

**Rancang Bangun Aplikasi Sistem Penjadwalan dan Permintaan  
Pemeliharaan Mesin Digital Printing Outdoor Berbasis Website  
(Studi Kasus: CV. Warna Makassar)**

**Tugas Akhir**

**diajukan untuk memenuhi salah satu syarat**

**memperoleh gelar sarjana**

**dari Program Studi Rekayasa Perangkat Lunak  
(Kampus Kota Surabaya)**

**Fakultas Informatika**

**Universitas Telkom**

**NIM 1201202002**

**David Win Syarif Hidayat**



**Program Studi Sarjana Rekayasa Perangkat Lunak  
(Kampus Kota Surabaya)**

**Fakultas Informatika**

**Universitas Telkom Surabaya**

**2024**

**LEMBAR PENGESAHAN**

**Rancang Bangun Aplikasi Sistem Penjadwalan dan Permintaan Pemeliharaan  
Mesin Digital Printing Outdoor Berbasis Website  
(Studi Kasus: CV. Warna Makassar)**

*Design of a Website-Based Outdoor Digital Printing Machine  
Maintenance Request and Scheduling System Application  
(Case Study: CV. Warna Makassar)*

**NIM : 1201202002**

**David Win Syarif Hidayat**

Tugas akhir ini telah diterima dan disahkan untuk memenuhi sebagian syarat memperoleh gelar pada Program Studi Sarjana Rekayasa Perangkat Lunak (Kampus Kota Surabaya)

Fakultas Informatika

Universitas Telkom

Surabaya, 15 Agustus 2024

Menyetujui

Pembimbing I,



Dahliar Ananda, S.T., M.T

NIP.09820030

Pembimbing II,



Arni Muarifah Amri, S.T., M.T.

NIP.23900007

Ketua Program Studi  
Perangkat Lunak (Kampus  
Kota Surabaya),



Fidi Wincoko Putro, S.ST., M.Kom  
NIP: 19870004

**LEMBAR PERNYATAAN**

Dengan ini saya, David Win Syarif Hidayat, menyatakan sesungguhnya bahwa Tugas Akhir saya dengan judul **Rancang Bangun Aplikasi Sistem Penjadwalan dan Permintaan Pemeliharaan Mesin Digital Printing Outdoor Berbasis Website (Studi Kasus: CV. Warna Makassar)** beserta dengan seluruh isinya adalah merupakan hasil karya sendiri, dan saya tidak melakukan penjiplakan yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan. Saya siap menanggung resiko/sanksi yang diberikan jika di kemudian hari ditemukan pelanggaran terhadap etika keilmuan dalam buku TA atau jika ada klaim dari pihak lain terhadap keaslian karya,

Surabaya, 15 Agustus 2024

Yang Menyatakan



David Win Syarif Hidayat

## Rancang Bangun Aplikasi Sistem Penjadwalan dan Permintaan Pemeliharaan Mesin Digital Printing Outdoor Berbasis Website (Studi Kasus: CV. Warna Makassar)

David Win Syarif Hidayat<sup>1</sup>, Dahliar Ananda, S.T, M.T<sup>2</sup>, Arni Muarifah Anri, S.T, M.T<sup>3</sup>

<sup>1,2,3</sup>Fakultas Informatika, Universitas Telkom, Surabaya

<sup>4</sup>Divisi Digital Service PT Telekomunikasi Indonesia

<sup>1</sup>davititts@students.telkomuniversity.ac.id, <sup>2</sup>ananda@telkomuniversity.ac.id,

<sup>3</sup>arnyrvivah@telkomuniversity.ac.id

---

### Abstrak

CV. Warna Makassar suatu perusahaan *supplier* digital printing dan mesin *digital printing outdoor* yang berada di kota Makassar, Sulawesi Selatan. Mesin digital printing outdoor menjadi mesin utama bagi beberapa *customer* CV. Warna Makassar. Pemeliharaan suatu mesin *digital printing outdoor* menjadi salah satu faktor penting untuk menjaga kualitas mesin. Terjadinya peningkatan permintaan untuk perawatan mesin digital printing outdoor *customer* sering terhambat sehingga tidak tersalurkan dengan baik kepada teknisi. Hal tersebut dikarenakan komunikasi yang masih menggunakan telepon atau aplikasi *whatsapp*. Identifikasi permasalahan pada mesin *digital printing* juga masih kurang jelas dari *customer*. Permasalahan lain adalah proses transaksi barang atau bahan digital printing yang terus meningkat dari *customer*. Dengan adanya sistem aplikasi berbasis website, memudahkan antara *customer* dengan *supplier* dalam proses transaksi permintaan perbaikan pada mesin digital printing outdoor. Teknisi bisa melakukan monitor transaksi pemeliharaan mesin dari *customer*. Teknisi juga dapat mengidentifikasi bagian mesin yang diperlukan untuk melakukan pemeliharaan mesin tersebut. Aplikasi dikembangkan menggunakan metode waterfall. Teknologi yang digunakan yaitu framework Laravel, Vuejs dan MySQL. Hasil penelitian ini bisa mengimplementasikan aplikasi dalam mengatasi permasalahan tersebut menggunakan metode dan teknologi framework yang digunakan. Serta pada *testing* yang menggunakan *Black Box Testing* memiliki hasil yang sesuai dan *test case scenario* yang memiliki hasil rata-rata *success*.

Kata kunci : CV.Warna Makassar, Digital Printing, Mesin *Digital Printing Outdoor*, Pemeliharaan, Permintaan

---

### Abstract

CV. Warna Makassar is a digital printing and outdoor digital printing machine supplier company located in the city of Makassar, South Sulawesi. Outdoor digital printing machines are the main machines for several CV. Warna Makassar customers. Maintenance of an outdoor digital printing machine is one of the important factors to maintain machine quality. The increase in requests for maintenance of customer outdoor digital printing machines is often hampered so that it is not properly channeled to technicians. This is due to communication that still uses telephone or whatsapp applications. Identification of problems with digital printing machines is also still unclear from customers. Another problem is the transaction process of digital printing goods or materials that continue to increase from customers. With a website-based application system, it makes it easier between customers and suppliers in the transaction process for repair requests on outdoor digital printing machines. Technicians can monitor machine maintenance transactions from customers. Technicians can also identify machine parts needed to maintain the machine. The application was developed using the waterfall method. The technology used is the Laravel framework, Vuejs and MySQL. The results of this study can implement applications in overcoming these problems using the methods and framework technology used. As well as in testing using Black Box Testing has appropriate results and test case scenarios that have average success results.

Keywords: CV.Warna Makassar, Digital Printing, Outdoor Digital Printing Machine, Maintenance, Requests

---

## 1. Pendahuluan

### Latar Belakang

*Digital Printing* merupakan metode percetakan modern yang masih sering digunakan lingkungan sekitar hingga sampai saat ini yaitu percetakan baliho, spanduk dan lain-lain. Pada *digital printing* hal yang paling sering digunakan pada kegiatan tersebut yaitu mesin digital printing. Mesin *digital printing* digunakan untuk melakukan pencetakan suatu gambar, seni, atau bahkan hingga promosi seperti iklan produk. Suatu mesin harus tetap dijaga perawatannya agar tetap menghasilkan kinerja percetakan dengan hasil yang maksimal. Kurangnya perawatan pada suatu mesin *digital printing* akan menyebabkan mesin bermasalah dan tidak bisa digunakan untuk melakukan percetakan.

CV. Warna Makassar merupakan suatu perusahaan supplier digital printing yang berada di kota Makassar, Sulawesi Selatan. Selain itu, CV. Warna Makassar merupakan *partner* perusahaan dari Alea Grafika yang berada di kota Surabaya, Jawa Timur. Alea Grafika merupakan perusahaan yang berdiri sejak 2010 bergerak dalam industri Bahan Offset, Bahan Digital Printing, Tinta, Sticker Camel, Pop Aging, Stragloss [1]. *Customer* CV. Warna Makassar lebih banyak mencakup daerah kota-kota pada pulau Sulawesi. Permintaan pesanan mesin atau bahan yang diminta customer akan sering menggunakan ekspedisi untuk customer yang berada diluar kota atau provinsi Sulawesi.

Oleh karena itu, jika terjadi pemeliharaan permintaan perawatan mesin teknisi akan sering melakukan perjalanan keluar kota. Sehingga belakangan ini pihak CV. Warna Makassar melakukan penyebaran untuk teknisinya di beberapa kota yang cukup jauh dari kota Makassar, provinsi Sulawesi Selatan. Produktivitas dalam penggunaan mesin industri percetakan, maka kebutuhan akan fungsi pemeliharaan mesin akan semakin bertambah besar. Tidak hanya kurangnya perawatan penyebab suatu mesin bermasalah memiliki banyak aspek mulai dari mesin sudah terlalu lama digunakan, *human error* (kesalahan pengguna), mesin yang kotor, teknisi mesin yang kurang perhatian terhadap mesin dan lain-lain. Industri *digital printing* sering terjadi peningkatan percetakan seperti pada saat masa-masa tertentu. Potensi lingkungan permintaan akan teknologi print digital yang semakin meningkat dalam kegiatan-kegiatan umum maupun pribadi seseorang, maka usaha digital printing mempunyai pangsa pasar prospektif dan tidak pernah surut [2]. Oleh karena itu peluang usaha dalam bisnis *digital printing* perusahaan - perusahaan yang memiliki jenis mesin *digital printing outdoor* akan lebih sering mendapatkan pesanan tersebut. Karena hasil cetak mesin *digital printing outdoor* memiliki ukuran yang lebih besar dari jenis mesin digital printing lainnya. Sehingga penempatan mesin tersebut harus memiliki ruangan atau tempat yang cukup dengan kondisi dan hasil cetak pada mesin *digital printing outdoor*.

Selain itu, persaingan pada industri digital printing pada setiap perusahaan digital printing yang ada juga memiliki pengaruh untuk bisa menarik perhatian pembeli untuk melakukan pemesanan percetakan spanduk atau baliho. Misalnya, merek pada mesin *digital printing outdoor*, karena teknologi pada suatu merek *digital printing outdoor* tentu saja berbeda-beda. Hal tersebut juga akan memberikan hasil cetak kualitas yang bagus jika digabungkan antara merek mesin pada digital printing yang berkualitas dan perawatan berkala pada suatu mesin yang digunakan. Perusahaan digital printing yang ingin bertahan dan meraih keunggulan bersaing dalam kegiatan bisnis, tidak hanya harus mampu menyajikan teknologi yang canggih dan terstandarisasi, tetapi juga harus mampu memberikan pelayanan yang berkualitas kepada para pelanggannya [3].

Kendala utama customer saat ini yaitu terjadinya peningkatan permintaan untuk perawatan mesin digital printing outdoor *customer* sering terhambat, sehingga tidak tersalurkan dengan baik kepada teknisi. Hal tersebut dikarenakan komunikasi yang masih menggunakan telepon atau aplikasi *whatsapp*. Identifikasi permasalahan pada mesin *digital printing* untuk melakukan penjadwalan perawatan juga masih kurang jelas. Karena laporan riwayat transaksi dalam penjadwalan perawatan mesin yang masih dilakukan secara manual dan belum terdata dengan baik. Seperti pendataan pada catatan kertas atau menggunakan file *excel*. Hal tersebut memungkinkan terjadinya kehilangan data yang disebabkan *human-error*. Permasalahan lainnya adalah terjadi permintaan proses transaksi barang atau bahan digital printing yang terus meningkat dari *customer*. Serta saat terjadi *event* tertentu, perawatan mesin yang harus segera diperbaiki membutuhkan pendataan dan informasi yang jelas.

Oleh karena itu, upaya untuk meningkatkan proses permintaan atau penjadwalan perbaikan pada mesin digital printing *outdoor* dan mengatasi kendala-kendala tersebut. Maka, dengan melakukan pengembangan sistem aplikasi perawatan atau penjadwalan mesin digital printing *outdoor* untuk CV. Warna Makassar dan *customer*. Dalam pengembangan penelitian ini menggunakan metode waterfall dengan tahapan-tahapannya yaitu identifikasi, analisis, implementasi dan pengujian. Fitur utama yang dikembangkan yaitu customer dapat melakukan permintaan transaksi perawatan mesin dan dari setiap transaksi tersebut terdapat laporan transaksi yang dapat dimonitoring oleh CV. Warna Makassar. Teknologi yang digunakan yaitu framework Laravel dan VueJS yang digunakan untuk membantu dalam pengembangan sistem aplikasi berbasis website.

Hasil akhir dari penelitian ini adalah rancang bangun aplikasi sistem penjadwalan atau permintaan pemeliharaan mesin *digital printing outdoor* berbasis website (Studi Kasus: CV. Warna Makassar). Dengan adanya aplikasi ini bisa mendaftarkan dan mengelola proses permintaan atau penjadwalan perbaikan pada mesin digital printing *outdoor*. Sehingga nantinya teknisi bisa melakukan monitor permintaan atau penjadwalan pemeliharaan mesin dari *customer*. Teknisi juga dapat mengidentifikasi bagian mesin yang diperlukan saat akan melakukan pemeliharaan mesin tersebut. Serta laporan setiap transaksi perawatan mesin.

## Topik dan Batasannya

### A. Rumusan Masalah

Berdasarkan penelitian, maka permasalahan dirumuskan sebagai berikut:

- 1) Bagaimana cara mengelola permintaan customer dalam melakukan perawatan mesin digital printing outdoor?
- 2) Metode pengembangan apa yang digunakan aplikasi sistem penjadwalan dan perawatan mesin digital printing outdoor?
- 3) Bagaimana mendata transaksi penjadwalan dan perawatan mesin digital printing outdoor?
- 4) Bagaimana teknisi CV. Warna Makassar dapat melakukan monitoring status transaksi permintaan perawatan mesin digital printing outdoor?
- 5) Apakah perancangan aplikasi sesuai dengan sistem atau fitur yang dikembangkan?

### B. Batasan Masalah

Batasan dari penelitian ini sebagai berikut:

- 1) Metode yang digunakan adalah Waterfall dengan framework Laravel, Vuejs dan MySQL
- 2) Penelitian hanya mencakup seluruh mesin *digital printing outdoor* yang ada pada industri digital printing CV. Warna Makassar
- 3) Penelitian ini berfokus pada perawatan dan penjadwalan mesin digital printing outdoor untuk customer CV. Warna Makassar
- 4) Tidak mencakup pembayaran secara online dalam aplikasi, sehingga pembayaran dilakukan saat bertemu langsung dengan teknisi CV. Warna Makassar yang telah dilakukan pemesanan melalui aplikasi tersebut.
- 5) Aplikasi yang dikembangkan penelitian ini berbasis website.

## Tujuan

Tujuan yang ingin dicapai dari penelitian ini yaitu sebagai berikut:

1. Mengelola perawatan permintaan pemeliharaan *mesin digital printing outdoor* pada aplikasi berbasis website di CV. Warna Makassar.
2. Menerapkan metode pengembangan *waterfall* dan mengikuti setiap langkah proses tahap *waterfall* secara terstruktur terhadap aplikasi yang dikembangkan.
3. Mengembangkan aplikasi dapat menyimpan data laporan dan detail transaksi perawatan mesin digital printing outdoor.
4. Mengembangkan aplikasi yang dapat digunakan oleh teknisi atau admin untuk melakukan monitoring status setiap transaksi perawatan.
5. Menguji data setiap fungsional fitur dalam aplikasi telah sesuai dengan perancangan sistem yang dikembangkan.

Manfaat yang ingin dicapai dari penelitian ini yaitu sebagai berikut:

1. Bagi akademik, memberikan referensi manfaat penelitian dalam mengetahui konsep pengembangan implementasi menggunakan metode waterfall dengan framework Laravel, Vuejs dan MySQL.
2. Bagi CV. Warna Makassar, aplikasi sistem penjadwalan atau permintaan pemeliharaan *mesin digital printing outdoor* dapat meningkatkan dan memudahkan proses penjadwalan dan permintaan perawatan.

## Organisasi Tulisan

Pada sub-bagian ini, memiliki struktur dan susunan tugas akhir. Terbagi menjadi beberapa bab yaitu sebagai berikut:

### 1) Pendahuluan

Bagian ini merupakan bagian awal dari isi penilitan tugas akhir yang menjelaskan dari latar belakang penilitian, rumusan masalah, tujuan dan manfaat, serta batasan penilitian.

### 2) Studi Terkait

Bagian ini membahas penilitian, teori-teori atau hasil penelitian yang sudah ada untuk dijadikan pendukung pada penilitian tugas akhir ini.

### 3) Sistem Yang Dibangun

Bagian ini membahas spesifik sistem yang digunakan pada penilitian aplikasi, metode yang digunakan, rancangan dan jenis pengujian yang digunakan.

### 4) Evaluasi

Bagian ini membahas hasil atau pengujian dari aplikasi penilitian yang telah dilakukan. Menjelaskan jenis

dan hasil yang telah ditemukan dari pengujian yang dilakukan.

## 5) Kesimpulan

Bagian ini membahas kesimpulan dan saran dari aplikasi penelitian yang telah dilakukan. Menjelaskan hasil akhir yang tersimpulkan dan beberapa saran yang dapat dikembangkan pada penelitian selanjutnya.

## 2. Studi Terkait

Studi terkait digunakan sebagai acuan penelitian tugas akhir ini untuk menghindari adanya persamaan dari penelitian sebelumnya. Tujuan lainnya yaitu untuk mengembangkan dan menambahkan penelitian tugas akhir studi kasus yang berbeda. Berikut beberapa studi terkait beberapa penelitian yaitu, sistem manajemen dalam pemeliharaan pada mesin percetakan yang berbasis desktop menggunakan teknologi Vb dengan metode waterfall [4]. Penelitian ini dirujuk karena menggunakan metode waterfall serta berkaitan dengan *digital printing*. Tetapi, yang membedakan dari penelitian tersebut berbasis desktop dan menggunakan Vb (*Visual Basic*).

Studi terkait yang kedua adalah perawatan peras komputer yang berbasis desktop [5]. Penelitian juga menggunakan waterfall model, hal yang membedakan yaitu masih berbasis desktop dengan studi kasus PT.Sochin Indonesia.

Studi terkait yang ketiga adalah Sistem Perawatan dan Perbaikan pada fasilitas-fasilitas PT.PLN yang menggunakan metode prototype [6]. Penelitian tersebut memiliki kesamaan tentang perawatan dan perbaikan, tetapi metode yang digunakan adalah metode prototype.

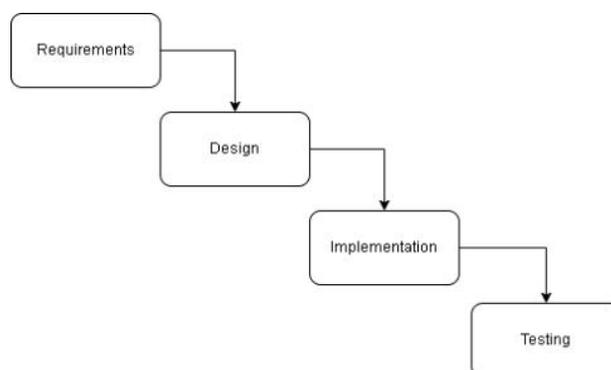
Studi terkait yang keempat adalah Aplikasi Penyedia Jasa Perawatan Berbasis Seluler [7]. Penelitian ini memiliki sedikit persamaan tentang perawatan pada aplikasi, tetapi memiliki perbedaan pada aplikasi yang berbasis seluler atau android dan database yang digunakan. Penelitian lainnya adalah Aplikasi Perawatan Wajah yang berbasis website [8]. Penelitian ini juga memiliki sedikit persamaan tentang perawatan pada aplikasi, tetapi yang memiliki perbedaan pada aplikasi yang tidak menggunakan teknologi atau framework pada pengembangannya.

Pada beberapa penelitian terdahulu atau studi terkait tersebut memiliki persamaan dan perbedaan dengan penelitian yang dilakukan saat ini. Beberapa persamaan yang sama yaitu seperti *digital printing*, berkaitan dengan perawatan dan perbaikan, serta metode yang digunakan pada penelitian terdahulu. Sedangkan untuk perbedaannya untuk hasil akhir dari penelitian adalah framework yang digunakan dan objek penelitian yang dituju.

## 3. Sistem yang Dibangun

### 3.1 Metode yang digunakan

Penelitian ini menggunakan metode *waterfall*, salah satu SDLC (*Software Development Life Cycle*). Metode Waterfall merupakan metode pengembangan yang memiliki tahapan yang terstruktur dan berurutan. Tahapan proses yang terpisah seperti spesifikasi kebutuhan, desain perangkat lunak, implementasi, pengujian, dan sebagainya [9]. Berikut penjelasan diagram secara detail dan rinci dalam mengembangkan penelitian ini dari beberapa langkah yang digunakan pada metode waterfall:



Gambar 3.1 Metode Waterfall

#### 1) Requirements

Requirement merupakan tahap awal dari metode pengembangan waterfall. Tahap ini merupakan tahap untuk mengidentifikasi dan menganalisa kebutuhan terhadap sistem aplikasi yang dikembangkan.

## 2) Design

Pada tahap ini merupakan tahap kedua setelah kebutuhan detail aplikasi telah di sepakati pada tahap waterfall sebelumnya. Seperti perancangan sistem, usecase diagram, activity diagram, design database dan UI/UX.

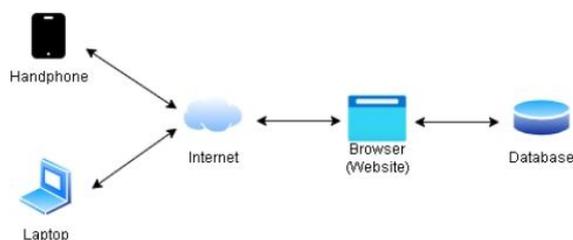
## 3) Implementation

Tahap implementasi merupakan tahap pembuatan kode dilakukan untuk menerapkan fitur-fitur dan sistem aplikasi. Sesuai dengan spesifikasi dan kebutuhan yang telah ditentukan pada tahap-tahap metode waterfall sebelumnya. Termasuk implementasi bahasa pemrograman dan framework yang dibuat.

## 4) Testing

Tahap ini merupakan tahap metode waterfall setelah proses pengembangan aplikasi telah dilakukan. Salah satunya yaitu pengujian *Black Box Testing (Equivalence Partioning)* yang dilakukan oleh penulis tugas akhir. *Black Box Testing* merupakan pengujian yang dilakukan untuk mendeteksi hasil *input* dan *output* dari perangkat lunak tanpa mengetahui struktur kode program [10]. Untuk *Equivalence Partioning* merupakan teknik pengujian yang dibagi menjadi beberapa partisi test untuk mengetahui partisi tersebut benar atau salah [11].

### 3.2 Deskripsi Umum Sistem



**Gambar 1.1** Deskripsi Umum Sistem

Dari gambar diatas menjelaskan bahwa gambaran umum sistem saat *user* atau pengguna melakukan akses pada aplikasi. Yang diakses melalui website melalui browser dengan syarat memiliki internet. Kemudian, aplikasi website akan mengambil data yang ada pada database. Website mengandung informasi yang bisa terdiri dari teks, gambar, audio dan lain-lain. Setiap website memiliki penamaan domain masing-masing yang digunakan sebagai identifikasi website [12].

### 3.3 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak (SKPL) yang meliputi kebutuhan fungsional, kebutuhan non-fungsional, detail fitur dan kasus penggunaan termasuk *use case diagram* dan *activity diagram*. Use Case Diagram adalah jenis diagram yang digunakan untuk menggambarkan suatu interaksi sistem untuk mengembangkan suatu perangkat lunak. Use Case Diagram adalah salah satu dari berbagai jenis diagram UML (*Unified Modified Language*) yang menggambarkan hubungan interaksi anatara sistem dan aktor [13].

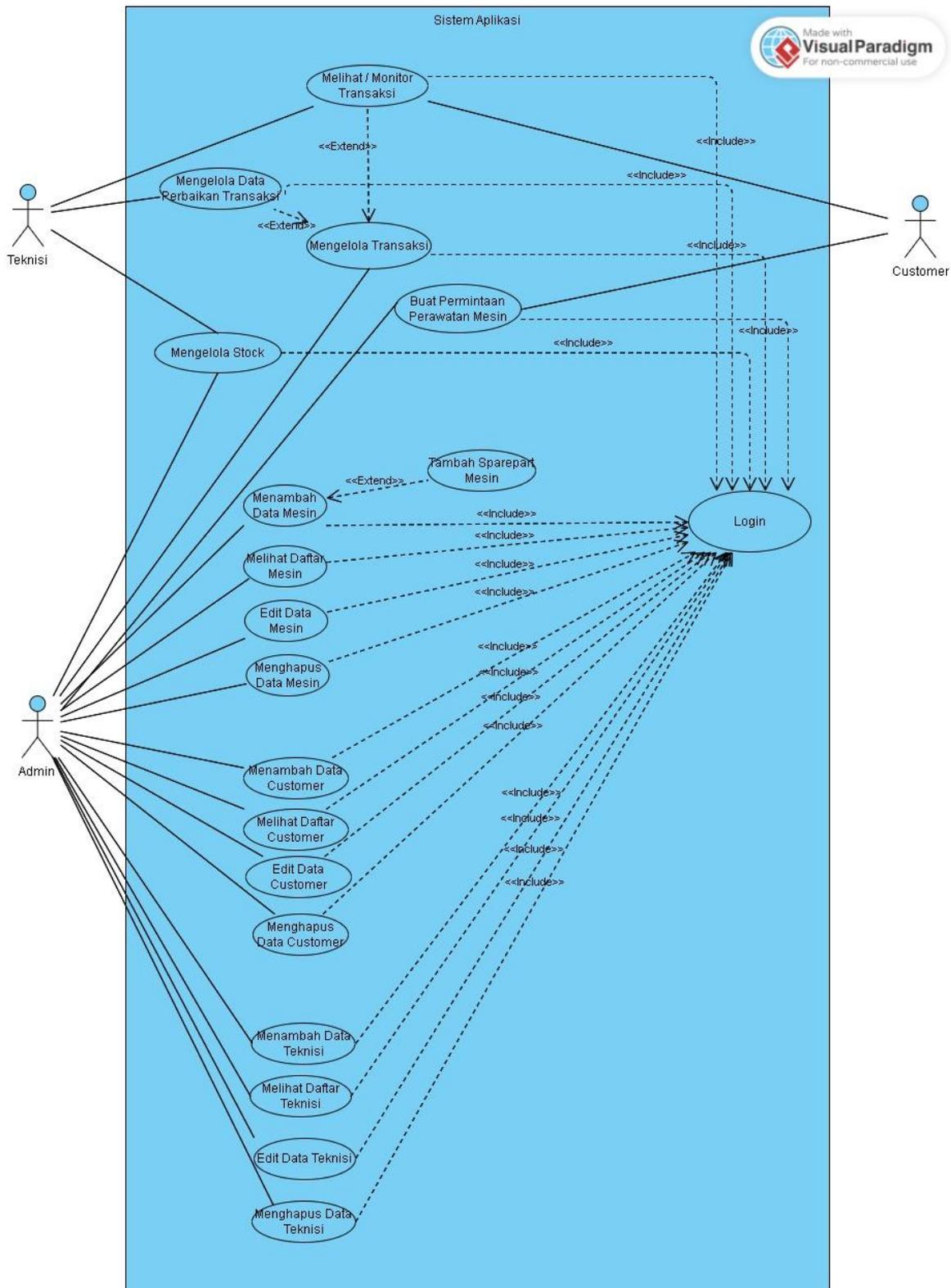
Sedangkan, Activity Diagram adalah merupakan diagram yang mengilustrasikan proses bisnis aktivitas pada suatu sistem atau fitur dalam perangkat lunak. Activity Diagram juga merupakan salah satu UML (*Unified Modified Language*). Activity Diagram dalam bahasa Indonesia diagram aktivitas, yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Proses dari suatu sistem digambarkan secara vertical. Activity Diagram merupakan pengembangan dari Use Case yang memiliki alur aktivitas [14].

### 3.6 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan aspek-aspek yang tidak berkaitan dengan fitur-fitur yang ada pada aplikasi. Aspek yang terdapat pada kebutuhan non-fungsional lebih umum seperti pengalaman pengguna. Berikut adalah beberapa kebutuhan non-fungsional pada aplikasi yaitu:

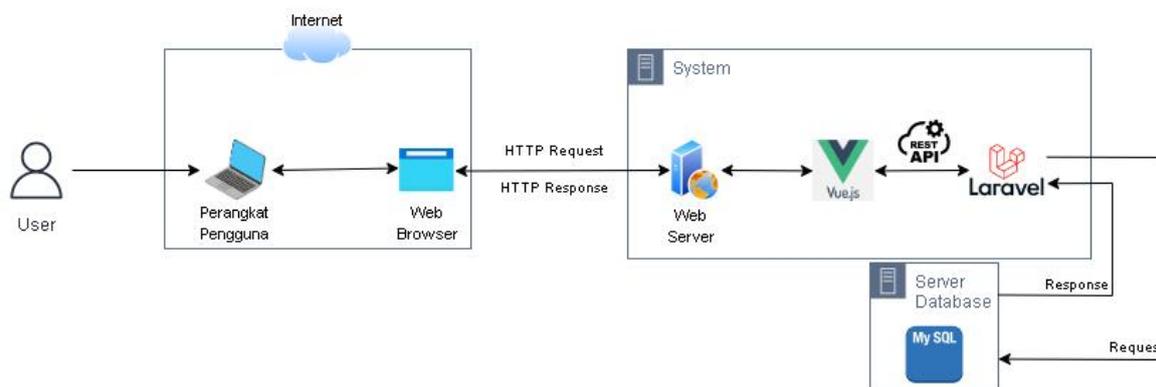
- 1) Aplikasi dapat digunakan secara fleksibel oleh pengguna untuk perangkat yang digunakan dengan tampilan yang responsif. Sehingga, pengguna mudah dan paham dalam menggunakan aplikasi.
- 2) Keandalan aplikasi bisa sesuai dengan yang diharapkan pengguna seperti fitur-fitur berfungsi dengan baik.

### 3.7 Use Case Diagram



Gambar 1.2 Use Case Diagram

### 3.8 Perancangan Arsitektur Sistem



Gambar 1.3 Arsitektur Sistem

Pada gambar diatas, merupakan ilustrasi gambaran sistem yang akan dikembangkan. Proses awal sistem dimulai dari sumber database MySQL. Dalam proses penyimpanan dan pengambilan database diakses menggunakan framework Laravel sebagai *backend* yang berfungsi sebagai REST API. Dari REST API tersebut digunakan sebagai proses *request & response* pada bagian frontend VueJS, sebelum digunakan oleh frontend REST API dilakukan uji tes *request & response* terlebih dahulu menggunakan *software Postman*. Karena VueJS sebagai frontend yang juga berfungsi menampilkan *user experience*. Semua proses tersebut akan masuk sebagai web server agar aplikasi dapat tampil pada browser. Setiap proses dari *Web Browser* ke *Web Server* dihubungkan dengan *HTTP Request* dan *HTTP Response*. Kemudian, dapat diakses pada web browser pengguna melalui perangkat pengguna menggunakan internet.

Aristektur sistem menggunakan REST API dilakukan agar dapat memperoleh data yang dibutuhkan. Karena melalui REST API data-data yang tampil berformat *JSON* tersebut berasal dari database. Yang dibuat dan diproses dalam *framework* Laravel. Sedangkan *framework* VueJS digunakan pada penelitian ini, karena digunakan untuk mengelola REST API tersebut agar data-data dapat ditampilkan pada tampilan browser secara *user experience* dengan desain tampilan yang sesuai aplikasi.

Pada penelitian ini, pengguna membuka browser yang memiliki akses internet menggunakan perangkat digital *handphone / laptop*. Saat aplikasi dibuka pada website browser, website tersebut bisa diakses karena ada proses *HTTP Request* dan *HTTP Response* yang berasal dari Web Server yang menghubungkan keseluruhan aplikasi. Kemudian, tampilan dan desain website diproses menggunakan VueJS. Yang juga berfungsi melakukan proses *request & response* terhadap REST API yang diprogram menggunakan Laravel. Langkah terakhir, database dapat diakses dalam pengambilan data-data. Hasil akhir *output* aplikasi yang ditampilkan pada *user* berupa informasi data transaksi penjadwalan dan permintaan, mesin, & sparepart.

### 3.9 Perancangan Basis Data

Perancangan basis data menjelaskan tentang struktur tabel dalam database. Perancangan desain database pada penelitian ini menggunakan ERD (*Entity Relationship Diagram*) dan database yang digunakan MySQL. MySQL adalah sebuah database management system (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal [18]. MySQL masih populer di dunia karena memiliki kemudahan dan kecepatan dalam mengelola database serta *open-source*. Selain itu, memiliki komunitas yang besar terutama dalam pengembangan situs website atau aplikasi berbasis data. Sedangkan untuk ERD, merupakan suatu diagram yang menggambarkan dan digunakan untuk mempresantisikan bentuk struktur data dalam sistem aplikasi. Pada diagram ERD terdapat keterkaitan antar entitas, hubungan entitas dan atribut entitas basis data yang saling terhubung atau tidak terhubung ke setiap entitas yang ada [19]. Tujuan ERD untuk memudahkan dalam pembuatan struktur basis data pada suatu sistem aplikasi.

### 3.10 Perancangan Antarmuka Pengguna

Perancangan antarmuka pengguna berupa halaman desain antarmuka pengguna pada aplikasi sistem permintaan atau penjadwalan perawatan mesin berbasis website di CV. Warna Makassar. Perancangan antarmuka ini sebagai rancangan bentuk sketsa awal dalam aplikasi sebelum dilakukan implementasi desain sebenarnya.

## 4. Evaluasi

### 4.1 Hasil Pengujian

Pengujian yang diperoleh memberikan hasil yang relevan baik berdasarkan metode uji test yang digunakan. Input-input utama berfungsi sesuai dengan format input yang sesuai. Penambahan atau perubahan data memiliki masing-masing notifikasi yang diperlukan pada setiap halaman aplikasi.

#### 4.2.1 Black Box Testing (Equivalence Partitioning)

Pada pengujian black box testing menggunakan metode *Equivalence Partitioning*. Beberapa partisi kondisi testing menentukan kondisi suatu input yang ada pada sistem aplikasi melalui metode tersebut. Seperti input penting pada buat perawatan transaksi, update transaksi dan lain-lain.

### 4.2 Analisis Hasil Pengujian

Berdasarkan hasil pengujian terdapat bagian atau kondisi tertentu yang memiliki kasus berbeda. Pada dasarnya saat input yang dimasukkan tidak sesuai dengan format atau syarat tertentu. Harus menampilkan notifikasi pesan kesalahan sesuai dengan syara format input. Analisa yang didapatkan rata-rata keseluruhan hasil pengujian atau notifikasi sesuai dengan kebutuhan input aplikasi. Hal tersebut diperoleh berdasarkan uji *test case scenario*.

#### 4.3.1 Test Case Scenario

Pengujian *test case scenario* untuk melakukan uji test pada setiap diagram *use case* dan menguji setiap fitur aplikasi pada masing-masing *use case*. Uji ini dilakukan untuk mengetahui setiap kondisi scenario yang berbeda. Untuk mendapatkan hasil yang diharapkan dan hasil yang terjadi pada fitur aplikasi. Kondisi scenario yang berhasil memiliki hasil uji sukses, sedangkan yang gagal memiliki hasil uji gagal. Bukti hasil pengujian berada bagian **Lampiran**.

## 5. Kesimpulan

Berdasarkan bagian evaluasi yaitu hasil implementasi, hasil pengujian dan hasil analisis pengujian, dapat disimpulkan bahwa:

- 1) Aplikasi sistem penjadwalan dan perawatan pemeliharaan mesin *digital printing outdoor* di CV. Warna Makassar telah berhasil diimplementasikan. Yang terdapat halaman dan form isian saat login sebagai admin atau customer untuk membuat penjadwalan perawatan mesin yang kemudian diproses kembali dari sisi admin dan teknisi. Terdapat 3 jenis pengguna dalam aplikasi yaitu admin, *customer*, dan teknisi.
- 2) Hasil penelitian aplikasi berhasil menerapkan metode pengembangan waterfall. Melalui tahapan struktur waterfall yaitu *requirements, design, implementation & testing*. Yang pada setiap tahapannya dapat menerapkan kebutuhan sistem, desain database, implementasi dan melakukan metode *Black Box Testing (Equivalenve Partitioning)*.
- 3) Pendataan setiap transaksi perawatan berhasil tersimpan pada sistem aplikasi. Dengan status yang sesuai pada setiap transaksi perawatan mesin. Dari tahap awal customer melakukan permintaan hingga proses akhir selesai.
- 4) Aplikasi berhasil menerapkan fitur bagi teknisi, yang dapat melakukan monitoring dan melakukan proses selanjutnya pada setiap transaksi perawatan. Karena teknisi juga memiliki peran utama setiap transaksi untuk melengkapi data saat perawatan mesin berlangsung.
- 5) Aplikasi sistem telah berhasil menyediakan fitur-fitur utama yang membantu proses penjadwalan dan perawatan mesin. Dengan menerapkan perancangan sistem awal yaitu menggunakan teknologi framework VueJS dan Laravel. Fitur tersebut diantaranya: membuat permintaan perawatan mesin, laporan data transaksi, pengelolaan stok ,serta pengelolaan data (mesin, *sparepart, customer, teknisi, transaksi*).

Selain itu, terdapat beberapa saran yang dapat digunakan dan dipertimbangkan pada tahap pengembangan selanjutnya. Berikut poin-poin sarannya, yaitu:

- 1) Menambahkan laporan data statistik juga pada bagian sisi *customer* dan teknisi, sehingga setiap jenis pengguna sistem aplikasi memiliki data statistik tersendiri pada halaman bagian dashboard.
- 2) Memberikan sistem pembayaran online atau pembayaran yang menggunakan *payment gateway* dalam sistem aplikasi. Agar setiap transaksi tidak perlu dilakukan pembayaran secara langsung tatap muka atau secara offline.
- 3) Memberikan informasi lebih lengkap pada setiap detail transaksi, seperti terdapat nota transaksi. Untuk membuat setiap transaksi memiliki bukti data perawatan mesin yang lebih baik dan lebih *valid*.

**Daftar Pustaka**

- [1] Alea Grafika, "About Us," *Alea Grafika*, 2010. [https://www.aleagrafika.co.id/page/view/2\\_about\\_us](https://www.aleagrafika.co.id/page/view/2_about_us) (accessed Jul. 04, 2023).
- [2] I. G. P. W. Putra, N. K. Murdani, and N. P. A. D. Laksmi, "Studi Kelayakan Bisnis Usaha Jasa Desain Grafis dan Digital Printing Pada Gede Project Design (Studi kasus pada Industri Kecil Unit Percetakan Desain Grafis 'Gede Project Design' Denpasar - Bali)," *J. Satyagraha*, vol. 4, no. 1, pp. 1–11, 2021.
- [3] D. Martin and A. Y. Hamali, "Pengaruh Kualitas Pelayanan Terhadap Loyalitas Pelanggan Pada Pt Adidaya Digital Printing Bandung," *J. Manage.*, vol. 3, no. 1, pp. 30–48, 2020.
- [4] V. Septiara, A. Ambarwati, and A. Nugroho, "Rancang Bangun Sistem Informasi Manajemen Pemeliharaan Mesin Percetakan Digital Printing Berbasis Desktopmenggunakan Vb," *Net (Studi Kasus Cv. Wujud Unggul). Sist. Inf.*, 2020.
- [5] Mulya Alfian, "Perancangan Aplikasi Manajemen Perawatan Perangkat Keras Komputer Pada PT. Socfin Indonesia," *J. Comput. Sci. Informatics Eng.*, vol. 01, no. 2, pp. 78–86, 2022, doi: 10.55537/cosie.v1i2.38.
- [6] A. Wantoro, S. Samsugi, and M. J. Suharyanto, "Sistem Monitoring Perawatan dan Perbaikan Fasilitas PT PLN (Studi Kasus : Kota Metro Lampung)," *J. TEKNO KOMPAK*, vol. 15, no. 1, pp. 116–130, 2021.
- [7] J. Maylia Suhendro, M. Sudarma, and D. Care Khrisne, "Rancang Bangun Aplikasi Seluler Penyedia Jasa Perawatan Dan Kecantikan Menggunakan Framework Flutter," *J. SPEKTRUM*, vol. 8, no. 2, p. 68, 2021, doi: 10.24843/spektrum.2021.v08.i02.p9.
- [8] Devi Indriani, "Aplikasi Kecantikan Dan Perawatan Wajah Berbasis Website," *J. Tek. dan Sci.*, vol. 1, no. 2, pp. 32–42, 2022, doi: 10.56127/jts.v1i2.27.
- [9] I. Sommerville, *Software Engineering (9th ed.; Boston, Ed.)*. Massachusetts: Pearson Education. 2011.
- [10] R. Setiawan, "Apa itu Black Box Testing," *Dicoding*, 2021. <https://www.dicoding.com/blog/black-box-testing/> (accessed Nov. 24, 2023).
- [11] J. Point, "Equivalence Partitioning Technique," *Javatpoint*, 2024. <https://www.javatpoint.com/equivalence-partitioning-technique-in-black-box-testing> (accessed Jul. 04, 2024).
- [12] K. Islam and D. Agushinta Rahayu, "Evaluasi Antarmuka Website Tokopedia menggunakan Metode Heuristic," *Energy - J. Ilm. Ilmu-Ilmu Tek.*, vol. 8, no. 1, pp. 33–38, 2018.
- [13] Dicoding Intern, "Apa itu Use Case Diagram?," *Dicoding*, 2021. <https://www.dicoding.com/blog/contoh-use-case-diagram/> (accessed Nov. 24, 2023).
- [14] Dicoding Intern, "Apa itu Activity Diagram? Beserta Pengertian, Tujuan, Komponen," *Dicoding*, 2021. <https://www.dicoding.com/blog/apa-itu-activity-diagram/> (accessed Nov. 24, 2023).
- [15] R. Patria, "Rest API adalah: Perbedaan Rest API dan Restful API," *Domainesia*, 2023. <https://www.domainesia.com/berita/rest-api-adalah/> (accessed Jul. 03, 2024).
- [16] Taylor Otwell, "About Laravel," *Laravel*, 2023. <https://github.com/laravel/laravel> (accessed Nov. 02, 2023).
- [17] VueJS, "What is Vue?," *Evan You*, 2023. <https://vuejs.org/guide/introduction.html#what-is-vue> (accessed Nov. 02, 2023).
- [18] Y. K., "Pengertian MySQL, Fungsi, dan Cara Kerjanya (Lengkap)," *Niagahoster*, 2022. <https://www.niagahoster.co.id/blog/mysql-adalah/> (accessed Nov. 02, 2023).
- [19] LucidChart, "What is an Entity Relationship Diagram (ERD)?," *Lucid Chart*. .

## Lampiran

No.	Kebutuhan Fungsional	Deskripsi
1.	Melakukan <i>Log-in</i>	User dapat melakukan login ke sistem aplikasi user yang dimaksud admin, Teknisi, customer sesuai batas hak akses masing-masing
2.	Mengelola data mesin digital printing outdoor	Admin bisa menambah, mengubah, menghapus data mesin digital printing dan bagian-bagian mesin
3.	Mengelola data bagian-bagian mesin atau bahan ( <i>sparepart</i> ) mesin digital printing outdoor	Admin bisa menambah, mengubah, menghapus data bagian-bagian mesin atau bahan digital printing ke mesin digital printing outdoor yang terkait
4.	Mengelola data Teknisi	Admin dapat menambah, mengubah, dan menghapus data Teknisi
5.	Mengelola data <i>customer</i>	Admin dapat menambah, mengubah, dan menghapus data <i>customer</i>
6.	Permintaan atau penjadwalan perawatan mesin digital printing outdoor	<i>Customer</i> bisa melakukan permintaan atau penjadwalan untuk perawatan mesin digital printing outdoor, termasuk bagian-bagian mesin atau bahan digital printing pada mesin
7.	Mengelola permintaan atau penjadwalan perawatan mesin digital printing outdoor	Admin dapat mengelola permintaan atau penjadwalan perawatan mesin digital printing outdoor yang diminta oleh <i>customer</i> , sedangkan Teknisi bisa memonitoring permintaan tersebut.
8.	Mengelola transaksi permintaan atau penjadwalan perawatan mesin digital printing outdoor	Admin dan Teknisi dapat mengelola monitoring status transaksi permintaan atau penjadwalan perawatan mesin digital printing outdoor yang diminta oleh <i>customer</i> , sedangkan <i>customer</i> bisatransaksi

Lampiran 1. 1 Kebutuhan Fungsional

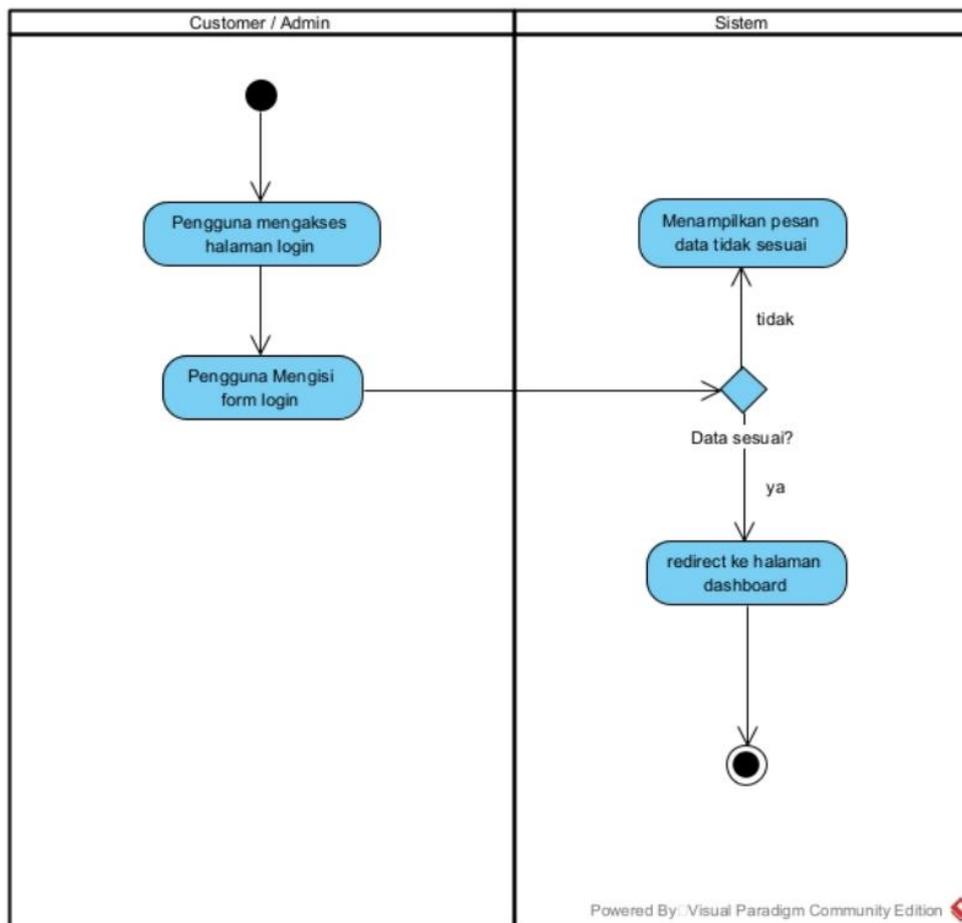
No.	Fitur	Detail	Data	Roles
1.	<i>Log-in</i>	Login	Data login username dan password	Admin, Teknisi, customer
2.	Pengelolaan mesin digital printing outdoor	Penambahan data mesin digital printing outdoor	Data mesin digital printing outdoor	Admin
		Edit data mesin digital printing outdoor	Pengelolaan data mesin digital printing outdoor yang telah ditambahkan	Admin
3.	Pengelolaan stock bagian-bagian mesin atau bahan ( <i>sparepart</i> ) mesin digital printing outdoor	Penambahan data stock	Data bagian-bagian mesin atau bahan mesin digital printing outdoor	Admin, Teknisi
		Stock opname	Pengurangan atau penambahan data stock bagian-bagian atau bahan mesin digital printing berdasarkan transaksi	Admin, Teknisi
4.	Pengelolaan User	Registrasi Customer	Data Customer	Admin
		Registrasi Teknisi	Data Teknisi	Admin
		Edit data pengguna	Pengelolaan data seluruh user yang telah diregistrasi termasuk customer dan Teknisi	Admin
5.	Pengelolaan transaksi permintaan atau penjadwalan perawatan mesin digital printing outdoor	Daftar laporan transaksi permintaan atau penjadwalan perawatan mesin digital printing outdoor	Data permintaan atau penjadwalan, Data mesin digital printing outdoor, Data Teknisi	Admin, Teknisi, Customer
		Buat permintaan atau penjadwalan perawatan mesin digital printing outdoor	Data permintaan atau penjadwalan, Data mesin digital printing outdoor	Customer
		Validasi transaksi permintaan atau penjadwalan perawatan mesin digital printing	Detail transaksi, detail mesin digital printing outdoor,	Admin

		outdoor	pembayaran	
		Mengelola data perbaikan dan <i>tracking</i> status transaksi permintaan atau penjadwalan perawatan mesin digital printing outdoor	Transaksi permintaan atau penjadwalan perawatan mesin digital printing outdoor	Admin, Teknisi

Lampiran 1. 2 Detail Fitur

Kode Use Case	Nama Case	Aktor
UC-001	Login	Customer, Admin
UC-002	Buat permintaan perawatan mesin	Admin, Customer
UC-003	Melihat / Monitor transaksi	Customer, Teknisi, Admin
UC-004	Mengelola data perbaikan transaksi	Admin, Teknisi
UC-005	Mengelola stock	Admin, Teknisi
UC-006	Menambah data mesin	Admin
UC-007	Menambah sparepart data mesin	Admin
UC-008	Melihat data mesin	Admin
UC-009	Edit data mesin	Admin
UC-010	Melihat daftar customer	Admin
UC-011	Menambah data customer	Admin
UC-012	Edit data customer	Admin
UC-013	Menghapus data customer	Admin
UC-014	Menambah Teknisi	Admin
UC-015	Melihat data Teknisi	Admin
UC-016	Edit data Teknisi	Admin
UC-017	Menghapus data Teknisi	Admin

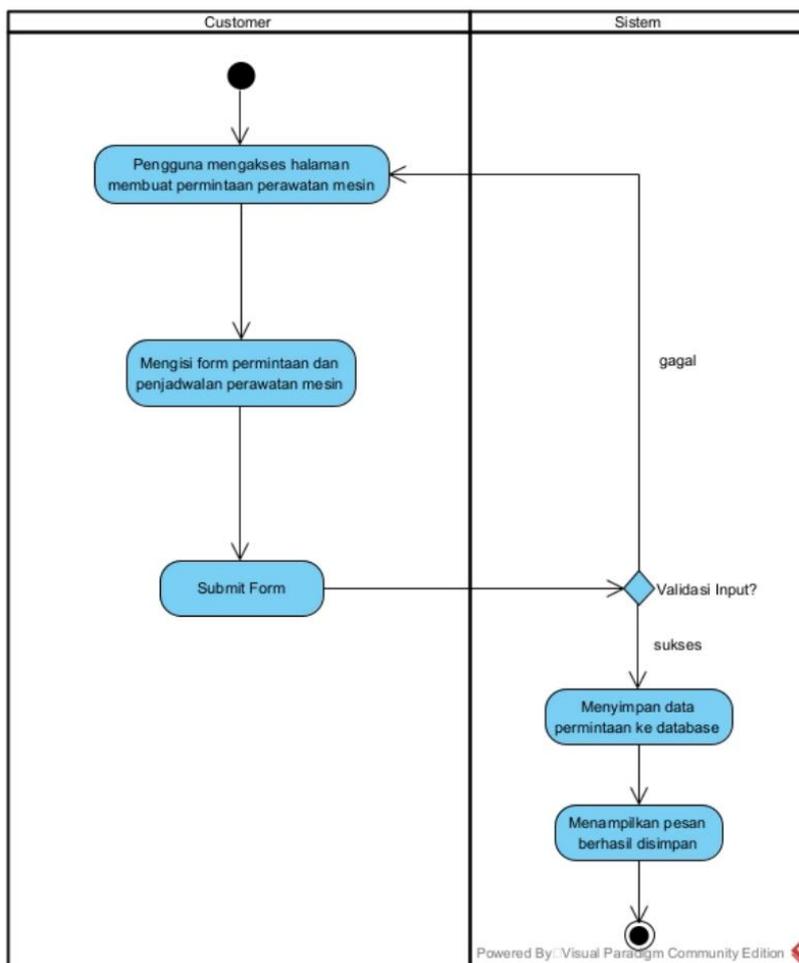
Lampiran 1. 3 Tabel Use Case Diagram



Lampiran 1.4 Activity Diagram UC-001

Kode Use Case	UC-001	
Nama Use Case	Login	
Aktor	Customer, Teknisi, Admin	
Deskripsi	Customer, Teknisi, Admin bisa melakukan login ke sistem aplikasi	
Relasi	-	
Kondisi Awal	Hanya bisa mengakses halaman login	
Kondisi Akhir	Bisa mengakses fitur aplikasi sesuai dengan hak akses	
Alur Kejadian Normal	Aktor	Sistem
	Mengisi form login	Login Berhasil
Alur Kejadian Tidak Normal	Aktor	Sistem
	Mengisi data form login yang salah	Menampilkan pesan data tidak sesuai

Lampiran 1 5 Use Case UC-001

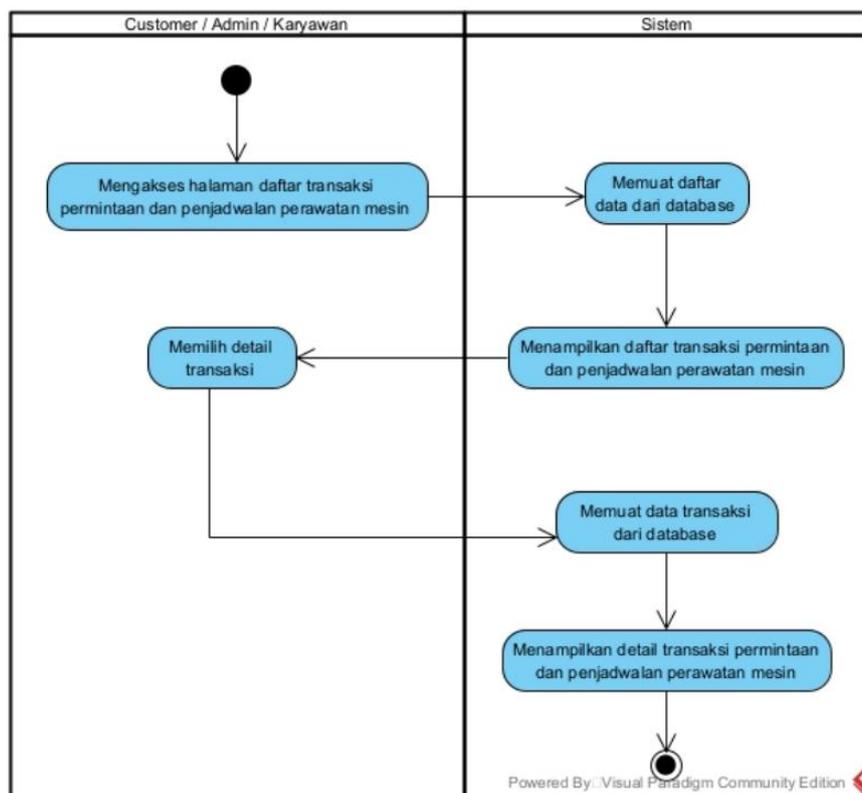


Lampiran 1.6 Activity Diagram UC-002

Kode Use Case	UC-002	
Nama Use Case	Buat permintaan dan penjadwalan perawatan mesin	
Aktor	Admin, Customer	
Deskripsi	Admin dan Customer bisa membuat permintaan perawatan mesin	
Relasi	UC-002	
Kondisi Awal	Halaman membuat permintaan perawatan mesin	
Kondisi Akhir	Data permintaan perawatan mesin berhasil disimpan	
Alur Kejadian Normal	Aktor	Sistem

	Admin, Customer mengakses halaman permintaan perawatan mesin	Menampilkan form halaman permintaan perawatan mesin
	Admin, Customer mengisi form permintaan perawatan mesin	Data disimpan ke database dan menampilkan data berhasil disimpan
Alur Kejadian Tidak Normal	Aktor	Sistem
	Admin, Customer mengakses halaman permintaan perawatan mesin	Menampilkan form halaman permintaan perawatan mesin
	Admin, Customer mengisi input yang salah pada form permintaan perawatan mesin	Data tidak berhasil disimpan dan menampilkan input tidak sesuai serta data tidak berhasil disimpan

Lampiran 1.7 Use Case UC-002

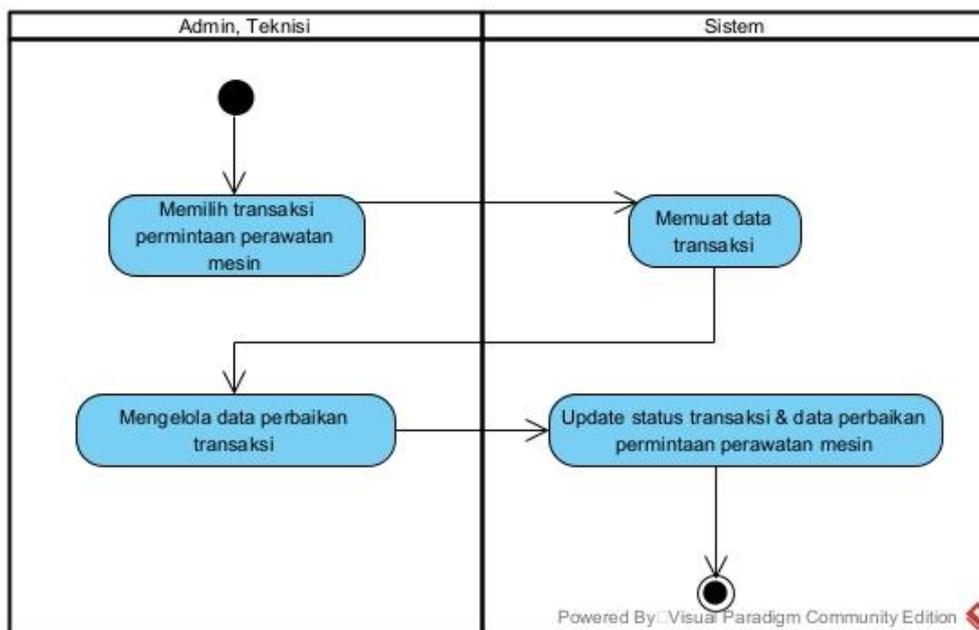


Lampiran 1.8 Activity Diagram UC-003

Kode Use Case	UC-003
Nama Use Case	Menampilkan transaksi
Aktor	Customer, Admin, Teknisi
Deskripsi	Customer bisa melihat daftar permintaan perawatan mesin yang telah dibuat sebelumnya Customer bisa melihat status permintaan perawatan mesin Admin bisa melihat daftar permintaan dari customer Admin bisa melihat status transaksi dari customer

	Pegawai bisa melihat daftar permintaan transaksi perawatan yang di-assign dari admin	
Relasi	-	
Kondisi Awal	Halaman daftar permintaan perawatan mesin	
Kondisi Akhir	Berhasil memuat data daftar permintaan perawatan mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. <i>Customer</i> / Admin / Teknisi mengakses halaman daftar transaksi	2. Memuat daftar data transaksi dari database, kemudian menampilkan daftar seluruh transaksi
	3. <i>Customer</i> / Admin / Teknisi salah satu detail transaksi permintaan perawatan mesin	4. Memuat detail data transaksi dari database, kemudian menampilkan data serta status transaksi

Lampiran 1.9 Use Case UC-003

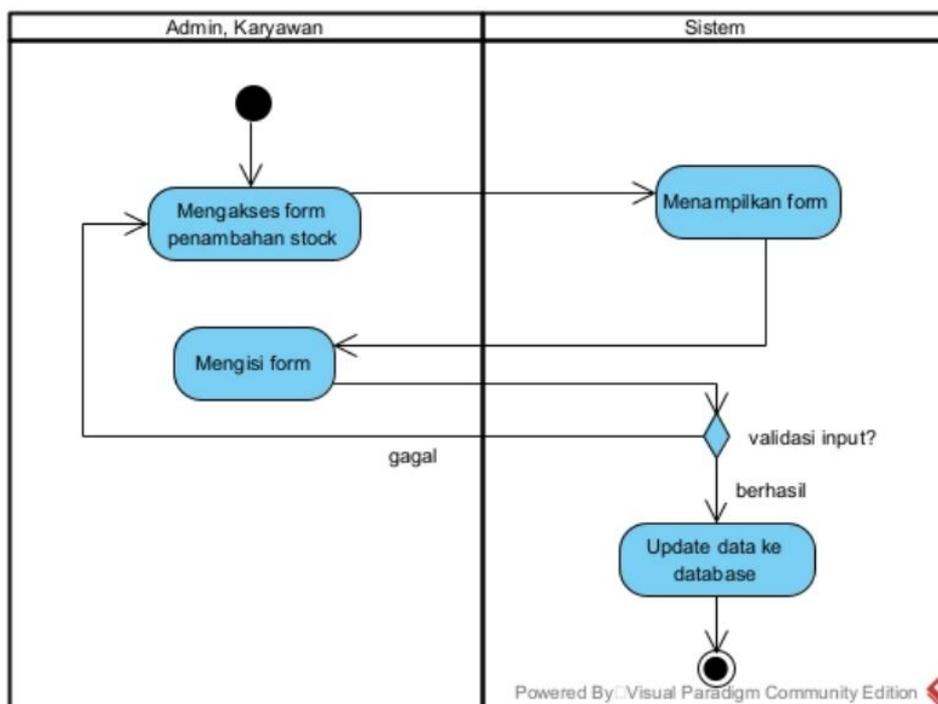


Lampiran 1.10 Activity Diagram UC-004

Kode Use Case	UC-004
Nama Use Case	Mengelola Data Perbaikan Transaksi
Aktor	Admin, Teknisi
Deskripsi	Admin dan teknisi bisa mengelola transaksi permintaan perawatan mesin
Relasi	-

Kondisi Awal	Memilih data perbaikan transaksi permintaan perawatan mesin	
Kondisi Akhir	Berhasil mengelola data perbaikan transaksi dari permintaan perawatan mesin serta memberikan catatan perawatan mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin / Teknisi memilih transaksi permintaan perawatan mesin	2. Memuat data transaksi permintaan
	3. Admin & teknisi melakukan pengelolaan data perbaikan pada transaksi	4. Mengubah status transaksi permintaan dan menyimpan data perbaikan ke database

Lampiran 1.11 Use Case UC-004

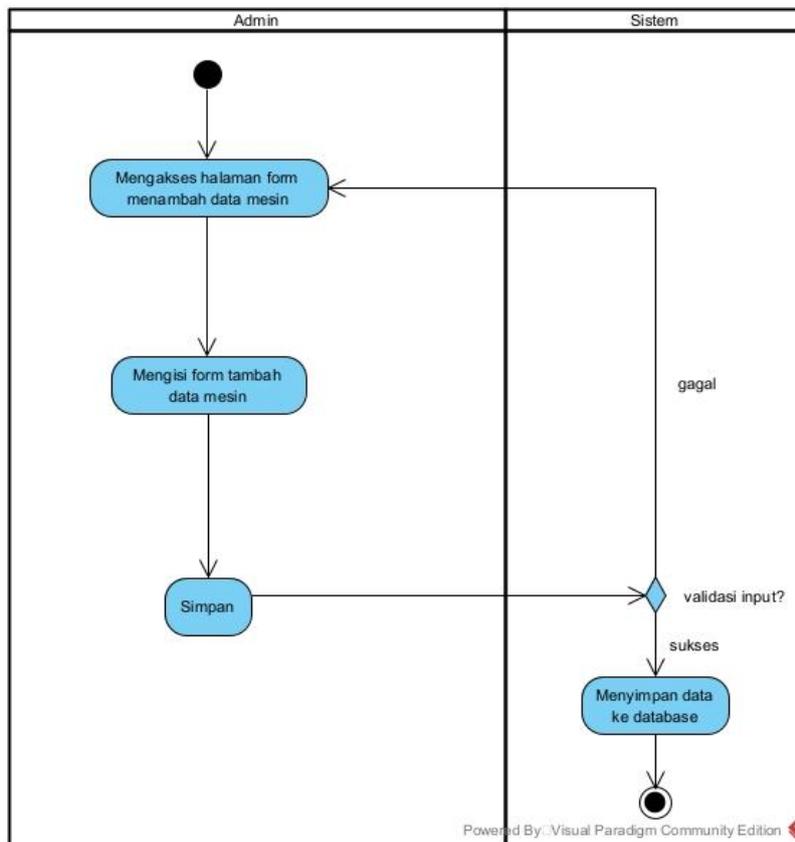


Lampiran 1. 12 Activity Diagram UC-005

Kode Use Case	UC-005
Nama Use Case	Menambah stock
Aktor	Admin, Teknisi

Deskripsi	Admin dan Teknisi bisa melakukan penambahan stock pada sparepart	
Relasi	-	
Kondisi Awal	Halaman form tambah stock	
Kondisi Akhir	Berhasil menambah data sparepart mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin dan Teknisi mengakses halaman form tambah stock	2. Memuat halaman form tambah stock
	3. Admin dan Teknisi mengisi form	4. Data input valid dan update data ke database
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin dan Teknisi mengakses halaman form tambah stock	2. Memuat halaman form tambah stock
	3. Admin dan Teknisi mengisi form	4. Data input tidak valid dan mereset halaman form tambah stock

Lampiran 1.13 Use Case UC-005

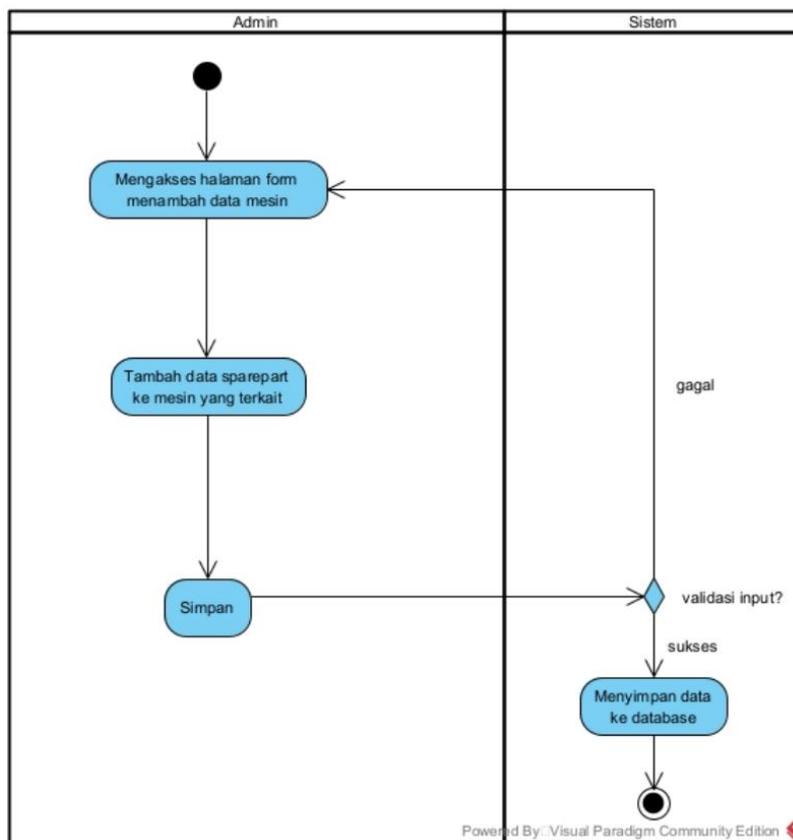


Lampiran 1.14 Activity Diagram UC-006

Kode Use Case	UC-006	
Nama Use Case	Menambah data mesin	
Aktor	Admin	
Deskripsi	Admin bisa menambah data mesin	
Relasi	UC-006	
Kondisi Awal	Berada di halaman form menambah data mesin	
Kondisi Akhir	Berhasil menambah data mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses form menambah data mesin	4. Validasi input form berhasil
	2. Admin mengisi form tambah data , serta menambah data bagian-bagian atau bahan mesin yang bersifat opsional	5. Menyimpan data ke database

	3. Admin submit untuk menyimpan data	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin mengakses form menambah data mesin	4. Validasi input form gagal
	2. Admin mengisi form tambah data, serta menambah data bagian-bagian atau bahan mesin yang bersifat opsional	5. Mengembalikan ke halaman form tambah data
	3. Admin submit untuk menyimpan data	

Lampiran 1.15 Use Case UC-006

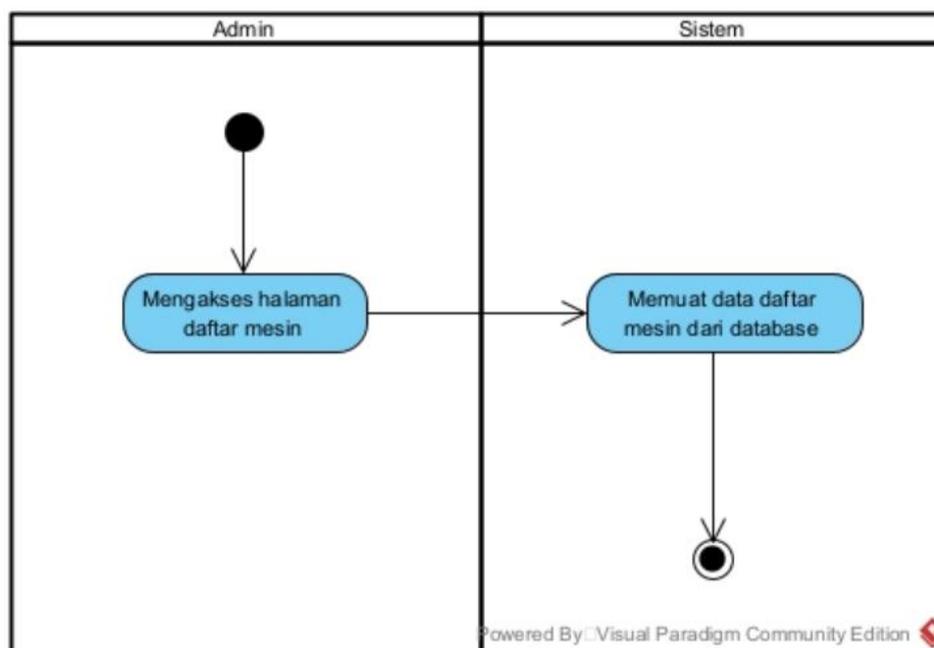


Lampiran 1.16 Activity Diagram UC-007

Kode Use Case	UC-007
Nama Use Case	Tambah sparepart mesin
Aktor	Admin

Deskripsi	Admin bisa menambah data bagian-bagain atau bahan mesin	
Relasi	UC-007	
Kondisi Awal	Berada di halaman form menambah data mesin	
Kondisi Akhir	Berhasil menambah data bagian-bagain atau bahan mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses form menambah data mesin	4. Validasi input form "BERHASIL"
	2. Admin mengisi form serta menambah data ( <i>sparepart</i> ) bagian-bagian atau bahan mesin yang bersifat opsional	5. Menyimpan data ke database
	3. Admin submit untuk menyimpan data	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin mengakses form menambah data mesin	4. Validasi input form "GAGAL"
	2. Admin mengisi form serta menambah data ( <i>sparepart</i> ) bagian-bagian atau bahan mesin yang bersifat opsional	5. Mengembalikan ke halaman form tambah data
	3. Admin submit untuk menyimpan data	

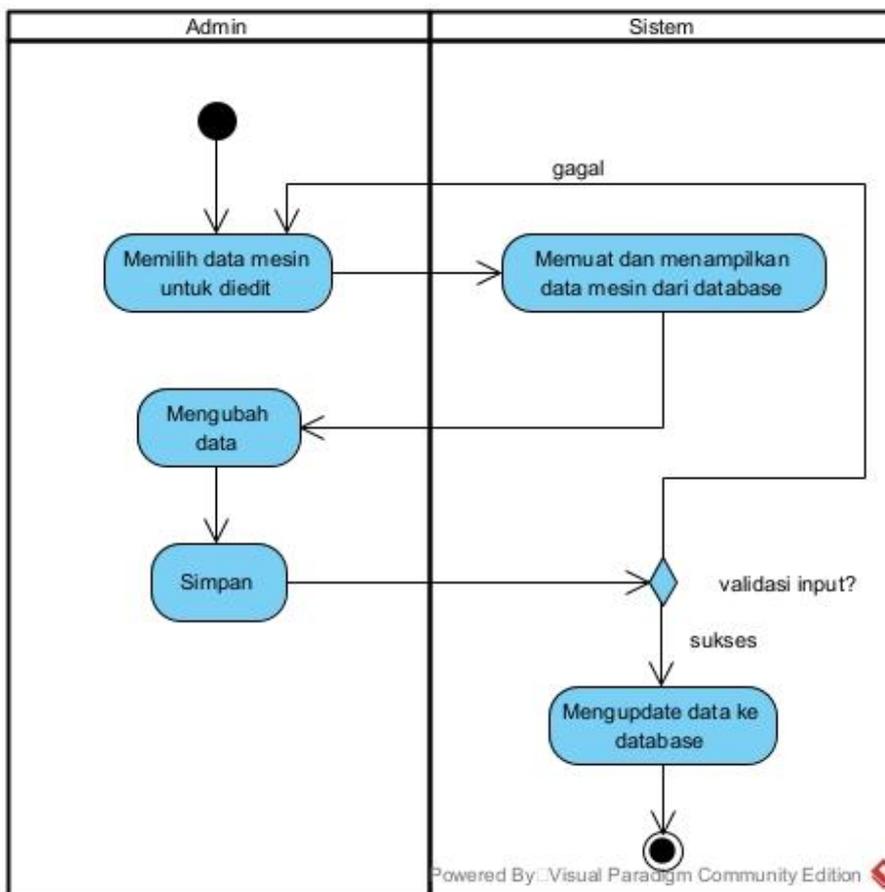
Lampiran 1.17 Use Case UC-007



Lampiran 1.18 Activity Diagram UC-008

Kode Use Case	UC-008	
Nama Use Case	Melihat daftar mesin	
Aktor	Admin	
Deskripsi	Admin bisa melihat daftar mesin pada halaman	
Relasi	-	
Kondisi Awal	Halaman daftar mesin	
Kondisi Akhir	Berhasil memuat daftar mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses halaman daftar mesin	2. Memuat data daftar mesin dari database

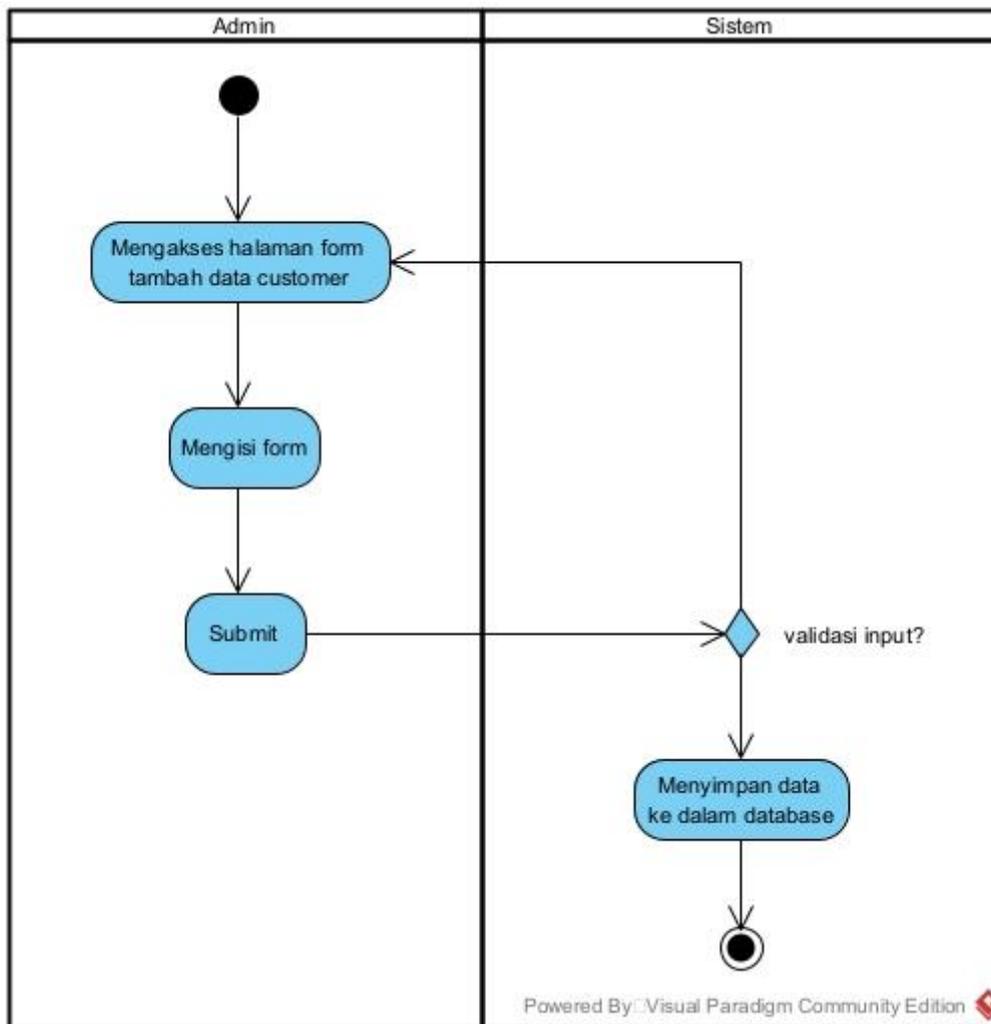
Lampiran 1.19 Use Case UC-008



Lampiran 1.20 Activity Diagram UC-009

Kode Use Case	UC-009	
Nama Use Case	Edit data mesin	
Aktor	Admin	
Deskripsi	Admin bisa mengubah data mesin	
Relasi	-	
Kondisi Awal	Memilih data mesin untuk diedit	
Kondisi Akhir	Berhasil mengubah data mesin	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin memilih data mesin untuk diedit	4. Validasi input form berhasil
	2. Admin mengubah data mesin	5. Menyimpan data ke database
	3. Admin melakukan submit untuk menyimpan data	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin memilih data mesin untuk diedit	4. Validasi input form gagal
	2. Admin mengubah data mesin	5. Mengembalikan ke halaman daftar mesin untuk memilih data mesin yang diedit
	3. Admin melakukan submit untuk menyimpan data	

Lampiran 1.21 Use Case UC-009

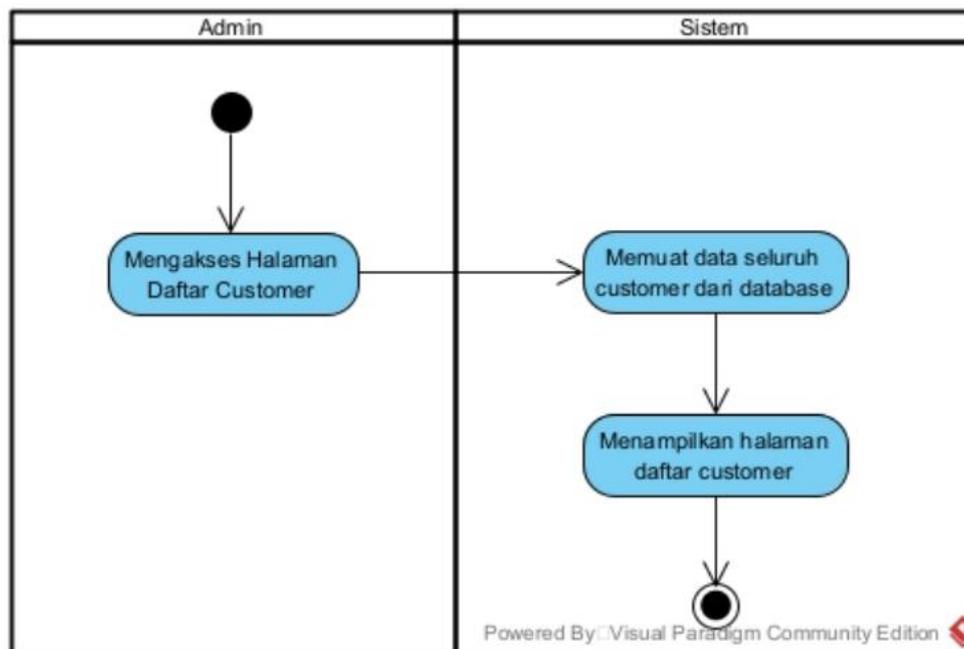


Lampiran 1.22 Activity Diagram UC-010

Kode Use Case	UC-010	
Nama Use Case	Menambah Data Customer	
Aktor	Admin	
Deskripsi	Admin bisa menambah data customer	
Relasi	-	
Kondisi Awal	Berada di halaman menambah data customer	
Kondisi Akhir	Berhasil menambah data customer	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses halaman daftar customer	4. Validasi input form berhasil
	2. Admin memilih data customer yang akan diedit	5. Mengupdate data ke database

	3. Admin mengubah data form customer dan melakukan submit data	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin mengakses halaman daftar customer	4. Validasi Input Form Gagal
	2. Admin memilih data customer yang akan diedit	
	3. Admin mengubah data form customer dan melakukan submit data	
	5. Data tidak berhasil disimpan. Kemudian, sistem mengarahkan kembali ke halaman semula yaitu halaman daftar customer	

Lampiran 1.23 Use Case UC-010

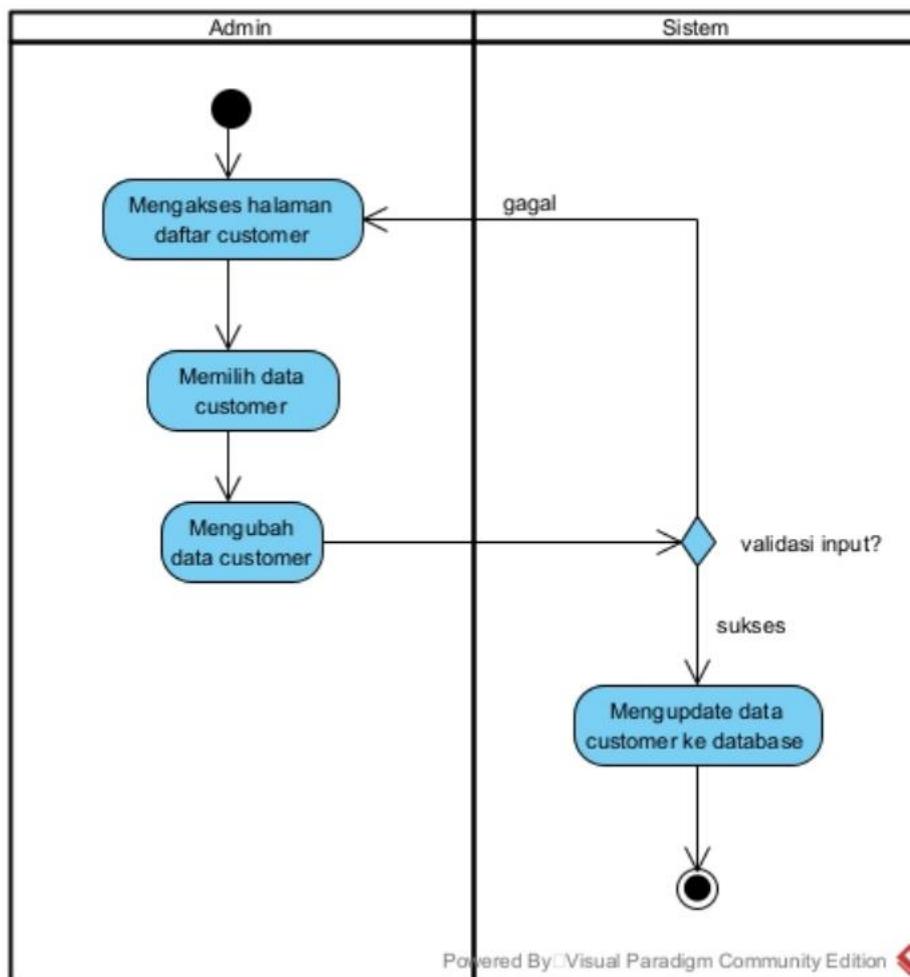


Lampiran 1.24 Activity Diagram UC-011

Kode Use Case	UC-011
Nama Use Case	Menampilkan Daftar Customer
Aktor	Admin
Deskripsi	Admin bisa melihat daftar customer
Relasi	-
Kondisi Awal	Berada di halaman daftar customer

Kondisi Akhir	Berhasil menampilkan seluruh daftar customer	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses form halaman daftar customer	2. Memuat dan mengambil data dari database
		3. Menampilkan halaman daftar customer

Lampiran 1.25 Use Case UC-011

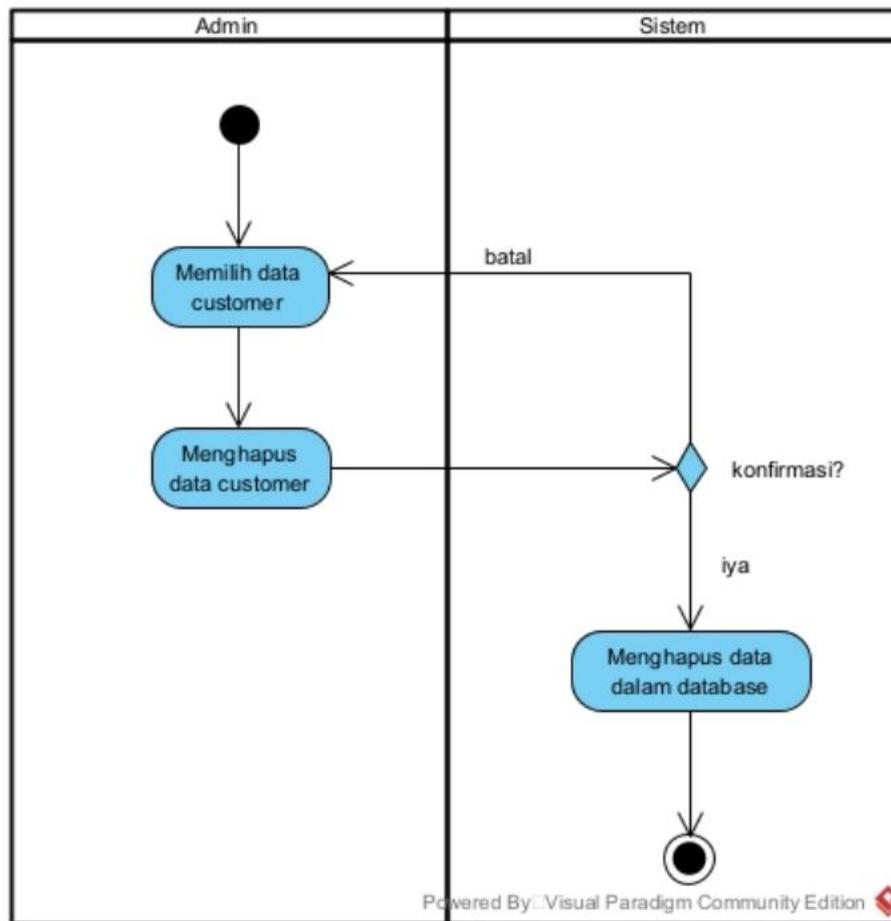


Lampiran 1.26 Activity Diagram UC-012

Kode Use Case	UC-012
Nama Use Case	Edit Data Customer
Aktor	Admin
Deskripsi	Admin bisa mengubah data customer
Relasi	-

Kondisi Awal	Berada di halaman daftar customer	
Kondisi Akhir	Berhasil mengubah data customer	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses halaman daftar customer	4. Validasi input form berhasil
	2. Admin memilih data mengubah data customer yang diedit	5. Mengupdate data ke database
	3. Admin melakukan submit data	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin mengakses halaman daftar customer	4. Validasi input form gagal dan tidak berhasil
	2. Admin memilih data mengubah data customer yang diedit	
	3. Admin melakukan submit data	
	5. Data tidak berhasil disimpan. Kemudian, sistem mengarahkan kembali ke halaman semula yaitu halaman daftar customer	

Lampiran 1.27 Use Case UC-012

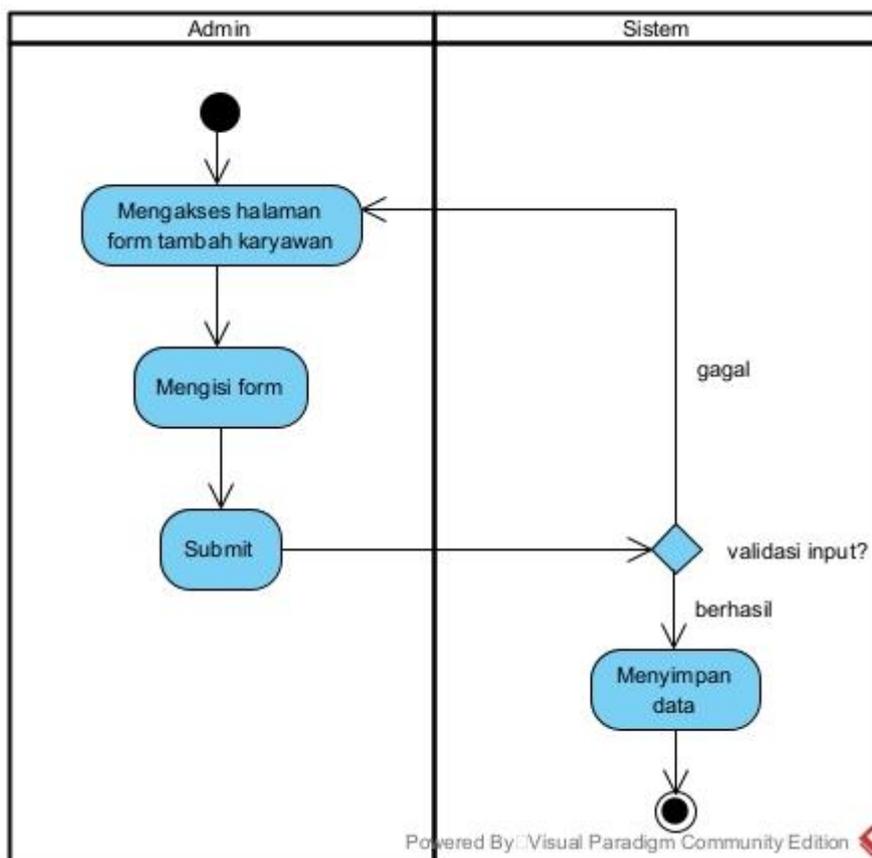


Lampiran 1.28 Activity Diagram UC-013

Kode Use Case	UC-013	
Nama Use Case	Menghapus Data Customer	
Aktor	Admin	
Deskripsi	Admin bisa menghapus data customer	
Relasi	-	
Kondisi Awal	Memilih data customer	
Kondisi Akhir	Berhasil menghapus data customer	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses halaman customer dan memilih data customer yang dihapus	3. Data customer dihapus dari database

	2. Admin melakukan konfirmasi “YA” pada halaman konfirmasi	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin mengakses halaman customer dan memilih data customer yang dihapus	3. Mengarahkan kembali ke halaman semula yaitu halaman daftar <i>customer</i> .
	2. Admin melakukan konfirmasi “BATAL” pada halaman konfirmasi	

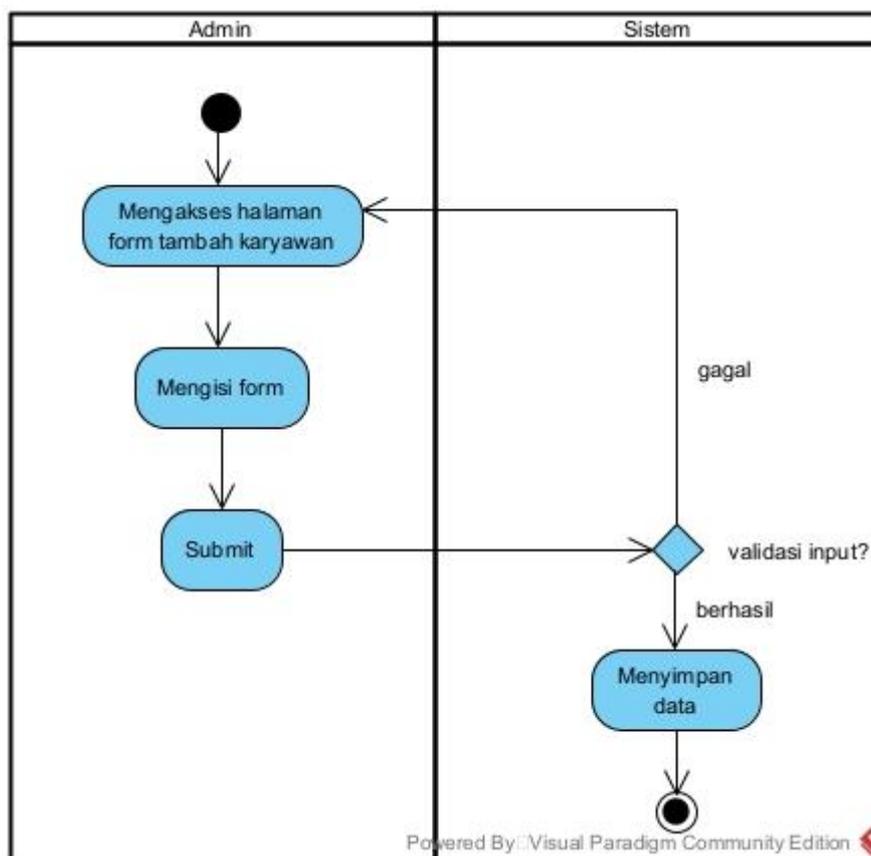
Lampiran 1.29 Use Case UC-013



Lampiran 1.30 Activity Diagram UC-014

Kode Use Case	UC-014	
Nama Use Case	Menambah Teknisi	
Aktor	Admin	
Deskripsi	Admin bisa menambah Teknisi	
Relasi	-	
Kondisi Awal	Mengakses halaman tambah Teknisi	
Kondisi Akhir	Berhasil menambah Teknisi	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses halaman form tambah Teknisi	3. Validasi input “berhasil”
	2. Mengisi form tambah data Teknisi dan melakukan submit	4. Menyimpan data Teknisi customer
Alur Kejadian Tidak Normal	1. Admin mengakses halaman form tambah Teknisi	3. Validasi input “gagal”
	2. Mengisi form tambah data Teknisi dan melakukan submit	4. Sistem mereset form tambah data / memberi tahu invalid input

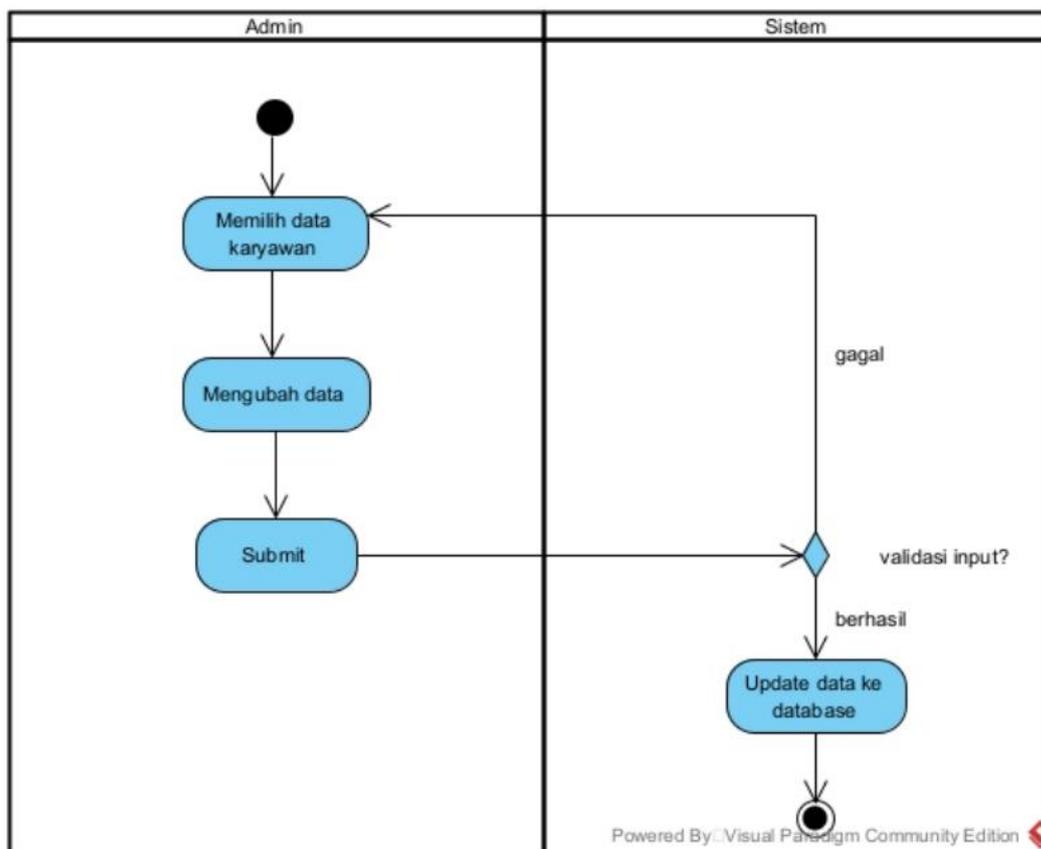
Lampiran 1.31 Use Case UC-014



Lampiran 1.32 Activity Diagram UC-015

Kode Use Case	UC-015	
Nama Use Case	Melihat Daftar Teknisi	
Aktor	Admin	
Deskripsi	Admin bisa melihat daftar Teknisi	
Relasi	-	
Kondisi Awal	Halaman daftar Teknisi	
Kondisi Akhir	Berhasil memuat halaman daftar Teknisi	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin mengakses halaman daftar Teknisi	2. Mengambil data dari database
		3. Memuat halaman dan menampilkan halaman

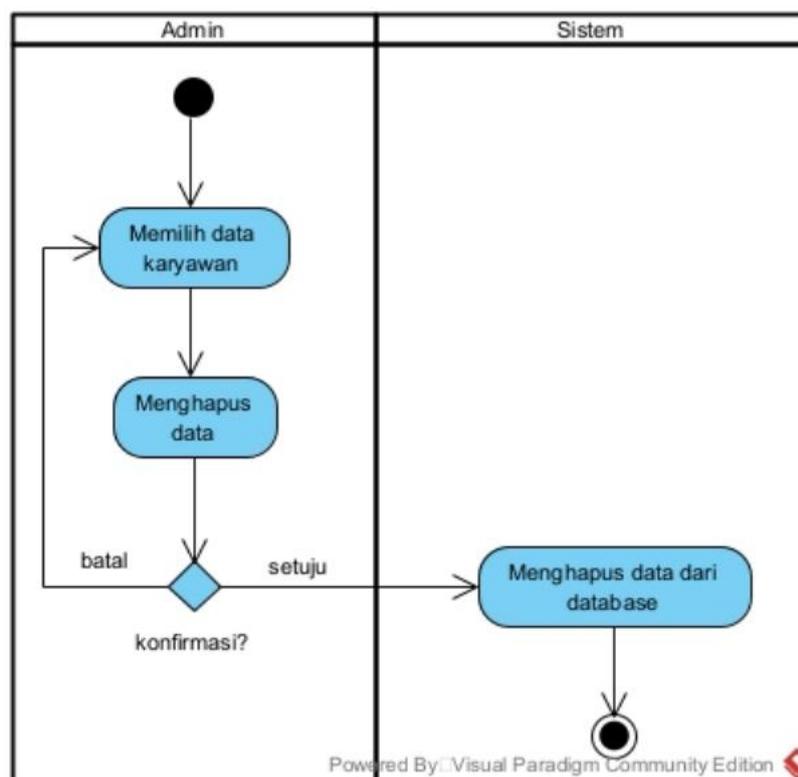
Lampiran 1.33 Use Case UC-015



Lampiran 1.34 Activity Diagram UC-016

Kode Use Case	UC-016	
Nama Use Case	Edit Teknisi	
Aktor	Admin	
Deskripsi	Admin bisa edit Teknisi	
Relasi	-	
Kondisi Awal	Mengakses halaman edit Teknisi	
Kondisi Akhir	Berhasil mengubah data Teknisi	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin memilih data Teknisi	3. Validasi input data "berhasil"
	2. Mengubah data Teknisi dan melakukan konfirmasi	4. Data berhasil di update ke database
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin memilih data Teknisi	3. Validasi input data "gagal"
	2. Mengubah data Teknisi dan melakukan konfirmasi	4. Mengambilkan ke halaman daftar Teknisi untuk dipilih ulang

Lampiran 1.35 Use Case UC-016



Lampiran 1.36 Activity Diagram UC-017

Kode Use Case	UC-017	
Nama Use Case	Menghapus Teknisi	
Aktor	Admin	
Deskripsi	Admin bisa menghapus data Teknisi	
Relasi	-	
Kondisi Awal	Mengakses halaman daftar Teknisi	
Kondisi Akhir	Berhasil menghapus data Teknisi	
Alur Kejadian Normal	Aktor	Sistem
	1. Admin memilih data Teknisi	4. Data dihapus dari database
	2. Admin Menghapus data Teknisi	
	3. Konfirmasi “SETUJU” menghapus data	
Alur Kejadian Tidak Normal	Aktor	Sistem
	1. Admin memilih data Teknisi	4. Mengambilakan ke halaman daftar Teknisi untuk dipilih kembali
	2. Menghapus data Teknisi	
	3. Konfirmasi “BAATAL” menghapus data	

Lampiran 1.37 Use Case UC-017

Nama Kolom	Tipe Data	Keterangan
Id	char	<i>Primary Key</i>
Email	Varchar	Email Pengguna ( <i>Primary Key</i> )
Password	Varchar	Password Pengguna

Lampiran 1.38 Tabel Data User

Nama Kolom	Tipe Data	Keterangan
Id	char	<i>Primary Key</i>
UserId	char	Tabel User ( <i>Foreign Key</i> )
Name	Varchar	Nama Pengguna Teknisi
JobRole	Varchar	Role jenis pekerjaan (Admin / Karyawan)
Handphone	Varchar	No.HP (Admin / Karyawan)

**Lampiran 1.39** Tabel Data Employees

Nama Kolom	Tipe Data	Keterangan
<b>Id</b>	char	<i>Primary Key</i>
UserId	char	Tabel User ( <i>Foreign Key</i> )
Name	Varchar	Nama Pengguna <i>customer</i>
Phone	varchar	Nomor HP Pengguna <i>customer</i>
Email	Varchar	Email Customer
Address	varchar	Alamat <i>Customer</i>
City	varchar	Kota Customer

**Lampiran 1.40** Tabel Data Customer

Nama Kolom	Tipe Data	Keterangan
<b>ID</b>	bigint	<i>Primary Key</i>
Title	Varchar	Judul catatan
Description	varchar	Deskripsi Catatan

**Lampiran 1.41** Tabel Data Note

Nama Kolom	Tipe Data	Keterangan
<b>ID</b>	bigint	<i>Primary Key</i>
Name	varchar	Nama Mesin
Brand	varchar	Nama Merek Mesin
Length	Varchar	Panjang Mesin
Width	Varchar	Lebar Mesin
Height	Varchar	Tinggi Mesin
Weight	Varchar	Berat Mesin
Class	Varchar	Tipe Kelas Mesin
Active	tinyint	Status Aktif / Nonaktif Mesin

**Lampiran 1.42** Tabel Data Machines

Nama Kolom	Tipe Data	Keterangan
<b>ID</b>	bigint	<i>Primary Key</i>
Name	varchar	Nama <i>Sparepart</i> Mesin
Brand	varchar	Merek <i>Sparepart</i> Mesin
Type	varchar	Tipe <i>Sparepart</i> Mesin
Qty	bigint	Jumlah stock <i>spareparts</i>
Min_Qty	bigint	Minimal stock <i>spareparts</i>
Price	varchar	Harga <i>Spareparts</i>

**Lampiran 1.43** Tabel Data Spareparts

Nama Kolom	Tipe Data	Keterangan
<b>ID</b>	char	<i>Primary Key</i>
CustomerID	char	<i>Foreign Key</i>
NoteID	bigint	<i>Foreign Key</i>
EmployeeAdminID	char	<i>Foreign Key</i>
TotalPrice	varchar	Jumlah pembayaran transaksi
Status	varchar	Status transaksi
DateRequest	varchar	Tanggal dibuat transaksi
PaymentDate	varchar	Tanggal dinyatakan lunas

		pada transaksi
--	--	----------------

Lampiran 1. 44 Tabel Data Transactions

Nama Kolom	Tipe Data	Keterangan
ID	bigint	Primary Key
TransactionID	Char	Foreign Key
EmployeeID	char	Foreign Key
MachineID	bigint	Foreign Key

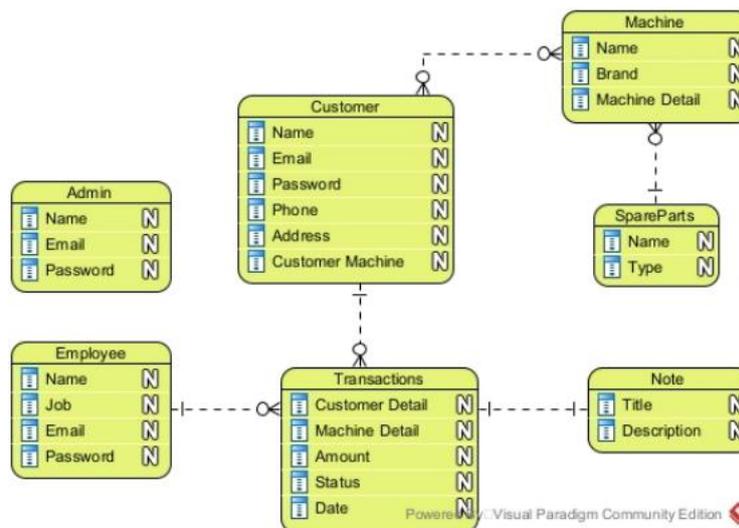
Lampiran 1. 45 Tabel Data Maintenance Detail

Nama Kolom	Tipe Data	Keterangan
ID	integer	Primary Key
SparePartsID	integer	Foreign Key
TransactionDetailID	integer	Foreign Key
UnitPrice	varchar	Harga Satuan Spareparts
Qty	bigint	Jumlah Sparepart Yang diambil

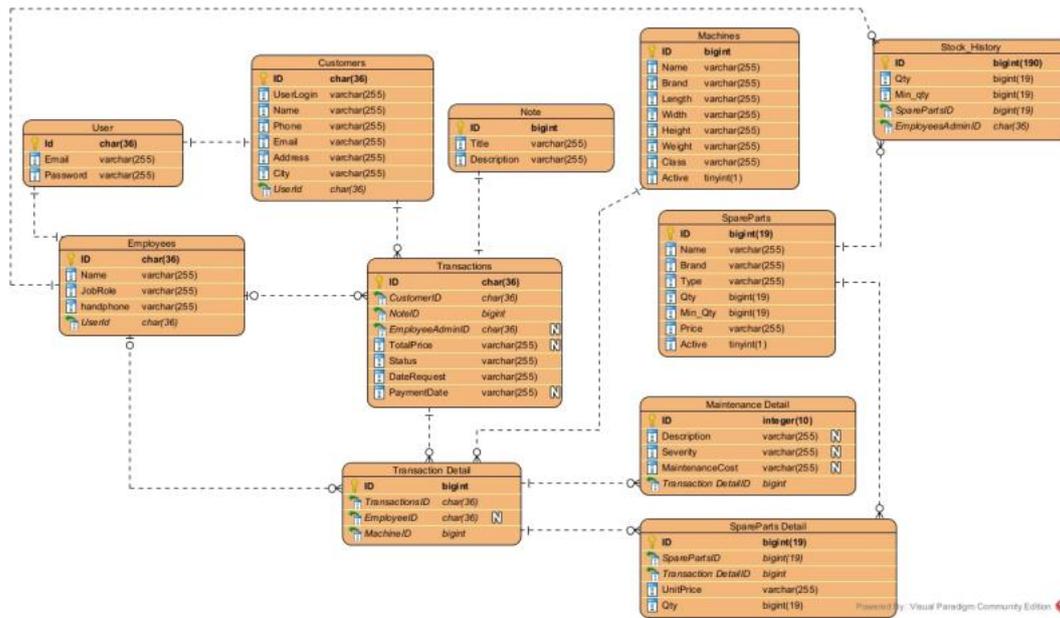
Lampiran 1.46 Tabel Data Spareparts Detail

Nama Kolom	Tipe Data	Keterangan
ID	bigint	Primary Key
Qty	bigint	Jumlah History Spareparts
Min_qty	bigint	Jumlah History Minimal Spareparts
SparePartsID	Bigint	Foreign Key
EmployeesAdminID	char	Foreign Key

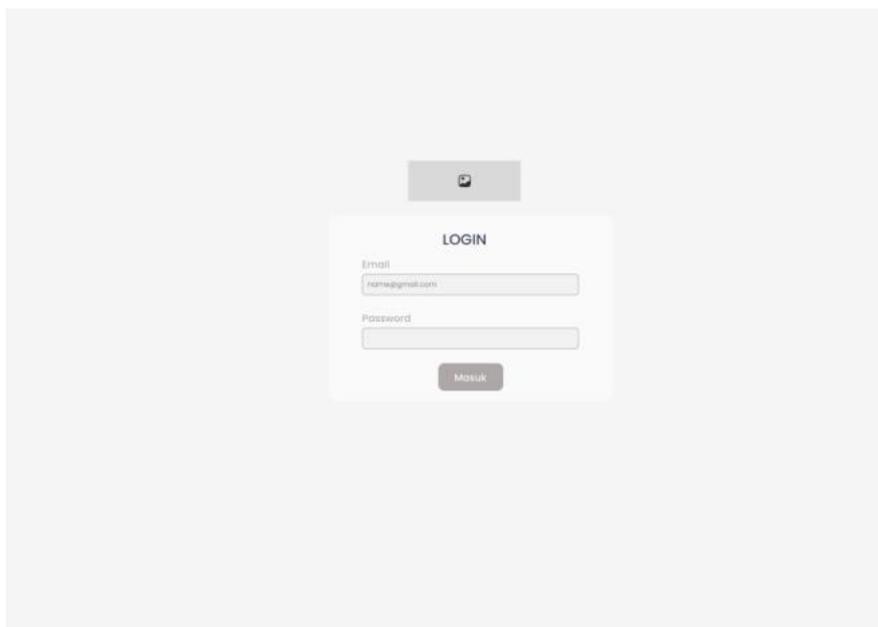
Lampiran 1.47 Tabel Data Stock History



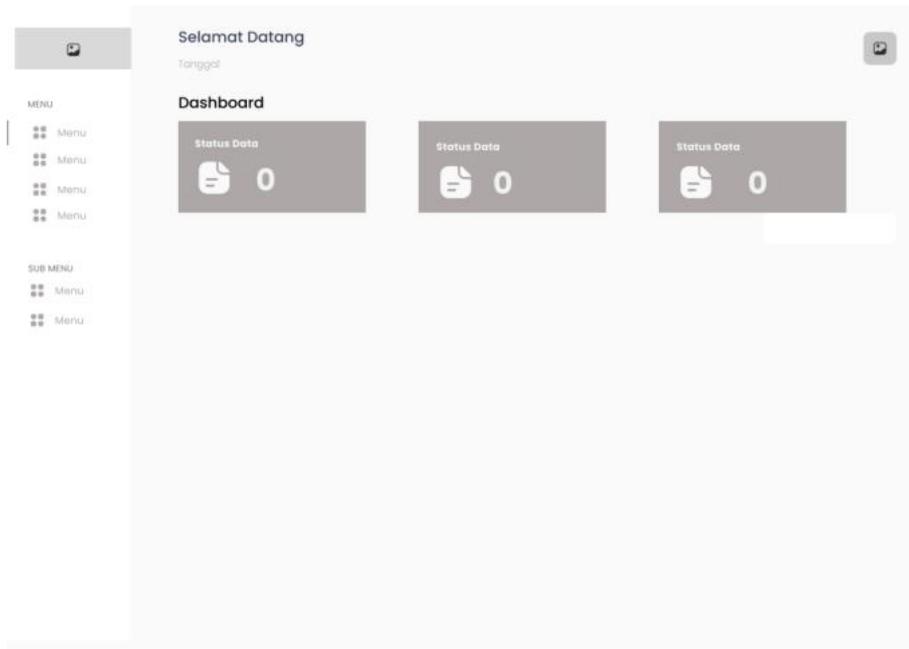
Lampiran 1.48 Conceptual Data Model



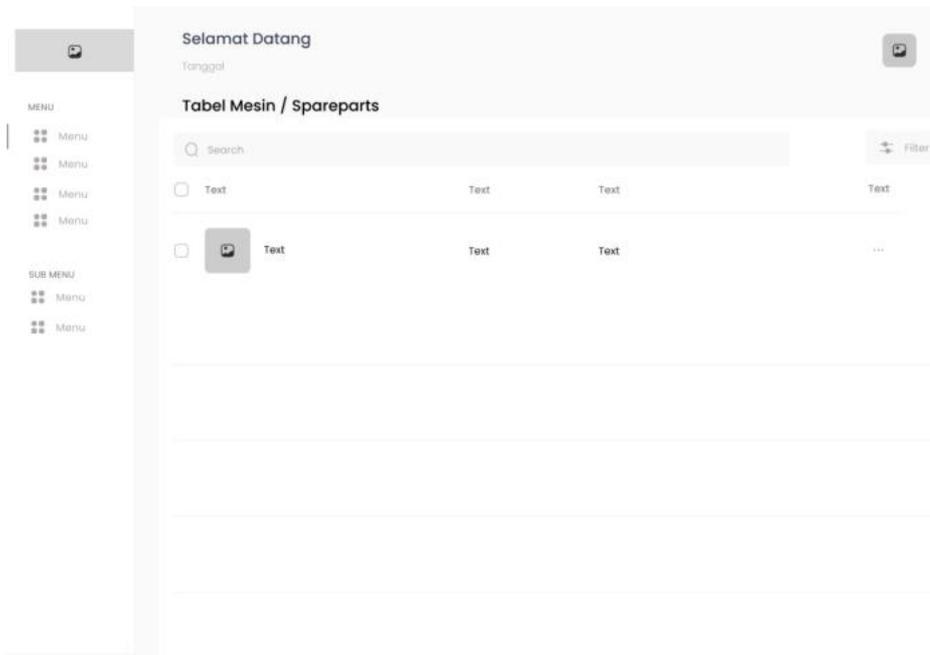
Lampiran 1.49 Physical Data Model



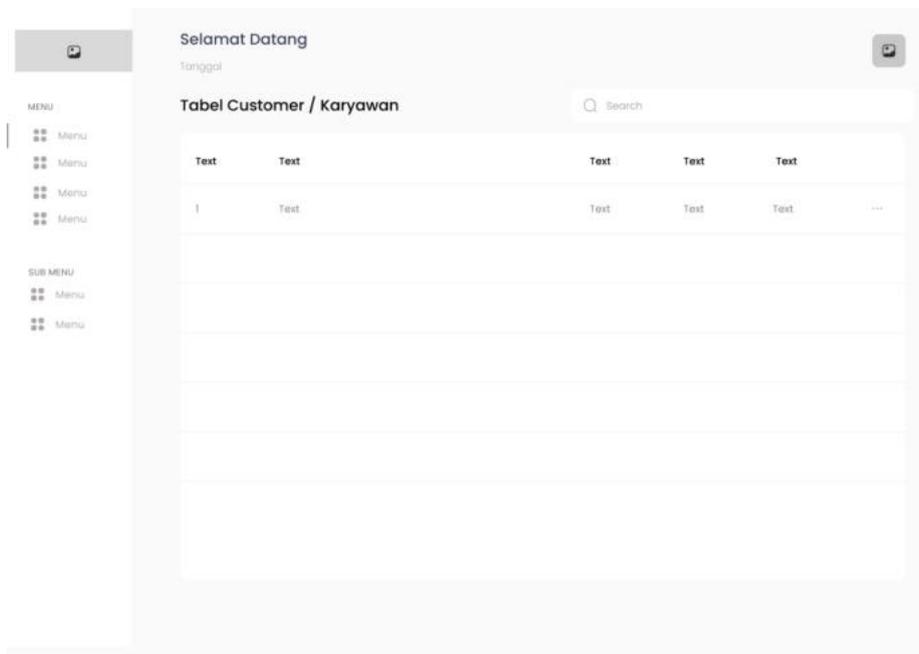
Lampiran 1.50 Perancangan Antarmuka (Login)



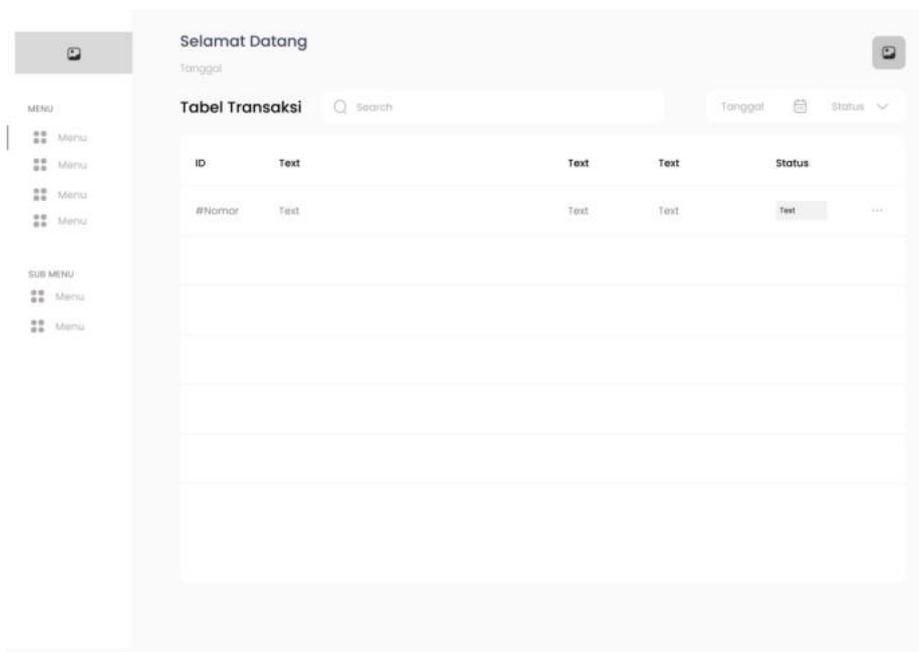
Lampiran 1.51 Perancangan Antarmuka (Dashboard)



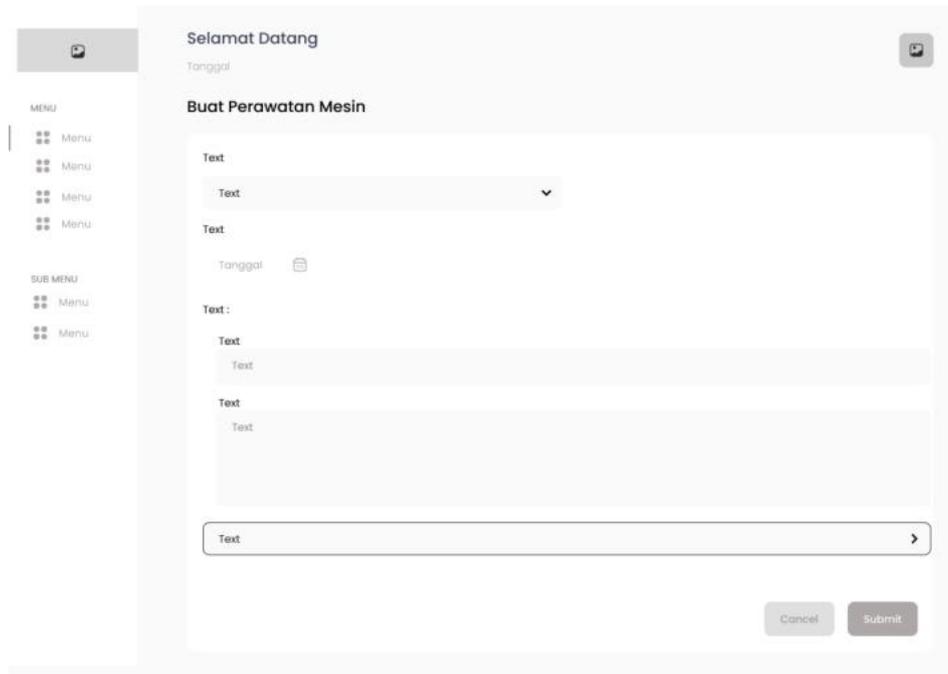
Lampiran 1.52 Perancangan Antarmuka (Tabel Mesin / Spareparts)



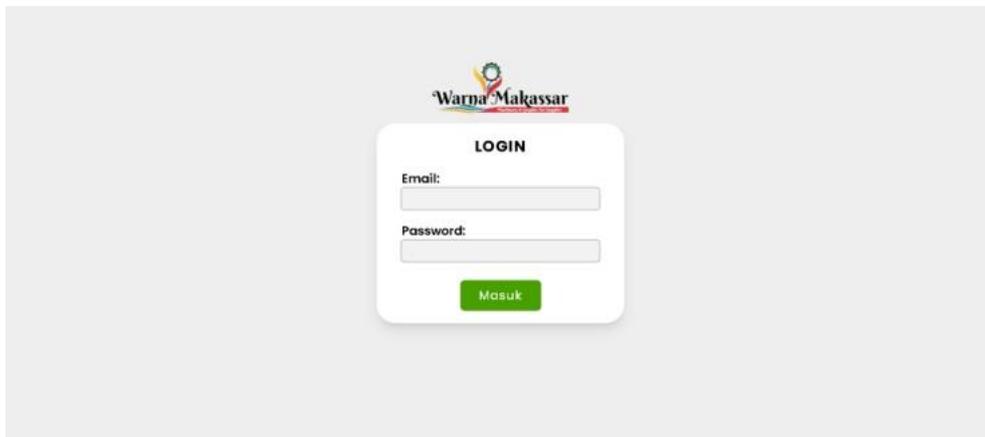
Lampiran 1.53 Perancangan Antarmuka (Tabel Customer / Karyawan)



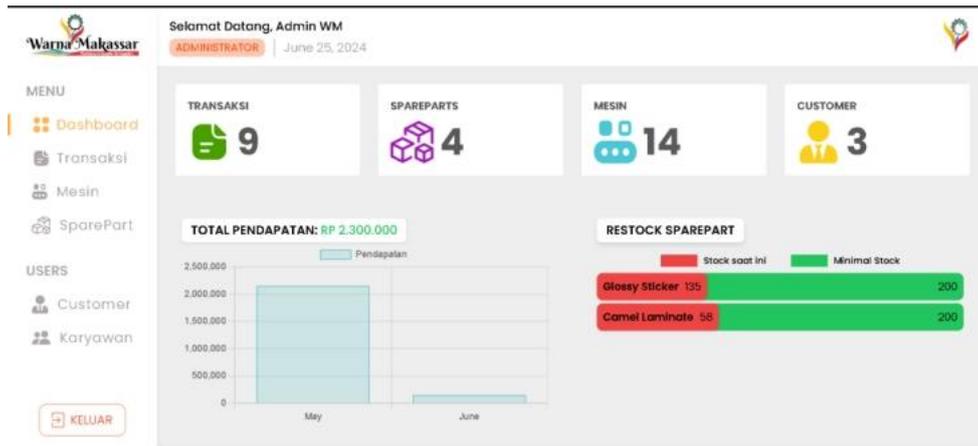
Lampiran 1.54 Perancangan Antarmuka (Tabel Transaksi)



Lampiran 1.55 Perancangan Antarmuka (Buat Perawatan Mesin)



Lampiran 1.56 Halaman Login



Lampiran 1.57 Halaman Dashboard (Admin)

**Daftar Transaksi**

Carilah berdasarkan customer...

ID	Customer	Nama Mesin	Status	Action
#16a91d32	RadikaNih	Machine Press 2015 GNK	Menunggu	
#68065d8a	RadikaNih	Blue Red SMAKDas	Diproses	
#E215689a	RadikaNih	Machine Printing XR-80 Type GT	Diproses	
#7a8a81a7	RadikaNih	Blues Winux Super	Belum Terbayar	
#8acd74f5	RadikaNih	Press Camel Super	Diproses	

Lampiran 1.58 Halaman Daftar Transaksi (Admin)

**Buat Perawatan**

← Kembali

Nama Mesin:

**SPAREPART MESIN**

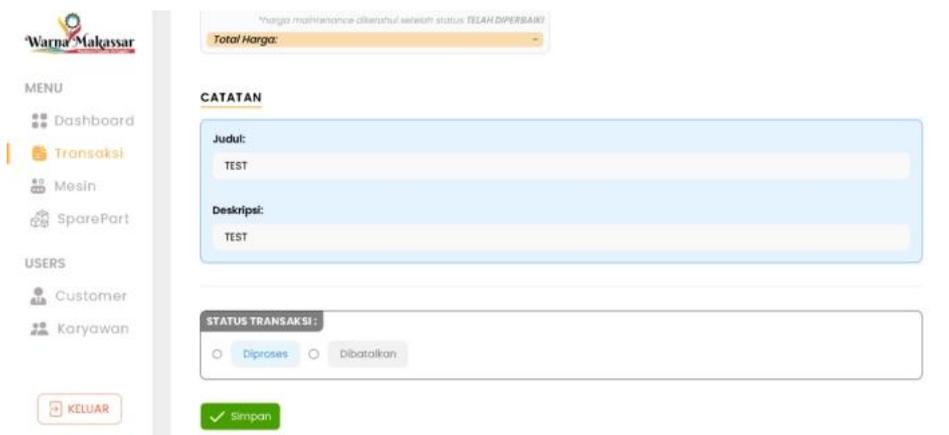
Nama Sparepart:  Jumlah:

Total Harga Sparepart: Rp 0

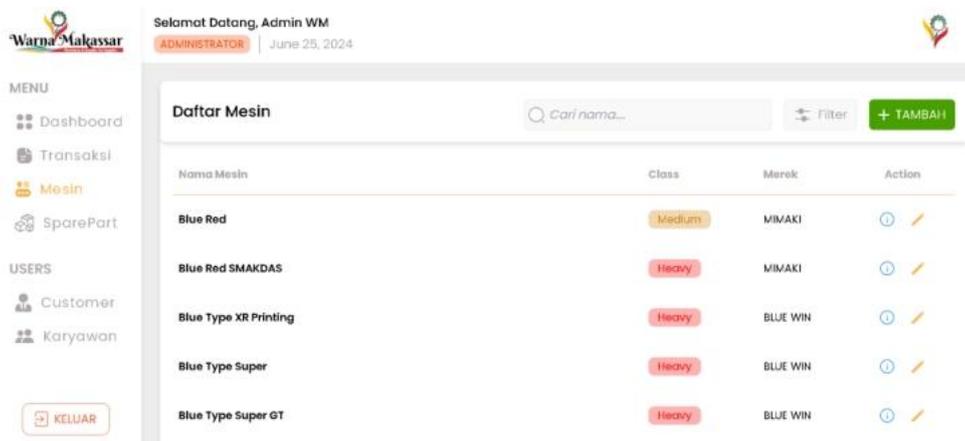
**CATATAN**

Judul Catatan:

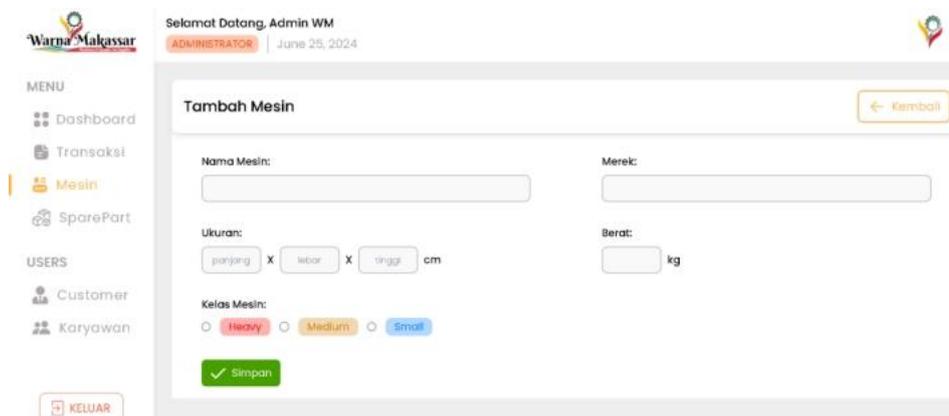
Lampiran 1.59 Halaman Tambah Transaksi (Admin)



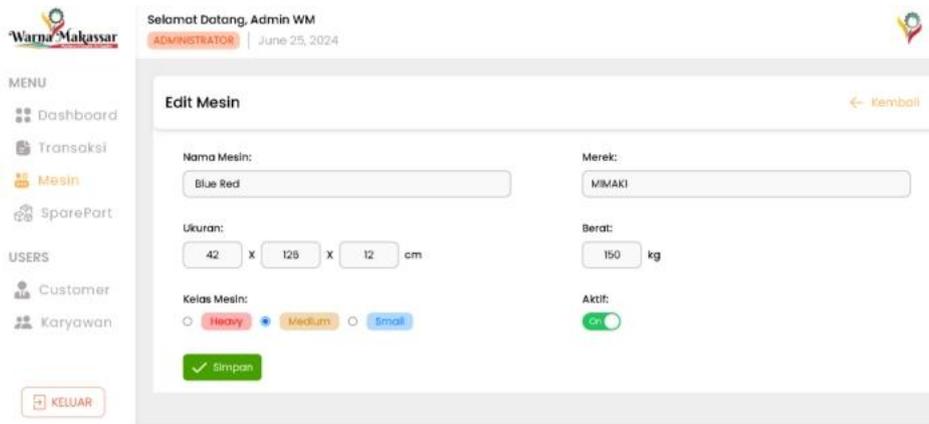
Lampiran 1.60 Halaman Update *Transaksi* (Admin)



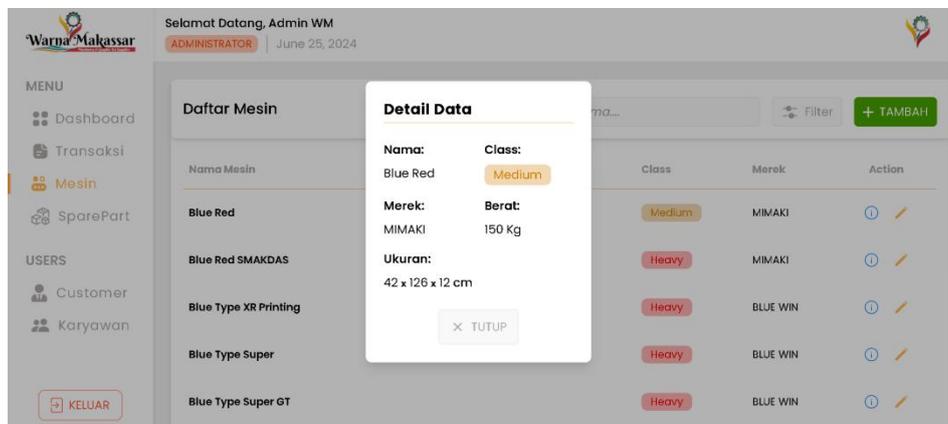
Lampiran 1.61 Halaman Daftar *Mesin* (Admin)



Lampiran 1.62 Halaman Tambah *Mesin* (Admin)



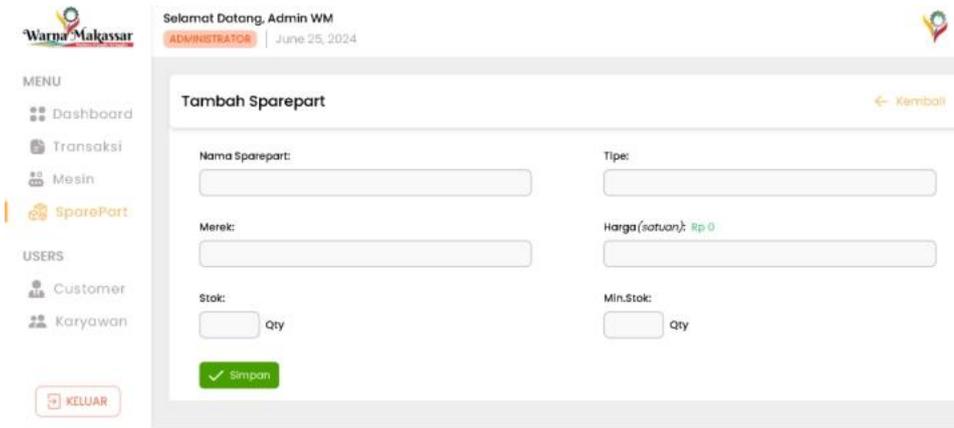
Lampiran 1.63 Halaman Edit Mesin (Admin)



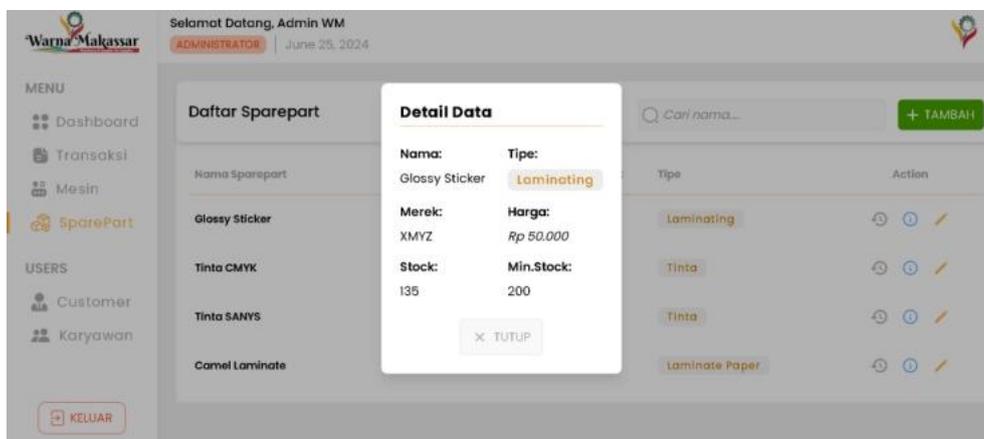
Lampiran 1.64 Halaman Detail Mesin (Admin)



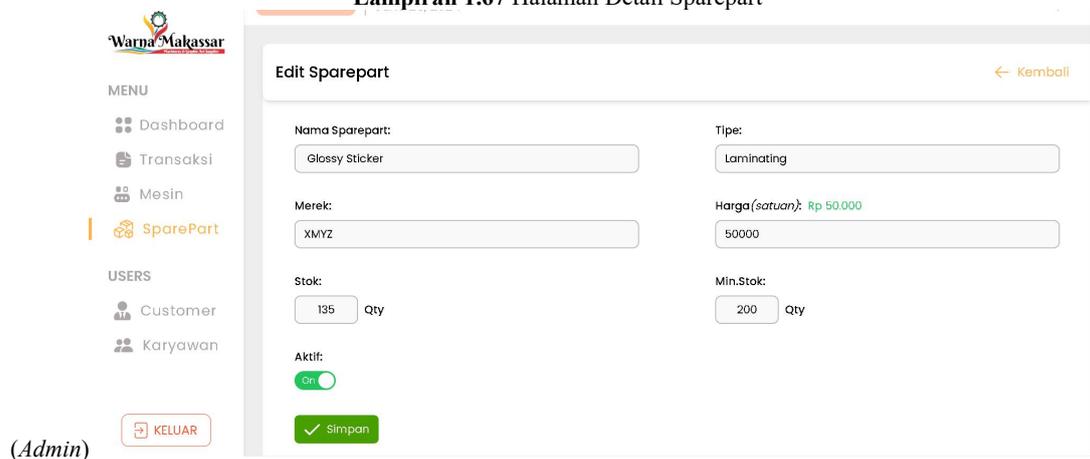
Lampiran 1.65 Halaman Detail Sparepart (Admin)



Lampiran 1.66 Halaman Tambah Sparepart (Admin)

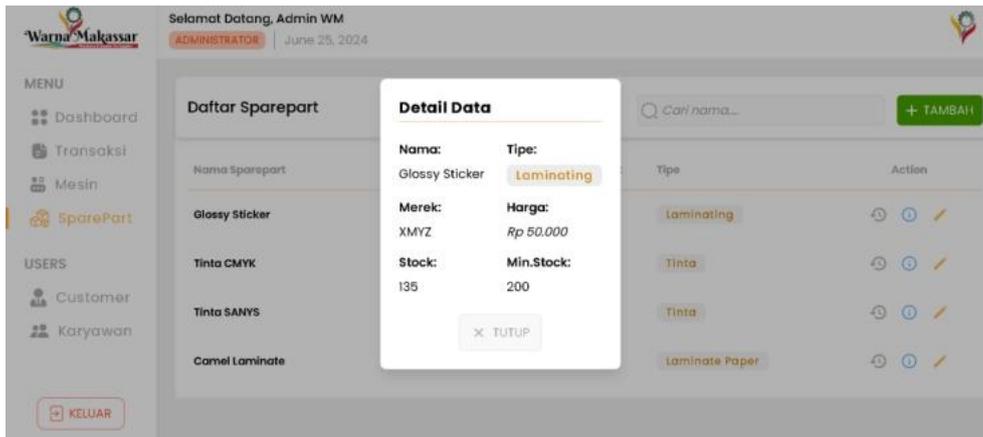


Lampiran 1.67 Halaman Detail Sparepart

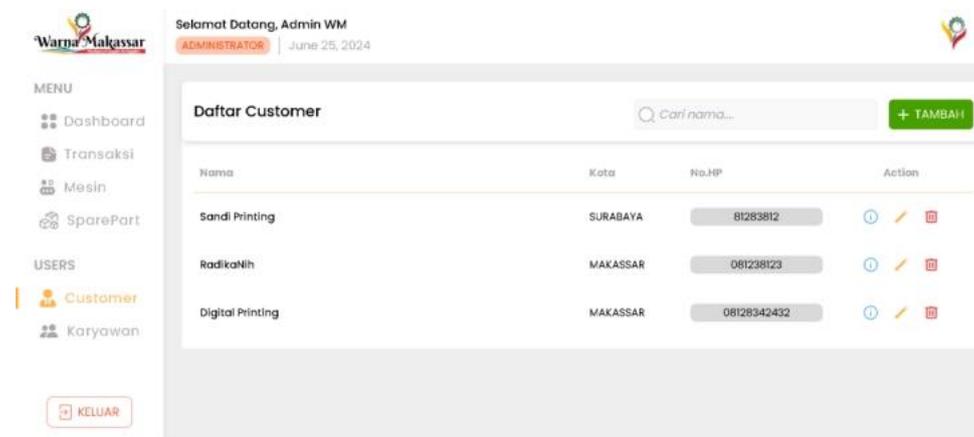


(Admin)

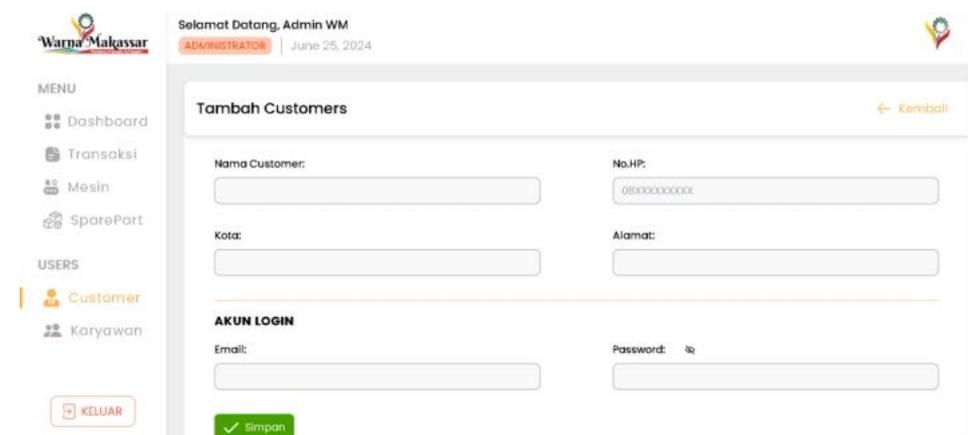
Lampiran 1.68 Halaman Edit Sparepart (Admin)



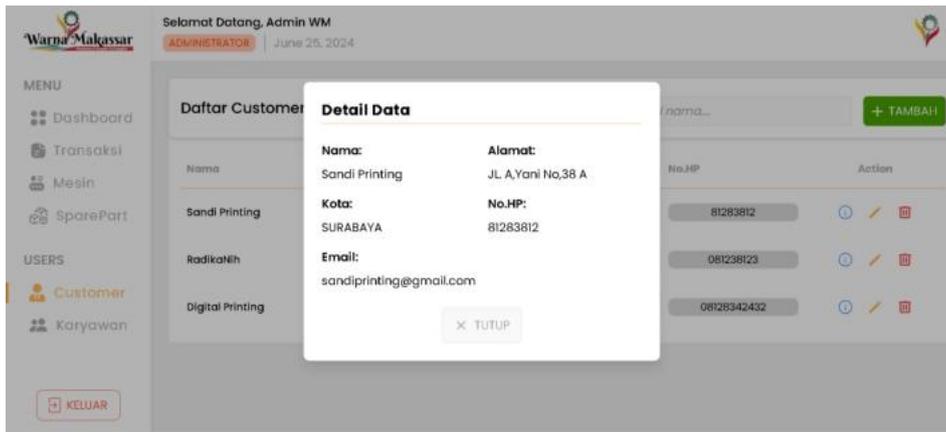
Lampiran 1.69 Halaman Riwayat Stok Sparepart (Admin)



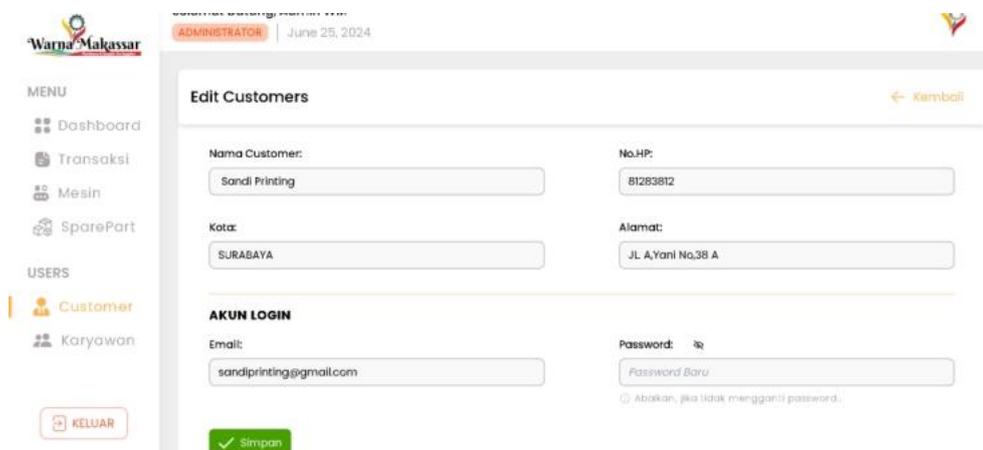
Lampiran 1.70 Halaman Daftar Customer (Admin)



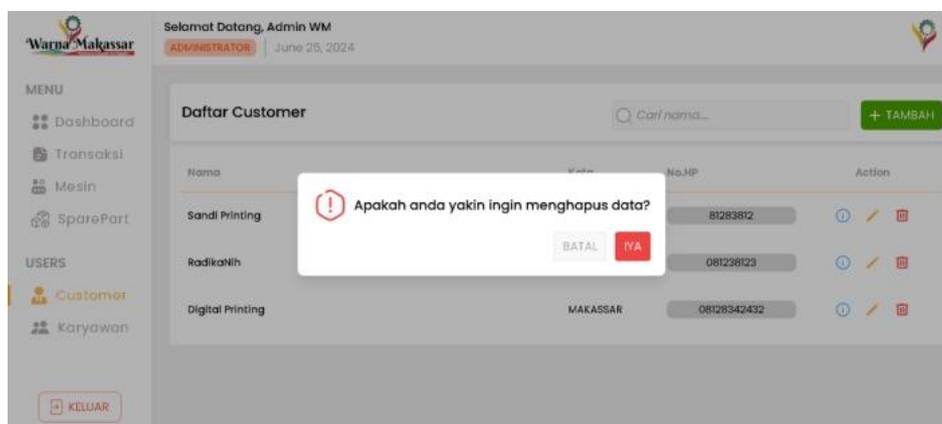
Lampiran 1.71 Halaman Tambah Customer (Admin)



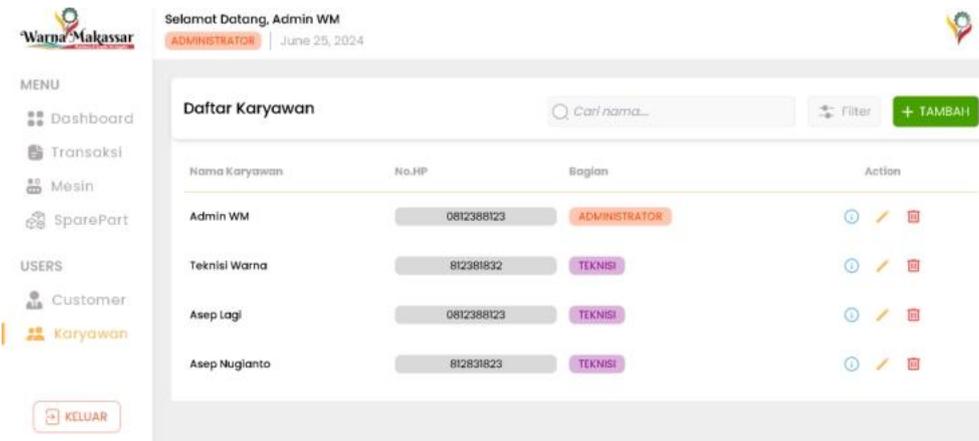
Lampiran 1.72 Halaman Detail Customer (Admin)



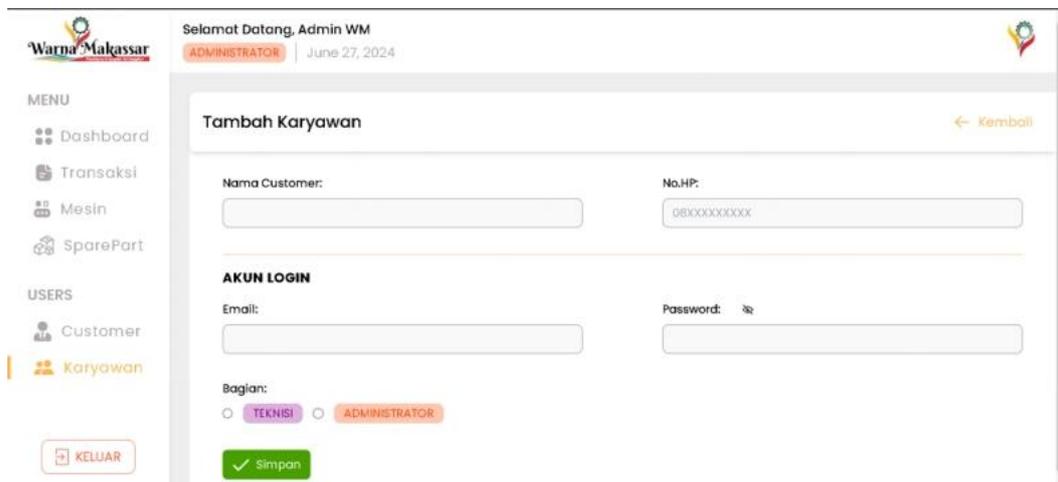
Lampiran 1.73 Halaman Edit Customer (Admin)



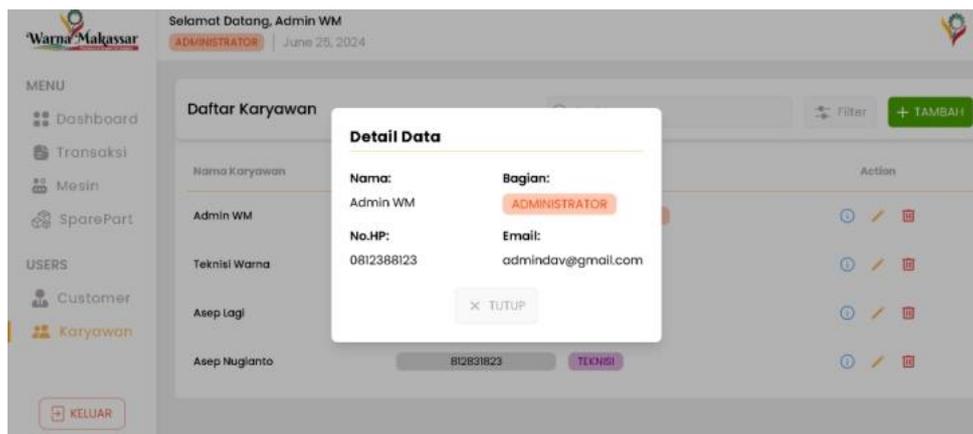
Lampiran 1.74 Halaman Hapus Customer (Admin)



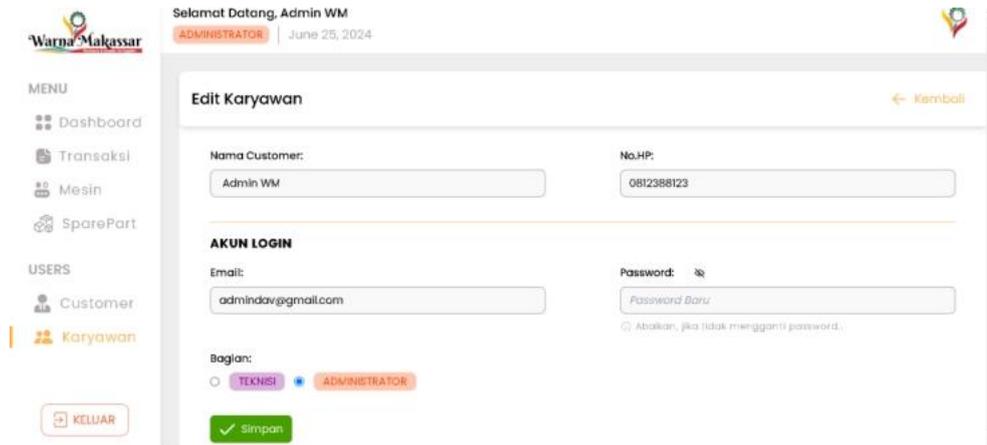
Lampiran 1.75 Halaman Daftar Karyawan (Admin)



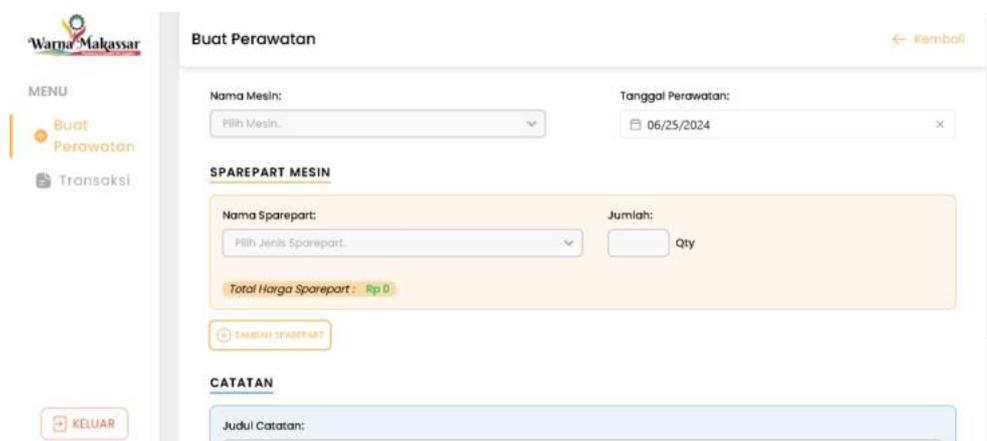
Lampiran 1.76 Halaman Tambah Karyawan (Admin)



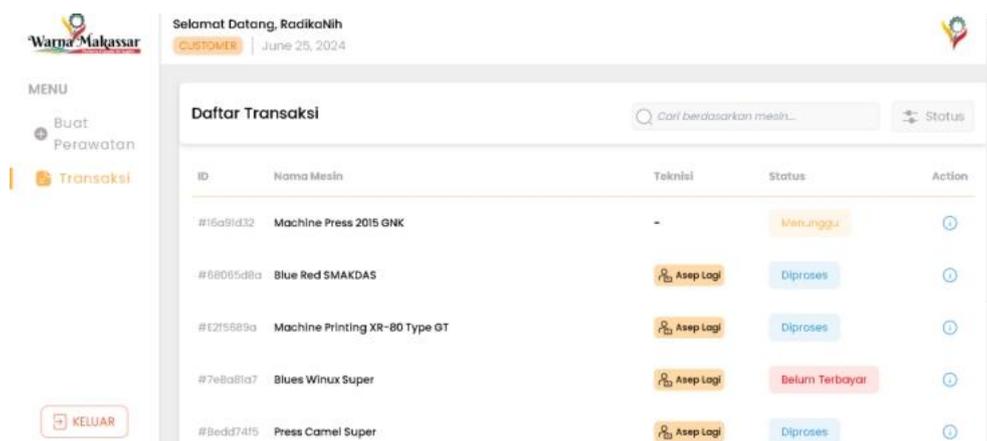
Lampiran 1.77 Halaman Detail Karyawan (Admin)



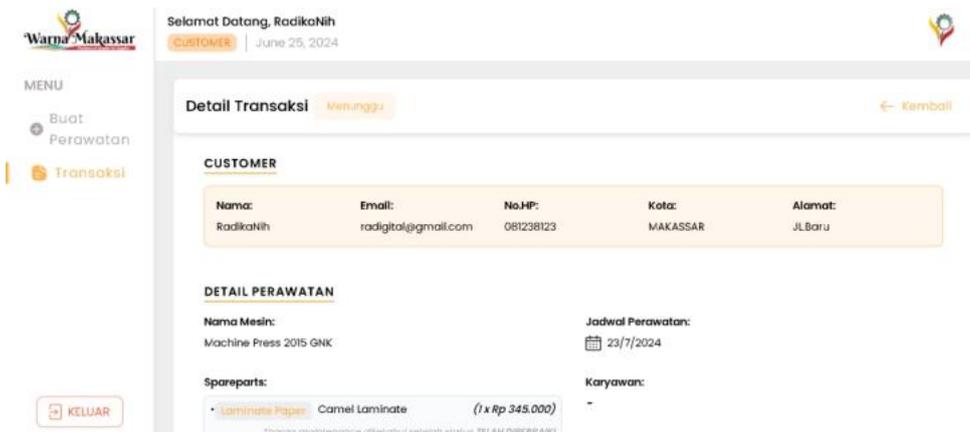
Lampiran 1.78 Halaman Edit Karyawan (Admin)



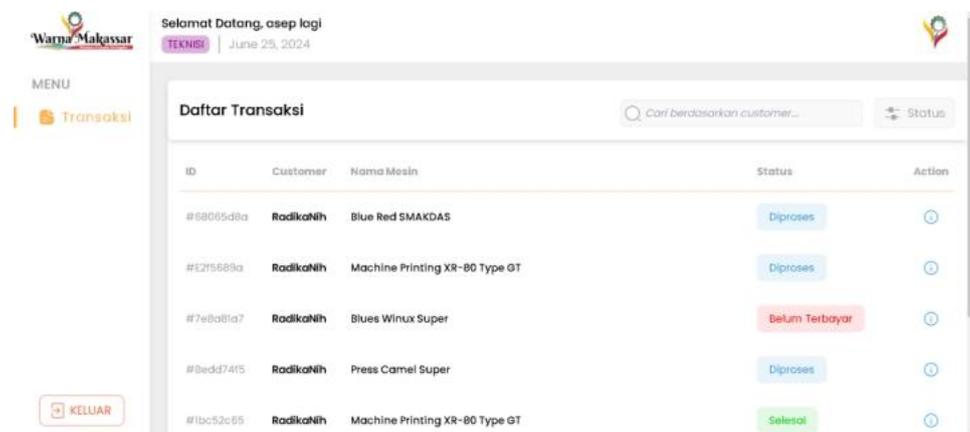
Lampiran 1.79 Halaman Buat Perawatan (Customer)



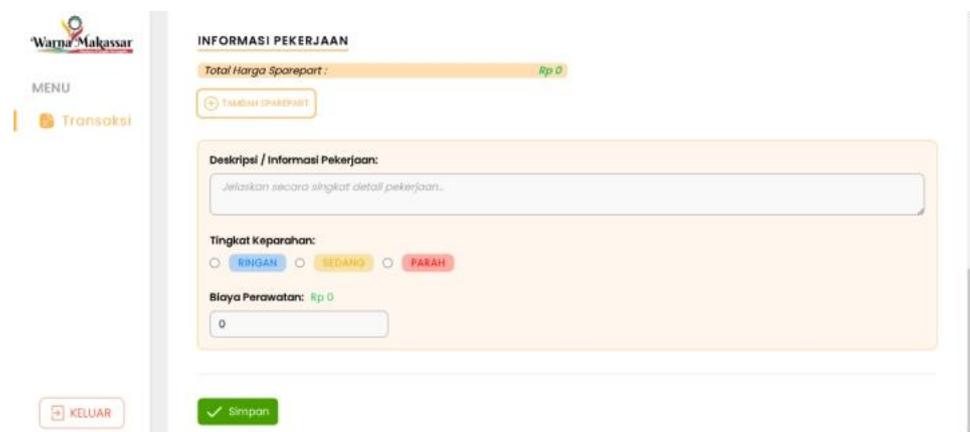
Lampiran 1.80 Halaman Daftar Transaksi (Customer)



Lampiran 1.81 Halaman Detail Transaksi (Customer)



Lampiran 1.82 Halaman Daftar Transaksi (Teknisi)



Lampiran 1.83 Halaman Update Transaksi (Teknisi)

Input Email			
No	Partisi Input	Kondisi Yang Diterima	Kondisi Yang Ditolak
1	Format Email Valid	user@example.com	user@.com, user@domain, <a href="#">user@domain..com</a>
2	Email Kosong	Tidak ada input	Null, ""
Input No.HP			
No	Partisi Input	Kondisi Yang Diterima	Kondisi Yang Ditolak
1	No.HP Valid	08123456789	+62-8123456789, 62 8123456789
2	Format Nomor	08xxxxxxxx	62xxxxxxxx, +62xxxxxxxx, +08xxxxxxxx, 08xx-xxxx-xxxx, 62xx-xxxx-xxxx
3	Format Nomor Tidak Valid	08sad, 23213se	0812-345-6789
4	Digit Valid	10-13 digit	-10 digit, > 13digit
5	No.HP Kosong	Tidak ada input	Null, ""
Input Qty & Min.Qty			
No	Partisi Input	Kondisi Yang Diterima	Kondisi Yang Ditolak
1	Angka Negatif	0	Seluruh angka negatif
2	Angka Nol	0	-100 sampai -1, 1 sampai 99
3	Angka Positif	1 - ribuan	10000 sampai Tidak terbatas
Input Qty Sparepart (Buat Perawatan)			
No	Partisi Input	Kondisi Yang Diterima	Kondisi Yang Ditolak
1	Angka Negatif	0	Seluruh angka negatif

2	Angka Nol	0	-100 sampai -1, 1 sampai 99
3	Angka Positif	1 - ribuan	10000 sampai Tidak terbatas
4	Qty Valid	Angka <i>quantity</i> yang tidak melebihi stock sparepart	Angka <i>quantity</i> melebihi stock sparepart
Input Tanggal Perawatan			
No	Partisi Input	Kondisi Yang Diterima	Kondisi Yang Ditolak
1	Tanggal Valid	Tanggal Sekarang – Setelahnya, Tanggal yang tersedia	Tanggal sebelum hari ini & tanggal block (tidak tersedia) pada <i>datepicker</i>
2	Tanggal Tidak Tersedia	Tanggal Masa Lampau, Tanggal yang tidak tersedia pada <i>datepicker</i>	Tanggal Sekarang – Setelahnya, Tanggal yang tersedia
Input Biaya Perawatan			
No	Partisi Input	Kondisi Yang Diterima	Kondisi Yang Ditolak
1	Angka Negatif	< 0	Seluruh angka negatif
2	Angka Nol	0	Seluruh angka negatif
3	Angka Positif	1 – tidak terbatas	Angka negatif
4	Valid Harga Biaya	0, 500000, Angka Positif	Angka Negatif, Rp.50.000

Lampiran 1.84 Black Box Testing (*Equivalence Partitioning*)

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-001	Login	User melakukan login dengan dengan data yang <u>SALAH</u>	User berada di halaman login	1. User mengisi data email & password	User berhasil masuk ke dalam sistem aplikasi sesuai kategori pengguna	Berhasil login ke dalam sistem aplikasi	Success
		2. User klik tombol "Login"					
		User melakukan login dengan dengan data yang <u>BENAR</u>	User berada di halaman login	1. User mengisi data email & password tidak sesuai	User tidak berhasil masuk ke dalam sistem aplikasi	Muncul notifikasi "Failed" dan gagal login	Success
		2. User klik tombol "Login"					

Lampiran 1.85 Test Case Scenario UC-001

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-002	Buat Transaksi Permintaan	Customer membuat permintaan perawatan mesin dengan data yang <u>BENAR</u>	Customer berhasil login & berada di halaman form buat permintaan mesin	1. Customer mengisi semua form pada form	Customer berhasil membuat transaksi permintaan	Berhasil membuat transaksi	Success
		2. Customer melakukan "Submit"					
		Customer membuat permintaan perawatan mesin dengan data yang <u>SALAH</u>	Customer berhasil login & berada di halaman form buat permintaan mesin	1. Customer tidak mengisi semua form / salah satu form	Customer gagal membuat transaksi permintaan	Muncul notifikasi "Failed"	Success
		2. Customer melakukan "Submit"					
		Admin membuat permintaan perawatan mesin dengan data yang <u>BENAR</u>	Admin berhasil login & berada di halaman form buat permintaan mesin	1. Admin mengisi semua form pada	Admin berhasil membuat transaksi permintaan	Berhasil membuat transaksi	Success
		2. Admin melakukan "Submit"					
		Admin membuat permintaan perawatan mesin dengan data yang <u>SALAH</u>	Admin berhasil login & berada di halaman form buat permintaan mesin	1. Admin tidak mengisi semua form / salah satu form	Admin gagal membuat transaksi permintaan	Muncul notifikasi "Failed"	Success
		2. Admin melakukan "Submit"					

Lampiran 1.86 Test Case Scenario UC-002

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-003	Transaksi	User melihat daftar transaksi yang telah dibuat	User berhasil login & berada di halaman dari menu "Transaksi" yaitu daftar tabel transaksi	1. User monitor daftar transaksi pada tabel	User berhasil melihat & monitor transaksi yang telah dibuat atau diproses	Berhasil melihat daftar transaksi	Success
		User melakukan pencarian transaksi berdasarkan nama mesin		1. User mencari nama transaksi mesin pada input "Cari Transaksi"	User berhasil menemukan data transaksi pada tabel	Data transaksi ditemukan	Success
		User melakukan pencarian transaksi berdasarkan nama mesin dengan data yang <b>SALAH</b>		1. User mencari nama transaksi mesin yang tidak ada pada input "Cari Transaksi"	User tidak menemukan data transaksi yang dicari	Data transaksi tidak ditemukan	Success
		User memilih filter pencarian status transaksi pada daftar transaksi		1. User klik tombol "Filter"	User bisa melihat daftar transaksi berdasarkan filter status yang dipilih	Daftar transaksi tampil dengan status yang dipilih	Success
		User memilih salah satu transaksi pada tabel untuk melihat detail transaksi		2. User memilih salah satu filter status transaksi			
				1. User klik tombol icon "Detail" pada transaksi	User bisa melihat detail informasi transaksi yang dipilih	Halaman detail transaksi tampil sesuai transaksi yang telah dipilih	Success

Lampiran 1.87 Test Case Scenario UC-003

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
		Admin mengelola transaksi dengan status "Menunggu" menjadi "Diproses"	Admin berhasil login & berada di halaman dari menu "Transaksi" yaitu daftar tabel transaksi	1. Admin memilih salah satu transaksi dengan status "Menunggu" 2. Admin klik tombol "Icon Detail" pada transaksi 3. Admin berada di halaman detail transaksi & memilih status transaksi menjadi "Diproses" 4. Admin mengelola transaksi untuk memilih assignment pada teknisi & memilih 5. Admin klik tombol "Submit" untuk update transaksi	Admin berhasil melakukan update transaksi menjadi status "Diproses"	Transaksi berhasil diupdate & status transaksi "Diproses"	Success
		Admin mengelola transaksi dengan status "Menunggu" menjadi "Dibatalkan"		1. Admin memilih salah satu transaksi dengan 2. Admin klik tombol "Icon Detail" pada transaksi 3. Admin berada di halaman detail transaksi & memilih status transaksi menjadi "Dibatalkan" 4. Admin klik tombol "Submit" untuk update transaksi	Admin berhasil melakukan update transaksi menjadi status "Dibatalkan"	Transaksi berhasil diupdate & status transaksi "Dibatalkan"	Success
		Teknisi mengelola transaksi	Teknisi berhasil login & berada di halaman dari	1. Teknisi memilih salah satu transaksi dengan status <i>Diproses</i> 2. Teknisi klik tombol "Icon Detail" pada transaksi 3. Teknisi berada di	Teknisi berhasil melengkapi form perawatan untuk update	Transaksi berhasil diupdate &	

UC-004	Transaksi	dengan status " <u>Diproses</u> " menjadi <u>Telah Diperbaiki</u>	menu "Transaksi" yaitu daftar tabel transaksi	halaman detail transaksi & melengkapi form perawatan	transaksi menjadi status " <u>Telah Diperbaiki</u> "	status transaksi " <u>Telah Diperbaiki</u> "	Success
				5. Teknisi klik tombol "Submit" untuk update transaksi			
		Admin mengelola transaksi dengan status <u>Telah Diperbaiki</u> menjadi " <u>Belum Terbayar</u> "	Admin berhasil login & berada di halaman dari menu "Transaksi" yaitu daftar tabel transaksi	1. Admin memilih salah satu transaksi dengan status <u>Telah Diperbaiki</u> 2. Admin klik tombol " <u>Icon Detail</u> " pada transaksi 3. Admin berada di halaman detail transaksi & memilih status transaksi menjadi <u>Belum Terbayar</u> 4. Admin klik tombol "Submit"	Admin berhasil melakukan update transaksi menjadi status <u>Belum Terbayar</u>	Transaksi berhasil diupdate & status transaksi <u>Belum Terbayar</u>	Success
		Admin mengelola transaksi dengan status <u>Telah Diperbaiki</u> menjadi " <u>Selesai</u> "		1. Admin memilih salah satu transaksi dengan status <u>Telah Diperbaiki</u> 2. Admin klik tombol " <u>Icon Detail</u> " pada transaksi 3. Admin berada di halaman detail transaksi & memilih status transaksi menjadi <u>Selesai</u> 4. Admin klik tombol "Submit"	Admin berhasil melakukan update transaksi menjadi status <u>Selesai</u>	Transaksi berhasil diupdate & status transaksi <u>Selesai</u>	Success
	Admin / Teknisi mengelola transaksi untuk update transaksi dengan tidak memilih status transaksi	Admin / Teknisi berhasil login & berada di halaman dari menu "Transaksi" yaitu daftar tabel transaksi	1. Admin memilih salah satu transaksi dengan status <u>Menunggu / Telah Diperbaiki / Belum Terbayar</u> 2. Admin klik tombol " <u>Icon Detail</u> " pada transaksi 3. Admin berada di halaman detail transaksi & tidak memilih status transaksi menjadi 4. Admin klik tombol "Submit"	Admin tidak berhasil melakukan update transaksi	Transaksi tidak berhasil diupdate & muncul notifikasi status belum terpilih	Success	

Lampiran 1.88 Test Case Scenario UC-004

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-005	Sparepart	Admin mengubah data stock / min.stock sparepart	Admin berhasil login & berada di halaman dari menu <u>Sparepart</u> yaitu daftar tabel sparepart	1. Admin memilih salah satu sparepart & klik tombol icon <u>Detail</u>	Admin berhasil mengubah data stock sparepart	Stok sparepart berhasil diupdate	Success
				2. Admin megubah isi form stock / min.stock			
				3. Admin klik tombol <u>SUBMIT</u>			
		Admin mengubah data stock / min.stock sparepart, tetapi tidak mengisi form stok		1. Admin memilih salah satu sparepart & klik tombol icon <u>Detail</u>	Admin tidak berhasil mengubah data stock sparepart	Stok sparepart tidak berhasil terupdate	
				2. Admin megubah isi form stock / min.stock			
				3. Admin klik tombol <u>SUBMIT</u>			

Lampiran 1.89 Test Case Scenario UC-005

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-006	Mesin	Admin tambah data mesin, mengisi ssesmua form	Admin berhasil login & berada di halaman dari menu Mesin yaitu daftar tabel <u>Mesin</u>	1. Admin klik tombol <b>TAMBAH</b>	Admin menambah data mesin	Mesin berhasil ditambah	Success
				2. Admin mengisi form tambah mesin			
				3. Admin klik tombol <u>SUBMIT</u>			
		Admin tambah data mesin dengan tidak mengisi salah satu form		1. Admin klik tombol <b>TAMBAH</b>	Admin tidak berhasil menambahkan Mesin	Mesin tidak berhasil ditambahkan	
				2. Admin tidak mengisi seluruh atau salah satu form tambah mesin			
				3. Admin klik tombol <u>SUBMIT</u>			

Lampiran 1.90 Test Case Scenario UC-006

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-007	Sparepart	Admin tambah data sparepart	Admin berhasil login & berada di halaman dari menu <u>Sparepart</u> yaitu daftar tabel Sparepart	1. Admin klik tombol <b>TAMBAH</b>	Admin menambah data Sparepart	Sparepart berhasil ditambah	Success
				2. Admin mengisi form tambah sparepart			
				3. Admin klik tombol <b>SUBMIT</b>			
		Admin tambah data sparepart dengan tidak mengisi salah satu form		1. Admin klik tombol <b>TAMBAH</b>	Admin tidak berhasil mengubah data stock Sparepart	Sparepart tidak berhasil ditambahkan	
2. Admin tidak mengisi seluruh atau salah satu form tambah sparepart							
3. Admin klik tombol <b>SUBMIT</b>							

Lampiran 1.91 Test Case Scenario UC-007

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-008	Mesin	Admin melihat / monitor daftar data mesin	Admin berhasil login & berada di halaman dari menu Mesin yaitu daftar tabel Mesin	1. Admin melihat daftar mesin pada tabel	Admin bisa melihat daftar mesin pada tabel	Mesin ditampilkan pada tabel	Success
		Admin mencari data mesin pada input pencarian		1. Admin mencari data mesin yang sesuai	Admin berhasil menemukan data mesin	Data mesin ditemukan dan ditampilkan pada tabel	Success
		Admin mencari data mesin yang tidak sesuai & tidak tersedia dalam input pencarian		1. Admin mencari data mesin yang tidak sesuai	Admin tidak berhasil menemukan data mesin	Data mesin tidak ditemukan dan tabel menampilkan <u>Data tidak ditemukan</u>	Success
		Admin mencari mesin melalui filter class mesin		1. Admin klik tombol <b>FILTER</b>	Admin bisa melihat daftar mesin sesuai filter kelas mesin yang dipilih	Daftar mesin sesuai kelas yang dipilih ditampilkan pada tabel	Success
				2. Admin memilih salah satu class mesin			
		Admin mencari mesin dengan kelas yang tidak tersedia pada daftar, melalui filter class mesin		1. Admin klik tombol <b>FILTER</b>	Admin tidak bisa menemukan daftar data mesin, sesuai filter kelas mesin yang dipilih	Daftar mesin sesuai kelas yang dipilih tidak ditampilkan pada tabel dan menampilkan <u>Data tidak ditemukan</u>	Success
2. Admin memilih salah satu class mesin yang belum tersedia pada daftar mesin							

Lampiran 1.92 Test Case Scenario UC-008

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-009	Mesin	Admin mengubah data mesin & mengisi semua form data mesin	Admin berhasil login & berada di halaman dari menu Mesin yaitu daftar tabel Mesin	1. Admin memilih salah satu data mesin & klik tombol icon <i>Detail</i>	Admin berhasil mengubah data mesin	Mesin berhasil diupdate	Success
		2. Admin mengubah data mesin pada form		3. Admin klik tombol <b>SUBMIT</b>			
UC-009	Mesin	Admin mengubah data mesin & tidak mengisi seluruh atau beberapa form data mesin	Admin berhasil login & berada di halaman dari menu Mesin yaitu daftar tabel Mesin	1. Admin memilih salah satu data mesin & klik tombol icon <i>Detail</i>	Admin tidak berhasil mengubah data mesin	Mesin tidak berhasil diupdate dan muncul notifikasi	Success
		2. Admin mengubah data mesin dan mengkosongkan salah satu form		3. Admin klik tombol <b>SUBMIT</b>			

Lampiran 1.93 Test Case Scenario UC-009

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-010	Customer	Admin melihat / monitor daftar data Customer	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin melihat daftar Customer pada tabel	Admin bisa melihat daftar Customer pada tabel	Customer ditampilkan pada tabel	Success
		Admin mencari data Customer pada input pencarian		1. Admin mencari data Customer yang sesuai	Admin berhasil menemukan data Customer	Data Customer ditemukan dan ditampilkan pada tabel	Success
		Admin mencari data Customer yang tidak sesuai & tidak tersedia dalam input pencarian		1. Admin mencari data Customer yang tidak sesuai	Admin tidak berhasil menemukan data Customer	Data Customer tidak ditemukan dan tabel menampilkan Data tidak ditemukan	Success

Lampiran 1.94 Test Case Scenario UC-010

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-011	Customer	Admin tambah data pengguna <i>Customer</i>	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin klik tombol <b>TAMBAH</b>	Admin berhasil menambahkan user <i>Customer</i>	Customer berhasil ditambahkan	Success
		2. Admin mengisi form pada data <i>customer</i>		3. Admin klik tombol <b>SUBMIT</b>			
UC-011	Customer	Admin tambah data pengguna <i>Customer</i> , dengan mengkosongkan salah satu / semua form	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin klik tombol <b>TAMBAH</b>	Admin tidak berhasil menambahkan pengguna <i>Customer</i>	Pengguna Customer tidak berhasil ditambahkan dan muncul notifikasi untuk melengkapi form	Success
		2. Admin mengisi form dan mengkosongkan		3. Admin klik tombol <b>SUBMIT</b>			

Lampiran 1.95 Test Case Scenario UC-011

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-012	Customer	Admin mengubah data <i>customer</i> tanpa mengubah form <i>password</i>	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin memilih salah satu data customer & klik tombol icon <i>Detail</i>	Admin berhasil mengubah data customer dan data password tidak berubah	Data customer berhasil diubah	Success
		2. Admin mengubah semua data form customer, kecuali form <i>password</i>					
		3. Admin klik tombol <i>SUBMIT</i>					
UC-012	Customer	Admin mengubah data <i>customer</i> termasuk mengubah form <i>password</i>	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin memilih salah satu data customer & klik tombol icon <i>Detail</i>	Admin berhasil mengubah data customer dan data password berubah	Data customer berhasil diubah	Success
		2. Admin mengubah semua data form					
		3. Admin klik tombol <i>SUBMIT</i>					
UC-012	Customer	Admin mengubah data <i>customer</i> , lalu mengkosongkan semua form / salah satu form kecuali form <i>password</i>	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin memilih salah satu data customer & klik tombol icon <i>Detail</i>	Admin tidak berhasil mengubah data customer	Data customer tidak berhasil diubah	Success
		2. Admin mengubah semua data form customer dan mengkosongkan semua form / salah satu form, kecuali form <i>password</i>					
		3. Admin klik tombol <i>SUBMIT</i>					

Lampiran 1.96 Test Case Scenario UC-012

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-013	Customer	Admin menghapus data customer	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin memilih salah satu data customer & klik icon tombol <i>HAPUS</i>	Admin berhasil menghapus data customer	Data customer berhasil terhapus	Success
		2. Admin memilih & klik button <i>IYA</i> pada konfirmasi hapus data					
UC-013	Customer	Admin menghapus data customer, tetapi membatalkan pada <i>pop-up</i> konfirmasi hapus data	Admin berhasil login & berada di halaman dari menu Customer yaitu daftar tabel Customer	1. Admin memilih salah satu data customer & klik icon tombol <i>HAPUS</i>	Admin berhasil mengubah data customer dan data password berubah	Data customer berhasil diubah	Success
		2. Admin memilih & klik button <i>BATAL</i> pada konfirmasi hapus data					

Lampiran 1.97 Test Case Scenario UC-013

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-014	Teknisi	Admin tambah data pengguna Teknisi	Admin berhasil login & berada di halaman dari menu Teknisi yaitu daftar tabel Teknisi	1. Admin klik tombol <b>TAMBAH</b>	Admin berhasil menambahkan user Teknisi	Teknisi berhasil ditambahkan	Success
		2. Admin mengisi form pada data Teknisi					
		Admin tambah data pengguna Teknisi, dengan mengkosongkan salah satu / semua form		3. Admin klik tombol <b>SUBMIT</b>	Admin tidak berhasil menambahkan pengguna Teknisi	Pengguna Teknisi tidak berhasil ditambahkan dan muncul notifikasi untuk melengkapi form	Success

Lampiran 1.98 Test Case Scenario UC-014

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-015	Karyawan	Admin melihat / monitor daftar data Karyawan	Admin berhasil login & berada di halaman dari menu Karyawan yaitu daftar tabel Karyawan	1. Admin melihat daftar Karyawan pada tabel	Admin bisa melihat daftar Karyawan pada tabel	Karyawan ditampilkan pada tabel	Success
		Admin mencari data Karyawan pada input pencarian		1. Admin mencari data Karyawan yang sesuai	Admin berhasil menemukan data Karyawan	Data Karyawan ditemukan dan ditampilkan pada tabel	Success
		Admin mencari data Karyawan yang tidak sesuai & tidak tersedia dalam input pencarian		1. Admin mencari data Karyawan yang tidak sesuai	Admin tidak berhasil menemukan data Karyawan	Data Karyawan tidak ditemukan dan tabel menampilkan Data tidak ditemukan	Success
		Admin mencari data Karyawan yang sesuai melalui filter pencarian bagian Karyawan		1. Admin klik tombol <u>Filter</u>	Admin berhasil menemukan data Karyawan sesuai filter <u>Bagian</u> yang dipilih	Data Karyawan ditemukan pada tabel	Success
				2. Admin memilih salah satu filter bagian Karyawan			
		Admin mencari data Karyawan yang tidak tersedia pada daftar karyawan melalui filter pencarian bagian Karyawan		1. Admin klik tombol <u>Filter</u>	Admin tidak menemukan data Karyawan sesuai filter <u>Bagian</u> yang dipilih	Data Karyawan tidak ditemukan pada tabel & tertulis <u>Data tidak ditemukan</u>	Success
				2. Admin memilih salah satu filter bagian Karyawan yang tidak tersedia pada daftar			

Lampiran 1.99 Test Case Scenario UC-015

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-016	Karyawan	Admin mengubah data Karyawan tanpa mengubah form password	Admin berhasil login & berada di halaman dari menu Karyawan yaitu daftar tabel Karyawan	1. Admin memilih salah satu data Karyawan & klik tombol icon Detail	Admin berhasil mengubah data Karyawan dan data password tidak berubah	Data Karyawan berhasil diubah	Success
				2. Admin mengubah semua data form Karyawan, kecuali form password			
				3. Admin klik tombol <b>SUBMIT</b>			
		Admin mengubah data Karyawan termasuk mengubah form password		1. Admin memilih salah satu data cusomer & klik tombol icon <u>Detail</u>	Admin berhasil mengubah data Karyawan dan data password berubah	Data Karyawan berhasil diubah	Success
				2. Admin mengubah semua data form			
				3. Admin klik tombol <b>SUBMIT</b>			
		Admin mengubah data Karyawan, lalu mengkosongkan semua form / salah satu form kecuali form password		1. Admin memilih salah satu data cusomer & klik tombol icon <u>Detail</u>	Admin tidak berhasil mengubah data Karyawan	Data Karyawan tidak berhasil diubah	Success
				2. Admin mengubah semua data form Karyawan dan mengkosongkan semua form / salah satu form , kecuali form password			
				3. Admin klik tombol <b>SUBMIT</b>			

Lampiran 1.100 Test Case Scenario UC-016

Test Case ID	Menu / Halaman	Test Case	Pre-Condition	Test Steps	Expected Result	Result	Status
UC-017	Karyawan	Admin menghapus data Karyawan	Admin berhasil login & berada di halaman dari menu Karyawan yaitu daftar tabel Karyawan	1. Admin memilih salah satu data Karyawan & klik icon tombol <b>HAPUS</b>	Admin berhasil menghapus data Karyawan	Data Karyawan berhasil terhapus	Success
		Admin menghapus data Karyawan, tetapi membatalkan pada pop-up konfirmasi hapus data		2. Admin memilih & klik button <i>IYA</i> pada konfirmasi hapus data			
				1. Admin memilih salah satu data Karyawan & klik icon tombol <b>HAPUS</b>	Admin berhasil mengubah data Karyawan dan data password berubah	Data Karyawan berhasil diubah	Success
				2. Admin memilih & klik button <i>BATAL</i> pada konfirmasi hapus data			

Lampiran 1.101 Test Case Scenario UC-017