



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - K141502

RANCANG BANGUN APLIKASI INDUSTRI RUMAH TANGGA MAKANAN RINGAN BERBASIS WEB MENGUNAKAN METODE ALGORITMA APRIORI TID

HILMA KAMILAH
NRP 05111440000188

Dosen Pembimbing
Fajar Baskoro, S.Kom., M.T.
Adhatus Solichah Ahmadiyah, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018



TUGAS AKHIR - K141502

**RANCANG BANGUN APLIKASI INDUSTRI
RUMAH TANGGA MAKANAN RINGAN BERBASIS
WEB MENGGUNAKAN METODE ALGORITMA
APRIORI TID**

HILMA KAMILAH
NRP 05111440000188

Dosen Pembimbing
Fajar Baskoro, S.Kom., M.T.
Adhatu Solichah Ahmadiyah, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

[Halaman ini sengaja dikosongkan]



FINAL PROJECT - K141502

**DESIGN AND IMPLEMENTATION OF *WEB*-BASED
APPLICATIONS FOR HOUSEHOLD INDUSTRY USING
APRIORI TID ALGORITHM METHOD**

HILMA KAMILAH
NRP 05111440000188

Advisor
Fajar Baskoro, S.Kom., M.T.
Adhatu Solichah Ahmadiyah, S.Kom., M.Sc.

INFORMATICS DEPARTMENT
Faculty of Information and Communication Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2018

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI INDUSTRI RUMAH TANGGA MAKANAN RINGAN BERBASIS *WEB* MENGUNAKAN METODE ALGORITMA APRIORI TID

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Rekayasa Perangkat Lunak
Program Studi S-1 Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh :

HILMA KAMILAH
NRP : 05111440000188

Disetujui oleh Dosen Pembimbing Tugas Akhir

Fajar Baskoro, S.Kom., M.T.
NIP: 19740403 199903 1 002



Adhatus Solichah Ahmadiyah, S.Kom.
NIP: 19850826 201504 2 002

M. SOEPARTEMEN
TEKNIK INFORMATIKA
(pembimbing 2)

SURABAYA
JULI 2018

[Halaman ini sengaja dikosongkan]

RANCANG BANGUN APLIKASI INDUSTRI RUMAH TANGGA MAKANAN RINGAN BERBASIS WEB MENGUNAKAN METODE ALGORITMA APRIORI TID

Nama Mahasiswa : Hilma Kamilah
NRP : 0511144000188
Departemen : Informatika FTIK-ITS
Dosen Pembimbing 1 : Fajar Baskoro, S.Kom., M.T.
Dosen Pembimbing 2 : Adhatus Solichah A., S.Kom., M.Sc.

ABSTRAK

Perdagangan elektronik atau yang biasa disebut dengan e-commerce adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti internet atau televisi, atau jaringan komputer lainnya. Banyak pedagang yang mulai melakukan pemasaran melalui situs marketplace yang sudah ada seperti Tokopedia, Bukalapak, atau sosial media seperti Facebook dan Instagram, ada pula yang membangun website sendiri dengan menggunakan template Content Management Sistem (CMS) yang telah ada untuk penjualan dagangan mereka.

Namun ada saatnya dimana pedagang harus pindah dari sana dan mulai membangun website sendiri untuk membantu dalam persaingan pemasaran penjualan. Terutama dalam industri rumah tangga yang memiliki banyak persaingan secara online. Untuk mengatasi masalah ini, solusi yang penulis tawarkan yaitu membuat website industri rumah tangga khususnya Makaroni Mewek.

Dalam aplikasi ini akan digunakan algoritma Apriori untuk membantu penjual dalam mendapatkan rekomendasi produk dan ditampilkan pada halaman untuk pembeli, sehingga dapat membantu penjual ketika ingin melakukan pemasaran, dan lebih terfokus untuk memenuhi keinginan pembeli.

Kata Kunci: algoritma apriori, e-commerce, marketplace, website.

DESIGN AND IMPLEMENTATION OF *WEB*-BASED APPLICATIONS FOR HOUSEHOLD INDUSTRY USING APRIORI TID ALGORITHM METHOD

Name : Hilma Kamilah
NRP : 05111440000188
Major : Informatika FTIK-ITS
Supervisor I : Fajar Baskoro, S.Kom., M.T.
Supervisor II : Adhatu Solichah A., S.Kom., M.Sc.

ABSTRACT

Electronic commerce or commonly referred to e-commerce is the dissemination, purchase, sale, marketing of goods and services through electronic systems such as internet, television, or other computer networks. Many merchants start to market their product through existing marketplace sites like Tokopedia, Bukalapak, or social media like Facebook, Instagram, and build their own websites using their existing Content Management System (CMS) template for their merchandise sales.

However, there are times where the merchant must move and start building their own website to have advantage in the sales marketing competition. Especially in the home industry that has a lot of competition online. To overcome this problem, the solution the writer offer is to make the website of home industry especially Makaroni Mewek.

This application an Apriori algorithm will be used to assist sellers in getting product recommendations and displayed on the page for buyers, so as to help sellers when they want to do marketing, and more focused on meeting buyer's desires.

Keywords: *Apriori algorithm, e-commerce, marketplace, website.*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Alhamdulillahirabbil'alamiin, puji syukur kepada Tuhan Yang Maha Esa atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan Tugas Akhir yang berjudul:

RANCANG BANGUN APLIKASI INDUSTRI RUMAH TANGGA MAKANAN RINGAN BERBASIS *WEB* MENGGUNAKAN METODE ALGORITMA APRIORI TID

Melalui lembar ini, penulis ingin menyampaikan ucapan terima kasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah SWT atas segala nikmat dan rahmat yang telah diberikan selama ini.
2. Ayah, Umik, Eyang dan keluarga penulis yang tiada henti- hentinya mencurahkan kasih sayang, perhatian dan doa kepada penulis selama ini.
3. Bapak Fajar Baskoro dan Ibu Adhatus Solichah selaku dosen wali dan dosen pembimbing yang memberikan nasihat, arahan, dan bantuan sehingga penulis dapat menyelesaikan Tugas Akhir ini.
4. Bapak dan Ibu dosen Teknik Informatika ITS yang telah membina selama menempuh studi.
5. Raras, Sabila, Rani, Ical, Farhan, Bebet, Nafia, Upik, yang telah menemani, mendukung, mendengar keluh kesah dan membantu dalam hal pengerjaan tugas akhir, perkuliahan, maupun psikis.
6. Teman-teman belajar dan bermain, Sita, Dini, Nay, Rara, Delia, Ghaly, dan teman-teman angkatan 2014 selama 4 tahun kuliah yang telah menemani, membantu dan mendukung dalam perkuliahan maupun aktivitas lain.

7. PH HMTC Inspirasi yang telah memberikan pengalaman dalam berorganisasi, pemikiran-pemikiran menarik, dan program-program kerja yang sibuk.
8. Teman-teman Lab NCC, Rafiar, Hendri, Gleen, Risma, Zahri, Hero, Emak, Zulfa, Azki, Ubut, Ical, Zayn, dan Nuzha, yang menyediakan waktu, tempat, serta perhatian untuk menyemangati selama di Lab.
9. Teman-teman Departemen Dalam Negeri HMTC Inspirasi, terutama Afif, kemudian Huda, Rezky, Adam, Tegar, Yaskee, Nuzul, Nahda, Astrid, Yuga, Penyok, dan Ariya, yang telah banyak memberikan inspirasi, canda tawa, suka duka, dan bantuan selama kuliah.
10. Kakak-kakak senior, dan adik-adik, Purin, Pede, Kupon, Deka, Aidil, yang telah turut mengisi hari-hari penulis pada masa perkuliahan.
11. Teman-teman ICT Global Startup 2017, Keke, Anne, Marthin, Habib, Tri, Rinaldy, Sunwin, Black, Thida, Soe Soe Maw, dan Jun, yang berjuang bersama selama 4 bulan di Korea Selatan.
12. Teman-teman SMA yang tidak lupa menyemangati dan mendengar curhat mengenai perkuliahan maupun tidak.
13. Serta pihak-pihak lain yang namanya tidak dapat penulis sebutkan satu -persatu.

Bagaimanapun penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir, namun mohon maaf bila terdapat kekurangan, dan kesalahan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juli 2018

Hilma Kamilah

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxiii
DAFTAR KODE SEMU.....	xxvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Tujuan.....	3
1.3. Rumusan Permasalahan.....	4
1.4. Batasan Permasalahan	4
1.5. Metodologi	4
1.6. Sistematika Penulisan.....	6
BAB II TINJAUAN PUSTAKA.....	9
2.1. Perdagangan Elektronik.....	9
2.2. Industri Rumah Tangga	10
2.3. Apriori	11
2.4. Codeigniter	12
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	15
3.1. Analisis.....	15
3.1.1. Analisis Permasalahan.....	15
3.1.2. Deskripsi Umum Sistem.....	16
3.1.3. Spesifikasi Kebutuhan Perangkat Lunak.....	17
3.2 Perancangan Sistem.....	45
3.2.1 Perancangan Arsitektur	45
3.2.2 Perancangan Perangkat Lunak	46
3.2.3 Perancangan Antarmuka Pengguna	47
3.2.4 Perancangan Data	57
3.3. Perancangan Algoritma Apriori	67
BAB IV IMPLEMENTASI.....	71

4.1 Lingkungan Implementasi Perangkat Lunak	71
4.2 Implementasi Antarmuka Pengguna	71
4.2.1 Implementasi Halaman Antarmuka Melihat Informasi	72
4.2.2 Implementasi Halaman Antarmuka Mengelola Data Informasi	74
4.2.3 Implementasi Halaman Antarmuka Melihat Produk	79
4.2.4 Implementasi Halaman Antarmuka Mengelola Data Produk	80
4.2.5 Implementasi Halaman Antarmuka Melakukan Pemesanan Produk	83
4.2.6 Implementasi Halaman Antarmuka Melihat Riwayat Transaksi	85
4.2.7 Implementasi Halaman Antarmuka Melihat Detail Pembelian	85
4.2.8 Implementasi Halaman Antarmuka Menghapus Data Pembelian	86
4.2.9 Implementasi Halaman Antarmuka Melakukan Pembayaran	87
4.2.10 Implementasi Halaman Antarmuka Mengelola Data Transaksi	88
4.2.11 Implementasi Halaman Antarmuka Melihat Tanya Jawab	89
4.2.12 Implementasi Halaman Antarmuka Mengirim Pertanyaan	90
4.2.13 Implementasi Halaman Antarmuka Menjawab Pertanyaan	91
4.2.14 Implementasi Halaman Antarmuka Menambah Pengguna Pembeli	92
4.2.15 Implementasi Halaman Antarmuka Menambah Pengguna sebagai Administrator/ <i>Reseller</i>	93
4.2.16 Implementasi Halaman Antarmuka Melihat Profil	93

4.2.17. Implementasi Halaman Antarmuka Melihat Riwayat Produk	95
4.3. Implementasi Algoritma Apriori	97
BAB V PENGUJIAN DAN EVALUASI	99
5.1. Lingkungan Pengujian.....	99
5.2. Pengujian Fungsionalitas.....	99
5.2.1. Skenario Pengujian Fungsionalitas Pembeli/ <i>Reseller</i>	99
5.2.2. Hasil Pengujian Fungsionalitas Pembeli/ <i>Reseller</i>	106
5.2.3. Pengujian Fungsionalitas Administrator	113
5.3. Pengujian Non-Fungsional Pembeli	126
5.4. Pengujian Algoritma Apriori.....	128
BAB VI KESIMPULAN DAN SARAN.....	135
6.1. Kesimpulan.....	135
6.2. Saran.....	135
DAFTAR PUSTAKA.....	137
LAMPIRAN A. KODE SEMU	139
BIODATA PENULIS.....	143

[Halaman ini sengaja dikosongkan]

DAFTAR GAMBAR

Gambar 2.1 Alur Kerja Framework Codeigniter.....	13
Gambar 3.1 Diagram Kasus Penggunaan.....	20
Gambar 3.2 Diagram Aktivitas UC001.....	22
Gambar 3.3 Diagram Aktivitas UC002.....	24
Gambar 3.4 Diagram Aktivitas UC003.....	25
Gambar 3.5 Diagram Aktivitas UC004.....	28
Gambar 3.6 Diagram Aktivitas UC005.....	30
Gambar 3.7 Diagram Aktivitas UC006.....	31
Gambar 3.8 Diagram Aktivitas UC007.....	32
Gambar 3.9 Diagram Aktivitas UC008.....	33
Gambar 3.10 Diagram Aktivitas UC009.....	35
Gambar 3.11 Diagram Aktivitas UC010.....	37
Gambar 3.12 Diagram Aktivitas UC011.....	38
Gambar 3.13 Diagram Aktivitas UC012.....	39
Gambar 3.14 Diagram Aktivitas UC013.....	40
Gambar 3.15 Diagram Aktivitas UC014.....	41
Gambar 3.16 Diagram Aktivitas UC015.....	43
Gambar 3.17 Diagram Aktivitas UC016.....	44
Gambar 3.18 Diagram Aktivitas UC017.....	44
Gambar 3.19 Rancangan Halaman Antarmuka Tambah Informasi	47
Gambar 3.20 Rancangan Halaman Antarmuka <i>Edit</i> Data Informasi	48
Gambar 3.21 Rancangan Antarmuka Menambah Data Produk ..	49
Gambar 3.22 Rancangan Halaman Antarmuka Mengubah Data Produk	50
Gambar 3.23 Rancangan Halaman Antarmuka Mengelola Data Transaksi	51
Gambar 3.24 Rancangan Halaman Antarmuka Mengubah Data Transaksi	52
Gambar 3.25 Rancangan Halaman Antarmuka Melihat Detail Data Transaksi Tertentu.....	52

Gambar 3.26 Rancangan Halaman Antarmuka Menambah Pengguna sebagai Pembeli	54
Gambar 3.27 Rancangan Halaman Antarmuka Masuk Pengguna	55
Gambar 3.28 Rancangan Halaman Antarmuka Melihat Profil....	56
Gambar 3.29 Rancangan Halaman Antarmuka Mengelola Data Tanya Jawab	57
Gambar 3.30 <i>Conceptual Data Modeling</i>	65
Gambar 3.31 <i>Physical Data Modeling</i>	66
Gambar 3.32 Langkah-langkah perhitungan Apriori.....	69
Gambar 4.1 Antarmuka Melihat Data Informasi.....	73
Gambar 4.2 Antarmuka Kelola Data Informasi.....	74
Gambar 4.3 Antarmuka Menambah Data Informasi	75
Gambar 4.4 Antarmuka Melihat Detail Data Informasi	76
Gambar 4.5 Antarmuka Mengubah Data Informasi	77
Gambar 4.6 Antarmuka sebelum Menghapus Data Informasi	79
Gambar 4.7 Antarmuka sesudah Menghapus Data Informasi	79
Gambar 4.8 Antarmuka Melihat Data Produk.....	80
Gambar 4.9 Antarmuka Kelola Data Produk.....	81
Gambar 4.10 Antarmuka Menambah Data Produk	82
Gambar 4.11 Antarmuka Mengubah Data Produk	82
Gambar 4.12 Antarmuka Melihat Detail Produk.....	82
Gambar 4.13 Antarmuka Melakukan Pemesanan Produk.....	83
Gambar 4.14 Antarmuka Konfirmasi Pembelian	83
Gambar 4.15 Antarmuka Melihat Riwayat Transaksi	85
Gambar 4.16 Antarmuka Melihat Detail Pembelian	86
Gambar 4.17 Antarmuka Menghapus Data Pembelian	86
Gambar 4.18 Antarmuka Melakukan Pembayaran.....	87
Gambar 4.19 Antarmuka Mengelola Data Transaksi	88
Gambar 4.20 Antarmuka melihat detail dan menggunggah resi kirim	89
Gambar 4.21 Antarmuka Melihat Tanya Jawab untuk Pengunjung, Pembeli, dan <i>Reseller</i>	90
Gambar 4.22 Antarmuka Melihat Tanya Jawab untuk Administrator.....	90

Gambar 4.23 Antarmuka Mengirim Pertanyaan	91
Gambar 4.24 Antarmuka Menjawab Pertanyaan.....	91
Gambar 4.25 Antarmuka Menambah Pengguna Pembeli	92
Gambar 4.26 Antarmuka Menambah Pengguna Administrator/ <i>Reseller</i>	94
Gambar 4.27 Antarmuka Melihat Profil Pengguna	94
Gambar 4.28 Antarmuka Halaman Riwayat Produk.....	95
Gambar 4.29 Hasil Perhitungan Algoritma Apriori	95
Gambar 4.30 Antarmuka Hasil Perhitungan Apriori.....	96
Gambar 4.31 Halaman Relasi Produk Pembeli	97
Gambar 5.1 Halaman Menu Utama.....	107
Gambar 5.2 Halaman Produk	107
Gambar 5.3 Kotak Keranjang Belanja	108
Gambar 5.4 Halaman Konfirmasi Pembelian.....	108
Gambar 5.5 Halaman Riwayat Transaksi	108
Gambar 5.6 Halaman Detail Data Transaksi	109
Gambar 5.7 Kotak <i>pop-up</i> menghapus data transaksi	109
Gambar 5.8 Halaman Pembayaran Berhasil.....	109
Gambar 5.9 Halaman Pembayaran Gagal	110
Gambar 5.10 Halaman Forum Tanya Jawab	110
Gambar 5.11 Halaman Kontak.....	111
Gambar 5.12 Halaman Daftar sebagai Pembeli	111
Gambar 5.13 Halaman Profil.....	112
Gambar 5.14 Halaman Tambah Data Informasi.....	115
Gambar 5.15 Halaman Ubah Data Informasi	115
Gambar 5.16 Halaman Detail Data Informasi	116
Gambar 5.17 Halaman Tambah Data Produk.....	118
Gambar 5.18 Halaman Ubah Data Produk	118
Gambar 5.19 Halaman Detail Data Produk	119
Gambar 5.20 Halaman Data Transaksi ketika Unggah Resi Kirim Berhasil.....	121
Gambar 5.21 Halaman Data Transaksi ketika Unggah Resi Kirim Gagal	121
Gambar 5.22 Halaman Mengisi Jawaban	122
Gambar 5.23 Halaman Daftar sebagai Administrator/ <i>Reseller</i> ..	124

Gambar 5.24 Halaman Riwayat Produk sebelum mengisi form	125
Gambar 5.25 Halaman Riwayat Produk setelah mengisi form..	125
Gambar 5.26 Halaman Relasi Produk pada Pembeli.....	125
Gambar 5.27 <i>Input minimum support</i> dan <i>minimum confidence</i>	129
Gambar 5.28 Hitung nilai <i>support</i> setiap <i>item</i>	129
Gambar 5.29 Ambil <i>item</i> yang memenuhi <i>minimum support</i> ...	130
Gambar 5.30 Kombinasi <i>Frequent item</i> dan <i>k-itemset</i> dan hitung nilai <i>support</i> -nya.....	130
Gambar 5.31 Ambil <i>item</i> yang memenuhi nilai <i>minimum support</i>	131
Gambar 5.32 Kombinasi dengan <i>k-itemset</i> sebelumnya dan hitung nilai <i>support</i>	131
Gambar 5.33 Ambil <i>item</i> yang memenuhi nilai <i>minimum support</i>	132
Gambar 5.34 Kombinasikan kembali dengan <i>k-itemset</i> dan hitung nilai <i>support</i> -nya.....	132
Gambar 5.35 Hingga didapatkan <i>Candidate item</i> terakhir dan menghasilkan aturan asosiasi.....	133

DAFTAR TABEL

Tabel 2.1 Perbandingan <i>marketplace</i> dengan <i>website</i> milik sendiri	10
Tabel 2.2 Perbandingan <i>Website</i> Industri Lain dengan Milik Sendiri	11
Tabel 3.1 Aktor pada Sistem	18
Tabel 3.2 Kebutuhan Fungsional pada Sistem	19
Tabel 3.3 Kebutuhan Non-Fungsional pada Sistem	20
Tabel 3.4 Deskripsi Kasus Penggunaan Sistem	21
Tabel 3.5 Rincian Alur Kasus Penggunaan Melihat Informasi ...	22
Tabel 3.6 Rincian Alur Kasus Penggunaan Mengelola Data Informasi	23
Tabel 3.7 Rincian Alur Kasus Penggunaan Melihat Produk	25
Tabel 3.8 Rincian Alur Kasus Penggunaan Mengelola Data Produk	26
Tabel 3.9 Rincian Alur Kasus Penggunaan Melakukan Pemesanan	29
Tabel 3.10 Rincian Alur Kasus Penggunaan Melihat Riwayat Transaksi	31
Tabel 3.11 Rincian Alur Kasus Penggunaan Melihat Detail Pemesanan	32
Tabel 3.12 Rincian Alur Kasus Penggunaan Menghapus Pembelian	33
Tabel 3.13 Rincian Alur Kasus Penggunaan Melakukan Pembyaran	34
Tabel 3.14 Rincian Alur Kasus Penggunaan Mengelola Data Transaksi	35
Tabel 3.15 Rincian Alur Kasus Penggunaan Melihat Forum Tanya Jawab	37
Tabel 3.16 Rincian Alur Kasus Penggunaan Mengirim Pertanyaan	38
Tabel 3.17 Rincian Alur Kasus Penggunaan Menjawab Pertanyaan	39

Tabel 3.18 Rincian Alur Kasus Penggunaan Menambah Pengguna Pembeli.....	41
Tabel 3.19 Rincian Alur Kasus Penggunaan Menambah Pengguna Administrator/ <i>Reseller</i>	42
Tabel 3.20 Rincian Alur Kasus Penggunaan Melihat Profil.....	43
Tabel 3.21 Rincian Alur Kasus Penggunaan Melihat Riwayat Pembelian Produk.....	45
Tabel 3.22 Perancangan Perangkat Lunak	46
Tabel 3.23 Atribut Antarmuka Kelola Data Informasi	48
Tabel 3.24 Atribut Antarmuka Mengelola Data Produk	50
Tabel 3.25 Atribut Antarmuka Mengelola Transaksi	53
Tabel 3.26 Atribut Antarmuka Menambah Pengguna	53
Tabel 3.27 Atribut Antarmuka Masuk Pengguna	55
Tabel 3.28 Atribut Antarmuka Melihat Profil Pengguna	56
Tabel 3.29 Atribut Antarmuka Mengelola Data Tanya Jawab	57
Tabel 3.30 Kamus Data Tabel User.....	58
Tabel 3.31 Kamus Data Tabel Role.....	59
Tabel 3.32 Kamus Data Tabel Informasi.....	60
Tabel 3.33 Kamus Data Tabel Jenis Info	60
Tabel 3.34 Kamus Data Tabel Produk.....	61
Tabel 3.35 Kamus Data Tabel Pertanyaan	62
Tabel 3.36 Kamus Data Tabel Transaksi.....	63
Tabel 3.37 Kamus Data Tabel History Pembelian	64
Tabel 4.1 Lingkungan Implementasi Perangkat Lunak	71
Tabel 5.1 Lingkungan Pengujian Fungsionalitas Perangkat Lunak	99
Tabel 5.2 Skenario 1 Pengujian Melihat Informasi	100
Tabel 5.3 Skenario 2 Pengujian Melihat Produk.....	100
Tabel 5.4 Skenario 3 Pengujian Melakukan Pemesanan	101
Tabel 5.5 Skenario 4 Pengujian Melihat Riwayat Transaksi.....	101
Tabel 5.6 Skenario 5 Pengujian Melihat Detail Pembelian	102
Tabel 5.7 Skenario 6 Pengujian Menghapus Data Pembelian ...	102
Tabel 5.8 Skenario 7 Pengujian Melakukan Pembayaran	103
Tabel 5.9 Skenario 8 Pengujian Melakukan Pembayaran	104
Tabel 5.10 Skenario 9 Pengujian Melihat Forum Tanya Jawab	104

Tabel 5.11 Skenario 10 Pengujian Mengirim Pertanyaan	105
Tabel 5.12 Skenario 11 Pengujian Menambah Pengguna Pembeli	105
Tabel 5.13 Skenario 12 Pengujian Melihat Profil Pengguna.....	106
Tabel 5.14 Hasil Kuisioner Pengujian Fungsionalitas	113
Tabel 5.15 Skenario 13 Pengujian Fungsionalitas Mengelola Data Informasi	114
Tabel 5.16 Skenario 14 Pengujian Fungsionalitas Mengelola Data Informasi	114
Tabel 5.17 Skenario 15 Pengujian Fungsionalitas Mengelola Data Informasi	115
Tabel 5.18 Skenario 16 Pengujian Fungsionalitas Menghapus Data Informasi	116
Tabel 5.19 Skenario 17 Pengujian Fungsionalitas Mengelola Data Produk	117
Tabel 5.20 Skenario 18 Pengujian Fungsionalitas Mengelola Data Produk	117
Tabel 5.21 Skenario 19 Pengujian Fungsionalitas Mengelola Data Produk	118
Tabel 5.22 Skenario 20 Pengujian Fungsionalitas Menghapus Data Produk	119
Tabel 5.23 Skenario 21 Pengujian Fungsionalitas Mengelola Transaksi	120
Tabel 5.24 Skenario 22 Pengujian Fungsionalitas Mengelola Transaksi	121
Tabel 5.25 Skenario 23 Pengujian Fungsionalitas Menjawab Pertanyaan	122
Tabel 5.26 Skenario 24 Pengujian Fungsionalitas Menambah Pengguna Administrator/ <i>Reseller</i>	123
Tabel 5.27 Skenario 25 Pengujian Fungsionalitas Melihat Riwayat Produk	124
Tabel 5.28 Hasil Survei Kepuasan	126
Tabel 5.29 Hasil Kuisioner Kualitas	127
Tabel 5.30 Hasil Perhitungan Apriori Manual	134

[Halaman ini sengaja dikosongkan]

DAFTAR KODE SEMU

Kode Semu 4.1 Melihat Informasi	72
Kode Semu 4.2 Menambah data informasi	76
Kode Semu 4.3 Melihat detail data informasi	77
Kode Semu 4.4 Mengubah data informasi	78
Kode Semu 4.5 Menghapus data informasi.....	78
Kode Semu 4.6 Melakukan pemesanan dan konfirmasi pembelian	84
Kode Semu 4.7 Melakukan pembayaran.....	87
Kode Semu 4.8 Mengirim pertanyaan.....	91
Kode Semu 4.9 Menambah pengguna sebagai Pembeli.....	93
Kode Semu 4.10 Perhitungan Apriori	98
Kode Semu A.1 Melihat produk.....	139
Kode Semu A.2 Menambah data produk	139
Kode Semu A.3 Melihat detail data produk	139
Kode Semu A.4 Menghapus data produk.....	139
Kode Semu A.5 Mengubah data produk	140
Kode Semu A.6 Melihat riwayat transaksi.....	140
Kode Semu A.7 Melihat detail pemesanan	140
Kode Semu A.8 Menghapus/Membatalkan pembelian	140
Kode Semu A.9 Melihat detail transaksi	141
Kode Semu A.10 Mengunggah resi pengiriman	141
Kode Semu A.11 Menghapus data transaksi.....	141
Kode Semu A.12 Melihat forum tanya jawab.....	141
Kode Semu A.13 Menjawab pertanyaan.....	141
Kode Semu A.14 Menambah pengguna sebagai Administrator/ <i>Reseller</i>	142
Kode Semu A.15 Melihat profil	142

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Pada bab ini akan dipaparkan mengenai garis besar Tugas Akhir yang meliputi latar belakang, tujuan, rumusan dan batasan permasalahan, metodologi pembuatan Tugas Akhir, dan sistematika penulisan.

1.1. Latar Belakang

Perkembangan teknologi informasi yang semakin pesat membuat pekerjaan menjadi lebih mudah. Banyak pula sistem yang bersifat aplikatif dan *real time* dibangun sehingga memudahkan setiap orang untuk mengakses informasi dimanapun dan kapanpun. Layanan internet telah dimanfaatkan para pengguna sebagai media pertukaran data bahkan dimanfaatkan sebagai sarana bisnis, salah satunya adalah perdagangan elektronik. Perdagangan elektronik atau yang biasa disebut *e-commerce* adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti internet atau televisi, *www*, atau jaringan komputer lainnya [1]. Dengan begitu generasi milenial Indonesia dapat menjadi aset penggerak ekonomi dimasa depan, dan mampu mengubah dan memengaruhi masyarakat. Generasi ini yang berusia muda, bergerak cepat, dan sangat mudah beradaptasi dengan perkembangan teknologi informasi yang menjadi faktor utama pada perubahan pola belanja masyarakat. Selain itu, dengan pertumbuhan ekonomi Indonesia yang saat ini semakin dinamis, sehingga perkembangan *e-commerce* atau perdagangan elektronik pun justru semakin pesat melesat [2].

Berdasarkan hal tersebut persaingan pada perdagangan dalam pasar *online* menjadi semakin sengit dan ketat. Banyak pedagang yang memulai memasarkan dagangan mereka melalui situs *marketplace* yang sudah ada seperti Tokopedia, Bukalapak, atau sosial media seperti Facebook dan Instagram, ada pula yang

membangun *website* sendiri dengan *template Content Management Sistem (CMS)* yang telah ada untuk penjualan dagangan mereka. Namun, ada saatnya dimana pedagang harus pindah dari sana dan mulai membangun *website* sendiri. Karena tidak semua *template* yang ada mampu mumpuni semua fitur yang diinginkan pedagang, atau pengembangan yang dilakukan nantinya tidak dapat leluasa karena adanya batasan-batasan di dalam *template*.

Dengan membangun aplikasi *website* sendiri, pedagang tentu akan mendapatkan beberapa keuntungan lain daripada hanya “menumpang” dari situs lain. Keuntungan yang didapatkan pedagang diantaranya adalah pedagang dapat memiliki kontrol yang lebih, dapat terbebas dari “perang harga”, membantu pembeli untuk bisa sepenuhnya fokus pada produk yang dijual, membangun *brand*, dan beberapa keuntungan lain [3]. Ketika pedagang membuat *website* sendiri, mereka juga dapat menambahkan fitur rekomendasi barang dagangan lain yang berhubungan kepada pembeli. Rekomendasi barang yang berhubungan dapat berdasarkan kesamaan jenis makanan atau yang lain. Dalam merekomendasikan barang dapat digunakan metode algoritma apriori pada *website* yang akan dibangun. Algoritma apriori merupakan variasi dari algoritma *association rules*, digunakan untuk menemukan sekelompok *item (itemsets)* yang berhubungan dengan *item* lain dari dalam *database* transaksi untuk menemukan aturan asosiatif antara suatu kombinasi *item*, dan dapat diketahui dengan menggunakan dua parameter, yaitu *support* (nilai penunjang dalam persentase dari kombinasi *item* tersebut dalam *database*) dan *confidence* (nilai kepastian, kuatnya hubungan antar *item* dalam aturan asosiasi) [4].

Aplikasi industri rumah tangga pada penjualan makanan ringan ini bertujuan untuk membantu memudahkan penjual untuk memasarkan produk mereka kepada pembeli, sehingga dapat meraih pembeli lebih mudah. Aplikasi ini akan digunakan oleh salah satu penjual yang telah memiliki label nama sendiri yaitu

Mewek. Label Mewek sendiri telah menjual berbagai macam makanan dan minuman ringan, namun menu andalannya adalah Makaroni Mewek. Pada jenis Makaroni Mewek sendiri memiliki berbagai macam pilihan rasa, diantaranya ada Original dengan tiga level, Balado, Keju, BBQ, Jagung Manis, dan lain-lain. Kemudian ada pula Bakso Tengiri Goreng (Basreng), Usus Goreng, Seblak, dan berbagai jenis lain mengikuti musim. Untuk penjualan minuman ringan ada Coklatini, Monggo Mango, Thai Tea, dan varian rasa lain. Pada aplikasi ini nantinya akan digunakan oleh empat macam pengguna, yang pertama adalah Administrator yang bertugas mengolah data penjualan, yang menjadi Administrator merupakan pemilik ataupun karyawan yang ditugaskan untuk mengelola aplikasi ini. Pengguna kedua adalah Pengunjung biasa yang tidak mempunyai akun pada aplikasi, pengguna ini hanya dapat melihat produk yang dijual, melihat beberapa menu lain dan melihat daftar pertanyaan serta dapat mengirim pertanyaan. Kemudian untuk pengguna yang ketiga dan keempat adalah Pembeli dan *Reseller* adalah pengguna yang terdaftar yang mana selain dapat melakukan hal yang sama dengan pengunjung biasa, pengguna ini juga dapat melakukan transaksi pembelian melalui aplikasi ini, namun khusus untuk *reseller* akan mendapatkan harga yang berbeda karena batas minimal barang yang dibeli pun berbeda.

1.2. Tujuan

Tujuan dari pembuatan tugas akhir ini adalah:

1. Menghasilkan sebuah aplikasi berbasis *web* untuk industri rumah tangga dalam penjualan makanan ringan yang dapat mengakomodasi kebutuhan penjual dalam menjual produk.
2. Menghasilkan tampilan riwayat barang yang saling berelasi dari data riwayat pembelian berdasarkan aturan Asosiasi pada halaman Administrator.

1.3. Rumusan Permasalahan

Berdasarkan latar belakang yang telah dibahas, adapun beberapa rumusan masalah yang dapat diangkat dalam tugas akhir ini, antara lain:

1. Bagaimana membuat *web* aplikasi industri rumah tangga yang dapat mengakomodasi kebutuhan penjual dalam menjual produk kepada pembeli dan *reseller*?
2. Bagaimana menampilkan riwayat pembelian produk yang berelasi dengan produk lainnya berdasarkan aturan Asosiasi pembeli menggunakan metode algoritma Apriori?

1.4. Batasan Permasalahan

Batasan masalah yang ditetapkan pada Tugas Akhir ini adalah sebagai berikut:

1. Aplikasi penjualan makanan ringan berbasis *web* dengan Framework Codeigniter, Bahasa PHP dan *database* MySQL.
2. Lingkup kerja sistem adalah mengolah data produk, data pembeli, dan data transaksi, data produk untuk *reseller*.
3. Menampilkan relasi produk dari riwayat pembelian produk menggunakan metode algoritma Apriori.

1.5. Metodologi

Adapun beberapa tahap dalam proses pengerjaan tugas akhir ini, yaitu sebagai berikut:

1. Studi Literatur

Studi literatur yang dilakukan dalam pengerjaan ini adalah mengenai algoritma Apriori dan perdagangan elektronik yang diperlukan untuk menyusun algoritma yang baik. Materi dapat diambil dari contoh tugas akhir, internet, buku, jurnal, maupun materi kuliah yang

berhubungan dengan metode algoritma Apriori. Sehingga, studi literatur ini dapat diterapkan pada perancangan aplikasi industri rumah tangga.

2. Analisis dan desain perangkat lunak

Analisis kebutuhan dan perancangan sistem dilakukan untuk merumuskan solusi yang tepat dalam pembuatan aplikasi serta kemungkinan yang dapat dilakukan dalam mengimplementasikan rancangan tersebut. Tahap desain meliputi arsitektur perangkat lunak yang akan digunakan, desain penggunaan yang terlibat dalam aplikasi, desain antarmuka, serta diagram-diagram pendukung untuk mendeskripsikan sistem aplikasi.

3. Implementasi

Pembangunan aplikasi industri rumah tangga ini akan dilakukan dengan menggunakan bahasa pemrograman PHP, HTML, CSS, Javascript, JQuery, dan Ajax. Desain pada aplikasi ini juga menggunakan *tools* PHPStorm sebagai *editor tools*, *framework* Codeigniter, basis data MySQL, dan XAMPP [5].

4. Uji coba dan evaluasi

Pengujian akan dilakukan oleh satu pengguna yang mana adalah salah satu karyawan dari perusahaan yang akan menjadi Administrator dari aplikasi. Pengguna tersebut akan menggunakan aplikasi sesuai dengan data perusahaan. Pengguna akan menggunakan sesuai dengan skenario yang telah ditetapkan. Diharapkan aplikasi ini dapat membantu dalam manajemen kegiatan perusahaan.

5. Penyusunan buku tugas akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematisa penulisan

buku tugas akhir secara garis besar antara lain:

1. Pendahuluan
 - a. Latar Belakang
 - b. Tujuan
 - c. Rumusan Masalah
 - d. Batasan Tugas Akhir
 - e. Metodologi
 - f. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

1.6. Sistematika Penulisan

Buku Tugas Akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini.

Bab I Pendahuluan

Bab yang berisi latar belakang masalah, tujuan, rumusan masalah, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan laporan akhir juga merupakan bagian dari bab ini.

Bab II Analisis dan Perancangan Sistem

Bab ini berisi tentang analisis permasalahan, deskripsi umum sistem, batasan sistem, kebutuhan fungsional, interaksi sistem dengan pengguna dan rancangan antarmuka pengguna.

Bab III Implementasi

Bab ini berisi implementasi dari perancangan dan implementasi fitur-fitur penunjang aplikasi

termasuk implementasi sistem dan implementasi antarmuka pengguna.

Bab IV Pengujian dan Evaluasi

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

Bab V Kesimpulan dan Saran

Bab ini menjelaskan kemampuan perangkat lunak dengan melakukan pengujian kebenaran dan pengujian kinerja dari sistem yang telah dibuat.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan Tugas Akhir.

Lampiran

Merupakan bab tambahan yang berisi daftar kode yang ada pada aplikasi ini.

[Halaman ini sengaja dikosongkan]

BAB II

TINJAUAN PUSTAKA

Pada Bab ini akan dibahas mengenai teori-teori yang menjadi dasar dari pembuatan Tugas Akhir. Teori-teori tersebut meliputi pengertian dan beberapa analisis terkait pembuatan *web* aplikasi untuk industri rumah tangga yang dapat mengakomodasi kebutuhan penjual dalam menjual produk kepada pembeli dan *reseller*.

2.1. Perdagangan Elektronik

Perdagangan elektronik atau yang biasa disebut *e-commerce* adalah suatu cara untuk memasarkan barang dan/atau jasa yang dimiliki seseorang atau kelompok melalui media-media elektronik. Media elektronik dapat berupa melalui internet, televisi, atau jaringan yang terhubung dengan komputer lainnya. Media yang saat ini sedang berkembang dengan pesat yaitu melalui internet, dengan menggunakan aplikasi *web*.

Saat ini banyak toko konvensional yang sudah memulai membuat toko *online* mereka sendiri, yang mana barang atau jasa yang mereka jual sama dengan toko konvensional, hanya saja dipasarkan dengan media elektronik. Media ini dikatakan berkembang sangat pesat karena banyak konsumen yang saat ini lebih memilih menggunakan media ini daripada mendatangi langsung toko yang ada.

Biasanya para pedagang memilih menjual barang dagangan mereka melalui *marketplace* yang sudah ada, namun tak sedikit pula yang memilih memulai membuat *website* mereka sendiri karena beberapa keuntungan yang tidak didapat jika melalui *marketplace* tetapi bisa mereka dapatkan jika membangun *website* sendiri.

Pada Tabel 2.1 menunjukkan beberapa perbandingan dari *marketplace* dengan *website* yang dibangun sendiri berdasarkan beberapa kategori [6]:

Tabel 2.1 Perbandingan *marketplace* dengan *website* milik sendiri

No	Kategori	Marketplace	<i>Website</i> milik sendiri
1.	Investasi	Harus mengeluarkan biaya sewa, servis dan jasa.	Tidak ada biaya sewa, servis dan jasa.
2.	Persaingan Merk	Bersaing dengan banyak merk yang menjual produk yang sama.	Hanya ada satu merk dalam satu <i>website</i> .
3.	Lingkup penjualan	Lingkup penjualan luas, skala nasional.	Pembeli yang masih lebih banyak memilih membeli toko yang memiliki lokasi di kota yang sama.
4.	Perolehan pelanggan	Persaingan dengan merk lain dalam perolehan pelanggan.	Dapat lebih fokus mendapatkan pelanggan.
5.	Pemasaran merk	Tidak adanya pemasaran pada satu merk khusus.	Dapat fokus memasarkan merk dan produk.
6.	Alur pembelian	Tidak langsung ke penjual.	Langsung ditangani oleh penjual.

2.2. Industri Rumah Tangga

Industri yaitu kegiatan yang dilakukan manusia yang mengolah bahan mentah menjadi barang jadi atau bahan setengah jadi untuk menghasilkan barang yang lebih tinggi nilainya. Industri rumah tangga merupakan salah satu upaya untuk meningkatkan mutu sumber daya manusia dan kemampuan untuk memanfaatkan

sumber daya alam secara optimal [7]. Saat ini beberapa industri rumah tangga telah memiliki *website*-nya sendiri. Namun masing-masing *website* tersebut memiliki fitur yang berbeda-beda, dan memiliki kelebihan dan kekurangan. Beberapa *website* dari industry yang telah ada memiliki beberapa perbedaan dengan *website* pada tugas akhir ini, yang dapat dilihat pada Tabel 2.2.

Tabel 2.2 Perbandingan *Website* Industri Lain dengan Milik Sendiri

No	Fitur	Lapis Kukus Pahlawan Surabaya	Lazizaa	Hisana Fried Chicken	Amanda Brownies	Makaroni Mewek
1.	Menampilkan informasi dan produk	√	√	√	√	√
2.	Mengirim pertanyaan/ menghubungi admin	√	√	√	√	√
3.	Menampilkan lokasi toko	√	√	√	√	√
4.	Hyperlink ke Sosial Media	√	√	√	√	√
5.	Menampilkan rekomendasi produk	×	×	×	×	√
6.	Melakukan pembelian	×	√	×	×	√

2.3. Apriori

Algoritma Apriori merupakan salah satu turunan dari algoritma *association rules*. *Association rules* adalah teknik *data mining* untuk menemukan aturan asosiatif antara suatu kombinasi *item*, yang bertujuan untuk mencari sekelompok *item* (*itemsets*) yang memiliki nilai *support* yang lebih besar daripada minimum *support*. *Support* dalam *association rules* adalah kombinasi *item* tersebut dalam *database*. Nilai *support* ini ditentukan dan menggunakan sekelompok *itemset* yang memiliki nilai *confidence* yang lebih besar daripada nilai minimum *confidence* yang ditetapkan. Tujuannya adalah untuk menentukan rasio dari nilai

support item yang dicari terhadap nilai *item* lainnya pada saat tidak saling tergantung satu sama lain [8].

Association Rule menjadi terkenal karena aplikasinya untuk menganalisis isi keranjang belanja pelanggan di pasar swalayan. Oleh karena itu, sering disebut *Market Basket Analysis*. *Association Rule* juga dikenal salah satu teknik *data mining* yang menjadi dasar dari berbagai teknik *data mining* lainnya. Salah satu tahap analisis asosiasi yang menarik perhatian banyak peneliti untuk menghasilkan algoritma yang efisien adalah analisis pola frekuensi tinggi (*frequent pattern mining*). Penting tidaknya suatu aturan asosiatif dapat diketahui dengan dua parameter [9]:

- *Support* (nilai penunjang) = Persentase kombinasi *item* tersebut dalam *database*, dengan rumus sebagai berikut:

$$\text{Support}(X) = \frac{n(X)}{n} \quad (2.1)$$

Dimana n adalah jumlah total transaksi dalam basis data, sedangkan $n(X)$ adalah jumlah transaksi yang mengandung *itemset* X , atau jumlah *item* yang terdapat dalam transaksi.

- *Confidence* (nilai kepastian) = Kuatnya hubungan antar *item* dalam aturan asosiasi.

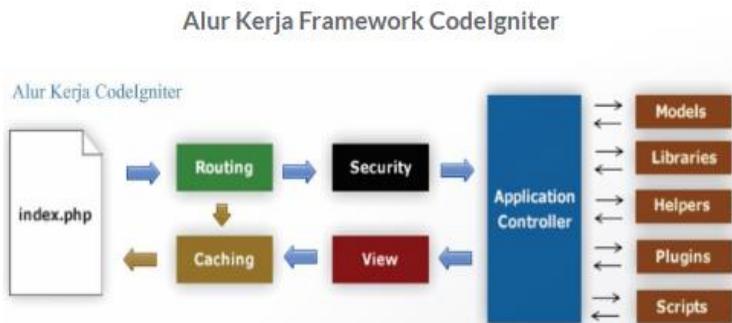
2.4. Codeigniter

Codeigniter merupakan aplikasi sumber terbuka yang berupa *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP, sehingga memudahkan *developer* untuk membuat aplikasi *web* dengan cepat mudah dibandingkan dengan membuatnya dari awal. MVC adalah sebuah logika aplikasi yang berdasarkan komponen-komponen, seperti: manipulasi data (*model*), *controller*, dan *user interface* [10].

1. Model, mewakili struktur data, yang berfungsi untuk membantu pengolahan basis data, sehingga seperti sesuatu yang dapat menghubungkan dengan *database* sistem untuk memanipulasi data (*insert, update, delete, search*), menangani

- validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan *view*.
2. *View*, bagian yang mengatur tampilan untuk pengguna (*user interface*). Menghasilkan halaman-halaman yang biasa dilihat oleh pengguna, yaitu halaman *web*. Menangani *presentation logic*. Pada suatu aplikasi *web* bagian ini biasanya berupa *file template HTML*, yang diatur oleh *controller*. Bagian ini tidak memiliki akses langsung terhadap bagian *model*.
 3. *Controller*, bagian yang menghubungkan antara *model* dan *view*. Berfungsi untuk menerima *request* dan data dari pengguna kemudian menentukan apa yang akan diproses oleh aplikasi.

Alur kerja framework Codeigniter dapat dilihat pada Gambar 2.1 [11].



Gambar 2.1 Alur Kerja Framework Codeigniter

- **Index.php:** Index.php disini berfungsi sebagai *file* pertama dalam program yang akan dibaca oleh program.
- **The Router:** *Router* akan memeriksa HTTP *request* untuk menentukan hal apa yang harus dilakukan oleh program.
- **Cache File:** Apabila dalam program sudah terdapat “*cache file*” maka *file* tersebut akan langsung dikirim ke *browser*. *File cache* inilah yang dapat membuat sebuah *website* dapat dibuka

dengan lebih cepat. *Cache file* dapat melewati proses yang sebenarnya harus dilakukan oleh program *codeigniter*.

- **Security:** Sebelum *file controller* dimuat keseluruhan, HTTP *request* dan data yang disubmit oleh pengguna akan disaring terlebih dahulu melalui fasilitas *security* yang dimiliki oleh *codeigniter*.
- **View:** Hal yang terakhir akan dilakukan adalah membaca semua program yang ada dalam *view file* dan mengirimkannya ke browser supaya dapat dilihat. Apabila *file view* sudah ada yang di “cache” maka *file view* baru yang belum ter-cache akan memperbarui *file view* yang sudah ada.
- **Controller:** *Controller* akan membuka *file model*, *core libraries*, *helper* dan semua *resources* yang dibutuhkan dalam program tersebut.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada Bab ini akan dijelaskan mengenai analisis kebutuhan dan perancangan perangkat lunak untuk mencapai tujuan dari tugas akhir.

3.1. Analisis

Tahap analisis dibagi menjadi beberapa bagian, antara lain analisis permasalahan, deskripsi umum sistem, dan spesifikasi kebutuhan perangkat lunak.

3.1.1. Analisis Permasalahan

Permasalahan utama yang diangkat pada pembuatan Tugas Akhir ini adalah bagaimana membangun aplikasi industri rumah tangga untuk makanan ringan dengan basis *web* menggunakan metode algoritma Apriori.

Generasi muda saat ini, senang bergerak cepat, dan sangat mudah beradaptasi dengan perkembangan teknologi informasi yang menjadi faktor utama pada perubahan pola belanja masyarakat. Berdasarkan hal tersebut persaingan pada perdagangan dalam pasar *online* menjadi semakin sengit dan ketat. Banyak pedagang yang mulai memasarkan dagangan mereka melalui situs *marketplace* yang sudah ada seperti Tokopedia, Bukalapak, atau sosial media seperti Facebook dan Instagram, ada pula yang membangun *website* sendiri dengan *template Content Management Sistem* (CMS) yang telah ada untuk penjualan dagangan mereka. Namun, semua yang telah ada tersebut hanya *template* bawaan, dan tidak semua *template* yang ada mampu mumpuni semua fitur yang diinginkan pedagang, atau pengembangan yang dilakukan nantinya tidak dapat leluasa karena adanya batasan-batasan di dalam *template*.

Untuk mengatasi permasalahan tersebut, penulis akan mengimplementasikan pembuatan aplikasi untuk industri rumah tangga pada makanan ringan “Mewek” dengan basis *web*. Namun,

ketika industri rumah tangga membuat *web* sendiri pun, lingkup pasarnya masih kecil, sehingga dibutuhkan usaha ekstra dalam memasarkan dagangan atau menarik pembeli. Jika dalam *website* yang dibuat tidak ada sesuatu yang dapat menarik pembeli maka akan percuma. Oleh karena itu, selain membuat *website* sendiri untuk “Mewek”, penulis juga akan membuat fitur tambahan berupa rekomendasi produk dengan perhitungan algoritma Apriori. Algoritma Apriori cukup terkenal untuk digunakan dalam perhitungan analisis keranjang pasar dalam transaksi penjualan. Analisis transaksi penjualan bertujuan untuk merancang strategi penjualan atau pemasaran yang efektif dengan memanfaatkan data transaksi penjualan yang ada. Selain itu, penggunaan algoritma Apriori ini juga dapat menemukan pola berupa produk-produk yang sering dibeli bersamaan atau produk yang cenderung muncul bersama dalam sebuah transaksi dari data transaksi yang pada umumnya berukuran besar. Perusahaan lalu dapat menggunakan pola ini untuk menempatkan produk yang sering dibeli ke dalam sebuah area yang berdekatan, merancang tampilan produk di katalog, merancang kupon diskon bagi pelanggan yang membeli produk tertentu, merancang penjualan paket produk, dan sebagainya [12].

3.1.2. Deskripsi Umum Sistem

Pada Tugas Akhir ini akan dibuat sebuah perangkat lunak berupa aplikasi berbasis *web* untuk industri rumah tangga pada makanan ringan “Mewek” dengan menggunakan algoritma Apriori. Desain aplikasi penjualan makanan ringan ini dapat digambarkan suatu aplikasi semacam *e-commerce* khusus untuk suatu merk dari produk makanan ringan, atau bisa juga minuman ringan. Aplikasi ini akan berbasis *web* namun dinamis dan responsif, sehingga pembeli dapat menggunakan aplikasi ini baik pada komputer maupun *mobile*. Mirip seperti aplikasi *web* HIJup, yang mana mereka memasarkan produk mereka dengan merk mereka sendiri dalam sebuah aplikasi *web* dan menjualnya secara *online* yang dapat disebut juga toko *online* (*online shop*), sehingga

pembeli tidak perlu datang ke toko mereka untuk memilih dan membeli barang. Pada aplikasi penjualan makanan ringan ini juga memakai konsep yang kurang lebih sama dengan Hijup, yaitu membangun sebuah *online shop*, dengan menampilkan informasi promo, produk, harga, melakukan transaksi, *booking product*, dan lain-lain. Dikarenakan *website* ini untuk penjualan makanan ringan, seperti makaroni, maka pembeli nantinya akan mendapat kebebasan memilih makaroni dengan rasa, dan jumlah yang mereka inginkan.

Pada halaman beranda *website* juga akan menampilkan rekomendasi produk yang banyak dibeli oleh pembeli, sehingga dapat memudahkan pembeli ketika memilih produk tanpa harus mencari produk mana yang menarik untuk dibeli, yang mana dalam mencari produk *best seller*. Pada data transaksi pasti akan bertambah terus menerus sehingga akan menghasilkan hasil yang berbeda secara berkala, algoritma Apriori cukup bagus dalam melakukan perhitungan yang berubah-ubah seperti transaksi penjualan, dan algoritma Apriori memang cukup terkenal bagus untuk digunakan dalam perhitungan analisis keranjang pasar. Dengan menggunakan algoritma Apriori, rekomendasi produk yang disarankan dapat berubah-ubah secara berkala, sehingga penjual dapat lebih mudah menentukan produk mana yang akan diproduksi lebih banyak agar dapat tetap memuaskan pembeli tanpa harus menolak pembeli karena kehabisan stok barang. Penjual juga dapat membuat atau memutuskan strategi yang bagus ketika memasarkan dagangan mereka, atau ketika ingin menarik pembeli dengan tawaran-tawaran promo dan semacamnya.

3.1.3. Spesifikasi Kebutuhan Perangkat Lunak

Subbab ini membahas spesifikasi kebutuhan perangkat lunak dari hasil analisis yang telah dilakukan. Bagian ini berisi kebutuhan perangkat lunak yang direpresentasikan dalam bentuk kebutuhan fungsional, kebutuhan non-fungsional, diagram kasus penggunaan, dan diagram aktivitas.

3.1.3.1. Aktor

Pengertian aktor atau pengguna adalah pihak-pihak, baik manusia maupun sistem atau perangkat lain yang terlibat dan berinteraksi secara langsung dengan sistem. Aktor pada sistem ini akan dijelaskan pada Tabel 3.1.

Tabel 3.1 Aktor pada Sistem

Aktor	Tugas	Hak Akses ke aplikasi
Administrator (pemilik, karyawan)	Mengelola data	<ul style="list-style-type: none"> ● Mengelola (melihat detail, menambah, mengubah, menghapus) semua data informasi, produk, dan transaksi ● Melihat pertanyaan ● Menjawab pertanyaan ● Mendaftarkan Administrator dan/atau <i>Reseller</i> baru ● Melihat data produk yang diminati pembeli maupun <i>reseller</i> dengan aturan Asosiasi algoritma Apriori
Pengunjung (pengguna tanpa login)	Melihat data dan mendaftar sebagai Pembeli atau <i>Reseller</i>	<ul style="list-style-type: none"> ● Melihat semua data ● Mengirim pertanyaan ● Mendaftar sebagai Pembeli atau <i>Reseller</i> baru ● Melihat rekomendasi produk
Pembeli dan <i>Reseller</i> (pengguna yang terdaftar)	Melihat data, melakukan transaksi, mendaftar sebagai Pembeli atau <i>Reseller</i>	<ul style="list-style-type: none"> ● Melihat semua data ● Mengirim pertanyaan ● Melakukan transaksi ● Melihat rekomendasi produk

3.1.3.2. Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan layanan yang harus dimiliki oleh perangkat lunak, reaksi terhadap suatu masukan, hasil yang dilakukan pada situasi khusus. Kebutuhan fungsional dari perangkat lunak untuk aplikasi ini dijelaskan pada Tabel 3.2.

Tabel 3.2 Kebutuhan Fungsional pada Sistem

Kode	Kebutuhan Fungsional	Deskripsi
F01	Menangani kelola data informasi	Menampilkan informasi mengenai data yang ada, melakukan penambahan, pengeditan, serta penghapusan data informasi.
F02	Menangani kelola data produk	Menampilkan data produk yang ada, melakukan penambahan, pengeditan, serta penghapusan data produk.
F03	Menangani kelola transaksi	Menampilkan transaksi yang akan dilakukan, menghitung total harga untuk pembayaran, menyimpan data transaksi baru. Mengunggah gambar bukti resi pembayaran. Mengunggah gambar bukti pengiriman.
F04	Menambah data pengguna	Menambah pengguna baru.
F05	Menangani kelola data forum tanya jawab	Menampilkan daftar pertanyaan pada forum tanya jawab, mengirim pertanyaan, serta mengirim jawaban dari suatu pertanyaan.
F06	Melihat data riwayat pembelian produk	Menampilkan produk yang diminati pembeli berdasarkan riwayat pembelian produk dari pembeli maupun <i>reseller</i> dengan aturan Asosiasi.

3.1.3.3. Kebutuhan Non-Fungsional

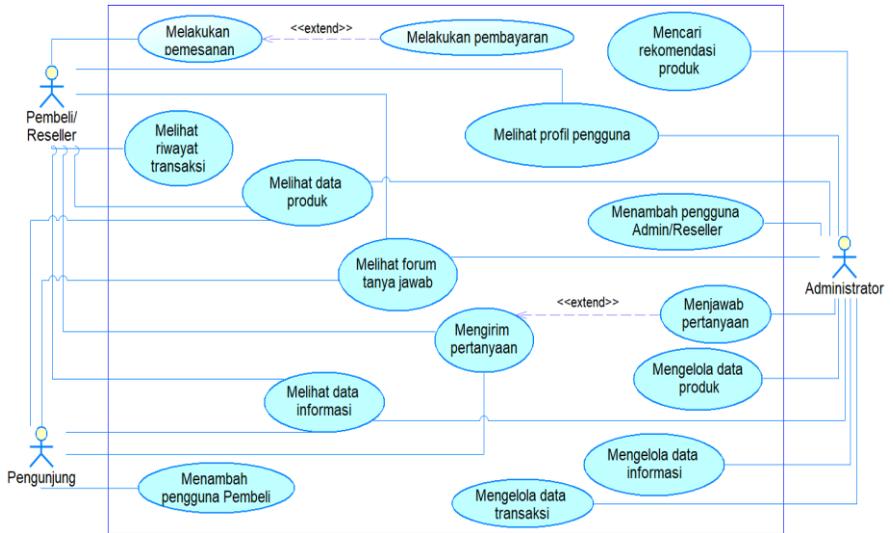
Kebutuhan non-fungsional mendefinisikan batasan layanan yang ditawarkan oleh sistem. Kebutuhan non-fungsional yang harus dipenuhi oleh sistem dapat dilihat pada Tabel 3.3.

Tabel 3.3 Kebutuhan Non-Fungsional pada Sistem

No	Parameter	Deskripsi
1	<i>Operational</i>	Aplikasi dapat memberikan <i>response</i> yang cepat dan tepat dari sistem
2	<i>Usability</i>	Aplikasi dibangun dengan antarmuka pengguna yang konsisten, mudah dipahami dan mudah dioperasikan, dapat dioperasikan kapanpun.
3	<i>Portability</i>	Aplikasi mudah untuk dioperasikan pada semua perangkat digital dengan mengakses <i>website</i> .
4	Bahasa	Bahasa yang digunakan pada antarmuka merupakan bahasa Indonesia.

3.1.3.4. Diagram Kasus Penggunaan

Kasus penggunaan perangkat lunak dijelaskan secara rinci dalam subbab ini. Kasus penggunaan didasarkan pada hasil analisis kebutuhan fungsional dari perangkat lunak. Kasus penggunaan pada aplikasi ini dapat dilihat pada Gambar 3.1.



Gambar 3.1 Diagram Kasus Penggunaan

Penjelasan lengkap mengenai kasus penggunaan beserta aktor yang melakukan dapat dilihat pada Tabel 3.4.

Tabel 3.4 Deskripsi Kasus Penggunaan Sistem

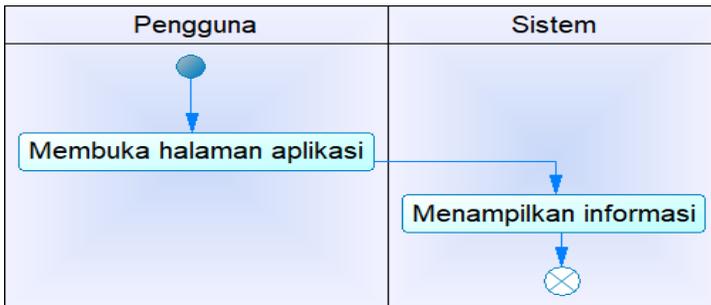
Kode Use Case	Nama Use Case	Aktor
UC001	Melihat informasi	Pengunjung, Pembeli, <i>Reseller</i> , Administrator
UC002	Mengelola data informasi	Administrator
UC003	Melihat produk	Pengunjung, Pembeli, <i>Reseller</i> , Administrator
UC004	Mengelola data produk	Administrator
UC005	Melakukan pemesanan	Pembeli, <i>Reseller</i>
UC006	Melihat riwayat transaksi	Pembeli, <i>Reseller</i>
UC007	Melihat detail pemesanan	Pembeli, <i>Reseller</i> , Administrator
UC008	Menghapus pemesanan	Pembeli, <i>Reseller</i> , Administrator
UC009	Melakukan pembayaran	Pembeli, <i>Reseller</i>
UC010	Mengelola data transaksi	Administrator
UC011	Melihat forum tanya jawab	Pengunjung, Pembeli, <i>Reseller</i> , Administrator
UC012	Mengirim pertanyaan	Pengunjung, Pembeli, <i>Reseller</i>
UC013	Menjawab pertanyaan	Administrator
UC014	Menambah pengguna Pembeli	Pengunjung
UC015	Menambah pengguna Administrator/ <i>Reseller</i>	Administrator
UC016	Melihat profil pengguna	Administrator, Pembeli, <i>Reseller</i>
UC017	Mencari riwayat pembelian	Administrator

3.1.3.4.1 Kasus Penggunaan Melihat Informasi

Kasus penggunaan ini diakses ketika pengguna membuka aplikasi. Penjelasan skenario dapat dilihat pada Tabel 3.5.

Tabel 3.5 Rincian Alur Kasus Penggunaan Melihat Informasi

Nama Use Case	Melihat data informasi
Nomor	UC001
Aktor	Pengunjung, Pembeli, <i>Reseller</i>
Deskripsi	Kejadian dimana pengguna dapat melihat data informasi
Trigger	Pengguna membuka halaman <i>web</i> aplikasi
Kondisi Awal	Data informasi telah tersimpan pada <i>database</i>
Kondisi Akhir	Sistem menampilkan data informasi yang ada
Alur Normal	1. Pengguna membuka halaman aplikasi 2. Sistem menampilkan informasi yang ada
Alur Alternatif	

**Gambar 3.2 Diagram Aktivitas UC001**

Skenario aktivitas dari kasus penggunaan selanjutnya digambarkan ke dalam diagram aktivitas yang dapat dilihat pada Gambar 3.2.

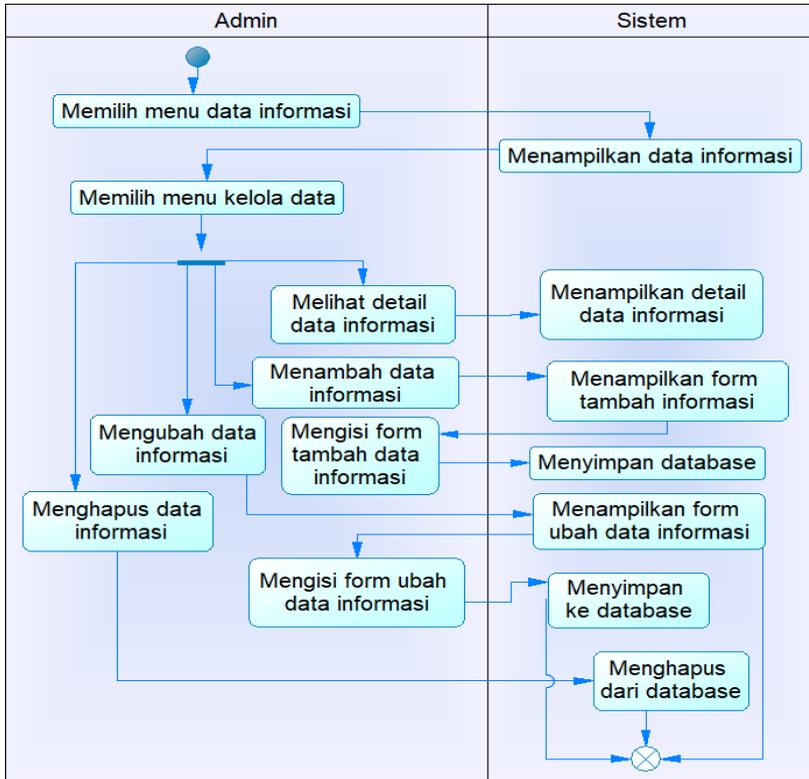
3.1.3.4.2 Kasus Penggunaan Mengelola Data Informasi

Kasus penggunaan mengelola data informasi ini diakses ketika Administrator membuka menu data informasi. Administrator dapat melihat detail, menambah, mengubah, dan menghapus data informasi pada *database*. Penjelasan skenario mengelola data informasi yang lebih rinci dapat dilihat pada Tabel 3.6.

Tabel 3.6 Rincian Alur Kasus Penggunaan Mengelola Data Informasi

Nama Use Case	Mengelola Data Informasi
Nomor	UC002
Aktor	Administrator
Deskripsi	Kejadian dimana Administrator dapat mengelola
Trigger	Administrator membuka menu data informasi
Kondisi Awal	Administrator belum mengelola data informasi
Kondisi Akhir	Administrator mengelola data informasi
Alur Normal	<ol style="list-style-type: none"> 1. Administrator membuka menu data informasi 2. Sistem menampilkan halaman data informasi beserta tombol detail, tambah, edit, dan hapus.
Alur Alternatif	<ol style="list-style-type: none"> 2.A. Administrator memilih melihat detail informasi <ol style="list-style-type: none"> 2.A.1. Administrator memilih tombol detail data informasi 2.A.2. Sistem menampilkan detail data informasi 2.B. Administrator memilih menambah data informasi <ol style="list-style-type: none"> 2.B.1. Administrator memilih tombol tambah data informasi 2.B.2. Sistem menampilkan form untuk menambah data informasi 2.B.3. Administrator mengisi form data informasi baru 2.B.4. Sistem menyimpan ke <i>database</i>. 2.C. Administrator memilih ubah data informasi <ol style="list-style-type: none"> 2.C.1. Administrator memilih tombol ubah data informasi 2.C.2. Sistem menampilkan form ubah data 2.C.3. Administrator mengisi data informasi yang ingin diubah 2.C.4. Sistem menyimpan ke <i>database</i> 2.D. Administrator memilih menghapus data informasi <ol style="list-style-type: none"> 2.D.1. Administrator memilih tombol hapus data informasi 2.D.2. Sistem menghapus data informasi

Berdasarkan skenario kasus penggunaan pada Tabel 3.6 selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mengelola data informasi dapat dilihat pada Gambar 3.3.



Gambar 3.3 Diagram Aktivitas UC002

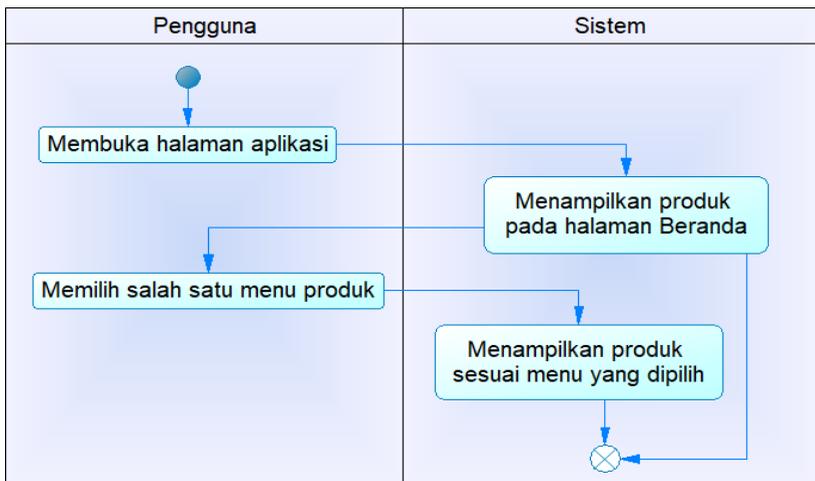
3.1.3.4.3 Kasus Penggunaan Melihat Produk

Pada kasus penggunaan melihat produk ini diakses ketika pengunjung atau pembeli membuka halaman *web*, maka produk-produk yang dijual akan ditampilkan. Data produk akan ditampilkan dalam bentuk gambar dari macam-macam produk. Penjelasan skenario yang dapat dilihat pada Tabel 3.7.

Tabel 3.7 Rincian Alur Kasus Penggunaan Melihat Produk

Nama Use Case	Melihat Produk
Nomor	UC003
Aktor	Pengunjung, Pembeli, <i>Reseller</i>
Deskripsi	Semua pengguna dapat melihat data produk yang ada
Trigger	Semua pengguna membuka menu produk
Kondisi Awal	Data produk telah tersimpan pada <i>database</i>
Kondisi Akhir	Data produk berhasil ditampilkan
Alur Normal	<ol style="list-style-type: none"> Pengunjung atau Pembeli memilih menu untuk melihat produk pada halaman <i>web</i> toko online “Mewek” Sistem menampilkan semua produk yang ada
Alur Alternatif	

Berdasarkan dari skenario kasus penggunaan melihat produk, selanjutnya digambarkan kedalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan melihat produk dapat dilihat pada Gambar 3.4.

**Gambar 3.4 Diagram Aktivitas UC003**

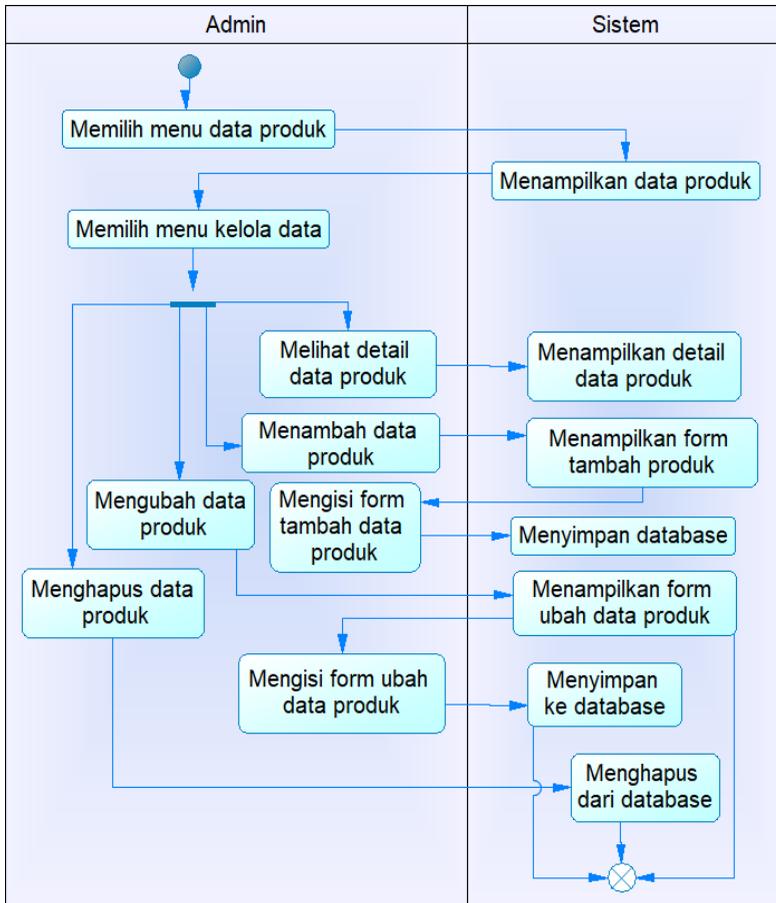
3.1.3.4.4 Kasus Penggunaan Mengelola Data Produk

Pada kasus penggunaan mengelola data produk ini diakses ketika Administrator hendak melakukan pengelolaan pada data produk yang ada, pengelolaan data dapat berupa menampilkan detail dari data, penambahan, perubahan, dan penghapusan data. Penjelasan skenario yang lebih rinci akan dijelaskan pada Tabel 3.8. Berdasarkan skenario yang telah dijelaskan, maka selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas untuk menggambarkan alur lebih jelas mengenai skenario mengelola data produk. Diagram aktivitas dari kasus penggunaan mengelola data produk dapat dilihat pada Gambar 3.5.

Tabel 3.8 Rincian Alur Kasus Penggunaan Mengelola Data Produk

Nama Use Case	Mengelola Data Produk
Nomor	UC004
Aktor	Administrator
Deskripsi	Administrator dapat mengelola data produk berupa melihat detail data produk, menambah data produk, mengubah data produk baru, dan menghapus data produk.
Trigger	Administrator membuka menu produk
Kondisi Awal	Administrator membuka menu produk
Kondisi Akhir	Admin mengelola data produk
Alur Normal	<ol style="list-style-type: none"> 1. Administrator membuka menu data produk 2. Sistem menampilkan menu data produk beserta tombol detail, tambah, edit, hapus pada halaman yang sama.

Alur Alternatif	<p>2.A. Administrator memilih melihat detail data produk</p> <p>2.A.1. Administrator memilih tombol lihat detail data produk</p> <p>2.A.2. Sistem menampilkan detail data produk</p> <p>2.B. Administrator memilih menambah data produk</p> <p>2.B.1. Administrator memilih tombol tambah data produk</p> <p>2.B.2. Sistem menampilkan form untuk menambah data produk</p> <p>2.B.3. Administrator mengisi form data produk baru</p> <p>2.B.4. Sistem menyimpan ke <i>database</i>.</p> <p>2.C. Administrator memilih mengubah data produk</p> <p>2.C.1. Administrator memilih tombol ubah data produk</p> <p>2.C.2. Sistem menampilkan form ubah data produk</p> <p>2.C.3. Administrator mengisi data produk yang ingin diubah</p> <p>2.C.4. Sistem menyimpan ke <i>database</i></p> <p>2.D. Administrator memilih menghapus data produk</p> <p>2.D.1. Administrator memilih tombol hapus data produk</p> <p>2.D.2. Sistem menghapus data produk</p>
------------------------	---



Gambar 3.5 Diagram Aktivitas UC004

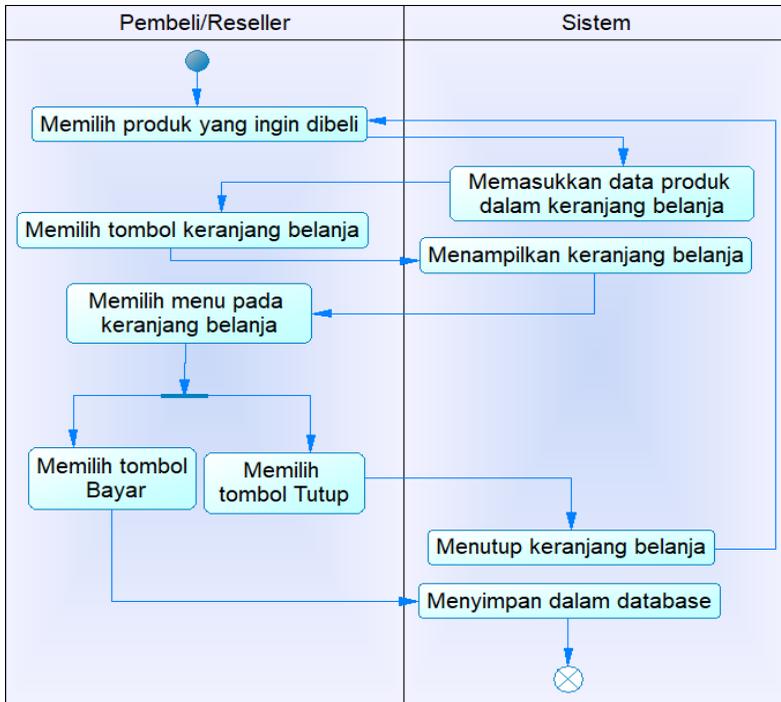
3.1.3.4.5 Kasus Penggunaan Melakukan Pemesanan

Pada kasus penggunaan melakukan pemesanan ini diakses ketika pembeli atau *reseller* hendak membeli produk melalui aplikasi. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.9.

Tabel 3.9 Rincian Alur Kasus Penggunaan Melakukan Pemesanan

Nama Use Case	Melakukan Pemesanan
Nomor	UC005
Aktor	Pembeli atau <i>Reseller</i>
Deskripsi	Pembeli atau <i>reseller</i> melakukan pemesanan produk untuk transaksi pembelian
Trigger	Pembeli atau <i>reseller</i> memilih produk untuk dibeli
Kondisi Awal	Keranjang belanja kosong
Kondisi Akhir	Produk dalam keranjang belanja masuk dalam <i>database</i> transaksi pembelian
Alur Normal	<ol style="list-style-type: none"> 1. Pembeli atau <i>reseller</i> telah memilih produk yang ingin dipesan 2. Sistem memasukkan data produk yang telah dipilih 3. Pembeli atau <i>reseller</i> membuka keranjang belanja 4. Sistem menampilkan keranjang belanja 5. Pembeli atau <i>reseller</i> memilih menu pada keranjang belanja
Alur Alternatif	<ol style="list-style-type: none"> 5.A. Pembeli memilih tombol Bayar <ol style="list-style-type: none"> 5.A.1. Sistem menampilkan detail pemesanan dan tombol konfirmasi 5.A.2 Pembeli memilih konfirmasi 5.A.3 Sistem menyimpan data pemesanan dalam data transaksi dan menampilkan riwayat transaksi 5.B. Pembeli memilih tombol Tutup <ol style="list-style-type: none"> 5.B.1. Sistem menampilkan halaman sebelumnya. 5.B.2. Pembeli dapat melanjutkan pemesanan.

Berdasarkan skenario kasus penggunaan pada Tabel 3.9 selanjutnya skenario tersebut digambarkan kedalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mengelola data produk dapat dilihat pada Gambar 3.6.



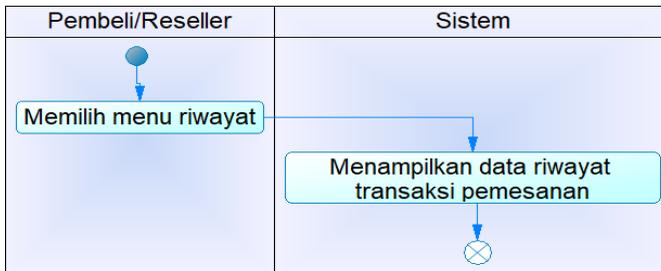
Gambar 3.6 Diagram Aktivitas UC005

3.1.3.4.6 Kasus Penggunaan Melihat Riwayat Transaksi

Pada kasus penggunaan melihat riwayat transaksi ini diakses ketika pembeli atau *reseller* memilih menu riwayat pada halaman *web*. Dalam hal ini jika pembeli atau *reseller* pernah melakukan pemesanan maka akan ditampilkan riwayat pemesanan yang pernah dilakukan. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.10.

Tabel 3.10 Rincian Alur Kasus Penggunaan Melihat Riwayat Transaksi

Nama Use Case	Melihat Riwayat Transaksi
Nomor	UC006
Aktor	Pembeli atau <i>Reseller</i>
Deskripsi	Pembeli atau <i>Reseller</i> melihat riwayat pembelian dari transaksi yang pernah dilakukan
Trigger	Pembeli atau <i>Reseller</i> memilih menu riwayat
Kondisi Awal	Data transaksi pembelian telah tersimpan dalam <i>database</i>
Kondisi Akhir	Menampilkan data riwayat transaksi pemesanan
Alur Normal	<ol style="list-style-type: none"> 1. Pembeli atau <i>Reseller</i> memilih menu riwayat 2. Sistem menampilkan data riwayat transaksi
Alur Alternatif	



Gambar 3.7 Diagram Aktivitas UC006

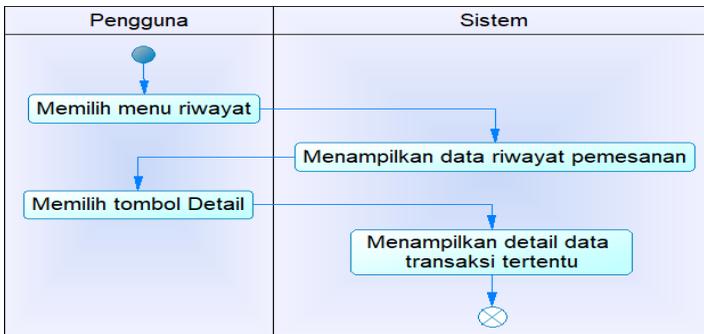
Berdasarkan skenario kasus penggunaan pada Tabel s3.10, selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan melihat riwayat transaksi dapat dilihat pada Gambar 3.7.

3.1.3.4.7 Kasus Penggunaan Melihat Detail Pemesanan

Pada kasus penggunaan melihat detail pemesanan ini diakses ketika pengguna memilih tombol Detil pada halaman riwayat pada salah satu data transaksi tertentu. Penjelasan skenario lebih rinci dapat dilihat pada Tabel 3.11.

Tabel 3.11 Rincian Alur Kasus Penggunaan Melihat Detail Pemesanan

Nama Use Case	Melihat Detail Pembelian
Nomor	UC007
Aktor	Pembeli, <i>Reseller</i> , dan Administrator
Deskripsi	Pengguna melihat detail pembelian dari transaksi tertentu
Trigger	Pengguna memilih tombol Detil
Kondisi Awal	Data transaksi pembelian telah tersimpan dalam <i>database</i> dan telah masuk halaman Riwayat
Kondisi Akhir	Menampilkan detail data transaksi tertentu
Alur Normal	<ol style="list-style-type: none"> 1. Pembeli atau <i>Reseller</i> memilih menu riwayat 2. Sistem menampilkan data riwayat transaksi pemesanan 3. Pengguna memilih tombol Detil 4. Sistem menampilkan detail data transaksi
Alur Alternatif	



Gambar 3.8 Diagram Aktivitas UC007

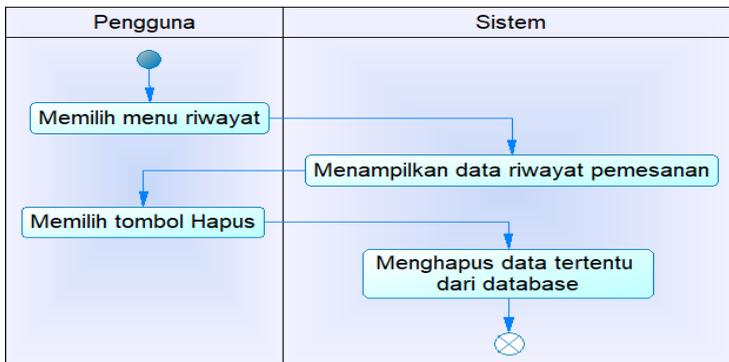
Selanjutnya skenario tersebut digambarkan dalam diagram aktivitas yang dapat dilihat pada Gambar 3.8.

3.1.3.4.8 Kasus Penggunaan Menghapus Pembelian

Pada kasus penggunaan menghapus pembelian ini diakses ketika pengguna memilih tombol Hapus pada halaman riwayat dari data transaksi tertentu. Penjelasan skenario dilihat pada Tabel 3.12.

Tabel 3.12 Rincian Alur Kasus Penggunaan Menghapus Pembelian

Nama Use Case	Menghapus Data Pembelian
Nomor	UC008
Aktor	Pembeli, <i>Reseller</i> , dan Administrator
Deskripsi	Pengguna menghapus data transaksi tertentu
Trigger	Pengguna memilih tombol Hapus
Kondisi Awal	Data transaksi pembelian telah tersimpan dalam <i>database</i> dan telah masuk halaman Riwayat
Kondisi Akhir	Menghapus data transaksi tertentu
Alur Normal	<ol style="list-style-type: none"> 1. Pembeli atau <i>Reseller</i> memilih menu riwayat 2. Sistem menampilkan data riwayat transaksi pemesanan 3. Pengguna memilih tombol Hapus 4. Sistem menghapus data transaksi
Alur Alternatif	

**Gambar 3.9 Diagram Aktivitas UC008**

Selanjutnya skenario tersebut digambarkan dalam diagram aktivitas yang dapat dilihat pada Gambar 3.9.

3.1.3.4.9 Kasus Penggunaan Melakukan Pembayaran

Pada kasus penggunaan melakukan pembayaran ini diakses ketika pembeli hendak mengunggah bukti pembayaran transaksi setelah mengirim biaya pembelian. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.13.

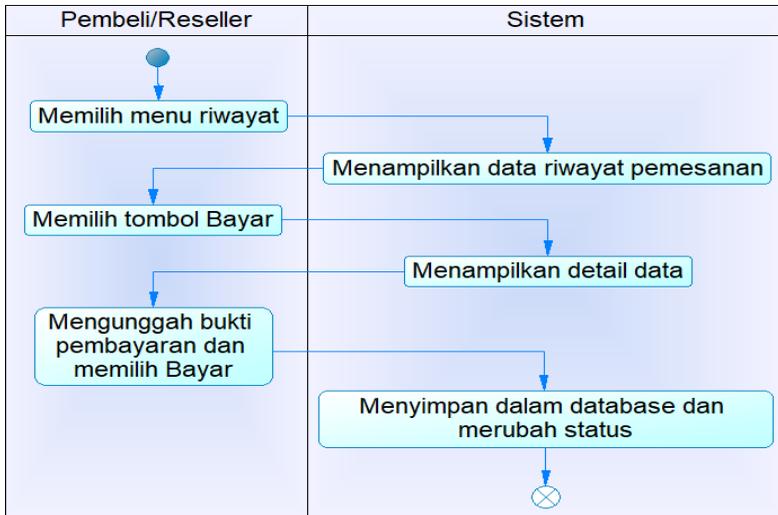
Tabel 3.13 Rincian Alur Kasus Penggunaan Melakukan Pembayaran

Nama Use Case	Melakukan Pembayaran
Nomor	UC009
Aktor	Pembeli atau <i>Reseller</i>
Deskripsi	Pembeli atau <i>Reseller</i> melakukan pembayaran dari pemesanan produk yang telah dilakukan
Trigger	Pembeli atau <i>Reseller</i> memilih menu Bayar pada halaman riwayat dari transaksi pemesanan yang dilakukan
Kondisi Awal	Menampilkan data riwayat transaksi pemesanan
Kondisi Akhir	Pembeli atau <i>Reseller</i> mengunggah bukti pembayaran
Alur Normal	<ol style="list-style-type: none"> 1. Pembeli atau <i>Reseller</i> memilih menu riwayat 2. Sistem menampilkan data riwayat transaksi pemesanan 3. Pembeli atau <i>Reseller</i> memilih tombol Bayar pada transaksi pemesanan yang belum dibayar 4. Sistem menampilkan detail transaksi yang belum dibayar 5. Pembeli atau <i>Reseller</i> mengunggah bukti pembayaran dan memilih Bayar 6. Sistem menyimpan dalam <i>database</i> dan merubah status pembayaran menjadi 1
Alur Alternatif	

Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan melakukan pembayaran dapat dilihat pada Gambar 3.10.

3.1.3.4.10 Kasus Penggunaan Mengelola Data Transaksi

Pada kasus penggunaan mengelola transaksi ini diakses ketika Administrator hendak melakukan perubahan data pada transaksi. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.14.



Gambar 3.10 Diagram Aktivitas UC009

Tabel 3.14 Rincian Alur Kasus Penggunaan Mengelola Data Transaksi

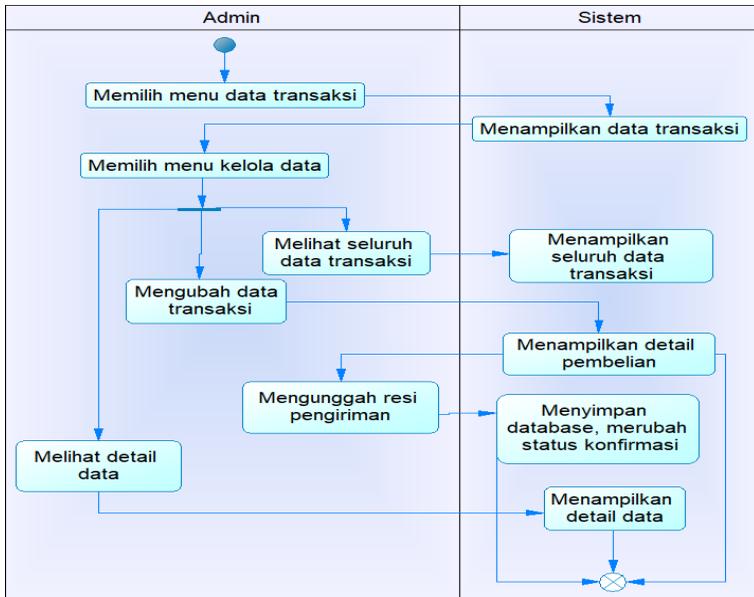
Nama Use Case	Mengelola Data Transaksi
Nomor	UC010
Aktor	Administrator
Deskripsi	Administrator dapat mengelola data transaksi berupa melihat seluruh data transaksi, melihat detail salah satu data, dan mengubah status konfirmasi pengiriman produk.
Trigger	Administrator memilih menu data transaksi
Kondisi Awal	Data transaksi telah tersimpan dalam <i>database</i>
Kondisi Akhir	Perubahan dalam <i>database</i> tersimpan kembali
Alur Normal	<ol style="list-style-type: none"> 1. Administrator memilih menu data transaksi 2. Sistem menampilkan data transaksi yang memiliki status konfirmasi pengiriman 0 namun status pembayaran 1, serta tombol untuk merubah data, tombol melihat detail transaksi, dan menu melihat seluruh data transaksi

Alur Alternatif	<p>2.A. Administrator memilih ubah data</p> <p>2.A.1. Administrator memilih tombol ubah pada salah satu data</p> <p>2.A.2. Sistem menampilkan detail data transaksi pembelian</p> <p>2.A.3. Administrator mengunggah resi pengiriman</p> <p>2.A.4. Sistem menyimpan dalam <i>database</i> dan merubah status konfirmasi menjadi 1</p> <p>2.B. Administrator memilih melihat detail data</p> <p>2.B.1. Administrator memilih tombol detail pada salah satu data</p> <p>2.B.2. Sistem menampilkan detail data transaksi yang dipilih</p> <p>2.C. Administrator memilih menu melihat seluruh data transaksi</p> <p>2.C.1. Administrator memilih menu seluruh data transaksi</p> <p>2.C.2. Sistem menampilkan seluruh data transaksi yang ada</p>
------------------------	---

Berdasarkan skenario kasus penggunaan pada Tabel 3.14. Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas kasus penggunaan mengelola transaksi dapat dilihat pada Gambar 3.11.

3.1.3.4.11 Kasus Penggunaan Melihat Forum Tanya Jawab

Pada kasus penggunaan melihat forum tanya jawab ini diakses ketika pengunjung atau pembeli memilih menu Tanya Jawab pada halaman *web* bagian bawah. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.15.



Gambar 3.11 Diagram Aktivitas UC10

Tabel 3.15 Rincian Alur Kasus Penggunaan Melihat Forum Tanya Jawab

Nama Use Case	Melihat Forum Tanya Jawab
Nomor	UC011
Aktor	Pembeli, Pengunjung, Administrator, <i>Reseller</i>
Deskripsi	Pengguna dapat melihat forum tanya jawab antara pengguna dengan Administrator
Trigger	Pengguna membuka menu Tanya Jawab
Kondisi Awal	Daftar pertanyaan dan jawaban telah tersimpan dalam <i>database</i>
Kondisi Akhir	Sistem menampilkan daftar pertanyaan dan jawaban yang ada
Alur Normal	<ol style="list-style-type: none"> 1. Pengunjung atau Pembeli memilih menu Tanya Jawab, pada halaman bawah <i>web</i> 2. Sistem menampilkan halaman menu Tanya Jawab
Alur Alternatif	



Gambar 3.12 Diagram Aktivitas UC011

Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas yang dapat dilihat pada Gambar 3.12.

3.1.3.4.12 Kasus Penggunaan Mengirim Pertanyaan

Pada kasus penggunaan mengirim pertanyaan ini diakses ketika pembeli hendak menanyakan sesuatu dengan menggunakan fungsi ini. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.16.

Tabel 3.16 Rincian Alur Kasus Penggunaan Mengirim Pertanyaan

Nama Use Case	Mengirim Pertanyaan
Nomor	UC012
Aktor	Pengunjung, Pembeli, dan <i>Reseller</i>
Deskripsi	Pengguna akan membuka menu Kontak untuk mengirim pertanyaan.
Trigger	Pengguna memilih menu Kontak
Kondisi Awal	Data pertanyaan baru akan tersimpan dalam <i>database</i> .
Kondisi Akhir	Sistem menyimpan data pertanyaan yang baru
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu Kontak 2. Sistem menampilkan halaman Kontak dengan form untuk mengirim pertanyaan 3. Pengguna mengisi form dan mengirim pertanyaan 4. Sistem menyimpan data pertanyaan baru
Alur Alternatif	

Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan mengirim pertanyaan dapat dilihat pada Gambar 3.13.



Gambar 3.13 Diagram Aktivitas UC012

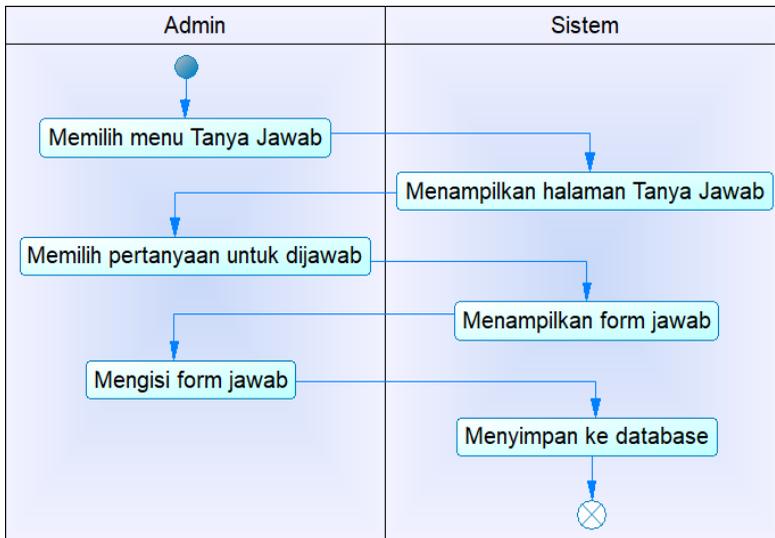
3.1.3.4.13 Kasus Penggunaan Menjawab Pertanyaan

Pada kasus penggunaan menjawab pertanyaan ini diakses ketika ada pertanyaan baru yang masuk, kemudian Administrator hendak memberikan jawaban dari pertanyaan tersebut. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.17. Selanjutnya berdasarkan skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan melihat pertanyaan dapat dilihat pada Gambar 3.14.

Tabel 3.17 Rincian Alur Kasus Penggunaan Menjawab Pertanyaan

Nama Use Case	Menjawab Pertanyaan
Nomor	UC013
Aktor	Administrator
Deskripsi	Administrator menjawab pertanyaan yang ada dalam daftar pertanyaan
Trigger	Administrator memilih menu Tanya Jawab
Kondisi Awal	Data pertanyaan telah ada dalam <i>database</i>
Kondisi Akhir	Sistem menyimpan jawaban dari pertanyaan

Alur Normal	<ol style="list-style-type: none"> 1. Administrator memilih halaman Tanya Jawab 2. Sistem menampilkan halaman Tanya Jawab 3. Administrator memilih pertanyaan untuk dijawab 4. Sistem menampilkan form untuk menjawab 5. Administrator mengisi form, memilih menyimpan 6. Sistem menyimpan jawaban dalam <i>database</i>, menampilkan pada halaman <i>web</i>
Alur Alternatif	



Gambar 3.14 Diagram Aktivitas UC013

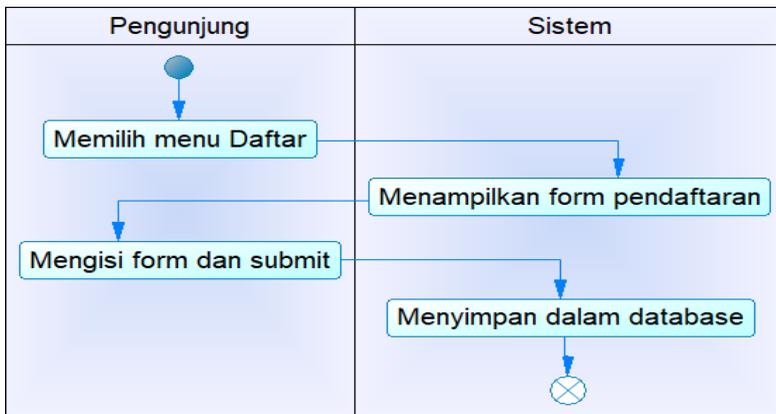
3.1.3.4.14 Kasus Penggunaan Menambah Pengguna Pembeli

Pada kasus penggunaan menambah pengguna ini diakses ketika pengunjung ingin membuat akun pada aplikasi sebagai pembeli agar dapat melakukan pemesanan dan pembelian dengan menggunakan aplikasi. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.18.

Tabel 3.18 Rincian Alur Kasus Penggunaan Menambah Pengguna Pembeli

Nama Use Case	Menambah Pengguna Pembeli
Nomor	UC014
Aktor	Pengunjung
Deskripsi	Pengunjung memilih menu Daftar untuk menjadi pembeli.
Trigger	Pengunjung memilih menu Daftar
Kondisi Awal	Pengunjung belum memiliki akun sebagai Pembeli
Kondisi Akhir	Sistem menampilkan form pendaftaran
Alur Normal	<ol style="list-style-type: none"> 1. Pengunjung memilih menu Daftar 2. Sistem menampilkan form pendaftaran 3. Pengunjung mengisi form pendaftaran, hingga memilih submit form 4. Sistem menyimpan data Pembeli baru.
Alur Alternatif	

Berdasarkan skenario kasus penggunaan pada. Selanjutnya skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menambah pengunjung pembeli dapat dilihat pada Gambar 3.15.



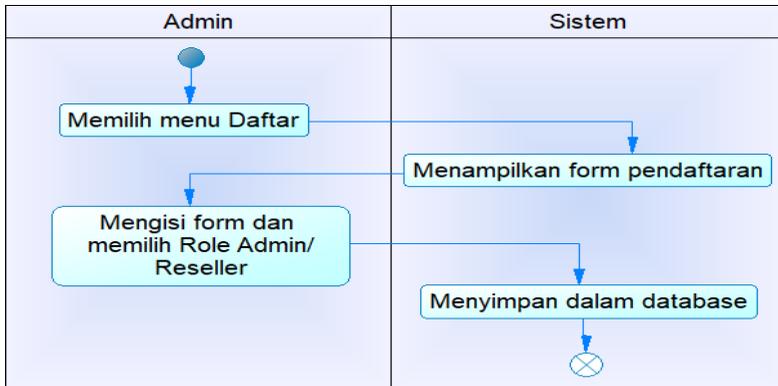
Gambar 3.15 Diagram Aktivitas UC014

3.1.3.4.15 Kasus Penggunaan Menambah Pengguna Administrator/*Reseller*

Pada kasus penggunaan menambah Administrator atau *reseller* ini diakses ketika Administrator ingin menambah pengguna sebagai Administrator baru atau *reseller* baru. Dalam hal ini Administrator diharuskan telah *login* terlebih dahulu. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.19. Selanjutnya berdasarkan skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan menambah pengguna Administrator atau *reseller* dapat dilihat pada Gambar 3.16.

Tabel 3.19 Rincian Alur Kasus Penggunaan Menambah Pengguna Administrator/*Reseller*

Nama Use Case	Menambah Pengguna Administrator/ <i>Reseller</i>
Nomor	UC015
Aktor	Administrator
Deskripsi	Administrator menambah Administrator atau
Trigger	Administrator memilih menu Daftar
Kondisi Awal	Administrator harus login terlebih dahulu
Kondisi Akhir	Sistem menyimpan data Administrator atau
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna telah login sebagai Administrator, dan memilih menu Daftar 2. Sistem menampilkan form pendaftaran 3. Administrator mengisi form pendaftaran, dan memilih submit untuk menyimpan data 4. Sistem menyimpan data kedalam <i>database</i>
Alur Alternatif	



Gambar 3.16 Diagram Aktivitas UC015

3.1.3.4.16 Kasus Penggunaan Melihat Profil

Pada kasus penggunaan melihat profil pengguna ini diakses ketika pengguna memilih menu profil. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.20. Selanjutnya berdasarkan skenario tersebut digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan melihat profil dapat dilihat pada Gambar 3.17.

Tabel 3.20 Rincian Alur Kasus Penggunaan Melihat Profil

Nama Use Case	Melihat profil pengguna
Nomor	UC016
Aktor	Pembeli, <i>Reseller</i> , Administrator
Deskripsi	Pengguna yang memiliki akun dapat melihat profil diri
Trigger	Pengguna memilih menu Profil
Kondisi Awal	Pengguna harus telah login
Kondisi Akhir	Sistem menampilkan data diri sesuai pengguna yang login
Alur Normal	<ol style="list-style-type: none"> 1. Pengguna memilih menu Profil 2. Sistem menampilkan data diri Pengguna
Alur Alternatif	



Gambar 3.17 Diagram Aktivitas UC016

3.1.3.4.17 Kasus Penggunaan Melihat Riwayat Produk

Pada kasus penggunaan melihat riwayat produk ini diakses ketika Administrator memilih menu riwayat produk, pada halaman ini dilakukan perhitungan dengan menggunakan aturan Asosiasi Apriori. Untuk dapat melihat riwayat produk pada halaman ini dibutuhkan masukkan untuk nilai minimum *support* dan nilai minimum *confidence* untuk menjadi syarat apa saja yang dapat ditampilkan. Penjelasan skenario yang lebih rinci dapat dilihat pada Tabel 3.21. Selanjutnya berdasarkan skenario dari kasus penggunaan melihat riwayat produk, digambarkan ke dalam diagram aktivitas. Diagram aktivitas dari kasus penggunaan melihat riwayat pembelian produk dengan aturan Asosiasi dapat dilihat pada Gambar 3.18.



Gambar 3.18 Diagram Aktivitas UC017

Tabel 3.21 Rincian Alur Kasus Penggunaan Melihat Riwayat Pembelian Produk

Nama Use Case	Melihat Riwayat Pembelian Produk
Nomor	UC017
Aktor	Administrator
Deskripsi	Administrator melihat riwayat pembelian produk berdasarkan aturan Asosiasi Apriori
Trigger	Administrator memasukkan nilai minimum <i>support</i> dan nilai minimum <i>confidence</i>
Kondisi Awal	Data transaksi telah tersimpan dalam riwayat pembelian
Kondisi Akhir	Sistem menampilkan riwayat pembelian berdasarkan aturan Asosiasi Apriori sesuai dengan nilai minimum <i>support</i> dan nilai minimum <i>confidence</i> yang telah dimasukkan
Alur Normal	<ol style="list-style-type: none"> 1. Administrator memasukkan nilai minimum <i>support</i> dan nilai minimum <i>confidence</i> 2. Sistem menampilkan hasil riwayat pembelian sesuai dengan aturan Asosiasi Apriori dari riwayat pembelian
Alur Alternatif	

3.2 Perancangan Sistem

Pada subbab ini dijelaskan mengenai tahapan perancangan sistem. Perancangan sistem ini dibagi menjadi beberapa bagian yang meliputi perancangan arsitektur sistem, perancangan antarmuka pengguna, serta perancangan data.

3.2.1 Perancangan Arsitektur

Dalam tugas akhir ini akan dirancang dan diaplikasikan sebuah perangkat lunak untuk industri rumah tangga makanan ringan berbasis *web* dengan menggunakan metode algoritma Apriori. Pada aplikasi ini, terdapat empat pengguna, yaitu Administrator, Pengunjung, Pembeli, dan *Reseller*. Administrator disini bertugas untuk mengelola data informasi, data produk, data

transaksi, mendaftarkan Administrator atau *Reseller* baru, dan mampu melihat riwayat pembelian yang diminati oleh Pembeli dengan menggunakan aturan asosiasi Apriori, dan ditampilkan berdasarkan inputan nilai minimum *support* dan nilai minimum *confidence*, yang mana hasil dari perhitungan ini akan menjadi rekomendasi produk yang akan ditampilkan pada halaman pengguna selain Administrator. Pengunjung merupakan pengguna yang belum terdaftar yang memiliki hak untuk melihat produk yang dijual, mendaftar sebagai Pembeli, melihat fitur pertanyaan serta mengirim pertanyaan pada halaman Kontak. Pembeli dan *Reseller* adalah pengguna yang telah mendaftarkan diri pada aplikasi ini, memiliki fitur melihat produk yang dijual, melakukan pemesanan produk dengan memilih produk, memasukkan jumlah produk sebanyak yang diinginkan. Selain itu Pembeli juga tetap dapat melihat semua halaman, serta mengirimkan pertanyaan langsung kepada Administrator melalui halaman Kontak yang dapat dipilih pada bagian bawah halaman, namun untuk *Reseller* harga pembelian produk berbeda dengan Pembeli karena jumlah minimum pembelian pun berbeda.

3.2.2 Perancangan Perangkat Lunak

Spesifikasi perangkat keras serta perangkat lunak yang digunakan dalam tahap perancangan perangkat lunak tugas akhir ini dapat dilihat pada Tabel 3.22.

Tabel 3.22 Perancangan Perangkat Lunak

Perangkat Keras	Komputer	HP ProBook 450 G4
	Prosesor	Intel® Core™ i5-7200U CPU @ 2.50GHz 2.71GHz
	Memori Primer	8 GB
	Memori Sekunder	1 TB
Perangkat Lunak	Sistem Operasi	<i>Windows</i> 10 Home 64-bit
	Perangkat Lunak	Microsoft Word 2016, Jetbrains PhpStorm 2017.3.1, XAMPP Version 7.1.10

3.2.3 Perancangan Antarmuka Pengguna

Pada bagian ini dijelaskan mengenai rancangan tampilan antarmuka pengguna dari sistem. Antarmuka pengguna yang berhubungan langsung dengan aktor, harus memiliki tampilan yang menarik bagi penggunanya. Aplikasi ini menggunakan antarmuka yang indah dilihat. Pengguna dapat mengakses menggunakan semua *web* browser dan hanya menggunakan keyboard dan mouse. Aplikasi ini memiliki tiga hak akses, yaitu Administrator, *reseller* dan pembeli dengan halaman *login* pengguna yang sama. Berikut akan dijelaskan masing rancangan antarmuka.

3.2.3.1 Antarmuka Menangani Kelola Data Informasi

Halaman ini akan ditampilkan ketika Administrator akan mengelola data informasi. Dalam hal kelola data ini dapat berupa menambah, mengubah, serta menghapus yang hanya dapat dilakukan oleh Administrator. Halaman ini akan ditampilkan setelah Administrator memilih menu “Data Informasi”. Rancangan halaman dapat dilihat pada Gambar 3.19 untuk menambah data informasi dan Gambar 3.20 untuk mengubah data informasi, dan atribut untuk antarmuka dapat dilihat pada Tabel 3.23.

The image shows a web browser window with the following elements:

- Browser title: A Web Page
- Address bar: http://
- Page title: MAKARONI MEWEK
- Navigation menu: Beranda, Data Informasi, Data Produk, Data Transaksi, Tanya Jawab
- Main content area: Tambah Informasi
- Form fields:
 - Judul Informasi:
 - Tanggal Mulai: / / (with calendar icon)
 - Tanggal Selesai: / / (with calendar icon)
 - Detail Info:
 - Detail Informasi:
 - Jenis Informasi:
 - Submit:

Gambar 3.19 Rancangan Halaman Antarmuka Tambah Informasi

The image shows a web browser window titled 'A Web Page' with a URL bar containing 'http://'. The page header is 'MAKARONI MEWEK'. Below the header is a navigation menu with 'Beranda', 'Data Informasi', 'Data Produk', 'Data Transaksi', and 'Tanya Jawab'. The 'Data Informasi' menu item is highlighted. The main content area is titled 'Edit Informasi' and contains a form with the following elements:

- Judul Informasi: A text input field.
- Tanggal Mulai: A date input field with a calendar icon.
- Tanggal Selesai: A date input field with a calendar icon.
- Detail Info: A small square icon.
- Detail Informasi: A text input field.
- Jenis Informasi: A text input field.
- Submit: A button.

Gambar 3.20 Rancangan Halaman Antarmuka *Edit* Data Informasi

Tabel 3.23 Atribut Antarmuka Kelola Data Informasi

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/ Keluaran
1	<i>edt_judulinfo</i>	<i>Edit text</i>	Input untuk memasukkan judul informasi	<i>String</i>
2	<i>edt_tglmulai</i>	<i>Edit text</i>	Input untuk memasukkan tgl mulai	<i>Date</i>
3	<i>edt_tglselesai</i>	<i>Edit text</i>	Input untuk memasukkan tgl selesai	<i>Date</i>
4	<i>edt_detailinfo</i>	<i>Button</i>	Input untuk memasukkan gambar detail info	<i>File</i>
5	<i>edt_deskripsi</i>	<i>Edit text</i>	Input untuk memasukkan deskripsi	<i>String</i>

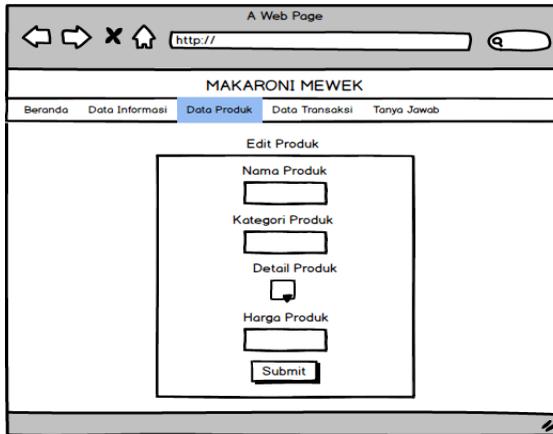
No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/Keluaran
6	<i>spinner_jenis_info</i>	<i>Dropdown List</i>	Berisi pilihan jenis informasi	<i>Button click</i>
7	<i>btn_addinfo</i>	<i>Button</i>	Tombol untuk menyimpan data informasi yang baru ditambahkan/ diubah	<i>Button click</i>

3.2.3.2 Antarmuka Menangani Kelola Data Produk

Halaman ini akan diakses ketika Administrator akan mengelola data produk. Dalam hal ini dapat berupa menambah, mengubah, serta menghapus yang hanya dapat dilakukan oleh Administrator ketika memilih menu “Data Produk”. Rancangan halaman dapat dilihat pada Gambar 3.21 untuk menambah data dan Gambar 3.22 untuk mengubah data. Atribut dari antarmuka terdiri dari enam atribut, diantaranya digunakan untuk mengisi form tambah dan ubah data, tombol untuk mengunggah gambar detail produk, dan tombol Submit untuk menyimpan data produk yang baru ditambah atau diubah, dapat dilihat pada Tabel 3.24.

The image shows a web browser window titled "A Web Page" with a navigation menu for "MAKARONI MEWEK". The "Data Produk" menu item is selected. The main content area displays a "Tambah Produk" form with the following fields: "Nama Produk", "Kategori Produk", "Detail Produk" (with a file upload icon), and "Harga Produk". A "Submit" button is located at the bottom of the form.

Gambar 3.21 Rancangan Antarmuka Menambah Data Produk



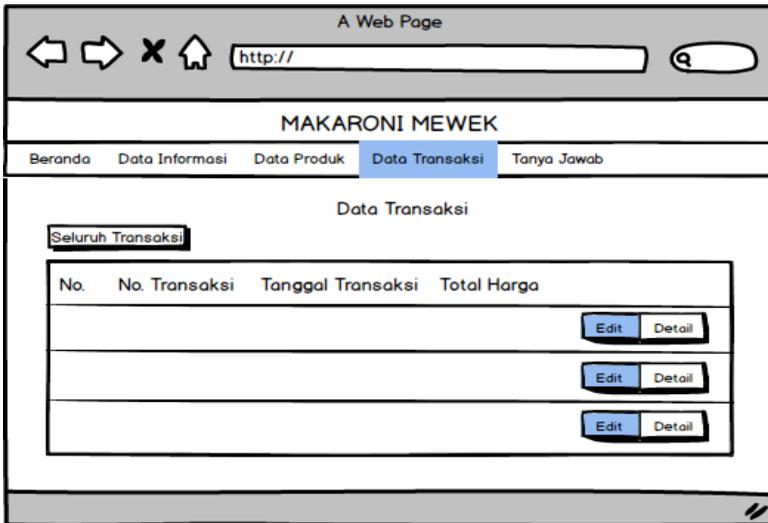
Gambar 3.22 Rancangan Halaman Antarmuka Mengubah Data Produk

Tabel 3.24 Atribut Antarmuka Mengelola Data Produk

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/Keluaran
1	<i>edt_nama produk</i>	<i>Edit text</i>	Input untuk memasukkan nama produk	<i>String</i>
2	<i>edt_kategori</i>	<i>Edit text</i>	Input untuk memasukkan kategori	<i>String</i>
3	<i>edt_infoproduk</i>	<i>Button</i>	Input untuk memasukkan gambar produk	<i>File</i>
4	<i>edt_bykproduk</i>	<i>Edit text</i>	Input untuk memasukkan banyaknya produk	<i>Number</i>
5	<i>edt_harga</i>	<i>Edit text</i>	Input untuk memasukkan harga produk	<i>Number</i>
6	<i>btn_addproduk</i>	<i>Button</i>	Tombol untuk menyimpan data produk yang baru ditambahkan/diubah	<i>Button click</i>

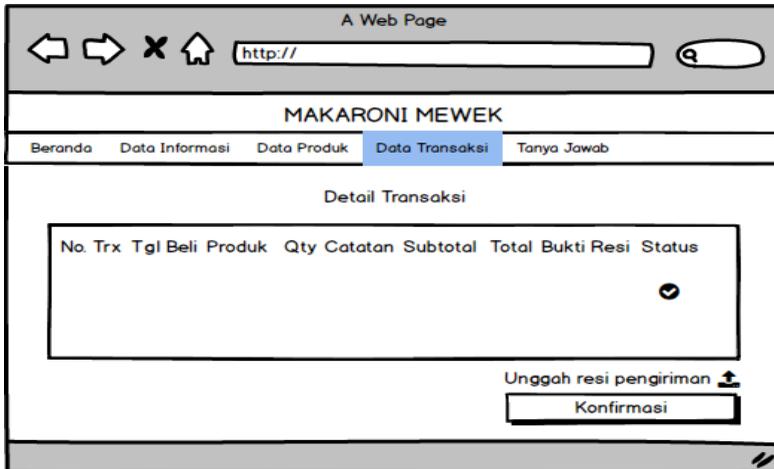
3.2.3.3 Antarmuka Menangani Kelola Transaksi

Halaman ini akan ditampilkan ketika Administrator akan mengelola data transaksi. Dalam hal kelola data ini dapat berupa melihat seluruh data transaksi, mengubah, serta melihat detail masing-masing data transaksi yang hanya dapat dilakukan oleh Administrator. Halaman ini akan ditampilkan ketika Administrator menekan menu “Data Transaksi”. Rancangan halaman antarmuka dapat dilihat pada Gambar 3.23 untuk mengelola data transaksi. Kemudian pada Tabel 3.25 merupakan atribut antarmuka untuk mengelola transaksi, terdapat tiga tombol yang digunakan untuk melihat keseluruhan data transaksi, mengubah data transaksi, dan melihat detail data transaksi tertentu.

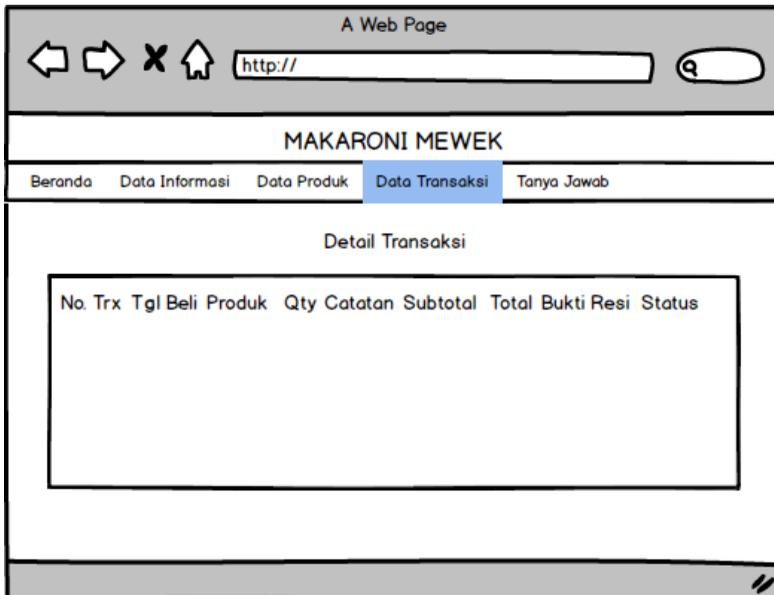


Gambar 3.23 Rancangan Halaman Antarmuka Mengelola Data Transaksi

Pada Gambar 3.24 untuk rancangan antarmuka mengubah data transaksi, dan Gambar 3.25 untuk rancangan antarmuka melihat detail data transaksi tertentu.



Gambar 3.24 Rancangan Halaman Antarmuka Mengubah Data Transaksi



Gambar 3.25 Rancangan Halaman Antarmuka Melihat Detail Data Transaksi Tertentu

Tabel 3.25 Atribut Antarmuka Mengelola Transaksi

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/ Keluaran
1	<i>btn_lihat transaksi</i>	<i>Button</i>	Tombol untuk masuk ke halaman data transaksi secara keseluruhan	<i>Button click</i>
2	<i>btn_ubah transaksi</i>	<i>Button</i>	Tombol untuk masuk ke halaman edit status	<i>Button click</i>
3	<i>btn_detail</i>	<i>Button</i>	Tombol untuk melihat detail data transaksi	<i>Button click</i>

3.2.3.4 Antarmuka Menambah Data Pengguna

Halaman ini akan ditampilkan ketika pengunjung atau Administrator memilih tombol *Daftar* pada halaman *web* bagian atas. Halaman ini menampilkan form untuk menambah data pengguna. Rancangan halaman antarmuka dapat dilihat pada Gambar 3.26. Atribut yang digunakan untuk halaman antarmuka daftar pengguna baru dapat dilihat pada Tabel 3.26.

Tabel 3.26 Atribut Antarmuka Menambah Pengguna

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/ Keluaran
1	<i>edt_namadpn</i>	<i>Edit text</i>	Input untuk memasukkan nama depan	<i>String</i>
2	<i>edt_namabkg</i>	<i>Edit text</i>	Input untuk memasukkan nama belakang	<i>String</i>
3	<i>edt_alamat</i>	<i>Edit text</i>	Input untuk memasukkan alamat pengguna	<i>String</i>
4	<i>edt_notelp</i>	<i>Edit text</i>	Input untuk memasukkan nomor telepon pengguna	<i>String</i>
5	<i>edt_email</i>	<i>Edit text</i>	Input untuk memasukkan email pengguna	<i>String</i>
6	<i>edt_password</i>	<i>Edit text</i>	Input untuk memasukkan password pengguna	<i>String</i>

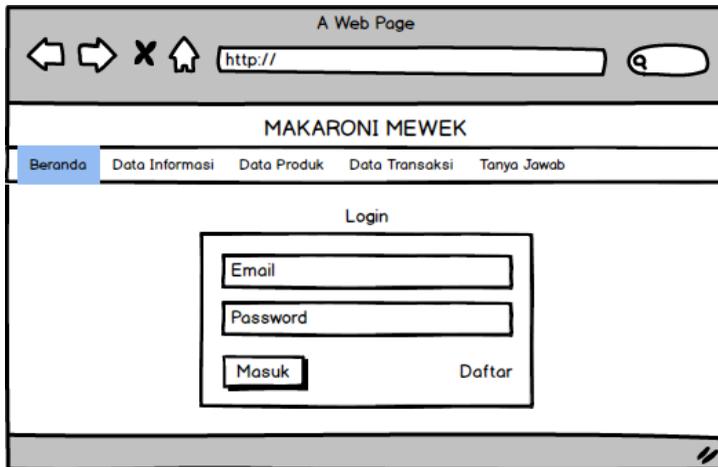
No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/Keluaran
7	<i>Spinner_role user</i>	<i>Dropdown List</i>	Berisi pilihan role pengguna yang dibawah Administrator	<i>Button click</i>
8	<i>btn_adduser</i>	<i>Button</i>	Tombol aksi untuk menyimpan data pengguna yang baru ditambahkan	<i>Button click</i>

The image shows a web browser window with the address bar containing 'http://'. The page title is 'MAKARONI MEWEK'. Below the title is a navigation menu with links: 'Beranda', 'Data Informasi', 'Data Produk', 'Data Transaksi', and 'Tanya Jawab'. The main content area is titled 'Daftar' and contains a registration form with the following fields: 'Nama Depan', 'Nama Belakang', 'Alamat', 'Handphone', 'Email', and 'Password'. Below these fields is a 'Select Status' dropdown menu with 'Admin' and 'Reseller' as options. A 'Submit' button is located at the bottom of the form.

Gambar 3.26 Rancangan Halaman Antarmuka Menambah Pengguna sebagai Pembeli

3.2.3.5 Antarmuka *Login* Pengguna

Halaman ini akan ditampilkan ketika pengguna akan menekan tombol Masuk pada atas halaman. Halaman ini menampilkan form untuk pengguna mengisi *email* dan *password* untuk dapat masuk ke dalam sistem jika pengguna telah memiliki akun pada aplikasi ini. Rancangan halaman dapat dilihat pada Gambar 3.27 dan atribut antarmuka dapat dilihat pada Tabel 3.27.



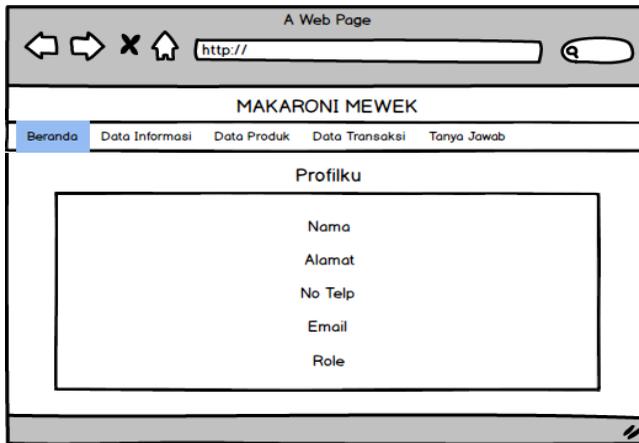
Gambar 3.27 Rancangan Halaman Antarmuka Masuk Pengguna

Tabel 3.27 Atribut Antarmuka Masuk Pengguna

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/Keluaran
1	<i>edt_username</i>	<i>Edit text</i>	Input untuk memasukkan <i>username</i> pengguna	<i>String</i>
2	<i>edt_password</i>	<i>Edit text</i>	Input untuk memasukkan <i>password</i> pengguna	<i>String</i>
3	<i>btn_login</i>	<i>Button</i>	Tombol untuk menuju halaman utama pengguna	<i>Button click</i>

3.2.3.6 Antarmuka Melihat Profil

Halaman ini akan ditampilkan ketika pengguna yang telah *login*, dan menekan tombol Profil pada atas halaman. Halaman ini menampilkan data pengguna sesuai dengan *role*-nya. Rancangan halaman dapat dilihat pada Gambar 3.28 dan atribut antarmuka dapat dilihat pada Tabel 3.28.



Gambar 3.28 Rancangan Halaman Antarmuka Melihat Profil

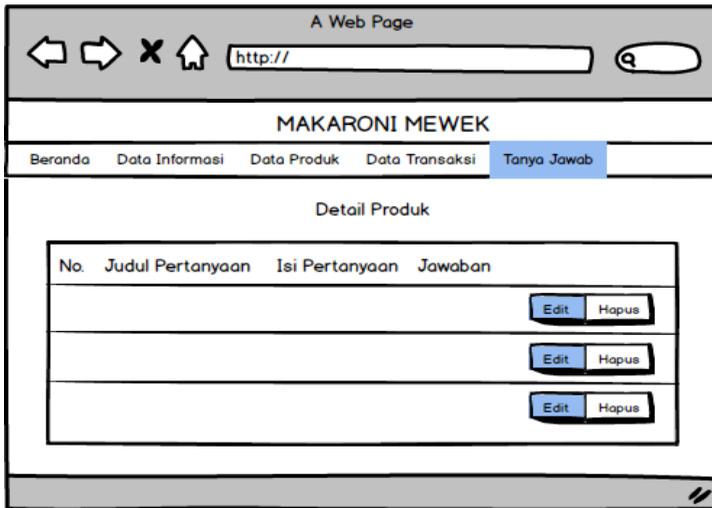
Tabel 3.28 Atribut Antarmuka Melihat Profil Pengguna

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/ Keluaran
1	<i>tv_nama</i> pengguna	<i>Text view</i>	Berisi informasi nama dari pengguna	<i>String</i>
2	<i>tv_alamat</i> pengguna	<i>Text view</i>	Berisi informasi alamat pengguna	<i>String</i>
3	<i>tv_telp</i> pengguna	<i>Text view</i>	Berisi informasi nomor telepon pengguna	<i>String</i>
4	<i>tv_email</i> pengguna	<i>Text view</i>	Berisi informasi email pengguna	<i>String</i>
5	<i>tv_role</i> pengguna	<i>Text view</i>	Berisi informasi role pengguna	<i>String</i>

3.2.3.7 Antarmuka Kelola Data Tanya Jawab

Halaman ini akan ditampilkan ketika Administrator memilih menu Tanya Jawab, dan melakukan ubah data ketika ada pertanyaan baru dari pengguna lain seperti, Pengunjung, Pembeli, dan *Reseller*. Rancangan halaman dapat dilihat pada Gambar 3.29. Pada Tabel 3.29 merupakan atribut pada halaman Tanya

Jawab, dapat melakukan ubah data untuk mengisi jawaban, dan menghapus pertanyaan dari daftar pertanyaan.



Gambar 3.29 Rancangan Halaman Antarmuka Mengelola Data Tanya Jawab

Tabel 3.29 Atribut Antarmuka Mengelola Data Tanya Jawab

No.	Nama Atribut Antarmuka	Jenis Atribut	Kegunaan	Masukan/ Keluaran
1	<i>btn_status transaksi</i>	<i>Button</i>	Tombol untuk masuk ke halaman edit status	<i>Button click</i>
2	<i>btn_hapus</i>	<i>Button</i>	Tombol untuk menghapus data	<i>Button click</i>

3.2.4 Perancangan Data

Dalam membuat suatu aplikasi berbasis *web*, dibutuhkan analisis kebutuhan berupa perancangan basis data. Basis data yang digunakan adalah MySQL. MySQL dipilih menjadi basis data aplikasi ini karena merupakan salah satu software DBMS yang *portable* dan bersifat *open source*, mudah digunakan, memiliki fitur keamanan yang baik.

Rancangan basis data ditampilkan dalam bentuk *Conceptual Data Model* (CDM) pada Gambar 3.30 dan *Physical Data Model* (PDM) pada Gambar 3.31. Sedangkan penjelasan lebih rinci dalam penggunaan tabel yang akan digunakan dalam basis data akan dijelaskan dalam kamus data pada subbab berikut.

3.2.4.1 Tabel User

Tabel *User* digunakan untuk menyimpan data pengguna yang memiliki hak akses pada sistem. Tabel ini mempunyai beberapa atribut, yaitu nama depan, nama belakang, alamat, no telp, *email*, *password*, dan *role* pengguna. Penjelasan lebih lanjut akan dijelaskan dalam Tabel 3.30.

Tabel 3.30 Kamus Data Tabel User

No.	Field	Type	Data Length	Deskripsi	Contoh
1	user_id	int	4	id untuk mengidentifikasi setiap pengguna	1, 2, 3, ...
2	Nama_dpn	varchar	35	Nama depan pengguna	Hilma
3	Nama_blg	varchar	35	Nama belakang pengguna	Kamilah
4	Alamat_user	varchar	50	Alamat pengguna	Jl. Pacar...
5	Telp_user	varchar	13	Telp pengguna	0856xx
6	Email_user	varchar	50	Email pengguna	hlm@gmail.com
7	Pass_user	varchar	50	Password pengguna	password
8	Role_user	int	4	Role <i>user</i>	Administrator

3.2.4.2 Tabel Role

Tabel *Role* adalah tabel yang digunakan untuk menyimpan data macam-macam hak akses untuk masuk ke dalam sistem. Tabel ini merupakan tabel utama yang mempunyai atribut-atribut yaitu *id role* dan nama *role*. Role terdiri dari Administrator, Pembeli, dan *Reseller*. Penjelasan lebih lanjut akan dijelaskan dalam Tabel 3.31.

Tabel 3.31 Kamus Data Tabel Role

No.	Field	Type	Data Length	Deskripsi	Contoh
1	role_id	int	2	id untuk mengidentifikasi setiap role untuk tabel user	1, 2, ...
2	role_name	varchar	10	Nama role	Administrator, <i>Reseller</i> , Pembeli

3.2.4.3 Tabel Informasi

Tabel Informasi adalah tabel yang digunakan untuk menyimpan data-data informasi mengenai adanya promo, adanya bazaar, pengumuman yang ditampilkan dalam aplikasi ini, atau hanya untuk menyimpan asset yang digunakan pada aplikasi ini. Tabel ini memiliki atribut-atribut, yaitu id informasi, judul informasi, tanggal mulai informasi, tanggal selesai informasi, detail informasi untuk foto atau gambar mengenai informasi, deskripsi informasi, dan jenis informasi. Penjelasan lebih lanjut akan dijelaskan dalam Tabel 3.32.

3.2.4.4 Tabel Jenis Info

Tabel Jenis Info adalah tabel yang digunakan untuk menyimpan data dari jenis-jenis informasi. Tabel ini mempunyai beberapa atribut yaitu id, dan nama jenis. Tabel ini akan dihubungkan dengan tabel Informasi. Penjelasan lebih lanjut akan dijelaskan dalam Tabel 3.33.

Tabel 3.32 Kamus Data Tabel Informasi

No.	Field	Type	Data Length	Deskripsi	Contoh
1	info_id	int	4	id untuk mengidentifikasi setiap informasi yang akan disimpan	1, 2, 3, ...
2	info_judul	varchar	35	Judul informasi	Opening Soon
3	info_tgl_mulai	date		tanggal mulai informasi ditampilkan	2018-05-18
4	info_tgl_selesai	date		tanggal selesai informasi ditampilkan	2018-05-31
5	info_detail	varchar	100	path lokasi gambar dalam folder	assets/images/ ...
6	info_deskripsi	varchar	200	deskripsi penjelasan informasi	Pembukaan Toko Baru daerah....

Tabel 3.33 Kamus Data Tabel Jenis Info

No.	Field	Type	Data Length	Deskripsi	Contoh
1	jenis_id	int	2	id untuk mengidentifikasi setiap jenis untuk tabel informasi	
2	jenis_nama	varchar	20	Nama jenis	Promo, Aset, Pengumuman

Tabel 3.34 Kamus Data Tabel Produk

No.	Field	Type	Data Length	Deskripsi	Contoh
1	produk_id	int	4	id untuk mengidentifikasi setiap produk	1, 2, 3, ...
2	produk_nama	varchar	50	nama dari produk	Seho
3	produk_kategori	varchar	50	kategori produk	4
4	produk_detail	varchar	100	path lokasi gambar dalam folder	assets/images/...
5	produk_harga	int	5	harga produk	5000

3.2.4.5 Tabel Produk

Tabel Produk adalah tabel yang digunakan untuk menyimpan data dari produk yang akan dijual melalui aplikasi ini. Tabel ini memiliki beberapa atribut, yaitu id, nama, kategori, detail, dan harga. Penjelasan rinci akan dijelaskan dalam Tabel 3.34.

3.2.4.6 Tabel Pertanyaan

Tabel Pertanyaan adalah tabel yang digunakan untuk menyimpan data dari daftar pertanyaan seputar transaksi maupun produk yang dijual melalui aplikasi ini. Tabel ini memiliki atribut-atribut, yaitu id, tanggal, judul, email pengguna, isi, dan jawaban. Data dari tabel ini akan digunakan pada halaman forum tanya jawab untuk ditampilkan, dan pada halaman kontak ketika ingin mengirimkan pertanyaan. Penjelasan lebih lanjut akan dijelaskan dalam Tabel 3.35.

Tabel 3.35 Kamus Data Tabel Pertanyaan

No.	Field	Type	Data Length	Deskripsi	Contoh
1	pertanyaan_id	int	4	id untuk mengidentifikasi setiap pertanyaan	1, 2, ...
2	pertanyaan_tgl	datetime		tanggal pertanyaan diajukan	2018-05-18
3	pertanyaan_judul	varchar	50	judul pertanyaan	Lama kirim
4	pertanyaan_email	varchar	50	email pengguna yang mengajukan pertanyaan	hlm@gmail.com
5	pertanyaan_isi	varchar	255	isi dari pertanyaan	Berapa lama...?
6	jawaban	varchar	255	jawaban dari pertanyaan	Tergantung jasa kirim

3.2.4.7 Tabel Transaksi

Tabel Transaksi adalah tabel yang digunakan untuk menyimpan data dari daftar transaksi yang dilakukan pembeli maupun *reseller*. Tabel ini memiliki atribut-atribut, yaitu id transaksi, catatan, bukti transaksi, status pembayaran, konfirmasi pengiriman barang, serta resi pengiriman. Dalam tabel ini ada pula atribut dari tabel lain, yaitu tabel User. Penjelasan lebih lanjut akan dijelaskan secara rinci pada Tabel 3.36.

Tabel 3.36 Kamus Data Tabel Transaksi

No.	Field	Type	Data Length	Deskripsi	Contoh
1	transaksi_id	int	4	id untuk mengidentifikasi setiap transaksi yang dilakukan	1, 2, ...
2	transaksi_tgl	date		tanggal transaksi dilakukan	2018-05-18
3	transaksi_user	int	2	Id Pembeli atau <i>reseller</i> yang melakukan transaksi	2, 3, ...
4	transaksi_total	int	6	total harga dalam sekali transaksi	100000
5	catatan	varchar	200	detail transaksi	Bumbu dipisah
6	bukti_transaksi	varchar	100	<i>path</i> lokasi gambar dalam folder	assets/images/..
7	Status	int	2	Status telah dibayar	0 atau 1
8	Konfirmasi	int	2	Status telah dikirim	0 atau 1
9	Resi_kirim	varchar	200	<i>path</i> lokasi gambar dalam folder	assets/images/..

3.2.4.8 Tabel History Pembelian

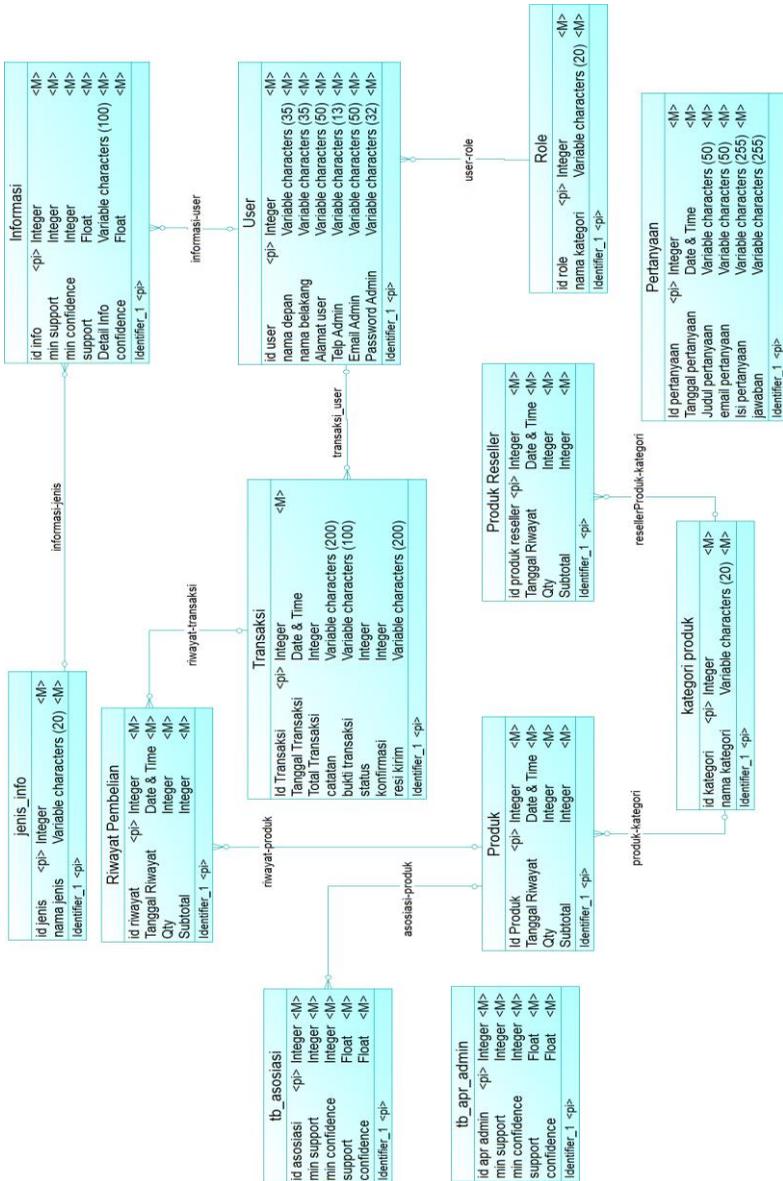
Tabel History Pembelian adalah tabel yang digunakan untuk menyimpan data perjalanan dari transaksi yang dilakukan oleh pembeli maupun *reseller*. Tabel ini memiliki atribut id history, id transaksi dari tabel Transaksi, id produk dari tabel produk, dan tanggal transaksi, banyaknya produk, subtotal untuk harga satuan masing-masing produk. Penjelasan lebih lanjut akan dijelaskan dalam Tabel 3.37.

Tabel 3.37 Kamus Data Tabel History Pembelian

No.	Field	Type	Data Length	Deskripsi	Contoh
1	history_id	int	4	id untuk mengidentifikasi history pembelian dari tabel transaksi	1,2,3, ...
2	history_transaksi	int	4	id transaksi yang digunakan	2
3	history_produk	int	4	id produk yang digunakan	2
4	history_tgl	varchar	20	status history pembelian	2018-05-18
5	Qty	int	4	Banyaknya produk	2
6	subtotal	int	5	Harga satuan dari produk	5000

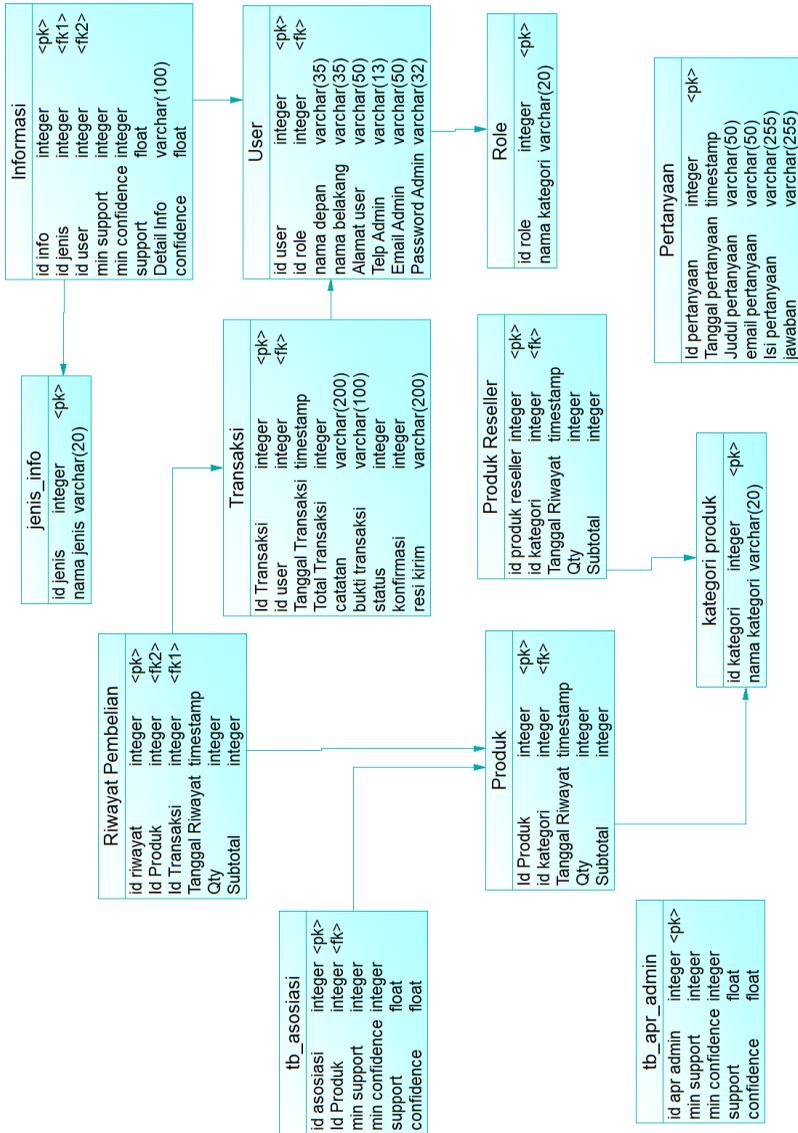
3.2.4.9 *Conceptual Data Modeling (CDM)*

Penjelasan mengenai hubungan basis data antar tabel dari perancangan data dapat dilihat pada Gambar 3.30 untuk diagram CDM dan pada Gambar 3.31 untuk diagram PDM.



Gambar 3.30 Conceptual Data Modeling

3.2.4.10 Physical Data Modeling (PDM)



Gambar 3.31 Physical Data Modeling

3.3. Perancangan Algoritma Apriori

Penting tidaknya sebuah aturan asosiasi (*association rule*) dapat diketahui dengan dua parameter, yaitu *support* (nilai penunjang), yaitu persentase kejadian kombinasi *item* yang muncul dalam sekumpulan transaksi, dan *confidence* (nilai kepastian), yaitu kuatnya hubungan antar *item* dalam aturan asosiasi.

Item yang dimaksudkan disini adalah produk yang dijual dalam aplikasi ini yaitu makaroni, usus kering, bakso goreng, dan lain-lain dengan label Mewek. Sedangkan *itemset* merupakan sekumpulan *item* dari suatu transaksi.

Kedua ukuran (*support* dan *confidence*) berguna dalam menentukan aturan asosiasi, yaitu untuk dibandingkan dengan batasan (*threshold*) yang ditentukan oleh pengguna (Administrator). Batasan tersebut umumnya terdiri atas minimum *support* sebagai batasan minimum dari nilai *support* dan minimum *confidence* sebagai batasan minimum dari nilai *confidence*.

Minimum *support* adalah perhitungan persentase dari seluruh transaksi, yang mana merupakan parameter yang diberikan kepada algoritma Apriori untuk memangkas aturan kandidat dengan menetapkan batas bawah untuk nilai *support* dari aturan asosiasi yang dihasilkan.

Minimum *confidence* adalah masukkan dari Administrator yang dalam perhitungannya direpresentasikan dalam bentuk persentase dari *item* dalam transaksi untuk menghitung seberapa sering *item* Y muncul dalam transaksi yang mengandung X.

Pada tahap membangkitkan *rule*, akan dibangkitkan lebih dahulu *candidate rule*. *Candidate rule* berisi semua kemungkinan *rule* yang memiliki *support* > *minimum support* karena masukkan *candidate rule* adalah *frequent item* (Fi). *Candidate rule* yang digunakan adalah *if x then y*, dimana X (sebab) adalah *antecedent* dan Y (akibat) sebagai *consequent*. Untuk *antecedent* boleh lebih dari satu *item*, sedangkan untuk *consequent* terdiri dari satu *item*.

Support dari suatu *association rule* adalah probabilitas dari *item* atau persentase kombinasi *item* A dan *item* B. Berdasarkan rumus dasar, didapatkan rumus untuk menghitung nilai *support* adalah sebagai berikut [13]:

$$\text{Support} = \frac{\sum \text{item yang dibeli sekaligus}}{\sum \text{jumlah seluruh transaksi}} \quad (3.1)$$

Untuk Σ *item* yang dibeli sekaligus (bisa lebih dari dua *item*) pada *If buy A then buy B*. Misal jumlah seluruh transaksi ada 5 transaksi. Dengan rincian pembelian sebagai berikut:

$T_1 = \{A, B, C\}$, $T_2 = \{B, C, D\}$, $T_3 = \{C, D, E, F\}$, $T_4 = \{A, B\}$, $T_5 = \{A, C, D, E\}$, dari data tersebut dapat diketahui nilai *support item* A adalah:

$$\text{Support}(A) = 3/5 \times 100\% = 60\%$$

Confidence adalah probabilitas kondisional, untuk aturan asosiasi $B|A$ didefinisikan seperti berikut:

$$\text{Confidence}(B|A) = \frac{\sum \text{item yang dibeli sekaligus}}{\sum \text{jumlah transaksi pada antecedent}} \quad (3.2)$$

Dengan menggunakan contoh data transaksi yang sama dengan sebelumnya, didapat transaksi yang membeli A terdapat 3 transaksi, sedangkan transaksi yang membeli *item* A dan B sekaligus ada 2 transaksi, sehingga nilai *confidence* $A \rightarrow B$ adalah:

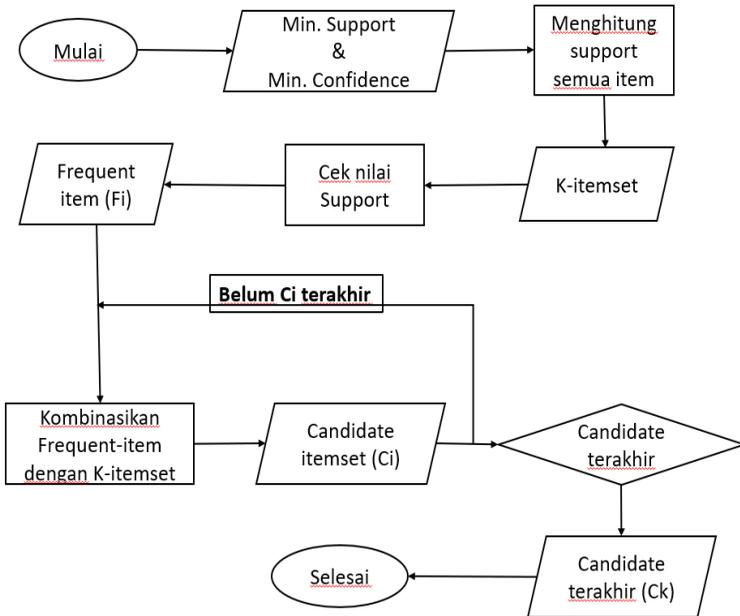
$$\text{Confidence}(B|A) = 2/3 \times 100\% = 66,67\%$$

Dalam aplikasi ini nantinya akan digunakan algoritma Apriori pada halaman riwayat produk untuk menentukan produk mana yang biasa dibeli bersamaan oleh pembeli dari data history pembelian. Algoritma Apriori membutuhkan beberapa hal dalam sistem, diantaranya adalah sebagai berikut:

1. Data untuk *testing* adalah data dari tabel history pembelian yang merupakan rincian dari tabel transaksi.
2. Dengan menggunakan algoritma Apriori akan dilakukan percobaan terhadap *itemset* yang terdapat dalam data history

- pembelian yaitu dengan membentuk kandidat-kandidat *itemset* atau biasa disebut dengan *k-itemset*.
3. Setiap *k-itemset* yang terbentuk sebelumnya kemudian dilakukan perhitungan nilai *support* untuk menghitung jumlah transaksi yang memuat semua *item* dalam *k-itemset* tersebut. *K-itemset* yang nilai *support*-nya lebih dari nilai minimum *support* yang disebut juga dengan *frequent item (Fi)*, kemudian dijadikan kandidat sebagai pembentukan aturan asosiasi (association rule).
 4. Dilakukan perhitungan confidence dari tiap *kitemset* untuk menentukan apakah kandidat tersebut dapat dijadikan sebagai aturan asosiasi (association rule) atau tidak.

Langkah-langkah perhitungan Apriori akan ditampilkan dalam bentuk flowchart pada Gambar 3.32 dimana langkah-langkah ini akan digunakan untuk skenario pengujian.



Gambar 3.32 Langkah-langkah perhitungan Apriori

[Halaman ini sengaja dikosongkan]

BAB IV IMPLEMENTASI

Bab ini membahas implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya. Sebelum penjelasan implementasi akan ditunjukkan terlebih dahulu lingkungan untuk melakukan implementasi.

Pada bagian implementasi ini juga akan dijelaskan mengenai fungsi-fungsi yang digunakan dalam program tugas akhir ini dan disertai dengan kode semu masing-masing fungsi utama.

4.1 Lingkungan Implementasi Perangkat Lunak

Spesifikasi perangkat keras serta perangkat lunak yang digunakan dalam tahap implementasi perangkat lunak tugas akhir ini seperti dijelaskan pada Tabel 4.1.

Tabel 4.1 Lingkungan Implementasi Perangkat Lunak

Perangkat Keras	Komputer	HP ProBook 450 G4
	Prosesor	Intel® Core™ i5-7200U CPU @ 2.50GHz 2.71GHz
	Memori Primer	8 GB
	Memori Sekunder	1 TB
Perangkat Lunak	Sistem Operasi	<i>Windows</i> 10 Home 64-bit
	Perangkat Lunak	ApacheFriends XAMPP Version 7.1.10 JetBrains PhpStorm 2017.3

4.2 Implementasi Antarmuka Pengguna

Implementasi antarmuka pengguna perangkat lunak berbasis *web*. Pada subbab ini akan dijelaskan dan ditampilkan tampilan halaman sesuai dengan rancangan antarmuka yang terdapat pada Bab 3. Penjelasan lebih rinci akan dijelaskan dalam subbab dibawah ini berdasarkan kasus penggunaan yang ada.

4.2.1 Implementasi Halaman Antarmuka Melihat Informasi

Halaman antarmuka melihat data informasi adalah halaman yang diakses ketika semua pengguna (pengunjung, Pembeli, *Reseller*, Administrator) membuka halaman *web* aplikasi. Dari halaman utama sudah menampilkan informasi berupa Pengumuman, dan Promo. Pengumuman dapat berupa acara atau kegiatan seperti bazar, *Flea Market*, atau semacamnya. Lalu dapat pula berupa kabar mengenai renovasi toko, buka toko di lokasi baru, dan lain-lain. Promo disini adalah pemberitahuan mengenai promo-promo yang sedang atau yang akan dilaksanakan, seperti promo tahun baru, promo bulan Ramadhan, promo ulang tahun, dan lain-lain. Pada menu halaman lain juga menampilkan informasi lainnya. Implementasi halaman antarmuka untuk melihat informasi dapat dilihat pada Gambar 4.1, dan fungsi yang digunakan untuk menampilkan halaman ini dapat dilihat pada Kode Semu 4.1. Untuk memanggil fungsi yang digunakan pertama-tama dilakukan inisialisasi variabel dengan nama data, digunakan untuk mengambil serta menyimpan sementara data yang diambil dari model untuk ditampilkan pada halaman *view*.

1	Data <- inisialisasi variabel mengambil data informasi dari <i>database</i> dengan model
2	Load <i>view</i> <- Menampilkan halaman data informasi

Kode Semu 4.1 Melihat Informasi

MAKARONI MEWEK
MAKARONI YANG BIKIN KAMU TERMEWEK-MEWEK

[Home](#)
[Menu](#)
[Kategori](#)
[Kontak](#)
[Sitemap](#)
[Daftar](#)

[Makaroni](#)
[Rp. 3000](#)
[Aneka dan Pilihan](#)
[FAQ](#)



Makaroni kering yang siap bikin kamu termewek-mewek

MAKARONI MEWEK Jogja

makaroni yang bikin kamu termewek-mewek

MAKARONI MEWEK Jogja

makaroni yang bikin kamu termewek-mewek

MAKARONI MEWEK Jogja

makaroni yang bikin kamu termewek-mewek



MAKARONI MEWEK Jogja

<p>Mewek Original Lv 1, 2, 3</p> <p>Regular 5 K Premium Package 8.5 K</p>	<p>Mewek Balado Mewek Keju Mewek BBQ Mewek Jaqung Manis Mewek Rumput Laut</p> <p>Regular 6 K Premium Package 9 K</p>	<p>Mewek Pizza Mewek Ayam Bakar Mewek Ayam Bawang Mewek Sapi Panggang</p> <p>Regular 7 K Premium Package 9.5 K</p>	
--	---	---	--

Available : Mie Kakek & Mie Nenek

Produk



Mie Nenek
Rp. 8.000



Mie Kakek
Rp. 8.000



Usus Goreng Reguler
Rp. 8.000



Usus Goreng Premium
Rp. 9.000



Usus Goreng Special
Rp. 11.000



Satek Holic
Rp. 10.000



Gado Gado
Rp. 8.000

Tentang Kami

Selamat datang di Makaroni Mewek. Kami adalah makaroni yang siap bikin kamu termewek-mewek. Kami adalah makaroni yang siap bikin kamu termewek-mewek. Kami adalah makaroni yang siap bikin kamu termewek-mewek.

Menu

Dendeng
Makaroni
Kakaroni
Aneka dan Pilihan
FAQ

Layanan Pengunjung

Bantuan
Daftar
Kontak

Akunmu

Menu
Daftar

MAKARONI MEWEK
MAKARONI YANG BIKIN KAMU TERMEWEK-MEWEK

[Facebook](#)
[Instagram](#)
[Twitter](#)

• Jalan Pamboran Sempati, Luragung Liris, Masing Sidiwa, Kota Magelang, Jawa Tengah 56132
+62 856-4313-3010
••• Hmawebmail@gmail.com

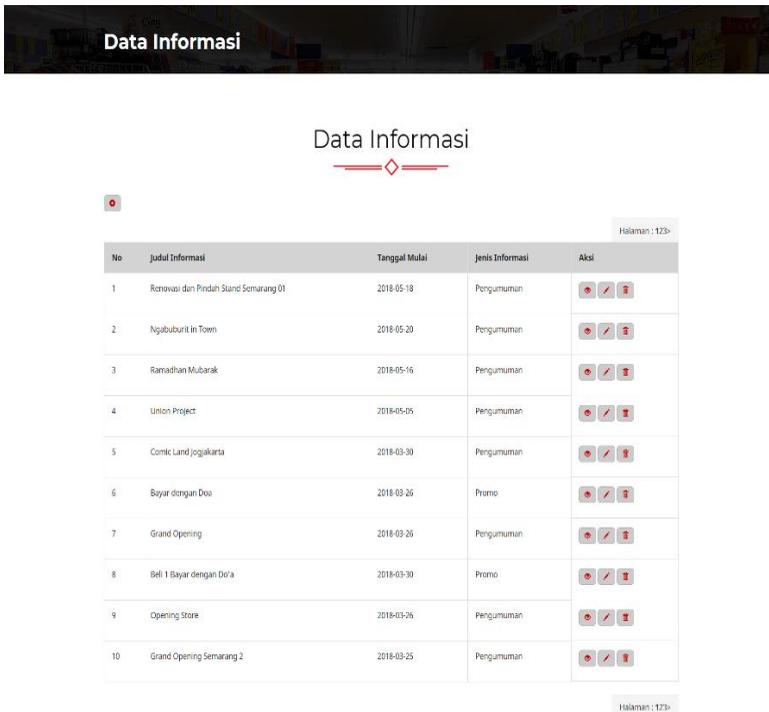
Copyright © 2018-2019 ME. 001144609078

Gambar 4.1 Antarmuka Melihat Data Informasi

4.2.2 Implementasi Halaman Antarmuka Mengelola Data Informasi

Halaman antarmuka mengelola data informasi adalah halaman yang diakses ketika Administrator ingin menambah, mengubah, melihat detail dan menghapus data informasi. Administrator harus *login* terlebih dahulu.

Pada Gambar 4.2 menampilkan halaman untuk melakukan kelola pada data informasi. Dalam halaman tersebut terdapat beberapa tombol untuk mengelola data, diantaranya adalah tombol *add* untuk menambah, tombol *eye* untuk melihat detail, tombol *edit* untuk mengubah data, dan tombol *trash* untuk menghapus data.



No	Judul Informasi	Tanggal Mulai	Jenis Informasi	Aksi
1	Retrovasi dan Pindah Stone Semarang 01	2018-05-18	Pengumuman	  
2	Ngabuburit In Town	2018-05-20	Pengumuman	  
3	Ramadhan Mubarak	2018-05-16	Pengumuman	  
4	Union Project	2018-05-05	Pengumuman	  
5	Comic Land jogjakarta	2018-03-30	Pengumuman	  
6	Bayar dengan Dora	2018-03-26	Promo	  
7	Grand Opening	2018-03-26	Pengumuman	  
8	Beli 1 Bayar dengan Dora	2018-03-30	Promo	  
9	Opening Store	2018-03-26	Pengumuman	  
10	Grand Opening Semarang 2	2018-03-25	Pengumuman	  

Gambar 4.2 Antarmuka Kelola Data Informasi

A white form titled "Tambah Informasi" with a red title. It contains several input fields: "Judul Informasi" (Nama Informasi), "Tanggal Mulai" (dd/mm/yyyy), "Tanggal Selesai" (dd/mm/yyyy), "Gambar Informasi" (Choose File, No file chosen), "Deskripsi" (Deskripsi Informasi), and "Jenis" (Pilih jenis informasi). A "Simpan" button is at the bottom.

Gambar 4.3 Antarmuka Menambah Data Informasi

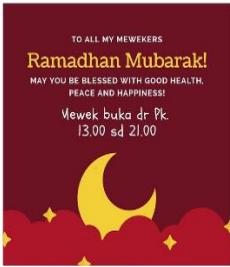
Kemudian dari halaman kelola data informasi, ketika Administrator memilih tombol *add* maka akan menampilkan halaman seperti pada Gambar 4.3 yang merupakan halaman antarmuka untuk menambah data informasi. Pada halaman menambah data informasi ini akan dibutuhkan fungsi untuk menambah data. Pada fungsi ini digunakan beberapa variabel, yaitu *filepath* untuk mengambil *path* atau nama lokasi dari *file* yang diunggah, *directory* untuk menginisialisasi lokasi baru dari *file* yang diunggah, *filename* untuk menyimpan *path* lokasi akhir dari *file* dengan nama *file*-nya, *extension* untuk menyimpan *filename* dengan *extension file*-nya, dan *namafile* untuk menyimpan *filepath* dengan *directory*. Kemudian dari variabel tersebut dimasukkan kedalam fungsi untuk menambah data informasi beserta mengunggah gambar untuk detail infonya, seperti yang dapat dilihat pada Kode Semu 4.2.

1	<p>If <code>move_upload_file</code> = kode unik untuk mengunggah <i>file</i> <code>dataInsert</code> <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari <i>view</i> <code>config</code> <- konfigurasi <i>file</i> yang akan diunggah <code>namadirektori</code> <- nama direktori <i>file</i> yang akan diunggah Memuat <i>library</i> untuk mengunggah <i>file</i> Menambah data pada tabel If berhasil menambah data Menampilkan halaman informasi Else Menampilkan halaman <i>form</i> tambah Else Menampilkan halaman <i>form</i> tambah</p>
---	---

Kode Semu 4.2 Menambah data informasi

Kemudian pada Gambar 4.4 merupakan halaman antarmuka untuk melihat detail data informasi. Untuk masuk halaman ini Administrator diharuskan memilih tombol lihat data pada halaman kelola data. Pada halaman ini digunakan fungsi untuk mengambil data secara detail berdasarkan id informasi yang dipilih dan variabel data untuk mengambil data yang kemudian ditampilkan. Fungsi ini dapat dilihat pada Kode Semu 4.3.

Detail Informasi

Detail Informasi						
No	Judul Informasi	Tanggal Mulai	Tanggal Selesai	Detail Informasi	Deskripsi Informasi	Jenis Informasi
1	Ramadhan Mubarak	2018-05-16	2018-05-18		Jadwal Buka Toko selama Bulan Ramadhan	Pengumuman

Gambar 4.4 Antarmuka Melihat Detail Data Informasi

1	<code>Id_info <- id informasi yang dipilih</code>
2	<code>Data <- data informasi untuk mengambil data dari <i>database</i></code>
3	<code>Load view <- Menampilkan halaman data informasi</code>

Kode Semu 4.3 Melihat detail data informasi



Ubah Informasi

Gambar 4.5 Antarmuka Mengubah Data Informasi

Lalu pada Gambar 4.5 merupakan halaman antarmuka untuk mengubah data informasi. Dengan memilih tombol *edit*, maka Administrator dapat masuk pada halaman ini. Pada halaman mengubah data informasi ini akan dibutuhkan beberapa fungsi yang kurang lebih mirip dengan fungsi menambah data, hanya saja fungsi ini dijalankan dengan berdasarkan id informasi yang dipilih, kemudian menampilkan data informasi yang telah ada. Fungsi ini dijelaskan pada Kode Semu 4.4.

1	<pre> If move_upload_file = kode unik untuk mengunggah file dataInsert <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari view config <- konfigurasi file yang akan diunggah namadirektori <- nama direktori file yang akan diunggah id info <- sesuai dengan id yang ingin diubah Memuat library untuk mengunggah file Menambah data pada tabel If berhasil menambah data Menampilkan halaman informasi Else Menampilkan halaman form ubah Else dataInsert <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari view config <- konfigurasi file yang akan diunggah namadirektori <- nama direktori file yang akan diunggah id info <- sesuai dengan id yang ingin diubah </pre>
---	--

Kode Semu 4.4 Mengubah data informasi

Kelola data informasi terakhir ada untuk menghapus data informasi yang dapat diakses ketika Administrator memilih tombol *trash* pada halaman kelola data seperti contoh pada Gambar 4.6 yaitu tampilan data sebelum dihapus, kemudian pada Gambar 4.7 setelah Administrator memilih menghapus data pada nomor tiga pada daftar data informasi. Untuk menghapus data dibutuhkan fungsi pada sistem, berdasarkan id informasi yang disimpan dalam variabel *id_info*, variabel data untuk menghapus data terpilih dari *database* yang kemudian mengambil data baru dari *database* untuk ditampilkan kembali pada halaman kelola data informasi. Fungsi menghapus data dapat dilihat pada Kode Semu 4.5.

1	<i>Id_info</i> <- id informasi yang dipilih
2	<i>Data</i> <- data informasi untuk mengambil data dari <i>database</i>
3	<i>Load view</i> <- Menampilkan halaman data informasi

Kode Semu 4.5 Menghapus data informasi

No	Judul Informasi	Tanggal Mulai	Jenis Informasi	Aksi
1	Logo	2017-01-01	Aset	  
2	Renovasi dan Pindah Stand Semarang 01	2018-05-18	Pengumuman	  
3	xxx	2018-05-07	Aset	  
4	Ngabuburit in Town	2018-05-20	Pengumuman	  

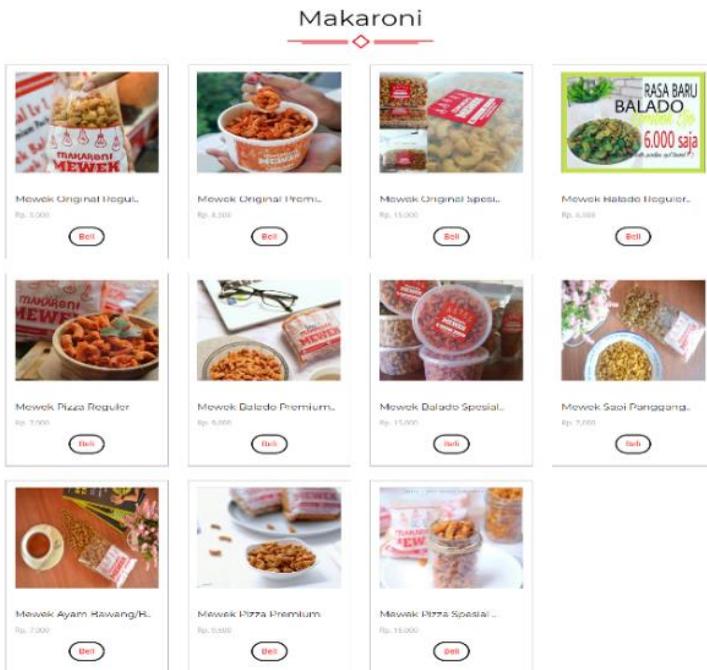
Gambar 4.6 Antarmuka sebelum Menghapus Data Informasi

No	Judul Informasi	Tanggal Mulai	Jenis Informasi	Aksi
1	Logo	2017-01-01	Aset	  
2	Renovasi dan Pindah Stand Semarang 01	2018-05-18	Pengumuman	  
3	Ngabuburit in Town	2018-05-20	Pengumuman	  
4	Ramadhan Mubarak	2018-05-16	Pengumuman	  

Gambar 4.7 Antarmuka sesudah Menghapus Data Informasi

4.2.3 Implementasi Halaman Antarmuka Melihat Produk

Halaman antarmuka melihat produk adalah halaman yang diakses ketika semua pengguna (pengunjung, Pembeli, *Reseller*, Administrator) memilih salah satu menu produk pada halaman *web* aplikasi. Menu yang dipilih seperti menu Makaroni, dan By Mewek yang terdapat *dropdown* menu produk lain yaitu menu Coklatini dan Jajanan. Hasil implementasi halaman antarmuka untuk melihat data produk dapat dilihat pada Gambar 4.8, sedangkan fungsi yang digunakan untuk menampilkan halaman ini dapat dilihat pada kode semu yang terlampir pada halaman 139. Untuk memanggil fungsi yang digunakan pertama-tama dilakukan inialisasi variabel dengan nama variabel data, digunakan untuk mengambil dan menyimpan data yang diambil dari model untuk ditampilkan pada halaman *view*.



Gambar 4.8 Antarmuka Melihat Data Produk

4.2.4 Implementasi Halaman Antarmuka Mengelola Data Produk

Halaman antarmuka mengelola data produk adalah halaman yang diakses ketika Administrator ingin melakukan kelola data produk seperti menambah, mengubah, melihat detail dan menghapus data produk. Dalam hal ini Administrator harus masuk ke dalam sistem terlebih dahulu dengan *login*. Pada Gambar 4.9 menampilkan halaman kelola data produk. Dalam halaman tersebut terdapat beberapa tombol untuk mengelola data, diantaranya adalah tombol *add* untuk menambah, tombol *eye* untuk melihat detail, tombol *edit* untuk mengubah data, dan tombol *trash* untuk menghapus data.

Data Produk

Data Produk

Halaman : <12>

No	Nama Produk	Kategori Produk	Harga Produk	Aksi
11	Mowek Pizzo Special Box	1	15000	  
12	Mie Iemok	3	8000	  
13	Mie Kakek	3	8000	  
14	Usus Goreng Regular	2	8000	  
15	Usus Goreng Premium	2	9500	  
16	Usus Goreng Special Box	2	15000	  
17	Oreo Coklatini	5	6000	  
18	Seblak Holic	4	10000	  
19	Makru Coklatini	5	6000	  
20	Cuskar Coklatini	5	5000	  

Halaman : <12>

Gambar 4.9 Antarmuka Kelola Data Produk

Pada Gambar 4.10 merupakan halaman *form* antarmuka untuk menambah data produk yang hanya dapat dilakukan setelah Administrator memilih tombol tambah pada halaman kelola data. Fungsi yang digunakan untuk menambah data produk kurang lebih sama dengan fungsi yang digunakan untuk menambah data informasi sebelumnya. Fungsi ini dapat dilihat pada kode semu halaman 139. Sedangkan pada Gambar 4.11 merupakan halaman *form* antarmuka untuk mengubah data produk, yang mana didapat setelah Administrator menekan tombol *edit* pada halaman kelola data produk. Fungsi yang digunakan untuk mengubah data produk pun sama dengan fungsi untuk mengubah data informasi sebelumnya. Fungsi mengubah data produk dijelaskan dalam kode semu yang telah terlampir pada halaman 140.

Tambah Produk

Nama Produk

Pilih Kategori Produk *

Info Produk

Choose File No file chosen

Harga

Harga Produk (cukup tulis dengan angka saja)

Submit

Gambar 4.10 Antarmuka Menambah Data Produk

Ubah Produk

Nama Produk

Kategori Produk

Info Produk



Choose File No file chosen

Harga Produk

Submit

Gambar 4.11 Antarmuka Mengubah Data Produk



Data Produk

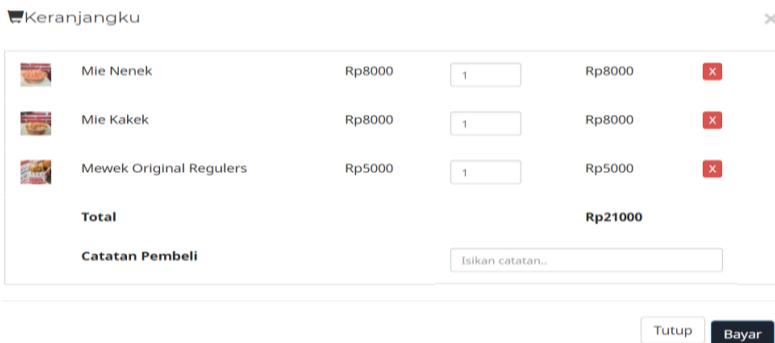
No	Nama Produk	Kategori Produk	Info Produk	Harga Produk
1	Bakso Goreng a.k.a Bawang	6		8000

Gambar 4.12 Antarmuka Melihat Detail Produk

Pada Gambar 4.12 merupakan halaman antarmuka untuk melihat detail dari data produk yang dipilih. Halaman ini dapat diakses ketika Administrator menekan tombol detail pada halaman kelola data produk. Fungsi untuk melihat detail produk yang dipilih telah dijelaskan dalam kode semu yang terlampir pada halaman 139, beserta kode semu untuk menghapus data produk.

4.2.5 Implementasi Halaman Antarmuka Melakukan Pemesanan Produk

Halaman antarmuka melakukan pemesanan produk diakses oleh Pembeli atau *Reseller* ketika ingin melakukan transaksi. Setelah Pembeli atau *Reseller* memilih produk maka produk yang dipilih selanjutnya akan langsung masuk Keranjang seperti pada Gambar 4.13, kemudian Pembeli atau *Reseller* dapat memilih tombol Tutup jika masih ingin memilih produk lain, dan memilih tombol Bayar ketika ingin melanjutkan pembelian dan melakukan konfirmasi pembelian seperti pada Gambar 4.14 untuk mengecek kembali pembelian yang baru saja dilakukan. Fungsi yang digunakan untuk melakukan pemesanan dan konfirmasi pembelian dapat dilihat pada Kode Semu 4.6.



Gambar 4.13 Antarmuka Melakukan Pemesanan Produk



Gambar 4.14 Antarmuka Konfirmasi Pembelian

1	<pre>dataInsert <- array data transaksi yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari view insert <- fungsi memasukkan dataInsert pada tabel transaksi</pre>
2	<pre>id_trx <- inialisasi id transaksi untuk menambahkan data transaksi dalam database</pre>
3	<pre>id_prod <- inialisasi id produk yang dipilih untuk ditambahkan pada tabel history pembelian dalam database</pre>
4	<pre>qty <- inialisasi banyaknya produk yang diambil untuk dimasukkan pada tabel history pembelian dalam database</pre>
5	<pre>subtotal <- inialisasi harga per produk yang dibeli untuk dimasukkan pada tabel history pembelian dalam database</pre>
6	<pre>For (i=0 sampai total per id_prod) dataInsert2 <- array data riwayat pembelian yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapat dari tabel transaksi insert2 <- fungsi memasukkan dataInsert2 pada tabel history pembelian</pre>
7	<pre>If session = null Menampilkan halaman login Else Get <- Mengambil id transaksi dari tabel transaksi Row <- Menyimpan id transaksi If (id_user = null and id transaksi > 0) or status = 0 Data_trx <- menampilkan data dari tabel transaksi Data_hsty <- menampilkan data dari tabel history Menampilkan halaman pembayaran beserta data yang didapat dari Data_trx dan Data_hsty Else Menampilkan peringatan error</pre>

Kode Semu 4.6 Melakukan pemesanan dan konfirmasi pembelian

4.2.6 Implementasi Halaman Antarmuka Melihat Riwayat Transaksi

Riwayat Transaksi



No	Tanggal Beli	Catatan	Total	Status	Aksi
#1	08 Jun 2018		Rp. 5,000	Sudah diproses	Detail
#2	11 Jun 2018		Rp. 19,000	Sudah diproses	Detail
#3	29 Jun 2018		Rp. 21,000	Belum diproses	Detail Bayar Total

Gambar 4.15 Antarmuka Melihat Riwayat Transaksi

Halaman antarmuka melihat riwayat transaksi diakses oleh Pembeli atau *Reseller* yang telah terdaftar dan sudah *login* terlebih dahulu, kemudian ingin melihat riwayat transaksi dari pembelian yang sedang atau yang telah dilakukan. Halaman implementasi melihat riwayat transaksi dapat dilihat pada Gambar 4.15. Pada halaman ini dapat juga melihat detail pembelian, pembayaran, dan menghapus pembelian. Fungsi untuk melihat riwayat transaksi kurang lebih sama dengan fungsi melihat data informasi. Untuk lengkapnya telah dijelaskan dalam kode semu yang telah terlampir pada halaman 140.

4.2.7 Implementasi Halaman Antarmuka Melihat Detail Pembelian

Halaman antarmuka melihat detail pembelian diakses oleh Pembeli atau *Reseller* yang telah *login* terlebih dahulu, dan telah melakukan pemesanan sebelumnya. Halaman ini diakses melalui halaman Riwayat dengan memilih tombol Detil, maka akan muncul halaman detail pembelian dari data transaksi tertentu seperti yang dapat dilihat pada Gambar 4.16, dan fungsi yang digunakan terdapat dalam kode semu yang telah terlampir pada halaman 140.

Detail Transaksi

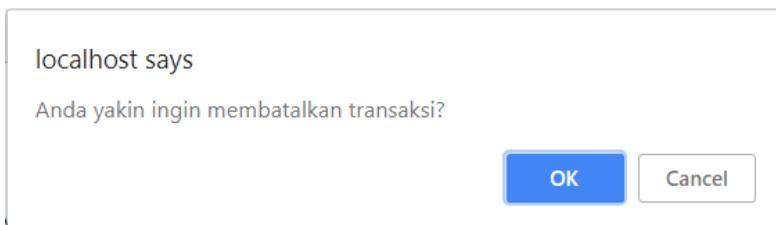
Detail Transaksi

No Transaksi	Tanggal Beli	Produk	Qty	Catatan	Subtotal	Total	Bukti Resi	Status
#46	23 Jun 2018	12	1		Rp. 8.000	Rp. 21.000		Segera upload bukti resi
#46	23 Jun 2018	13	1		Rp. 8.000	Rp. 21.000		Segera upload bukti resi
#46	23 Jun 2018	1	1		Rp. 5.000	Rp. 21.000		Segera upload bukti resi

Gambar 4.16 Antarmuka Melihat Detail Pembelian

4.2.8 Implementasi Halaman Antarmuka Menghapus Data Pembelian

Halaman antarmuka melihat detail pembelian diakses oleh Pembeli atau *Reseller* yang telah login terlebih dahulu, dan telah melakukan pemesanan sebelumnya. Administrator juga dapat mengakses fungsi ini untuk menghapus data pembelian yang telah lama tidak dibayar, sehingga dianggap batal melakukan pemesanan. Halaman ini diakses melalui halaman Riwayat dengan memilih tombol Hapus, maka akan muncul kotak peringatan untuk menghapus data pembelian seperti yang dapat dilihat pada Gambar 4.17. Fungsi yang digunakan untuk menghapus atau membatalkan data pembelian ini kurang lebih sama dengan menghapus data informasi, yaitu berdasarkan id transaksi yang dipilih. Untuk lebih lengkapnya telah dijelaskan dalam kode semu yang telah terlampir pada halaman 140.



Gambar 4.17 Antarmuka Menghapus Data Pembelian

4.2.9 Implementasi Halaman Antarmuka Melakukan Pembayaran

Upload Resi Pembayaran

Tanggal Transaksi:
23 Jun 2018

Total:
Rp. 21,000

Jenis Produk	Harga Satuan	Qty	Subtotal
 Mie Nenek	8000	1	Rp. 8.000
 Mie Kakak	8000	1	Rp. 8.000
 Mowek Original Regulars	5000	1	Rp. 5.000

Choose File: No file chosen

Upload

Cancel

Catatan:

Gambar 4.18 Antarmuka Melakukan Pembayaran

Pada Gambar 4.18 merupakan implementasi dari halaman antarmuka melakukan pembayaran. Halaman ini diakses oleh Pembeli atau *Reseller* yang telah terdaftar dan sudah *login* terlebih dahulu, kemudian ingin melakukan pembayaran setelah melakukan pemesanan produk. Setelah Pembeli atau *Reseller* mengonfirmasi pemesanan, maka akan muncul halaman Upload Resi Pembayaran, atau bisa juga melalui menu Riwayat kemudian memilih tombol Bayar pada salah satu daftar transaksi yang ada. Kemudian fungsi yang digunakan untuk melakukan pembayaran terdapat pada Kode Semu 4.7 dengan menggunakan inisialisasi variabel yang sama dengan ketika mengunggah gambar data informasi.

1	<pre> If move_upload_file = kode unik untuk mengunggah file dataInsert <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari view config <- konfigurasi file yang akan diunggah namadirektori <- nama direktori file yang akan diunggah id transaksi <- sesuai dengan id yang ingin diunggah Memuat library untuk mengunggah file Menambah data pada tabel Merubah status menjadi 1 Menampilkan halaman riwayat transaksi Else Menampilkan halaman riwayat transaksi </pre>
---	--

Kode Semu 4.7 Melakukan pembayaran

4.2.10 Implementasi Halaman Antarmuka Mengelola Data Transaksi

Data Transaksi*

— ◆ —

SELURUH DATA

Melihat list seluruh data transaksi yang barangnya belum dibayar

Halaman : 123-

No	No Transaksi	Nama Pembeli	Tanggal Transaksi	Total Harga	Catatan	Status Bayar	Aksi
1	#4	Hilma	2018-06-05 06:48:28	5000		0	 
2	#9	Raras	2018-06-08 12:52:29	5000		1	 
3	#10	Hilma	2018-06-08 17:56:34	22000	bumbu dipisah	0	 
4	#11	Hilma	2018-06-08 19:23:48	32500		0	 
5	#15	Himawan	2018-06-08 19:51:56	5000		0	 
6	#16	Himawan	2018-06-08 21:37:05	7000		0	 
7	#17	Himawan	2018-06-08 21:34:23	7000		0	 
8	#18	Himawan	2018-06-08 21:39:01	5000		1	 
9	#19	Himawan	2018-06-08 22:25:56	6000	bumbu dipisah	1	 
10	#21	Rani Aulia	2018-06-09 11:26:10	5000		1	 

Halaman : 123-

Gambar 4.19 Antarmuka Mengelola Data Transaksi

Pada Gambar 4.19 merupakan halaman antarmuka mengelola data transaksi yang hanya dapat diakses oleh Administrator ketika ingin mengunggah resi pengiriman dari transaksi yang dilakukan dengan Pembeli atau *Reseller* dengan memilih tombol *edit*. Kemudian akan muncul halaman seperti pada Gambar 4.20 untuk melihat detail. Setelah Administrator mengunggah resi pengiriman, maka status konfirmasi dari Administrator akan berubah menjadi 1 yang artinya barang telah dikirim ke Pembeli atau *Reseller*.

Detil Transaksi

No Transaksi	Tanggal Beli	Produk	Qty	Catatan	Subtotal	Total	Bukti Resi	Status	Konfirmasi
419	08 Jun 2018		1		Rp. 5,000	Rp. 5,000		Barang segera dikirim	Anda akan segera menetapkan resi pengiriman
X Hapus Choose File No file chosen									

Gambar 4.20 Antarmuka melihat detail dan mengunggah resi kirim

Fungsi yang digunakan untuk menampilkan detail transaksi dari data transaksi yang dipilih, mengunggah resi pengiriman data transaksi tersebut, serta menghapus data transaksi terdapat dalam kode semu yang terlampir pada halaman 141.

4.2.11. Implementasi Halaman Antarmuka Melihat Tanya Jawab

Halaman antarmuka melihat forum tanya jawab diakses ketika pengguna ingin memilih menu Forum pada bagian bawah halaman untuk Pengunjung, Pembeli, maupun *Reseller*, maka akan ditampilkan halaman antarmuka seperti pada Gambar 4.21. Sedangkan halaman antarmuka melihat tanya jawab untuk Administrator dapat diakses ketika memilih menu Data Forum. Halaman antarmuka ini akan menampilkan data pertanyaan dari pengguna lain serta jawaban dari Administrator seperti yang dapat dilihat pada Gambar 4.22.



Gambar 4.21 Antarmuka Melihat Tanya Jawab untuk Pengunjung, Pembeli, dan Reseller

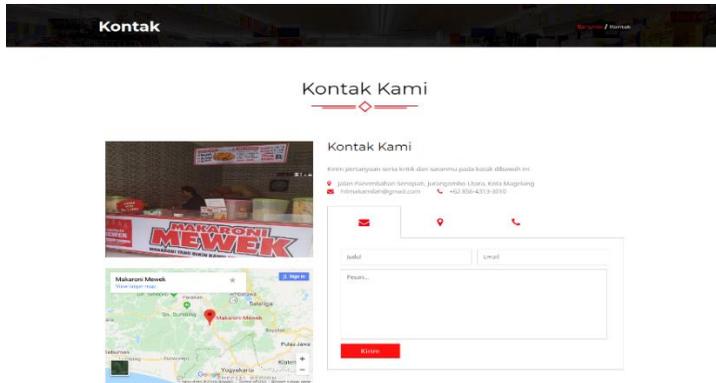


Gambar 4.22 Antarmuka Melihat Tanya Jawab untuk Administrator

Fungsi untuk menampilkan halaman forum tanya jawab telah dijelaskan dalam kode semu yang telah terlampir pada halaman 141.

4.2.12. Implementasi Halaman Antarmuka Mengirim Pertanyaan

Halaman antarmuka mengirim pertanyaan diakses ketika semua pengguna, baik pengguna yang telah memiliki akun maupun tidak, memilih menu Kontak pada bagian bawah halaman. Halaman akan menampilkan *form* seperti Gambar 4.23 untuk diisi oleh pengguna. Fungsi yang digunakan untuk mengirim pertanyaan terdapat dalam Kode Semu 4.8 dengan mengisi judul, email, dan isi pertanyaan yang kemudian data yang dimasukkan disimpan sementara dalam variabel terpisah, kemudian variabel-variabel tadi dimasukkan dalam *array* dataInsert.



Gambar 4.23 Antarmuka Mengirim Pertanyaan

1	dataInsert <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari <i>view</i>
2	If berhasil mengirim pertanyaan Menampilkan halaman forum tanya jawab Else Menampilkan halaman kontak

Kode Semu 4.8 Mengirim pertanyaan

4.2.13. Implementasi Halaman Antarmuka Menjawab Pertanyaan



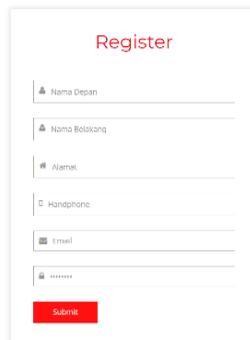
Gambar 4.24 Antarmuka Menjawab Pertanyaan

Halaman antarmuka menjawab pertanyaan diakses ketika Administrator memilih menu Data Forum. Seperti yang dapat

dilihat pada Gambar 4.22 terdapat dua tombol yaitu, *edit* dan *delete* pada setiap data pertanyaan. Untuk menjawab pertanyaan Administrator dapat memilih tombol *edit*, kemudian akan muncul *form* pada halaman Ubah Pertanyaan seperti pada Gambar 4.24, untuk mengubah data, Administrator diharuskan mengisi *form* jawaban kemudian memilih tombol *submit* untuk menyimpan kembali data pertanyaan yang telah diisi jawabannya. Fungsi yang digunakan untuk menjawab pertanyaan kurang lebih sama dengan fungsi untuk mengirim pertanyaan, hanya saja untuk menjawab pertanyaan dilakukan berdasarkan id pertanyaan yang dipilih. Fungsi ini telah dijelaskan dalam kode semu yang telah terlampir pada halaman 141.

4.2.14. Implementasi Halaman Antarmuka Menambah Pengguna Pembeli

Halaman antarmuka menambah pengguna sebagai Pembeli berisi *form* untuk menambah data pengguna sebagai Pembeli. Halaman ini diakses ketika Pengunjung memilih menu Daftar pada bagian atas halaman. Form yang muncul seperti pada Gambar 4.25. Fungsi untuk menambah data pengguna sebagai pembeli dijelaskan pada Kode Semu 4.9 dengan melakukan inisialisasi variabel terlebih dahulu yang kemudian menggunakan fungsi *insert* dalam *array* untuk memasukkan variabel-variabel tadi dalam *database*.



The image shows a web form titled "Register" in red text. Below the title are six input fields, each with a small icon to its left: "Nama Depan" (with a person icon), "Nama Belakang" (with a person icon), "Alamat" (with a location pin icon), "Handphone" (with a phone icon), "Email" (with an envelope icon), and a password field (with a key icon). At the bottom of the form is a red button labeled "Submit".

Gambar 4.25 Antarmuka Menambah Pengguna Pembeli

1	<code>nmDpn <- inisialisasi untuk mengambil data nama depan pengguna</code>
2	<code>nmBlkg <- inisialisasi untuk mengambil data nama belakang pengguna</code>
3	<code>Alamat <- inisialisasi untuk mengambil data alamat pengguna</code>
4	<code>Telp <- inisialisasi untuk mengambil data telp pengguna</code>
5	<code>email <- inisialisasi untuk mengambil data email pengguna</code>
6	<code>pass <- inisialisasi untuk mengambil data password pengguna</code>
7	<code>Role <- inisialisasi untuk memasukkan pengguna dalam role Pembeli</code>
8	<code>dataInsert <- inisialisasi untuk menyimpan data array yang berisi atribut tabel yang akan ditambah beserta nilai yang didapatkan dari view</code>
9	<code>If berhasil menambah pengguna Pembeli Menampilkan halaman untuk login Else Menampilkan halaman Daftar</code>

Kode Semu 4.9 Menambah pengguna sebagai Pembeli

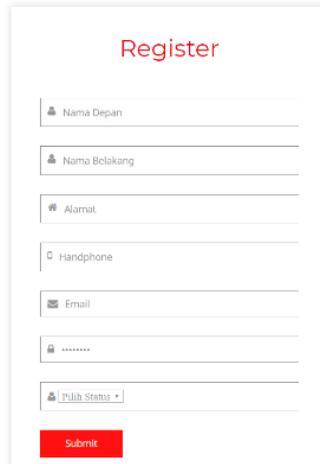
4.2.15. Implementasi Halaman Antarmuka Menambah Pengguna sebagai Administrator/Reseller

Halaman antarmuka menambah pengguna Administrator atau *Reseller* berisi *form* untuk menambah data pengguna. Halaman ini hanya dapat diakses ketika Administrator telah *login*, dan memilih menu Daftar. Form yang muncul dapat dilihat pada Gambar 4.26. Fungsi yang digunakan sama dengan menambah Pembeli yang dijelaskan dalam kode semu yang terlampir pada halaman 142.

4.2.16. Implementasi Halaman Antarmuka Melihat Profil

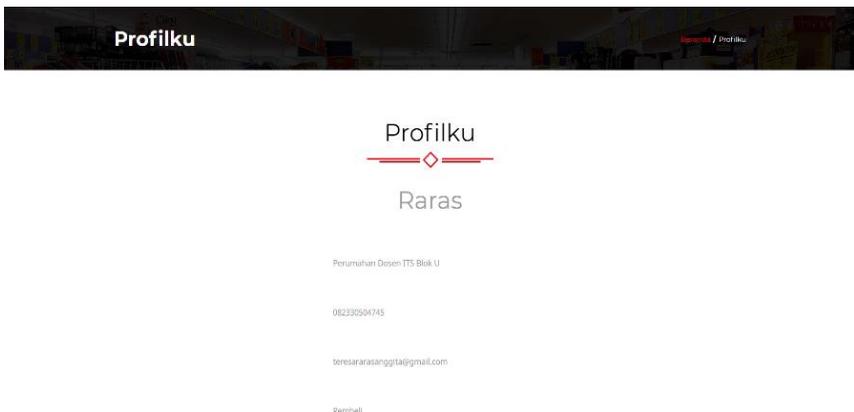
Halaman antarmuka melihat profil akan ditampilkan setelah pengguna yang telah terdaftar berhasil *login* dan memilih menu Profil yang ada dibagian atas atau bawah halaman. Halaman ini berisi detail informasi pengguna yang sedang *login* seperti yang dapat dilihat pada Gambar 4.27. Untuk menampilkan halaman ini

digunakan fungsi yang sama seperti fungsi melihat data informasi, namun pada halaman ini digunakan inialisasi id user terlebih dahulu untuk mengambil data dari pengguna yang sedang *login*. Fungsi ini dijelaskan pada kode semu yang telah terlampir pada halaman 142.



The image shows a registration form titled "Register" in red text. The form contains several input fields, each with a small icon to its left: "Nama Depan" (with a person icon), "Nama Belakang" (with a person icon), "Alamat" (with a house icon), "Handphone" (with a phone icon), "Email" (with an envelope icon), and a password field (with a lock icon). Below the password field is a dropdown menu labeled "Pilih Status". At the bottom of the form is a red "Submit" button.

Gambar 4.26 Antarmuka Menambah Pengguna Administrator/*Reseller*



Gambar 4.27 Antarmuka Melihat Profil Pengguna

4.2.17. Implementasi Halaman Antarmuka Melihat Riwayat Produk

Halaman antarmuka melihat riwayat produk ini diakses ketika Administrator memilih menu Riwayat Produk seperti pada Gambar 4.28. Kemudian pada implementasi halaman ini dilakukan perhitungan Apriori oleh sistem dengan memasukkan nilai minimum *support* dan nilai minimum *confidence* dan memilih tombol proses. Selanjutnya akan ditampilkan halaman mengenai produk yang berkaitan yang paling sering dibeli berdasarkan data riwayat pembelian berdasarkan dengan perhitungan algoritma apriori. Hasil dari perhitungan berdasarkan input yang dilakukan dapat menampilkan hasil seperti yang dapat dilihat pada Gambar 4.29 dengan rincian transaksinya terdapat pada Gambar 4.30.

Untuk pengguna pembeli atau *reseller*, halaman melihat rekomendasi produk dapat dilihat pada halaman beranda dan juga akan muncul ketika pembeli melakukan konfirmasi pembelian transaksi yang dapat dilihat pada Gambar 4.31.

Riwayat Produk

—  —

Min support * %

Min confidence * %

[Proses](#)

Gambar 4.28 Antarmuka Halaman Riwayat Produk

Hasil Aturan Apriori				
No	Rule	Support	Confidence	
1	Jika konsumen membeli Mewek Balado Reguler maka membeli Mewek Original Regulers	0.21	6/6	1
2	Jika konsumen membeli Mewek Original Regulers maka membeli Mewek Balado Reguler	0.21	6/13	0.46

Gambar 4.29 Hasil Perhitungan Algoritma Apriori

C adalah produk yang dibeli
T adalah transaksi yang dilakukan

Sekelompok Item																	
Transaksi	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17
T4	☒																
T9	☒																
T10					☒	☒											
T11				☒				☒		☒	☒						
T15	☒																
T16				☒													
T17				☒													
T18	☒																
T19															☒		
T21	☒																
T23																☒	
T24													☒				
T26																☒	☒
T27		☒			☒												
T32	☒	☒	☒	☒													
T33								☒	☒	☒							
T34	☒			☒													
T35	☒			☒	☒												
T36	☒			☒	☒											☒	
T37	☒			☒								☒					
T38						☒	☒										
T39												☒	☒		☒		
T40					☒											☒	
T41					☒								☒			☒	☒
T42	☒	☒		☒													☒
T43												☒	☒			☒	
T44	☒					☒	☒	☒									
T45	☒	☒	☒														

Gambar 4.30 Antarmuka Hasil Perhitungan Apriori

Total:
Rp. 15,000

Jenis Produk	Harga Satuan	Qty	Subtotal
 Mewek Ori Regulars	5000	1	Rp. 5,000
 Seblak Hollic	10000	1	Rp. 10,000

Relasi Produk



Gambar 4.31 Halaman Relasi Produk Pembeli

4.3. Implementasi Algoritma Apriori

Untuk menampilkan halaman hasil aturan asosiasi dari algoritma Apriori dilakukan perhitungan terlebih dahulu dengan seperti yang dapat dilihat pada Kode Semu 4.10, yang mana terdapat inisialisasi terlebih dahulu untuk masukkan nilai minimum *support* dan nilai minimum *confidence* kemudian disimpan sementara dalam variabel *min_support* dan *min_confidence*. Selanjutnya produk-produk yang ada dihitung nilai *support* dan nilai *confidence* dari setiap produk dalam suatu transaksi dan diambil data produk yang memiliki nilai *support* lebih dari minimum *support* dan minimum *confidence* sehingga dapat dikatakan memenuhi syarat dari hasil aturan asosiasi. Setelah itu nama produk yang memenuhi syarat tersebut disimpan dalam variabel *get_apr_name* yang kemudian digunakan untuk ditampilkan pada halaman *view*.

1	Min_support <- masukkan untuk batas nilai minimal kombinasi produk yang akan ditampilkan pada halaman <i>view</i>
2	Min_confidence <- masukkan untuk batas nilai minimal dari hubungan kemunculan antar produk yang akan ditampilkan pada halaman <i>view</i>
3	Get_apr_name <- mengambil nama produk dari perhitungan Apriori yang memenuhi syarat kemudian ditampilkan pada halaman <i>view</i>
4	Support <- (produk a + produk b + ... + produk n) transaksi / total transaksi
5	Confidence <- (produk a + produk b + ... + produk n) transaksi / jumlah transaksi produk a

Kode Semu 4.10 Perhitungan Apriori

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas uji coba dan evaluasi terhadap perangkat lunak yang telah dikembangkan dari implementasi aplikasi industri rumah tangga makanan ringan berbasis *web*.

5.1. Lingkungan Pengujian

Lingkungan pengujian sistem pada pengerjaan Tugas Akhir ini dilakukan seperti dijelaskan pada Tabel 5.1:

Tabel 5.1 Lingkungan Pengujian Fungsionalitas Perangkat Lunak

Spesifikasi	Deskripsi
Jenis Perangkat	Laptop/Komputer
Sistem Operasi	Windows 10 Home
Memori Internal	8 GB
RAM	1 GB
Jaringan	Akses Internet

5.2. Pengujian Fungsionalitas

Pengujian fungsionalitas ini adalah pengujian fungsi-fungsi yang berjalan pada aplikasi berdasarkan kasus penggunaan. Pada pengujian ini akan dibagi berdasarkan pengguna, yaitu pembeli atau *reseller* dan Administrator. Pengujian fungsionalitas ini dilakukan dengan membuat skenario pengujian terlebih dahulu untuk menjadi arahan bagi penguji dalam melakukan uji coba, kemudian memberikan tanggapan berdasarkan hasil uji yang telah dilakukan.

5.2.1. Skenario Pengujian Fungsionalitas Pembeli/*Reseller*

Skenario pengujian merupakan tahapan-tahapan untuk penguji dalam melakukan uji coba aplikasi yang dilakukan berdasarkan kasus penggunaan untuk aktor pembeli atau *reseller*.

A. Skenario Pengujian Melihat Informasi

Pada pengujian SCF-001, proses untuk melihat informasi dapat dilakukan oleh semua pengguna, baik pengguna terdaftar maupun yang tidak terdaftar. Melihat data informasi yang dimaksudkan disini, sistem dapat menampilkan semua data yang ada, seperti data pengumuman, promo, maupun aset informasi dari aplikasi itu sendiri. Proses ini akan dijelaskan pada Tabel 5.2.

Tabel 5.2 Skenario 1 Pengujian Melihat Informasi

No. Pengujian	SCF-001
Skenario Pengujian	Melihat data informasi
Kondisi Awal	Data informasi telah tersimpan
Langkah-langkah	1. Penguji membuka halaman <i>web</i> aplikasi
Hasil yang diharapkan	Aplikasi dapat menampilkan data informasi
Hasil yang diperoleh	Data informasi ditampilkan
Hasil Pengujian	Berhasil

B. Skenario Pengujian Melihat Produk

Pada proses pengujian SCF-002 merupakan proses untuk melihat produk yang dapat dilakukan oleh semua pengguna. Melihat produk yang dimaksudkan disini adalah sistem dapat menampilkan semua produk yang ada seperti, Makaroni, Jajanan, dan Coklatini. Proses ini akan dijelaskan pada Tabel 5.3.

Tabel 5.3 Skenario 2 Pengujian Melihat Produk

No. Pengujian	SCF-002
Skenario Pengujian	Melihat data produk
Kondisi Awal	Data produk telah tersimpan
Langkah-langkah	1. Penguji membuka halaman aplikasi 2. Penguji memilih salah satu menu seperti, Makaroni atau By Mewek
Hasil yang diharapkan	Aplikasi dapat menampilkan data produk
Hasil yang diperoleh	Data produk ditampilkan
Hasil Pengujian	Berhasil

C. Skenario Pengujian Melakukan Pemesanan Produk

Pada pengujian SCF-003 merupakan proses melakukan pemesanan yang hanya dapat dilakukan oleh pengguna sebagai Pembeli atau *Reseller*. Proses ini akan dijelaskan pada Tabel 5.4.

Tabel 5.4 Skenario 3 Pengujian Melakukan Pemesanan

No. Pengujian	SCF-003
Skenario Pengujian	Melakukan Pemesanan Produk
Kondisi Awal	Data pesanan baru belum tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Pembeli 2. Penguji memilih produk dan klik “Beli” 3. Penguji memilih <i>icon</i> keranjang pada kanan atas 4. Penguji memilih tombol <i>Bayar</i> pada kotak keranjang belanja 5. Penguji melakukan konfirmasi pemesanan
Hasil yang diharapkan	Aplikasi dapat menyimpan data pesanan pada tabel transaksi
Hasil yang diperoleh	Data pesanan telah masuk <i>database</i>
Hasil Pengujian	Berhasil

D. Skenario Pengujian Melihat Riwayat Transaksi

Pada pengujian SCF-004 merupakan proses melihat riwayat transaksi yang hanya dapat dilakukan oleh Pembeli dan *Reseller*. Proses ini dilakukan ketika pengguna memilih tombol “Riwayat” dan akan dijelaskan pada Tabel 5.5.

Tabel 5.5 Skenario 4 Pengujian Melihat Riwayat Transaksi

No. Pengujian	SCF-004
Skenario Pengujian	Melihat riwayat transaksi
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Pembeli atau <i>Reseller</i> 2. Penguji memilih tombol “Riwayat” untuk melihat riwayat transaksi
Hasil yang diharapkan	Aplikasi dapat menampilkan data transaksi
Hasil yang diperoleh	Data riwayat transaksi ditampilkan
Hasil Pengujian	Berhasil

E. Skenario Pengujian Melihat Detail Pembelian

Pengujian SCF-005 merupakan proses melihat detail pembelian setelah Pembeli dan *Reseller* memilih *detail* pada halaman riwayat transaksi. Proses akan dijelaskan pada Tabel 5.6.

Tabel 5.6 Skenario 5 Pengujian Melihat Detail Pembelian

No. Pengujian	SCF-005
Skenario Pengujian	Melihat Detail Pembelian
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Pembeli atau <i>Reseller</i> 2. Penguji memilih tombol “Riwayat” untuk melihat riwayat transaksi 3. Penguji memilih tombol “Detail”
Hasil yang diharapkan	Aplikasi dapat menampilkan detail data transaksi
Hasil yang diperoleh	Detail data transaksi ditampilkan
Hasil Pengujian	Berhasil

F. Skenario Pengujian Menghapus Data Pembelian

Pengujian SCF-006 merupakan proses menghapus data pembelian setelah Pembeli atau *Reseller* memilih tombol *hapus* pada salah satu data pada halaman riwayat transaksi. Proses ini akan dijelaskan pada Tabel 5.7.

Tabel 5.7 Skenario 6 Pengujian Menghapus Data Pembelian

No. Pengujian	SCF-006
Skenario Pengujian	Menghapus Data Pembelian
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Pembeli atau <i>Reseller</i> 2. Penguji memilih tombol “Riwayat” untuk melihat riwayat transaksi 3. Penguji memilih tombol “Hapus”
Hasil yang diharapkan	Aplikasi dapat memunculkan <i>pop-up</i> menghapus data
Hasil yang diperoleh	Data transaksi dihapus dan memunculkan <i>pop-up</i> menghapus data
Hasil Pengujian	Berhasil

G. Skenario Pengujian Melakukan Pembayaran

Proses pengujian melakukan pembayaran ini dilakukan ketika pengguna memilih tombol “Bayar” pada halaman riwayat transaksi. Pengguna untuk proses ini adalah Pembeli dan *Reseller*. Proses ketika pengguna melakukan pembayaran ketika telah mengunggah gambar bukti pembayaran dan berhasil memunculkan pemberitahuan (*alert message*) telah berhasil melakukan pembayaran akan dijelaskan pada Tabel 5.8. Sedangkan proses ketika pengguna melakukan pembayaran ketika tidak mengunggah gambar bukti pembayaran dan berhasil memunculkan *pop-up* gagal melakukan pembayaran akan dijelaskan pada Tabel 5.9.

Tabel 5.8 Skenario 7 Pengujian Melakukan Pembayaran

No. Pengujian	SCF-007
Skenario Pengujian	Melakukan Pembayaran ketika telah berhasil mengunggah gambar bukti pembayaran
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Pembeli atau <i>Reseller</i> 2. Penguji memilih tombol “Riwayat” untuk melihat riwayat transaksi 3. Penguji memilih tombol “Bayar” 4. Penguji mengunggah gambar resi pembayaran 5. Penguji memilih tombol “Bayar” pada halaman pembayaran
Hasil yang diharapkan	Aplikasi dapat menyimpan data dan memunculkan pemberitahuan telah berhasil melakukan pembayaran
Hasil yang diperoleh	Data transaksi ter- <i>update</i> dan status pembayaran berubah
Hasil Pengujian	Berhasil

Tabel 5.9 Skenario 8 Pengujian Melakukan Pembayaran

No. Pengujian	SCF-008
Skenario Pengujian	Melakukan Pembayaran ketika telah gagal mengunggah gambar bukti pembayaran
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Pembeli atau <i>Reseller</i> 2. Penguji memilih tombol “Riwayat” untuk melihat riwayat transaksi 3. Penguji memilih tombol “Bayar” 4. Penguji tidak mengunggah gambar resi pembayaran 5. Penguji memilih tombol “Bayar” pada halaman pembayaran
Hasil yang diharapkan	Aplikasi dapat memunculkan pemberitahuan telah gagal melakukan pembayaran
Hasil yang diperoleh	Data transaksi tidak <i>ter-update</i> dan status pembayaran tetap
Hasil Pengujian	Berhasil

H. Skenario Pengujian Melihat Forum Tanya Jawab

Pengujian fungsionalitas SCF-009 melihat forum tanya jawab dilakukan ketika pengguna memilih menu Forum pada bagian bawah halaman. Proses ini akan dijelaskan pada Tabel 5.10.

Tabel 5.10 Skenario 9 Pengujian Melihat Forum Tanya Jawab

No. Pengujian	SCF-009
Skenario Pengujian	Melihat forum tanya jawab
Kondisi Awal	Data telah tersimpan
Langkah-langkah	1. Penguji memilih menu forum tanya jawab
Hasil yang diharapkan	Aplikasi dapat menampilkan data forum tanya jawab
Hasil yang diperoleh	Data daftar tanya jawab ditampilkan
Hasil Pengujian	Berhasil

I. Skenario Pengujian Mengirim Pertanyaan

Pengujian fungsionalitas SCF-010 mengirim pertanyaan dilakukan ketika pengguna memilih Kontak dan mengisi form pertanyaan pada halaman. Proses ini dijelaskan pada Tabel 5.11.

Tabel 5.11 Skenario 10 Pengujian Mengirim Pertanyaan

No. Pengujian	SCF-010
Skenario Pengujian	Mengirim pertanyaan
Kondisi Awal	Data pertanyaan baru belum tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji memilih menu Kontak 2. Penguji mengisi form pertanyaan 3. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan data pertanyaan baru
Hasil yang diperoleh	Data pertanyaan ditampilkan pada halaman forum
Hasil Pengujian	Berhasil

J. Skenario Pengujian Menambah Pengguna Pembeli

Pengujian fungsionalitas SCF-011 menambah pengguna pembeli dilakukan ketika memilih menu Daftar. Proses ini akan dijelaskan pada Tabel 5.12.

Tabel 5.12 Skenario 11 Pengujian Menambah Pengguna Pembeli

No. Pengujian	SCF-011
Skenario Pengujian	Menambah data pengguna Pembeli
Kondisi Awal	Data pengguna Pembeli baru belum tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji memilih tombol <i>Daftar</i> untuk menambah data pengguna sebagai Pembeli 2. Penguji mengisi semua isian form 3. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan dan menampilkan data pengguna Pembeli yang baru
Hasil yang diperoleh	Data Pembeli baru ditampilkan
Hasil Pengujian	Berhasil

K. Skenario Pengujian Melihat Profil Pengguna

Pengujian fungsionalitas SCF-012 melihat profil pengguna dilakukan ketika memilih menu Profil. Proses ini akan dijelaskan pada Tabel 5.13

Tabel 5.13 Skenario 12 Pengujian Melihat Profil Pengguna

No. Pengujian	SCF-012
Skenario Pengujian	Melihat detail profil pengguna
Kondisi Awal	Data pengguna telah tersimpan
Langkah-langkah	1. Penguji login terlebih dahulu 2. Penguji masuk ke halaman profil dengan memilih menu “Profil”
Hasil yang diharapkan	Aplikasi dapat menampilkan data profil pengguna
Hasil yang diperoleh	Data profil pengguna ditampilkan
Hasil Pengujian	Berhasil

5.2.2. Hasil Pengujian Fungsionalitas Pembeli/*Reseller*

Berikut hasil pengujian fungsionalitas dari uji coba untuk aktor pembeli atau *reseller*, serta hasil kuisioner mengenai terpenuhi atau tidaknya seluruh fitur.

A. Hasil Pengujian Melihat Informasi

Hasil pengujian dari SCF-001 melihat informasi akan ditampilkan pada halaman *web* aplikasi seperti pada Gambar 5.1.

B. Hasil Pengujian Melihat Produk

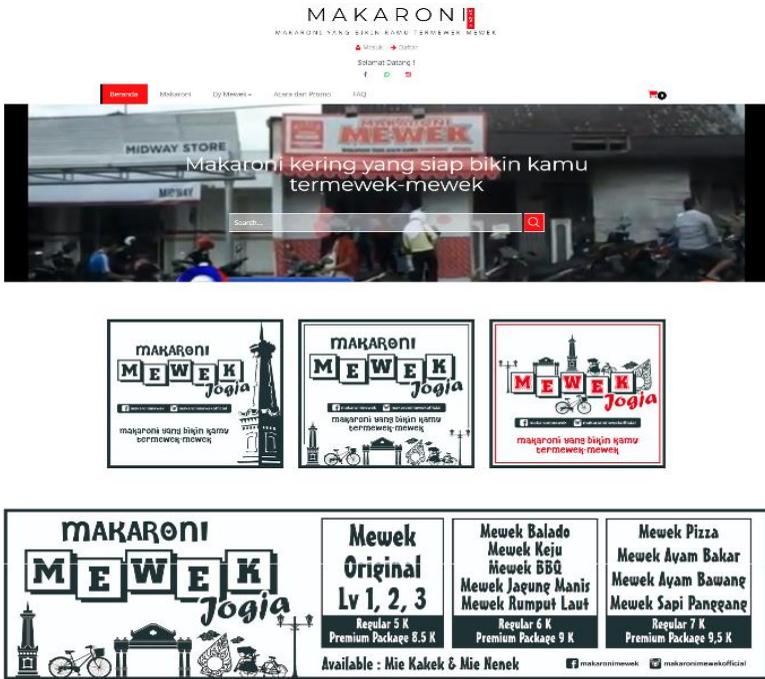
Hasil pengujian SCF-002 melihat produk akan ditampilkan halaman produk seperti pada Gambar 5.2.

C. Hasil Pengujian Melakukan Pemesanan Produk

Hasil pengujian SCF-003 akan ditampilkan pada kotak keranjang seperti pada Gambar 5.3. Kemudian setelah klik “Bayar” maka akan menampilkan halaman konfirmasi pembelian yang dapat dilihat bukti pengujiannya pada Gambar 5.4.

D. Hasil Pengujian Melihat Riwayat Transaksi

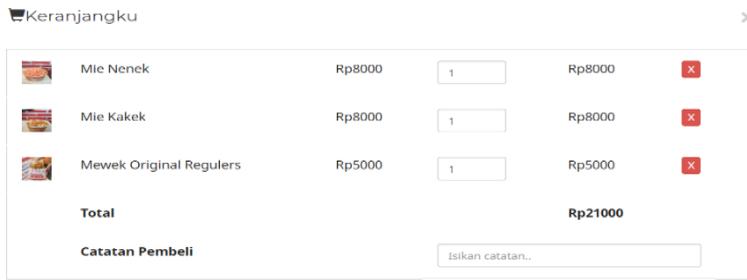
Hasil pengujian SCF-004 melihat riwayat transaksi akan ditampilkan seperti pada Gambar 5.5.



Gambar 5.1 Halaman Menu Utama



Gambar 5.2 Halaman Produk



Gambar 5.3 Kotak Keranjang Belanja

Konfirmasi Pembelian



Gambar 5.4 Halaman Konfirmasi Pembelian

Riwayat Transaksi

No	Tanggal Beli	Catatan	Total	Status	Aksi
#1	08 Jun 2018		Rp. 5,000	Sudah diproses	Detail
#2	11 Jun 2018		Rp. 19,000	Sudah diproses	Detail
#3	23 Jun 2018		Rp. 21,000	Sudah diproses	Detail Bayar Batal

Gambar 5.5 Halaman Riwayat Transaksi

E. Hasil Pengujian Melihat Detail Pembelian

Hasil pengujian SCF-005 akan ditampilkan pada halaman detail transaksi seperti pada Gambar 5.6.

Detail Transaksi

No Transaksi	Tanggal Beli	Produk	Qty	Catatan	Subtotal	Total	Bukti Resi	Status
#46	23 Jun 2018	12	1		Rp. 8,000	Rp. 21,000		Segera upload bukti resi
#46	23 Jun 2018	13	1		Rp. 8,000	Rp. 21,000		Segera upload bukti resi

Gambar 5.6 Halaman Detail Data Transaksi

F. Hasil Pengujian Menghapus Data Pembelian

Hasil pengujian SCF-006 akan ditampilkan kotak *pop-up* pada halaman riwayat transaksi ketika akan menghapus data transaksi seperti yang dapat dilihat pada Gambar 5.7.

localhost says

Anda yakin ingin membatalkan transaksi?

OK

Cancel

Gambar 5.7 Kotak *pop-up* menghapus data transaksi

G. Hasil Pengujian Melakukan Pembayaran

Hasil pengujian SCF-007 akan ditampilkan pemberitahuan (*alert message*) pada halaman riwayat transaksi ketika berhasil melakukan pembayaran seperti pada Gambar 5.8.

Riwayat Transaksi

Sukses! Rcsi pembayaran berhasil diunggah.

No	Tanggal Beli	Catatan	Total	Status	Aksi
#1	08 Jun 2018		Rp. 5,000	Batal	Detail Bayar Batal
#2	08 Jun 2018		Rp. 7,000	Batal	Detail Bayar Batal
#3	08 Jun 2018		Rp. 7,000	Batal	Detail Bayar Batal
#4	08 Jun 2018		Rp. 5,000	Sukses	Detail

Gambar 5.8 Halaman Pembayaran Berhasil

Proses ketika pengguna melakukan pembayaran ketika tidak mengunggah gambar bukti pembayaran dan berhasil memunculkan *pop-up* gagal melakukan pembayaran akan dijelaskan pada Tabel 5.9, dan hasil pengujian SCF-008 akan ditampilkan pemberitahuan (*alert message*) ketika gagal melakukan pembayaran seperti pada Gambar 5.9.



Gambar 5.9 Halaman Pembayaran Gagal

H. Hasil Pengujian Melihat Forum Tanya Jawab

Hasil pengujian fungsionalitas SCF-009 melihat forum tanya jawab akan ditampilkan seperti pada Gambar 5.10.

Forum Tanya Jawab

Pertanyaan: Saya ingin menjadi bagian dari Makaroni Mewek. Apakah saya bisa menjadi reseller dari Makaroni Mewek? Jika bisa, Bagaimana caranya?

Jawaban

Tentu Bisa! Silahkan kontak kami via Whatsapp, akan kami beritahu lebih lanjut mengenai alur menjadi Reseller. Terima Kasih!

Pertanyaan: Setelah saya selesai membayar dan mengunggah bukti resi, bagaimana saya tahu jika barang telah dikirim?

Jawaban

Jika bukti resi telah diunggah, Admin akan segera melakukan pengiriman barang dan mengunggah resi pengiriman yang dapat anda lihat pada halaman Riwayat.

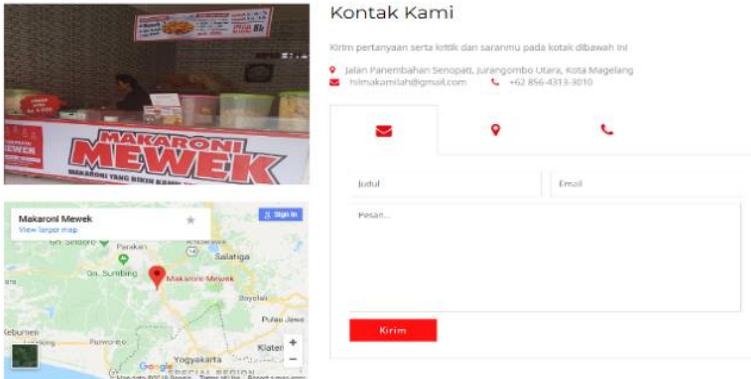
Terima Kasih!

Gambar 5.10 Halaman Forum Tanya Jawab

I. Hasil Pengujian Mengirim Pertanyaan

Hasil pengujian fungsionalitas SCF-010 mengirim pertanyaan akan ditampilkan seperti pada Gambar 5.11.

Kontak Kami



Gambar 5.11 Halaman Kontak

J. Hasil Pengujian Menambah Pengguna Pembeli

Hasil pengujian SCF-011 menambah pengguna sebagai Pembeli akan ditampilkan seperti pada Gambar 5.12.

Register

Gambar 5.12 Halaman Daftar sebagai Pembeli

K. Hasil Pengujian Melihat Profil Pengguna

Hasil pengujian SCF-012 melihat profil pengguna yang sedang *login* akan ditampilkan pada halaman daftar seperti pada Gambar 5.13.



Gambar 5.13 Halaman Profil

5.2.2.1. Hasil Kuisisioner Penguji Fungsionalitas

Survei ini dilakukan setelah penguji melakukan uji coba pada setiap skenario pengujian fungsionalitas, untuk memastikan semua fitur terpenuhi. Jika terpenuhi maka akan diberi nilai 1 dan jika tidak terpenuhi diberi nilai 0. Dari hasil survei terhadap 21 orang responden menyatakan bahwa semua fitur berhasil dijalankan, artinya 100% telah terpenuhi.

Tabel 5.14 Hasil Kuisisioner Pengujian Fungsionalitas

No.	Pernyataan	Terpenuhi	
		0	1
1	Melihat informasi	0	21
2	Melihat produk	0	21
3	Menambah akun sebagai Pembeli	0	21
4	Masuk (<i>login</i>) ke dalam sistem	0	21
5	Melihat profil	0	21
6	Melakukan pemesanan	0	21
7	Melihat riwayat transaksi	0	21
8	Melihat detail pemesanan	0	21
9	Menghapus/membatalkan pemesanan	0	21
10	Melakukan pembayaran	0	21
11	Melihat forum tanya jawab	0	21
12	Mengirim pertanyaan	0	21

5.2.3. Pengujian Fungsionalitas Administrator

Pengujian fungsionalitas Administrator ini dilakukan dengan cara pengujian mencoba aplikasi terlebih dahulu dengan skenario yang telah ditentukan dan berdasarkan kasus penggunaan untuk aktor Administrator.

5.2.3.1. Pengujian Fungsionalitas Mengelola Data Informasi

Pada pengujian fungsionalitas mengelola data informasi dapat berupa proses untuk menambah data informasi baru, mengubah data informasi, melihat detail data informasi, dan menghapus data informasi yang mana hanya dapat dilakukan jika pengguna login sebagai Administrator. Untuk pengujian SCF-013 akan dijelaskan pada Tabel 5.15. Kemudian pada pengujian SCF-014, proses mengubah data informasi merupakan salah satu proses pengujian untuk mengelola data informasi, yang mana hanya dapat dilakukan jika pengguna login sebagai Administrator. Proses ini dilakukan ketika Administrator ingin mengubah salah satu data informasi yang ada. Proses ini akan dijelaskan pada Tabel 5.16.

Tabel 5.15 Skenario 13 Pengujian Fungsionalitas Mengelola Data Informasi

No. Pengujian	SCF-013
Skenario Pengujian	Menambah data informasi
Kondisi Awal	Data informasi baru belum tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Informasi 3. Penguji memilih tombol <i>tambah</i> untuk menambah data informasi 4. Penguji mengisi semua isian form 5. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan dan menampilkan data informasi yang baru
Hasil yang diperoleh	Data informasi baru disimpan dan ditampilkan
Hasil Pengujian	Berhasil

Tabel 5.16 Skenario 14 Pengujian Fungsionalitas Mengelola Data Informasi

No. Pengujian	SCF-014
Skenario Pengujian	Mengubah data informasi
Kondisi Awal	Data informasi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Informasi 3. Penguji memilih tombol untuk mengubah data informasi pada salah satu data 4. Penguji mengubah data informasi pada form 5. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan dan menampilkan data informasi yang baru diubah
Hasil yang diperoleh	Data informasi disimpan dan ditampilkan
Hasil Pengujian	Berhasil

Pada pengujian SCF-015 yaitu proses melihat detail data informasi ini dilakukan ketika Administrator ingin memilih tombol detail. Proses akan dijelaskan pada Tabel 5.17.

Hasil pengujian SCF-013 akan ditampilkan pada Gambar 5.14. Kemudian untuk hasil pengujian SCF-014 akan ditampilkan pada halaman mengubah data informasi seperti pada Gambar 5.15. Sedangkan hasil pengujian SCF-015 akan ditampilkan pada halaman detail data informasi seperti pada Gambar 5.16.

Tabel 5.17 Skenario 15 Pengujian Fungsionalitas Mengelola Data Informasi

No. Pengujian	SCF-015
Skenario Pengujian	Melihat detail data informasi
Kondisi Awal	Data informasi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Informasi 3. Penguji memilih tombol <i>detail</i> pada salah satu data
Hasil yang diharapkan	Aplikasi dapat menampilkan detail data informasi yang dipilih
Hasil yang diperoleh	Detail data informasi ditampilkan
Hasil Pengujian	Berhasil

Gambar 5.14 Halaman Tambah Data Informasi

Ubah Informasi

Gambar 5.15 Halaman Ubah Data Informasi

Pada pengujian SCF-016 merupakan proses menghapus data informasi. Proses ini akan dijelaskan pada Tabel 5.18.



Gambar 5.16 Halaman Detail Data Informasi

Tabel 5.18 Skenario 16 Pengujian Fungsionalitas Menghapus Data Informasi

No. Pengujian	SCF-016
Skenario Pengujian	Menghapus data informasi
Kondisi Awal	Data informasi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Informasi 3. Penguji memilih tombol <i>hapus</i> pada salah satu data
Hasil yang diharapkan	Aplikasi dapat menghapus data informasi yang dipilih
Hasil yang diperoleh	Data informasi terhapus
Hasil Pengujian	Berhasil

5.2.3.2. Pengujian Fungsionalitas Mengelola Data Produk

Pada pengujian fungsionalitas mengelola data produk merupakan proses untuk menambah data produk baru, mengubah data produk, melihat detail data produk, dan menghapus data produk. Proses ini hanya dapat dilakukan oleh Administrator.

Proses menambah data produk akan dijelaskan pada Tabel 5.19. Pada pengujian SCF-017 dan SCF-018, proses mengubah data produk merupakan salah satu proses pengujian untuk mengelola data produk, yang mana hanya dapat dilakukan jika pengguna *login* sebagai Administrator. Proses ini akan dijelaskan pada Tabel 5.20.

Tabel 5.19 Skenario 17 Pengujian Fungsionalitas Mengelola Data Produk

No. Pengujian	SCF-017
Skenario Pengujian	Menambah data produk
Kondisi Awal	Data produk baru belum tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Produk 3. Penguji memilih tombol <i>tambah</i> 4. Penguji mengisi data produk pada form 5. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan dan menampilkan data produk yang baru
Hasil yang diperoleh	Data produk baru disimpan dan ditampilkan
Hasil Pengujian	Berhasil

Tabel 5.20 Skenario 18 Pengujian Fungsionalitas Mengelola Data Produk

No. Pengujian	SCF-018
Skenario Pengujian	Mengubah data produk
Kondisi Awal	Data produk telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Produk 3. Penguji memilih tombol untuk mengubah data produk pada salah satu data 4. Penguji mengubah data produk pada form 5. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan dan menampilkan data produk yang baru diubah
Hasil yang diperoleh	Data produk disimpan dan ditampilkan
Hasil Pengujian	Berhasil

Pengujian SCF-019 merupakan proses melihat detail data produk melalui halaman yang sama dengan sebelumnya. Administrator harus telah login terlebih dahulu kemudian masuk pada halaman Data Produk. Proses akan dijelaskan pada Tabel 5.21, dan hasil pengujian SCF-017 akan ditampilkan *form* menambah data produk seperti pada Gambar 5.17, dan hasil pengujian SCF-018 akan ditampilkan seperti pada Gambar 5.18.

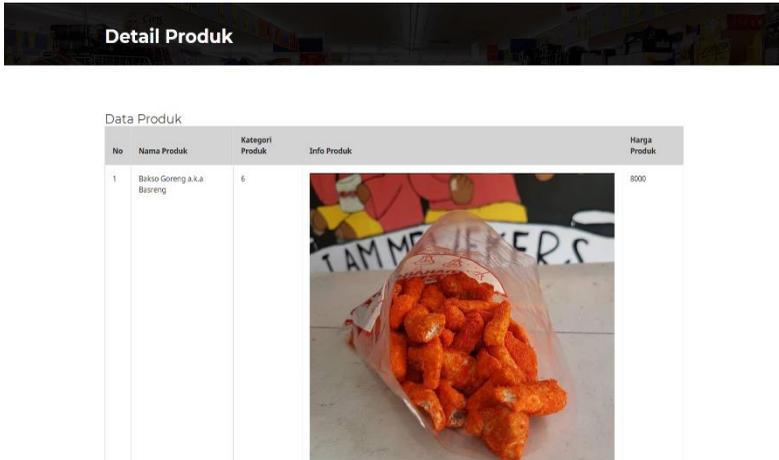
Tabel 5.21 Skenario 19 Pengujian Fungsionalitas Mengelola Data Produk

No. Pengujian	SCF-019
Skenario Pengujian	Melihat detail data produk
Kondisi Awal	Data produk telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Produk 3. Penguji memilih tombol <i>detail</i> pada salah satu data
Hasil yang diharapkan	Aplikasi dapat menampilkan detail data produk yang dipilih
Hasil yang diperoleh	Detail data produk ditampilkan
Hasil Pengujian	Berhasil

Gambar 5.17 Halaman Tambah Data Produk

Gambar 5.18 Halaman Ubah Data Produk

Hasil pengujian SCF-019 akan ditampilkan pada halaman melihat detail data produk seperti pada Gambar 5.19. Pada pengujian SCF-020 merupakan proses menghapus data produk. Proses ini akan dijelaskan pada Tabel 5.22.



Gambar 5.19 Halaman Detail Data Produk

Tabel 5.22 Skenario 20 Pengujian Fungsionalitas Menghapus Data Produk

No. Pengujian	SCF-020
Skenario Pengujian	Menghapus data produk
Kondisi Awal	Data produk telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Produk 3. Penguji memilih tombol <i>hapus</i> pada salah satu data
Hasil yang diharapkan	Aplikasi dapat menghapus data produk yang dipilih
Hasil yang diperoleh	Data produk terhapus
Hasil Pengujian	Berhasil

5.2.3.3. Pengujian Fungsionalitas Mengelola Transaksi

Pengujian fungsionalitas mengelola transaksi dilakukan ketika Administrator akan mengubah atau menghapus data transaksi. Proses mengubah data transaksi dan berhasil mengunggah resi pengiriman barang akan dijelaskan pada proses pengujian SCF-021 pada Tabel 5.23, dan hasil pengujian SCF-021 akan ditampilkan pada halaman data transaksi ketika berhasil mengunggah resi pengiriman seperti pada Gambar 5.20. Proses mengubah data transaksi dan gagal mengunggah resi pengiriman barang akan dijelaskan pada proses pengujian SCF-022 pada Tabel 5.24, dan hasil pengujian akan ditampilkan di halaman riwayat transaksi ketika gagal mengunggah resi pengiriman seperti pada Gambar 5.21.

Tabel 5.23 Skenario 21 Pengujian Fungsionalitas Mengelola Transaksi

No. Pengujian	SCF-021
Skenario Pengujian	Mengubah Data Transaksi dan mengunggah bukti pengiriman
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Transaksi 3. Penguji memilih tombol <i>detail</i> pada salah satu data transaksi yang telah melakukan pembayaran 4. Penguji mengubah data transaksi dengan mengunggah bukti pengiriman 5. Penguji memilih tombol “Konfirmasi”
Hasil yang diharapkan	Aplikasi dapat menyimpan data dan memunculkan pemberitahuan telah berhasil melakukan pengiriman
Hasil yang diperoleh	Data transaksi ter- <i>update</i> dan status konfirmasi pengiriman berubah
Hasil Pengujian	Berhasil

Data Transaksi*

Sukses! Resi pengiriman berhasil diunggah.

Gambar 5.20 Halaman Data Transaksi ketika Unggah Resi Kirim Berhasil

Tabel 5.24 Skenario 22 Pengujian Fungsionalitas Mengelola Transaksi

No. Pengujian	SCF-022
Skenario Pengujian	Mengubah Data Transaksi dan mengunggah bukti pengiriman
Kondisi Awal	Data transaksi telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Transaksi 3. Penguji memilih tombol <i>detail</i> pada salah satu data transaksi yang telah melakukan pembayaran 4. Penguji tidak mengunggah bukti pengiriman 5. Penguji memilih tombol “Konfirmasi”
Hasil yang diharapkan	Aplikasi dapat menyimpan data dan memunculkan pemberitahuan telah berhasil melakukan pengiriman
Hasil yang diperoleh	Data transaksi ter- <i>update</i> dan status konfirmasi pengiriman berubah
Hasil Pengujian	Berhasil

Data Transaksi*

Gagal! Tolong unggah ulang resi anda

Gambar 5.21 Halaman Data Transaksi ketika Unggah Resi Kirim Gagal

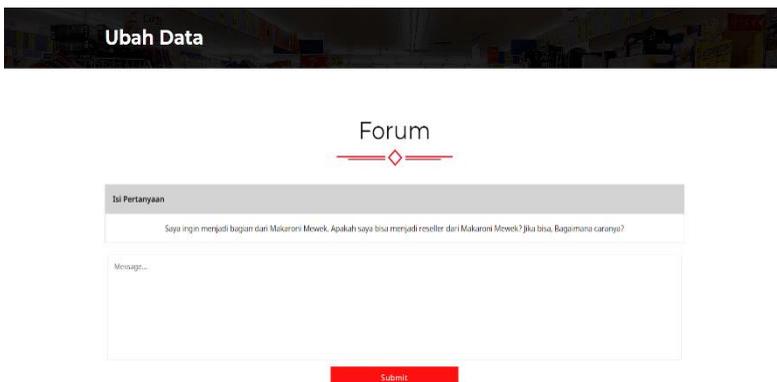
5.2.3.4. Pengujian Fungsionalitas Menjawab Pertanyaan

Pengujian fungsionalitas SCF-023 mengirim pertanyaan dilakukan ketika Administrator memilih menu Data Forum dan

memilih ubah pada salah satu daftar pertanyaan. Proses ini akan dijelaskan pada Tabel 5.25, dan hasil pengujian SCF-023 akan ditampilkan pada halaman data forum seperti pada Gambar 5.22.

Tabel 5.25 Skenario 23 Pengujian Fungsionalitas Menjawab Pertanyaan

No. Pengujian	SCF-023
Skenario Pengujian	Menjawab pertanyaan
Kondisi Awal	Data pertanyaan telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu Data Forum 3. Penguji memilih tombol <i>edit</i> untuk mengisi jawaban dari data pertanyaan tertentu 4. Penguji mengisi pada form 5. Penguji memilih tombol “Submit”
Hasil yang diharapkan	Aplikasi dapat menyimpan data jawaban
Hasil yang diperoleh	Data jawaban ditampilkan
Hasil Pengujian	Berhasil



The screenshot shows a forum interface. At the top, there is a dark header with the text "Ubah Data". Below this is a white header area with the word "Forum" centered, flanked by two horizontal lines and a small diamond symbol. The main content area has a grey header "Taj Pertanyaan" and a question: "Saya ingin menjadi bagian dari Mekarani Mewek. Apakah saya bisa menjadi reseller dari Mekarani Mewek? Jika bisa, bagaimana caranya?". Below the question is a text input field with the placeholder "Message...". At the bottom of the form is a red "Submit" button.

Gambar 5.22 Halaman Mengisi Jawaban

5.2.3.5. Pengujian Fungsionalitas Menambah Pengguna Administrator/Reseller

Pengujian fungsionalitas SCF-024 menambah pengguna pembeli dilakukan ketika pengguna memilih menu Daftar. Proses ini akan dijelaskan pada Tabel 5.26, dan hasil pengujian akan ditampilkan pada halaman daftar seperti pada Gambar 5.23.

Tabel 5.26 Skenario 24 Pengujian Fungsionalitas Menambah Pengguna Administrator/Reseller

No. Pengujian	SCF-024
Skenario Pengujian	Menambah data pengguna Administrator/Reseller
Kondisi Awal	Data pengguna Administrator/Reseller baru belum tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih tombol untuk menambah data pengguna sebagai Administrator/Reseller 3. Penguji mengisi semua isian form 4. Penguji memilih tombol "Submit"
Hasil yang diharapkan	Aplikasi dapat menyimpan dan menampilkan data pengguna Administrator/Reseller yang baru
Hasil yang diperoleh	Data Administrator/Reseller baru ditampilkan
Hasil Pengujian	Berhasil

Register

Nama Depan
 Nama Belakang
 Alamat
 Handphone
 Email
 Password
 Pilih Status

Submit

Gambar 5.23 Halaman Daftar sebagai Administrator/Reseller

5.2.3.6. Pengujian Fungsionalitas Melihat Riwayat Produk/Perhitungan Algoritma Apriori

Pengujian fungsionalitas SCF-025 menambah pengguna pembeli dilakukan ketika pengguna memilih menu Daftar. Proses ini akan dijelaskan pada Tabel 5.27, dan hasil pengujian akan ditampilkan pada halaman riwayat produk seperti pada Gambar 5.24 dan setelah pengisian form akan menampilkan seperti pada Gambar 5.25.

Tabel 5.27 Skenario 25 Pengujian Fungsionalitas Melihat Riwayat Produk

No. Pengujian	SCF-025
Skenario Pengujian	Melihat riwayat produk
Kondisi Awal	Data pengguna telah tersimpan
Langkah-langkah	<ol style="list-style-type: none"> 1. Penguji login sebagai Administrator 2. Penguji memilih menu riwayat produk 3. Penguji memasukkan nilai minimum <i>support</i> dan nilai minimum <i>confidence</i> 4. Penguji memilih tombol “Proses”
Hasil yang diharapkan	Aplikasi dapat menampilkan data riwayat produk berkaitan berdasarkan riwayat pembelian
Hasil yang diperoleh	Data riwayat produk ditampilkan
Hasil Pengujian	Berhasil

Riwayat Produk

Min support * %

Min confidence * %

Proses

Gambar 5.24 Halaman Riwayat Produk sebelum mengisi form

L3 (Large 3-itemset)					
No	Item1	Item2	Item3	Qty	Support
1	Mewek Ori Regulers	Usus Goreng Reguler	Seblak Holic	3	0.05

Hasil Aturan Apriori				
No	Rule	Support	Confidence	
1	Jika konsumen membeli Usus Goreng Reguler, Seblak Holic maka membeli Mewek Ori Regulers	0.05	3/7	0.43
2	Jika konsumen membeli Mewek Ori Regulers, Seblak Holic maka membeli Usus Goreng Reguler	0.05	3/3	1
3	Jika konsumen membeli Mewek Ori Regulers, Usus Goreng Reguler maka membeli Seblak Holic	0.05	3/5	0.6

Gambar 5.25 Halaman Riwayat Produk setelah mengisi form

Hasil dari aturan asosiasi yang telah dilakukan oleh administrator disimpan dalam database yang kemudian ditampilkan menjadi relasi produk pada halaman konfirmasi pembelian ketika pembeli melakukan transaksi yang mana dapat dilihat pada Gambar 5.26.

Total:
Rp. 15,000

Jenis Produk	Harga Satuan	Qty	Subtotal
 Mewek Ori Regulers	5000	1	Rp. 5,000
 Seblak Holic	10000	1	Rp. 10,000

Relasi Produk



Gambar 5.26 Halaman Relasi Produk pada Pembeli

5.3. Pengujian Non-Fungsional Pembeli

Telah dilakukan survei kepada penguji yang telah melakukan uji coba pada aplikasi Makaroni Mewek. Penguji berjumlah 21 orang sebagai pembeli. Terdapat 2 aspek penilaian dalam survei ini, yaitu kemudahan penggunaan fitur dan kepuasan menggunakan aplikasi. Pengujian dilakukan untuk memastikan kebutuhan non-fungsional dapat terpenuhi berdasarkan skenario dari setiap kasus penggunaan untuk aktor pembeli atau *reseller*. Penguji memiliki rentang umur 13-50 tahun, berjenis kelamin laki-laki (L) sebanyak 6 orang, dan perempuan (P) sebanyak 15 orang. Setelah penguji melakukan uji coba, akan diminta survei untuk menilai aplikasi berdasarkan kualitas dengan hasil survei dapat dilihat pada Tabel 5.29. Sedangkan aspek penilaian kedua yaitu survei kepuasan dari responden, dengan hasil 91% dari 21 orang menyatakan puas dengan penggunaan aplikasi.

Tabel 5.28 Hasil Survei Kepuasan

No	Pertanyaan	Penilaian					Rata-rata	%
		1	2	3	4	5		
1	Apakah informasi yang disediakan aplikasi ini mudah dimengerti?	0	0	0	5	16	4,76	95%
2	Apakah penggunaan menu atau fitur aplikasi menu mudah dipahami?	0	0	0	3	18	4,86	97%
3	Apakah aplikasi sesuai dengan kebutuhan?	0	0	3	10	8	4,24	85%
4	Apakah aplikasi dapat dengan mudah dipelajari?	0	0	0	4	17	4,81	96%
5	Apakah aplikasi mempunyai kemampuan dan fungsi sesuai yang diharapkan?	0	0	3	9	9	4,29	86%
6	Secara keseluruhan apakah penggunaan aplikasi ini memuaskan?	0	0	0	16	5	4,24	85%
Total rata-rata keseluruhan							4,53	91%

Tabel 5.29 Hasil Kuisioner Kualitas

No.	Pernyataan	Penilaian					Rata-rata	%
		1	2	3	4	5		
1	Kemudahan melihat informasi dari aplikasi seperti pengumuman, promo	0	0	0	0	21	5	100%
2	Kemudahan melihat produk-produk yang dijual seperti makaroni, coklatini, seblak	0	0	0	0	21	5	100%
3	Kemudahan pengguna untuk menambah akun dalam <i>website</i>	0	0	0	4	17	4,81	96%
4	Kemudahan pengguna masuk (login) ke dalam sistem	0	0	0	3	18	4,86	97%
5	Kemudahan pengguna melihat profil diri	0	0	0	1	20	4,95	99%
6	Kemudahan pembeli melakukan pemesanan dengan menggunakan aplikasi	0	0	3	12	6	4,14	83%
7	Kemudahan pembeli melihat riwayat transaksi yang dilakukan oleh pembeli tersebut	0	0	0	9	12	4,57	91%
8	Kemudahan pembeli melihat detail pemesanan yang telah dilakukan	0	0	0	9	12	4,57	91%
9	Kemudahan pembeli ketika ingin menghapus/membatalkan pemesanan yang dilakukan	0	0	0	9	12	4,57	91%
10	Kemudahan pembeli ketika ingin melakukan pembayaran dari transaksi yang dilakukan	0	0	4	11	6	4,10	82%
11	Kemudahan pengguna melihat forum tanya jawab	0	0	0	4	17	4,81	96%
12	Kemudahan pengguna untuk mengirim pertanyaan pada Admin	0	0	1	4	16	4,71	94%
Hasil rata-rata penilaian secara keseluruhan							4,67	93%

Berdasarkan respon dari survei pengujian aplikasi didapat nilai 4,67 yang mana dalam persen adalah 93% mengenai kemudahan dari penggunaan aplikasi menyatakan mudah.

5.4. Pengujian Algoritma Apriori

Algoritma Apriori digunakan untuk melakukan perhitungan untuk menampilkan rekomendasi produk pada halaman Pengunjung, Pembeli, dan *Reseller*. Untuk mencari rekomendasi produk hanya dapat dilakukan ketika Administrator telah *login*, kemudian memilih menu Riwayat Produk. Pada halaman ini akan muncul *form* minimum *support* dan minimum *confidence* yang mana harus diisi terlebih dahulu sebagai syarat untuk menampilkan rekomendasi produk, sehingga produk yang direkomendasikan jika rata-rata dari nilai *support* dan nilai *confidence* dari kombinasi produk dalam suatu transaksi memenuhi.

Pada pengujian algoritma Apriori dilakukan dengan 2 cara, pertama dilakukan uji coba pada aplikasi dengan beberapa kali memasukkan nilai minimum *support* dan nilai minimum *confidence* yang berbeda. Pengujian kedua yaitu dengan percobaan manual yang dilakukan dengan menggunakan Microsoft Excel.

Kemudian langkah-langkah pengujian pada aplikasi, yang diambil untuk menerapkan algoritma Apriori pada halaman riwayat produk akan dijelaskan dengan skenario sistem sebagai berikut:

Skenario keseluruhan sistem:

1. Masukkan minimum *support* dan minimum *confidence*.
2. Hitung nilai *support* dari setiap *item* pada transaksi untuk menjadi kandidat pertama *item*, disebut dengan *k-itemset*.
3. Ambil *item* yang memiliki nilai *support* lebih tinggi dari minimum *support*, kemudian disebut *Frequent item* (F_i).
4. Kombinasikan *frequent item* dengan *k-itemset*.
5. Hitung nilai *support* dari kombinasi *item*, yang kemudian disebut *Candidate item* (C_i).
6. Ambil C_i yang memenuhi minimum *support*.
7. Kombinasikan hingga mendapatkan C_i terakhir, untuk membentuk hasil aturan asosiasi.

Riwayat Produk



Min support *

 %

Min confidence *

 %

 Proses

Gambar 5.27 *Input minimum support dan minimum confidence*

Pada Gambar 5.27 adalah *form* untuk langkah dari skenario pertama untuk pengujian yang dilakukan dengan melalui sistem. Kemudian pada Gambar 5.28 yaitu langkah kedua dari skenario, yang mana semua *item* pada transaksi dihitung nilai *support*-nya.

C1 (Kandidat 1-Itemset)			
No	Item1	Qty	Support
1	Mewek Ori Regulers	23	0.42
2	Mewek Ori Premium	6	0.11
3	Mewek Ori Spesial Box	5	0.09
4	Mewek Balado Reguler	9	0.16
5	Mewek Pizza Reguler	10	0.18
6	Mewek Balado Premium	1	0.02
7	Mewek Balado Spesial Box	4	0.07
8	Mewek Sapi Panggang Reguler	4	0.07
9	Mewek Ayam Bawang/Bakar Reguler	8	0.15
10	Mewek Pizza Premium	2	0.04
11	Mewek Pizza Spesial Box	2	0.04
12	Mie Nenek	4	0.07

Gambar 5.28 *Hitung nilai support setiap item*

L1 (Large 1-Itemset)			
No	Item1	Qty	Support
1	Mewek Ori Regulers	23	0.42
2	Mewek Ori Premium	6	0.11
3	Mewek Ori Spesial Box	5	0.09
4	Mewek Balado Reguler	9	0.16
5	Mewek Pizza Reguler	10	0.18
6	Mewek Balado Spesial Box	4	0.07
7	Mewek Sapi Panggang Reguler	4	0.07
8	Mewek Ayam Bawang/Bakar Reguler	8	0.15
9	Mewek Pizza Premium	2	0.04
10	Mewek Pizza Spesial Box	2	0.04
11	Mie Nenek	4	0.07
12	Mie Kakek	5	0.09

Gambar 5.29 Ambil *item* yang memenuhi minimum *support*

Pada Gambar 5.29 yaitu melakukan langkah ketiga dari skenario pengujian. Sedangkan Gambar 5.30 yaitu melanjutkan langkah keempat dari skenario pengujian.

C2 (Kandidat 2-Itemset)				
No	Item1	Item2	Qty	Support
1	Mewek Ori Regulers	Mewek Ori Premium	3	0.05
2	Mewek Ori Regulers	Mewek Ori Spesial Box	2	0.04
3	Mewek Ori Regulers	Mewek Balado Reguler	6	0.11
4	Mewek Ori Regulers	Mewek Pizza Reguler	3	0.05
5	Mewek Ori Regulers	Mewek Balado Spesial Box	1	0.02
6	Mewek Ori Regulers	Mewek Sapi Panggang Reguler	2	0.04
7	Mewek Ori Regulers	Mewek Ayam Bawang/Bakar Reguler	2	0.04
8	Mewek Ori Regulers	Mewek Pizza Premium	0	0
9	Mewek Ori Regulers	Mewek Pizza Spesial Box	1	0.02
10	Mewek Ori Regulers	Mie Nenek	1	0.02
11	Mewek Ori Regulers	Mie Kakek	2	0.04
12	Mewek Ori Regulers	Usus Goreng Reguler	4	0.07
13	Mewek Ori Regulers	Oran Caklatini	2	0.04

Gambar 5.30 Kombinasi *Frequent item* dan *k-itemset* dan hitung nilai *support*-nya.

L2 (Large 2-itemset)				
No	Item1	Item2	Qty	Support
1	Mewek Ori Regulers	Mewek Ori Premium	3	0.05
2	Mewek Ori Regulers	Mewek Ori Spesial Box	2	0.04
3	Mewek Ori Regulers	Mewek Balado Reguler	6	0.11
4	Mewek Ori Regulers	Mewek Pizza Reguler	3	0.05
5	Mewek Ori Regulers	Mewek Sapi Panggang Reguler	2	0.04
6	Mewek Ori Regulers	Mewek Ayam Bawang/Bakar Reguler	2	0.04
7	Mewek Ori Regulers	Mie Kakek	2	0.04
8	Mewek Ori Regulers	Usus Goreng Reguler	4	0.07
9	Mewek Ori Regulers	Oreo Coklatini	2	0.04
10	Mewek Ori Regulers	Seblak Holic	2	0.04
11	Mewek Ori Regulers	Matcha Coklatini	3	0.05
12	Mewek Ori Regulers	Mewek Ori Spesial Box	2	0.04

Gambar 5.31 Ambil *item* yang memenuhi nilai minimum *support*

Pada Gambar 5.31 dan Gambar 5.32 sebenarnya hanya mengulangi langkah ketiga dan keempat dari skenario pengujian.

C3 (Kandidat 3-itemset)					
No	Item1	Item2	Item3	Qty	Support
1	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	2	0.04
2	Mewek Ori Regulers	Mewek Ori Premium	Mewek Balado Reguler	2	0.04
3	Mewek Ori Regulers	Mewek Ori Premium	Mewek Pizza Reguler	0	0
4	Mewek Ori Regulers	Mewek Ori Premium	Mewek Sapi Panggang Reguler	0	0
5	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ayam Bawang/Bakar Reguler	0	0
6	Mewek Ori Regulers	Mewek Ori Premium	Mie Kakek	0	0
7	Mewek Ori Regulers	Mewek Ori Premium	Usus Goreng Reguler	0	0
8	Mewek Ori Regulers	Mewek Ori Premium	Oreo Coklatini	0	0
9	Mewek Ori Regulers	Mewek Ori Premium	Seblak Holic	0	0
10	Mewek Ori Regulers	Mewek Ori Premium	Matcha Coklatini	1	0.02
11	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ralarin Spesial Box	0	0

Gambar 5.32 Kombinasi dengan *k-itemset* sebelumnya dan hitung nilai *support*

L3 (Large 3-Itemset)					
No	Item1	Item2	Item3	Qty	Support
1	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	2	0.04
2	Mewek Ori Regulers	Mewek Ori Premium	Mewek Balado Reguler	2	0.04
3	Mewek Ori Regulers	Mewek Pizza Reguler	Oreo Coklatini	2	0.04
4	Mewek Ori Regulers	Usus Goreng Reguler	Seblak Holic	2	0.04
5	Mewek Pizza Reguler	Usus Goreng Reguler	Seblak Holic	2	0.04
6	Mewek Pizza Reguler	Usus Goreng Reguler	Matcha Coklatini	2	0.04
7	Mewek Pizza Reguler	Seblak Holic	Matcha Coklatini	2	0.04
8	Mie Kakek	Usus Goreng Reguler	Oreo Coklatini	2	0.04
9	Usus Goreng Reguler	Oreo Coklatini	Matcha Coklatini	2	0.04
10	Usus Goreng Reguler	Oreo Coklatini	Bakso Goreng a.k.a Basreng	2	0.04
11	Usus Goreng Reguler	Seblak Holic	Matcha Coklatini	2	0.04
12	Mie Goreng Reguler	Caklak Miler	Bakso Goreng a.k.a Basreng	2	0.04

Gambar 5.33 Ambil *item* yang memenuhi nilai minimum *support*

Pada Gambar 5.33 dan Gambar 5.34 juga mengulangi langkah ketiga dan keempat dikarenakan *item* yang ada masih dapat dikombinasikan dan nilai *support* yang dihasilkan masih memenuhi nilai minimum *support*.

C4 (Kandidat 4-Itemset)						
No	Item1	Item2	Item3	Item4	Qty	Support
1	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Mewek Balado Reguler	1	0.02
2	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Mewek Pizza Reguler	0	0
3	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Oreo Coklatini	0	0
4	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Usus Goreng Reguler	0	0
5	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Seblak Holic	0	0
6	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Matcha Coklatini	0	0
7	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Mie Kakek	0	0
8	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Bakso Goreng a.k.a Basreng	0	0
9	Mewek Ori Regulers	Mewek Ori Premium	Mewek Ori Spesial Box	Cuoklat Coklatini	0	0
10	Mewek Ori Regulers	Mewek Ori Premium	Mewek Balado Reguler	Mewek Pizza Reguler	0	0
11	Mewek Ori Regulers	Mewek Ori Premium	Mewek Balado Reguler	Oreo Coklatini	0	0

Gambar 5.34 Kombinasikan kembali dengan *k-itemset* dan hitung nilai *support*-nya.

L4 (Large 4-itemset)						
No	Item1	Item2	Item3	Item4	Qty	Support
1	Mewek Pizza Reguler	Usus Goreng Reguler	Seblak Holic	Matcha Coklatini	2	0.04
2	Oreo Coklatini	Usus Goreng Reguler	Matcha Coklatini	Bakso Goreng a.k.a Basreng	2	0.04

Hasil Aturan Apriori			
No	Rule	Support	Confidence
1	Jika konsumen membeli Usus Goreng Reguler, Seblak Holic, Matcha Coklatini maka membeli Mewek Pizza Reguler	0.04	2/2 1
2	Jika konsumen membeli Mewek Pizza Reguler, Seblak Holic, Matcha Coklatini maka membeli Usus Goreng Reguler	0.04	2/2 1
3	Jika konsumen membeli Mewek Pizza Reguler, Usus Goreng Reguler, Matcha Coklatini maka membeli Seblak Holic	0.04	2/2 1
4	Jika konsumen membeli Mewek Pizza Reguler, Usus Goreng Reguler, Seblak Holic maka membeli Matcha Coklatini	0.04	2/2 1
5	Jika konsumen membeli Usus Goreng Reguler, Matcha Coklatini, Bakso Goreng a.k.a Basreng maka membeli Oreo Coklatini	0.04	2/2 1
6	Jika konsumen membeli Oreo Coklatini, Matcha Coklatini, Bakso Goreng a.k.a Basreng maka membeli Usus Goreng Reguler	0.04	2/2 1
7	Jika konsumen membeli Oreo Coklatini, Usus Goreng Reguler, Bakso Goreng a.k.a Basreng maka membeli Matcha Coklatini	0.04	2/2 1
8	Jika konsumen membeli Oreo Coklatini, Usus Goreng Reguler, Matcha Coklatini maka membeli Bakso Goreng a.k.a Basreng	0.04	2/2 1

Gambar 5.35 Hingga didapatkan *Candidate item* terakhir dan menghasilkan aturan asosiasi.

Pada Gambar 5.35 didapatkan hasil *Candidate item* terakhir dapat menghasilkan pula aturan asosiasi dengan percobaan minimum *support* 3% atau 0,03 dan minimum *confidence* 2% atau 0,02. Hasil tersebut merupakan kombinasi *item* yang memiliki nilai *support* dan nilai *confidence* lebih dari minimum *support* dan minimum *confidence* yang telah ditentukan pada masukkan awal, dan menghasilkan 6 rekomendasi produk yang berbeda, namun dengan 8 aturan asosiasi.

Kemudian untuk membuktikan hasil perhitungan sistem betul atau tidak dilakukan juga perhitungan manual menggunakan Microsoft Excel. Hasil rekapitulasi dari beberapa percobaan pengujian manual dapat dilihat pada Tabel 5.30.

Tabel 5.30 Hasil Perhitungan Apriori Manual

No	Min Support	Min Confidence	Hasil Aturan Asosiasi	Support	Confidence
1	0,01	0,01	9	0,02	1
2	0,01	0,02	9	0,02	1
3	0,02	0,02	8	0,04	1
4	0,03	0,02	8	0,04	1
5	0,04	0,02	14	0,06	0,30
6	0,05	0,02	14	0,06	0,30
7	0,06	0,02	6	0,10	0,52
8	0,07	0,02	6	0,10	0,52
9	0,08	0,02	4	0,11	0,43
10	0,09	0,02	4	0,11	0,43
11	0,1	0,02	4	0,11	0,43

Dari hasil pengujian manual tersebut, jika dibandingkan dengan uji coba pada sistem, dengan kita masukkan minimum *support* 0,03 dan minimum *confidence* 0,02, menurut perhitungan manual didapatkan rata-rata nilai *support* 0,04 dan rata-rata nilai *confidence* 1 dengan menampilkan hasil 8 aturan asosiasi dari 6 macam sama dengan hasil dari perhitungan sistem yang menampilkan hasil perhitungan yang sama seperti yang dapat dilihat pada Gambar 5.34.

Berdasarkan hasil dari kedua jenis pengujian, dapat dilihat hasil aturan asosiasi terbanyak dibangkitkan oleh minimum *support* 0,04 atau 4%, dan hasil aturan asosiasi terkecil dibangkitkan oleh minimum *support* terbesar yaitu 0,1 atau 10%. Maka dapat disimpulkan bahwa semakin tinggi minimum *support* maka hasil aturan yang didapat akan semakin sedikit.

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan yang diperoleh selama pengerjaan Tugas Akhir dan saran mengenai pengembangan yang dapat dilakukan terhadap Tugas Akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari hasil pengamatan selama proses perancangan, implementasi, dan pengujian perangkat lunak yang dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Aplikasi industri rumah tangga dengan label “Mewek” berhasil mengakomodasi kebutuhan penjual dalam menjual produk pada pembeli dan *reseller*, yang didukung dengan adanya beberapa fitur berupa, Mengelola Data Informasi, Mengelola Data Produk, Mengelola Data Transaksi, dan dapat memberikan Rekomendasi Produk.
2. Algoritma Apriori berhasil menampilkan riwayat pembelian produk yang berelasi dengan produk lainnya, dan menampilkan sebagai rekomendasi produk yang sering dibeli bersamaan, dengan menggunakan perhitungan algoritma Apriori.
3. Aplikasi Mewek berhasil membantu penjual dalam mendapatkan data mengenai produk yang sering dibeli oleh pelanggan dengan menggunakan perhitungan algoritma Apriori dan dapat optimal dengan minimum *support* 0.4 dan minimum *confidence* 0.02, karena dapat menampilkan 4 produk yang berelasi dari total transaksi saat ini.

6.2. Saran

Berdasarkan kesimpulan pembahasan tugas akhir yang dilakukan, maka beberapa saran yang dapat diusulkan pada aplikasi ini adalah sebagai berikut:

1. Melakukan pengembangan dalam hal transaksi pembayaran dan konfirmasi pembayaran dengan cara membuat *payment gateway*, bekerja sama dengan *PayPal*, *Credit Card*, dan sebagainya.

DAFTAR PUSTAKA

- [1] "Perdagangan Elektronik," [Online]. Available: <http://www.wikipedia.com>. [Accessed Desember 2017].
- [2] "Menerka Arah Ekonomi Digital Indonesia 2018," 30 December 2017. [Online]. Available: <https://kumparan.com>. [Accessed 4 January 2018].
- [3] "Kenapa Harus Memiliki Online Shop Sendiri?," 17 February 2017. [Online]. Available: <https://www.kaskus.co.id/>. [Accessed 17 January 2018].
- [4] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," *IEEE*, pp. 487-499, 1994.
- [5] "List of Web-Designing & Web-Development Programming Languages.," 5 January 2017. [Online]. Available: <https://www.webakruti.com>. [Accessed 5 January 2018].
- [6] "What are the pros and cons of running a marketplace vs. a pure ecommerce site?," 16 January 2018. [Online]. Available: <https://www.quora.com>. [Accessed 28 January 2018].
- [7] Hestanto, "Pengertian Istilah Industri Kecil dan Kerajinan Rumah Tangga," 18 February 2017. [Online]. Available: <http://www.hestanto.web.id/pengertian-istilah-industri-kecil-dan-kerajinan-rumah-tangga/>. [Accessed 09 July 2018].
- [8] M. Fauzy, K. R. S. W and I. Asror, "PENERAPAN METODE ASSOCIATION RULE MENGGUNAKAN ALGORITMA APRIORI PADA SIMULASI PREDIKSI HUJAN WILAYAH KOTA BANDUNG," *JITTER*, vol. II, pp. 221-227, 2016.
- [9] "Pengertian Algoritma Apriori," 13 May 2016. [Online]. Available: <https://www.ilmuskripsi.com/2016/05/algoritma-apriori.html>. [Accessed 28 March 2018].

- [10] "Codeigniter," 22 February 2016. [Online]. Available: <https://id.wikipedia.org/wiki/CodeIgniter>. [Accessed 8 March 2018].
- [11] "Mengenal Apa itu Framework Codeigniter," 4 August 2017. [Online]. Available: <https://idcloudhost.com/panduan/mengenal-apa-itu-framework-codeigniter/>. [Accessed 8 March 2018].
- [12] H. D. Anggraeni, R. Saputra and B. Noranita, "APLIKASI DATA MINING ANALISIS DATA TRANSAKSI PENJUALAN OBAT MENGGUNAKAN ALGORITMA APRIORI," *Jurnal Masyarakat Informatika*, vol. IV, pp. 1-8, 2013.
- [13] R. Purbaningtyas, A. Arizal and T. Wardoyo, "PENERAPAN ALGORITMA APRIORI TID SEBAGAI METODE OPTIMASI Pencarian Data pada Aplikasi Mobile Sidoarjo on Hands (SoH)," *Jurnal Penelitian Pos dan Informatika*, vol. 7, pp. 87-96, 2017.

LAMPIRAN A. KODE SEMU

1	Data <- inialisasi mengambil data produk dari <i>database</i>
2	Load <i>view</i> <- Menampilkan halaman data produk

Kode Semu A.1 Melihat produk

1	<p>If <i>move_upload_file</i> = kode unik untuk mengunggah <i>file</i> <i>dataInsert</i> <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari <i>view</i> <i>config</i> <- konfigurasi <i>file</i> yang akan diunggah <i>namadirektori</i> <- nama direktori <i>file</i> yang akan diunggah Memuat <i>library</i> untuk mengunggah <i>file</i> Menambah data pada tabel If berhasil menambah data Menampilkan halaman produk Else Menampilkan halaman <i>form</i> tambah Else Menampilkan halaman <i>form</i> tambah</p>
---	--

Kode Semu A.2 Menambah data produk

1	<i>Id_produk</i> <- inialisasi id produk yang dipilih
2	Data <- mengambil data produk dari <i>database</i>
3	Load <i>view</i> <- Menampilkan halaman data produk

Kode Semu A.3 Melihat detail data produk

1	<i>Id_produk</i> <- inialisasi id produk yang dipilih
2	Data <- menghapus data produk dari <i>database</i>
3	Load <i>view</i> <- Menampilkan halaman data produk

Kode Semu A.4 Menghapus data produk

1	<p>If <code>move_upload_file</code> = kode unik untuk mengunggah <i>file</i> <code>dataInsert</code> <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari <i>view</i> <code>config</code> <- konfigurasi <i>file</i> yang akan diunggah <code>namadirektori</code> <- nama direktori <i>file</i> yang akan diunggah <code>id produk</code> <- sesuai dengan id yang ingin diubah Memuat <i>library</i> untuk mengunggah <i>file</i> Menambah data pada tabel If berhasil menambah data Menampilkan halaman produk Else Menampilkan halaman <i>form</i> tambah Else <code>dataInsert</code> <- array yang berisi atribut tabel yang akan ditambah beserta nilai dari atribut tersebut yang didapatkan dari <i>view</i> <code>config</code> <- konfigurasi <i>file</i> yang akan diunggah <code>namadirektori</code> <- nama direktori <i>file</i> yang akan diunggah <code>id produk</code> <- sesuai dengan id yang ingin diubah</p>
---	---

Kode Semu A.5 Mengubah data produk

1	<code>Id_user</code> <- inialisasi id user yang dipilih
2	<code>Data</code> <- mengambil data transaksi dari <i>database</i> berdasarkan pengguna
3	Load <i>view</i> <- Menampilkan halaman riwayat transaksi

Kode Semu A.6 Melihat riwayat transaksi

1	<code>Id_transaksi</code> <- inialisasi id transaksi yang dipilih
2	<code>Data</code> <- mengambil data history pembelian dari <i>database</i> berdasarkan id transaksi
3	Load <i>view</i> <- Menampilkan halaman detail pemesanan

Kode Semu A.7 Melihat detail pemesanan

1	<code>Id</code> <- id transaksi
2	<code>Data</code> <- menghapus data transaksi dari <i>database</i>
3	Load <i>view</i> <- Menampilkan halaman riwayat transaksi

Kode Semu A.8 Menghapus/Membatalkan pembelian

1	Id_transaksi <- inialisasi id transaksi yang dipilih
2	Data <- mengambil data history pembelian dari <i>database</i> berdasarkan id transaksi
3	Load <i>view</i> <- Menampilkan halaman detail transaksi

Kode Semu A.9 Melihat detail transaksi

1	<pre>If move_upload_file = kode unik untuk mengunggah file dataInsert <- array yang berisi atribut tabel yang akan ditambah beserta nilai yang didapatkan dari view config <- konfigurasi file yang akan diunggah namadirektori <- nama direktori file yang akan diunggah id_transaksi <- sesuai dengan id yang ingin diunggah Memuat library untuk mengunggah file Menambah data pada tabel Merubah konfirmasi status menjadi 1 Menampilkan halaman riwayat transaksi Else Menampilkan halaman riwayat transaksi</pre>
---	---

Kode Semu A.10 Mengunggah resi pengiriman

1	Id_transaksi <- inialisasi id transaksi yang dipilih
2	Data <- menghapus data transaksi dari <i>database</i>
3	Load <i>view</i> <- Menampilkan halaman riwayat transaksi

Kode Semu A.11 Menghapus data transaksi

1	Data <- mengambil data tabel pertanyaan dari <i>database</i>
2	Load <i>view</i> <- Menampilkan halaman tanya jawab

Kode Semu A.12 Melihat forum tanya jawab

1	dataInsert <- array yang berisi atribut tabel pertanyaan yang akan ditambah beserta nilai yang didapatkan dari <i>view</i>
2	Load <i>view</i> <- Menampilkan halaman data forum

Kode Semu A.13 Menjawab pertanyaan

1	<code>dataInsert <- array yang berisi atribut tabel <i>user</i> yang akan ditambah beserta nilai yang didapatkan dari <i>view</i></code>
2	<code>If berhasil menambah pengguna Administrator/Reseller Menampilkan halaman index Else Menampilkan halaman Daftar</code>

Kode Semu A.14 Menambah pengguna sebagai Administrator/Reseller

1	<code>Id_user <- inialisasi id user yang telah <i>login</i></code>
2	<code>Data <- mengambil data pengguna yang sedang <i>login</i></code>
3	<code>Load <i>view</i> <- Menampilkan halaman profil pengguna</code>

Kode Semu A.15 Melihat profil

BIODATA PENULIS



Hilma Kamilah, anak kedua dari empat bersaudara yang lahir di Surabaya, 16 Desember 1994. Penulis telah menempuh Pendidikan formal mulai dari SD Negeri Ketabang I/288 Surabaya (2001-2007), SMP Negeri 19 Surabaya (2007-2010), SMA Negeri 17 Surabaya (2010-2013) dan terakhir sebagai mahasiswa Departemen Informatika Institut Teknologi Sepuluh Nopember dengan rumpun mata kuliah Rekayasa Perangkat Lunak (2014-2018). Lulus dari SMA penulis melanjutkan Pendidikan di departemen Informatika. Semasa kuliah penulis aktif mengikuti berbagai kepanitiaan diantaranya Schematics 2015, Schematics 2016, GERIGI ITS 2015, dan kegiatan lain di dalam departemen maupun diluar departemen. Penulis juga aktif menjadi anggota Himpunan Mahasiswa Teknik Computer-Informatika pada tahun 2015 dan menjadi pengurus harian Himpunan Mahasiswa Teknik Computer-Informatika pada tahun 2016. Penulis dapat dihubungi melalui email hilmakamilah@gmail.com