

**SKRIPSI**

**RANCANG BANGUN SISTEM INFORMASI MONTIR  
BERBASIS *WEBSITE* DI BANGKINANG KOTA  
( *ANALYSYS SYSTEM* )**



**NAMA : NURFAJRI NALDO**

**NIM : 1755201041**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI  
RIAU  
2021**

**SKRIPSI**

**RANCANG BANGUN SISTEM INFORMASI MONTIR  
BERBASIS *WEBSITE* DI BANGKINANG KOTA  
( *ANALYSIS SYSTEM* )**



**NAMA : NURFAJRI NALDO**

**NIM : 1755201041**

Diajukan Sebagai Persyaratan Untuk Mendapatkan  
Gelar Sarjana S1 Teknik Informatika

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI  
RIAU  
2021**

# **BAB I**

## **PENDAHULUAN**

### **A. Latar Belakang**

Pada zaman sekarang ini, teknologi informasi semakin berkembang dan mendukung penyajian informasi secara cepat dan tepat. Dapat diketahui bahwa di dalam dunia bisnis, sistem informasi sudah menjadi bagian penting bagi sebuah perusahaan dan organisasi. Oleh sebab itu, sebuah perusahaan dan organisasi sangatlah bergantung terhadap informasi dan pengetahuan yang dimiliki, karena dengan kekuatan sebuah perusahaan dan organisasi dapat kita lihat. Dengan adanya sistem informasi dalam berbagai bidang perusahaan dan industri lainnya, membuat suatu perusahaan/industri tersebut menjadi berkembang. Menurut (Gama, 2011), pentingnya sistem informasi memberikan fungsi penyajian informasi, penyampaian informasi yang dapat diolah dan diakses secara mudah. Maka dengan adanya perkembangan teknologi informasi dan sistem informasi mampu membuat setiap orang untuk menciptakan usaha-usaha baru yang bisa meningkatkan pendapatan suatu daerah pada khususnya dan pendapatan negara pada umumnya. Penerapan sistem informasi pada otomotif merupakan hal yang penting dikarenakan akan sangat mendukung dalam pengolahan data, transaksi, dan pembuatan laporan dengan mudah secara terkomputerisasi. Contoh salah satu usaha di bidang otomotif seperti bengkel, dapat berkembang dengan menggunakan sistem informasi sebagai salah satu strategi bisnisnya. Semakin banyaknya pengguna kendaraan bermotor saat ini, persaingan dalam usaha

bengkel juga meningkat. Hampir setiap daerah usaha bengkel ini ada. Di mana pada saat ini jumlah pengguna kendaraan pribadi juga mengalami peningkatan yang sangat pesat, saat ini para pengguna kendaraan di Indonesia, khususnya sudah jarang menggunakan kendaraan umum untuk aktivitas sehari-hari. Pengguna kendaraan lebih banyak memilih kendaraan pribadi dibandingkan kendaraan umum.

Dari banyaknya jumlah pengguna kendaraan di daerah Bangkinang Kota pada tahun 2015, sebanyak 332.410 jumlah kendaraan bermotor di Kabupaten Kampar (Riau, 2015). Banyak permasalahan yang ditemui pada kendaraan, pelayanan *service* suatu bengkel menjadi jawaban dari kebutuhan tersebut. Hal lain yang sering dialami oleh pengguna kendaraan adalah kerusakan yang bisa terjadi setiap saat di manapun pada kendaraan mereka, peran bengkel menjadi faktor penting dalam menyediakan fasilitas yang dibutuhkan oleh pengguna kendaraan. Dari permasalahan tersebut maka diperlukan suatu sistem agar pengguna kendaraan bisa dengan mudah menemukan lokasi, rute, dan informasi bengkel terdekat yang ada di sekitarnya. Untuk meningkatkan jumlah pelanggan bengkel yang ada, maka pelayanan atau dukungan dari sistem informasi sangat penting untuk mempertahankan keberlangsungan bengkel tersebut khususnya untuk meningkatkan jumlah pendapatan melalui peningkatan jumlah pelanggan. Seperti fasilitas jasa *service* yang datang ke tempat pengguna kendaraan berada untuk menyediakan kemudahan penggunaan fasilitas jasa *service* melalui pembuatan aplikasi berbasis *web*.

Berdasarkan semua permasalahan yang telah dipaparkan serta demi memajukan Kecamatan Bangkinang Kota, maka topik yang diambil untuk tugas akhir ini adalah **“Rancang Bangun Sistem Informasi Montir Berbasis *Website* di Bangkinang Kota”**.

#### **B. Rumusan Masalah**

Berdasarkan latar belakang yang telah dijelaskan sebelumnya dapat diambil suatu rumusan masalah dalam penelitian ini, yaitu “Bagaimana membangun sistem informasi penyedia jasa montir berbasis *web*”. Sehingga sistem informasi pencarian montir tersebut dapat membantu masyarakat dalam menemukan montir berdasarkan kebutuhannya.

#### **C. Batasan Masalah**

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem ini dibangun untuk memudahkan masyarakat Bangkinang Kota dalam mencari jasa montir.
2. Sistem ini membahas seputar pencarian jasa montir berdasarkan nama, alamat, pengalaman dan *review* bengkel yang tertera pada *website*.

#### **D. Tujuan Penelitian**

Penelitian ini bertujuan untuk:

1. Untuk membangun sistem informasi penyedia jasa layanan montir berbasis *web* yang dapat mempermudah dan membantu masyarakat dalam mencari montir sesuai dengan permasalahan yang dialami.
2. Untuk membantu montir yang bekerja *freelance* dalam mendapatkan konsumen.

## **E. Manfaat Penelitian**

Penelitian ini bermanfaat untuk:

1. Konsumen dapat dengan mudah mencari montir sesuai dengan permasalahan yang dialami.
2. Konsumen dapat menghemat waktu dalam melakukan pencarian jasa montir yang dibutuhkan.
3. Sebagai media promosi bagi montir dalam mencari *customer*.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **A. Sistem**

Secara umum, sistem adalah suatu kesatuan yang terdiri atas komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi, atau energi untuk mencapai suatu tujuan.

Menurut (Saputra et al., 2018), sistem secara terminologi dipakai dalam berbagai macam cara yang luas sehingga sangat sulit untuk mendefinisikan atau mengartikannya sebagai suatu pernyataan yang merangkum seluruh penggunaannya dan yang cukup ringkas untuk dapat memenuhi apa yang menjadi maksudnya. Hal tersebut disebabkan oleh pengertian sistem yang bergantung dari latar belakang orang yang mencoba untuk mendefinisikannya. Menurut hukum, sistem dipandang sebagai suatu kumpulan aturan-aturan yang membatasi, baik dari kapasitas sistem itu sendiri maupun lingkungan di mana sistem itu sedang berada untuk memberikan jaminan keadilan dan keserasian.

Pengertian sistem menurut para ahli (Saputra et al., 2018):

#### 1. Pengertian Sistem menurut Arifin Rahman

Arifin Rahman mengatakan bahwa Sistem dalam kamus *Webster New Collegiate Dictionary* menyatakan bahwa kata “syn” dan “Histanai” berasal dari bahasa Yunani, artinya menempatkan bersama. Sehingga menurut Arifin Rahman bahwa pengertian sistem adalah sekumpulan

pendapat (*Collection of Opinions*), prinsip-prinsip, dan lain-lain yang telah membentuk satu kesatuan yang saling berhubungan antar satu sama lain.

## 2. Pengertian Sistem Menurut Ludwig Von Bertalanffy

Ludwig Von Bertalanffy menyatakan bahwa pengertian sistem adalah suatu kumpulan unsur yang berada pada kondisi yang saling berinteraksi.

## 3. Pengertian sistem menurut R. Fagen dan A.Hall

R.Fagen dan A.Hall menyatakan bahwa pengertian sistem adalah suatu kumpulan objek yang meliputi hubungan antara objek tersebut atau *understanding of the system is a set of objects, which includes the relationship between the object*, serta hubungan antara sifat yang mereka punya (*the relationship between their properties*).

## B. Informasi

Informasi adalah sekumpulan data atau fakta yang telah diproses dan diolah sedemikian rupa sehingga menghasilkan sesuatu yang bisa dipahami dan memberikan manfaat bagi penerimanya. Data dan fakta adalah “bahan baku” informasi, tetapi tidak semuanya bisa diolah menjadi informasi.

Selain pengertian di atas, terdapat beberapa definisi yang dirumuskan para ahli baik dari dalam maupun luar negeri.

Berikut adalah pengertian informasi menurut para ahli:

### 1. Menurut (Anggraeni & Irviani, 2017)

Informasi adalah sekumpulan data atau fakta yang diorganisasi atau diolah dengan cara tertentu sehingga mempunyai arti bagi penerima.

## 2. Menurut (Trimahardhika & Sutinah, 2017)

Informasi merupakan suatu data yang telah diolah, diklasifikasikan dan diinterpretasikan serta digunakan untuk proses pengambilan keputusan.

### C. Montir

Montir adalah orang yang mempunyai keahlian dalam memperbaiki dan memasang mesin kendaraan bermotor. Montir disebut juga sebagai mekanik baik motor, mobil, truk dan kendaraan lainnya. Pekerjaan montir adalah pekerjaan yang tidak hanya memperbaiki mobil atau kendaraan ketika terjadi kerusakan, tetapi juga melakukan pengecekan sebelum terjadi kerusakan. Bekerja menjadi mekanik harus menguasai dan memahami mekanisme mesin dan struktur mobil untuk memudahkan pekerjaan.

Ciri-ciri montir yang handal:

1. Montir akan bersedia mendengarkan keluhan,
2. Menyiapkan alternatif perbaikan yang lebih menguntungkan,
3. Memberikan detail perawatan yang lengkap,
4. Komponen lama diserahkan kembali.

Montir memiliki deskripsi kerja sebagai berikut:

1. Membuat perkiraan yang menyebabkan mobil ataupun kendaraan mengalami kerusakan,
2. Melakukan perbaikan pada mobil atau kendaraan sejenisnya sesuai dengan tingkat kerusakan yang terjadi. Sehingga ketika membongkar mesin hanya bagian yang dibutuhkan saja,

3. Melakukan pengecekan pada seluruh bagian mobil atau kendaraan sejenisnya untuk menanggulangi terjadinya kerusakan,
4. Melakukan pemesanan kepada *supplier* mengenai peralatan otomotif yang perlu perbaikan,
5. Melakukan pelaporan kondisi kendaraan setelah perawatan dan juga menyusun tanggal perawatan selanjutnya yang dilakukan secara rutin.

Minimal pendidikan montir adalah SMK atau sederajat yang berkaitan dengan dunia otomotif. Jadi sebelum bekerja sebaiknya memiliki pengalaman terlebih dahulu, seperti magang di perusahaan otomotif atau perusahaan lainnya yang berhubungan dengan otomotif. Jenjang pendidikan memiliki pengaruh dengan pengalaman dan kepercayaan ketika melamar pekerjaan.

#### **D. Website**

(Guntur & Eko, 2015), menyatakan bahwa menurut pendapat Arief (2011:7), “*Web* adalah salah satu aplikasi yang berisikan dokumen-dokumen *multimedia* (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol *Hypertext Transfer Protocol* (HTTP) dan untuk mengakses menggunakan perangkat lunak yang disebut *browser*”.

*Website* awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep *hyperlink*, yang memudahkan *surfer* atau pengguna internet melakukan penelusuran informasi di internet. Informasi yang disajikan dengan *web* menggunakan konsep multimedia, informasi dapat disajikan dengan menggunakan banyak media, seperti teks, gambar, animasi, suara, atau film.

## E. Internet

Internet adalah suatu jaringan komunikasi yang memiliki fungsi untuk menghubungkan antara satu media elektronik dengan media elektronik yang lain dengan cepat dan tepat. Jaringan komunikasi tersebut, akan menyampaikan beberapa informasi yang dikirim melalui transmisi sinyal dengan frekuensi yang telah disesuaikan. Untuk standar global dalam penggunaan jaringan internet sendiri menggunakan *Transmission Control Protocol/Internet Protocol (TCP/IP)*.

Berikut adalah definisi internet menurut para ahli:

1. Menurut (Sibero, 2011)

Internet atau yang merupakan kependekan dari *Inter-connected Network* merupakan sebuah jaringan komputer yang menghubungkan antar komputer secara *global*.

2. Menurut (Sarwono & Prihartono, 2012)

Internet merupakan sebuah kumpulan jaringan yang memiliki skala *global*. Bahkan, lebih lanjut Sarwono mengatakan bahwa tidak ada satupun orang yang mampu bertanggung jawab untuk menjalankan internet itu sendiri.

## F. XAMPP

(Afifah, 2018) menyatakan bahwa XAMPP merupakan “paket *web-server* berbasis *open source* yang dapat dipasang pada beberapa sistem operasi yang ada (Windows, Linux, dan Mac OS)”. XAMPP merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan meng-*install* XAMPP, maka tidak perlu lagi melakukan instalasi dan konfigurasi

*web-server* Apache, PHP dan MySQL secara manual (Amin, 2016).

Menurut (Madcoms, 2016), “XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla, dan lain.” XAMPP berfungsi untuk memudahkan instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan *web* memerlukan PHP, Apache, MySQL dan PhpMyadmin”.

Ada beberapa Fungsi dari XAMPP, anatra lain sebagai berikut:

1. *Setting Database* phpMyAdmin

Dengan menggunakan XAMPP halaman phpMyAdmin bisa diakses. Sama seperti saat menggunakan hosting atau *Virtual Private Server* (VPS) akan membutuhkan phpMyAdmin untuk mengedit, menghapus, menambah *database*, dan menambahkan *user*.

2. Menjalankan Laravel Pada Komputer

Laravel adalah salah satu *framework* php yang dipakai untuk mempermudah dalam mengembangkan *website*. Dengan menjalankan Laravel pada *localhost* komputer menggunakan XAMPP, jadi lebih mudah untuk memodifikasi *script* dan membuat fitur baru *website* tanpa harus takut merusak kode inti *website* yang ada di *server*.

Komponen penting pada XAMPP sebagai berikut:

1. Config

Pada menu config bisa dilakukan beberapa konfigurasi dasar untuk penggunaan XAMPP seperti mengganti:

- a) Editor: yang akan digunakan untuk mengubah *default*

*text editor.*

b) Browser: yang akan digunakan untuk akses *web server*.

## 2. Netstat

Netstat sering digunakan untuk memastikan apakah ada program/aplikasi lain yang menggunakan port XAMPP.

## 3. Shell

Menu shell pada XAMPP, digunakan untuk menampilkan *command prompt* pada Windows. Fungsinya untuk melakukan konfigurasi *web server* atau memperbaiki *error* pada *web server*.

## 4. Explorer

*Explorer* merupakan tombol *shortcut* untuk menuju ke lokasi instalasi XAMPP. Ketika diklik *icon explorer* maka akan diarahkan ke lokasi *file* instalasi XAMPP.

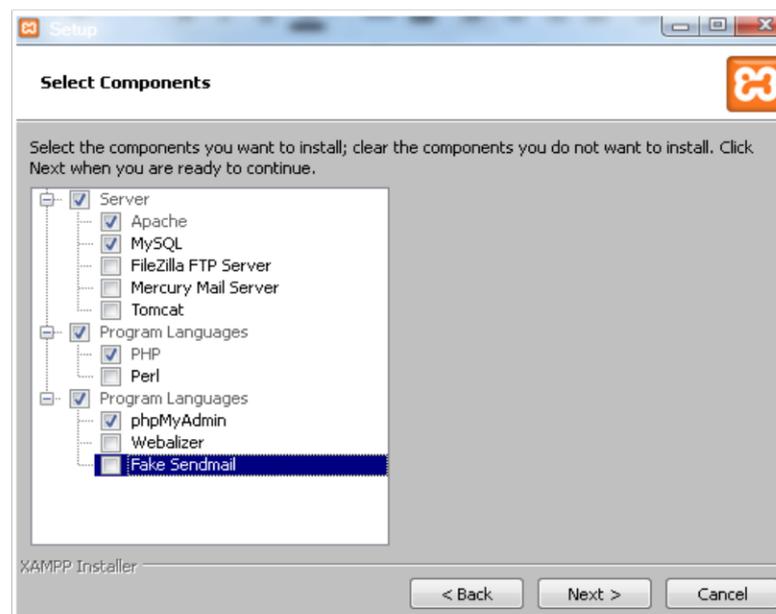
### Langkah cara instal XAMPP

1. Download XAMPP melalui *website apache friends*,
2. Lakukan instalasi setelah selesai mengunduh, klik tombol *Next* seperti pada gambar 2.1.



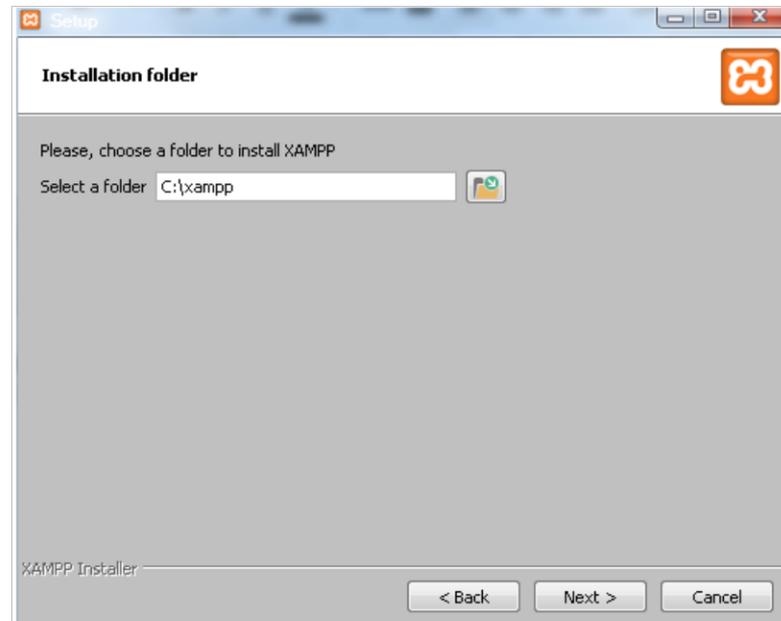
Gambar 2. 1. *Setup XAMPP*

3. Pada tampilan selanjutnya akan muncul pilihan mengenai komponen mana dari XAMPP yang ingin dan tidak ingin di instal. Beberapa pilihan seperti Apache dan PHP adalah bagian penting untuk menjalankan *website* dan akan otomatis diinstal. Selanjutnya, ceklis MySQL dan phpMyAdmin untuk pilihan lainnya biarkan saja seperti gambar 2.2.



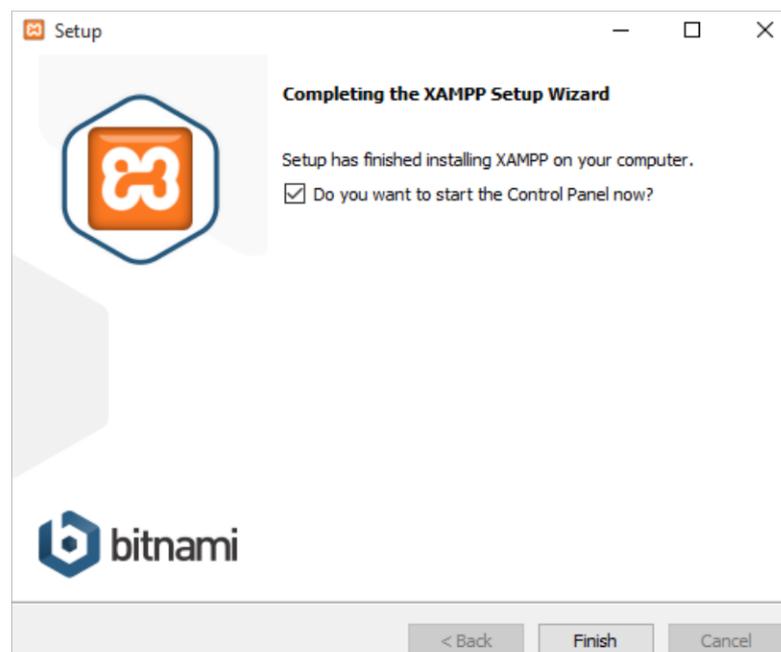
Gambar 2. 2. *Select Components*

4. Berikutnya pilih folder tujuan dimana XAMPP ingin diinstal, klik tombol *Next* seperti pada gambar 2.3.



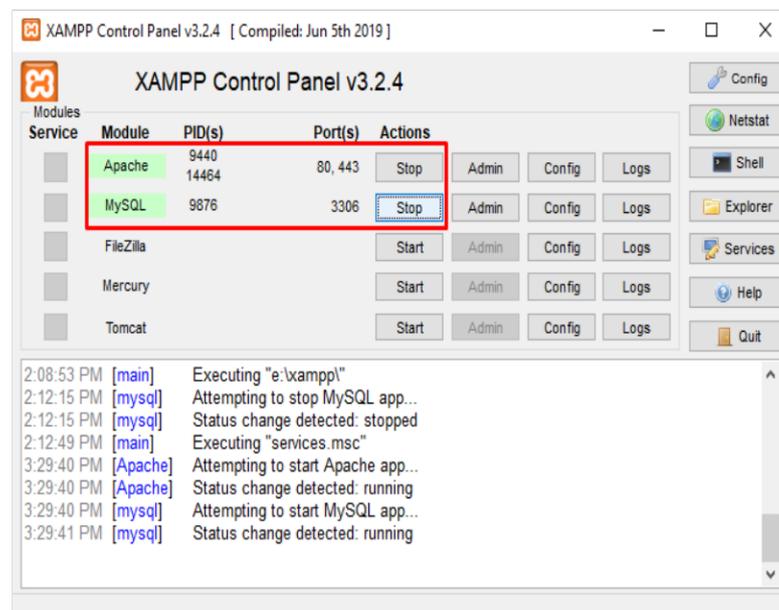
Gambar 2. 3 Pilihan folder tujuan

5. Setelah berhasil diinstal, akan muncul notifikasi untuk langsung menjalankan *control panel*. Lalu klik *Finish* seperti gambar 2.4.



Gambar 2. 4 Instalasi selesai

6. Buka aplikasi XAMPP kemudian klik tombol *Start* pada Apache dan MySQL. Jika berhasil dijalankan, Apache dan MySQL akan berwarna hijau seperti gambar 2.5.



Gambar 2. 5 XAMPP Control Panel

### G. Database (Basis Data)

Menurut (Rozaq et al., 2015), “Basis Data (*Database*) terdiri atas 2 kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya”.

Menurut (Rosa & Shalauddin, 2011), “Basis Data merupakan salah satu bagian dalam rekayasa perangkat lunak yang terkomputerisasi dan bertujuan utama memelihara data yang sudah diolah atau media

penyimpanan informasi agar dapat diakses dengan mudah dan cepat”. Sedangkan menurut (Yakub, 2015), “basis data (*database*) merupakan kumpulan data yang saling berhubungan atau punya relasi”.

Dapat disimpulkan bahwa basis data bagian dari rekayasa perangkat lunak yang terkomputerisasi sebagai media penyimpanan informasi yang saling berhubungan atau punya relasi untuk penyimpanan data informasi agar dapat diakses dengan mudah dan cepat.

## H. MySQL

Menurut (Madcoms, 2016), “MySQL adalah sistem manajemen *database* SQL yang bersifat *open source* dan paling populer saat ini. Sistem *database* MySQL mendukung beberapa fitur seperti *multithreaded*, *multiuser* dan SQL *Database Management System* (DBMS)”.

Menurut (Arizona, 2017), “MySQL adalah bahasa yang digunakan untuk mengelola data pada RDBMS”. Sedangkan menurut (Risdiyansyah, 2017), “MySQL merupakan *database server* yang bersifat *multiuser* dan *multi-threaded*. SQL adalah bahasa *database* standar yang memudahkan penyimpanan, pengubahan dan akses informasi. Pada MySQL dikenal istilah *database* dan tabel. Tabel adalah sebuah struktur data dua dimensi yang terdiri dari baris-baris *record* dan kolom”.

MySQL adalah sebuah perangkat lunak sistem manajemen basis data *Structure Query Language* (SQL) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi

GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

Tidak sama dengan proyek-proyek seperti Apache, di mana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, di mana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

## I. Visual Studio

Visual Studio adalah kode *editor* yang dikembangkan oleh Microsoft untuk Windows, Linux dan Mac OS. Pada visual studio *code* juga sudah tersedia fitur *debugging control version* seperti git dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, dan meng-*install* ekstensi yang menambah fungsionalitas tambahan dengan mudah.

Microsoft visual studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi *personal*, ataupun komponen aplikasinya, dalam bentuk aplikasi *console*, aplikasi Windows, ataupun aplikasi *web*. Visual studio mencakup kompiler, *Software Development Kit* (SDK), *Integrated Development Environment* (IDE), dan dokumentasi (umumnya

berupa MSDN *library*). Kompiler yang dimasukkan ke dalam paket visual studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, VisualJ++, Visual J#, Visual FoxPro, dan Visual SourceSafe.

Microsoft visual studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bentuk bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (dalam bentuk Microsoft *intermediate language* di atas .NET Framework). Selain itu, visual studio juga dapat digunakan untuk mengembangkan aplikasi *Silverlight*, aplikasi Windows *mobile* (yang berjalan di atas .NET *Compact Framework*).

#### **J. Hypertext Markup Language (HTML)**

Menurut (Hidayatullah & Kawistara, 2015), “*Hypertext Markup Language* (HTML) adalah bahasa standar yang digunakan untuk menampilkan halaman *web*”. Yang bisa dilakukan dengan HTML menurut (Hidayatullah & Kawistara, 2015), yaitu:

1. Mengatur tampilan dari halaman *web* dan isinya,
2. Membuat tabel dalam halaman *web*,
3. Mempublikasikan halaman *web* secara *online*,
4. Membuat *form* yang bisa digunakan untuk menangani registrasi dan transaksi *via web*.

Bahasa ini disebut *Hypertext* karena ini dari halaman-halaman *web* tersebut tidak hanya berupa teks saja tetapi dapat berupa multimedia ataupun *link* yang menghubungkan halaman *web* satu dengan yang lainnya.

Sebuah dokumen HTML diawali dengan tag pembuka `<html>` dan diakhiri dengan tag penutup `</html>` dan didalamnya terdiri dari dua elemen utama yaitu: *Head*: berisi informasi *browser* dan halaman *web*. Serta *Body*: memuat isi atau *content* yang akan ditampilkan pada halaman *web*.

### **K. *Cascading Style Sheets (CSS)***

Menurut (Abdulloh, 2015), *Cascading Style Sheets (CSS)* adalah “skrip yang digunakan untuk mengatur desain *website*”. Walaupun HTML mempunyai kemampuan untuk mengatur tampilan *website*, namun kemampuannya sangat terbatas. Fungsi CSS adalah memberikan pengaturan yang lebih lengkap agar struktur *website* yang dibuat dengan HTML terlihat lebih rapi dan indah.

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada teks, warna tabel, ukuran border, warna border, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda.

### **L. *Hypertext Preprocessor (PHP)***

(Ripai, 2017), menyatakan bahwa “PHP adalah bahasa pemrograman yang bisa disisipkan kedalam *script* HTML dan berjalan di sisi *server*, biasanya dipakai untuk membuat *website* agar lebih dinamis”.

PHP juga dapat berjalan pada berbagai *web server* seperti *Internet Information Server (IIS)*, *Personal Web Server (PWS)*, Apache, Xitami. PHP juga mampu berjalan di banyak sistem operasi yang beredar saat ini, diantaranya : sistem operasi Microsoft Windows (semua versi), Linux, Mac OS dan Solaris. PHP dapat dibangun sebagai modul *web server* Apache dan

sebagai *binary* yang dapat berjalan sebagai *Common Gateway Interface* (CGI).

Sama halnya dengan bahasa pemrograman lainnya, PHP juga memiliki kelebihan dan kekurangannya. Adapun Kelebihan dari PHP, yaitu:

1. Mudah dipelajari,
2. Program yang ringkas,
3. Program *open source*,
4. Proses *upgrade* yang mudah.

Setelah mengetahui kelebihan dari PHP. Berikut beberapa kekurangan dari PHP, yaitu:

1. Persaingan ketat,
2. Tidak dilengkapi dengan tipe data,
3. Rawan pembajakan.

#### **M. *Framework* Laravel**

Menurut (Awaludin, 2016), seorang *Senior Web Developer* yang telah menjabarkan tentang setiap fungsi *framework* laravel di dalam bukunya “Menyelami *Framework* Laravel”, manfaat menggunakan Laravel adalah berbagai abstraksi yang dimiliki oleh Laravel dapat lebih fokus memikirkan *logic* bisnis dari aplikasi yang dibuat. Hal ini disebabkan oleh waktu tidak perlu lagi disia-siakan dengan mengurus hal-hal dasar di PHP.

Sedangkan tujuan menggunakan Laravel pada sistem yang dibangun saat ini adalah untuk lebih menjaga keamanan *website* tersebut, dan mempercepat proses pekerjaan setiap karyawan. Laravel lebih dipilih karena

hanya Laravel yang memiliki sifat *templating* yang lebih mudah dari sifat *templating framework*, yaitu Laravel bekerja di bawah level suatu *library*. Jadi, jika suatu *project* membutuhkan sebuah fungsi merubah *text* menjadi excel maka pada *native* PHP harus membuat *script function* untuk hal tersebut agar tidak dibuat berulang-ulang. Sedangkan pada Laravel sudah di sediakan berbagai macam *library* sesuai kebutuhan sistem saat itu.

#### **N. JavaScript**

Bahasa pemrograman yang bersifat *client side* yang pemrosesannya dilakukan oleh *client* sering digunakan pada *web browser* untuk menciptakan halaman *web* yang menarik.

Menurut (Kadir & Triwahyuni, 2013), “JavaScript adalah bahasa pemrograman yang biasa diletakkan bersama kode HTML untuk menentukan suatu tindakan”.

Sedangkan Menurut (Sibero, 2011), “Javascript adalah bahasa skrip (*scripting language*), yaitu kumpulan instruksi perintah yang digunakan untuk mengendalikan beberapa bagian dari sistem operasi”.

Berdasarkan pendapat yang dikemukakan di atas dapat disimpulkan bahwa, JavaScript adalah bahasa pemrograman atau bahasa skrip yang berisi kumpulan instruksi perintah yang diletakkan bersama kode HTML.

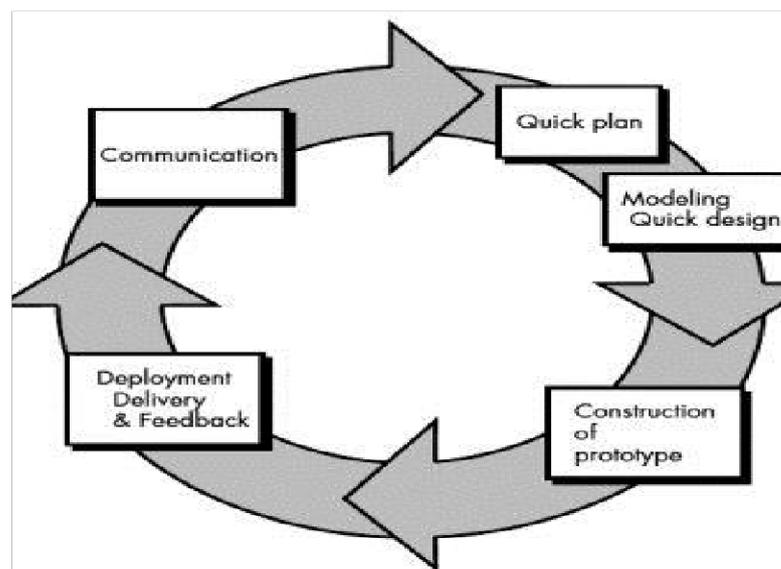
#### **O. Prototype Model**

*Prototype* adalah model atau simulasi dari alur pengembangan suatu produk, model ini harus bersifat representatif dari produk akhirnya. Pada pengembangan sistem seringkali terjadi keadaan di mana pengguna sistem

sebenarnya telah mendefinisikan secara umum atau tujuan perangkat lunaknya meskipun belum mendefinisikan secara rinci masukan, proses dan keluaran.

Sebuah *prototype* merupakan versi awal dari sistem perangkat lunak yang digunakan untuk menampilkan konsep-konsep, percobaan rancangan, dan menemukan lebih banyak masalah dan solusi yang memungkinkan (Fanani et al., 2020).

Sistem *prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik. Penggunaan metode *prototyping* di dalam penelitian ini bertujuan agar gambaran aplikasi yang akan dibangun melalui tahap pembangunan aplikasi *prototype* yang akan dievaluasi oleh *user*. Aplikasi *prototype* yang telah dievaluasi oleh *user* selanjutnya akan dijadikan acuan untuk membuat aplikasi yang dijadikan produk akhir sebagai keluaran dari penelitian ini.



Gambar 2. 6 *Prototype Model*  
Sumber : (Pressman: 2010)

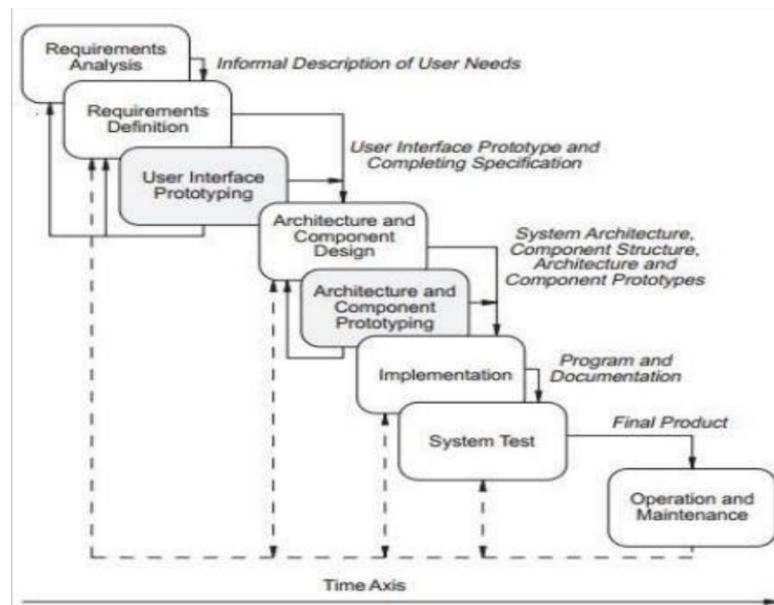
Gambar 2.6 menjelaskan bahwa metode *prototyping* dimulai dengan mendengarkan kebutuhan dan masukan dari pengguna. Pengembang dan pengguna bertemu dan bersama-sama menentukan tujuan keseluruhan untuk perangkat lunak dan mengidentifikasi apapun persyaratan yang diperlukan. Lalu pengembang membuat sebuah gambaran tentang aplikasi yang selanjutnya dapat dipresentasikan kepada pelanggan.

Berdasarkan model *prototype* yang telah digambarkan diatas, maka dapat diuraikan pembahasan masing-masing tahap dalam model tersebut adalah sebagai berikut:

1. *Communication*/komunikasi tim perancang perangkat lunak melakukan pertemuan dengan para *stakeholder* untuk menentukan kebutuhan perangkat lunak yang saat itu diketahui dan untuk menggambarkan area-area dimana definisi lebih jauh untuk literasi selanjutnya.
2. *Quick Plan*/perencanaan secara cepat dalam perencanaan ini literasi pembuatan *prototype* dilakukan secara cepat. Setelah itu dilakukan pemodelan dalam bentuk “rancangan cepat”.
3. *Modeling Quick Design*/model rancangan cepat pada tahap ini dilakukan pemodelan perencanaan ditahap sebelumnya dengan menggunakan pemodelan terstruktur dalam bentuk *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD) dan *Flowchart* untuk menggambarkan analisis dan desain sistem.
4. *Construction of Prototype*/pembuatan *prototype* dalam pembuatan rancangan cepat berdasarkan pada representasi aspek-aspek perangkat

lunak yang akan terlihat oleh para *end user* (misalnya rancangan antarmuka pengguna atau format tampilan). Rancangan cepat merupakan dasar untuk memulai konstruksi pembuatan *prototype*.

5. *Deployment Delivery & Feedback*/penyerahan dan memberikan umpan balik terhadap pengembangan *prototype* kemudian diserahkan kepada para *stakeholder* untuk mengevaluasi *prototype* yang telah dibuat sebelumnya dan memberikan umpan-balik yang akan digunakan untuk memperbaiki spesifikasi kebutuhan. Literasi terjadi saat pengembang melakukan perbaikan terhadap *prototype* tersebut.



**Gambar 2.7** Alur proses metode *Prototype*

Gambar 2.7 menjelaskan mengenai alur pengembangan sistem dengan menggunakan metode *Prototyping Oriented Software*. Pada tahap pertama, dilakukan analisis kebutuhan dan pendefinisian kebutuhan. Kebutuhan yang dimaksudkan disini adalah kebutuhan pelanggan atau pengguna. Selanjutnya pada tahap kedua dilakukan pembuatan *prototype* dari aplikasi yang akan

dibangun, mulai dari *user interface prototyping* dan dilanjutkan hingga penyusunan arsitektur dan komponen-komponen yang berkaitan dengan aplikasi yang akan dibangun. Selanjutnya dilakukan pengembangan sistem, di mana aplikasi akan dibangun sesuai dengan *prototype* yang telah dibuat sebelumnya, dan setelah aplikasi berhasil dibuat sesuai dengan kebutuhan maka dilakukan proses pengujian aplikasi sebelum aplikasi tersebut diimplementasikan.

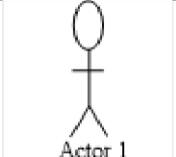
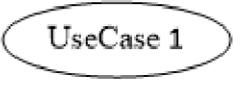
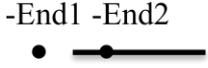
#### P. *Unified Modelling language (UML)*

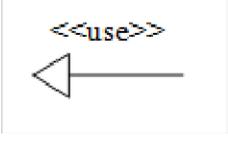
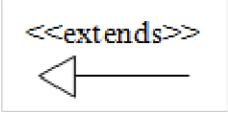
Menurut (Sulianta, 2019) dalam buku Teknik Perancangan Arsitektur Sistem Informasi “*Unified Modelling language (UML)* merupakan kumpulan diagram-diagram yang sudah memiliki standar untuk membangun perangkat lunak berbasis objek”. UML memiliki banyak diagram diantaranya:

##### 1. *Use Case Diagram*

*Use Case Diagram* merupakan diagram yang harus dibuat pertama kali saat pemodelan perangkat lunak berorientasi objek dilakukan. Tabel 2.1 menunjukkan simbol yang digunakan pada *Use Case Diagram*.

**Tabel 2. 1 *Use Case Diagram***

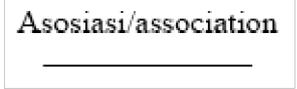
Simbol	Nama	Keterangan
	Aktor	Merupakan penggunaan dari sistem. Penamaan aktor menggunakan kata benda
	<i>Use Case</i>	Merupakan pekerjaan yang dilakukan oleh aktor. Penamaan <i>use case</i> dengan kata kerja.
	Asosiasi	Hubungan antara aktor dengan <i>use case</i>

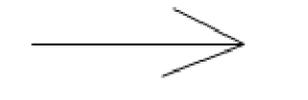
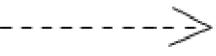
Simbol	Nama	Keterangan
	<i>Include</i>	Hubungan antara <i>use case</i> dengan <i>use case</i> , <i>include</i> menyatakan bahwa sebelum pekerjaan dilakukan harus mengerjakan pekerjaan lain terlebih dahulu.
	<i>Extends</i>	Hubungan antara <i>use case</i> dengan <i>use case</i> , <i>extends</i> menyatakan bahwa jika pekerjaan yang dilakukan tidak sesuai atau terdapat kondisi khusus, maka lakukan pekerjaan itu.

## 2. Class Diagram

*Class Diagram* dibuat setelah diagram *use case* dibuat terlebih dahulu. Pada diagram ini harus menjelaskan hubungan apa saja yang terjadi antara suatu objek dengan objek lainnya sehingga terbentuklah suatu sistem aplikasi. Simbol yang digunakan untuk membuat *class diagram* terlihat pada tabel 2.2.

Tabel 2. 2 Simbol-simbol *Class Diagram*

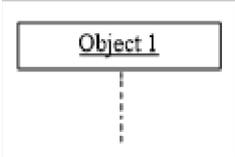
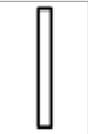
Simbol	Deskripsi
<p><b>kelas</b></p> 	Kelas pada struktur sistem
<p><b>Antar muka/Interface</b></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
<p><b>Asosiasi/association</b></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .

Simbol	Deskripsi
<p>Asosiasi berarah/ directed association</p> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
<p>Kebergantungan/dependency</p> 	Relasi antar kelas dengan makna ketergantungan antar kelas.
<p>Agresiasi/aggregation</p> 	Relasi antar kelas dengan makna semua-bagian.

### 3. Sequence Diagram

*Sequence diagram* adalah diagram yang dibuat untuk mengetahui alur dari interaksi antar objek. Isi dari *sequence diagram* harus sama dengan *use case* dan diagram kelas, seperti yang terlihat pada tabel 2.3.

Tabel 2. 3 Simbol-simbol *Sequence Diagram*

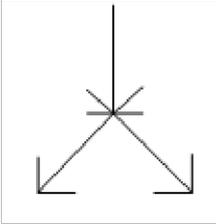
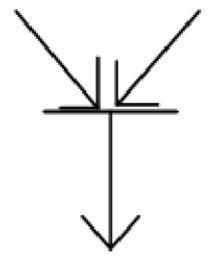
Simbol	Nama	Keterangan
	Objek/aktor	Sebuah objek yang berasal dari kelas. Atau dapat dinamai dengan kelasnya saja. Aktor termasuk objek. Garis putus-putus menunjukkan garis hidup suatu objek.
	Aktivasi	Menunjukkan masa hidup dari objek.
<p>Meessage 1</p> 	Pesan	Interaksi antara satu objek dengan objek lainnya. Objek dapat mengirimkan pesan ke objek lain. Interaksi antar objek ditunjukkan pada bagian operasi pada diagram kelas.

Simbol	Nama	Keterangan
	Return	Pesan kembalian dari komunikasi antar objek.

4.

Menggambarkan aktifitas-aktifitas, *object*, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas, seperti yang terlihat pada tabel 2.4.

Tabel 2. 4 Simbol-simbol Activity Diagram

Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
	<i>End Point</i> , akhir aktivitas.
	<i>Activities</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penganbungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.

## **BAB III**

### **METODE PENELITIAN**

#### **A. Lokasi dan Subjek Penelitian**

Dalam penulisan tugas akhir ini penulis melakukan penelitian di daerah Bangkinang Kota, Kecamatan Bangkinang Kota, Kabupaten Kampar, Provinsi Riau. Lokasi tersebut dipilih karena saat proses penelitian, sangat disadari bahwasanya sulit mencari montir yang di inginkan. Oleh sebab itu, Kecamatan Bangkinang Kota dipilih sebagai lokasi penelitian.

Subjek dalam penelitian ini adalah pemilik bengkel, selain itu untuk menunjang kelengkapan data penelitian, mahasiswa juga menjadi subjek penelitian.

#### **B. Populasi dan Sampel**

##### **1. Populasi**

Berdasarkan survei yang telah dilakukan, populasi yang digunakan dalam penelitian ini sebanyak 30 bengkel yang tersebar di Bangkinang Kota.

##### **2. Sampel**

Pemilik bengkel yang akan dijadikan sasaran sampel penelitian menggunakan taraf kesalahan 5%. Untuk menghitung penentuan jumlah sampel dari populasi tertentu yang dikembangkan, maka pengambilan sampel menggunakan rumus Slovin.

$$n = \frac{N}{N \cdot (d^2) + 1}$$

Dimana:

n : ukuran sampel

N : populasi

d : taraf nyata atau batas kesalahan

### C. Metode Pengumpulan Data

#### 1. Observasi

Observasi dilaksanakan dengan pengamatan atau peninjauan langsung di Bangkinang Kota. Dalam pelaksanaan observasi dilakukan di:

Nama tempat : Bangkinang Kota, Kampar.

Waktu Pelaksanaan : 13 september 2021 - 16 september 2021

Observasi digunakan untuk mendapatkan data yang dibutuhkan dalam penelitian sistem informasi montir di Bangkinang Kota.

#### 2. Wawancara

Wawancara atau *interview* adalah sebuah proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab sambil bertatap muka antara pewawancara dengan orang yang diwawancarai, dengan atau tanpa menggunakan pedoman wawancara serta melakukan wawancara dengan pemilik bengkel.

Tabel 3. 1 Pedoman wawancara

No	Aspek yang ditanyakan	Indikator	Item pertanyaan
1	a. Riwayat pendidikan montir b. Tentang bengkel dan montir	a. Riwayat pendidikan montir b. Pendidikan terakhir montir c. Perizinan terhadap bengkel d. Kendala yang dialami oleh montir terhadap pelanggan	a. Bagaimana riwayat pendidikan anda? b. Apa pendidikan terakhir anda? c. Apakah bengkel ini bersedia apabila kami masukkan ke <i>web</i> dalam penyelesaian tugas akhir? d. Apakah kendala yang sering dialami oleh montir saat memperbaiki kendaraan pelanggan?
2	Fitur-fitur montir	a. Fitur-fitur yang diinginkan	<ul style="list-style-type: none"> <li>• Seperti apa fitur yang Ibu/Bapak inginkan di dalam <i>website</i> ?</li> <li>• Apakah sebelumnya Ibu/Bapak pernah berlangganan pada <i>website</i> montir?</li> </ul>

No	Aspek yang ditanyakan	Indikator	Item pertanyaan
			<ul style="list-style-type: none"> <li data-bbox="1078 344 1347 674">• Jika pernah, apa fitur yang Ibu/Bapak inginkan tetapi tidak ada pada <i>website</i> penyedia montir tersebut?</li> <li data-bbox="1078 707 1347 920">• Jika tidak pernah, apakah kesulitan Ibu/Bapak dalam mencari montir?</li> </ul>

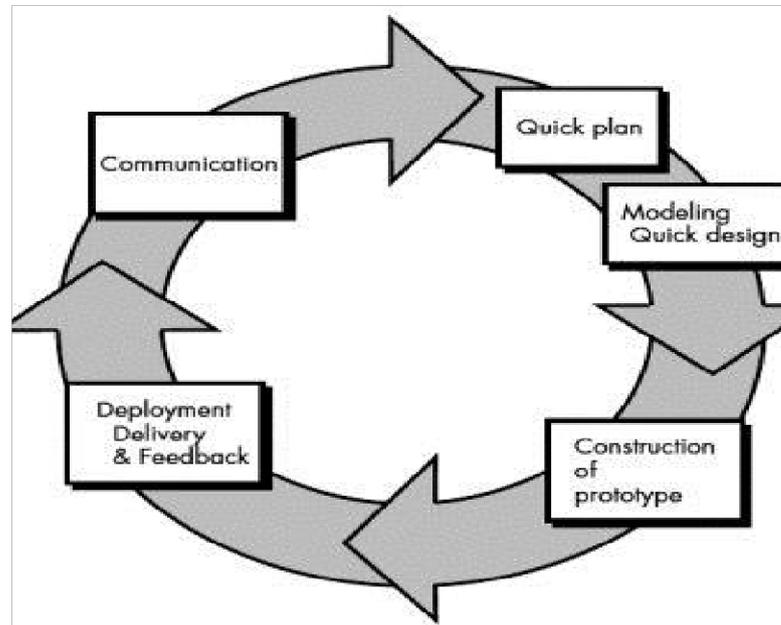
### 3. Dokumentasi

Suatu pengumpulan data dengan cara melihat langsung sumber- sumber dokumen yang terkait. Dengan arti lain bahwa dokumentasi sebagai pengambilan data melalui dokumen tertulis maupun elektronik. Digunakan sebagai pendukung kelengkapan data yang lain.

#### D. Metode Analisis

Untuk melakukan penelitian data yang dibutuhkan dalam proses seleksi montir, digunakan metode kualitatif. Dengan metode ini, dilakukan wawancara dan observasi secara langsung. Tujuannya agar mendapatkan informasi dan data yang akurat mengenai montir yang ada di Kecamatan Bangkinang Kota.

## E. Metode Perancangan



Gambar 3. 1 Metode *Prototype*

Metode yang digunakan dalam pengembangan sistem ini adalah *prototype*. Dalam pengembangan *prototype*, pelanggan dapat melakukan evaluasi kembali terhadap rancangan yang telah dibuat. Dengan menggunakan model pengembangan *prototype* akan sangat membantu bagi *developer* untuk menjaga efisiensi dari algoritma yang digunakan.

Selanjutnya akan dijelaskan secara detail beberapa tahap proses yang digunakan pada pengembangan sistem model *prototype* ini. Diantaranya adalah sebagai berikut:

### a. *Communication*

Dilakukan proses analisis terhadap kebutuhan pengguna sistem yang sangat dibutuhkan komunikasi untuk mencapai sebuah tujuan yang ingin dicapai dengan cara melakukan wawancara langsung kepada pelanggan.

b. *Quick plan*

Dibuat beberapa rancangan *prototype*, yakni dengan memperkirakan sumber-sumber daya yang diperlukan, produk yang akan dihasilkan, dan melakukan *tracking* dalam proses pengerjaan sistem.

c. *Modelling Quick Design*

Dilakukan pemodelan perencanaan di tahap sebelumnya dengan menggunakan pemodelan berorientasi objek dalam bentuk *class diagram*, untuk menggambarkan dan menganalisis desain sistem.

d. *Construction of Prototype*

Pada tahap ini akan dilakukan pembuatan *prototype* dengan memperlihatkan tampilan *interface* kepada para *end user*.

e. *Deployment Delivery & Feedback*

*Prototype* yang telah dibuat pada tahap sebelumnya kemudian diserahkan kepada para *stakeholder* untuk dilakukan evaluasi dan menerima *feedback* yang akan digunakan untuk memperbaiki spesifikasi kebutuhan.

## **F. Implementasi**

Implementasi terdapat 3 bagian yaitu implementasi *database*, implementasi sistem, serta pengujian sistem.

### 1. Implementasi *Database*

Implementasi *database* dirancang melalui rancangan yang telah ditentukan sebelumnya. Pengimplementasian *database* dilakukan pada sebuah *local server* yaitu pada phpmyadmin (MySQL).

### 2. Implementasi Sistem

Setelah perancangan dirancang, lakukan implementasi sistem dengan bahasa pemrograman PHP. Selanjutnya sistem akan dibangun dengan kode program dengan menggunakan bahasa PHP *tools* Sublime sebagai *text editor*, dan Xampp sebagai *server* lokal untuk membangun sistem.

## **G. Teknik Pengujian**

Metode pengujian yang dilakukan adalah metode pengujian langsung, yaitu *blackbox testing*. *Blackbox testing* digunakan untuk menguji fungsi khusus dari perangkat lunak yang dirancang. Perangkat lunak diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diketahui.

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **A. Analisa Sistem Berjalan**

Analisa sistem berjalan merupakan langkah awal yang harus dilaksanakan sebelum memberikan sistem usulan. Berdasarkan analisa akan dijadikan alasan kenapa sistem usulan dibutuhkan. Analisa sistem berupa identifikasi permasalahan yang akan diselesaikan. Saat ini jumlah pengguna kendaraan pribadi di Indonesia juga mengalami peningkatan yang sangat pesat, diantaranya sudah jarang menggunakan kendaraan umum untuk aktivitas sehari-hari. Pengguna kendaraan lebih banyak memilih kendaraan pribadi dibandingkan kendaraan umum.

Pelayanan *service* suatu bengkel menjadi jawaban dari kebutuhan tersebut. Hal lain yang sering dialami oleh pengguna kendaraan adalah kerusakan yang bisa terjadi setiap saat di manapun pada kendaraan mereka. Dengan adanya faktor tersebut, peran bengkel menjadi penting dalam menyediakan fasilitas yang dibutuhkan oleh pengguna kendaraan.

#### **B. Identifikasi Permasalahan**

Berdasarkan observasi yang dilakukan pada bengkel, ditemukan suatu permasalahan yang terjadi pada pengguna kendaraan yaitu kerusakan yang bisa terjadi setiap saat dan di manapun pada kendaraan mereka. Susahnya mencari bengkel apabila ada kerusakan kendaraan ditengah jalan maka peran bengkel menjadi faktor penting dalam menyediakan fasilitas yang dibutuhkan oleh pengguna kendaraan. Dari permasalahan tersebut maka diperlukan suatu

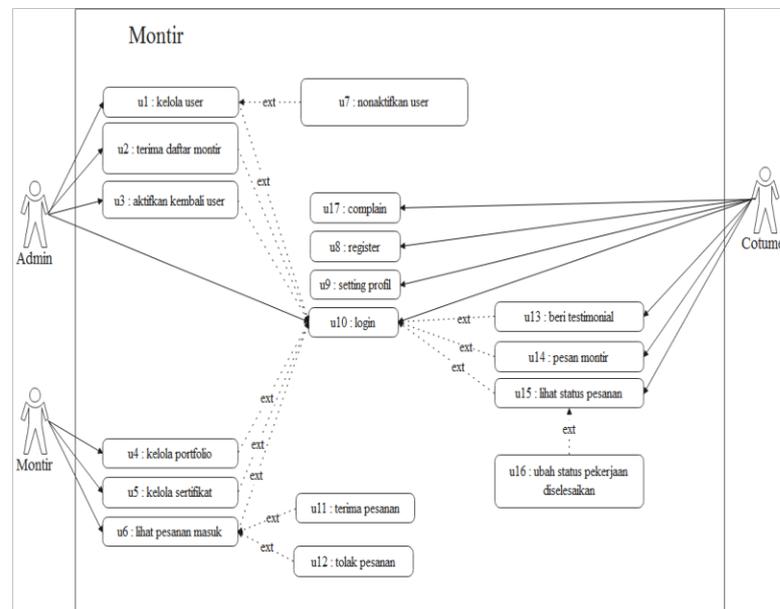
sistem agar pengguna kendaraan bisa dengan mudah menemukan lokasi, rute, dan informasi bengkel terdekat yang ada di sekitarnya.

Untuk meningkatkan jumlah pelanggan bengkel yang ada, maka pelayanan atau dukungan dari sistem informasi sangat penting untuk mempertahankan keberlangsungan bengkel tersebut khususnya untuk meningkatkan jumlah pendapatan melalui peningkatan jumlah pelanggan. Seperti fasilitas jasa *service* yang datang ketempat pengguna kendaraan berada untuk menyediakan kemudahan penggunaan fasilitas jasa *service* melalui pembuatan aplikasi berbasis *web*.

### C. Perancangan Sistem dengan *Unified Modelling Language* (UML)

Pada tahap perancangan sistem digunakan *unified modelling language* (UML). Diagram UML yang akan digunakan adalah *use case diagram* sebagai *tools* untuk memperlihatkan hubungan pengguna dengan sistem, dan *class diagram* sebagai *tools* untuk menggambarkan struktur data sebuah system.

## 1. Use Case Diagram



Gambar 4.1 Use Case Diagram

### a. Business Actor

*Business actor* menggambarkan interaksi pengguna dengan sistem. Seorang *actor* dapat memberikan informasi masukan kepada sistem. Sistem dapat menerima informasi dan dapat memberikan informasi. Pada sistem informasi montir ini, terdapat 3 aktor sebagai berikut :

#### 1) Costumer

*Costumer* merupakan orang yang menggunakan layanan jasa montir. Aktor ini bisa mengakses sistem untuk *register*, *login*, *setting profil*, beri testimoni, pesan montir, ubah status pekerjaan diselesaikan dan lihat status pesanan.

#### 2) Admin

Admin merupakan orang yang mengelola sistem dengan tugas melakukan validasi data sistem montir, admin dapat mengakses *login*, *kelola user*, terima daftar montir, aktifkan kembali *user* dan nonaktifkan *user*.

### 3) Montir

Montir merupakan aktor yang mempunyai usaha bengkel yang dapat mengakses sistem untuk *register*, *setting profil*, *login*, kelola portfolio, kelola sertifikat, lihat pesanan masuk, terima pesanan, dan tolak pesanan.

#### b. *Business Use Case List*

Setiap *use case* adalah cuplikan bisnis. Pada tahap ini *use case* dapat melibatkan komunikasi dua arah antara sejumlah *actor*. Berikut adalah *list business use case*:

- 1) B1 : *Costumer* mengakses *website* montir mencari bengkel yang terdekat atau bisa menghubungi montir bengkel tersebut.
- 2) B2 : Jika *costumer* telah menemukan bengkel yang ingin dituju, maka *costumer* harus melakukan *login* apabila telah memiliki akun pada *website* tersebut. Bagi *costumer* yang belum memiliki akun diharuskan mengisi *form register*.
- 3) B3 : *Costumer* yang telah *login* dapat melengkapi data diri pada bagian *setting profile*.

- 4) B4 : *Costumer* yang telah melengkapi data diri dapat melanjutkan pesan montir .
- 5) pB5 : *Costumer* yang telah menyelesaikan proses pesan montir akan diarahkan pada halaman lihat status pesanan.
- 6) B6 : Montir mengakses *website* montir untuk kelola portfolio, kelola sertifikat, dan lihat pesanan masuk yang ditampilkan pada *website*.
- 7) B7 : Jika ada yang pesan masuk, maka montir dapat menerima pesanan atau menolak pesanan.

c. *System Actor*

Aktor sistem adalah entitas dari dunia nyata yang berinteraksi dengan sistem melalui *use case*. Pada *website* montir ini terdapat tiga aktor yang memiliki peran berbeda-beda, yaitu admin sebagai pengelola *web*, *costumer* sebagai penyewa jasa montir, montir sebagai penyedia jasa perbaiki kendaraan.

d. *System Use Case*

*System use case* menyediakan alur kerja khusus *system* berikutnya. Adapun *system use case* sebagai berikut:

- 1) U1 Kelola *user*: admin dapat mengelola semua *user* pada *website*.
- 2) U2 Terima daftar montir: admin menerima pendaftar montir.

- 3) U3 Aktifkan kembali *user*: admin dapat mengaktifkan kembali *user*.
- 4) U4 Kelola portfolio: montir dapat mengelola portpolio pada *website*.
- 5) U5 Kelola sertifikat: montir dapat mengelola sertifikat
- 6) U6 Lihat pesanan masuk: montir dapat melihat pesanan masuk pada *website*.
- 7) U7 Nonaktifkan *user*: admin dapat menonaktifkan *user*.
- 8) U8 *Register*: semua pengguna *website* dapat melakukan pendaftaran akun.
- 9) U9 *Setting profile*: semua pengguna *website* dapat melakukan *setting profile*.
- 10) U10 *Login*: pengguna *website* dapat melakukan *login* pada sistem.
- 11) U11 Terima pesanan: montir dapat menerima pesanan
- 12) U12 Tolak pesanan: montir dapat menolak pesanan
- 13) U13 Beri testimoni: *costumer* dapat memberikan testimoni.
- 14) U14 Pesan montir: *costumer* dapat melakukan pemesanan montir.
- 15) U15 Lihat status pesanan: *costumer* dapat melihat status pesanan

16) U16 Ubah status pekerjaan diselesaikan: *costumer* dapat mengubah status pekerjaan diselesaikan.

17) U17 *Complain*: *costumer* dapat melakukan *Complain* kepada montir.

e. Deskripsi *Use Case Diagram*

Pada tahap ini menjelaskan deskripsi *use case* antara *actor* dan sistem.

1) Deskripsi *use case admin* sistem informasi montir berbasis *website* di Bangkinang Kota seperti tabel 4.1.

**Tabel 4. 1 Deskripsi Use Case Admin**

No	Use case	Deskripsi
1	U1 Kelola <i>user</i>	Admin dapat mengelola semua <i>user</i> pada <i>website</i>
2	U2 Terima daftar montir	Admin menerima pendaftar montir
3	U3 Aktifkan kembali <i>user</i>	Admin dapat mengaktifkan kembali <i>user</i>
4	U7 Nonaktifkan <i>user</i>	Admin dapat menonaktifkan <i>user</i>
5	U8 <i>Register</i>	Pengguna dapat melakukan <i>register</i>
6	U10 <i>Login</i>	Pengguna dapat melakukan <i>login</i>

2) Deskripsi *use case costumer* sistem informasi montir berbasis *website* di Bangkinang Kota seperti tabel 4.2.

**Tabel 4. 2 Deskripsi Use Case Costumer**

No	Use case	Deskripsi
1	U13 Beri testimoni	<i>Costumer</i> dapat memberikan testimoni

No	Use case	Deskripsi
2	U14 Pesan montir	Costumer dapat melakukan pemesanan montir
3	U15 Lihat status pesanan	Costumer dapat melihat status pesanan
4	U16 Ubah status pekerjaan diselesaikan	Costumer dapat mengubah status pekerjaan diselesaikan.
5	U8 Register	Pengguna dapat melakukan <i>register</i>
6	U17 Complain	Pengguna dapat melakukan <i>complain</i>
7	U9 Setting profile	Pengguna dapat melakukan <i>setting profile</i>
8	U10 Login	Pengguna dapat melakukan <i>login</i>

- 3) Deskripsi *use case* montir pada sistem informasi montir berbasis *website* di Bangkinang Kota seperti tabel 4.3.

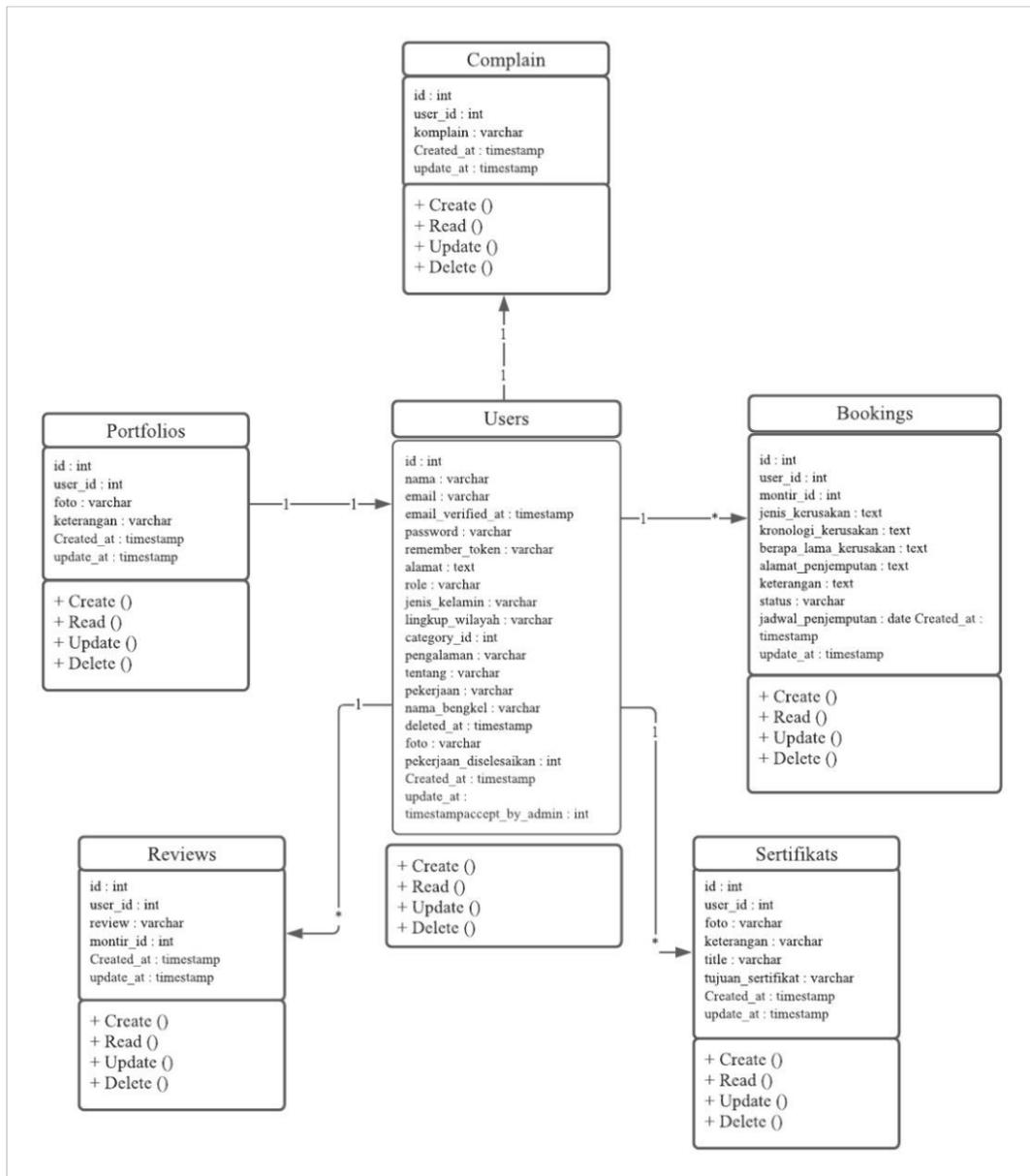
Tabel 4. 3 Deskripsi Use Case Montir

No	Use case	Deskripsi
1	U4 Kelola portfolio	Montir dapat mengelola portpolio pada <i>website</i> .
2	U5 Kelola sertifikat	Montir dapat mengelola sertifikat
3	U6 Lihat pesanan masuk	Montir dapat melihat pesanan masuk pada <i>website</i> .
4	U8 Register	Montir dapat melakukan <i>register</i>
5	U9 Setting profile	Montir dapat melakukan <i>setting profile</i>
6	U10 Login	Montir dapat melakukan <i>login</i>
7	U11 Terima pesanan	Montir dapat menerima pesanan
8	U12 Tolak pesanan	Montir dapat menolak pesanan

## 2. Class Diagram

*Class diagram* merupakan penjelasan lengkap dari beberapa *class* yang ditangani oleh sistem. Dimana tiap-tiap *class* dipasangkan

dengan beberapa *attribute* dan *method* yang diperlukan. Berikut adalah skema dari model *class diagram* Rancang Bangun Sistem Informasi Montir Berbasis *Website* di Kecamatan Bangkinang Kota yang dapat dilihat pada gambar 4.2.



Gambar 4.2 Class Diagram

Tabel 4. 4 Kelas, Atribut dan *Method*

No	Nama kelas	Atribut	<i>Method</i>
1	Users	id : int <sup>[11]</sup> <sub>[SEP]</sub> nama : varchar <sup>[11]</sup> email : varchar <sup>[11]</sup> <sub>[SEP]</sub> email_verified_at : timestamp <sup>[11]</sup> <sub>[SEP]</sub> password : varchar <sup>[11]</sup> <sub>[SEP]</sub> remember_token : varchar <sup>[11]</sup> <sub>[SEP]</sub> alamat : text <sup>[11]</sup> <sub>[SEP]</sub> role : varchar <sup>[11]</sup> <sub>[SEP]</sub> jenis_kelamin : varchar <sup>[11]</sup> <sub>[SEP]</sub> lingkup_wilayah : varchar <sup>[11]</sup> <sub>[SEP]</sub> category_id : int <sup>[11]</sup> <sub>[SEP]</sub> pengalaman : varchar <sup>[11]</sup> tentang : varchar <sup>[11]</sup> <sub>[SEP]</sub> pekerjaan : varchar <sup>[11]</sup> <sub>[SEP]</sub> nama_bengkel : varchar <sup>[11]</sup> <sub>[SEP]</sub> deleted_at : timestamp <sup>[11]</sup> <sub>[SEP]</sub> foto : varchar <sup>[11]</sup> pekerjaan_diselesaikan : int <sup>[11]</sup> <sub>[SEP]</sub> Created_at : timestamp <sup>[11]</sup> <sub>[SEP]</sub> update_at : timestamp <sup>[11]</sup> <sub>[SEP]</sub> accept_by_admin : int	+ Create () + Read () <sup>[11]</sup> <sub>[SEP]</sub> + Update () <sup>[11]</sup> <sub>[SEP]</sub> + Delete () <sup>[11]</sup> <sub>[SEP]</sub>

No	Nama kelas	Atribut	Method
2	Sertifikats	id : int user_id: int <sup>[1..1]</sup> <sub>{SEP}</sub> foto : varchar keterangan : varchar title : varchar <sup>[1..1]</sup> <sub>{SEP}</sub> tujuan_sertifikat : varchar <sup>[1..1]</sup> <sub>{SEP}</sub> Created_at : timestamp <sup>[1..1]</sup> <sub>{SEP}</sub> update_at : timestamp	+ Create () + Read () <sup>[1..1]</sup> <sub>{SEP}</sub> + Update () <sup>[1..1]</sup> <sub>{SEP}</sub> + Delete ()
3	Portfolios	id : int user_id : int foto : varchar <sup>[1..1]</sup> <sub>{SEP}</sub> keterangan : varchar Created_at : timestamp <sup>[1..1]</sup> <sub>{SEP}</sub> update_at : timestamp	+ Create () + Read () <sup>[1..1]</sup> <sub>{SEP}</sub> + Update () <sup>[1..1]</sup> <sub>{SEP}</sub> + Delete () <sup>[1..1]</sup> <sub>{SEP}</sub>
4	<i>Bookings</i>	id : int <sup>[1..1]</sup> <sub>{SEP}</sub> user_id : int <sup>[1..1]</sup> <sub>{SEP}</sub> montir_id : int <sup>[1..1]</sup> <sub>{SEP}</sub> jenis_kerusakan : text <sup>[1..1]</sup> <sub>{SEP}</sub> kronologi_kerusakan : text <sup>[1..1]</sup> <sub>{SEP}</sub> berapa_lama_kerusakan : text <sup>[1..1]</sup> <sub>{SEP}</sub> alamat_penjemputan : text <sup>[1..1]</sup> <sub>{SEP}</sub> keterangan : text <sup>[1..1]</sup> <sub>{SEP}</sub> status : varchar <sup>[1..1]</sup> <sub>{SEP}</sub> jadwal_penjemputan : date <sup>[1..1]</sup> <sub>{SEP}</sub> Created_at : timestamp <sup>[1..1]</sup> <sub>{SEP}</sub> update_at : timestamp	+ Create () + Read () <sup>[1..1]</sup> <sub>{SEP}</sub> + Update () <sup>[1..1]</sup> <sub>{SEP}</sub> + Delete ()

No	Nama kelas	Atribut	Method
5	<i>Reviews</i>	id : int user_id : int review : varchar montir_id : int Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()
6	<i>Complain</i>	id : int user_id : int complain : varchar Created_at : timestamp update_at : timestamp	+ Create () + Read () + Update () + Delete ()

Berdasarkan tabel 4.4 terdapat 10 *class* yang dibangun pada Rancang Bangun Sistem Informasi Montir Berbasis *Website* di Kecamatan Bangkinang Kota antara lain:

- a. *Users* merupakan *class* yang berisi data-data pengguna dengan berbagai *role*.
- b. *Portfolios* merupakan *class* yang menampung data-data portfolio montir.
- c. *Bookings* merupakan *class* yang menampung data-data *booking* pengguna.
- d. *Reviews* merupakan *class* yang menampung data-data tinjauan pengguna

- e. *Sertifikat* merupakan *class* yang menampung data-data sertifikat montir.
- f. *Complain* merupakan *class* yang menampung data-data *Complain* montir.

#### **D. Perancangan *Database***

Dalam sistem yang dibangun terdapat beberapa bisnis proses yang ingin diselesaikan adalah Rancang Bangun Sistem Informasi Montir Berbasis *Website* di Kecamatan Bangkinang Kota. Pada perancangan sistem ini terdapat 10 tabel.

##### 1. Merancang Relasi Antar Tabel

Pada tahapan ini tampilan data dan relasi tabel pada sistem informasi montir berbasis *website* di Kecamatan Bangkinang Kota yang dibuat dengan menyesuaikan antara bisnis proses dan sistem yang akan dibangun, maka dapat dibentuk suatu *database* yang nantinya akan digunakan untuk menyimpan keseluruhan proses yang diperlukan.

##### 2. *Database* dan Struktur Tabel

Nama *database* yang digunakan pada sistem ini adalah montir. Montir memiliki tabel yang saling berelasi untuk mendukung efisiensi pada setiap kata hubungannya daftar tabel yang ada pada sistem. Berikut tabel-tabel yang ada didalam *database* montir.

a. Tabel *Users*

Tabel *users* digunakan untuk menampung semua data-data pengguna sistem yang dapat dilihat pada tabel 4.5.

Tabel 4.5 *Users*

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>Name</i>	<i>Varchar(60)</i>	Nama pengguna
3	<i>Email</i>	<i>Varchar(60)</i>	<i>Email</i> pengguna
4	<i>Email_verified_at</i>	<i>Timestamp</i>	<i>Email</i> verifikasi
5	<i>Password</i>	<i>Varchar(60)</i>	<i>Password</i> pengguna
6	<i>Remember_token</i>	<i>Varchar(60)</i>	Autentikasi pengguna
7	No_hp	<i>Varchar(60)</i>	Nomor telepon pengguna
8	Alamat	<i>Text</i>	Alamat pengguna
9	<i>Role</i>	<i>Varchar(60)</i>	<i>Role</i> pengguna
10	Jenis_kelamin	<i>Varchar(60)</i>	Jenis kelamin pengguna
11	Lingkup_wilayah	<i>Varchar(60)</i>	Lingkup wilayah pengguna
12	Category_id	<i>Integer</i>	Menampilkan data-data kategori
13	Pengalaman	<i>Varchar(60)</i>	Pengalaman pengguna
14	Tentang	<i>Varchar(60)</i>	Tentang pengguna
15	Pekerjaan	<i>Varchar(60)</i>	Pekerjaan pengguna
16	Nama_bengkel	<i>Varchar(60)</i>	Nama bengkel pengguna
17	Foto	<i>Varchar(60)</i>	Foto pengguna
18	<i>Accept_by_admin</i>	<i>Timestamp</i>	Verifikasi oleh admin
19	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
20	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

b. Tabel *Portfolio*

Tabel *portfolio* merupakan *class* yang menampung data-data *portfolio* montir dapat dilihat pada tabel 4.6.

**Tabel 4. 6 *Portfolio***

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	User_id	<i>Integer</i>	Id pengguna
3	Foto	<i>Varchar(60)</i>	Foto pengguna
4	Keterangan	<i>Varchar(60)</i>	Keterangan pengguna
5	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
6	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

c. Tabel *Booking*

Tabel *booking* merupakan *class* yang menampung data-data *booking* pengguna dapat dilihat pada tabel 4.7.

**Tabel 4. 7 *Booking***

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	User_id	<i>Integer</i>	Id user
3	Montir_id	<i>Integer</i>	Id montir
4	Jenis_kerusakan	<i>Text</i>	Deskripsi kategori mobil
5	Berapa_lama_kerusakan	<i>Text</i>	Lama kerusakan
6	Alamat_penjemputan	<i>Text</i>	Alamat penjemputan kendaraan
7	Keterangan	<i>Text</i>	Keterangan kendaraan
8	Status	<i>Varchar(60)</i>	Status
9	Jadwal_penjemputan	<i>Date</i>	Jadwal penjemputan kendaraan

No	Nama kolom	Tipe data	Keterangan
10	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
11	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

d. Tabel *Review*

Tabel *review* merupakan *class* yang menampung data-data tinjauan pengguna dapat dilihat pada tabel 4.8.

Tabel 4. 8 *Review*

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>User_id</i>	<i>Integer</i>	Id pengguna
3	<i>Reviews</i>	<i>Text</i>	Tinjauan pengguna
5	<i>Montir_id</i>	<i>Integer</i>	Id montir
6	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
5	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

e. Tabel *Sertifikat*

Tabel *sertifikat* merupakan *class* yang menampung data-data sertifikat montir dapat dilihat pada tabel 4.9.

Tabel 4. 9 *Sertifikat*

No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	<i>User_id</i>	<i>Integer</i>	Id pengguna
3	Foto	<i>Varchar(60)</i>	Foto pengguna
4	Keterangan	<i>Varchar(60)</i>	Keterangan sertifikat

No	Nama kolom	Tipe data	Keterangan
6	Title	<i>Varchar(60)</i>	Judul sertifikat
7	Tujuan_sertifikats	<i>Integer</i>	Tujuan sertifikat
10	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
11	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

f. Tabel *Complaint*

Tabel *Complaint* merupakan *class* yang menampung data-data *Complaint* pengguna dapat dilihat pada tabel 4.10.

**Tabel 4. 10 *Complaint***

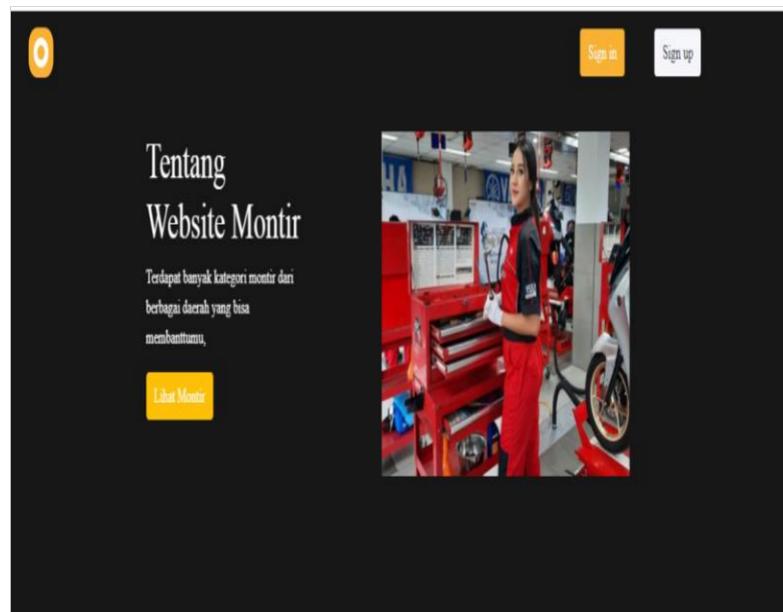
No	Nama kolom	Tipe data	Keterangan
1	<i>Id</i>	<i>Integer</i>	<i>Primary key</i>
2	User_id	<i>Integer</i>	Id pengguna
3	<i>Complaint</i>	<i>Text</i>	<i>Complaint</i> pengguna
5	Montir_id	<i>Integer</i>	Id montir
6	<i>Created_at</i>	<i>Timestamp</i>	Tanggal pengguna mendaftar
5	<i>Updated_at</i>	<i>Timestamp</i>	Tanggal pengguna melakukan <i>update profile</i>

**E. Implementasi *User Interfaces***

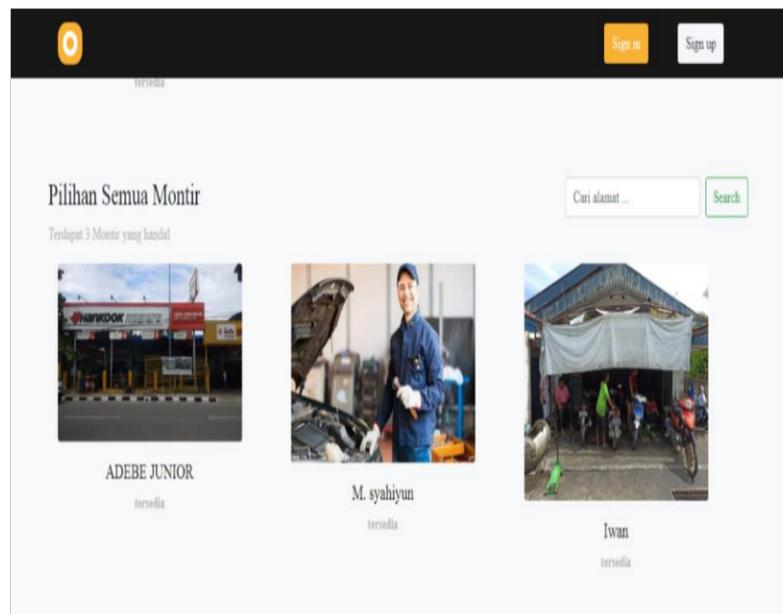
Implementasi adalah tahap penerapan sekaligus pengujian bagi sistem berdasarkan hasil analisa dan perancangan sistem, pada tahap ini merupakan tahap implementasi hasil menjadi sebuah sistem informasi montir berbasis *website* di Kecamatan Bangkinang Kota.

## 1. Halaman *Home*

Halaman *home* merupakan halaman awal dari *website* montir. Berikut tampilan halaman *home* pada gambar 4.3 yang menampilkan sebuah *background* foto montir dan *list* semua montir pada halaman *home* beserta fitur pencarian montir, berikut tampilan pada gambar 4.4.



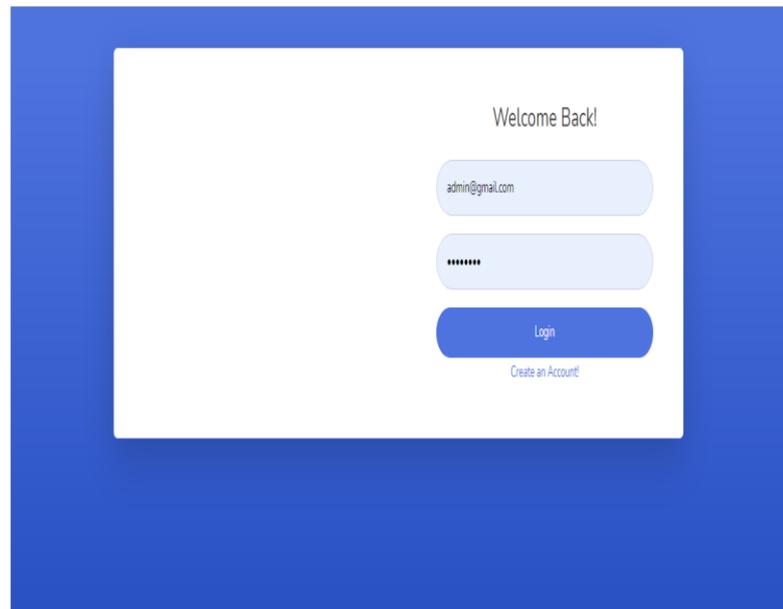
Gambar 4. 3 Halaman *Home*



**Gambar 4. 4 Halaman *Home List Semua Montir Dan Fitur Pencarian Montir***

## 2. Halaman *Login*

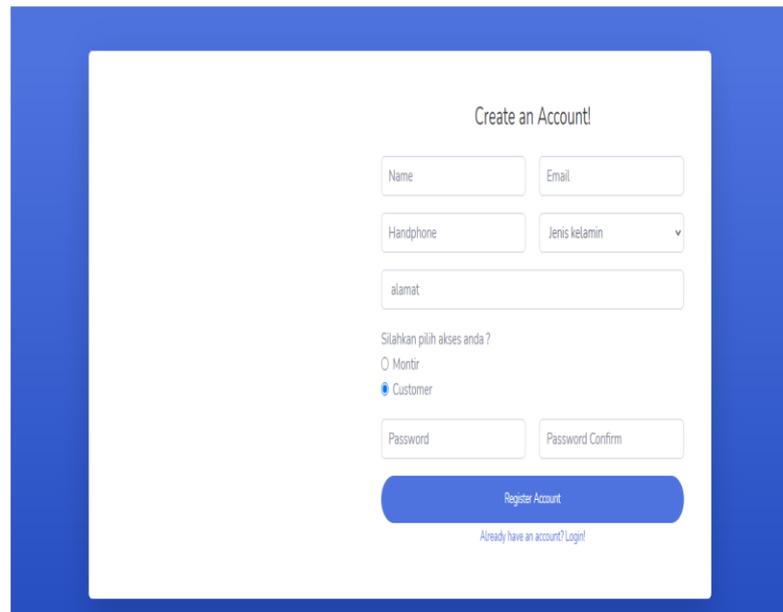
*Login* merupakan tampilan untuk melanjutkan penggunaan sistem seperti mengakses *dashboard* ataupun melakukan *bookings* montir, pada saat melakukan *login username* dan *password* yang dimasukan harus sesuai dengan *value* yang ada pada *database*, berikut tampilan halaman *login*, seperti gambar 4.5.



**Gambar 4. 5 Halaman *Login***

3. Halaman *Register Costumer* dan Montir

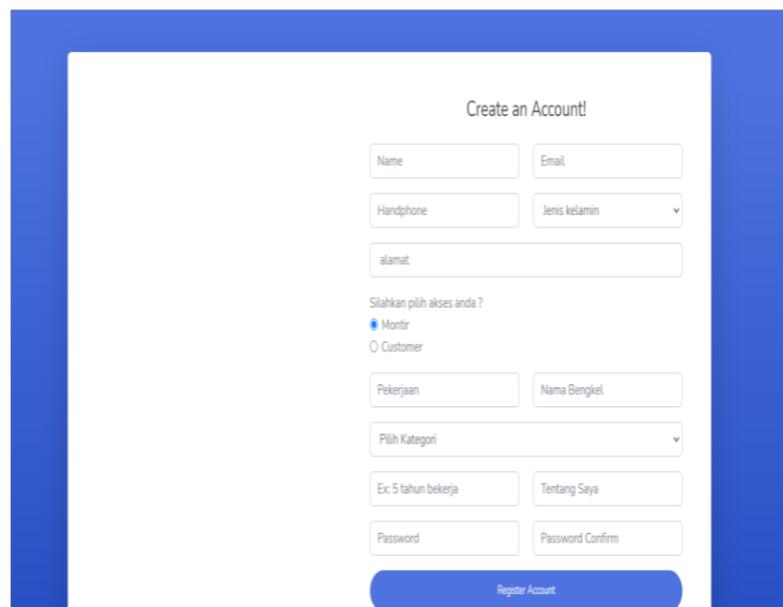
Merupakan halaman untuk melakukan pendaftaran *website*. Pada *form registrasi* harus mengisi semua *form registrasi* yang tersedia sesuai dengan data pribadi, pada saat melakukan pendaftaran pengunjung *website* dapat memilih hak akses seperti montir ataupun *costumer*, berikut tampilan halaman *register costumer* seperti gambar 4.6 dan halaman *register montir* pada halaman 4.7.



The screenshot shows a registration form titled "Create an Account!". The form includes the following fields and options:

- Name (text input)
- Email (text input)
- Handphone (text input)
- Jenis kelamin (dropdown menu)
- alamat (text input)
- Silahkan pilih akses anda? (radio buttons):
  - Montir
  - Customer
- Password (text input)
- Password Confirm (text input)
- Register Account (blue button)
- Already have an account? Login! (text link)

**Gambar 4. 6 Halaman *Register Costumer***



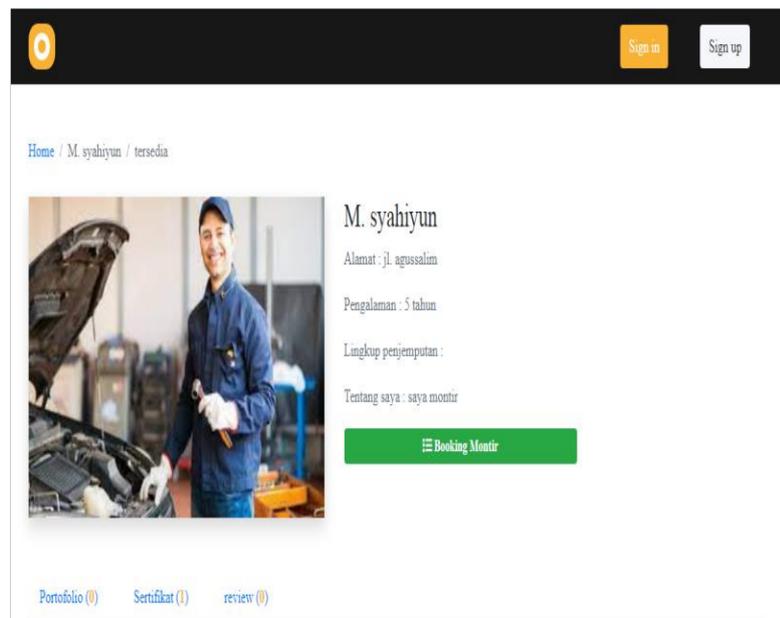
The screenshot shows a registration form titled "Create an Account!". The form includes the following fields and options:

- Name (text input)
- Email (text input)
- Handphone (text input)
- Jenis kelamin (dropdown menu)
- alamat (text input)
- Silahkan pilih akses anda? (radio buttons):
  - Montir
  - Customer
- Pekerjaan (text input)
- Nama Bengkel (text input)
- Pilih Kategori (dropdown menu)
- Ex: 5 tahun bekerja (text input)
- Tentang Saya (text input)
- Password (text input)
- Password Confirm (text input)
- Register Account (blue button)

**Gambar 4. 7 Halaman *Register Montir***

#### 4. Halaman *Booking*

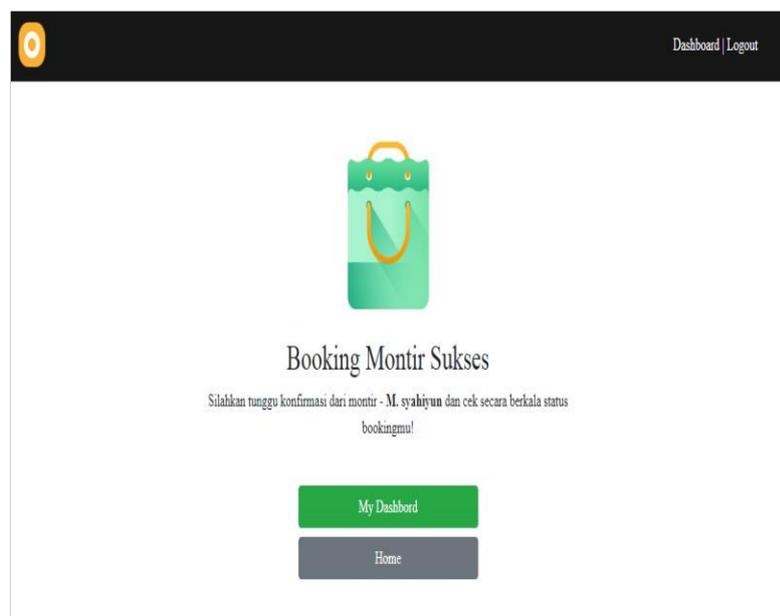
Pada halaman ini akan ditampilkan nama montir, alamat, pengalaman dan lingkup penjemputan seperti gambar 4.8.



Gambar 4. 8 Halaman *Booking*

#### 5. Halaman Sukses *Booking*

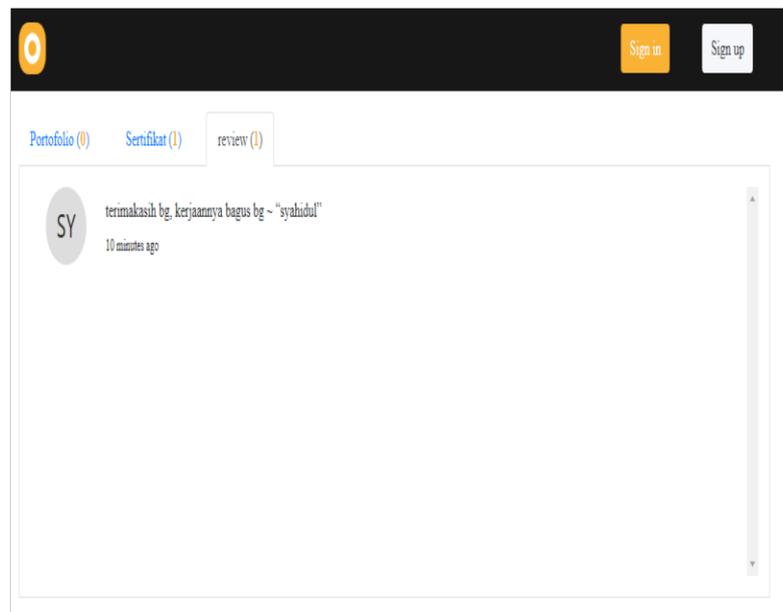
Pada halaman ini akan ditampilkan *illustration* bahwa permintaan *booking* telah dikirimkan kepada montir seperti gambar 4.9.



Gambar 4. 9 Halaman Sukses *Booking*

## 6. Halaman Detail Montir

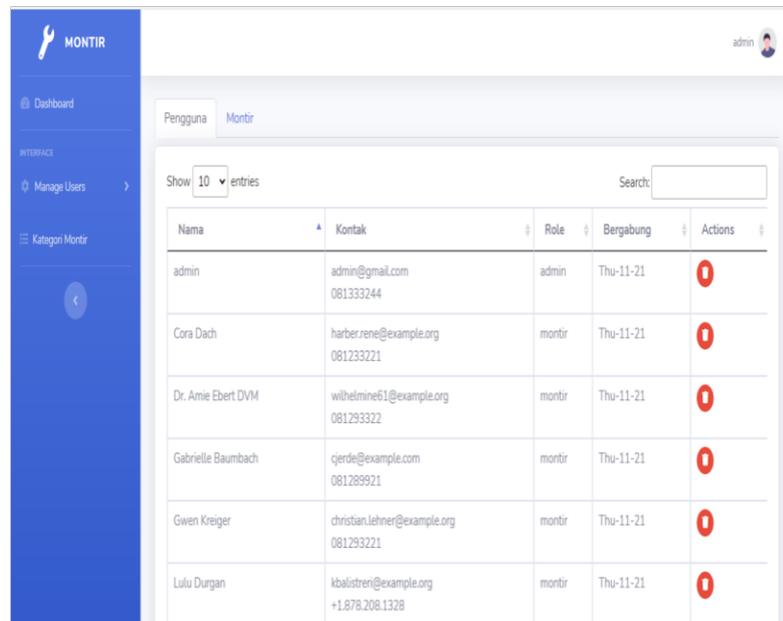
Pada halaman ini tersedia *Section portfolio* , *sertifikats* dan *review* seperti gambar 4.10.



**Gambar 4. 10 Halaman Detail Montir**

## 7. Halaman *Dashboard* Admin

Pada halaman *dashboard* admin ditampilkan *menu-menu* atau informasi yang dikhususkan untuk *role* admin, mengelola seluruh pengguna dan menerima pendaftaran montir seperti gambar 4.11.

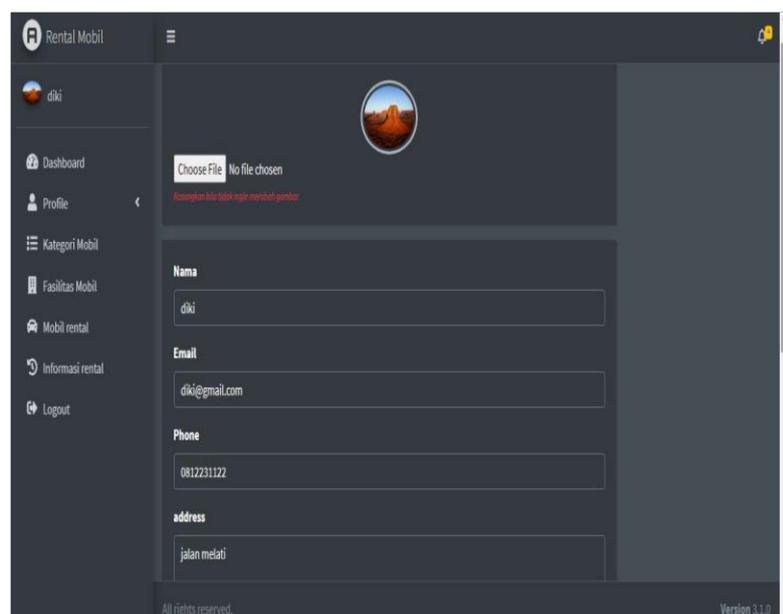


Nama	Kontak	Role	Bergabung	Actions
admin	admin@gmail.com 081333244	admin	Thu-11-21	
Cora Dach	harber.rene@example.org 081233221	montrir	Thu-11-21	
Dr. Amie Ebert DVM	wilhelmine61@example.org 081293322	montrir	Thu-11-21	
Gabrielle Baumbach	cjerde@example.com 081289921	montrir	Thu-11-21	
Gwen Kreiger	christian.lehner@example.org 081293221	montrir	Thu-11-21	
Lulu Durgan	kbalistreni@example.org +1.878.208.1328	montrir	Thu-11-21	

**Gambar 4. 11 Halaman *Dashboard* Admin**

a. Halaman Pengguna Dinonaktifkan Pada *Dashboard* Admin

Pada tampilan ini menampilkan admin dapat kembali mengaktifkan pengguna yang telah dinonaktifkan seperti gambar 4.12.



Rental Mobil

diki

Choose File No file chosen

Responen file tidak dapat membaca gambar.

Nama  
diki

Email  
diki@gmail.com

Phone  
0812231122

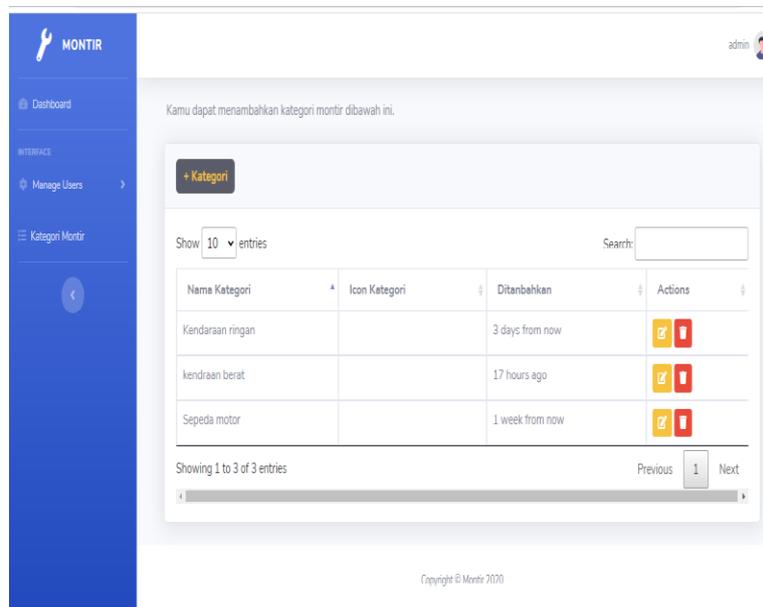
address  
jalan melati

All rights reserved. Version 3.1.0

**Gambar 4. 12 Halaman Pengguna Dinonaktifkan Pada *Dashboard* Admin**

b. Halaman Kategori *Dashboard* Admin

Pada halaman ini menampilkan beberapa kategori seperti kendaraan berat, sepeda motor, dan kendaraan ringan admin dapat menghapus dan menambah kategori seperti gambar 4.13.



**Gambar 4. 13 Halaman Kategori *Dashboard* Admin**

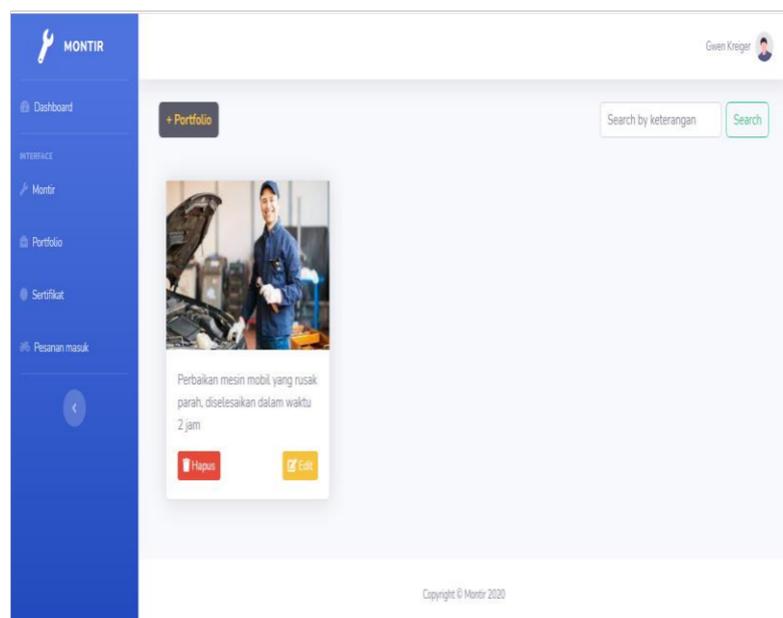
8. Halaman *Dashboard* Montir

Pada halaman *dashboard* montir ditampilkan *menu-menu* atau informasi yang dikhususkan untuk *role* montir seperti gambar 4.14.

**Gambar 4. 14 Halaman *Dashboard Montir***

a. Halaman *Dashboard Montir Kelola Portfolio*

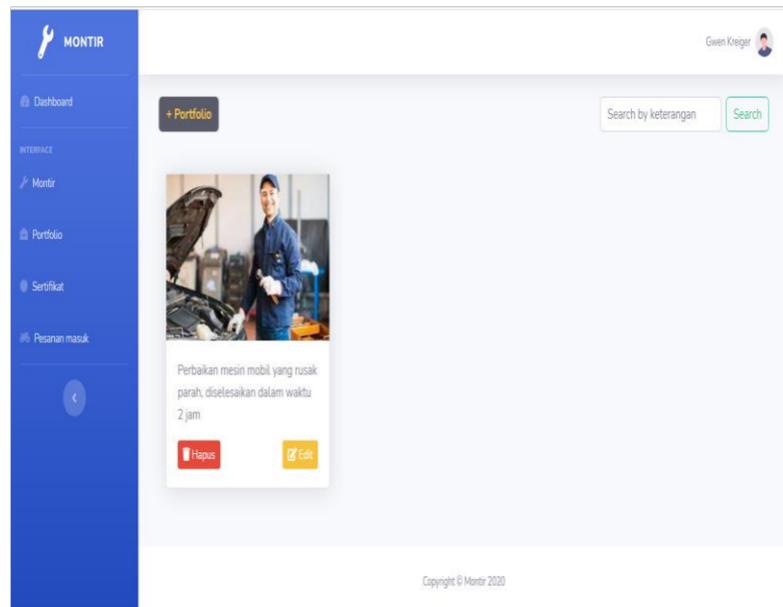
Pada halaman ini menampilkan, montir dapat mengelola portfolio seperti menghapus dan menambah portfolio seperti gambar 4.1.



**Gambar 4. 15 Halaman *Dashboard Montir Kelola Portfolio***

b. Halaman *Dashboard* Montir Kelola Sertifikat

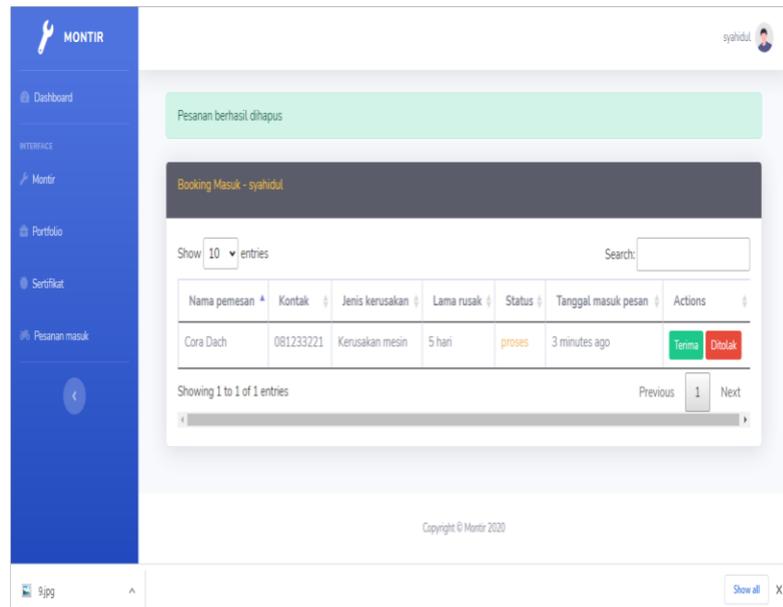
Pada halaman ini montir dapat mengelola sertifikat seperti menambah dan menghapus sertifikat seperti gambar 4.16.



**Gambar 4. 16 Halaman *Dashboard* Montir Kelola Sertifikat**

c. Halaman *Dashboard* Montir Kelola Pesan Masuk

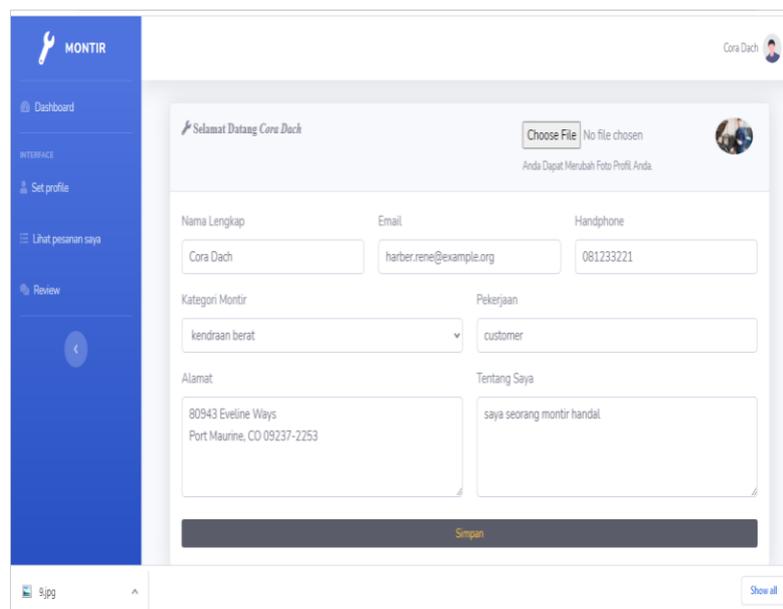
Pada halaman ini montir dapat melihat semua pesan masuk, montir dapat menerima dan menolak pesanan seperti gambar 4.17



Gambar 4. 17 Halaman *Dashboard Montir Kelola Pesan Masuk*

## 9. Halaman *Dashboard Costumer*

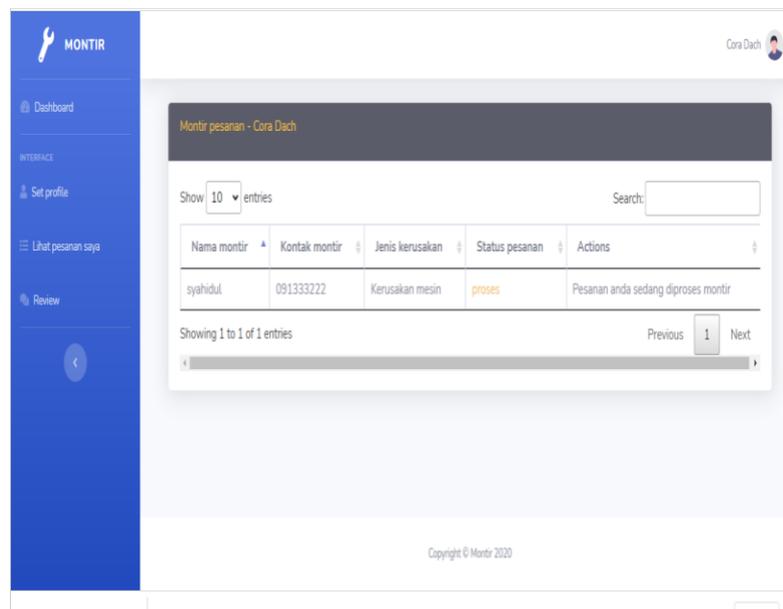
Pada halaman *dashboard customer* ditampilkan *menu-menu* atau informasi yang dikhususkan untuk *role costumer* seperti gambar 4.18.



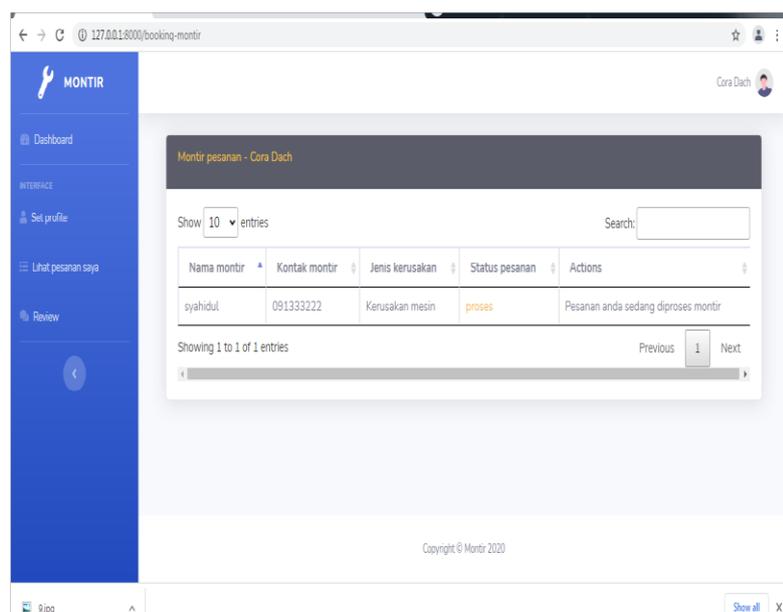
Gambar 4. 18 Halaman *Dashboard Costumer*

a. Halaman *Dashboard Costumer* Lihat Status *Booking*

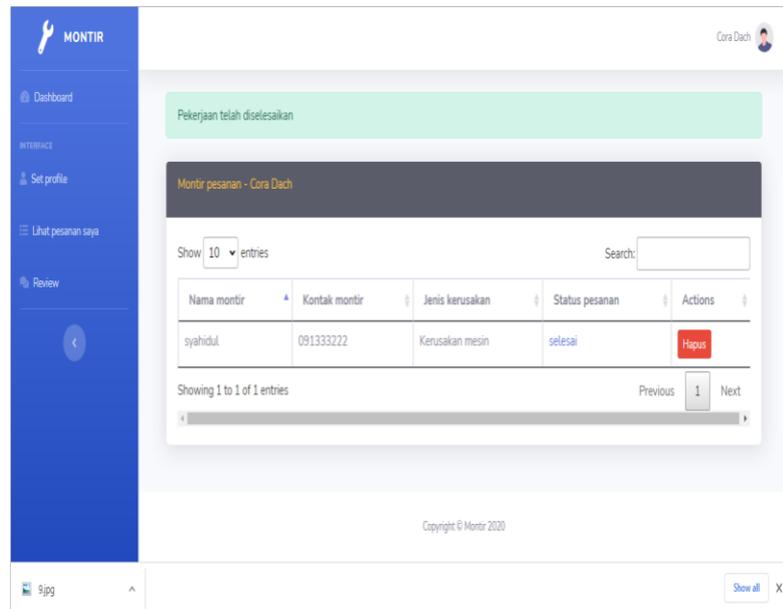
Pada halaman *dashboard customer*, akan ada 3 status yaitu proses, diterima dan *customer* dapat menekan tombol pekerjaan diselesaikan jika montir telah menyelesaikan pekerjaannya seperti gambar 4.19, 4.20, dan 4.21.



**Gambar 4. 19** Halaman *Dashboard Costumer* Lihat Status *Booking*



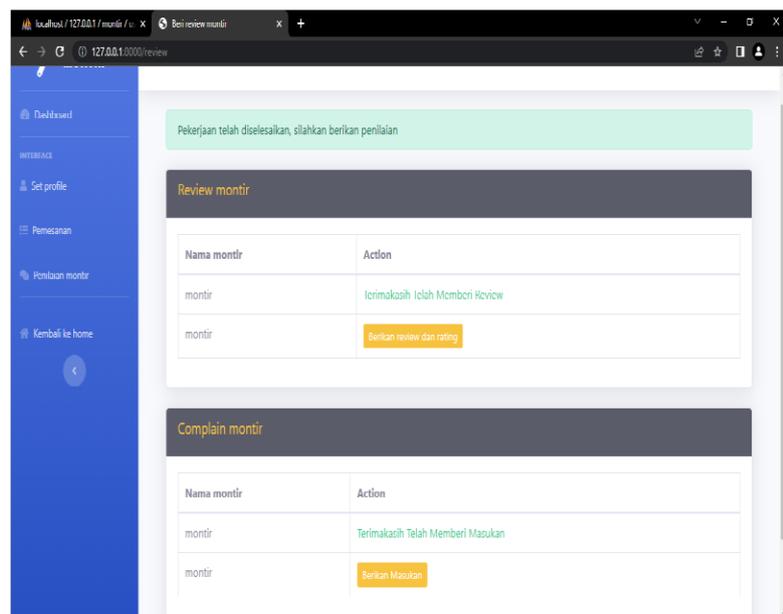
**Gambar 4. 20** Halaman *Diterima Costumer*



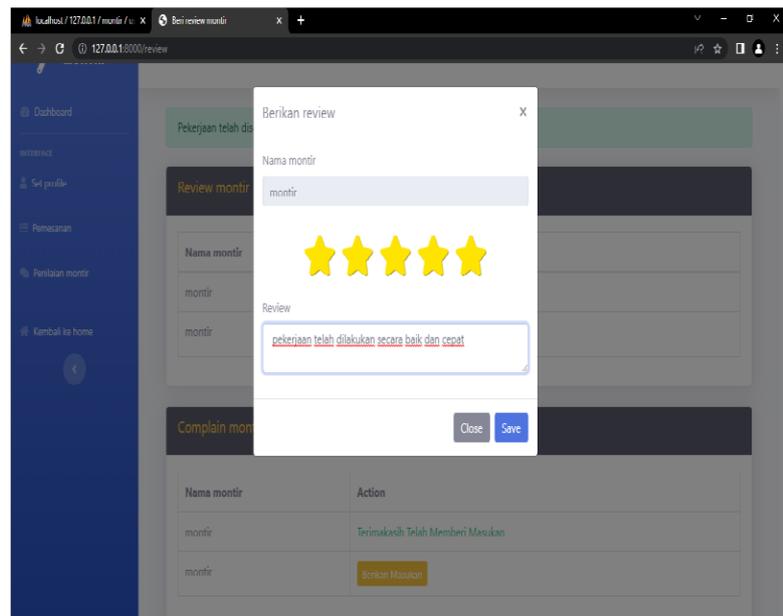
**Gambar 4. 21 Halaman Pekerjaan Telah Diselesaikan**

b. Halaman *Dashboard Costumer Review*

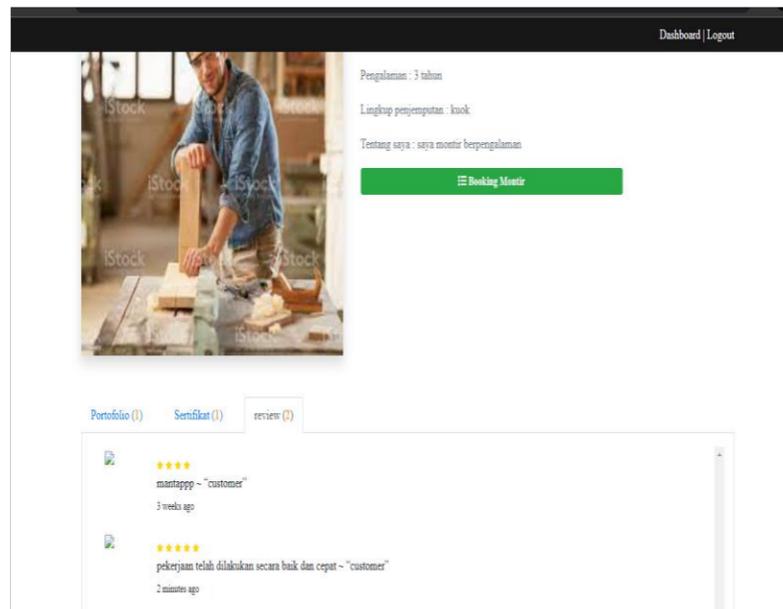
Halaman ini menampilkan *costumer* memberikan *review* setelah *costumer* menekan tombol pekerjaan telah diselesaikan seperti gambar 4.22.



**Gambar 4. 22 Halaman Dashboard Costumer Review**



Gambar 4. 23 Halaman Berikan *Review*



Gambar 4. 24 Halaman Semua *Review*

## **BAB V**

### **PENUTUP**

#### **A. Kesimpulan**

Berdasarkan hasil yang didapatkan dari serangkaian proses penelitian mengenai Rancang Bangun Sistem Informasi Montir Berbasis *Website* di Bangkinang Kota, maka dapat disimpulkan bahwa :

1. Sistem informasi ini dapat menyajikan informasi cepat dan akurat yang dibutuhkan oleh pelanggan.
2. Dengan dibuatnya sistem informasi montir berbasis *website* ini dapat membantu dan mempermudah masyarakat khususnya pengguna kendaraan motor dalam melakukan kegiatan servis secara rutin dengan memanggil montir ke tempat lokasi pelanggan tersebut berada.

#### **B. Saran**

Saran untuk penelitian selanjutnya adalah sebagai berikut :

1. Bisa digunakan di berbagai daerah khususnya di seluruh Indonesia tidak hanya di daerah Bangkinang Kota saja.
2. Perlu ditambahkan fitur pemetaan supaya *costumer* bisa memberikan titik lokasi, dan montir dengan cepat bisa menuju ke lokasi.

## DAFTAR PUSTAKA

- Abdulloh, R. (2015). *Web Programming is Easy*. Jakarta: PT. Elex Media Komputindo, 7.
- Afifah, K. N. (2018). *Perancangan Sistem Informasi Persediaan Barang Berbasis Web Pada Pt. Mas Sumbiri*. Universitas Islam Sultan Agung.
- Amin, M. (2016). Analisis Penjualan Komputer Pada Putra Jawa Computer Berbasis Web. *Technologia: Jurnal Ilmiah*, 7(4).
- Anggraeni, E. Y., & Irviani, R. (2017). Pengantar Sistem Informasi, Yogyakarta, CV. Andi Offset.
- Arizona, N. D. (2017). *Aplikasi Pengolahan Data Anggaran Pendapatan dan Belanja Desa (APBDES) Pada Kantor Desa Bakau Kecamatan Jawai Berbasis Web*. *Cybernetics*, Vol. 01 (No. 02), 105–119.
- Awaludin, R. (2016). *Seminggu Belajar Laravel 5.2*. Bandung: LeanPublishing.
- Fanani, M. F., Kharisma, A. P., & Wardhono, W. S. (2020). *Pengembangan Aplikasi Mobile Learning Bahasa Jawa (JawApp) berbasis Android dengan Metode Prototyping*. 4(6), 1767–1774.
- Gama, M. (2011). *Universitas Komputer Indonesia*.
- Guntur, W., & Eko, S. W. (2015). Perancangan Website Sebagai Media Informasi Dan Promosi Batik Khas Kabupaten Kulonprogo. *AMIK BSI Yogyakarta, Yogyakarta*.
- Hidayatullah, P., & Kawistara, J. K. (2015). Pemrograman Web: Web sistem Informasi Akademik. *Yayasan Ahmar Cendekia Indonesia: Bandung*.
- Kadir, A., & Triwahyuni, T. C. (2013). Pengantar Teknologi Informasi Edisi Revisi, Yogyakarta: CV. Andi Offset.
- Madcoms. (2016). 2) 1,2. 2(2), 12–26.
- Riau, B. (2015). *Jumlah Kendaraan Bermotor di Provinsi Riau Menurut Kabupaten/Kota dan Jenis Kendaraan (unit), 2015*. BPS Riau. <https://riau.bps.go.id/statictable/2020/11/02/370/jumlah-kendaraan-bermotor-di-provinsi-riau-menurut-kabupaten-kota-dan-jenis-kendaraan-unit-2015.html>
- Ripai, I. (2017). Rancang Bangun Media Pembelajaran Menggunakan Android Untuk Matakuliah Pemrograman Internet Menggunakan Magazine App Maker. *Ict Learning*, 3(1), 1–6.
- Risdiansyah, D. (2017). Perancangan Sistem Informasi Bimbingan Konseling Berbasis Desktop pada SMA Kemala Bhayangkari 1 Kubu Raya. *Jurnal*

*Khatulistiwa Informatika*, 5(2).

- Rosa, A. S., & Shalauddin, M. (2011). *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*.
- Rozaq, A., Lestari, K. F., & Handayani, S. (2015). Sistem informasi produk dan data calon jamaah haji dan umroh pada Pt. Travellindo Lusiyana Banjarmasin berbasis web. *Jurnal Positif*, 1(1), 1–13.
- Saputra, R., Widodo, A. W., & Brata, A. H. (2018). *Pengembangan Sistem Rental Kamera Online*. 2(6), 2221–2226.
- Sarwono, J., & Prihartono, K. (2012). *Perdagangan online: cara bisnis di internet*. Jakarta: PT Elex Media Komputindo.
- Sibero, A. F. K. (2011). *Kitab suci web programming*. Yogyakarta: Mediakom.
- Sulianta, F. (2019). *Teknik Perancangan Arsitektur Sistem Informasi*.
- Trimahardhika, R., & Sutinah, E. (2017). Penggunaan Metode Rapid Application Development Dalam Perancangan Sistem Informasi Perpustakaan. *Jurnal Informatika*, 4(2).
- Yakub, H. (2015). *Aplikasi Penjualan Sparepart Mobil Menggunakan Code Igniter*. 08, 35–40.
- Abdulloh, R. (2015). *Web Programming is Easy*. Jakarta: PT. Elex Media Komputindo, 7.
- Afifah, K. N. (2018). *Perancangan Sistem Informasi Persediaan Barang Berbasis Web Pada Pt. Mas Sumbiri*. Universitas Islam Sultan Agung.
- Amin, M. (2016). Analisis Penjualan Komputer Pada Putra Jawa Computer Berbasis Web. *Technologia: Jurnal Ilmiah*, 7(4).
- Anggraeni, E. Y., & Irviani, R. (2017). *Pengantar Sistem Informasi*, Yogyakarta, CV. Andi Offset.
- Arizona, N. D. (2017). *Aplikasi Pengolahan Data Anggaran Pendapatan dan Belanja Desa (APBDES) Pada Kantor Desa Bakau Kecamatan Jawai Berbasis Web*. *Cybernetics*, Vol. 01 (No. 02), 105–119.
- Awaludin, R. (2016). *Seminggu Belajar Laravel 5.2*. Bandung: LeanPublishing.
- Fanani, M. F., Kharisma, A. P., & Wardhono, W. S. (2020). *Pengembangan Aplikasi Mobile Learning Bahasa Jawa (JawApp) berbasis Android dengan Metode Prototyping*. 4(6), 1767–1774.
- Gama, M. (2011). *Universitas Komputer Indonesia*.
- Guntur, W., & Eko, S. W. (2015). *Perancangan Website Sebagai Media Informasi*

- Dan Promosi Batik Khas Kabupaten Kulonprogo. *AMIK BSI Yogyakarta, Yogyakarta*.
- Hidayatullah, P., & Kawistara, J. K. (2015). Pemrograman Web: Web sistem Informasi Akademik. *Yayasan Ahmar Cendekia Indonesia: Bandung*.
- Kadir, A., & Triwahyuni, T. C. (2013). Pengantar Teknologi Informasi Edisi Revisi, Yogyakarta: CV. *Andi Offset*.
- Madcoms. (2016). 2) 1,2. 2(2), 12–26.
- Riau, B. (2015). *Jumlah Kendaraan Bermotor di Provinsi Riau Menurut Kabupaten/Kota dan Jenis Kendaraan (unit), 2015*. BPS Riau. <https://riau.bps.go.id/statictable/2020/11/02/370/jumlah-kendaraan-bermotor-di-provinsi-riau-menurut-kabupaten-kota-dan-jenis-kendaraan-unit-2015.html>
- Ripai, I. (2017). Rancang Bangun Media Pembelajaran Menggunakan Android Untuk Matakuliah Pemrograman Internet Menggunakan Magazine App Maker. *Ict Learning*, 3(1), 1–6.
- Risdiansyah, D. (2017). Perancangan Sistem Informasi Bimbingan Konseling Berbasis Desktop pada SMA Kemala Bhayangkari 1 Kubu Raya. *Jurnal Khatulistiwa Informatika*, 5(2).
- Rosa, A. S., & Shalauddin, M. (2011). *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*.
- Rozaq, A., Lestari, K. F., & Handayani, S. (2015). Sistem informasi produk dan data calon jamaah haji dan umroh pada Pt. Travellindo Lusiyanan Banjarmasin berbasis web. *Jurnal Positif*, 1(1), 1–13.
- Saputra, R., Widodo, A. W., & Brata, A. H. (2018). *Pengembangan Sistem Rental Kamera Online*. 2(6), 2221–2226.
- Sarwono, J., & Prihartono, K. (2012). Perdagangan online: cara bisnis di internet. *Jakarta: PT Elex Media Komputindo*.
- Sibero, A. F. K. (2011). *Kitab suci web programming*. Yogyakarta: *Mediakom*.
- Sulianta, F. (2019). *Teknik Perancangan Arsitektur Sistem Informasi*.
- Trimahardhika, R., & Sutinah, E. (2017). Penggunaan Metode Rapid Application Development Dalam Perancangan Sistem Informasi Perpustakaan. *Jurnal Informatika*, 4(2).
- Yakub, H. (2015). *Aplikasi Penjualan Sparepart Mobil Menggunakan Code Igniter*. 08, 35–40.