Sesuai
Menambahkan *wishlist* (*User*)	Akan memasukan data mobil pada halaman wishlist dari *user*	Sesuai
Menghapus *wishlist*	Data mobil akan hilang dari halaman *wihlist*	Sesuai

Fungsi yang diuji (Penguji/Aktor)	Sekenario Uji	Keterangan
Registrasi (*User*)	Menampilkan eror apabila data yang diisi tidak sesuai dengn data yang diminta	Sesuai
	Menampilkan pesan eror apabila data yang dimasukkan sudah pernah dugunakan seperti *email* dan *username*.	Sesuai
	Memunculkan pesan sukses apabila registrasi berhasil dilakukan	Sesuai
Login (Super Admin, *Admin*,*User*)	Menampilkan pesan *error* apabila *user* memasukkan kombinasi *username* atau *email* dan *password* yang tidak sesuai.	Sesuai
	Mengatar ke beranda sesuai dengan role *user*.	Sesuai
(*User*)	*user* apabila telah dihapus	

Setelah melakukan pengujian dan setelah menuliskan hasil pengujian pada tabel hasil pengujian di atas dapat ditarik kesimpulan bahwa sistem yang dibuat telah sesuai denganharapan atau telah sesuai dengan keinginan. Dimana semua fungsi telah berjalan dengan baik dan sesuai dengan tujuan dari setiap fungsi tersebut dibuat. Jika terjadi masalah atau ketidak sesuaian sistem itu tidak terlepas dari masih adanya kekurangan dari sistem yang akan terus diperbaiki dan diuji kembali.

4.3 Pengujian Algoritma

1. Menentukan merek dan maksimal uang

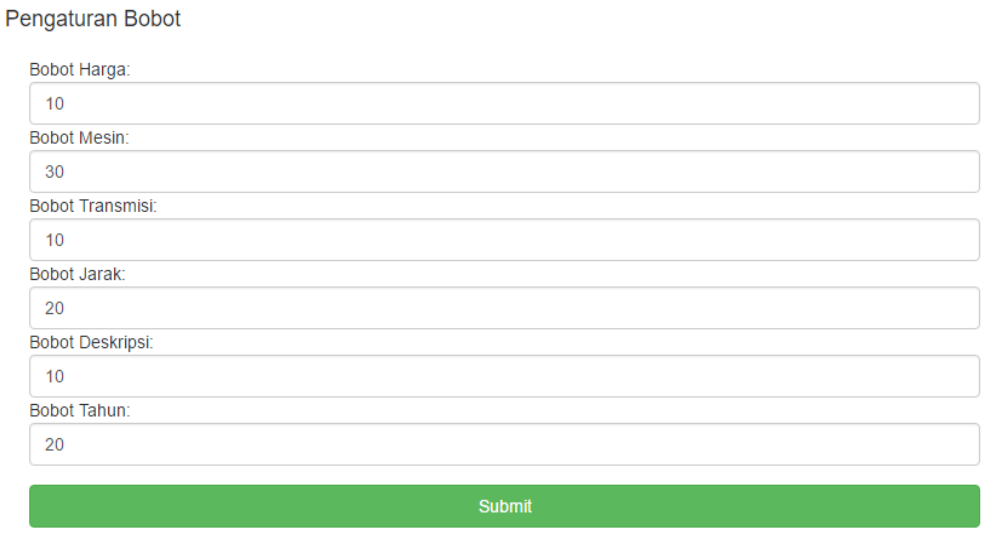
   Pada pengujian sistem yang dilakukan dengan studi kasus lima mobil dengan merek honda dan uang maksimum Rp. 900.000.000,-. Untuk leebih detailnya akan dijelaskan pada tabel 4.4

Tabel 4. 4 Mobil Honda

id	id_users	nama	merek	model	tahun	mesin	transmisi	jarak_tempuh	harga	lokasi	deskripsi
1	2	HONDA Jazz RS A/T 2013	HONDA	Jazz RS A/T 2013	2013	1500	M/T	51182	178000000	Jakarta Timur, Jakarta	Speksifikasi : DIJUAL buat sesama pemakai !!!! TI...
11	2	HONDA Accord Maestro 1992	HONDA	Accord Maestro 1992	1992	2000	M/T	150000	42500000	Depok, Jawa Barat	body masih kaleng, cat mulus, list2 lengkap, ac d...
18	15	HONDA Cr-v 2.0 i-VTEC M/T 2008	HONDA	Cr-v 2.0 i-VTEC M/T 2008	2008	1997	M/T	100000	178000000	Bandung, Jawa Barat	Bismillah.. Asallamu'alaikum...punten bapak ibu /...
20	15	HONDA Jazz RS A/T 2009	HONDA	Jazz RS A/T 2009	2009	1500	A/T	56000	141000000	Depok, Jawa Barat	Jazz RS 1.5 Hatchback Silver 2009: - Kondisi sang...
21	15	HONDA Accord Maestro	HONDA	Accord Maestro 1991	1991	2000	M/T	200000	35000000	Tangerang, Banten	Honda Accord Maestro 2.0 th 1991 Silver Untuk yan...

2. Menentukan Bobot Setiap Kriteria

Selanjutnya yaitu menentukan bobot yang di *setting* pada menu pengaturan bobot oleh member. Pada gambar 1.1 member mengatur nilai bobot dari kriterianya.



Gambar 4. 16 Pengaturan Bobot

3. Hasil Datafikasi

Tabel 3.2 merupakan tabel kecocokan yang menjelaskan dari masingmasing kriteria tersebut akan ditentukan sub-sub kreiterianya. Pada sub kriteria terbagi menjadi beberapa bilangan-bilangan fuzzy. Dari masing-masing bobot tersebut,

dibuat suatu variabel yang akan dirubah ke dalam bilangan fuzzy dengan rumus yaitu variabel ken/banyaknya sub kriteria-1.

Contohnya : terdapat sub kriteria 5, maka dicari bobot sub kriterianya.

0/(5-1)=0/4=0 → dikenfersikan menjadi 0,01 agar nilai yang keluar tidak menghasilkan *error.*

1/(5-1)=1/4=0,25

2/(5-1)=2/4=0,5

3(5-1)=3/4=0,75

4/(5-1)=4/4=1

Adapun Penentuan nilai bobot ditentukan pengamatan terhadap situs *E-commerce,* dan wawancara terhadap beberapa pengguna *E-commerce,* atau ditentukan oleh pengguna. Adapun nilai bobot preferensi yang diterapkan secara *default* pada sistem pendukung keputusan ini yaitu, harga 30%, mesin 20%, transmisi 15%, jarak 15%, deskripsi 10%,dan tahun 10%. Pada sistem nilai bobot preferensi sepenuhnya dapat diatur oleh pengguna yang telah menjadi member.

tabel 4. 5 Sub-Kriteria

Kode	**Sub-Kriteria**	**Range**	**Bobot**
C1	Sangat murah	0 dan <=50.000.000	0,01
	Murah	>50.000.000 dan <=100.000.000	025
	Sedang	>100.000.000 dan <=200.000.000	0,5
	Mahal	>200.000.000 dan <=300.000.000	0,75
	Sangat mahal	<=500.000.000	1
C2	Kecil	0-1000	0,01
	Sedang	1000-2000	0,5
	Besar	>=2000	1
C3	Manual	M/T	0,01

	Automatic	A/T	1
C4	Sangat dekat	0-2000	0,01
	Dekat	2000-10.000	0,25
	Terjangkau	10.000-50.000	0,5
	Jauh	50.000-100.000	0,75
	Sangat jauh	>100.000	1
C5	Tidak jelas	0-50	0,01
	Kurang Jelas	50-200	0,3
	Jelas	200-1000	0,6
	Sangat jelas	>=2000	1
C6	Lama	0-20	0,01
	Dekade	20-40	0,3
	5 tahun terakhir	40-60	0,6
	Tahun ini	60-80	1

Berikut merupakan hasil dari datafikasi berdasarkan dari lima mobil yang menjadi acuan.

tabel 4. 6 Hasil Deskripsi

Mobil	Harga	Mesin	Transmisi	Jarak	Deskripsi	Tahun
Mobil 1	0.5	0.5	0.01	0.75	0.3	0.6
Mobil 2	0.01	1	0,01	1	0.3	0.3
Mobil 3	0.5	0.5	0.01	0.75	0.3	0.6
Mobil 4	0.5	0.5	1	0.75	0.3	0.6
Mobil 5	0.01	1	0.01	1	0.3	0.3

4. Normalisasi

Setelah mendapatkan hasil datafikasi dari lima mobil honda, maka mencari nilai normalisasi. Adapun nilai normalisasi perhitungannya terbagi menjadi du acara berdasrkab jenis dari keriteria *cost* ataupun *benefit. Cost* merupakan kriteria yang apabila nilai rendah makan menjadi prioritas, begitupun sebaliknya dengan *benefit.*

$$r11\frac{\min(0.5,0.01,0.5,0.5,0.01)}{0.5} = 0.02$$

$$r2020\frac{0.5}{maks(0.5,1,0.5,0.5,1)} = 0.1$$

Maka akan diperoleh hasil normalisasi berdasrkan jenis dari *cost* ataupun *benefit*.

Mobil	Harga	Mesin	Transmisi	Jarak	Deskripsi	Tahun
Mobil 1	0.02	0.5	1	0.75	1	0.5
Mobil 2	1	1	1	1	1	1
Mobil 3	0.02	0.5	1	0.75	1	0.5
Mobil 4	0.02	0.5	0.01	0.75	1	0.5
Mobil 5	1	1	1	1	1	1

5. Preferensi

Yang terakhir yaitu mengalikan antara bobot yang sudah di atur oleh member dengan hasil dari normalisasi. Adapun perhitungannnya adalah sebagai berikut:

r11 x bobot

Mobil	Harga	Mesin	Transmisi	Jarak	Deskripsi	Tahun	Total
Mobil 1	0.002	0.05	0.15	0.1125	0.1	0.2	0.6445
Mobil 2	0.1	0.10	0.15	0.15	0.1	0.40	1
Mobil 3	0.002	0.05	0.15	0.1125	0.1	0.2	0.6145
Mobil 4	0.002	0.05	0.001	0.1125	0.1	0.2	0.4745
Mobil 5	0.1	0.10	0.15	0.15	0.1	0.40	1

Dari hasil akhir maka mobil yang direkomendasikan yaitu :

a. Mobil 2 (Honda Accord maestro 1992)
b. Mobil 5 (Honda Accord maestro 1991)
c. Mobil 1 (Honda Jazz RS M/T 2013)
d. Mobil 3(Honda Cr-v 2.0 i-VTEC M/T 2008)
e. Mobil 4 (Honda Jazz RS A/T 2009)

BAB V
PENUTUP

5.1 Kesimpulan

Semakin banyaknya masyarakat yang menggunakan E-commerce sebagai media bisnis, maka diperlukanya suatu wadah yang mampu membuat pengunjung merasa *user friendly* dan mampu menjawab permintaan pengguna..

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis didapatkan kesimpulan sebagai berikut:

1. Pengguna dapat mengatur bobot dengan metode Simple Additive Weighting yang berhasil diterapkan untuk merekomendasikan mobil sesuai kriteria.
2. Dari hasil uji semakin sedikit jumlah barang yang akan direkomendasikan maka kesesuaian ranking semakin kecil. Sebaliknya, semakin banyak jumlah barang, maka kesesuaian semakin akurat.
3. Kata kunci mempengaruhi nilai presentase kesesuaian, dikarenakan jumlah data uji keseluruhan mempengaruhi nilai preferensi pada perankingan.
4. Untuk memperbarui data dilakukan dengan metode HTML DOM berdasarkan halaman web pada rajamobil.com

5.2 Saran

Saran yang didapatkan berdasarkan aplikasi sistem pendukung keputusan rekomendasi pembelian mobil menggunakan metode simple additive weighting adalah sebagai berikut:

1. Studi kasus yang diambil datanya yaitu dari rajamobil.com, alangkah baiknya apabila menambah E-commerce yang lain. Sehinngga data yang ditampilkan dalam rekomendasi lebih banyak.
2. Data yang ditampilkan bisa saja berbeda dengan data aslinya, sehingga disarankan lebih baik menggunakan data API untuk mendapatkan data secara real time.

DAFTAR PUSTAKA

Alfarabi Hadi, D. (2016). Codeigniter Part 1 : Pengertian dan Cara Menggunakan CodeIgniter. Retrieved February 9, 2016, from http://www.malasngoding.com/pengertian-dan-cara-menggunakan-codeigniter/

Bernadi, J. (2013). Aplikasi Sistem Informasi Penjualan Berbasis Web pada Toko Velg YQ. *ComTech*, *4*, 731–741.

Cordiaz, M., & Prahasto, T. (2013). Analisis Faktor Kredibilitas Website E-Commerce Indonesia Studi pada Online Purchasing, *1*, 41–47.

Downing, D., & Covington, M. (1992). *Kamus Istilah Komputer*. jakarta: erlangga.

Eniyati, S. (2011). Perancangan Sistem Pendukung Pengambilan Keputusan untuk Penerimaan Beasiswa dengan Metode SAW (Simple Additive Weighting). *Dinamik-Jurnal Teknologi Informasi*, *16*(Sri Eniyati), 171–177. Retrieved from http://www.unisbank.ac.id/ojs/index.php/fti1/article/view/364

Fingar, P., Kumar, H., & Sharma, T. (2000). *Enterprise E-Commerce* (1st ed.). Meghan-Kiffer Press.

Kamaludin, A. (2012). Sistem Pendukung keputusan dalam Pemilihan Alternatif Alat Kontrasepsi Menggunakan Simple Additive Weighting. *Jurusan Teknik Informatika Uin Sgd Bandung*, (april).

Kasman, A. D. (2015). *Framework Laravel 5 Panduan Praktis dan Trik Jitu*. (A. Saputra, Ed.) (Cetakan Pe). CIrebon: CV. ASFA Solution.

Kusrini. (2007). *Konsep dan Aplikasi Sistem Pendukung Keputusan*. Yogyakarta: Andi Offset.

Laudon, K. C., & Traver, C. G. (2013). *E-commerce 2013 : Bussuness, Technology, Society*. (tenth). Prentice-Hall,Inc.

Lee, I. (2014). *Trends in E-Business, E-Services, and E-Commerce : Impact of Technology on Goods, Service, and BussinesTransactions*. IGI Global.

Li, H., & Hong, J. (2013). Factors Influencing Consumers ' Online Repurchasing Behavior : A Review and Research Agenda, *2013*(December), 161–166.

Li, L., Chai, Y., & Liu, Y. (2011). Inter-Group and Intra-Group Externalities of Two-Sided Markets in Electronic Commerce, *2011*(March), 52–58. http://doi.org/10.4236/jssm.2011.41008

Nugroho, A. (2005). *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek* (Revisi). Bandung: Informatika.

Setiaji, P. (2013). Sistem Pendukung Keputusan Dengan Metode Simple Additive Weighting. *Universitas Muria Kudus*. Retrieved from http://jurnal.umk.ac.id/index.php/simet/article/download/117/113

Sprague, R. H. J., & Hugh J. Watson. (1993). Historical Overview of Decision Support Systems (DSS), 1753–1755.

Turban, E., & Aronson, J. E. (2001). *Decision Support Systems and Intelligent Systems* (6th ed.). new jersey: Prentice Hall.