

**FUNCTIONAL SPESIFICATION DOCUMENT PENGEMBANGAN SISTEM
INFORMASI MANAJEMEN PENJUALAN SUKU CADANG DAN
SERVIS SEPEDA MOTOR BERBASIS WEB (STUDI KASUS ANWAR
MOTOR)**

DAFTAR ISI

FUNCTIONAL SPESIFICATION DOCUMENT PENGEMBANGAN SISTEM INFORMASI MANAJEMEN PENJUALAN SUKU CADANG DAN SERVIS SEPEDA MOTOR BERBASIS WEB (STUDI KASUS ANWAR MOTOR)	i
DAFTAR ISI	ii
DAFTAR TABEL	vii
DAFTAR GAMBAR	xi
BAB 1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Manfaat	2
1.5 Batasan Masalah	3
1.6 Sistematika Pembahasan	3
BAB 2 METODOLOGI	5
2.1 Rekayasa Kebutuhan	5
2.2 Perancangan	6
2.3 Implementasi	6
2.4 Pengujian	6
2.5 Kesimpulan dan Saran	7
BAB 3 REKAYASA KEBUTUHAN	8
3.1 Gambaran Umum Sistem	8
3.2 Elisitasi Kebutuhan dan Analisis Kebutuhan	8
3.3 Pemodelan Proses Bisnis	8
3.4 Identifikasi Aktor	12
3.5 Daftar Kebutuhan Fungsional Sistem	13
3.5.1 Daftar Kebutuhan Non-Fungsional Sistem	18
3.6 <i>Use Case Diagram</i>	18
3.7 <i>Use Case Scenario</i>	20
3.7.1 <i>Use Case Scenario Login</i>	20

3.7.2 Use Case Scenario Melihat Laporan Suku Cadang Rendah	20
3.7.3 Use Case Scenario Membeli Suku Cadang	21
3.7.4 Use Case Scenario Melihat Suku Cadang	22
3.7.5 Use Case Scenario Mengubah Suku Cadang	22
3.7.6 Use Case Scenario Menghapus Suku Cadang	23
3.7.7 Use Case Scenario Melihat Servis	23
3.7.8 Use Case Scenario Menambah Servis	24
3.7.9 Use Case Scenario Mengubah Servis	25
3.7.10 Use Case Scenario Menghapus Servis	25
3.7.11 Use Case Scenario Melihat <i>Mechanic</i>	26
3.7.12 Use Case Scenario Menambah <i>Mechanic</i>	26
3.7.13 Use Case Scenario Mengubah <i>Mechanic</i>	27
3.7.14 Use Case Scenario Menghapus <i>Mechanic</i>	27
3.7.15 Use Case Scenario Melihat Gaji <i>Mechanic</i>	28
3.7.16 Use Case Scenario Mengubah Status Peminjaman <i>Mechanic</i> ..	28
3.7.17 Use Case Scenario Menambah Pelanggan	29
3.7.18 Use Case Scenario Melihat Pelanggan	30
3.7.19 Use Case Scenario Mengubah Pelanggan	30
3.7.20 Use Case Scenario Menghapus Pelanggan	31
3.7.21 Use Case Scenario Menambah Transaksi Penjualan Suku Cadang	31
3.7.22 Use Case Scenario Melihat Transaksi Penjualan Suku Cadang ..	32
3.7.23 Use Case Scenario Mengubah Transaksi Pembelian Suku Cadang	32
3.7.24 Use Case Scenario Menghapus Transaksi Pembelian Suku Cadang	33
3.7.25 Use Case Scenario Menambah Transaksi Servis	33
3.7.26 Use Case Scenario Melihat Transaksi Servis	34
3.7.27 Use Case Scenario Mengubah Transaksi Servis	35
3.7.28 Use Case Scenario Menghapus Transaksi Servis	35
3.7.29 Use Case Scenario Menambah Transaksi Servis dan Suku Cadang	36
3.7.30 Use Case Scenario Melihat Transaksi Servis dan Suku Cadang .	36
3.7.31 Use Case Scenario Mengubah Transaksi Servis dan Suku Cadang	37

3.7.32 <i>Use Case Scenario</i> Menghapus Transaksi Servis dan Suku Cadang	38
3.7.33 <i>Use Case Scenario</i> Menambah Pinjaman	38
3.7.34 <i>Use Case Scenario</i> Melihat Pinjaman	39
3.7.35 <i>Use Case Scenario</i> Mengubah Pinjaman	39
3.7.36 <i>Use Case Scenario</i> Menghapus Pinjaman	40
3.7.37 <i>Use Case Scenario</i> Melihat Profil	40
3.7.38 <i>Use Case Scenario</i> Mengubah Profil	41
3.7.39 <i>Use Case Scenario</i> Mencetak <i>Invoice</i> Transaksi	41
3.7.40 <i>Use Case Scenario</i> Logout	42
3.7.41 <i>Use Case Scenario</i> Menambah Laporan Stok Rendah	42
3.7.42 <i>Use Case Scenario</i> Menambah Laporan Stok Rendah	43
BAB 4 PERANCANGAN DAN IMPLEMENTASI	45
4.1 Perancangan	45
4.1.1 Perancangan Arsitektur	45
4.1.2 Perancangan Data	49
4.1.3 Perancangan Komponen	52
4.1.4 Perancangan Antarmuka	53
4.2 Implementasi Sistem	58
4.2.1 Spesifikasi Sistem	58
4.2.2 Implementasi Kode Program	58
4.2.3 Implementasi Basis Data	61
4.2.4 Implementasi Antarmuka	62
BAB 5 PENGUJIAN	66
5.1 Pengujian Unit	66
5.1.1 Pengujian Unit Menambah Laporan Stok Rendah	66
5.1.2 Pengujian Unit Mengubah Suku Cadang	68
5.1.3 Pengujian Unit Mengubah Profil	71
5.2 Pengujian Integrasi	73
5.2.1 Pengujian Integrasi Menambah Transaksi Servis dan Suku Cadang	73
5.3 Pengujian Validasi	78
5.3.1 Pengujian Validasi Menambah Transaksi Servis dan Suku Cadang	78

5.3.2 Pengujian Validasi Melihat Suku Cadang Rendah	79
5.3.3 Pengujian Validasi Membeli Suku Cadang	79
5.3.4 Pengujian Validasi Melihat Suku Cadang	80
5.3.5 Pengujian Validasi Mengubah Suku Cadang	81
5.3.6 Pengujian Validasi Menghapus Suku Cadang	81
5.3.7 Pengujian Validasi Melihat Servis	82
5.3.8 Pengujian Validasi Menambah Servis	83
5.3.9 Pengujian Validasi Mengubah Servis	83
5.3.10 Pengujian Validasi Menghapus Servis	84
5.3.11 Pengujian Validasi Melihat <i>Mechanic</i>	84
5.3.12 Pengujian Validasi Menambah <i>Mechanic</i>	85
5.3.13 Pengujian Validasi Mengubah <i>Mechanic</i>	85
5.3.14 Pengujian Validasi Menghapus <i>Mechanic</i>	86
5.3.15 Pengujian Validasi Melihat Gaji <i>Mechanic</i>	86
5.3.16 Pengujian Validasi Mengubah Status Peminjaman <i>Mechanic</i> ..	87
5.3.17 Pengujian Validasi Menambah Pelanggan	87
5.3.18 Pengujian Validasi Melihat Pelanggan	88
5.3.19 Pengujian Validasi Mengubah Pelanggan	88
5.3.20 Pengujian Validasi Menghapus Pelanggan	89
5.3.21 Pengujian Validasi Menambah Transaksi Penjualan Suku Cadang	89
5.3.22 Pengujian Validasi Melihat Transaksi Penjualan Suku Cadang .	90
5.3.23 Pengujian Validasi Mengubah Transaksi Pembelian Suku Cadang	90
5.3.24 Pengujian Validasi Menghapus Transaksi Pembelian Suku Cadang	91
5.3.25 Pengujian Validasi Menambah Transaksi Servis	92
5.3.26 Pengujian Validasi Melihat Transaksi Servis	92
5.3.27 Pengujian Validasi Mengubah Transaksi Servis	93
5.3.28 Pengujian Validasi Menghapus Transaksi Servis	93
5.3.29 Pengujian Validasi Melihat Transaksi Servis dan Suku Cadang .	94
5.3.30 Pengujian Validasi Mengubah Transaksi Servis dan Suku Cadang	94
5.3.31 Pengujian Validasi Menghapus Transaksi Servis dan Suku Cadang	95

5.3.32 Pengujian Validasi Menambah Pinjaman	96
5.3.33 Pengujian Validasi Melihat Pinjaman	96
5.3.34 Pengujian Validasi Mengubah Pinjaman	96
5.3.35 Pengujian Validasi Menghapus Transaksi Servis	97
5.3.36 Pengujian Validasi Melihat Profil	98
5.3.37 Pengujian Validasi Mengubah Profil	98
5.3.38 Pengujian Validasi Mencetak <i>Invoice</i> Transaksi	99
5.3.39 Pengujian Validasi Logout	100
5.3.40 Pengujian Validasi <i>Login</i>	100
5.3.41 Pengujian Validasi Menambah Laporan Suku Cadang Rendah	101
5.3.42 Pengujian Compatibility	101
LAMPIRAN A HASIL WAWANCARA	102
LAMPIRAN B HASIL WAWANCARA	104
LAMPIRAN C DOKUMEN	106

DAFTAR TABEL

Tabel 3.1 Identifikasi Aktor	12
Tabel 3.2 Kebutuhan Fungsional	13
Tabel 3.3 Kebutuhan Non-Fungsional Sistem	18
Tabel 3.4 <i>Use Case Scenario Login</i>	20
Tabel 3.5 <i>Use Case Scenario</i> Melihat Laporan Suku Cadang Rendah	20
Tabel 3.6 <i>Use Case Scenario</i> Membeli Suku Cadang	21
Tabel 3.7 <i>Use Case Scenario</i> Melihat Suku Cadang	22
Tabel 3.8 <i>Use Case Scenario</i> Mengubah Suku Cadang	22
Tabel 3.9 <i>Use Case Scenario</i> Menghapus Suku Cadang	23
Tabel 3.10 <i>Use Case Scenario</i> Melihat Servis	23
Tabel 3.11 <i>Use Case Scenario</i> Menambah Servis	24
Tabel 3.12 <i>Use Case Scenario</i> Mengubah Servis	25
Tabel 3.13 <i>Use Case Scenario</i> Menghapus Servis	25
Tabel 3.14 <i>Use Case Scenario</i> Melihat <i>Mechanic</i>	26
Tabel 3.15 <i>Use Case Scenario</i> Menambah <i>Mechanic</i>	26
Tabel 3.16 <i>Use Case Scenario</i> Mengubah <i>Mechanic</i>	27
Tabel 3.17 <i>Use Case Scenario</i> Menghapus <i>Mechanic</i>	27
Tabel 3.18 <i>Use Case Scenario</i> Melihat Gaji <i>Mechanic</i>	28
Tabel 3.19 <i>Use Case Scenario</i> Mengubah Status Peminjaman <i>Mechanic</i>	28
Tabel 3.20 <i>Use Case Scenario</i> Menambah Pelanggan	29
Tabel 3.21 <i>Use Case Scenario</i> Melihat Pelanggan	30
Tabel 3.22 <i>Use Case Scenario</i> Mengubah Pelanggan	30
Tabel 3.23 <i>Use Case Scenario</i> Menghapus Pelanggan	31
Tabel 3.24 <i>Use Case Scenario</i> Menambah Transaksi Penjualan Suku Cadang	31
Tabel 3.25 <i>Use Case Scenario</i> Melihat Transaksi Penjualan Suku Cadang	32
Tabel 3.26 <i>Use Case Scenario</i> Mengubah Transaksi Pembelian Suku Cadang	32
Tabel 3.27 <i>Use Case Scenario</i> Menghapus Transaksi Pembelian Suku Cadang ...	33
Tabel 3.28 <i>Use Case Scenario</i> Menambah Transaksi Servis	34
Tabel 3.29 <i>Use Case Scenario</i> Melihat Transaksi Servis	34
Tabel 3.30 <i>Use Case Scenario</i> Mengubah Transaksi Servis	35

Tabel 3.31 <i>Use Case Scenario</i> Menghapus Transaksi Servis	35
Tabel 3.32 <i>Use Case Scenario</i> Menambah Transaksi Servis dan Suku Cadang	36
Tabel 3.33 <i>Use Case Scenario</i> Melihat Transaksi Servis dan Suku Cadang	37
Tabel 3.34 <i>Use Case Scenario</i> Mengubah Transaksi Servis dan Suku Cadang	37
Tabel 3.35 <i>Use Case Scenario</i> Menghapus Transaksi Servis dan Suku Cadang	38
Tabel 3.36 <i>Use Case Scenario</i> Menambah Pinjaman	38
Tabel 3.37 <i>Use Case Scenario</i> Melihat Pinjaman	39
Tabel 3.38 <i>Use Case Scenario</i> Mengubah Pinjaman	39
Tabel 3.39 <i>Use Case Scenario</i> Menghapus Pinjaman	40
Tabel 3.40 <i>Use Case Scenario</i> Melihat Profil	41
Tabel 3.41 <i>Use Case Scenario</i> Mengubah Profil	41
Tabel 3.42 <i>Use Case Scenario</i> Mencetak <i>Invoice</i> Transaksi	42
Tabel 3.43 <i>Use Case Scenario Logout</i>	42
Tabel 3.44 <i>Use Case Scenario</i> Laporan Stok Rendah	43
Tabel 3.45 <i>Use Case Scenario</i> Laporan Stok Rendah	43
Tabel 4.1 Algoritme Menambah Laporan Stok Rendah	52
Tabel 4.2 Algoritme Mengubah Suku Cadang	52
Tabel 4.3 Algoritme Mengubah Profil	53
Tabel 4.4 Uraian rancangan antarmuka halaman <i>Login</i>	54
Tabel 4.5 Uraian rancangan antarmuka halaman <i>Dashboard Manager</i>	55
Tabel 4.6 Uraian rancangan antarmuka halaman Laporan Stok Rendah	55
Tabel 4.7 Uraian rancangan antarmuka halaman Pembelian Barang	57
Tabel 4.8 Spesifikasi Perangkat Keras Sistem	58
Tabel 4.9 Spesifikasi Perangkat Lunak Sistem	58
Tabel 4.10 Implementasi Kode Menambah Laporan Stok Rendah	59
Tabel 4.11 Implementasi Kode Mengubah Suku Cadang	60
Tabel 4.12 Implementasi Kode Mengubah Profil	60
Tabel 5.1 Algoritme Fungsi <i>add_report()</i>	66
Tabel 5.2 Hasil Pengujian Unit Fungsi <i>store(Request \$request)</i>	67
Tabel 5.3 Algoritme Fungsi <i>update()</i>	68
Tabel 5.4 Hasil Pengujian Unit Fungsi <i>update()</i>	70
Tabel 5.5 Algoritme Fungsi <i>update()</i>	71

Tabel 5.6 Hasil Pengujian Unit Fungsi <i>update(Request \$request)</i>	72
Tabel 5.7 Pseudocode Algoritme Fungsi <i>store add_transaction()</i>	74
Tabel 5.8 Hasil Pengujian Integrasi Fungsi <i>add_transaction()</i>	76
Tabel 5.9 Pengujian Validasi Menambah Transaksi Servis dan Suku Cadang	78
Tabel 5.10 Pengujian Validasi Melihat Suku Cadang Rendah	79
Tabel 5.11 Pengujian Validasi Membeli Suku Cadang	79
Tabel 5.12 Pengujian Validasi Melihat Suku Cadang	80
Tabel 5.13 Pengujian Validasi Mengubah Suku Cadang	81
Tabel 5.14 Pengujian Validasi Menghapus Suku Cadang	81
Tabel 5.15 Pengujian Validasi Melihat Servis	82
Tabel 5.16 Pengujian Validasi Menambah Servis	83
Tabel 5.17 Pengujian Validasi Mengubah Servis	83
Tabel 5.18 Pengujian Validasi Menghapus Servis	84
Tabel 5.19 Pengujian Validasi Melihat <i>Mechanic</i>	84
Tabel 6.20 Pengujian Validasi Menambah <i>Mechanic</i>	85
Tabel 5.21 Pengujian Validasi Mengubah <i>Mechanic</i>	85
Tabel 5.22 Pengujian Validasi Menghapus <i>Mechanic</i>	86
Tabel 5.23 Pengujian Validasi Melihat Gaji <i>Mechanic</i>	86
Tabel 6.24 Pengujian Validasi Mengubah Status Peminjaman <i>Mechanic</i>	87
Tabel 5.25 Pengujian Validasi Menambah Pelanggan	87
Tabel 5.26 Pengujian Validasi Melihat Pelanggan	88
Tabel 6.27 Pengujian Validasi Mengubah Pelanggan	88
Tabel 5.28 Pengujian Validasi Menghapus Pelanggan	89
Tabel 5.29 Pengujian Validasi Menambah Transaksi Penjualan Suku Cadang	89
Tabel 5.30 Pengujian Validasi Melihat Pelanggan	90
Tabel 5.31 Pengujian Validasi Mengubah Transaksi Pembelian Suku Cadang	90
Tabel 5.32 Pengujian Validasi Menghapus Transaksi Pembelian Suku Cadang ...	91
Tabel 5.33 Pengujian Validasi Menambah Transaksi Servis	92
Tabel 5.34 Pengujian Validasi Melihat Transaksi Servis	92
Tabel 5.35 Pengujian Validasi Mengubah Transaksi Servis	93
Tabel 5.36 Pengujian Validasi Menghapus Transaksi Servis	93
Tabel 5.37 Pengujian Validasi Melihat Transaksi Servis dan Suku Cadang	94

Tabel 5.38 Pengujian Validasi Mengubah Transaksi Servis dan Suku Cadang	94
Tabel 5.39 Pengujian Validasi Menghapus Transaksi Servis dan Suku Cadang	95
Tabel 5.40 Pengujian Validasi Menambah Pinjaman	96
Tabel 5.41 Pengujian Validasi Melihat Pinjaman	96
Tabel 5.42 Pengujian Validasi Mengubah Pinjaman	97
Tabel 5.43 Pengujian Validasi Menghapus Transaksi Servis	97
Tabel 5.44 Pengujian Validasi Melihat Profil	98
Tabel 5.45 Pengujian Validasi Mengubah Profil	98
Tabel 5.46 Pengujian Validasi Mencetak <i>Invoice</i> Transaksi	99
Tabel 5.47 Pengujian Validasi Logout	100
Tabel 5.48 Pengujian Validasi <i>Login</i>	100
Tabel 5.49 Pengujian Validasi Menambah Laporan Suku Cadang Rendah	101

DAFTAR GAMBAR

Gambar 3.1 Metodologi proyek	5
Gambar 4.1 Transaksi Bengkel <i>As-Is</i>	9
Gambar 4.2 Transaksi Servis <i>To-Be</i>	9
Gambar 4.3 Transaksi Penjualan Suku Cadang <i>To-Be</i>	10
Gambar 4.4 Peminjaman Uang <i>As-Is</i>	10
Gambar 4.5 Peminjaman Uang <i>To-Be</i>	11
Gambar 4.6 Pembelian Suku Cadang <i>As-Is</i>	11
Gambar 4.7 Pembelian Suku Cadang <i>To-Be</i>	12
Gambar 4.8 Deskripsi Aktor	12
Gambar 3.9 <i>Use Case Diagram</i>	19
Gambar 4.1 Sequence Menambah Laporan Stok Rendah	45
Gambar 4.2 Sequence Diagram Mengubah Suku Cadang	46
Gambar 4.3 Sequence Mengubah Profil	47
Gambar 4.4 Perancangan Class Diagram	48
Gambar 4.5 <i>Conceptual Data Model (CDM)</i>	50
Gambar 4.6 <i>Physical Data Model (PDM)</i>	51
Gambar 4.7 Perancangan Antarmuka <i>Login</i>	54
Gambar 4.8 Antarmuka <i>Dashboard Manager</i>	54
Gambar 4.9 Perancangan Antarmuka Laporan Stok Rendah	55
Gambar 4.10 Perancangan Antarmuka Pembelian Suku Cadang	56
Gambar 4.11 Perancangan Antarmuka Pembelian Suku Cadang	56
Gambar 4.12 Implementasi Basis Data	62
Gambar 4.13 Implementasi Antarmuka <i>Login</i>	62
Gambar 4.14 Implementasi Antarmuka <i>Dashboard Manager</i>	63
Gambar 4.15 Implementasi Antarmuka Laporan Stok Rendah	63
Gambar 4.16 Implementasi Antarmuka Laporan Stok Rendah	65
Gambar 4.17 Implementasi Antarmuka Laporan Stok Rendah	65
Gambar 5.1 .1 <i>Flow Graph</i> fungsi <i>store(Request \$request)</i>	67
Gambar 5.2 .2 <i>Flow Graph</i> fungsi <i>update()</i>	69
Gambar 5.3 .3 <i>Flow Graph</i> fungsi <i>update()</i>	72

Gambar 5.4 <i>Flow Graph</i> fungsi <i>add_transaction()</i>	75
Gambar 5.5 Hasil Pengujian Compatibility	101
Gambar 7.1 Nota Pembelian Suku Cadang ke atau <i>supplier</i>	106
Gambar 7.2 Nota Transaksi Penjualan Suku Cadang	106
Gambar 7.3 Penggajian Karyawan	107

BAB 1 PENDAHULUAN

1.1 Latar belakang

Sepeda motor merupakan alat transportasi dengan ukuran yang kecil serta memiliki kecepatan tinggi, sehingga disukai oleh sebagian orang untuk alat transportasi sehari-hari. Sepeda motor telah terbukti menjadi alat transportasi yang populer, terutama Negara di Asia (Vaya dkk, 2019). Sepeda motor membutuhkan servis selama penggunaan, servis tersebut dilakukan di bengkel motor. Bengkel motor melakukan servis seperti ganti oli serta suku cadang yang rusak.

Anwar motor merupakan salah satu bengkel motor yang melakukan servis serta penjualan suku cadang, oli, dan aksesoris motor di Kabupaten Gresik. Anwar motor melayani rata-rata 100 pelanggan setiap harinya. Anwar motor memiliki pelayanan yaitu servis, pembelian, dan penjualan suku cadang. Pembelian suku cadang di anwar motor yaitu dengan cara menghubungi *supplier* untuk membeli suku cadang. Pembelian suku cadang dilakukan apabila stok telah habis. Namun, jika stok suku cadang tersedia, maka anwar motor dapat melakukan penjualan suku cadang.

Anwar motor melayani penjualan suku cadang yang dibutuhkan oleh pelanggan. Kemudian, pelanggan dapat meminta kepada *mechanic* untuk mengganti suku cadang yang telah dibeli. Selain itu, anwar motor memiliki beberapa *mechanic* untuk melayani servis motor sesuai dengan kebutuhan pelanggan. *mechanic* di anwar motor mendapatkan fasilitas untuk melakukan peminjaman uang. Peminjaman uang dapat dibayar melalui pemotongan gaji *mechanic* setiap bulan.

Manajemen bengkel anwar motor hampir tidak memiliki pencatatan untuk transaksi penjualan. Hal tersebut dapat mempengaruhi dalam aktivitas pembelian dan penjualan suku cadang serta servis motor. Pembelian suku cadang dilakukan ketika stok telah habis, namun seringkali pihak bengkel kehabisan stok suku cadang ketika diperlukan. Selain itu, menghambat dalam melakukan servis motor serta penjualan suku cadang menjadi lebih lama karena harus mengecek setiap suku cadang yang dibutuhkan pelanggan.

Pencatatan peminjaman uang tidak semua tercatat sehingga menimbulkan pencatatan yang tidak konsisten, padahal pembayaran peminjaman uang dibayar melalui gaji *mechanic*. Penggajian *mechanic* merupakan hasil dari servis yang didapatkan dari setiap *mechanic* itu sendiri, tanpa persentase untuk bengkel. Namun, jika seorang *mechanic* memiliki peminjaman uang akan dilakukan pemotongan gaji.

Berdasarkan dari permasalahan yang telah dijabarkan, maka proyek ini memiliki solusi dengan membangun sebuah aplikasi web yang dalam pengembangannya menerapkan teknologi pengembangan web sehingga aplikasi dapat diakses melalui platform web browser. Aplikasi yang dibangun nantinya diharapkan dapat membantu bengkel anwar motor untuk melakukan pencatatan lebih lengkap setiap transaksi di bengkel anwar motor.

1.2 Rumusan Masalah

Dari apa yang sudah dijabarkan pada latar belakang, bisa ditarik beberapa rumusan masalah dari proyek ini yaitu :

1. Bagaimana hasil analisis kebutuhan untuk mengembangkan Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor)?
2. Bagaimana hasil perancangan Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor)?
3. Bagaimana hasil implementasi metode *Waterfall Development* dilakukan pada perancangan Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor)?
4. Bagaimana hasil pengujian Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor)?

1.3 Tujuan

Tujuan yang ingin dicapai pada proyek ini adalah:

1. Untuk mengetahui dan mendapatkan hasil analisis serta spesifikasi kebutuhan Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor).
2. Untuk mengetahui dan mendapatkan hasil dari perancangan perangkat lunak dengan spesifikasi kebutuhan yang sesuai dengan Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor).
3. Untuk mengetahui dan mendapatkan rancangan sistem dan hasil implementasi dengan mempertimbangkan spesifikasi kebutuhan Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor).
4. Untuk mengetahui dan mendapatkan hasil pengujian yang dilakukan pada Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor).

1.4 Manfaat

Manfaat proyek ini bagi Anwar Motor adalah:

1. Mempermudah Anwar Motor dalam mengelola suku cadang.

2. Membantu proses pelaporan keuangan di bengkel Anwar Motor.

1.5 Batasan Masalah

Batasan-batasan masalah dalam proyek ini agar rumusan masalah tidak meluas dan tetap fokus adalah sebagai berikut:

1. Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor yang berbasis web agar sistem mudah diakses oleh *manager*, *cashier*, dan *mechanic*.
2. Lingkup Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor Berbasis Web (Studi Kasus Bengkel Anwar Motor) ini hanya mengarah pada pengelolaan suku cadang baik dalam penjualan dan pembelian, *mechanic*, dan peminjaman uang.

1.6 Sistematika Pembahasan

Pada susunan proyek ini, terdapat sistematika penulisan skripsi sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini memaparkan tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika pembahasan sebagai dasar penjelasan proyek pengembangan Sistem Informasi Manajemen Bengkel (Studi Kasus Anwar Motor).

BAB 2 TINJAUAN PUSTAKA

Bab ini berisikan beberapa proyek yang telah dilakukan sebelumnya berdasarkan metode dan jenis pengembangan yang menyerupai dengan proyek pengembangan Sistem Informasi Manajemen Bengkel (Studi Kasus Anwar Motor). dan dapat mendukung proyek ini.

BAB 3 METODOLOGI proyek

Bab ini berisi metodologi atau tahapan yang dilakukan untuk membangun sistem, yang terdiri atas studi literatur, perancangan sistem, implementasi sistem, pengujian serta analisis sistem.

BAB 4 PERANCANGAN SISTEM

Bab ini berisi tentang perancangan dalam pembangunan sistem seperti spesifikasi kebutuhan, perancangan basis data, *user interface* (antarmuka) dan alur sistem.

BAB 5 IMPLEMENTASI

Bab ini berisikan beberapa hasil dari analisis kebutuhan dan perancangan yang sudah dilakukan dan diimplementasikan ke sistem.

BAB 6 PENGUJIAN DAN ANALISIS

Bab ini berisi mengenai pengujian sistem dengan menggunakan teknik pengujian *white box* dalam pengujian unit dan integrasi, dan teknik

pengujian *black box* dalam pengujian validasi dan pengujian *compatibility* serta menganalisis proses pembangunan sistem.

BAB 7 PENUTUP

Bab ini berisikan kesimpulan yang diambil dari keseluruhan proyek ini dengan menambahkan beberapa saran sebagai bahan untuk mengembangkan proyek selanjutnya yang lebih baik.

BAB 2 METODOLOGI

Metodologi yang digunakan dalam proyek ini mengadopsi dari SDLC, yaitu menggunakan *Waterfall*. Alur dari metodologi dapat dilihat pada Gambar 3.1, serta uraian alur metodologi terdapat pada sub-bab 3.1 sampai 3.6



Gambar 3.1 Metodologi proyek

2.1 Rekayasa Kebutuhan

Rekayasa Kebutuhan menggunakan pendekatan OOA. Tahapan rekayasa kebutuhan meliputi elisitasi, analisis serta spesifikasi kebutuhan. Elisitasi dan analisis kebutuhan dilakukan dengan cara identifikasi serta analisis kebutuhan dengan metode wawancara secara terstruktur dan pengamatan dokumen. Narasumber dalam wawancara yaitu Pak Fitrul selaku pemilik Anwar Motor untuk mendapatkan informasi terkait suku cadang yang di jual, serta proses bisnis pada setiap transaksi dan Pak Mubin selaku *mechanic* untuk mendapatkan informasi terkait servis.

Pengamatan dokumen yaitu nota transaksi dan penggajian *mechanic*. Nota transaksi terdapat informasi seperti tanggal pembelian, nama supplier, nama suku cadang, nama pelanggan, servis, nama *mechanic* dan tanggal penjualan suku cadang. Penggajian *mechanic* terdapat informasi seperti gaji *mechanic*, peminjaman uang dan tanggal peminjaman. Informasi tersebut menjadi acuan pelaporan dalam sistem yang akan dikembangkan.

Analisis kebutuhan menentukan kebutuhan fungsional dan non fungsional yang di dapatkan dari tahapan elisitasi kebutuhan. Kemudian, kebutuhan yang

telah terkumpul ditambahkan spesifikasi kebutuhan. Spesifikasi kebutuhan tersebut dibuat lebih detail dengan menggunakan pemodelan dan pengembangan sistem berupa *use case diagram* serta *use case scenario*.

2.2 Perancangan

Tahapan perancangan untuk mengubah kebutuhan yang didapatkan menjadi rancangan. Sistem Informasi Manajemen Penjualan Suku Cadang dan Servis Sepeda Motor menggunakan pendekatan *OOD* yang didapatkan dari analisis kebutuhan. Tahapan perancangan terdiri dari perancangan arsitektur yang menggunakan pemodelan Diagram UML yaitu *Sequence Diagram* serta *Class Diagram*, dan perancangan data menggunakan pemodelan yaitu *Entity Relationship Diagram* dan *Physical Data Modeling*, serta perancangan komponen ditulis dalam bentuk *pseudocode* untuk 3 algoritme utama dan perancangan antarmuka.

2.3 Implementasi

Tahapan ini menggunakan pendekatan OOP dan teknologi pengembangan yaitu Bahasa pemrograman PHP untuk pengembangan web dengan menggunakan *framework* Laravel. Implementasi *database* yang digunakan adalah MySQL. Implementasi antarmuka (*User Interface*) menggunakan teknologi HTML, CSS, dan javascript.

2.4 Pengujian

Tahapan ini menggunakan pendekatan OOT yang bertujuan melakukan pemeriksaan sistem yang dikembangkan baik kebutuhan fungsional maupun non-fungsional. Pengujian yang dilakukan terhadap kebutuhan fungsional yaitu pengujian unit, pengujian integrasi dan pengujian validasi. Pengujian yang dilakukan terhadap kebutuhan non-fungsional yaitu pengujian *compatibility*.

Pengujian *white-box* dilakukan untuk menguji *method* setiap unit serta menguji integrasi antar unit guna berjalan sesuai keinginan. Dalam pengujian *white-box* diperlukan pengujian *basis path* yang berguna menguji setiap jalur yang terdapat pada pengujian unit maupun integrasi. Tahap selanjutnya pengujian *black-box* dilakukan dengan menguji fungsional tanpa melihat kode program dengan harapan sesuai dengan skenario yang diharapkan. Pengujian *Compatibility* berguna mengetahui kompatibilitas sistem terhadap berbagai macam *browser*.

2.5 Kesimpulan dan Saran

Kesimpulan merupakan tahapan terakhir dengan melampirkan dokumentasi proyek yang dikembangkan. Kesimpulan dibuat untuk menjawab rumusan masalah. Saran bertujuan memperbaiki kesalahan dalam pengembangan aplikasi sehingga diperbaiki kedepannya serta untuk penyempurnaan dalam penulisan.

BAB 3 REKAYASA KEBUTUHAN

3.1 Gambaran Umum Sistem

Pengembangan sistem informasi manajemen penjualan suku cadang dan servis sepeda motor berbasis web (studi kasus anwar motor) merupakan sistem yang dibangun untuk membantu dan memudahkan pengguna dalam melakukan pembelian suku cadang dan jasa bengkel motor secara *online*. Di dalam sistem ini *manager* bengkel dapat melakukan pengecekan stok suku cadang untuk mengetahui suku cadang mana yang hampir habis. Di dalam akun tersebut terdapat fasilitas yang didapat oleh *manager* bengkel yakni melihat suku cadang di setiap cabang dan juga dapat menambahkan suku cadang dan jasa.

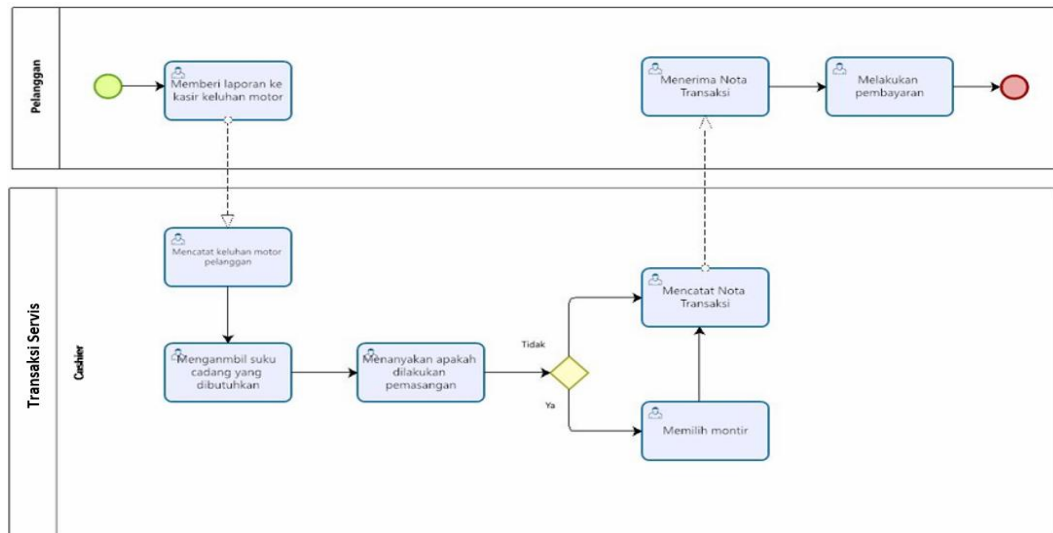
Untuk pengguna yang ingin bertindak sebagai *manager* dapat melihat grafik pemasukan dan pengeluaran, mengolah data stok suku cadang, melihat stok suku cadang di cabang lainnya, mengolah data *mechanic* dan melihat data pelanggan. Untuk pengguna yang ingin bertindak sebagai *cashier* dapat melihat data suku cadang, jasa, dan *mechanic*. Selanjutnya, *cashier* setiap dapat melakukan penjualan dengan fitur jual suku cadang dan jasa yang ada pada sistem, mengisi data atau keterangan mengenai data pelanggan yang membeli, mencetak struk pembelian suku cadang dan jasa berupa deskripsi suku cadang yang dibeli, jasa yang dibutuhkan dan total harga yang kemudian akan diberikan ke pelanggan. Untuk pengguna yang ingin bertindak sebagai *mechanic* dapat melihat pendapatan perhari berdasarkan servis yang sudah dilakukan.

3.2 Elisitasi Kebutuhan dan Analisis Kebutuhan

Tahap pertama merupakan elisitasi kebutuhan guna menggali permasalahan di Anwar Motor dengan menggunakan teknik wawancara. Narasumber dalam wawancara yaitu bapak Fitrul sebagai *manager* Anwar Motor, lampiran A merupakan hasil wawancara dengan *manager* dan bapak Mubin sebagai salah satu *mechanic* di Anwar Motor, dan lampiran B merupakan hasil dari wawancara dengan *mechanic*. Selain wawancara dilakukan pengamatan dokumen yaitu dokumen transaksi yang terkait dilihat pada Lampiran C.

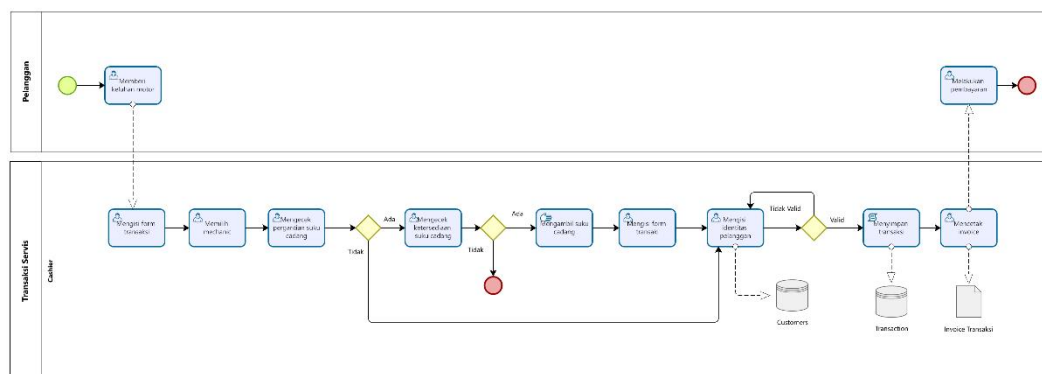
3.3 Pemodelan Proses Bisnis

Tahapan ini mengolah hasil wawancara dalam bentuk proses bisnis. Pada proyek ini BPMN (*Business Process Model Notation*) dibuat adalah 3 program yaitu penjualan suku cadang dan servis, pembelian suku cadang, dan peminjaman uang oleh *mechanic*. Proses bisnis terbagi dalam 2 tahap yaitu *As-Is* dan *To-Be*. Proses bisnis *As-Is* yaitu menerangkan proses bisnis sebelum terdapat sistem yang akan dikembangkan, untuk proyek ini terdapat Proses bisnis transaksi servis, peminjaman uang dan pembelian suku cadang. Proses bisnis *To-Be* menerangkan proses bisnis setelah terdapat sistem yang akan dikembangkan, untuk proyek ini terdapat proses bisnis transaksi servis, penjualan suku cadang, peminjaman uang dan pembelian suku cadang.



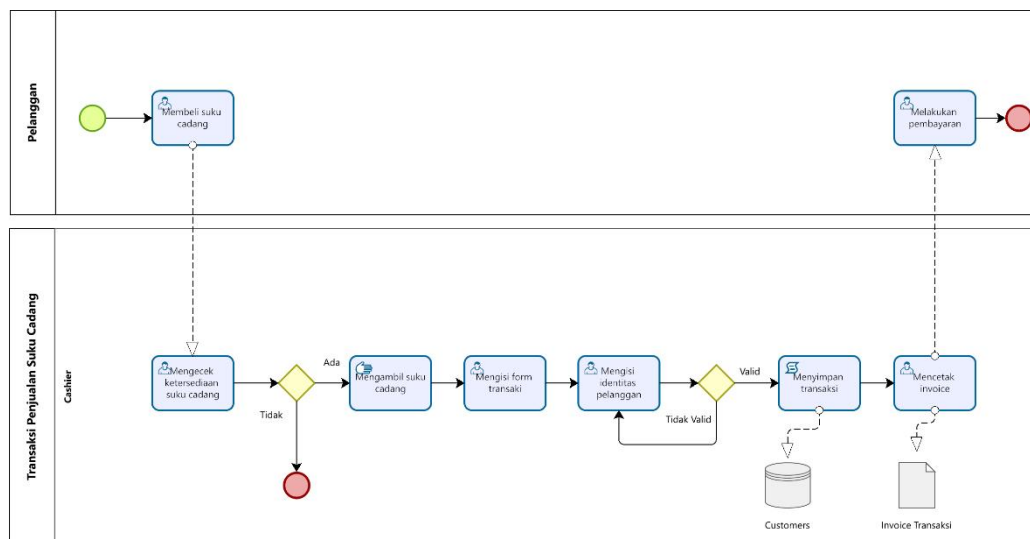
Gambar 4.1 Transaksi Bengkel As-Is

Proses bisnis Transaksi Bengkel As-Is diilustrasikan pada Gambar 4.1. Proses bisnis Transaksi Bengkel As-Is menggambarkan aktivitas dari pelayanan transaksi pada bengkel di awali memberi laporan ke kasir keluhan motor yang di akhiri melakukan pembayaran. Aktor yang terlibat dalam aktivitas ini adalah Pelanggan dan *Cashier*. Proses bisnis transaksi bengkel As-Is menjadi acuan untuk proses bisnis transaksi servis *To-be* dan proses bisnis penjualan suku cadang *To-be*.



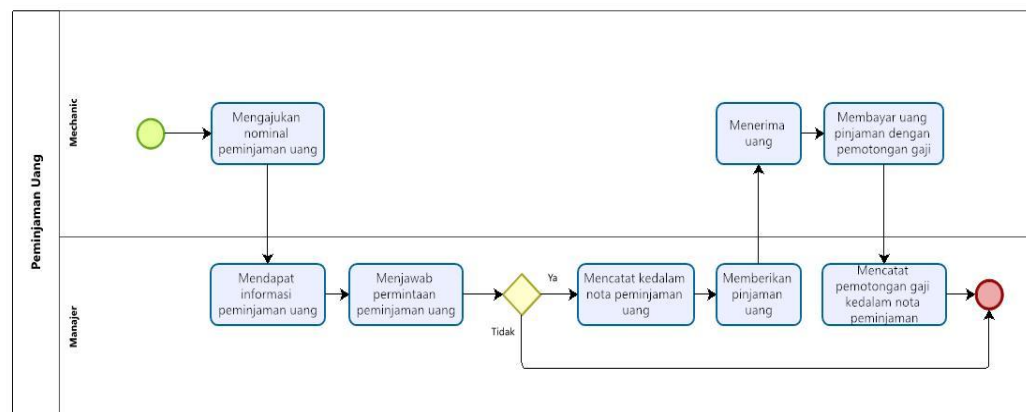
Gambar 4.2 Transaksi Servis To-Be

Proses bisnis Transaksi Servis *To-Be* diilustrasikan pada Gambar 4.2. Proses bisnis Transaksi Servis *To-Be* menggambarkan aktivitas dari pelayanan transaksi servis di awali memberi keluhan motor yang di akhiri melakukan pembayaran. Aktor yang terlibat dalam aktivitas ini adalah Pelanggan dan *Cashier*.



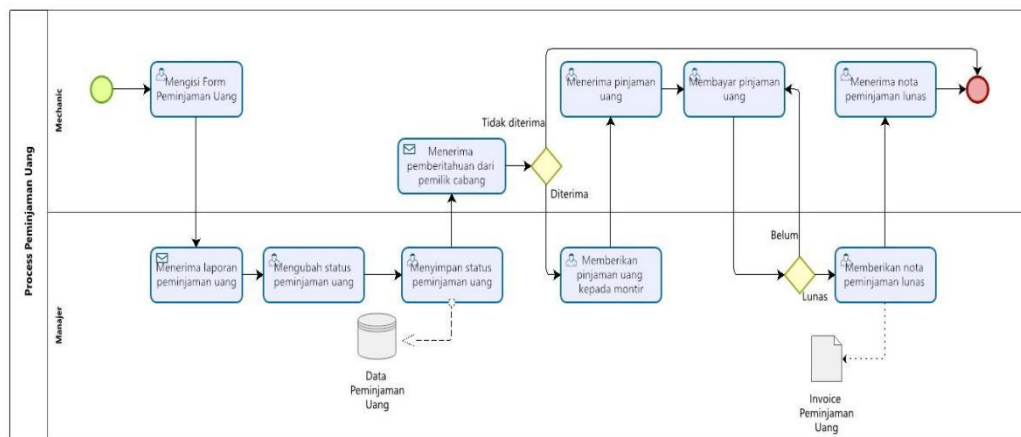
Gambar 4.3 Transaksi Penjualan Suku Cadang To-Be

Proses bisnis Transaksi Penjualan Suku Cadang *To-Be* diilustrasikan pada Gambar 4.3. Proses bisnis Transaksi Penjualan Suku Cadang *To-Be* menggambarkan aktivitas dari pelayanan transaksi penjualan suku cadang diawali membeli suku cadang yang diakhiri melakukan pembayaran. Aktor yang terlibat dalam aktivitas ini adalah Pelanggan dan *Cashier*.



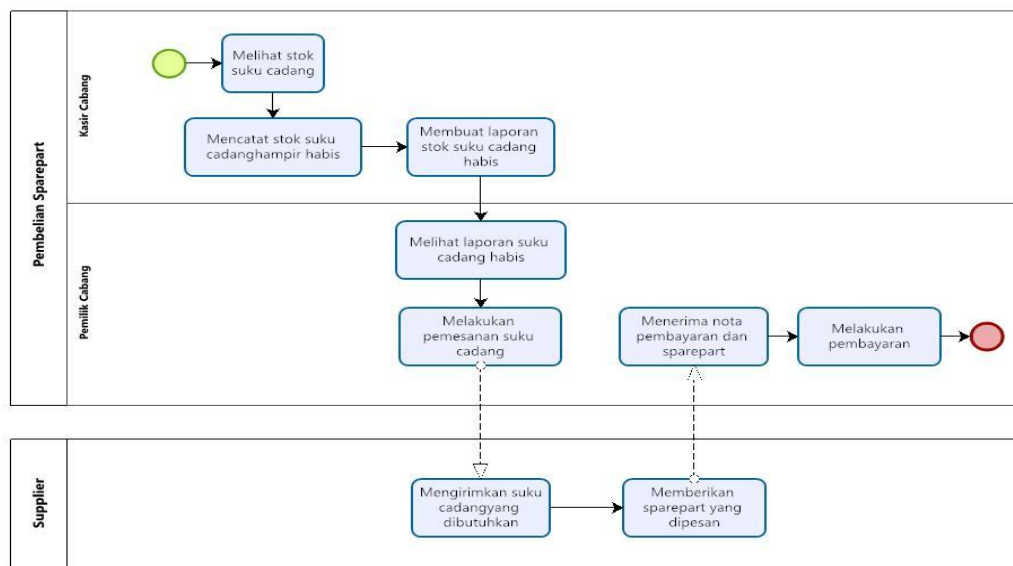
Gambar 4.4 Peminjaman Uang As-Is

Proses bisnis Peminjaman Uang *As-Is* diilustrasikan pada Gambar 4.4. Proses bisnis Peminjaman Uang *As-Is* menggambarkan aktivitas dari peminjaman uang diawali mengajukan nominal peminjaman uang yang diakhiri melakukan mencatat pemotongan gaji kedalam nota peminjaman. Aktor yang terlibat dalam aktivitas ini adalah Manager dan *Mechanic*.



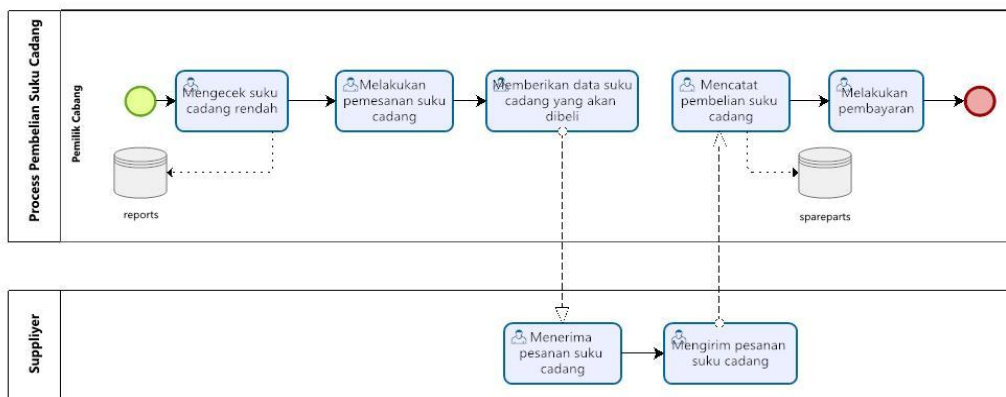
Gambar 4.5 Peminjaman Uang To-Be

Proses bisnis Peminjaman Uang *To-Be* diilustrasikan pada Gambar 4.5. Proses bisnis Peminjaman Uang *To-Be* menggambarkan aktivitas dari peminjaman uang diawali mengisi *form* peminjaman uang yang diakhiri menerima nota peminjaman uang lunas. Aktor yang terlibat dalam aktivitas ini adalah Manager dan *Mechanic*.



Gambar 4.6 Pembelian Suku Cadang As-Is

Proses bisnis Pembelian Suku Cadang *As-Is* diilustrasikan pada Gambar 4.6. Proses bisnis Pembelian Suku Cadang *As-Is* menggambarkan aktivitas dari pembelian suku cadang diawali melihat stok suku cadang yang diakhiri melakukan pembayaran. Aktor yang terlibat dalam aktivitas ini adalah Manager, Supplier dan *Cashier*.

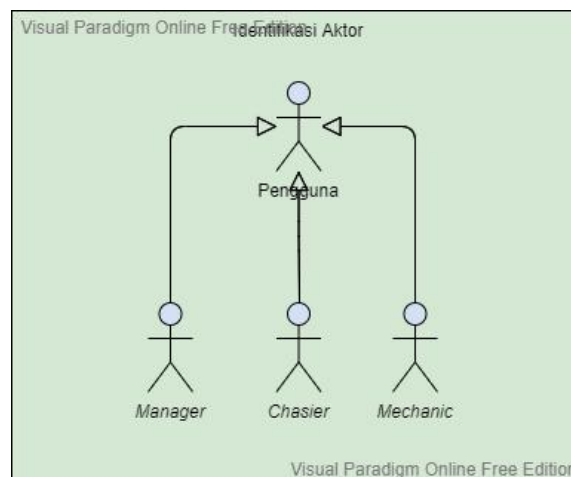


Gambar 4.7 Pembelian Suku Cadang To-Be

Proses bisnis Pembelian Suku Cadang *To-Be* diilustrasikan pada Gambar 4.7. Proses bisnis Pembelian Suku Cadang *To-Be* menggambarkan aktivitas dari pembelian suku cadang diawali mengecek suku cadang rendah yang diakhiri melakukan pembayaran. Aktor yang terlibat dalam aktivitas ini adalah Manager dan Supplier.

3.4 Identifikasi Aktor

Dalam tahap ini dilakukan pemetaan pengguna dalam pengoperasian sistem serta peranan masing-masing pengguna yang terlibat dalam sistem. Pada gambar 4.1 identifikasi aktor terdapat 3 aktor pengguna sistem, yakni *manager*, *cashier* dan *mechanic*.



Gambar 4.8 Deskripsi Aktor

Selanjutnya pada Tabel 4.1 memaparkan tentang peranan aktor yang terlibat sesuai dengan deskripsi. Pada proyek ini didapat dua aktor dengan dari hasil kebutuhan yang telah dilakukan.

Tabel 3.1 Identifikasi Aktor

Spesifikasi	Task
-------------	------

Pengguna	Pengguna merupakan aktor yang berada diluar sistem.
<i>Manager</i>	<i>Manager</i> disini bisa berfungsi sebagai pengelola suku cadang dan jasa pada sistem. Jika terdapat perubahan data suku cadang, jasa dan <i>mechanic</i> maka akan memiliki notifikasi kepada <i>manager</i> bahwa stok suku cadang, jasa dan <i>mechanic</i> terdapat pembaruan.
<i>Cashier</i>	<i>Cashier</i> disini bisa berfungsi sebagai pencatatan suku cadang dan jasa yang akan dijual pada sistem. Jika terdapat transaksi penjualan maka dapat melakukan memasukkan data suku cadang dan jasa yang akan dijual berdasarkan <i>mechanic</i> yang melakukan servis dan mencetak struk pembelian kepada pembeli.
<i>Mechanic</i>	<i>Mechanic</i> disini bisa berfungsi sebagai penyedia servis motor pada sistem. Jika terdapat transaksi penjualan maka dapat mendapatkan gaji berdasarkan pendapatan servis setiap motor.

3.5 Daftar Kebutuhan Fungsional Sistem

Di bagian daftar kebutuhan fungsional dari sistem yang ada pada Tabel 4.2 di bawah ini, akan dijelaskan mengenai kebutuhan yang harus ada pada perangkat lunak secara lengkap sesuai dengan peran aktor masing-masing beserta deskripsi.

Tabel 3.2 Kebutuhan Fungsional

No	Kode Kebutuhan	Nama Use Case	Deskripsi
1	SIM-F-001	<i>Login</i>	Sistem memiliki fungsi untuk memberi pengguna akses ke sistem.
			Spesifikasi : Pengguna dapat masuk ke sistem sesuai dengan <i>username</i> dan <i>password</i> yang tersedia.
2	SIM-F-002	Melihat Laporan suku cadang Rendah	Sistem memiliki fungsi untuk memperlihatkan semua laporan suku cadang rendah
			Spesifikasi : Data laporan suku cadang rendah ditampilkan berupa tanggal pelaporan, nama suku cadang, <i>supplier</i> , jumlah dan aksi.

3	SIM-F-003	Membeli suku cadang	Sistem memiliki fungsi untuk menambah pembelian suku cadang.
			Spesifikasi : <i>Form</i> pembelian suku cadang <i>supplier</i> , nama barang, jumlah, satuan, harga beli, harga jual, foto suku cadang, metode pembayaran, foto nota pembelian
4	SIM-F-004	Melihat suku cadang	Sistem memiliki fungsi untuk memperlihatkan semua suku cadang.
			Spesifikasi : Data suku cadang ditampilkan berupa nama suku cadang, kategori, <i>supplier</i> , jumlah, harga beli, harga jual, dan aksi.
5	SIM-F-005	Mengubah suku cadang	Sistem memiliki fungsi untuk mengubah suku cadang yang sudah tersimpan
			Spesifikasi : <i>Form</i> ubah suku cadang berupa nama suku cadang, kategori, <i>supplier</i> , jumlah, harga beli dan harga jual.
6	SIM-F-006	Menghapus suku cadang	Sistem memiliki fungsi untuk menghapus suku cadang yang sudah tersimpan.
7	SIM-F-007	Melihat servis	Sistem mampu untuk melihat servis
			Spesifikasi : Data servis ditampilkan berupa nama jasa dan biaya.
8	SIM-F-008	Menambah Servis	Sistem memiliki fungsi untuk menambahkan servis.
			Spesifikasi : <i>Form</i> tambah servis berupa nama jasa dan biaya.
9	SIM-F-009	Mengubah Servis	Sistem memiliki fungsi untuk mengubah servis yang sudah tersimpan.
			Spesifikasi : <i>Form</i> ubah servis berupa nama jasa dan harga.
10	SIM-F-010	Menghapus Servis	Sistem memiliki fungsi untuk menghapus jenis yang sudah tersimpan
11	SIM-F-011	Melihat <i>mechanic</i>	Sistem mampu untuk melihat <i>mechanic</i>
			Spesifikasi : Data <i>mechanic</i> yang ditampilkan berupa nama jasa dan biaya
12	SIM-F-012	Menambah <i>mechanic</i>	Sistem memiliki fungsi untuk menambahkan <i>mechanic</i>
			Spesifikasi : <i>Form</i> tambah <i>mechanic</i> berupa nama <i>mechanic</i> , mulai bekerja, <i>username</i> dan <i>password</i> .
13	SIM-F-013	Mengubah <i>mechanic</i>	Sistem memiliki fungsi untuk mengubah detail <i>mechanic</i> yang sudah tersimpan
			Spesifikasi : <i>Form</i> ubah <i>mechanic</i> berupa nama <i>mechanic</i> , mulai bekerja, <i>username</i> dan

			<i>password</i>
14	SIM-F-014	Menghapus <i>mechanic</i>	Sistem memiliki fungsi untuk menghapus detail <i>mechanic</i> yang sudah tersimpan
15	SIM-F-015	Melihat Gaji <i>mechanic</i>	Sistem memiliki fungsi untuk memperlihatkan gaji <i>mechanic</i> Spesifikasi : Data gaji <i>mechanic</i> ditampilkan berupa tanggal transaksi, nominal gaji dan aksi
16	SIM-F-016	Mengubah Status Peminjaman <i>mechanic</i>	Sistem memiliki fungsi untuk mengubah peminjaman yang sudah diajukan <i>mechanic</i> Spesifikasi : Data peminjaman <i>mechanic</i> ditampilkan tanggal pengajuan, nama <i>mechanic</i> , nominal, keterangan, status dan aksi. Aksi berisi tombol 'setuju' dan 'tidak' untuk mengubah status pinjaman <i>mechanic</i>
17	SIM-F-017	Menambah pelanggan	Sistem memiliki fungsi untuk menambahkan pelanggan Spesifikasi : Form tambah pelanggan berupa nama pelanggan, nomor telepon dan aksi.
18	SIM-F-018	Melihat pelanggan	Sistem memiliki fungsi untuk memperlihatkan semua pelanggan Spesifikasi : Data pelanggan ditampilkan berupa nama pelanggan, nomor telepon, aksi.
19	SIM-F-019	Mengubah pelanggan	Sistem memiliki fungsi untuk mengubah pelanggan yang sudah tersimpan Spesifikasi : Form ubah pelanggan berupa nama pelanggan, nomor telepon dan aksi.
20	SIM-F-021	Menghapus pelanggan	Sistem memiliki fungsi untuk menghapus pelanggan yang sudah tersimpan
21	SIM-F-021	Menambah Transaksi Pembelian suku cadang	Sistem memiliki fungsi untuk menambahkan transaksi suku cadang Spesifikasi : <i>Form</i> tambah transaksi berupa tanggal transaksi, nama pelanggan, nama <i>mechanic</i> , servis, suku cadang total transaksi, dan aksi (detail, delete, invoice). Mengisi seluruh <i>form</i> kecuali nama <i>mechanic</i> dan servis.
22	SIM-F-022	Melihat Transaksi Pembelian suku cadang	Sistem memiliki fungsi untuk memperlihatkan transaksi suku cadang Spesifikasi : Data transaksi ditampilkan berupa tanggal transaksi, nama pelanggan, total transaksi, dan aksi
23	SIM-F-023	Mengubah Transaksi Pembelian	Sistem memiliki fungsi untuk mengubah transaksi pembelian suku cadang Spesifikasi : <i>Form</i> ubah transaksi berupa tanggal

		suku cadang	transaksi, nama pelanggan, nama <i>mechanic</i> , servis, suku cadang total transaksi, dan aksi (detail, delete, invoice). Mengisi seluruh <i>form</i> tanpa nama <i>mechanic</i> dan servis.
24	SIM-F-024	Menghapus Transaksi Pembelian suku cadang	Sistem memiliki fungsi untuk menghapus transaksi suku cadang
25	SIM-F-025	Menambah Transaksi servis	<p>Sistem memiliki fungsi untuk menambahkan transaksi servis</p> <p>Spesifikasi : <i>Form</i> tambah transaksi berupa tanggal transaksi, nama pelanggan, nama <i>mechanic</i>, servis, suku cadang total transaksi, dan aksi (detail, delete, invoice). Mengisi seluruh <i>form</i> kecuali suku cadang.</p>
26	SIM-F-026	Melihat Transaksi servis	<p>Sistem memiliki fungsi untuk memperlihatkan transaksi servis</p> <p>Spesifikasi : Data transaksi ditampilkan berupa tanggal transaksi, nama pelanggan, total transaksi, dan aksi</p>
27	SIM-F-027	Mengubah Transaksi servis	<p>Sistem memiliki fungsi untuk mengubah transaksi servis</p> <p>Spesifikasi : <i>Form</i> edit transaksi berupa tanggal transaksi, nama pelanggan, nama <i>mechanic</i>, servis, suku cadang total transaksi, dan aksi (detail, delete, invoice). Mengisi seluruh <i>form</i> kecuali suku cadang.</p>
28	SIM-F-028	Menghapus Transaksi servis	Sistem memiliki fungsi untuk menghapus transaksi servis
29	SIM-F-029	Menambah Transaksi servis dan suku cadang	<p>Sistem memiliki fungsi untuk menambahkan transaksi servis dan suku cadang</p> <p>Spesifikasi : <i>Form</i> tambah transaksi berupa tanggal transaksi, nama pelanggan, nama <i>mechanic</i>, servis, suku cadang total transaksi, dan aksi (detail, delete, invoice). Mengisi seluruh <i>form</i> dengan lengkap</p>
30	SIM-F-030	Melihat Transaksi servis dan suku cadang	<p>Sistem memiliki fungsi untuk memperlihatkan transaksi servis dan suku cadang</p> <p>Spesifikasi : Data transaksi ditampilkan berupa tanggal transaksi, nama pelanggan, total transaksi, dan aksi</p>
31	SIM-F-031	Mengubah Transaksi	Sistem memiliki fungsi untuk mengubah transaksi suku cadang

		Pembelian suku cadang	Spesifikasi : <i>Form</i> ubah transaksi berupa tanggal transaksi, nama pelanggan, nama <i>mechanic</i> , servis, suku cadang total transaksi, dan aksi (detail, delete, invoice). Mengisi seluruh <i>form</i> dengan lengkap
32	SIM-F-032	Menghapus Transaksi servis dan suku cadang	Sistem memiliki fungsi untuk menghapus transaksi servis dan suku cadang
33	SIM-F-033	Menambah Peminjaman	Sistem memiliki fungsi untuk menambahkan peminjaman uang <i>mechanic</i> Spesifikasi : <i>Form</i> tambah peminjaman berupa nominal pinjaman, persentase pemotongan gaji, keterangan.
34	SIM-F-034	Melihat Pinjaman	Sistem memiliki fungsi untuk memperlihatkan peminjaman uang <i>mechanic</i> Spesifikasi : Data peminjaman ditampilkan
35	SIM-F-035	Mengubah Pinjaman	Sistem memiliki fungsi untuk mengubah peminjaman uang <i>mechanic</i> Spesifikasi : <i>Form</i> ubah peminjaman berupa nominal pinjaman, persentase pemotongan gaji, keterangan.
36	SIM-F-036	Menghapus Pinjaman	Sistem memiliki fungsi untuk menghapus peminjaman uang <i>mechanic</i>
37	SIM-F-037	Melihat Profil	sistem memiliki fungsi untuk melihat profil <i>mechanic</i> Spesifikasi : Data profil ditampilkan berupa nama <i>mechanic</i> , mulai bekerja, umur, detail, dan tombol ubah profil
38	SIM-F-038	Mengubah Profil	Sistem memiliki fungsi untuk mengubah profil <i>mechanic</i> Spesifikasi : <i>Form</i> edit profil ditampilkan berupa nama <i>mechanic</i> , tahun masuk, umur, alamat, password baru, password lama, tombol kembali, dan simpan
39	SIM-F-039	Mencetak Invoice Transaksi	Sistem mampu untuk mencetak <i>invoice</i> transaksi Spesifikasi : Data invoice atau nota transaksi disimpan lalu dicetak.
40	SIM-F-040	Logout	Sistem mampu untuk menghapus hak akses pengguna untuk mengakses sistem Spesifikasi : Pengguna dapat keluar dari sistem sesuai dan berada pada <i>form</i> login

41	SIM-F-41	Menambah Laporan suku cadang Rendah	Sistem mampu untuk menambah laporan suku cadang rendah
			Spesifikasi : Menekan tombol laporkan stok rendah pada halaman suku cadang dengan kondisi stok suku cadang ≤ 10

3.5.1 Daftar Kebutuhan Non-Fungsional Sistem

Pada Tabel 4.3 di bawah ini merupakan daftar kebutuhan non-fungsional yang mengacu pada kinerja sebuah sistem maupun kemudahan penggunaannya. Dan dijelaskan juga deskripsi dari masing-masing fungsinya.

Tabel 3.3 Kebutuhan Non-Fungsional Sistem

No	Kode Fungsi	Nama Fungsi	Deskripsi Fungsi
1	SIM-NF-001	Kompatibilitas	Sistem harus bisa berjalan di web peramban yang biasa digunakan oleh aktor-aktor yang ada yaitu <i>Google Chrome</i> dan <i>Microsoft Edge</i> .

3.6 Use Case Diagram

Diagram *use case* dipergunakan untuk membuat model *behavior* dari aktor terhadap sistem. Aktor yang terdapat pada proyek ini yaitu *manager*, *cashier*, dan *mechanic*. Perilaku untuk setiap aktor dijabarkan pada Gambar 3.8.

3.7 Use Case Scenario

Use Case Scenario memaparkan mengenai alur *use case* secara terperinci yang terdapat pada *use case diagram*. Penjelasan dapat dilihat pada Tabel 3.4 – 3.44.

3.7.1 Use Case Scenario Login

Tabel 3.4. memaparkan tentang *use case scenario login* yang dilakukan pengguna agar dapat masuk ke sistem

Tabel 3.4 Use Case Scenario Login

<i>Login</i>	
Kode Kebutuhan	SIM-F-001
Objektif	Memberi hak akses agar dapat masuk ke sistem
Aktor	Pengguna
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor mengisi <i>form login</i> berupa <i>username</i> dan <i>password</i> 2. Aktor menekan tombol <i>login</i> 3. Sistem menampilkan halaman <i>dashboard</i>
Aliran Alternatif	1. Sistem menampilkan pesan ' <i>username</i> atau <i>password</i> salah', jika <i>username</i> atau <i>password</i> tidak cocok dengan <i>database</i> . 2. Sistem menampilkan peringatan untuk mengisi, jika kolom <i>username</i> dan <i>password</i> tidak diisi
Kondisi Akhir	Halaman <i>dashboard</i> ditampilkan berdasarkan pengguna

3.7.2 Use Case Scenario Melihat Laporan Suku Cadang Rendah

Tabel 3.5 memaparkan tentang *use case scenario* melihat laporan suku cadang rendah yang dilakukan *manager* dan *cashier* guna melihat suku cadang rendah

Tabel 3.5 Use Case Scenario Melihat Laporan Suku Cadang Rendah

Melihat Laporan Suku Cadang Rendah	
Kode Kebutuhan	SIM-F-002
Objektif	Menampilkan laporan suku cadang rendah
Aktor	<i>manager</i> dan <i>cashier</i>

Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman Laporan Stok Rendah 2. Sistem menampilkan halaman Laporan Stok Rendah berupa tabel yang tersedia informasi tanggal pelaporan, nama suku cadang, <i>supplier</i>, jumlah dan aksi.
Aliran Alternatif	-
Kondisi Akhir	Halaman laporan stok rendah ditampilkan

3.7.3 Use Case Scenario Membeli Suku Cadang

Tabel 3.6 memaparkan tentang *use case scenario* membeli suku cadang yang dilakukan *manager* guna membeli suku cadang.

Tabel 3.6 Use Case Scenario Membeli Suku Cadang

Membeli Suku Cadang	
Kode Kebutuhan	SIM-F-003
Objektif	Membeli suku cadang
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman Pembelian suku cadang 2. Sistem menampilkan halaman Riwayat Pembelian suku cadang berupa tabel yang tersedia informasi tanggal transaksi, nama <i>supplier</i>, total transaksi dan aksi. 3. Aktor menekan tombol tambah 4. Aktor membuka halaman Pembelian suku cadang 5. Sistem menampilkan <i>form</i> pembelian suku cadang yang tersedia yaitu <i>supplier</i>, nama suku cadang, kategori, jumlah, satuan, harga beli, harga jual, foto suku cadang, dan tombol tambah 6. Aktor mengisi data pada <i>form</i> pembelian suku cadang dengan lengkap dan meminta sistem untuk memproses 7. Sistem menambahkan suku cadang ke dalam daftar pembelian 8. Aktor mengisi data nota pembayaran 9. Sistem menampilkan pesan data telah ditambahkan
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan "lengkapi data"

Kondisi Akhir	Menambah pembelian suku cadang dan terdapat pesan “pembelian berhasil ditambahkan” pada halaman pembelian suku cadang
---------------	-----------------------------------------------------------------------------------------------------------------------

3.7.4 Use Case Scenario Melihat Suku Cadang

Tabel 3.7 memaparkan *use case scenario* melihat suku cadang yang dilakukan *manager* dan *cashier* guna melihat stok suku cadang.

Tabel 3.7 Use Case Scenario Melihat Suku Cadang

Melihat Suku Cadang	
Kode Kebutuhan	SIM-F-004
Objektif	Melihat stok suku cadang
Aktor	<i>manager</i> dan <i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor memilih <i>navbar</i> suku cadang 2. Sistem menampilkan halaman stok suku cadang berupa tabel yang tersedia informasi nama suku cadang, harga jual, jumlah barang dan aksi.
Aliran Alternatif	-
Kondisi Akhir	Menampilkan stok suku cadang

3.7.5 Use Case Scenario Mengubah Suku Cadang

Tabel 3.8 memaparkan *use case scenario* mengubah suku cadang yang dilakukan *manager* guna mengubah detail dari suku cadang.

Tabel 3.8 Use Case Scenario Mengubah Suku Cadang

Mengubah Suku Cadang	
Kode Kebutuhan	SIM-F-005
Objektif	Mengubah suku cadang
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman suku cadang
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>edit</i> pada suku cadang yang dipilih 2. Sistem menampilkan <i>form</i> edit suku cadang yang tersedia yaitu nama suku cadang, kategori, jumlah, unit, harga beli,

	<p>harga jual dan tombol submit</p> <p>3. Aktor mengisi <i>form</i> mengubah suku cadang dengan lengkap dan meminta sistem untuk memproses</p> <p>4. Sistem menampilkan pesan data telah ditambahkan</p>
Aliran Alternatif	Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah suku cadang dan menyimpan ke dalam <i>database</i>

3.7.6 Use Case Scenario Menghapus Suku Cadang

Tabel 3.9 memaparkan *use case scenario* menghapus suku cadang yang dilakukan manager guna menghapus suku cadang yang sudah ada.

Tabel 3.9 Use Case Scenario Menghapus Suku Cadang

Menghapus Suku Cadang	
Kode Kebutuhan	SIM-F-006
Objektif	Menghapus suku cadang
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman suku cadang
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>delete</i> pada suku cadang yang dipilih 2. Sistem menampilkan konfirmasi yaitu “Apakah Anda yakin ingin menghapus data ini?” 3. Aktor menekan tombol ok dan meminta sistem untuk memproses 4. Sistem menampilkan pesan data telah dihapus
Aliran Alternatif	<ol style="list-style-type: none"> 1. Aktor menekan tombol batal dan suku cadang tidak jadi dihapus
Kondisi Akhir	Menghapus suku cadang dan terdapat pesan “suku cadang berhasil dihapus” pada halaman suku cadang

3.7.7 Use Case Scenario Melihat Servis

Tabel 3.10 memaparkan *use case scenario* melihat servis yang dilakukan oleh *manager* dan *cashier* guna melihat daftar servis.

Tabel 3.10 Use Case Scenario Melihat Servis

Melihat Servis	
Kode	SIM-F-007

Kebutuhan	
Objektif	Menampilkan servis
Aktor	<i>manager</i> dan <i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman servis 2. Sistem menampilkan halaman Daftar servis berupa tabel yang tersedia informasi nama servis, harga servis dan aksi.
Aliran Alternatif	-
Kondisi Akhir	Menampilkan seluruh servis

3.7.8 Use Case Scenario Menambah Servis

Tabel 3.11 memaparkan *use case scenario* menambah servis yang dilakukan oleh *manager* guna menambah servis baru ke dalam sistem.

Tabel 3.11 Use Case Scenario Menambah Servis

Menambah Servis	
Kode Kebutuhan	SIM-F-008
Objektif	Menambah servis
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman suku cadang
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol tambah data 2. Sistem menampilkan <i>form</i> tambah servis yang tersedia yaitu nama jasa, biaya, dan tombol tambah 3. Aktor mengisi data pada <i>form</i> tambah servis dengan lengkap dan meminta sistem untuk memproses 4. Sistem menampilkan pesan data telah ditambahkan dan menambahkan servis ke dalam daftar pembelian 5. Sistem menampilkan halaman servis
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan "lengkapi data"
Kondisi Akhir	Menambah servis dan terdapat pesan "servis berhasil ditambahkan" pada halaman servis

3.7.9 Use Case Scenario Mengubah Servis

Tabel 3.12 memaparkan *use case scenario* mengubah servis yang dilakukan *manager* guna mengubah servis yang sudah ada.

Tabel 3.12 Use Case Scenario Mengubah Servis

Mengubah Servis	
Kode Kebutuhan	SIM-F-009
Objektif	Mengubah servis
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman servis
Aliran Utama	<ol style="list-style-type: none">1. Aktor menekan tombol edit2. Sistem menampilkan <i>form</i> edit servis yang tersedia yaitu nama jasa, biaya, dan tombol tambah3. Aktor mengisi data pada <i>form</i> edit servis dengan lengkap dan meminta sistem untuk memproses4. Sistem menampilkan pesan data telah mengalami perubahan dan mengubah servis ke dalam daftar pembelian5. Sistem menampilkan halaman servis
Aliran Alternatif	<ol style="list-style-type: none">1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah servis dan terdapat pesan “servis berhasil diubah” pada halaman servis

3.7.10 Use Case Scenario Menghapus Servis

Use case scenario untuk menghapus servis dapat dilihat pada tabel 3.13 yang dilakukan oleh *manager* guna menghapus servis yang ada.

Tabel 3.13 Use Case Scenario Menghapus Servis

Menghapus Servis	
Kode Kebutuhan	SIM-F-010
Objektif	Menghapus servis
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman jasa
Aliran Utama	<ol style="list-style-type: none">1. Aktor menekan tombol hapus pada servis yang dipilih2. Sistem menampilkan konfirmasi

	3. Aktor menekan tombol ya, hapus data ini 4. Sistem menampilkan halaman servis
Aliran Alternatif	1. Aktor menekan tombol batal maka suku cadang tidak jadi dihapus dan menampilkan halaman servis
Kondisi Akhir	Menghapus servis dan terdapat pesan “servis berhasil dihapus” pada halaman servis

3.7.11 Use Case Scenario Melihat Mechanic

Tabel 3.14 memaparkan *use case scenario* melihat *mechanic* yang dilakukan *manager* guna melihat daftar *mechanic*.

Tabel 3.14 Use Case Scenario Melihat Mechanic

Melihat <i>Mechanic</i>	
Kode Kebutuhan	SIM-F-011
Objektif	Menampilkan data <i>mechanic</i>
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor membuka halaman <i>mechanic</i> 2. Aktor memilih menu daftar <i>mechanic</i>
Aliran Alternatif	-
Kondisi Akhir	Menampilkan data <i>mechanic</i> pada halaman <i>list</i> mekanik

3.7.12 Use Case Scenario Menambah Mechanic

Tabel 3.15 memaparkan *use case scenario* menambah *mechanic* yang dilakukan *manager* guna menambah *mechanic* baru ke sistem.

Tabel 3.15 Use Case Scenario Menambah Mechanic

Menambah <i>Mechanic</i>	
Kode Kebutuhan	SIM-F-012
Objektif	Menambah <i>mechanic</i>
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman <i>list</i> mekanik
Aliran	1. Aktor menekan tombol tambah data

Utama	<ol style="list-style-type: none"> 2. Aktor mengisi data pada form yang tersedia seperti nama <i>mechanic</i>, mulai bekerja, <i>username</i>, dan <i>password</i> 3. Aktor menekan tombol tambah data yang berada di bawah form 4. Sistem menampilkan pesan data telah ditambahkan
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Menambah <i>mechanic</i> dan terdapat pesan “ <i>mechanic</i> berhasil ditambahkan” pada halaman mekanik

3.7.13 Use Case Scenario Mengubah Mechanic

Tabel 3.16 memaparkan *use case scenario* untuk mengubah *mechanic* yang dilakukan *manager* guna mengubah *mechanic* yang sudah ada.

Tabel 3.16 Use Case Scenario Mengubah Mechanic

Mengubah <i>mechanic</i>	
Kode Kebutuhan	SIM-F-013
Objektif	Mengubah <i>mechanic</i>
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman <i>list</i> mekanik
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>update</i> pada <i>mechanic</i> yang dipilih 2. Aktor mengubah data yang dibutuhkan pada form yang muncul seperti nama <i>mechanic</i>, mulai bekerja, <i>username</i>, dan <i>password</i> 3. Aktor menekan tombol <i>update</i> data
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah <i>mechanic</i> dan terdapat pesan “ <i>mechanic</i> berhasil diubah” pada halaman mekanik

3.7.14 Use Case Scenario Menghapus Mechanic

Tabel 3.17 memaparkan *use case scenario* menghapus *mechanic* yang dilakukan oleh *manager* guna menghapus *mechanic* yang ada dalam sistem.

Tabel 3.17 Use Case Scenario Menghapus Mechanic

Menghapus <i>Mechanic</i>	
Kode	SIM-F-014

Kebutuhan	
Objektif	Menghapus <i>mechanic</i>
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman <i>list</i> mekanik
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>delete</i> pada <i>mechanic</i> yang dipilih 2. Sistem menampilkan konfirmasi 3. Aktor menekan tombol ya, hapus data ini
Aliran Alternatif	<ol style="list-style-type: none"> 1. Aktor menekan tombol batal dan <i>mechanic</i> tidak jadi dihapus
Kondisi Akhir	Menghapus <i>mechanic</i> dan terdapat pesan " <i>mechanic</i> berhasil dihapus" pada halaman mekanik

3.7.15 Use Case Scenario Melihat Gaji *Mechanic*

Tabel 3.18 memaparkan *use case scenario* melihat gaji *mechanic* yang dilakukan *mechanic* dan *manager* guna melihat gaji *mechanic*.

Tabel 3.18 Use Case Scenario Melihat Gaji *Mechanic*

Melihat Gaji <i>mechanic</i>	
Kode Kebutuhan	SIM-F-015
Objektif	Melihat gaji <i>mechanic</i>
Aktor	<i>Mechanic</i> dan <i>manager</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman gaji
Aliran Alternatif	-
Kondisi Akhir	Menampilkan data gaji <i>mechanic</i> pada halaman gaji

3.7.16 Use Case Scenario Mengubah Status Peminjaman *Mechanic*

Tabel 3.19 memaparkan *use case scenario* mengubah status peminjaman *mechanic*.

Tabel 3.19 Use Case Scenario Mengubah Status Peminjaman *Mechanic*

Mengubah Status Peminjaman <i>Mechanic</i>	
Kode	SIM-F-016

Kebutuhan	
Objektif	Mengubah status pinjaman <i>mechanic</i>
Aktor	<i>manager</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan <i>navbar mechanic</i> 2. Aktor menekan <i>navbar</i> pinjaman 3. Sistem menampilkan tabel data pinjaman berupa nomor, tanggal pengajuan, nama mekanik, nominal pinjaman, sisa pinjaman, catatan, status, dan aksi 4. Aktor menekan tombol setuju
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah status peminjaman <i>mechanic</i> dan terdapat pesan “status peminjaman <i>mechanic</i> berhasil diubah” pada halaman pinjaman

3.7.17 Use Case Scenario Menambah Pelanggan

Tabel 3.21 memaparkan *use case scenario* untuk menambah pelanggan.

Tabel 3.20 Use Case Scenario Menambah Pelanggan

Menambah Pelanggan	
Kode Kebutuhan	SIM-F-017
Objektif	Menambah pelanggan
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor memilih <i>navbar</i> pelanggan 2. Sistem menampilkan halaman pelanggan 3. Aktor <i>menekan</i> tombol tambah data 4. Sistem menampilkan <i>form</i> tambah pelanggan yang tersedia yaitu nama pelanggan, dan nomor telepon 5. Aktor mengisi data pada <i>form</i> tambah pelanggan yang tersedia yaitu nama pelanggan, dan nomor telepon 6. Aktor menekan tombol tambah data yang berada di bawah form 7. Sistem menampilkan pesan data telah ditambahkan
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”

Kondisi Akhir	Menambah pelanggan dan terdapat pesan “pelanggan berhasil ditambahkan” pada halaman pelanggan
---------------	-----------------------------------------------------------------------------------------------

3.7.18 Use Case Scenario Melihat Pelanggan

Tabel 3.21 memaparkan *use case scenario* melihat pelanggan.

Tabel 3.21 Use Case Scenario Melihat Pelanggan

Melihat Pelanggan	
Kode Kebutuhan	SIM-F-018
Objektif	Menampilkan semua data pelanggan kepada <i>cashier</i>
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor membuka halaman pelanggan
Aliran Alternatif	-
Kondisi Akhir	Menampilkan data pelanggan

3.7.19 Use Case Scenario Mengubah Pelanggan

Tabel 3.22 memaparkan *use case scenario* mengubah pelanggan.

Tabel 3.22 Use Case Scenario Mengubah Pelanggan

Mengubah Pelanggan	
Kode Kebutuhan	SIM-F-019
Objektif	Mengubah pelanggan
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman pelanggan
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>update</i> pada pelanggan yang dipilih 2. Aktor mengubah data yang dibutuhkan pada form ubah pelanggan 3. Sistem memproses <i>update</i> data
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”

Kondisi Akhir	Mengubah pelanggan dan terdapat pesan “pelanggan berhasil diubah” pada halaman pelanggan
---------------	------------------------------------------------------------------------------------------

3.7.20 Use Case Scenario Menghapus Pelanggan

Tabel 3.23 memaparkan *use case scenario* menghapus pelanggan.

Tabel 3.23 Use Case Scenario Menghapus Pelanggan

Menghapus Pelanggan	
Kode Kebutuhan	SIM-F-020
Objektif	Menghapus pelanggan
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman pelanggan
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>delete</i> pada <i>mechanic</i> yang dipilih 2. Sistem menampilkan konfirmasi 3. Aktor menekan tombol ya, hapus data ini
Aliran Alternatif	<ol style="list-style-type: none"> 1. <i>Cashier</i> menekan tombol batal dan pelanggan tidak jadi dihapus
Kondisi Akhir	Menghapus pelanggan dan terdapat pesan “pelanggan berhasil dihapus” pada halaman pelanggan

3.7.21 Use Case Scenario Menambah Transaksi Penjualan Suku Cadang

Tabel 3.24 memaparkan *use case scenario* menambah transaksi penjualan suku cadang.

Tabel 3.24 Use Case Scenario Menambah Transaksi Penjualan Suku Cadang

Menambah Transaksi Pembelian Suku Cadang	
Kode Kebutuhan	SIM-F-021
Objektif	Menambah transaksi penjualan suku cadang
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi 3. Aktor menekan tombol tambah transaksi Penjualan 4. Sistem menampilkan form transaksi yang tersedia seperti nama suku cadang, servis, <i>mechanic</i>, pelanggan dan tombol tambah

	5. Aktor mengisi data pada form transaksi seperti suku cadang dan nama pelanggan dan menekan tombol simpan 6. Sistem menambahkan data transaksi dan menampilkan pesan data transaksi telah ditambahkan
Aliran Alternatif	1. Apabila nama pelanggan tidak diisi maka sistem akan menampilkan pesan untuk melengkapi
Kondisi Akhir	Menambah transaksi penjualan suku cadang “transaksi berhasil ditambahkan” pada halaman transaksi

3.7.22 Use Case Scenario Melihat Transaksi Penjualan Suku Cadang

Tabel 3.25 memaparkan *use case scenario* melihat transaksi penjualan suku cadang.

Tabel 3.25 Use Case Scenario Melihat Transaksi Penjualan Suku Cadang

Melihat Transaksi Penjualan Suku Cadang	
Kode Kebutuhan	SIM-F-022
Objektif	Mengubah melihat transaksi penjualan suku cadang
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi
Aliran Alternatif	-
Kondisi Akhir	Menampilkan data transaksi pada halaman transaksi

3.7.23 Use Case Scenario Mengubah Transaksi Pembelian Suku Cadang

Tabel 3.26 mengubah *use case scenario* mengubah transaksi pembelian suku cadang.

Tabel 3.26 Use Case Scenario Mengubah Transaksi Pembelian Suku Cadang

Mengubah Transaksi Pembelian suku cadang	
Kode Kebutuhan	SIM-F-023
Objektif	Mengubah transaksi pembelian suku cadang
Aktor	<i>cashier</i>
Kondisi	Berada di halaman <i>dashboard</i>

Awal	
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi 3. Aktor menekan tombol edit transaksi Penjualan 4. Sistem menampilkan form transaksi yang tersedia seperti nama suku cadang, servis, <i>mechanic</i>, pelanggan dan tombol tambah 5. Aktor mengisi data pada form transaksi seperti suku cadang dan nama pelanggan dan menekan tombol simpan 6. Sistem menambahkan data transaksi dan menampilkan pesan data transaksi telah ditambahkan
Aliran Alternatif	<ol style="list-style-type: none"> 1. Apabila nama pelanggan tidak diisi maka sistem akan menampilkan pesan untuk melengkapi
Kondisi Akhir	Mengubah transaksi penjualan suku cadang “transaksi berhasil diubah” pada halaman transaksi

3.7.24 Use Case Scenario Menghapus Transaksi Pembelian Suku Cadang

Tabel 3.27 memaparkan *use case scenario* menghapus transaksi pembelian suku cadang.

Tabel 3.27 Use Case Scenario Menghapus Transaksi Pembelian Suku Cadang

Menghapus Transaksi Pembelian suku cadang	
Kode Kebutuhan	SIM-F-024
Objektif	Menghapus transaksi pembelian suku cadang
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman Transaksi
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>delete</i> pada transaksi yang dipilih 2. Sistem menampilkan konfirmasi 3. Aktor menekan tombol ya, hapus data ini
Aliran Alternatif	<ol style="list-style-type: none"> 1. Aktor menekan tombol batal maka transaksi tidak terhapus
Kondisi Akhir	Menghapus transaksi penjualan suku cadang “transaksi berhasil dihapus” pada halaman transaksi

3.7.25 Use Case Scenario Menambah Transaksi Servis

Tabel 3.28 memaparkan *use case scenario* menambah transaksi servis.

Tabel 3.28 Use Case Scenario Menambah Transaksi Servis

Menambah Transaksi Servis	
Kode Kebutuhan	SIM-F-025
Objektif	Menambah transaksi servis
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi 3. Aktor menekan tombol tambah transaksi Penjualan 4. Sistem menampilkan form transaksi yang tersedia seperti nama suku cadang, servis, <i>mechanic</i>, pelanggan dan tombol tambah 5. Aktor mengisi data pada form transaksi seperti servis, <i>mechanic</i> dan nama pelanggan dan menekan tombol simpan 6. Sistem menambahkan data transaksi dan menampilkan pesan data transaksi telah ditambahkan
Aliran Alternatif	<ol style="list-style-type: none"> 1. Apabila Nama pelanggan dan <i>mechanic</i> tidak diisi maka sistem akan menampilkan pesan untuk melengkapi
Kondisi Akhir	Menambah transaksi penjualan servis “transaksi berhasil ditambahkan” pada halaman transaksi

3.7.26 Use Case Scenario Melihat Transaksi Servis

Use case scenario untuk Melihat Transaksi Servis dapat dilihat pada tabel 3.29.

Tabel 3.29 Use Case Scenario Melihat Transaksi Servis

Melihat Transaksi servis	
Kode Kebutuhan	SIM-F-026
Objektif	Untuk mengubah Melihat Transaksi servis
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi
Aliran	-

Alternatif	
Kondisi Akhir	Menampilkan data transaksi

3.7.27 Use Case Scenario Mengubah Transaksi Servis

Tabel 3.30 memaparkan *use case scenario* mengubah transaksi servis.

Tabel 3.30 Use Case Scenario Mengubah Transaksi Servis

Menghapus Transaksi Servis	
Kode Kebutuhan	SIM-F-027
Objektif	Mengubah transaksi servis
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman Transaksi
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi 3. Aktor menekan tombol edit transaksi Penjualan 4. Sistem menampilkan form transaksi yang tersedia seperti nama suku cadang, servis, <i>mechanic</i>, pelanggan dan tombol tambah 5. Aktor mengedit data pada form transaksi seperti servis, <i>mechanic</i> dan nama pelanggan dan menekan tombol simpan 6. Sistem menambahkan data transaksi dan menampilkan pesan data transaksi telah diubah
Aliran Alternatif	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah transaksi servis “transaksi berhasil diubah” pada halaman transaksi

3.7.28 Use Case Scenario Menghapus Transaksi Servis

Tabel 3.31 memaparkan *use case scenario* menghapus transaksi servis.

Tabel 3.31 Use Case Scenario Menghapus Transaksi Servis

Menghapus Transaksi Servis	
Kode Kebutuhan	SIM-F-028
Objektif	Menghapus transaksi servis
Aktor	<i>cashier</i>

Kondisi Awal	Berada di halaman Transaksi
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor menekan tombol <i>delete</i> pada transaksi yang dipilih 2. Sistem menampilkan konfirmasi 3. Aktor menekan tombol ya, hapus data ini
Aliran Alternatif	<ol style="list-style-type: none"> 1. Aktor menekan tombol batal dan transaksi tidak terhapus
Kondisi Akhir	Menghapus transaksi servis “transaksi berhasil dihapus” pada halaman transaksi

3.7.29 Use Case Scenario Menambah Transaksi Servis dan Suku Cadang

Tabel 3.32 memaparkan *use case scenario* menambah transaksi servis dan suku cadang.

Tabel 3.32 Use Case Scenario Menambah Transaksi Servis dan Suku Cadang

Menambah Transaksi Servis Dan Suku Cadang	
Kode Kebutuhan	SIM-F-029
Objektif	Menambah transaksi servis dan suku cadang
Aktor	<i>cashier</i>
Kondisi Awals	Berada di halaman transaksi
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman tambah data transaksi 2. Sistem menampilkan <i>form</i> tambah data transaksi yang tersedia yaitu nama pelanggan, nama <i>mechanic</i>, nama suku cadang, dan nama Jasa. 3. Aktor mengisi data pada <i>form</i> tambah data transaksi dengan lengkap dan meminta sistem untuk memproses 4. Sistem menampilkan pesan data telah ditambahkan
Aliran Alternatifs	<ol style="list-style-type: none"> 1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhirs	Menambah transaksi servis dan suku cadang dan terdapat pesan “transaksi berhasil ditambahkan” pada halaman transaksi

3.7.30 Use Case Scenario Melihat Transaksi Servis dan Suku Cadang

Tabel 3.33 memaparkan *use case scenario* melihat transaksi servis dan suku cadang.

Tabel 3.33 Use Case Scenario Melihat Transaksi Servis dan Suku Cadang

Melihat Transaksi Servis Dan Suku Cadang	
Kode Kebutuhan	SIM-F-030
Objektif	Melihat transaksi servis dan suku cadang
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi
Aliran Alternatif	-
Kondisi Akhir	Menampilkan data transaksi

3.7.31 Use Case Scenario Mengubah Transaksi Servis dan Suku Cadang

Tabel 3.34 memaparkan *use case scenario* mengubah transaksi servis dan suku cadang.

Tabel 3.34 Use Case Scenario Mengubah Transaksi Servis dan Suku Cadang

Mengubah Transaksi Servis Dan Suku Cadang	
Kode Kebutuhan	SIM-F-031
Objektif	Mengubah transaksi servis dan suku cadang
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor membuka halaman tambah data transaksi 2. Sistem menampilkan <i>form</i> edit data transaksi yang tersedia yaitu nama pelanggan, nama <i>mechanic</i> , nama suku cadang, dan nama Jasa. 3. Aktor mengisi data pada <i>form</i> edit data transaksi dengan lengkap dan meminta sistem untuk memproses
Aliran Alternatif	1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah transaksi servis dan suku cadang dan terdapat pesan “transaksi berhasil diubah” pada halaman transaksi

3.7.32 Use Case Scenario Menghapus Transaksi Servis dan Suku Cadang

Tabel 3.35 memaparkan *use case scenario* menghapus transaksi servis dan suku cadang.

Tabel 3.35 Use Case Scenario Menghapus Transaksi Servis dan Suku Cadang

Menghapus Transaksi Servis Dan Suku Cadang	
Kode Kebutuhan	SIM-F-032
Objektif	Menghapus transaksi servis
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman Transaksi servis
Aliran Utama	<ol style="list-style-type: none">1. Aktor menekan tombol <i>delete</i> pada transaksi yang dipilih2. Sistem menampilkan konfirmasi3. Aktor menekan tombol ya, hapus data ini
Aliran Alternatif	<ol style="list-style-type: none">1. Aktor menekan tombol batal dan Transaksi tidak jadi dihapus
Kondisi Akhir	Menghapus transaksi servis dan suku cadang dan terdapat pesan “transaksi berhasil dihapus” pada halaman transaksi

3.7.33 Use Case Scenario Menambah Pinjaman

Tabel 3.36 memaparkan *use case scenario* menambah pinjaman dapat.

Tabel 3.36 Use Case Scenario Menambah Pinjaman

Menambah Pinjaman	
Kode Kebutuhan	SIM-F-033
Objektif	Menambah pinjaman
Aktor	<i>mechanic</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	<ol style="list-style-type: none">1. Aktor membuka halaman pinjaman2. Sistem menampilkan halaman pinjaman3. Aktor menekan tombol ajukan pinjaman4. Sistem menampilkan form yang tambah pinjaman, yaitu nominal pinjaman, presentase potongan gaji, dan keterangan5. Aktor mengisi data pada form yang tambah pinjaman, yaitu nominal pinjaman, presentase potongan gaji, dan keterangan

	6. Aktor menekan tombol tambah data yang berada di bawah form 7. Sistem menampilkan pesan data telah ditambahkan
Aliran Alternatif	1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Menambah pinjaman dan terdapat pesan “pinjaman berhasil ditambahkan” pada halaman pinjaman

3.7.34 Use Case Scenario Melihat Pinjaman

Tabel 3.37 memaparkan *use case scenario* melihat pinjaman.

Tabel 3.37 Use Case Scenario Melihat Pinjaman

Melihat Pinjaman	
Kode Kebutuhan	SIM-F-034
Objektif	Melihat pinjaman
Aktor	<i>mechanic</i>
Kondisi Awal	Berada di halaman <i>dashboard</i>
Aliran Utama	1. Aktor membuka halaman pinjaman 2. Sistem menampilkan halaman pinjaman
Aliran Alternatif	-
Kondisi Akhir	Menampilkan data pinjaman

3.7.35 Use Case Scenario Mengubah Pinjaman

Tabel 3.38 memaparkan *use case scenario* mengubah pinjaman dapat.

Tabel 3.38 Use Case Scenario Mengubah Pinjaman

Mengubah Pinjaman	
Kode Kebutuhan	SIM-F-035
Objektif	Mengubah pinjaman
Aktor	<i>mechanic</i>
Kondisi Awal	Berada di halaman Pinjaman
Aliran Utama	1. Aktor membuka halaman pinjaman 2. Sistem menampilkan halaman pinjaman

	<ol style="list-style-type: none"> Aktor menekan tombol edit Sistem menampilkan form edit data pinjaman, yaitu nominal pinjaman, presentase potongan gaji, dan keterangan Aktor mengisi data pada form edit data pinjaman, yaitu nominal pinjaman, presentase potongan gaji, dan keterangan Aktor menekan tombol tambah data yang berada di bawah form
Aliran Alternatif	<ol style="list-style-type: none"> Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah pinjaman dan terdapat pesan “pinjaman berhasil diubah” pada halaman pinjaman

3.7.36 Use Case Scenario Menghapus Pinjaman

Tabel 3.39 memaparkan *use case scenario* menghapus pinjaman.

Tabel 3.39 Use Case Scenario Menghapus Pinjaman

Menghapus Pinjaman	
Kode Kebutuhan	SIM-F-036
Objektif	Menghapus pinjaman
Aktor	<i>mechanic</i>
Kondisi Awal	Berada di halaman Pinjaman
Aliran Utama	<ol style="list-style-type: none"> Aktor menekan tombol <i>delete</i> pada pinjaman yang belum disetujui Sistem menampilkan konfirmasi Aktor menekan tombol ya, hapus data ini, data akan diproses
Aliran Alternatif	<ol style="list-style-type: none"> Aktor menekan tombol batal dan Transaksi tidak jadi dihapus
Kondisi Akhir	Menghapus pinjaman dan terdapat pesan “pinjaman berhasil dihapus” pada halaman pinjaman

3.7.37 Use Case Scenario Melihat Profil

Tabel 3.40 memaparkan *use case scenario* melihat profil.

Tabel 3.40 Use Case Scenario Melihat Profil

Melihat Profil	
Kode Kebutuhan	SIM-F-037
Objektif	Melihat profil
Aktor	<i>mechanic</i>
Kondisi Awal	Berada di halaman <i>dashboard mechanic</i>
Aliran Utama	1. Aktor membuka halaman profil 2. Sistem menampilkan halaman profil
Aliran Alternatif	-
Kondisi Akhir	Menampilkan profil mekanik

3.7.38 Use Case Scenario Mengubah Profil

Tabel 3.41 memaparkan *use case scenario* mengubah profil.

Tabel 3.41 Use Case Scenario Mengubah Profil

Mengubah Profil	
Kode Kebutuhan	SIM-F-038
Objektif	Mengubah profil <i>mechanic</i>
Aktor	<i>mechanic</i>
Kondisi Awal	Berada di halaman Profil
Aliran Utama	1. Aktor menekan tombol ubah profil 2. Sistem menampilkan halaman ubah profil pengguna 3. Aktor mengubah data yang dibutuhkan pada form ubah profil pengguna yaitu nama, umur, alamat, <i>password</i> baru dan <i>password</i> lama. Dan menekan tombol simpan 4. Sistem menyimpan perubahan data profil <i>mechanic</i>
Aliran Alternatif	1. Terdapat salah satu form yang tidak diisi maka sistem menampilkan pesan “lengkapi data”
Kondisi Akhir	Mengubah profil dan terdapat pesan “profil berhasil diubah” pada halaman profil

3.7.39 Use Case Scenario Mencetak Invoice Transaksi

Tabel 3.42 *use case scenario* mencetak *invoice* transaksi.

Tabel 3.42 Use Case Scenario Mencetak Invoice Transaksi

Mencetak <i>Invoice</i> Transaksi	
Kode Kebutuhan	SIM-F-039
Objektif	Mencetak <i>invoice</i> transaksi
Aktor	<i>cashier</i>
Kondisi Awal	Berada di halaman <i>Dashboard</i>
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor membuka halaman transaksi 2. Sistem menampilkan halaman transaksi 3. Aktor menekan tombol <i>invoice</i> 4. Sistem mengunduh file <i>invoice</i> 5. Aktor menyimpan dan mencetak <i>invoice</i>
Aliran Alternatif	<ol style="list-style-type: none"> 1. Jika unduhan tidak disimpan maka <i>invoice</i> tidak akan tercetak
Kondisi Akhir	Mencetak dan menyimpan <i>Invoice</i> transaksi

3.7.40 Use Case Scenario Logout

Tabel 3.43 memaparkan *use case scenario logout*.

Tabel 3.43 Use Case Scenario Logout

Logout	
Kode Kebutuhan	SIM-F-040
Objektif	Menghilangkan hak akses user dari sistem
Aktor	<i>manager , cashier , mechanic</i>
Kondisi Awal	Berada di halaman sistem
Aliran Utama	<ol style="list-style-type: none"> 1. Aktor memilih profil dan menekan tombol logout 2. Sistem mengeluarkan pengguna dan mengarahkan ke halaman <i>login</i>
Aliran Alternatif	-
Kondisi Akhir	Aktor diarahkan ke halaman <i>login</i>

3.7.41 Use Case Scenario Menambah Laporan Stok Rendah

Tabel 3.44 *use case scenario* menambah laporan stok rendah.

Tabel 3.44 Use Case Scenario Laporan Stok Rendah

Laporan Stok Rendah	
Kode Kebutuhan	SIM-F-041
Objektif	Menambah Laporan stok rendah
Aktor	<i>manager</i>
Kondisi Awal	Untuk menampilkan laporan stok rendah
Aliran Utama	<ol style="list-style-type: none">1. Aktor memilih <i>navbar</i> suku cadang2. Aktor memilih <i>navbar</i> daftar suku cadang3. Sistem menampilkan <i>list</i> suku cadang berupa no, nama suku cadang, harga jual, jumlah barang, aksi4. Aktor menekan kolom jumlah barang5. Sistem mengurutkan barang mulai dari terendah6. Aktor menekan tombol “Lapor Stok Rendah”
Aliran Alternatif	-
Kondisi Akhir	Menambah laporan stok rendah dan terdapat pesan “suku cadang berhasil dilaporkan” pada halaman <i>list</i> suku cadang

3.7.42 Use Case Scenario Menambah Laporan Stok Rendah

Tabel 3.45 use case scenario menambah laporan stok rendah.

Tabel 3.45 Use Case Scenario Laporan Stok Rendah

Laporan Stok Rendah	
Kode Kebutuhan	SIM-F-041
Objektif	Menambah Laporan stok rendah
Aktor	<i>manager</i>
Kondisi Awal	Untuk menampilkan laporan stok rendah
Aliran Utama	<ol style="list-style-type: none">1. Aktor memilih <i>navbar</i> suku cadang2. Aktor memilih <i>navbar</i> daftar suku cadang3. Sistem menampilkan <i>list</i> suku cadang berupa no, nama suku cadang, harga jual, jumlah barang, aksi4. Aktor menekan kolom jumlah barang5. Sistem mengurutkan barang mulai dari terendah6. Aktor menekan tombol “Lapor Stok Rendah”
Aliran	-

Alternatif	
Kondisi Akhir	Menambah laporan stok rendah dan terdapat pesan “suku cadang berhasil dilaporkan” pada halaman <i>list</i> suku cadang

BAB 4 PERANCANGAN DAN IMPLEMENTASI

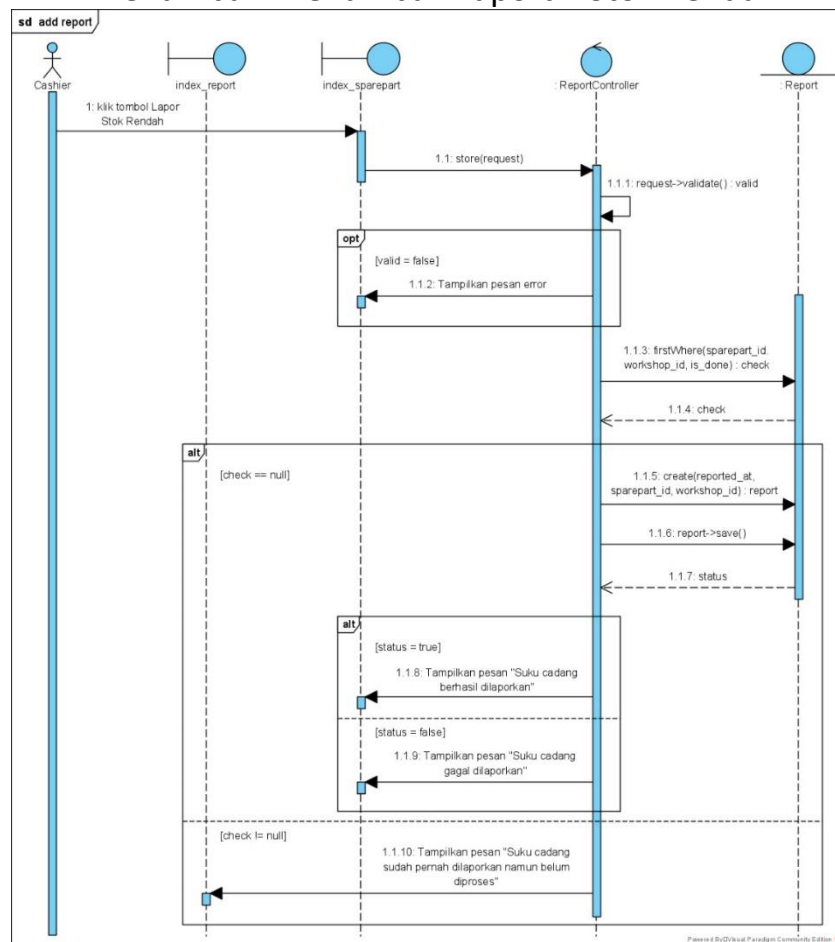
4.1 Perancangan

4.1.1 Perancangan Arsitektur

Pemodelan *sequence diagram* dan *class diagram* digunakan pada perancangan arsitektur.

4.1.1.1 Sequence Diagram

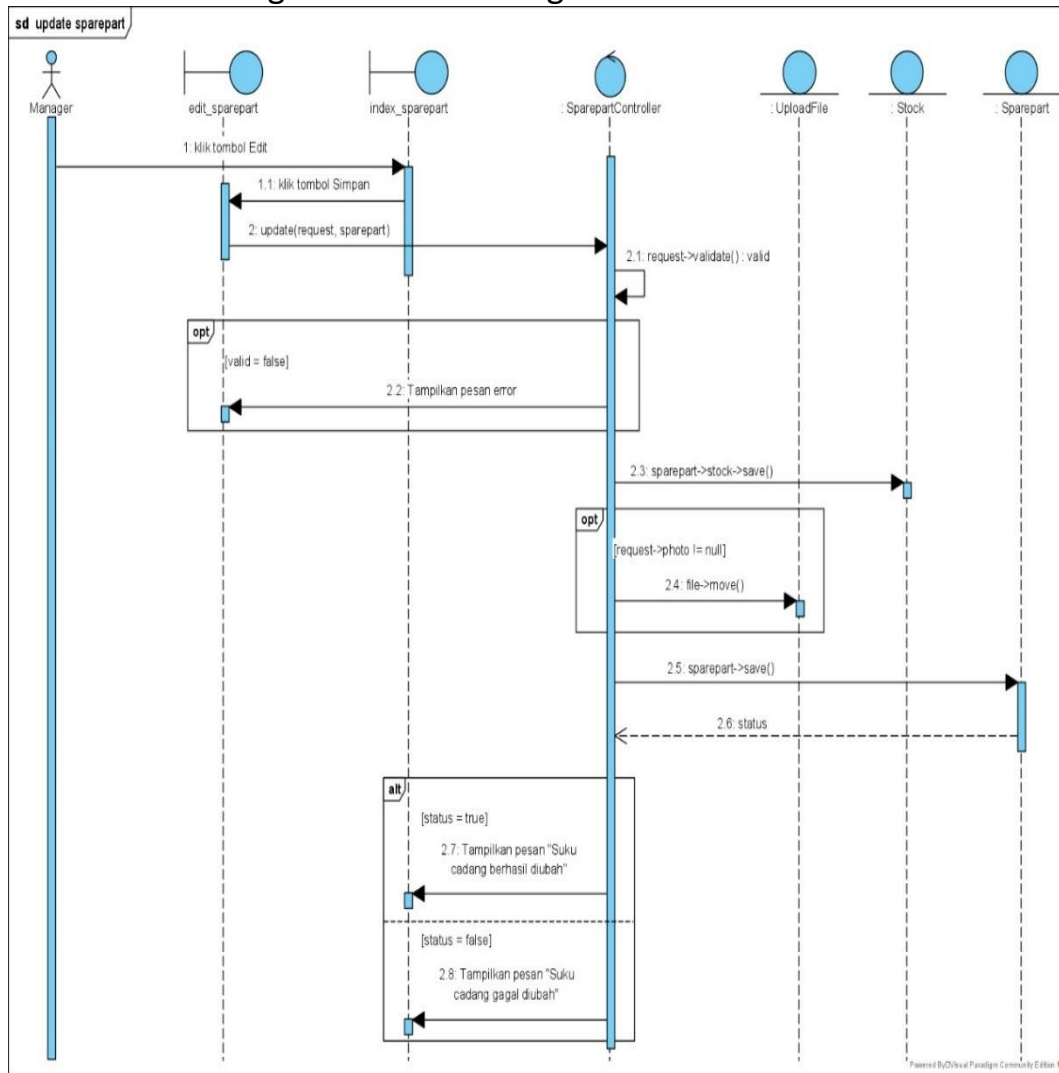
1. Menambah Menambah Laporan Stok Rendah



Gambar 4.1 Sequence Menambah Laporan Stok Rendah

Sequence Diagram untuk Menambah Laporan Stok Rendah diilustrasikan pada Gambar 4.1. *Sequence diagram* memiliki dua *boundary* yaitu *index_report* dan *index_sparepart*, satu objek *controller* yaitu *ReportController* dan satu objek *entity* yaitu *Report*. Aktor yang terdapat pada *sequence diagram* ini yaitu *Cashier*.

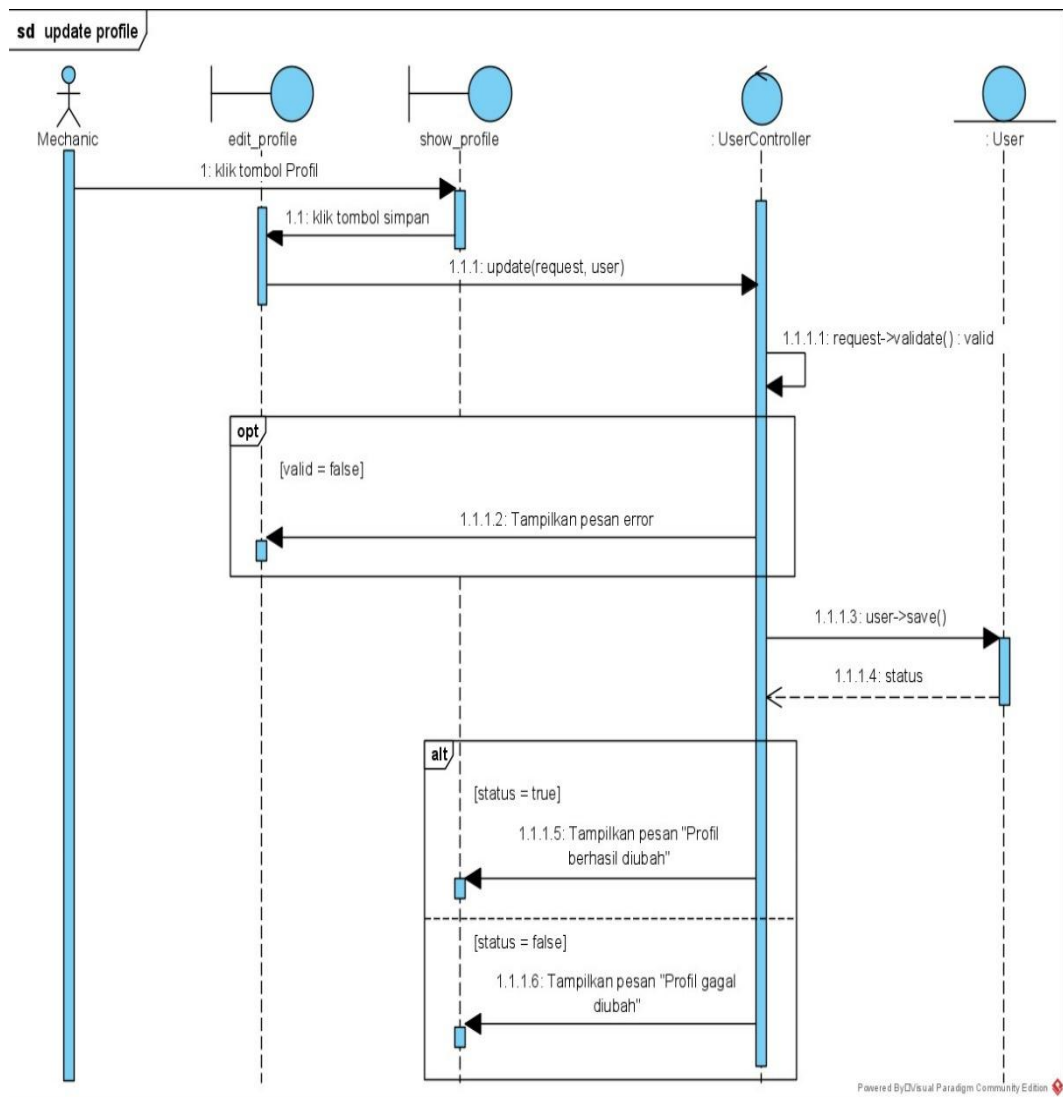
2. Mengubah Suku Cadang



Gambar 4.2 Sequence Diagram Mengubah Suku Cadang

Sequence Diagram untuk Mengubah Suku Cadang diilustrasikan pada Gambar 4.2. *Sequence diagram* memiliki dua *boundary* yaitu *edt_sparepart* dan *index_sparepart*, satu objek *controller* yaitu *SparepartController* dan tiga objek *entity* yaitu *UploadFile*, *Stock* dan *Sparepart*. Aktor yang terdapat pada *sequence diagram* ini yaitu *Manager*.

3. Mengubah Profil

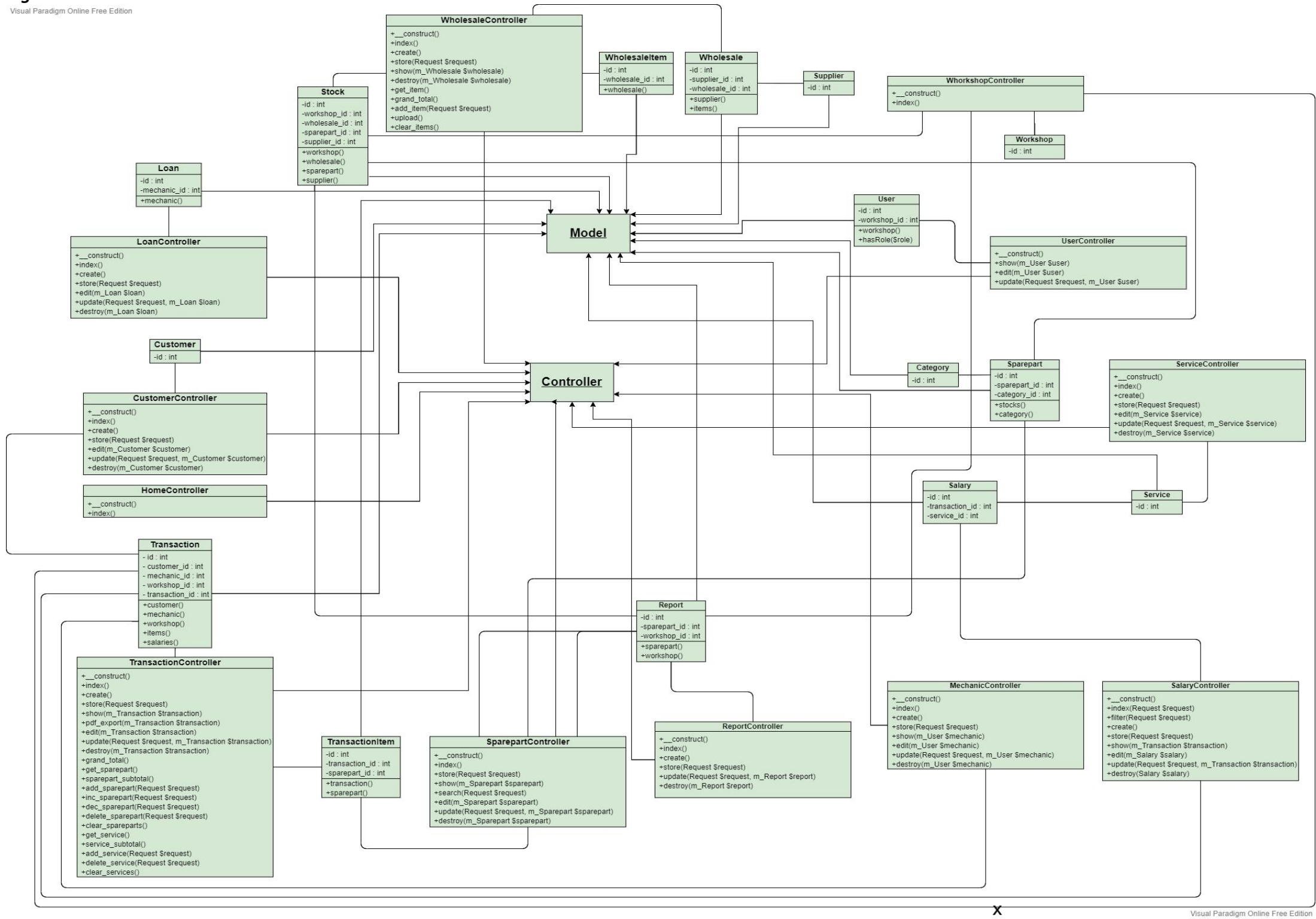


Gambar 4.3 Sequence Mengubah Profil

Sequence Diagram untuk Mengubah Profil diilustrasikan pada Gambar 4.3. *Sequence diagram* memiliki dua *boundary* yaitu *edt_profil* dan *show_profile*, satu objek *controller* yaitu *UserController* dan satu objek *entity* yaitu *User*. Aktor yang terdapat pada *sequence diagram* ini yaitu *Mechanic*.

4.1.1.2 Class Diagram

Visual Paradigm Online Free Edition



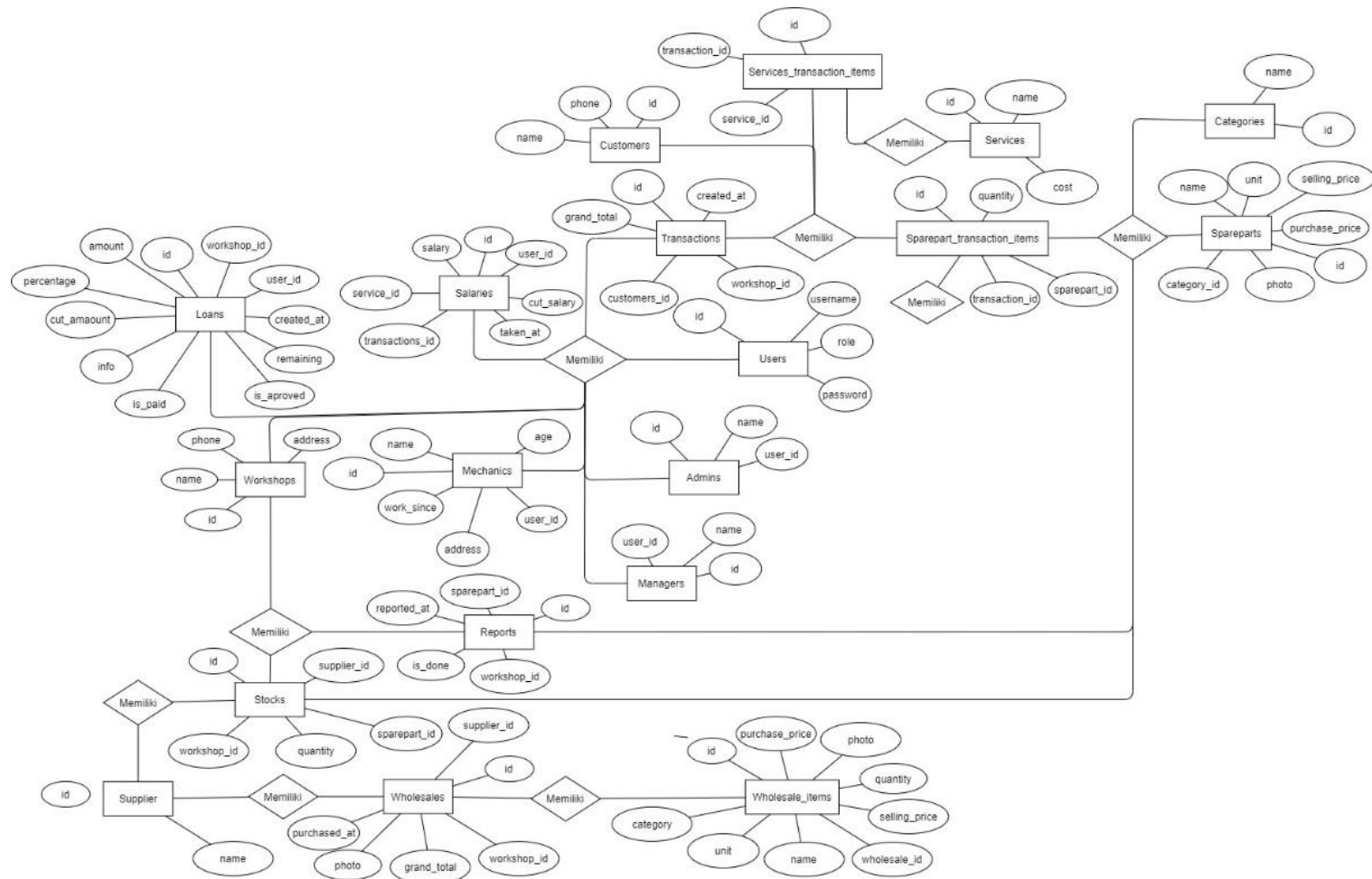
Visual Paradigm Online Free Edition

Gambar 4.4 Perancangan Class Diagram

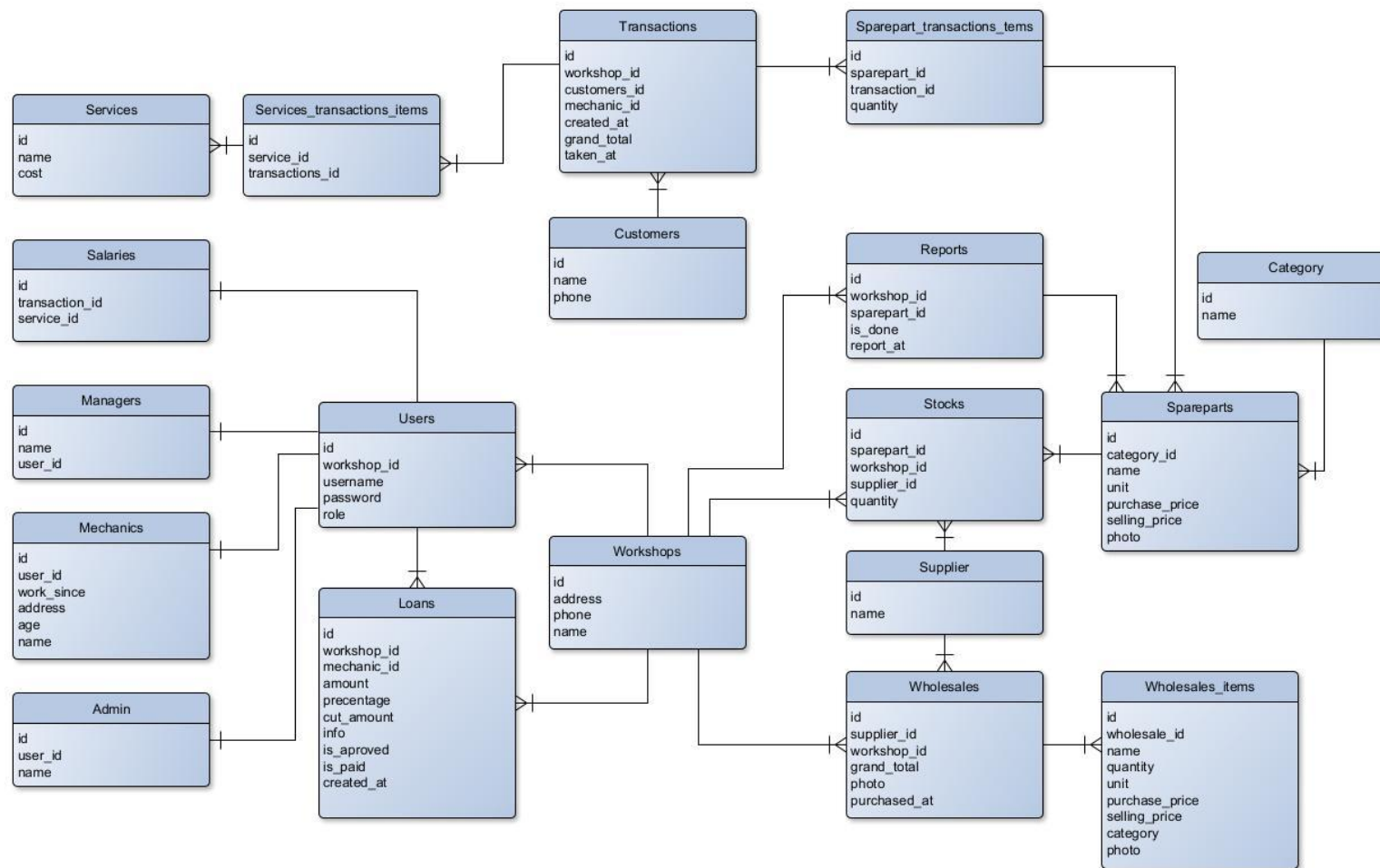
4.1.2 Perancangan Data

Basis data adalah salah satu hal yang penting dalam pengembangan sistem. Basis data akan menjadi tempat menyimpan data yang ada dalam sistem nantinya. Untuk itu perlu adanya penggambaran basis data yang jelas untuk nantinya diimplementasikan. Pada perancangan basis data akan digunakan Entity Relational Diagram (ERD) yang merupakan pemodelan dari *Conceptual Data Model* (CDM). CDM akan menampilkan entitas serta atribut dan deskriptifnya yang diilustrasikan pada Gambar 5.5. Setelah itu, CDM akan menjadi acuan untuk pemodelan ke dalam bentuk *Physical Data Model* (PDM) dengan nama entitas menjadi nama tabel, atribut dan deskriptif menjadi kolom yang akan diilustrasikan pada Gambar 4.6.

Pada gambar 4.5 terdapat 18 *entity*, setiap *entitas* memiliki relasi *one to one*. *Entity* tersebut diantaranya *Loans*, *Services*, *Salaries*, *Stocks*, *Supplier*, *Wholesales*, *Wholesales_items*, *Workshops*, *Reports*, *manager*, *Pengguna*, *cashier*, *Transactions*, *Spareparts*, *Transactions_items*.



Gambar 4.5 Conceptual Data Model (CDM)



Gambar 4.6 Physical Data Model (PDM)

4.1.3 Perancangan Komponen

Pada bagian perancangan komponen akan memaparkan rincian komponen-komponen algoritme yang akan digunakan dalam implementasi sistem. Terdapat 3 contoh perancangan algoritme yang akan dibahas pada bab ini yaitu algoritme menambah transaksi servis dan suku cadang, algoritme mengubah suku cadang, dan algoritme mengubah profil *mechanic*.

4.1.3.1 Algoritme Menambah Laporan Stok Rendah

Class : ReportController

Method : store()

Algoritme menambah Laporan Stok Rendah dapat dilihat pada tabel 4.1

Tabel 4.1 Algoritme Menambah Laporan Stok Rendah

Algoritme menambah laporan stok rendah	
1	Begin
2	User as Cashier
3	\$request->validate by <i>sparepart_id</i>
4	\$check = Report::firstWhere([
5	' <i>sparepart_id</i> ' => <i>sparepart_id</i> ,
6	' <i>workshop_id</i> ' => <i>workshop_id</i> ,
7	' <i>is_done</i> ' => '0']);
8	if(\$check)
9	show toast 'Suku cadang sudah pernah dilaporkan
10	namun belum diproses"
11	redirect('/ <i>sparepart</i>)
12	else
13	create object report
14	\$report->reported_at = format('Y-m-d')
15	\$report-> <i>sparepart_id</i> = <i>sparepart_id</i> ;
16	\$report-> <i>workshop_id</i> = <i>workshop_id</i> ;
17	
18	If save() == success
19	show toast 'Suku cadang berhasil dilaporkan'
20	else
21	show toast 'Suku cadang gagal dilaporkan"
22	redirect('/report');
23	End

4.1.3.2 Algoritme Mengubah Suku Cadang

Class : SparepartController

Method : update()

Algoritme Mengubah suku cadang dapat dilihat pada tabel 4.2

Tabel 4.2 Algoritme Mengubah Suku Cadang

Kode program mengubah suku cadang	
1	Begin
2	Get <i>sparepart</i> from database
3	Show edit form
4	Fill form input field with <i>sparepart</i> data
5	Get <i>name</i> , <i>quantity</i> , <i>unit</i> , <i>purchase_price</i> ,
6	<i>selling_price</i> , <i>category</i> from input
7	If validator catch error

8	Show error in form
9	EndIf
10	Save <i>sparepart</i> to database
11	If success
12	Show toast "Suku cadang berhasil diubah"
13	Else
14	Show toast "Suku cadang gagal diubah"
15	Show <i>sparepart</i> list
16	End

4.1.3.3 Algoritme Mengubah Profil

Class : MechanicController

Method : update()

Algoritme Mengubah Profil dapat dilihat pada tabel 4.3

Tabel 4.3 Algoritme Mengubah Profil

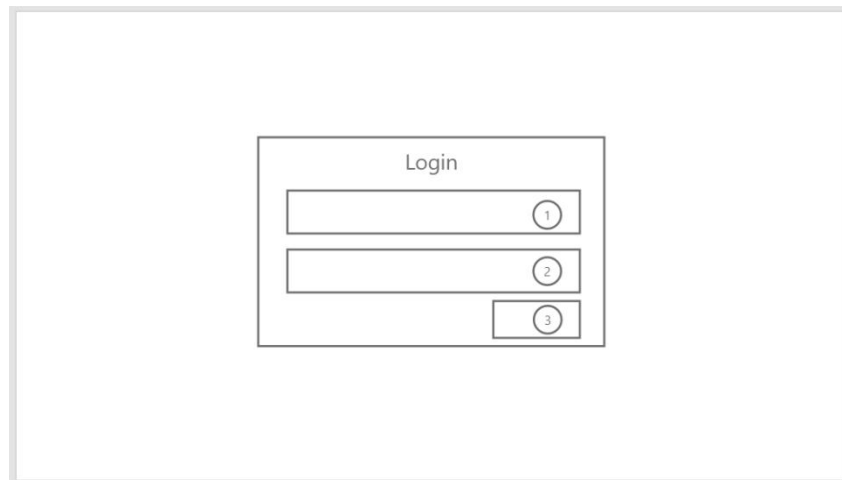
Kode program mengubah profil <i>Mechanic</i>	
1	Begin
2	Get <i>Mechanic</i> from database
3	Show edit form
4	Fill form input field with <i>Mechanic</i> data
5	Get <i>name, worked_since, age, address</i> from input
6	If validator catch error
7	Show error in form
8	Endif
9	Save <i>Mechanic</i> to database
10	If success
11	Show toast " <i>Mechanic</i> berhasil diubah"
12	Else
13	Show toast " <i>Mechanic</i> gagal diubah"
14	Show <i>Mechanic</i> list
15	End

4.1.4 Perancangan Antarmuka

Perancangan antarmuka ini ditujukan untuk menggambarkan antarmuka atau UI yang dipakai oleh sistem. Antarmuka ini nantinya adalah bagian yang bisa dilihat dan dapat diintegrasikan langsung oleh user. Perancangan antarmuka ini digambarkan dengan menggunakan adobe XD.

4.1.4.1 Perancangan Antarmuka *Login*

Perancangan antarmuka *login* dapat dilihat pada gambar 4.7.



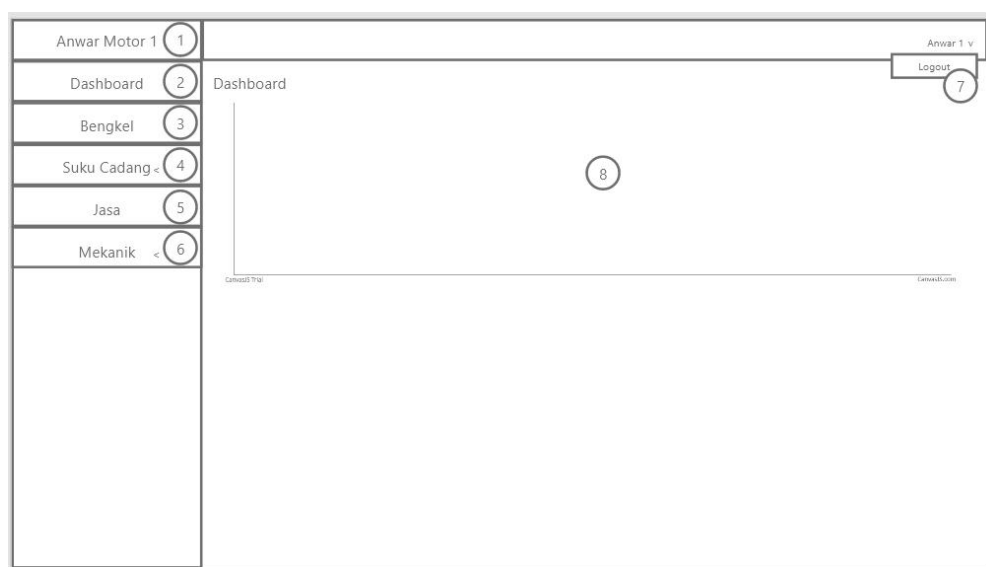
Gambar 4.7 Perancangan Antarmuka *Login*

Tabel 4.4 Uraian rancangan antarmuka halaman *Login*

No.	Nama Objek	Tipe	Keterangan
1.	<i>Input Username</i>	<i>Textfield</i>	untuk mengisi <i>username</i> pengguna
2.	<i>Input Password</i>	<i>Password</i>	untuk mengisi <i>password</i> pengguna
3.	<i>Login</i>	<i>Button</i>	Untuk mengarahkan user masuk ke dalam system

4.1.4.2 Perancangan Antarmuka *Dashboard Manager*

Perancangan antarmuka *dashboard* pemilik *manager* dapat dilihat pada gambar 4.8.



Gambar 4.8 Antarmuka *Dashboard Manager*

Tabel 4.5 Uraian rancangan antarmuka halaman *Dashboard Manager*

No.	Nama Objek	Tipe	Keterangan
1.	Nama Pengguna	<i>Text</i>	Nama dari pengguna
2.	<i>Dashboard</i>	<i>Navbar</i>	untuk mengarahkan ke halaman <i>dashboard</i>
3.	Bengkel	<i>Navbar</i>	untuk mengarahkan ke halaman bengkel
4.	Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman suku cadang
5.	Jasa	<i>Navbar</i>	untuk mengarahkan ke halaman servis
6.	Mekanik	<i>Navbar</i>	untuk mengarahkan ke halaman <i>mechanic</i>
7.	Logout	<i>Button</i>	untuk mengarahkan pengguna keluar dari sistem
8.	Grafik	<i>Graph</i>	untuk menampilkan grafik penjualan suku cadang

4.1.4.3 Perancangan Antarmuka Laporan Stok Rendah

Perancangan antarmuka Laporan Stok Rendah dapat dilihat pada gambar 4.9.



Gambar 4.9 Perancangan Antarmuka Laporan Stok Rendah

Tabel 4.6 Uraian rancangan antarmuka halaman Laporan Stok Rendah

No.	Nama Objek	Tipe	Keterangan
1.	Nama Pengguna	<i>Text</i>	Nama dari pengguna
2.	<i>Dashboard</i>	<i>Navbar</i>	untuk mengarahkan ke halaman <i>dashboard</i>
3.	Bengkel	<i>Navbar</i>	untuk mengarahkan ke halaman bengkel
4.	Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman suku cadang
5.	Daftar Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman daftar suku cadang
6.	Laporan Stok Rendah	<i>Navbar</i>	untuk mengarahkan ke halaman laporan stok rendah
7.	Pembelian Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman pembelian suku cadang

8.	Jasa	<i>Navbar</i>	untuk mengarahkan ke halaman servis
9.	Mekanik	<i>Navbar</i>	untuk mengarahkan ke halaman <i>mechanic</i>
10.	Logout	<i>Button</i>	untuk mengarahkan pengguna keluar dari sistem
11.	Tabel Suku Cadang Rendah	<i>Table</i>	untuk menampilkan tabel suku cadang rendah
12.	Proses	<i>Button</i>	Untuk melakukan mengubah status suku cadang rendah

4.1.4.4 Perancangan Antarmuka Pembelian Suku Cadang

Perancangan antarmuka Laporan Pembelian suku cadang dapat dilihat pada gambar 4.10.

Gambar 4.10 Perancangan Antarmuka Pembelian Suku Cadang

Gambar 4.11 Perancangan Antarmuka Pembelian Suku Cadang

Tabel 4.7 Uraian rancangan antarmuka halaman Pembelian Barang

No.	Nama Objek	Tipe	Keterangan
1.	Nama Pengguna	<i>Text</i>	Nama dari pengguna
2.	<i>Dashboard</i>	<i>Navbar</i>	untuk mengarahkan ke halaman <i>dashboard</i>
3.	Bengkel	<i>Navbar</i>	untuk mengarahkan ke halaman bengkel
4.	Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman suku cadang
5.	Daftar Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman daftar suku cadang
6.	Laporan Stok Rendah	<i>Navbar</i>	untuk mengarahkan ke halaman laporan stok rendah
7.	Pembelian Suku Cadang	<i>Navbar</i>	untuk mengarahkan ke halaman pembelian suku cadang
8.	Jasa	<i>Navbar</i>	untuk mengarahkan ke halaman servis
9.	Mekanik	<i>Navbar</i>	untuk mengarahkan ke halaman <i>mechanic</i>
10.	Logout	<i>Button</i>	untuk mengarahkan pengguna keluar dari sistem
11.	Tambah	<i>Button</i>	untuk mengarahkan ke halaman pembelian suku cadang
12.	Tabel Pembelian Suku Cadang	<i>Table</i>	untuk melakukan mengubah status suku cadang rendah
13.	Tanggal Pembelian	<i>Date</i>	untuk mengisi tanggal pembelian
14.	Nama Suku Cadang	<i>Textfield</i>	untuk mengisi nama suku cadang
15.	Kategori	<i>Textfield</i>	untuk mengisi kategori
16.	Supplier	<i>Textfield</i>	untuk mengisi nama supplier
17.	Jumlah	<i>Texfield</i>	untuk mengisi jumlah
18.	Satuan	<i>Dropdown</i>	untuk mengisi satuan
19.	Harga Beli	<i>Texfield</i>	untuk mengisi harga beli
20.	Harga Jual	<i>Texfield</i>	untuk mengisi harga jual
21.	Foto Suku Cadang	<i>File</i>	untuk mengisi foto suku cadang
22.	Tombol +	<i>Button</i>	untuk menambah pembelian suku cadang
23.	Tabel Suku Cadang Rendah	<i>Table</i>	untuk menampilkan tabel suku cadang rendah
24.	Tabel Pembelian Suku Cadang Rendah	<i>Table</i>	untuk menampilkan daftar pembelian suku cadang
25.	Total Pembelian	<i>Integer</i>	untuk menampilkan harga total pembelian suku cadang
26.	Foto Nota Pembelian Suku Cadang	<i>File</i>	untuk mengisi foto nota pembelian suku cadang
27.	Submit	<i>Button</i>	untuk menyimpan data pembelian suku cadang
28.	Delete	<i>Button</i>	Untuk menghapus data pembelian suku cadang per <i>item</i>
29.	Delete	<i>Button</i>	Untuk menghapus seluruh data pembelian suku cadang

4.2 Implementasi Sistem

Implementasi sistem ini dilakukan setelah perancangan selesai dilakukan. Semua hal yang sudah dirancang akan diimplementasikan untuk membuat sebuah sistem yang utuh. Implementasi tersebut diantaranya implementasi kode program, dengan menggunakan *framework laravel* yang mana akan menulis kode program tersebut dan dijalankan di dalam sistem. Kemudian implementasi basis data menggunakan MySQL, membuat entity-entity yang sudah dirancang lengkap dengan atributnya.

4.2.1 Spesifikasi Sistem

Pada bagian ini akan dijelaskan spesifikasi perangkat yang digunakan dalam merancang dan mengimplementasikan sistem. Dalam pembuatan sistem, perangkat yang digunakan merupakan komputer pribadi. Spesifikasi yang dijelaskan di sini akan meliputi spesifikasi dari komputer pribadi tersebut. tiap-tiap spesifikasi tersebut terdiri dari CPU atau Processor, Memory atau RAM dan yang terakhir VGA atau Graphics Card. selain perangkat keras juga dijelaskan perangkat lunak. Perangkat lunak yang digunakan adalah untuk dokumentasi, desain, menulis kode, bahasa pemrograman, basis data dan perangkat untuk melihat sistem yang dalam hal ini adalah peramban.

Spesifikasi perangkat keras sistem dapat dilihat di tabel 4.8.

Tabel 4.8 Spesifikasi Perangkat Keras Sistem

Perangkat	Deskripsi
<i>System Model</i>	Asus GL503GE
<i>Processor</i>	Intel Core i7-8750H
<i>Memory</i>	8192 MB
<i>Graphics Card</i>	NVIDIA GeForce GTX 1050Ti

Spesifikasi perangkat lunak sistem yang dilihat pada tabel 4.9.

Tabel 4.9 Spesifikasi Perangkat Lunak Sistem

Perangkat	Deskripsi
Dokumentasi	Microsoft Office Word
Desain	Visual Paradigm, yEd Graph Editor, AdobeXD
<i>Editor</i>	Visual Studio Code
Bahasa Pemrograman	PHP
Basis Data	MySQL
Peramban	Microsoft Edge

4.2.2 Implementasi Kode Program

Implementasi komponen dilakukan setelah algoritme dalam bentuk pseudocode selesai dirancang. Implementasi ini dilakukan dengan menuliskan kode program ke dalam bahasa pemrograman yang dipilih. Kode program tersebut ditulis menggunakan teks editor yang nantinya disimpan menjadi file dengan ekstensi yang sesuai dengan bahasa pemrograman. Penjelasan implementasi kode program ini terbagi menjadi tiga yakni menambah transaksi servis dan suku cadang, mengubah suku cadang dan mengubah *mechanic*.

Implementasi komponen merupakan proses mengubah hasil dari perancangan komponen menjadi sebuah kode program yang dapat dijalankan atau dieksekusi. Bahasa pemrograman yang digunakan dalam implementasi ini adalah PHP dengan menggunakan *framework Laravel*. Terdapat tiga implementasi komponen yaitu implementasi kode menyimpan data *pinjaman*.

4.2.2.1 Implementasi Kode Menambah Laporan Stok Rendah

Class : ReportController

Method : store(Request, request)

Kode program accept spesifikasi kebutuhan dapat dilihat pada tabel 4.10

Tabel 4.10 Implementasi Kode Menambah Laporan Stok Rendah

Kode program menambah laporan stok rendah	
1	public function store(Request \$request)
2	{
3	Auth::user()->hasRole(['cashier']);
4	\$request->validate([
5	'sparepart_id' =>
6	'required exists:spareparts,id',
7]);
8	\$check = Report::firstWhere([
9	'sparepart_id' => \$request->sparepart_id,
10	'workshop_id' => Auth::user()->workshop_id,
11	'is_done' => '0'
12]);
13	if(\$check) {
14	session()->flash('toast', ['info', 'Suku cadang
15	sudah pernah dilaporkan namun belum diproses']);
16	return redirect('/sparepart');
17	} else {
18	\$report = new Report();
19	\$report->reported_at = \Carbon\Carbon::now()-
20	>format('Y-m-d');
21	\$report->sparepart_id = \$request->sparepart_id;
22	\$report->workshop_id = Auth::user()-
23	>workshop_id;
24	if(\$report->save()) session()->flash('toast',
25	['success', 'Suku cadang berhasil dilaporkan']);
26	else session()->flash('toast', ['error', 'Suku
27	cadang gagal dilaporkan']);
28	return redirect('/report');
29	}
30	}

4.2.2.2 Implementasi Kode Mengubah Suku Cadang

Class : SparepartController

Method : update(Request \$request, Sparepart \$sparepart)

Kode program accept spesifikasi kebutuhan dapat dilihat pada tabel 4.11

Tabel 4.11 Implementasi Kode Mengubah Suku Cadang

Kode program mengubah suku cadang	
1	public function update(Request \$request, Sparepart \$sparepart)
2	{
3	Auth::user()->hasRole(['manager']);
4	\$request->validate([
5	'name' => 'required',
6	'quantity' => 'required numeric',
7	'unit' => 'required in:Unit,Set',
8	'purchase_price' => 'required',
9	'selling_price' => 'required',
10	'category' => 'required',
11]);
12	\$sparepart->name = \$request->name;
13	\$sparepart->unit = \$request->unit;
14	\$sparepart->category_id =
15	Category::firstWhere('name', \$request->category) ?
16	Category::firstWhere('name', \$request->category)->id :
17	Category::create(['name' => \$request->category]);
18	\$sparepart->purchase_price = str_replace('.', '',
19	substr(\$request->purchase_price, 3));
20	\$sparepart->selling_price = str_replace('.', '',
21	substr(\$request->selling_price, 3));
22	\$sparepart->stock->quantity = \$request->quantity;
23	\$sparepart->stock->save();
24	if(\$request->has('photo')) {
25	\$file = \$request->file('photo');
26	\$filename = \$sparepart->id . '_' .
27	str_replace([' ', '/'], '-', strtolower(\$sparepart->name)) .
28	'.'. \$file->getClientOriginalExtension();
29	\$file->move(public_path("uploads/product\\"),
30	\$filename);
31	\$sparepart->photo = \$filename;
32	}
33	if(\$sparepart->save()) session()->flash('toast',
34	['success', 'Suku cadang berhasil diubah']);
35	else session()->flash('toast', ['error', 'Suku
36	cadang gagal diubah']);
37	return redirect('/sparepart');
38	}

4.2.2.3 Implementasi Kode Mengubah Profil

Class : MechanicController

Method : update(Request \$request, User \$Mechanic)

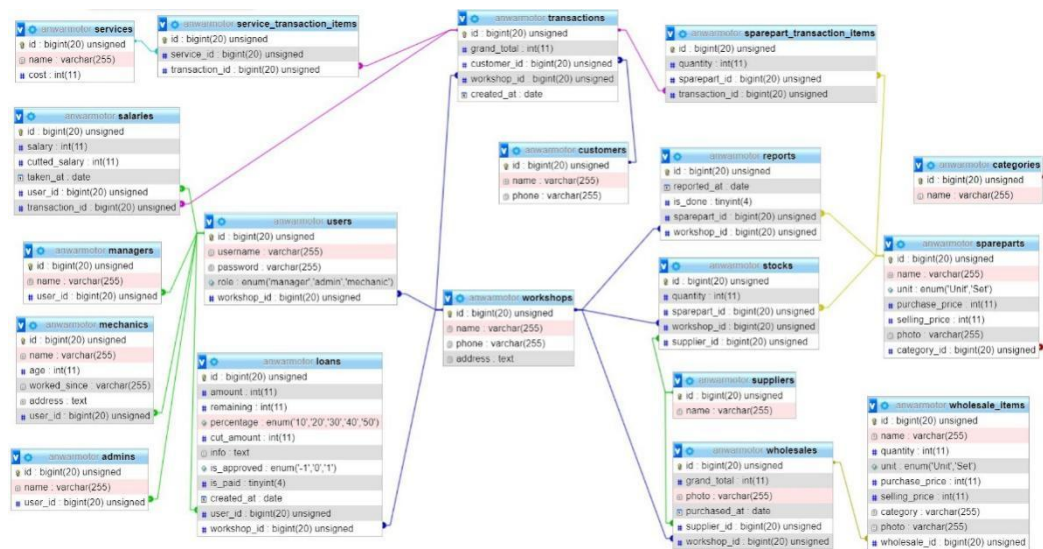
Kode program accept spesifikasi kebutuhan dapat dilihat pada tabel 4.12

Tabel 4.12 Implementasi Kode Mengubah Profil

Kode program mengubah profil <i>Mechanic</i>	
1	public function update(Request \$request, User \$user)
2	{
3	Auth::user()->hasRole(['manager']);
4	User::firstWhere('id', \$user->id)->role == 'mechanic' && \$user->workshop_id == Auth::user()- >workshop_id ? : abort('403');
5	\$request->validate([
6	'name' =>
7	['required', Rule::unique('mechanics', 'name')->ignore(\$user->
8	data, 'name')],
9	'worked_since' => 'required numeric gt:0',
10	'age' => 'required numeric gt:0',
11	'address' => 'required'
11]);
12	\$user->username = str_replace(' ', '', strtolower(\$request->name));
13	\$user->save();
14	\$data = Mechanic::firstWhere('user_id', \$user->id);
	\$data->name = ucwords(\$request->name);
	\$data->worked_since = \$request->worked_since;
	\$data->age = \$request->age;
	\$data->address = ucwords(\$request->address);
	if(\$data->save()) session()->flash('toast',
	['success', 'Mekanik berhasil diubah']);
	else session()->flash('toast', ['error', 'Mekanik
	gagal diubah']);
	return redirect('/mechanic');
	}

4.2.3 Implementasi Basis Data

Implementasi data merupakan proses dimana data-data yang telah didapatkan pada bagian perancangan sebelumnya dibuat menjadi sebuah sistem basis data atau *database*. Gambar 4.12 akan memaparkan sebuah skema *database* yang telah dibuat berdasarkan ERD dengan menggunakan *database MySQL*.



Gambar 4.12 Implementasi Basis Data

4.2.4 Implementasi Antarmuka

4.2.4.1 Implementasi Antarmuka Login

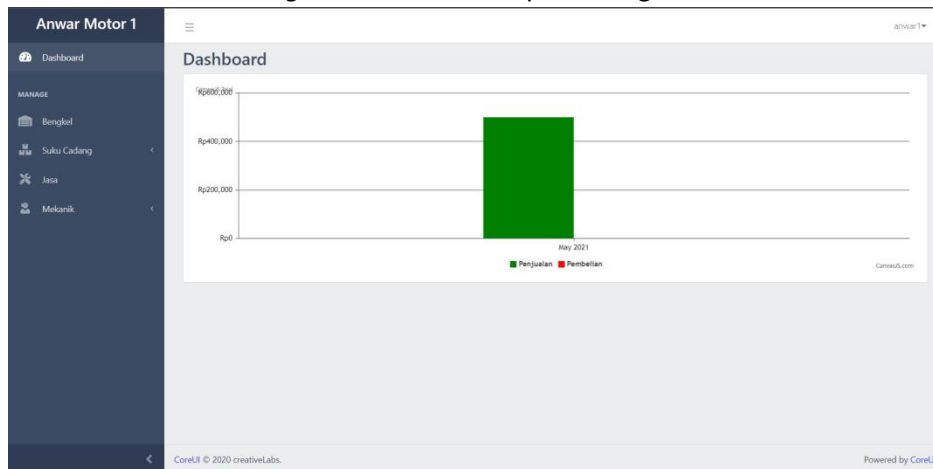
Pada implementasi antarmuka *login* terdapat halaman yang digunakan pengguna untuk masuk kedalam sistem. Untuk dapat mengakses halaman ini, pengguna mengisi *form login* berupa *username* dan *password*, kemudian memilih tombol "*login*". Setelah menekan tombol "*login*", maka *pengguna* akan dialihkan ke halaman *dashboard* atau halaman utama masing masing pengguna. Berikut pada gambar 4.13 merupakan implementasi dari antarmuka *login* sesuai dari hasil perancangan antarmuka.

Gambar 4.13 Implementasi Antarmuka Login

4.2.4.2 Implementasi Antarmuka Dashboard Manager

Pada implementasi antarmuka *dashboard manager* terdapat halaman yang digunakan *manager* untuk melihat data grafik penjualan dan pembelian suku cadang. Untuk dapat mengakses halaman ini, *manager* mengisi *form login* berupa *username* adalah "*manager1*", "*manager2*" atau "*manager3*" dan *password* adalah "*a*", dikarenakan bengkel Anwar Motor memiliki 3 cabang, kemudian memilih tombol "*login*". Setelah menekan tombol "*login*", maka *manger* akan dialihkan ke halaman *dashboard*

atau halaman utama *manager*. Berikut pada gambar 4.14 merupakan implementasi dari antarmuka *dashboard manager* sesuai dari hasil perancangan antarmuka.



Gambar 4.14 Implementasi Antarmuka *Dashboard Manager*

4.2.4.3 Implementasi Antarmuka Laporan Stok Rendah

Pada implementasi antarmuka laporan stok rendah terdapat halaman yang digunakan *manager* untuk melihat data suku cadang rendah. Untuk dapat mengakses halaman ini, *manager* harus masuk ke dalam sistem. Setelah *login*, *manager* menekan menu *navigation bar* “suku cadang” dan memilih menu “laporan stok rendah”, maka *manger* akan dialihkan ke halaman laporan stok rendah. Berikut pada gambar 4.15 merupakan implementasi dari antarmuka laporan stok rendah sesuai dari hasil perancangan antarmuka.

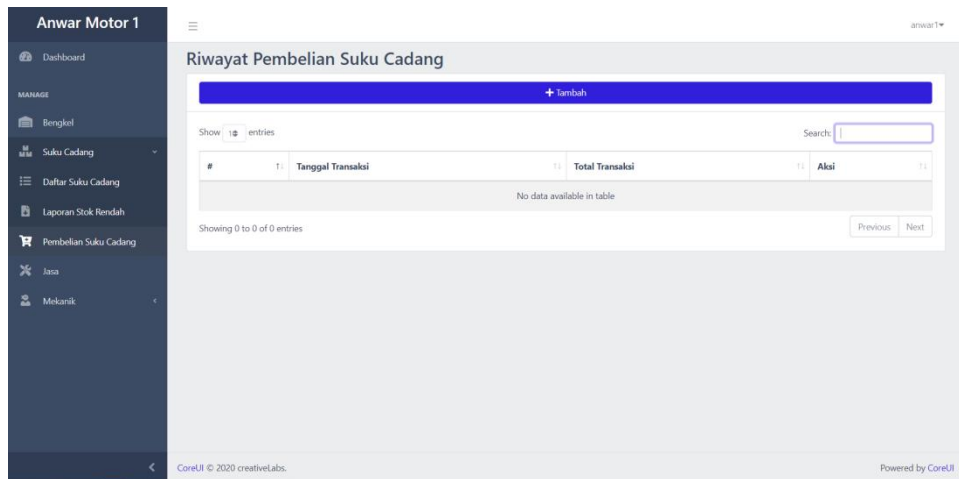
Gambar 4.15 Implementasi Antarmuka Laporan Stok Rendah

4.2.4.4 Implementasi Antarmuka Pembelian Suku Cadang

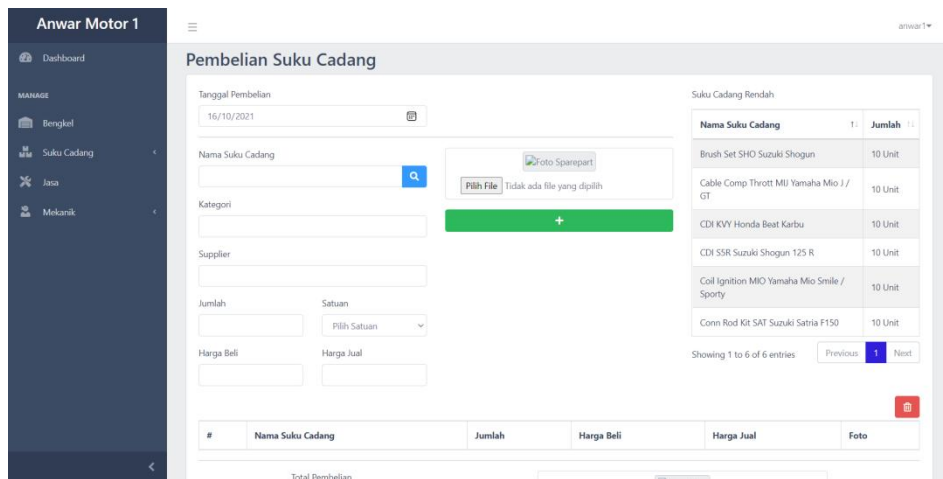
Pada implementasi antarmuka pembelian suku cadang terdapat halaman yang digunakan *manager* untuk melihat riwayat data pembelian suku cadang. Untuk dapat mengakses halaman ini, *manager* harus masuk ke dalam sistem. Setelah *login*, *manager* menekan menu *navigation bar* “suku cadang” dan memilih menu “pembelian suku cadang”, maka *manger* akan dialihkan ke halaman riwayat pembelian suku cadang. Untuk melakukan pembelian barang *manager* menekan tombol tambah, maka *manger*

akan dialihkan ke halaman pembelian suku cadang yang dimana terdapat form untuk diisi sesuai dengan pembelian suku cadang.

Kemudian setelah mengisi keterangan dan *upload* gambar, maka selanjutnya menekan tombol “+” yang berwarna hijau untuk menyimpan data yang sudah diisi sebelumnya. Jika membeli barang lebih dari 1 maka melakukan pengisian ulang form pembelian barang. Setelah itu *manager* menekan tombol “*submit*” untuk menyimpan data pembelian barang. Berikut pada gambar 4.16 dan 4.17 merupakan implementasi dari antarmuka pembelian suku cadang sesuai dari hasil perancangan antarmuka.



Gambar 4.16 Implementasi Antarmuka Laporan Stok Rendah



Gambar 4.17 Implementasi Antarmuka Laporan Stok Rendah

BAB 5 PENGUJIAN

5.1 Pengujian Unit

Pengujian unit dilakukan terhadap 3 sampel yaitu algoritme dalam *method add_report()* dari klas *ReportController*, algoritme dalam *method update()* dari klas *SparepartController* dan algoritme dalam *method update()* dari klas *MechanicController*. Klas-klas berikut tersebut bukan merupakan klas main sehingga dibutuhkan klas *PengujianController*. Klas *PengujianController* adalah klas yang dibuat sebagai pengganti dalam mengeksekusi pengujian.

5.1.1 Pengujian Unit Menambah Laporan Stok Rendah

Fungsi *add_report()* digunakan sebagai menyimpan data laporan stok rendah yang berada pada *ReportController*. Algoritme fungsi *add_report()* dapat dilihat pada Tabel 5.1.

1. Algoritme Fungsi *add_report()*

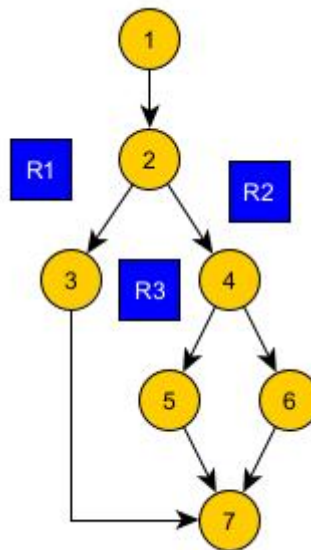
Tabel 5.1 Algoritme Fungsi *add_report()*

Node	Algoritme fungsi menambah laporan stok rendah
1	Begin Get session, grand_total, customer_id from input
2	User as cashier Inisialisasi check = Report sparepart_id = request->sparepart_id, workshop_id = workshop_id, is_done = 0
3	if (\$check) toast "Suku cadang sudah pernah dilaporkan namun belum diproses redirect /sparepart
4	else instansiasi objek report objek report->reported_at = format('Y-m-d'); objek report->sparepart_id = sparepart_id; objek report->workshop_id = workshop_id;
5	if objek report->save toas Suku cadang berhasil dilaporkan
6	else Suku cadang gagal dilaporkan
7	End

2. Basis Path Testing

a. Flow Graph

Flow graph store(Request \$request) dapat dilihat pada Gambar 6.1. Flow Graph tersebut tersusun dari 7 node, 8 edge, 3 region dan 2 buah node yang bercabang.



Gambar 5.1.1 Flow Graph fungsi *store(Request \$request)*

b. *Cyclomatic Complexity*

$V(G) = \text{Jumlah Region} = 3$

$V(G) = (\text{jumlah edge} - \text{jumlah node}) + 2 = (8 - 7) + 2 = 1 + 2 = 3$

$V(G) = P + 1 = 2 + 1 = 3$

c. *Independent Path*

Jalur 1 = 1-2-3-7

Jalur 2 = 1-2-4-5-7

Jalur 3 = 1-2-4-6-7

Mengikuti poin c terdapat 3 jalur untuk independent path terdapat terdapat 3 kasus uji untuk fungsi *add_report()*. Hasil pengujian unit fungsi *add_report()* dapat dilihat pada Tabel 5.2.

Tabel 5.2 Hasil Pengujian Unit Fungsi *store(Request \$request)*

Jalur	Prosedur Uji	<i>Expected result</i>	<i>Result</i>	Status
1	Klas PengujianController menjalankan <i>method</i> <i>add_report()</i> dari klas <i>ReportController</i> dengan <i>input</i> berupa 'sparepart_id' : '159', 'workshop_id' : '1',	Data suku cadang rendah berhasil ditambahkan	Data suku cadang rendah berhasil ditambahkan	<i>Passed</i>

	'is_done' : '0'			
2	Klas PengujianController menjalankan <i>method add_report()</i> dari klas <i>ReportController</i> dengan <i>input</i> berupa 'sparepart_id' : '25', 'workshop_id' : '1', 'is_done' : '1'	Data suku cadang sudah pernah dilaporkan namun belum diproses	Data suku cadang sudah pernah dilaporkan namun belum diproses	Passed
3	Klas PengujianController menjalankan <i>method add_report()</i> dari klas <i>ReportController</i> dengan <i>input</i> berupa 'sparepart_id' : '1', 'workshop_id' : '1', 'is_done' : '0'	suku cadang gagal dilaporkan	suku cadang gagal dilaporkan	Passed

5.1.2 Pengujian Unit Mengubah Suku Cadang

Fungsi *update()* digunakan sebagai mengubah data suku cadang yang berada pada *SparepartController*. Algoritme fungsi *update()* dapat dilihat pada Tabel 5.3.

1. Algoritme Fungsi *update()*

Tabel 5.3 Algoritme Fungsi *update()*

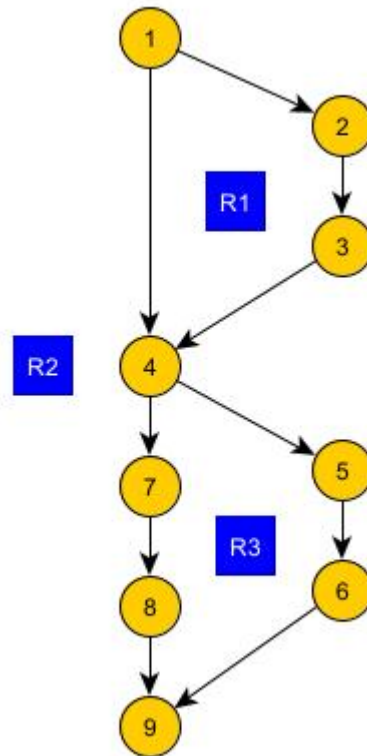
Node	Algoritme fungsi mengubah suku cadang
1	Begin
2	Get <i>sparepart</i> from <i>database</i> Show edit form Fill form input field with <i>sparepart</i> data Get <i>name</i> , <i>quantity</i> , <i>unit</i> , <i>purchase_price</i> , <i>selling_price</i> , <i>category</i> from input If validator catch error
3	Show error in form
4	EndIf Save <i>sparepart</i> to <i>database</i>
5	If success
6	Show toast "Suku cadang berhasil diubah"
7	Else

	Show toast "Suku cadang gagal diubah"
8	Show <i>sparepart</i> list
9	End

2. Basis Path Testing

a. Flow Graph

Flow graph update() dapat dilihat pada Gambar 6.1. *Flow Graph* tersebut tersusun dari 9 *node*, 10 *edge*, 3 *region* dan 2 buah *node* yang bercabang.



Gambar 5.2.2 Flow Graph fungsi *update()*

b. Cyclomatic Complexity

$V(G) = \text{Jumlah Region} = 3$

$V(G) = (\text{jumlah edge} - \text{jumlah node}) + 2 = (10 - 9) + 2 = 1 + 2 = 3$

$V(G) = P + 1 = 2 + 1 = 3$

c. Independent Path

Jalur 1 = 1-4-7-8-9

Jalur 2 = 1-4-5-6-9

Jalur 3 = 1-2-3-9

Mengikuti poin c terdapat 3 jalur untuk *independent path* terdapat 3 kasus uji untuk fungsi *update()*. Hasil pengujian unit fungsi *update()* dapat dilihat pada Tabel 5.4.

Tabel 5.4 Hasil Pengujian Unit Fungsi *update()*

Jalur	Prosedur Uji	<i>Expected result</i>	<i>Result</i>	Status
1	Klas PengujianController menjalankan method <i>update()</i> dari klas <i>SparepartController</i> dengan <i>input</i> berupa 'name' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma', 'quantity' : 'null', 'unit' : 'Set', 'purchase_price' : 'Rp 405.000', 'selling_price' : 'Rp 500.000', 'category' : 'Honda'	Mengembalikan user ke halaman form dengan <i>error</i>	Mengembalikan user ke halaman form dengan <i>error</i>	<i>Passed</i>
2	Klas PengujianController menjalankan method <i>update()</i> dari klas <i>SparepartController</i> dengan <i>input</i> berupa 'name' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma', 'quantity' : 'null', 'unit' : 'Set', 'purchase_price' : 'null', 'selling_price' : 'null', 'category' : 'Honda'	Mengembalikan user ke halaman form dengan <i>error</i>	Mengembalikan user ke halaman form dengan <i>error</i>	<i>Passed</i>
3	Klas PengujianController menjalankan method <i>update()</i> dari klas	Mengembalikan user ke halaman list suku cadang dan data	Mengembalikan user ke halaman list suku cadang dan data	<i>Passed</i>

	<i>SparepartController</i> dengan <i>input</i> berupa 'name': 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma', 'quantity': '57', 'unit': 'Set', 'purchase_price': 'Rp 405.000', 'selling_price': 'Rp 500.000', 'category': 'Honda'	berhasil disimpan ke dalam <i>database</i>	berhasil disimpan ke dalam <i>database</i>	
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------	--------------------------------------------	--

5.1.3 Pengujian Unit Mengubah Profil

Fungsi *update()* digunakan sebagai mengubah profil *mechanic* yang berada pada *MechanicController*. Algoritme fungsi *update()* dapat dilihat pada Tabel 5.5.

1. Algoritme Fungsi *update()*

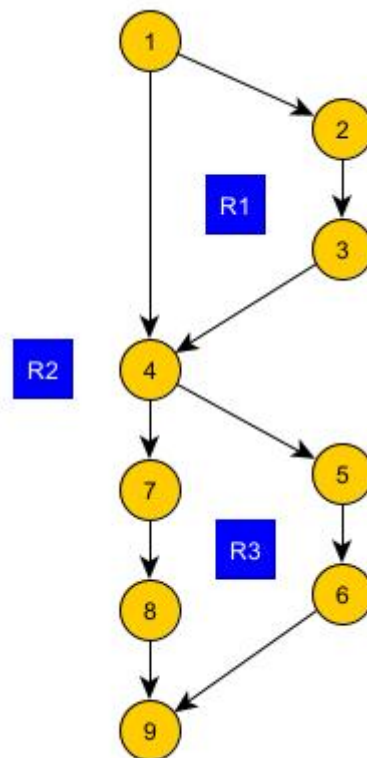
Tabel 5.5 Algoritme Fungsi *update()*

Node	Algoritme fungsi mengubah profil <i>Mechanic</i>
1	Begin
2	Get <i>Mechanic</i> from <i>database</i> Show edit form Fill form input field with <i>Mechanic</i> data Get <i>name</i> , <i>work_since</i> , <i>age</i> , <i>address</i> from input If validator catch error
3	Show error in form
4	EndIf Save <i>Mechanic</i> to <i>database</i>
5	If success
6	Show toast " <i>Mechanic</i> berhasil diubah"
7	Else Show toast " <i>Mechanic</i> gagal diubah"
8	Show <i>Mechanic</i> list
9	End

2. Basis Path Testing

a. Flow Graph

Flow graph update() dapat dilihat pada Gambar 6.1. *Flow Graph* tersebut tersusun dari 9 *node*, 10 *edge*, 3 *region* dan 2 buah *node* yang bercabang.



Gambar 5.3.3 Flow Graph fungsi *update()*

b. *Cyclomatic Complexity*

$V(G) = \text{Jumlah Region} = 3$

$V(G) = (\text{jumlah edge} - \text{jumlah node}) + 2 = (10 - 9) + 2 = 1 + 2 = 3$

$V(G) = P + 1 = 2 + 1 = 3$

c. *Independent Path*

Jalur 1 = 1-4-7-8-9

Jalur 2 = 1-4-5-6-9

Jalur 3 = 1-2-3-9

Mengikuti poin c terdapat 3 jalur untuk *independent path* terdapat 3 kasus uji untuk fungsi *update()*. Hasil pengujian unit fungsi *update()* dapat dilihat pada Tabel 5.6.

Tabel 5.6 Hasil Pengujian Unit Fungsi *update(Request \$request)*

Jalur	Prosedur Uji	<i>Expected result</i>	<i>Result</i>	Status
1	Klas PengujianController menjalankan method <i>update()</i> dari klas <i>MechanicController</i> dengan input berupa	Mengembalikan user ke halaman form dengan <i>error</i>	Mengembalikan user ke halaman form dengan <i>error</i>	<i>Passed</i>

	'name' : 'Mubin', 'worked_since' : '1997', 'age' : 'a', 'address' : 'Malang'			
2	Klas PengujianController menjalankan method <i>update()</i> dari klas <i>MechanicController</i> 'name' : 'Mubin', 'worked_since' : 'a', 'age' : '33', 'address' : 'Malang'	Mengembalikan user ke halaman form dengan <i>error</i>	Mengembalikan user ke halaman form dengan <i>error</i>	<i>Passed</i>
3	Klas PengujianController menjalankan method <i>update()</i> dari klas <i>MechanicController</i> 'name' : 'Mubin', 'worked_since' : '1997', 'age' : '33', 'address' : 'Malang'	Mengembalikan user ke halaman list suku cadang dan data berhasil disimpan ke dalam <i>database</i>	Mengembalikan user ke halaman list suku cadang dan data berhasil disimpan ke dalam <i>database</i>	<i>passed</i>

5.2 Pengujian Integrasi

Pengujian integrasi dilakukan untuk menguji komponen yang saling berinteraksi di dalam sistem untuk melakukan sebuah fungsionalitas. Pengujian integrasi dilakukan pada klas yang memiliki hubungan dengan klas lainnya. Dalam proyek ini, pengujian integrasi dilakukan dengan teknik pengujian *white box* dengan metode *basis path testing*. Hanya akan terdapat 1 operasi yang menjadi sampel dalam pengujian ini yaitu fungsi *store(Request \$request)*.

5.2.1 Pengujian Integrasi Menambah Transaksi Servis dan Suku Cadang

Pada pengujian ini terdapat fungsi Fungsi *add_transaction()* dari klas *TransactionController()* yang didalam fungsi tersebut akan memanggil fungsi *new Customer()* dari klas *Customer*, fungsi *new Transaction()* dari klas *Transaction*, fungsi *new Salary()* dari klas *Salary*, fungsi *new ServiceTransactionItem()* dari klas *ServiceTransactionItem*, fungsi *new Nitification()* dari klas *Notification* dan fungsi *new Report()* dari klas *Report*. Algoritme fungsi *store(Request \$request)* dapat dilihat pada Tabel 5.7.

1. Pseudocode Algoritme Fungsi *store add_transaction()*

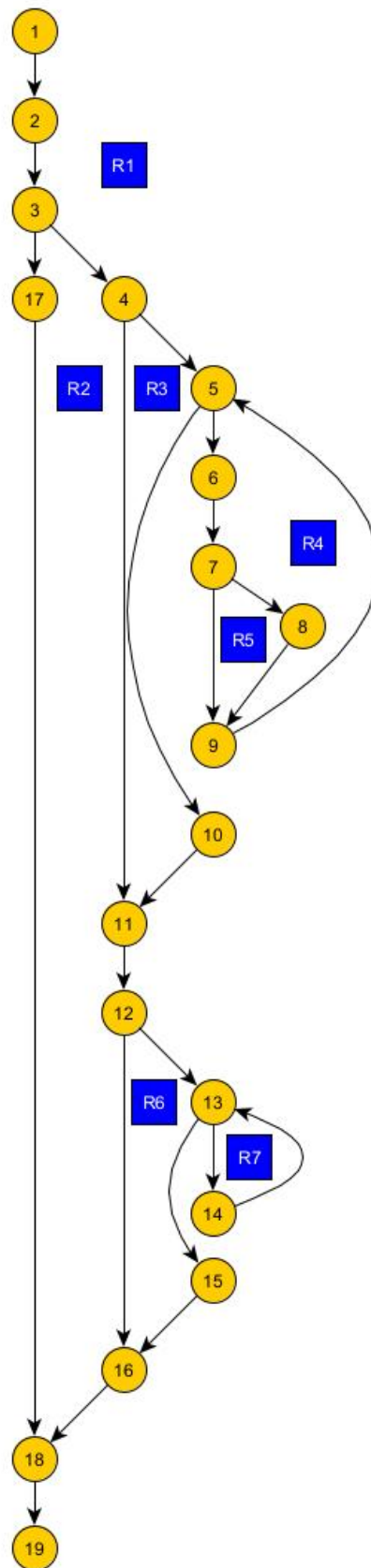
Tabel 5.7 Pseudocode Algoritme Fungsi *store add_transaction()*

Node	Algoritme fungsi <i>store(Request \$request)</i>
1	Begin
2	Validasi input Inisialisasi transaction: Transaction
3	If save transaction = true Toast sukses
4	If session has <i>spareparts</i>
5	Loop <i>spareparts</i>
6	Inisialisasi item: <i>SparepartTransactionItem</i> Save item Stock decrement
7	If stock <= 10
8	Inisialisasi report: Report Save report
9	End if
10	End loop Clear <i>spareparts</i>
11	End if
12	If session has <i>services</i>
13	Loop <i>services</i>
14	Inisialisasi item: <i>ServiceTransactionItem</i> Save item Stock decrement
15	End loop Inisialisasi salary: Salary Save salary Clear <i>services</i>
16	End if
17	Else Toast gagal
18	End if
19	End

2. Basis Path Testing

a. Flow Graph

Flow graph add_transaction() dapat dilihat pada Gambar 6.5. *Flow Graph* tersebut tersusun dari 19 *node*, 24 *edge*, 7 *region* dan 6 buah *node* yang bercabang.



Gambar 5.4 *Flow Graph* fungsi `add_transaction()`

b. *Cyclomatic Complexity*

$V(G) = \text{Jumlah Region} = 7$

$V(G) = (E - N) + 2 = (24 - 19) + 2 = 5 + 2 = 7$

$V(G) = P + 1 = 6 + 1 = 7$

c. *Independent Path*

Jalur 1 = 1-2-3-4-5-6-7-8-9-5-10-11-12-13-14-13-15-16-18-19

Jalur 2 = 1-2-3-4-5-6-7-9-5-10-11-12-13-14-13-15-16-18-19

Jalur 3 = 1-2-3-4-5-6-7-8-9-5-10-11-12-16-18-19

Jalur 4 = 1-2-3-4-5-6-7-9-5-10-11-12-16-18-19

Jalur 5 = 1-2-3-4-11-12-13-14-13-15-16-18-19

Jalur 6 = 1-2-3-4-11-12-16-18-19

Jalur 7 = 1-2-3-17-18-19

Mengikuti poin c dapat terdapat 7 jalur untuk *independent path* terdapat 7 kasus uji untuk fungsi *add_transaction()*. Hasil pengujian unit fungsi *add_transaction()* dapat dilihat pada Tabel 5.8.

Tabel 5.8 Hasil Pengujian Integrasi Fungsi *add_transaction()*

Jalur	Prosedur Uji	<i>Expected result</i>	Result	Status
1	Klas PengujianController menjalankan <i>method store()</i> dari klas <i>TransactionController</i> dengan 'sparepart' : 'Velg Zigen Falcon P5M 17160/185 SupraGrandKharisma', 'service' : 'GantiVelg Balok' 'mechanic' : 'Mubin' 'customer' : 'Iqbaal'	Mengembalikan user ke halaman transaksi penjualan dan data berhasil disimpan ke dalam <i>database</i>	Mengembalikan user ke halaman transaksi penjualan dan data berhasil disimpan ke dalam <i>database</i>	<i>passed</i>
2	Klas PengujianController menjalankan <i>method store()</i> dari klas <i>TransactionController</i> dengan 'sparepart' : 'Velg Zigen Falcon P5M	Mengembalikan user ke halaman <i>form</i> dengan <i>error</i>	Mengembalikan user ke halaman <i>form</i> dengan <i>error</i>	<i>passed</i>

	17160/185 SupraGrandKharisma', 'service' : 'GantiVelg Balok' ' mechanic ' : 'Mubin' 'customer' : 'null'			
3	Klas PengujianController menjalankan <i>method store()</i> dari klas <i>TransactionController</i> dengan 'sparepart' : 'Velg Zigen Falcon P5M 17160/185 SupraGrandKharisma', 'service' : 'GantiVelg Balok' ' mechanic ' : 'null' 'customer' : 'lqbaal'	Mengembal ikan user ke halaman <i>form</i> dengan <i>error</i>	Mengembalik an user ke halaman <i>form</i> dengan <i>error</i>	<i>passed</i>
4	Klas PengujianController menjalankan <i>method store()</i> dari klas <i>TransactionController</i> dengan 'sparepart' : 'null', 'service' : 'GantiVelg Balok' ' mechanic ' : 'null' 'customer' : 'lqbaal'	Mengembal ikan user ke halaman <i>form</i> dengan <i>error</i>	Mengembalik an user ke halaman <i>form</i> dengan <i>error</i>	<i>passed</i>
5	Klas PengujianController menjalankan <i>method store()</i> dari klas <i>TransactionController</i> dengan 'sparepart' : 'null', 'service' : 'GantiVelg Balok' ' mechanic ' : 'Mubin' 'customer' : 'null'	Mengembal ikan user ke halaman <i>form</i> dengan <i>error</i>	Mengembalik an user ke halaman <i>form</i> dengan <i>error</i>	<i>passed</i>

6	Klas PengujianController menjalankan <i>method</i> store() dari klas <i>TransactionController</i> dengan 'sparepart' : 'Velg Zigen Falcon P5M 17160/185 SupraGrandKharisma', 'service' : 'GantiVelg Balok' 'mechanic' : 'null' 'customer' : 'null'	Mengembalikan <i>user</i> ke halaman <i>form</i> dengan <i>error</i>	Mengembalikan <i>user</i> ke halaman <i>form</i> dengan <i>error</i>	<i>passed</i>
7	Klas PengujianController menjalankan <i>method</i> store() dari klas <i>TransactionController</i> dengan 'sparepart' : 'null', 'service' : "null" 'mechanic' : "null" 'customer' : 'lqbaal'	Mengembalikan <i>user</i> ke halaman <i>form</i> dengan <i>error</i>	Mengembalikan <i>user</i> ke halaman <i>form</i> dengan <i>error</i>	<i>passed</i>

5.3 Pengujian Validasi

5.3.1 Pengujian Validasi Menambah Transaksi Servis dan Suku Cadang

Tabel 5.9 merupakan hasil pengujian validasi menambah transaksi servis dan suku cadang.

Tabel 5.9 Pengujian Validasi Menambah Transaksi Servis dan Suku Cadang

Kode Kebutuhan		SIM-F-029		
Test case Number		VA-01		
Test case Name		Menambah Transaksi Servis Dan Suku Cadang		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> tambah data transaksi dengan lengkap	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' 'quantity' : '1' 'service' : 'Ganti Velg	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>

		Balok' 'mechanic' : 'Mubin' 'customer': 'Iqbaal'		
2.	Aktor mengisi form tambah data transaksi dengan tidak lengkap	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' 'quantity' : '1' 'service' : 'Ganti Velg Balok' 'mechanic' : 'Mubin' 'customer': 'null'	Aktor dikembalikan ke halaman form dengan error	<i>Passed</i>

5.3.2 Pengujian Validasi Melihat Suku Cadang Rendah

Tabel 5.10 merupakan hasil pengujian validasi melihat suku cadang rendah.

Tabel 5.10 Pengujian Validasi Melihat Suku Cadang Rendah

Kode Kebutuhan		SIM-F-002		
Test case Number		VA-02		
Test case Name		Melihat Suku Cadang Rendah		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman Laporan Stok Rendah	<i>Role = cashier & manager</i>	Sistem menampilkan halaman Laporan Stok Rendah	<i>Passed</i>

5.3.3 Pengujian Validasi Membeli Suku Cadang

Tabel 5.11 merupakan hasil pengujian validasi Membeli suku cadang.

Tabel 5.11 Pengujian Validasi Membeli Suku Cadang

Kode Kebutuhan		SIM-F-003		
Test case Number		VA-03		
Test case Name		Membeli Suku Cadang		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi form pembelian suku cadang dengan lengkap	'supplier' : 'PT. Honda Astra Motor' 'sparepart' : 'Bearing Ball 6300' 'category' : 'Honda' 'quantity' : '20' 'unit' : 'unit'	Data yang dimasukkan aktor tersimpan di dalam database	<i>Passed</i>

		'purchase_price' : 'Rp 14.580' 'selling_price' : 'Rp18.000' 'photo' : '25_bearing-ball-6300.jpg' 'photo wholesale' : '5_2021-06-15.png'		
2.	Aktor mengisi <i>form</i> pembelian suku cadang dengan tidak lengkap	'supplier' : 'PT. Honda Astra Motor' 'sparepart' : 'Bearing Ball 6300' 'category' : 'null' 'quantity' : '20' 'unit' : 'null' 'purchase_price' : 'Rp 14.580' 'selling_price' : 'Rp18.000' 'photo' : '25_bearing-ball-6300.jpg' 'photo wholesale' : '5_2021-06-15.png'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.4 Pengujian Validasi Melihat Suku Cadang

Berikut merupakan hasil pengujian validasi Melihat suku cadang dapat dilihat pada Tabel 5.12

Tabel 5.12 Pengujian Validasi Melihat Suku Cadang

Kode Kebutuhan		SIM-F-004		
Test case Number		VA-04		
Test case Name		Melihat Suku Cadang		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman suku cadang	Role = cashier & manager	Sistem menampilkan halaman Daftar suku cadang	<i>Passed</i>

5.3.5 Pengujian Validasi Mengubah Suku Cadang

Tabel 5.13 merupakan hasil pengujian validasi Mengubah suku cadang.

Tabel 5.13 Pengujian Validasi Mengubah Suku Cadang

Kode Kebutuhan		SIM-F-005		
Test case Number		VA-05		
Test case Name		Mengubah Suku Cadang		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> mengubah suku cadang dengan lengkap	'supplier' : 'PT. Honda Astra Motor' 'sparepart' : 'Bearing Ball 6300' 'category' : 'Honda' 'quantity' : '20' 'unit' : 'unit' 'purchase_price' : 'Rp 14.580' 'selling_price' : 'Rp18.000' 'photo' : '25_bearing-ball-6300.jpg'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2.	Aktor mengisi <i>form</i> mengubah suku cadang dengan tidak lengkap	'supplier' : 'PT. Honda Astra Motor' 'sparepart' : 'Bearing Ball 6300' 'category' : 'null' 'quantity' : '20' 'unit' : 'null' 'purchase_price' : 'Rp 14.580' 'selling_price' : 'Rp18.000' 'photo' : '25_bearing-ball-6300.jpg'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.6 Pengujian Validasi Menghapus Suku Cadang

Berikut merupakan hasil pengujian validasi Menghapus suku cadang dapat dilihat pada Tabel 5.14

Tabel 5.14 Pengujian Validasi Menghapus Suku Cadang

Kode Kebutuhan		SIM-F-006
Test case Number		VA-06
Test case Name		Menghapus Suku Cadang

No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor memilih data form yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus form ini”, aktor menekan tombol “Ya”	<i>Role = cashier</i>	Data form yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>
2.	Aktor memilih data form yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus form ini”, aktor menekan tombol “Tidak”	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman <i>servis</i> dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>

5.3.7 Pengujian Validasi Melihat Servis

Tabel 5.15 merupakan hasil pengujian validasi melihat servis.

Tabel 5.15 Pengujian Validasi Melihat Servis

Kode Kebutuhan		SIM-F-007		
<i>Test case Number</i>		VA-07		
<i>Test case Name</i>		Melihat Servis		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor membuka halaman servis	<i>Role = cashier</i>	Sistem menampilkan halaman Daftar suku cadang	<i>Passed</i>

5.3.8 Pengujian Validasi Menambah Servis

Tabel 5.16 merupakan hasil pengujian validasi menambah servis.

Tabel 5.16 Pengujian Validasi Menambah Servis

Kode Kebutuhan		SIM-F-008		
Test case Number		VA-08		
Test case Name		Menambah Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> menambah servis dengan lengkap	'name' : 'Bor', 'cost' : '5000'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> menambah servis dengan tidak lengkap	'name' : 'Bor', 'cost' : 'null'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.9 Pengujian Validasi Mengubah Servis

Tabel 5.17 merupakan hasil pengujian validasi mengubah servis.

Tabel 5.17 Pengujian Validasi Mengubah Servis

Kode Kebutuhan		SIM-F-009		
Test case Number		VA-09		
Test case Name		Mengubah Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> mengubah servis dengan lengkap	'name' : 'Bor', 'cost' : '5000'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> mengubah servis dengan tidak lengkap	'name' : 'Bor', 'cost' : 'null'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.10 Pengujian Validasi Menghapus Servis

Tabel 5.18 merupakan hasil pengujian validasi menghapus servis.

Tabel 5.18 Pengujian Validasi Menghapus Servis

Kode Kebutuhan		SIM-F-010		
Test case Number		VA-10		
Test case Name		Menghapus Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor memilih data form yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus form ini”, aktor menekan tombol “Ya”	<i>Role = cashier</i>	Data form yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>
2	Aktor memilih data form yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus form ini”, aktor menekan tombol “Tidak”	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman suku <i>mechanic</i> dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>

5.3.11 Pengujian Validasi Melihat *Mechanic*

Tabel 5.19 merupakan hasil pengujian validasi melihat *mechanic*.

Tabel 5.19 Pengujian Validasi Melihat *Mechanic*

Kode Kebutuhan		SIM-F-011		
Test case Number		VA-11		
Test case Name		Melihat <i>Mechanic</i>		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman <i>mechanic</i>	<i>Role = cashier</i>	Sistem menampilkan halaman	<i>Passed</i>

			<i>mechanic</i>	
--	--	--	-----------------	--

5.3.12 Pengujian Validasi Menambah *Mechanic*

Tabel 6.20 merupakan hasil pengujian validasi menambah *mechanic*.

Tabel 6.20 Pengujian Validasi Menambah *Mechanic*

Kode Kebutuhan		SIM-F-012		
Test case Number		VA-12		
Test case Name		Menambah <i>Mechanic</i>		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> menambah <i>mechanic</i> dengan lengkap	'name' : 'Mubin', 'age' : '33', 'address' : 'Malang', 'worked_since' : '1997', 'user_id' : '7'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> menambah <i>mechanic</i> dengan tidak lengkap	'name' : 'Mubin', 'age' : 'a', 'address' : 'Malang', 'worked_since' : '1997', 'user_id' : '7'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.13 Pengujian Validasi Mengubah *Mechanic*

Tabel 5.21 merupakan hasil pengujian validasi mengubah *mechanic*.

Tabel 5.21 Pengujian Validasi Mengubah *Mechanic*

Kode Kebutuhan		SIM-F-013		
Test case Number		VA-13		
Test case Name		Mengubah <i>Mechanic</i>		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> Mengubah <i>mechanic</i> dengan lengkap	'name' : 'Mubin', 'age' : '33', 'address' : 'Malang', 'worked_since' : '1997', 'user_id' : '7'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> Mengubah	'name' : 'Mubin', 'age' : 'a',	Aktor dikembalikan	<i>Passed</i>

	<i>mechanic</i> dengan tidak lengkap	'address' : 'Malang', 'worked_since' : '1997', 'user_id' : '7'	ke halaman <i>form</i> dengan <i>error</i>	
--	--------------------------------------	-------------------------------------------------------------------------	--------------------------------------------------	--

5.3.14 Pengujian Validasi Menghapus *Mechanic*

Tabel 5.22 merupakan hasil pengujian validasi menghapus *mechanic*.

Tabel 5.22 Pengujian Validasi Menghapus *Mechanic*

Kode Kebutuhan		SIM-F-014		
Test case Number		VA-14		
Test case Name		Menghapus <i>Mechanic</i>		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor memilih data form yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus form ini”, aktor menekan tombol “Ya”	<i>Role = cashier</i>	Data form yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>
2	Aktor memilih data form yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus form ini”, aktor menekan tombol “Tidak”	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman suku cadaang dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>

5.3.15 Pengujian Validasi Melihat Gaji *Mechanic*

Tabel 5.23 merupakan hasil pengujian validasi melihat gaji *mechanic*.

Tabel 5.23 Pengujian Validasi Melihat Gaji *Mechanic*

Kode Kebutuhan	SIM-F-015
Test case Number	VA-15

<i>Test case Name</i>		Melihat Gaji <i>Mechanic</i>		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor membuka halaman Gaji <i>mechanic</i>	<i>Role = cashier</i>	Sistem menampilkan halaman Gaji <i>mechanic</i>	<i>Passed</i>

5.3.16 Pengujian Validasi Mengubah Status Peminjaman *Mechanic*

Tabel 6.24 merupakan hasil pengujian validasi Mengubah Status Peminjaman *Mechanic*.

Tabel 6.24 Pengujian Validasi Mengubah Status Peminjaman *Mechanic*

Kode Kebutuhan		SIM-F-016		
<i>Test case Number</i>		VA-16		
<i>Test case Name</i>		Mengubah Status Peminjaman <i>Mechanic</i>		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor memilih data yang ingin diubah, dan aktor menekan tombol “Setuju”	<i>Role = cashier</i>	Data yang dipilih aktor mengalami perubahan dari <i>database</i>	<i>Passed</i>
2	Aktor memilih data form yang ingin diubah, dan aktor menekan tombol “Tidak Setuju”	<i>Role = cashier</i>	Data yang dipilih aktor mengalami perubahan dari <i>database</i>	<i>Passed</i>

5.3.17 Pengujian Validasi Menambah Pelanggan

Tabel 5.25 merupakan hasil pengujian validasi menambah pelanggan.

Tabel 5.25 Pengujian Validasi Menambah Pelanggan

Kode Kebutuhan		SIM-F-017		
<i>Test case Number</i>		VA-17		
<i>Test case Name</i>		Menambah Pelanggan		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor mengisi <i>form</i> menambah pelanggan	'name' : 'Iqbaal', 'phone' : '123',	Data yang dimasukkan aktor tersimpan di	<i>Passed</i>

	dengan lengkap		dalam <i>database</i>	
2	Aktor mengisi <i>form</i> menambah pelanggan dengan tidak lengkap	'name' : 'Iqbaal', 'phone' : 'a',	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.18 Pengujian Validasi Melihat Pelanggan

Tabel 5.26 merupakan hasil pengujian validasi melihat pelanggan.

Tabel 5.26 Pengujian Validasi Melihat Pelanggan

Kode Kebutuhan		SIM-F-018		
Test case Number		VA-18		
Test case Name		Melihat pelanggan		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman pelanggan	<i>Role = cashier</i>	Sistem menampilkan halaman pelanggan	<i>Passed</i>

5.3.19 Pengujian Validasi Mengubah Pelanggan

Tabel 6.27 merupakan hasil pengujian validasi mengubah pelanggan.

Tabel 6.27 Pengujian Validasi Mengubah Pelanggan

Kode Kebutuhan		SIM-F-019		
Test case Number		VA-19		
Test case Name		Mengubah Pelanggan		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> Mengubah pelanggan dengan lengkap	'name' : 'Iqbaal', 'phone' : '123',	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> Mengubah pelanggan dengan tidak lengkap	'name' : 'Iqbaal', 'phone' : 'null',	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.20 Pengujian Validasi Menghapus Pelanggan

Tabel 5.28 merupakan hasil pengujian validasi menghapus pelanggan.

Tabel 5.28 Pengujian Validasi Menghapus Pelanggan

Kode Kebutuhan		SIM-F-020		
Test case Number		VA-20		
Test case Name		Menghapus Pelanggan		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor memilih data pelanggan yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Ya”	<i>Role = cashier</i>	Data form yang dipilih aktor terhapus dari database	<i>Passed</i>
2	Aktor memilih data pelanggan yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Tidak”	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman pelanggan dan form yang dipilih tidak terhapus dari database	<i>Passed</i>

5.3.21 Pengujian Validasi Menambah Transaksi Penjualan Suku Cadang

Tabel 5.29 merupakan hasil pengujian validasi menambah transaksi penjualan suku cadang.

Tabel 5.29 Pengujian Validasi Menambah Transaksi Penjualan Suku Cadang

Kode Kebutuhan		SIM-F-021		
Test case Number		VA-21		
Test case Name		Menambah Transaksi Penjualan Suku Cadang		
No.	Prosedur Uji	Input	Expected result	Status

1.	Aktor mengisi <i>form</i> menambah Transaksi yaitu suku cadang dan pelanggan dengan lengkap	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' 'customer': 'Iqbaal'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> menambah Transaksi yaitu suku cadang dan pelanggan dengan tidak lengkap	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' 'customer': 'null'	Aktor dikembalikan ke halaman <i>form</i> transaksi dengan <i>error</i>	<i>Passed</i>

5.3.22 Pengujian Validasi Melihat Transaksi Penjualan Suku Cadang

Tabel 5.30 merupakan hasil pengujian validasi melihat pelanggan.

Tabel 5.30 Pengujian Validasi Melihat Pelanggan

Kode Kebutuhan		SIM-F-022		
Test case Number		VA-22		
Test case Name		Melihat Pelanggan		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman Transaksi	Role = cashier	Sistem menampilkan halaman Transaksi	<i>Passed</i>

5.3.23 Pengujian Validasi Mengubah Transaksi Pembelian Suku Cadang

Tabel 5.31 merupakan hasil pengujian validasi mengubah transaksi pembelian suku cadang.

Tabel 5.31 Pengujian Validasi Mengubah Transaksi Pembelian Suku Cadang

Kode Kebutuhan		SIM-F-023		
Test case Number		VA-23		
Test case Name		Mengubah Transaksi Pembelian suku cadang		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> mengubah Transaksi yaitu suku cadang	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma'	Data yang dimasukkan aktor tersimpan di	<i>Passed</i>

	dan pelanggan dengan lengkap	'customer': 'Iqbaal'	dalam <i>database</i>	
2	Aktor mengisi <i>form</i> mengubah Transaksi yaitu suku cadang dan pelanggan dengan tidak lengkap	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' 'customer': 'null'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.24 Pengujian Validasi Menghapus Transaksi Pembelian Suku Cadang

Tabel 5.32 merupakan hasil pengujian validasi menghapus transaksi pembelian suku cadang.

Tabel 5.32 Pengujian Validasi Menghapus Transaksi Pembelian Suku Cadang

Kode Kebutuhan		SIM-F-024		
Test case Number		VA-24		
Test case Name		Menghapus Transaksi Pembelian Suku Cadang		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Ya”	<i>Role = cashier</i>	Data form yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>
2	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Tidak”	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman transaksi dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>

5.3.25 Pengujian Validasi Menambah Transaksi Servis

Tabel 5.33 merupakan hasil pengujian validasi menambah transaksi servis.

Tabel 5.33 Pengujian Validasi Menambah Transaksi Servis

Kode Kebutuhan		SIM-F-025		
Test case Number		VA-25		
Test case Name		Menambah Transaksi Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> menambah Transaksi yaitu servis, <i>mechanic</i> dan pelanggan dengan lengkap	'service' : 'Ganti velg balok' 'mechanic' : 'Mubin' 'customer': 'Iqbaal'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> menambah Transaksi yaitu servis, <i>mechanic</i> dan pelanggan dengan tidak lengkap	'service' : 'Ganti velg balok' 'mechanic' : 'null' 'customer': 'null'	Aktor dikembalikan ke halaman <i>form</i> transaksi dengan <i>error</i>	<i>Passed</i>

5.3.26 Pengujian Validasi Melihat Transaksi Servis

Tabel 5.34 merupakan hasil pengujian validasi melihat transaksi servis.

Tabel 5.34 Pengujian Validasi Melihat Transaksi Servis

Kode Kebutuhan		SIM-F-026		
Test case Number		VA-26		
Test case Name		Melihat Transaksi Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman Transaksi	<i>Role = cashier</i>	Sistem menampilkan halaman Transaksi	<i>Passed</i>

5.3.27 Pengujian Validasi Mengubah Transaksi Servis

Tabel 5.35 merupakan hasil pengujian validasi mengubah transaksi servis.

Tabel 5.35 Pengujian Validasi Mengubah Transaksi Servis

Kode Kebutuhan		SIM-F-027		
Test case Number		VA-27		
Test case Name		Mengubah Transaksi Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> mengubah Transaksi yaitu servis, <i>mechanic</i> dan pelanggan dengan lengkap	'service' : 'Ganti velg balok' 'mechanic' : 'Mubin' 'customer': 'Iqbaal'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> mengubah Transaksi yaitu servis, <i>mechanic</i> dan pelanggan dengan tidak lengkap	'service' : 'Ganti velg balok' 'mechanic' : 'null' 'customer': 'null'	Aktor dikembalikan ke halaman <i>form</i> transaksi dengan <i>error</i>	<i>Passed</i>

5.3.28 Pengujian Validasi Menghapus Transaksi Servis

Tabel 5.36 merupakan hasil pengujian validasi menghapus transaksi servis.

Tabel 5.36 Pengujian Validasi Menghapus Transaksi Servis

Kode Kebutuhan		SIM-F-028		
Test case Number		VA-28		
Test case Name		Menghapus Transaksi Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan "Apakah Anda yakin ingin menghapus data ini", aktor menekan tombol "Ya"	<i>Role = cashier</i>	Data form yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>

2	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Tidak”	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman transaksi dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>
---	--------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------	---------------------------------------------------------------------------------------------------------------	---------------

5.3.29 Pengujian Validasi Melihat Transaksi Servis dan Suku Cadang

Tabel 5.37 merupakan hasil pengujian validasi melihat transaksi servis dan suku cadang.

Tabel 5.37 Pengujian Validasi Melihat Transaksi Servis dan Suku Cadang

Kode Kebutuhan		SIM-F-030		
Test case Number		VA-29		
Test case Name		Melihat Transaksi Servis Dan Suku Cadang		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor membuka halaman Transaksi	<i>Role = cashier</i>	Sistem menampilkan halaman Transaksi	<i>Passed</i>

5.3.30 Pengujian Validasi Mengubah Transaksi Servis dan Suku Cadang

Tabel 5.38 merupakan hasil pengujian validasi mengubah transaksi servis dan suku cadang.

Tabel 5.38 Pengujian Validasi Mengubah Transaksi Servis dan Suku Cadang

Kode Kebutuhan		SIM-F-031		
Test case Number		VA-30		
Test case Name		Mengubah Transaksi servis dan suku cadang		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor mengisi <i>form</i> mengubah Transaksi dengan lengkap	'sparepart' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' 'quantity' : '1' 'service' : 'Ganti Velg	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>

		Balok' ' <i>mechanic</i> ' : 'Mubin' ' <i>customer</i> ': 'Iqbaal'		
2	Aktor mengisi <i>form</i> mengubah Transaksi dengan tidak lengkap	' <i>sparepart</i> ' : 'Velg Zigen Falcon P5M 17-160/185 Supra-Grand-Kharisma' ' <i>quantity</i> ' : '1' ' <i>service</i> ' : 'Ganti Velg Balok' ' <i>mechanic</i> ' : 'Mubin' ' <i>customer</i> ': 'null'	Aktor dikembalikan ke halaman <i>form</i> dengan <i>error</i>	<i>Passed</i>

5.3.31 Pengujian Validasi Menghapus Transaksi Servis dan Suku Cadang

Tabel 5.39 merupakan hasil pengujian validasi menghapus transaksi servis dan suku cadang.

Tabel 5.39 Pengujian Validasi Menghapus Transaksi Servis dan Suku Cadang

Kode Kebutuhan		SIM-F-032		
Test case Number		VA-31		
Test case Name		Menghapus Transaksi Servis Dan Suku Cadang		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan "Apakah Anda yakin ingin menghapus data ini", aktor menekan tombol "Ya"	<i>Role = cashier</i>	Data <i>form</i> yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>
2	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan "Apakah Anda yakin ingin menghapus data ini", aktor	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman transaksi dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>

	menekan tombol “Tidak”			
--	------------------------	--	--	--

5.3.32 Pengujian Validasi Menambah Pinjaman

Tabel 5.40 merupakan hasil pengujian validasi menambah pinjaman.

Tabel 5.40 Pengujian Validasi Menambah Pinjaman

Kode Kebutuhan		SIM-F-033		
Test case Number		VA-32		
Test case Name		Menambah Pinjaman		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> menambah Pinjaman dengan lengkap	'amount' : '500000' 'percentage': '10' 'info': 'Kebutuhan Keluarga' 'created_at' : '2021-06-17'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> menambah Pinjaman dengan tidak lengkap	'amount' : '500000' 'percentage': '10' 'info': 'null' 'created_at' : '2021-06-17'	Aktor dikembalikan ke halaman <i>form</i> transaksi dengan error	<i>Passed</i>

5.3.33 Pengujian Validasi Melihat Pinjaman

Tabel 5.41 merupakan hasil pengujian validasi melihat pinjaman.

Tabel 5.41 Pengujian Validasi Melihat Pinjaman

Kode Kebutuhan		SIM-F-034		
Test case Number		VA-33		
Test case Name		Melihat Pinjaman		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor membuka halaman Pinjaman	<i>Role = cashier</i>	Sistem menampilkan halaman Transaksi	<i>Passed</i>

5.3.34 Pengujian Validasi Mengubah Pinjaman

Tabel 5.42 merupakan hasil pengujian validasi mengubah pinjaman servis.

Tabel 5.42 Pengujian Validasi Mengubah Pinjaman

Kode Kebutuhan		SIM-F-035		
Test case Number		VA-34		
Test case Name		Mengubah Pinjaman		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor mengisi <i>form</i> menambah Pinjaman dengan lengkap, jika belum di setujui	'amount' : '500000' 'percentage': '10' 'info': 'Kebutuhan Keluarga' 'created_at' : '2021-06-17'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>
2	Aktor mengisi <i>form</i> menambah Pinjaman dengan tidak lengkap jika belum di setujui	'amount' : '500000' 'percentage': '10' 'info': 'null' 'created_at' : '2021-06-17'	Aktor dikembalikan ke halaman <i>form</i> transaksi dengan <i>error</i>	<i>Passed</i>

5.3.35 Pengujian Validasi Menghapus Transaksi Servis

Tabel 5.43 merupakan hasil pengujian validasi menghapus transaksi servis.

Tabel 5.43 Pengujian Validasi Menghapus Transaksi Servis

Kode Kebutuhan		SIM-F-036		
Test case Number		VA-35		
Test case Name		Menghapus Transaksi Servis		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Ya”. Jika belum disetujui	<i>Role = cashier</i>	Data <i>form</i> yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>

2	Aktor memilih data transaksi yang ingin dihapus, dan ketika box menampilkan “Apakah Anda yakin ingin menghapus data ini”, aktor menekan tombol “Tidak”. Jika belum disetujui	<i>Role = cashier</i>	Aktor diarahkan kembali ke halaman transaksi dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>
---	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------	---------------------------------------------------------------------------------------------------------------	---------------

5.3.36 Pengujian Validasi Melihat Profil

Tabel 5.44 merupakan hasil pengujian validasi melihat profil.

Tabel 5.44 Pengujian Validasi Melihat Profil

Kode Kebutuhan		SIM-F-037		
Test case Number		VA-36		
Test case Name		Melihat Profil		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor membuka halaman Profil	<i>Role = cashier</i>	Sistem menampilkan halaman Profil	<i>Passed</i>

5.3.37 Pengujian Validasi Mengubah Profil

Tabel 5.45 merupakan hasil pengujian validasi mengubah profil.

Tabel 5.45 Pengujian Validasi Mengubah Profil

Kode Kebutuhan		SIM-F-038		
Test case Number		VA-37		
Test case Name		Mengubah Profil		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor mengisi <i>form</i> edit profil dengan lengkap	'nama' : 'Mubin' 'Tahun masuk': '2010' 'umur': '29' 'alamat' : 'Pongangan' 'password baru' : 'b'	Data yang dimasukkan aktor tersimpan di dalam <i>database</i>	<i>Passed</i>

		'password lama' : 'a'		
2	Aktor mengisi <i>form</i> edit profil dengan lengkap	'nama' : 'Mubin' 'Tahun masuk': 'null' 'umur': 'a' 'alamat' : 'null' 'password baru' : 'b' 'password lama' : 'a'	Aktor dikembalikan ke halaman <i>form</i> edit profil	<i>Passed</i>

5.3.38 Pengujian Validasi Mencetak *Invoice* Transaksi

Tabel 5.46 merupakan hasil pengujian validasi mencetak invoice transaksi.

Tabel 5.46 Pengujian Validasi Mencetak *Invoice* Transaksi

Kode Kebutuhan		SIM-F-039		
Test case Number		VA-38		
Test case Name		Menghapus Mencetak <i>Invoice</i> Transaksi		
No.	Prosedur Uji	Input	Expected result	Status
1.	Aktor memilih invoice transaksi yang ingin dicetak, dan ketika box menampilkan “Apakah Anda yakin ingin mengunduh file ini”, aktor menekan tombol “Ya”.	<i>Role = mechanic</i>	Data <i>form</i> yang dipilih aktor terhapus dari <i>database</i>	<i>Passed</i>
2	Aktor memilih invoice transaksi yang ingin dicetak, dan ketika box menampilkan “Apakah Anda yakin ingin mengunduh file ini”, aktor menekan tombol “Tidak”.	<i>Role = mechanic</i>	Aktor diarahkan kembali ke halaman suku cadaang dan <i>form</i> yang dipilih tidak terhapus dari <i>database</i>	<i>Passed</i>

5.3.39 Pengujian Validasi Logout

Tabel 5.47 merupakan hasil pengujian validasi logout.

Tabel 5.47 Pengujian Validasi Logout

Kode Kebutuhan		SIM-F-040		
Test case Number		VA-39		
Test case Name		Logout		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor memilih profil dan menekan tombol logout.	Role = Pengguna	Aktor keluar dari sistem	<i>Passed</i>

5.3.40 Pengujian Validasi Login

Tabel 5.48 merupakan hasil pengujian validasi login.

Tabel 5.48 Pengujian Validasi Login

Kode Kebutuhan		SIM-F-001		
Test case Number		VA-40		
Test case Name		Login		
No.	Prosedur Uji	Input	<i>Expected result</i>	Status
1.	Aktor mengisi form login dengan username dan password yang terdaftar di dalam sistem.	Role = Pengguna	Aktor masuk ke dalam sistem	<i>Passed</i>
2.	Aktor mengisi form login dengan username dan password yang tidak terdaftar di dalam sistem.	Role = Pengguna	Aktor tidak dapat masuk ke dalam sistem	<i>Passed</i>

5.3.41 Pengujian Validasi Menambah Laporan Suku Cadang Rendah

Tabel 5.49 merupakan hasil pengujian validasi menambah laporan suku cadang rendah.

Tabel 5.49 Pengujian Validasi Menambah Laporan Suku Cadang Rendah

Kode Kebutuhan		SIM-F-041		
Test case Number		VA-41		
Test case Name		Menambah Laporan Suku Cadang Rendah		
No.	Prosedur Uji	Input	Expected result	Status
1.	Stok suku cadang ≤ 10	Role = cashier, manager	Data suku cadang rendah ditambahkan	Passed
2.	Stok suku cadang > 10	Role = cashier, manager	Data suku cadang rendah tidak ditambahkan	Passed

5.3.42 Pengujian Compatibility

Pengujian *Compatibility* berfungsi menguji kebutuhan non-fungsional tahap perancangan. Pengembangan sistem ini tidak dibatasi menggunakan satu *web browser* maka perlu dilakukan pengujian *compabitbility*. Aplikasi *Sortsite* digunakan adalah *version 6.40.902.0*. dalam pengujian ini, dikarenakan mempunyai 3 nilai pada pengujian *compatibility*, yaitu *missing content of functionality* yang berarti ketika dilakukan pengujian pada *browser* terdapat fungsionalitas, *major layout / performance problem* dan *minor layout / performance*.

Browser	IE	Edge	Firefox	Safari	Opera	Chrome	iOS	Android	Key
Version	11	88	84	14	73	88	≤ 13 14	88	
Critical Issues	✓	✓	✓	✓	✓	✓	✓	✓	● Missing content or functionality
Major Issues	⚠	✓	⚠	⚠	✓	✓	⚠	⚠	⚠ Major layout or performance problems
Minor Issues	⚠	✓	⚠	⚠	✓	✓		✓	⚠ Minor layout or performance problems

Gambar 5.5 Hasil Pengujian Compatibility

Berdasarkan Gambar 5.5 bahwa pengujian menunjukkan hasil yang baik namun terdapat beberapa masalah yaitu *major layout* pada IOS kurang dari sama dengan dari 14, *Safari* versi 14, *Firefox* versi 84, *IE* versi 11, tetapi pengujian tidak ditemukan masalah pada *Edge* versi 88, *Opera* versi 73, *Chrome* versi 88 dan *Android* 88.

LAMPIRAN A HASIL WAWANCARA

Pewawancara	: Muhammad Amin Iqbaal Alam
Narasumber	: Bapak Fitrul (<i>Manager</i> Anwar Motor II)
Lokasi	: Bengkel Anwar Motor

1. Berapa jumlah Cabang Bengkel Anwar Motor?

Bengkel Anwar Motor memiliki 3 Cabang Bengkel di Gresik

2. Berapa jumlah jenis suku cadang yang dimiliki oleh Anwar Motor?

453 jenis suku cadang

3. Bagaimana alur untuk transaksi penjualan barang?

Pelanggan datang ke Anwar Motor menuju *cashier*, dan menanyakan kendala motor. *cashier* menyampaikan kendala motor ke *mechanic* untuk diservis, jika terdapat penggantian suku cadang maka *mechanic* menyampaikan suku cadang yang diganti untuk dicatat ke nota transaksi. Setelah selesai mengalami servis, pelanggan melakukan transaksi pembayaran. Jika tidak terdapat pergantian suku cadang maka yang dicatat dalam nota transaksi hanya servis. Namun jika pelanggan hanya ingin membeli suku cadang saja maka yang dicatat dalam nota transaksi hanya suku cadang yang dibeli.

4. Bagaimana alur untuk pembelian suku cadang ke *supplier*?

Menghubungi *supplier* untuk pembelian Suku Cadang. Pembelian Suku Cadang dilakukan apabila stok telah habis, dan *supplier* akan datang ke Bengkel Anwar Motor untuk memiliki suku cadang yang telah dipesan serta memiliki nota pembelian serta pengecekan suku cadang yang dibeli dari *supplier*.

5. Bagaimana penggajian *mechanic*?

Untuk penggajian *mechanic* di Anwar Motor sesuai dengan servis yang telah dikerjakan tanpa ada potongan apapun, untuk penggajiannya sebulan sekali.

6. Bagaimana pengajuan pinjaman *mechanic*?

Untuk pengajuan pinjaman *mechanic* di Anwar Motor dari *mechanic* mengajukan pinjaman beserta nominalnya, nanti dicatat dan ke buku keuangan, *manager* memiliki uang sesuai yang diajukan, untuk pembayaran pinjaman akan dipotong gaji *mechanic*.

LAMPIRAN B HASIL WAWANCARA

Pewawancara	: Muhammad Amin Iqbaal Alam
Narasumber	: Bapak Mubin (<i>Mechanic</i> Anwar Motor III)
Lokasi	: Bengkel Anwar Motor

1. Apa servis di Anwar Motor beserta harganya?

Untuk Bor seharga 5.000, Ganti bearing 15.000 sampai 30.000, Cek kelistrikan 25.000, Center Body 60.000, Ganti Piston 150000, Ganti V Belt 30000, Ganti Key Set 30.000, Ganti Cilinder Matic dan Bebek 150.000, Ganti Cilinder Sport 300.000, Ganti Valve 100.000, Ganti Race Set 80.000, Ganti Pipe Front 25.000, Ganti Piece Set 30.000, Ganti Pad Set 10.000, Ganti Fase 20.000, Ganti Filter Matic dan Bebek 10.000, Ganti Filter Sport 15.000, Ganti Knalpot 10.000, Ganti Lifter Assy 10.000, Ganti Handle 10.000, Ganti Tromol 30.000, Ganti Horn Comp 10.000, Ganti Gear Primary 50.000.

2. Apakah gaji *mechanic* itu sesuai servis yang dikerjakan?

Iyaa sesuai dengan yang dikerjakan, jadi penghasilan perbulannya tidak tetap

3. Berapa lama seluruh *mechanic* ini bekerja di bengkel Anwar Motor?

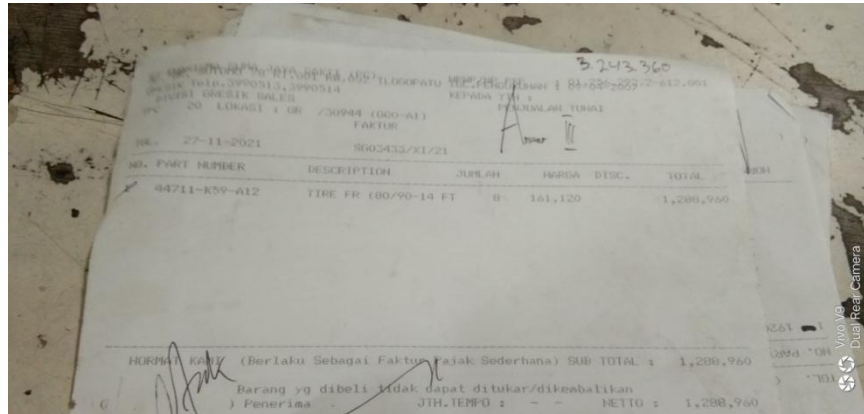
Ada yang 2 tahun, 5 tahun bahkan 8 tahun

4. Berapa rata-rata umur *mechanic* ini bekerja di bengkel Anwar Motor?

Umur 27 sampai 31

LAMPIRAN C DOKUMEN

1. Nota Pembelian Suku Cadang ke *supplier*



Gambar 7.1 Nota Pembelian Suku Cadang ke atau *supplier*

2. Nota Transaksi Penjualan Suku Cadang

[illegible]

Gambar 7.2 Nota Transaksi Penjualan Suku Cadang

3. Penggajian Karyawan



Gambar 7.3 Penggajian Karyawan