

**PENGEMBANGAN SISTEM *SUPPLY CHAIN MANAGEMENT* BARANG
RONGSOK BERBASIS *WEB* (STUDI KASUS: BANDAR LAMPUNG)**

(Skripsi)

Oleh

FALAHYAN

1917051049



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2024

**PENGEMBANGAN SISTEM *SUPPLY CHAIN MANAGEMENT* BARANG
RONGSOK BERBASIS WEB (STUDI KASUS: BANDAR LAMPUNG)**

Oleh

FALAHYAN

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2024

ABSTRAK

PENGEMBANGAN SISTEM *SUPPLY CHAIN MANAGEMENT* BARANG RONGSOK BERBASIS WEB (STUDI KASUS: BANDAR LAMPUNG)

Oleh

Falahyan

Penelitian ini mengembangkan sistem Supply Chain Management (SCM) berbasis web untuk meningkatkan efisiensi pengelolaan barang rongsok di Kota Bandar Lampung. Sistem ini menggunakan React.js untuk antarmuka pengguna, Node.js sebagai platform back-end, dan PostgreSQL untuk manajemen basis data. Proses pengembangan menggunakan metodologi Waterfall, dengan hasil pengujian menunjukkan bahwa sistem mampu menjalankan fungsi utamanya dan menangani hingga 1000 pengguna simultan tanpa penurunan kinerja. Pengujian intuitivitas antarmuka menunjukkan mayoritas pengguna merasa sistem mudah digunakan. Kesimpulannya, sistem SCM ini menawarkan solusi efektif dalam pengelolaan barang rongsok, berpotensi mendukung pelestarian lingkungan dan pengembangan ekonomi lokal, serta dapat diterapkan di kota-kota lain dengan tantangan serupa dan juga sistem ini telah melalui tahap pengujian performa, beban, dan intuitivitas antarmuka dengan hasil sistem berhasil menanggung 1.000 pengguna secara simultan dan presentase intuitivitas antarmuka sebesar 82%.

Kata kunci: Supply Chain Management, Barang Rongsok, Daur Ulang

ABSTRACT

DEVELOPMENT OF A WEB-BASED SCRAP MATERIALS SUPPLY CHAIN MANAGEMENT SYSTEM (CASE STUDY: BANDAR LAMPUNG)

By

Falahyan

This research develops a web-based Supply Chain Management (SCM) system to improve the efficiency of scrap management in Bandar Lampung City. The system uses React.js for the user interface, Node.js as the back-end platform, and PostgreSQL for database management. The development process uses the Waterfall methodology, with test results showing that the system is able to perform its main functions and handle up to 1000 simultaneous users without performance degradation. Interface intuitiveness testing showed the majority of users found the system easy to use. In conclusion, this SCM system offers an effective solution for scrap management, has the potential to support environmental preservation and local economic development, and can be applied in other cities with similar challenges also this system has gone through performance, load, and interface intuitiveness testing stages with the results of the system successfully bearing 1,000 simultaneous users and the percentage of interface intuitiveness of 82%..

Keywords: Supply Chain Management, Scrap Materials, Recycling

Judul Skripsi : **PENGEMBANGAN SISTEM *SUPPLY CHAIN*
MANAGEMENT BARANG RONGSOK
BERBASIS WEB (STUDI KASUS: BANDAR
LAMPUNG)**

Nama Mahasiswa : Falahyan

Nomor Pokok Mahasiswa : 1917051049

Mahasiswa

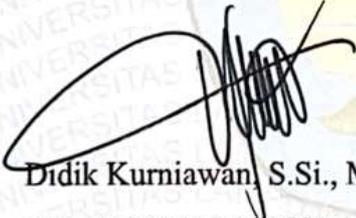
Program Studi : **Ilmu Komputer**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



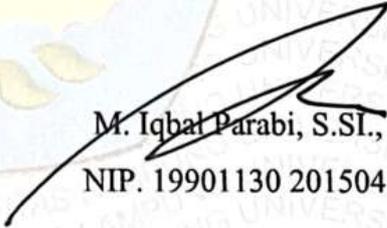
MENYETUJUI

1. Komisi Pembimbing



Didik Kurniawan, S.Si., M.T.

NIP. 19800419 200501 1 004



M. Iqbal Parabi, S.Si., M.T.

NIP. 19901130 201504 1 002

2. Ketua Jurusan Ilmu Komputer



Dwi Sakethi, S.Si., M.Kom.

NIP. 19680611 199802 1 001

MENGESAHKAN

1. Tim Penguji

Ketua

: Didik Kurniawan, S.Si., M.T.



Sekretaris

: M. Iqbal Parabi, S.SI., M.T.



Anggota

: Tristiyanto, S.Kom., M.I.S., Ph.D



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Heri Satria, S.Si., M.Si.

NIP. 19711001200501 1 002

Tanggal Lulus Ujian Skripsi : 15 Agustus 2024

PERNYATAAN

Saya yang bertanda tangan dibawah ini, menyatakan bahwa skripsi saya dengan judul **“Pengembangan Sistem *Supply Chain Management* Barang Rongsok Berbasis *Web* (Studi Kasus: Bandar Lampung)”** merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka bersedia menerima sanksi berupa pencabutan gelar yang telah saya terima.

Bandar Lampung, 15 Agustus 2024



Falahyan

NPM. 1917051049

RIWAYAT HIDUP



Lahir pada hari Selasa, 24 Juli 2001. Anak kedua dari Bapak Drs. Syaekhuddin, M.M. dan Ibu Supiyanti, menyelesaikan pendidikan pada tahun 2013 di SD Negeri 01 Way Mengaku. Kemudian menyelesaikan pendidikan menengah di SMP Negeri 57 Jakarta Selatan pada tahun 2016 dan lulus dari pendidikan menengah atas di SMA Negeri 1 Liwa Pada tahun 2019.

Pada tahun 2019, terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Universitas Lampung melalui jalur SBMPTN. Kegiatan-kegiatan yang dilakukan selama menjadi mahasiswa sebagai berikut.

1. Menjadi anggota Adapter Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2019
2. Menjadi Anggota Eksternal Himpunan Mahasiswa Jurusan Ilmu Komputer 2020
3. Menjadi Asisten Dosen Jurusan Ilmu Komputer pada tahun 2021 hingga 2022.
4. Mengikuti ujian sertifikasi dan mendapat sertifikat *Junior Web Developer* oleh Lembaga Sertifikasi Profesi Informatika pada tahun 2022.
5. Melaksanakan Kerja Praktik pada bulan Januari 2022 di Radar TV Lampung.
6. Mengikuti Kuliah Kerja Nyata 2023 periode 1 di Pekon Argomulyo, Kecamatan Batu Ketulis, Lampung Barat, Lampung.
7. Mengikuti Magang Kampus Merdeka di PT. Tunas Dwipa Matra pada Agustus hingga Desember 2022.

MOTTO

“Berani berbuat berani bertanggung jawab.”

(Falahyan)

PERSEMBAHAN

Puji dan syukur kepada ke hadirat Allah S.W.T. atas segala rahmat, hidayah, dan karunia-Nya sehingga dapat menyelesaikan skripsi ini dengan sebaik-baiknya.

Kupersembahkan karya ini kepada :

Kedua Orang Tuaku, Mertuaku, dan Istriku Tersayang

Yang selalu mendukung dan mendoakan setiap perjalanan yang saya lalui. Saya ucapkan terimakasih sebesar-besarnya atas kasih sayang, didikan dan pengorbanan yang telah diberikan kepada saya dan tak akan mungkin bisa terbalaskan.

Seluruh Keluarga Besar Ilmu Komputer 2019

**Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Lampung**

SANWACANA

Puji Syukur ke hadirat Tuhan Yang Maha Esa. Atas rahmat dan berkat-Nya, penulis dapat menyelesaikan skripsi yang berjudul **“Pengembangan Sistem *Supply Chain Management* Barang Rongsok Berbasis *Web* (Studi Kasus: Bandar Lampung)”** dengan tepat waktu. Dalam melaksanakan penelitian dan pembuatan skripsi ini, penulis banyak mendapat bimbingan dan dukungan dari berbagai pihak, sehingga pada kesempatan ini penulis ingin menyampaikan ungkapan terima kasih ini kepada:

1. Allah S.W.T. yang menjadi sumber kekuatan dan kecerdasan , yang selalu memberikan petunjuk selama penulis menyelesaikan skripsi.
2. Kedua Orang tua, Drs. H. Syaekhuddin, M.M. dan Hj. Supiyanti serta keluarga yang telah memberikan dukungan dan motivasi secara finansial dan fasilitas kepada penulis untuk menyelesaikan dengan baik.
3. Kedua Mertua , H. Abdul Rosyid, S.Ag. dan Hj. Linda Susilawati, S.Ag., M.Ag. yang memberikan dukugan moral dan doa yang tiada henti kepada penulis untuk menyelesaikan dengan baik.
4. Istriku tersayang, Indah Siti Aisyah yang selalu mendampingi penulis sepanjang penelitian ini berlangsung, sehingga dapat menyelesaikan penelitian .
5. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan FMIPA Universitas Lampung.
6. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
7. Almarhumah Ibu Yohana Tri Utami, M.Kom. selaku Dosen Pembimbing Akademik.

8. Bapak Didik Kurniawan, S.Si., M.T., selaku Dosen Pembimbing 1 yang telah membimbing sehingga penulis dapat menyelesaikan skripsi dengan baik.
9. Bapak Muhammad Iqbal Parabi, S.Si., M.T., selaku Dosen Pembimbing 2 yang telah membimbing serta memberi masukan dalam proses pembuatan skripsi.
10. Bapak Tristiyanto, S.Kom., M.I.S., Ph.D. selaku Dosen Pembahas yang telah banyak memberikan masukan dalam penelitian skripsi ini.
11. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu serta pengalaman semasa bangku perkuliahan.
12. Ibu Ade Nora Maela dan seluruh staf di Jurusan Ilmu Komputer yang telah sabar membantu segala urusan administrasi di masa perkuliahan.
13. Candra Martadinata, Farra Dzakiyyah Berliana selaku rekan penghibur penulis selama melaksanakan penelitian.
14. Teman-teman Jurusan Ilmu Komputer Universitas Lampung angkatan 2019 yang senantiasa memberikan dukungan dan telah berjuang bersama selama menjalankan studi perkuliahan.

Bandar Lampung, 15 Agustus 2024

Falahyan

NPM.1917051049

DAFTAR ISI

	Halaman
DAFTAR ISI	iii
DAFTAR TABEL	vii
DAFTAR GAMBAR	viii
I. PENDAHULUAN	1
1.1 Latar Belakang dan Masalah	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
II. TINJAUAN PUSTAKA	4
2.1 <i>Supply Chain Management (SCM)</i>	5
2.1.1 <i>Supply Chain Management (SCM) Barang Rongsok</i>	6
2.2 Sistem Informasi	8
2.3 Sistem Informasi Berbasis <i>Web</i>	9
2.3.1 <i>Teknologi Web</i>	9
2.3.1.1 HTML	9
2.3.1.2 Bahasa Pemrograman	10
2.3.1.3 Sistem Interaksi	12
2.3.1.4 <i>Database Management System (DBMS)</i>	13
2.4 Barang Rongsok	16
2.4.1 Definisi	16
2.4.2 Masalah	16

2.4.3.	Potensi Ekonomi	17
2.4.4.	Manfaat	18
2.5.	Penelitian Terdahulu	19
2.5.1.	Implementasi <i>Supply Chain Management</i> (SCM) Toko Alat dan Bahan Bangunan Berbasis <i>Web</i> (Studi Kasus: TB. Bojong Indah)	20
2.5.2.	Rancang Bangun <i>Supply Chain Management</i> Budidaya Jamur Berbasis <i>Web</i> (Studi Kasus: Budidaya Jamur Jatayutm)	21
2.5.3.	Aplikasi Jual Beli Barang Rongsok Berbasis Jarak Menggunakan Fitur <i>Location Based Service</i>	21
2.5.4.	Sistem Informasi Pemantauan Penjualan Barang Rongsok Menggunakan <i>Framework</i> CI.4 Pada CV. Sumber Baja	21
2.5.5.	Perancangan Sistem Informasi Pengelolaan Barang Rongsok Berbasis <i>Web</i> Menggunakan Kerangka Kerja CodeIgniter	22
2.6.	Pengembangan Perangkat Lunak	22
2.6.1.	Metode <i>Waterfall</i>	23
2.6.2.	<i>Unified Modelling Language</i>	24
2.6.2.1.	<i>Use Case Diagram</i>	24
2.6.2.2.	<i>Class Diagram</i>	25
2.6.3.	Pengujian Sistem	26
2.6.3.1.	Pengujian Fungsionalitas Sistem	27
2.6.3.2.	Pengujian Performa dan Beban Sistem	27
2.6.3.3.	Pengujian Intuitivitas Antarmuka Sistem	28
III.	METODOLOGI PENELITIAN	29
3.1.	Waktu dan Tempat Penelitian	29
3.2.	Perangkat Penelitian	29
3.2.1.	Perangkat Lunak (<i>Software</i>)	29
3.2.2.	Perangkat Keras (<i>Hardware</i>)	30

3.3.	Jenis dan Sumber Data	30
3.3.1.	Data Primer	30
3.3.2.	Data Sekunder	30
3.4.	Tahapan Penelitian	31
3.4.1.	Analisa Kebutuhan	31
3.4.1.1.	Analisis Kebutuhan Bisnis	31
3.4.1.2.	Analisis Kebutuhan Fungsional	32
3.4.1.3.	Analisis Kebutuhan Non-Fungsional	32
3.4.2.	Perencanaan	32
3.4.2.1.	Rancangan <i>Class Diagram</i>	32
3.4.2.2.	Rancangan <i>Use Case Diagram</i>	33
3.4.2.3.	Rancangan ERD (<i>Entity Relationship Diagram</i>)	37
3.4.2.4.	Rancangan Tampilan Sistem (<i>Interface</i>)	37
3.4.3.	Implementasi	46
3.4.3.1.	Implementasi <i>Library React.js</i>	46
3.4.3.2.	Implementasi <i>Node.js</i>	46
3.4.4.	Pengujian	46
3.4.4.1.	Fungsionalitas Sistem	47
3.4.4.2.	Performa dan Beban Sistem	48
3.4.4.3.	Intuitivitas Antarmuka Sistem	49
3.4.5.	Pemeliharaan	51
3.4.6.	Penulisan Laporan	51
IV.	HASIL DAN PEMBAHASAN	52
4.1.	Hasil	52
4.2.	Pembahasan	53
4.2.1.	Implementasi <i>Node.js (server-side)</i>	53

4.2.2.	Implementasi React.js (<i>client-side</i>).....	55
4.2.3.	Pengujian Sistem	70
4.2.3.1.	Pengujian Fungsionalitas	70
4.2.3.2.	Pengujian Performa dan Beban.....	81
4.2.3.4.	Pengujian Intuitivitas Antarmuka	83
4.2.3.	Pemeliharaan Sistem.....	85
4.2.4.	Penulisan Laporan	85
V.	SIMPULAN DAN SARAN.....	86
5.1.	Simpulan	86
5.2.	Saran.....	87
	DAFTAR PUSTAKA.....	88
	LAMPIRAN	91

DAFTAR TABEL

Tabel	Halaman
1. Penelitian Terdahulu.....	19
2. Elemen Use Case Diagram	25
3. Elemen Use Case Diagram	26
4. Skenario Testing Registrasi	47
5. Skenario Testing Autentikasi.....	47
6. Skenario Testing Fitur Sistem	48
7. Perancangan Pengujian Performa dan Beban Sistem	49
8. Rancangan Pertanyaan Kuesioner Pengujian Intuitivitas Antarmuka Sistem	49
9. Bobot Jawaban Kuisisioner	50
10. Kriteria Interpretasi Interval	51
11. Skenario Testing Registrasi Sistem	71
12. Skenario Testing Autentikasi dan Otorisasi	72
13. Skenario Testing Fitur Sistem Supply Chain Management Barang Rongsok	73
14. Hasil pengujian performa dan beban	82
15. Hasil Kuisisioner Pengujian Intuitivitas Antarmuka.....	83
16. Analisis Intuitivitas Antarmuka.....	84

DAFTAR GAMBAR

Gambar	Halaman
1. Tahapan Penelitian.....	31
2. Rancangan Class Diagram.....	33
3. Rancangan Use Case Diagram (Customer)	34
4. Rancangan Use Case Diagram (Mitra).....	35
5. Rancangan Use Case Diagram (Admin).....	36
6. Rancangan ER Diagram	37
7. Rancangan Tampilan Homepage.....	38
8. Rancangan Tampilan Register Screen.....	38
9. Rancangan Tampilan Login Screen.....	39
10. Rancangan Tampilan Product Screen.....	40
11. Rancangan Tampilan Forum Screen Barang Rongsok.....	40
12. Rancangan Tampilan Forum Screen Mitra Barang Rongsok.....	41
13. Rancangan Tampilan Konfigurasi Profil Customer	41
14. Rancangan Tampilan Konfigurasi Mendaftar Menjadi Mitra Customer.....	42
15. Rancangan Tampilan Konfigurasi Profil Mitra	42
16. Rancangan Tampilan Konfigurasi Produk Mitra.....	43
17. Rancangan Tampilan Profil Mitra	43
18. Rancangan Tampilan Konfigurasi User.....	44
19. Rancangan Tampilan Konfigurasi Products	44
20. Rancangan Tampilan Konfigurasi Mitra	45
21. Rancangan Tampilan Konfigurasi Forum	45
22. Desain Database.....	54
23. Tampilan Homepage Screen.....	55
24. Tampilan Register Screen.....	56
25. Tampilan Login Screen.....	56
26. Tampilan Product Screen.....	57
27. Tampilan Forum Barang Rongsok.....	58

28. Tampilan Forum Mitra Barang Rongsok.....	58
29. Tampilan Konfigurasi Profile Customer	59
30. Tampilan Konfigurasi Mendaftar Menjadi Mitra	60
31. Tampilan Konfigurasi Profil Mitra	60
32. Tampilan Konfigurasi Produk Mitra	61
33. Tampilan Profil Mitra	62
34. Tampilan Daftar Order Customer	62
35. Tampilan Daftar Transaksi Mitra	63
36. Tampilan Dashboard Admin.....	64
37. Tampilan Konfigurasi User	64
38. Tampilan Konfigurasi Product	65
39. Tampilan Konfigurasi Transaksi	66
40. Tampilan Kongifurasi Mitra	66
41. Tampilan Konfigurasi Forum	67
42. Tampilan Keranjang Product.....	68
43. Tampilan Upload Transfer Receipt	68
44. Tampilan Upload Delivery Receipt.....	69
45. Tampilan Form Lupa Password.....	69
46. Tampilan Form Reset Password	70
47. Script Pengujian Performa dan Beban.....	81
48. Grafik Performance and Load Testing	82

I. PENDAHULUAN

1.1 Latar Belakang dan Masalah

Barang rongsok dalam Kamus Besar Bahasa Indonesia (KBBI) diketahui bahwa arti kata rongsok adalah suatu barang yang mana sudah dalam keadaan rusak sama sekali, sehingga sudah tidak dapat digunakan kembali sebagaimana mestinya, barang rongsok memiliki potensi untuk didaur ulang dan menjadi bahan baku untuk produk baru.

Manajemen dalam pengumpulan, pengolahan, dan distribusi barang rongsok yang kurang memadai mengakibatkan banyaknya limbah yang tidak terkelola dengan baik. Masalah ini perlu di berikan solusi yang intuitif agar bisa mendukung kegiatan manajemen limbah agar dapat terkelola dengan baik dan memberikan dampak positif bagi industri terkait.

Kota Bandar Lampung sebagai salah satu kota di Indonesia yang memiliki jumlah penduduk yang cukup besar, memiliki potensi besar dalam mengelola barang rongsok. Data Dinas Lingkungan Hidup (DLH) Provinsi Lampung menunjukkan tiap masyarakat rata-rata menghasilkan sampah sebanyak 0,45 kilogram/hari. Data DLH pada 2020 dituliskan bahwa timbunan sampah di Lampung mencapai 1.630.317,05 ton/tahun (Rahman, 2021).

Teknologi informasi dan internet yang saat ini sedang berkembang pesat, guna memanfaatkannya pengembangan sistem *supply chain management* barang rongsok berbasis web dapat menjadi solusi untuk mengoptimalkan manajemen barang rongsok.

Supply Chain Management adalah pengintegrasian sumber bisnis yang kompeten dalam penyaluran barang, mencakup perencanaan dan pengelolaan aktivitas pengadaan dan logistik serta informasi terkait mulai dari tempat bahan baku sampai tempat konsumsi, termasuk koordinasi dan kolaborasi dengan jaringan mitra usaha (pemasok, manufaktur, retailer, distributor, pergudangan, transportasi dan konsumen) untuk memenuhi kebutuhan pelanggan (Septiar, Fajar, & Darmawan, 2020).

Dalam pengumpulan, pengolahan, dan distribusi barang rongsok, masih banyak terjadi masalah seperti kurangnya koordinasi antar pemulung, pedagang, dan pengolahan barang rongsok, sehingga banyak barang rongsok yang tidak terkumpul dan terbuang sia-sia. Selain itu, masalah lain yang terjadi adalah kurangnya informasi mengenai harga dan ketersediaan barang rongsok bagi pemulung dan pedagang. Hal ini menyebabkan kurangnya efisiensi dalam pengumpulan dan pengolahan barang rongsok, serta kesulitan dalam memenuhi permintaan pasar yang semakin tinggi terhadap produk-produk yang terbuat dari barang rongsok.

Dalam konteks ini, pengembangan sistem *supply chain management* barang rongsok berbasis web dapat menjadi solusi untuk mengatasi masalah tersebut dengan menyediakan platform yang terintegrasi untuk menghubungkan seluruh pelaku dalam rantai pasok barang rongsok, mulai dari pemulung hingga pengolahan barang rongsok, dengan tujuan untuk meningkatkan efisiensi dan efektivitas manajemen barang rongsok secara keseluruhan. Dengan adanya sistem ini, diharapkan dapat mempermudah koordinasi antar pelaku dalam rantai pasok barang rongsok, meningkatkan efisiensi dalam pengumpulan dan pengolahan barang rongsok, serta memberikan informasi yang akurat mengenai harga dan ketersediaan barang rongsok bagi pemulung dan pedagang.

Pengembangan sistem *supply chain management* barang rongsok ini akan dibangun berbasis web dalam hal pembangunan ini didukung oleh kerangka kerja bernama

React.js dan *Node.js*. *React.js* adalah sebuah *library JavaScript* yang digunakan untuk membangun *user interface* pada *website*. *React.js* dikembangkan oleh *Facebook* dan menjadi salah satu teknologi populer di kalangan *developer web* karena kemampuannya dalam membuat aplikasi *web* yang dinamis dan mudah untuk dikelola (React.js, 2023). *Node.js* merupakan sebuah *platform server-side* yang dibangun di atas *engine JavaScript V8* milik *Google*. *Node.js* memungkinkan pengembang web untuk menjalankan *JavaScript* di sisi server, dan memberikan kemampuan untuk mengembangkan aplikasi *web real-time* dan *scalable*. *Node.js* menggunakan *event-driven, non-blocking I/O model*, yang memungkinkan aplikasi *Node.js* dapat menangani banyak permintaan secara efisien (Joyent, 2023).

Pemaparan di atas menunjukkan bahwa penelitian ini akan melakukan Pengembangan Sistem *Supply Chain Management* Barang Rongsok Berbasis *Web* dengan Studi Kasus Kota Bandar Lampung untuk menjawab permasalahan yang telah dipaparkan.

1.2. Rumusan Masalah

Rumusan masalah yang bisa disimpulkan untuk penelitian ini yaitu bagaimana mengembangkan Sistem *Supply Chain Management* Barang Rongsok berbasis *web* untuk memenuhi pengumpulan dan distribusi barang rongsok.

1.3. Batasan Masalah

Penelitian ini dibatasi oleh beberapa hal, yaitu:

1. Sistem *Supply Chain Management* ini hanya untuk pengumpulan dan pendistribusian barang rongsok.
2. Sistem *Supply Chain Management* ini hanya menyediakan fitur pembelian, penjualan dan pencatatan mitra bahan baku produk daur ulang.
3. Sistem *Supply Chain Management* ini menggunakan *library React.js* untuk pengembangan *front-end*.

4. Sistem *Supply Chain Management* ini menggunakan *runtime-environment* Node.js untuk pengembangan *back-end*.
5. Sistem *Supply Chain Management* ini hanya mencakup pengembangan *web*, bukan aplikasi *mobile*.
6. Sistem *Supply Chain Management* ini tidak mencakup *payment gateway* untuk mendukung transaksi dalam jaringan.

1.4. Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah untuk mengembangkan sistem untuk pengumpulan dan pendistribusian barang rongsok di Kota Bandar Lampung guna mendukung industri daur ulang barang rongsok.

1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut.

1. Memberikan alternatif solusi untuk mengembangkan sistem *supply chain management* barang rongsok yang lebih modern dan efisien dengan menggunakan teknologi terkini.
2. Memberikan panduan dan referensi bagi *developer* atau pengembang yang ingin mengembangkan sistem *supply chain management* serupa dengan teknologi React.js, Node.js, dan PostgreSQL.
3. Menambah pengetahuan dan pemahaman mengenai penggunaan teknologi React.js, Node.js, dan PostgreSQL dalam pengembangan aplikasi web.
4. Memberikan manfaat bagi masyarakat yang ingin menjual, mengelola, dan mendistribusikan barang rongsok dengan lebih mudah dan efisien melalui sistem *supply chain management* yang dikembangkan.
5. Memberikan manfaat untuk industri daur ulang barang rongsok.

II. TINJAUAN PUSTAKA

2.1. *Supply Chain Management (SCM)*

Supply Chain Management (SCM) adalah suatu mekanisme yang melibatkan perencanaan, pengendalian, dan pengelolaan aliran barang, informasi, dan jasa dari titik awal produksi hingga titik konsumsi (Mentzer, et al., 2011). Komponen-komponen utama dalam SCM meliputi perencanaan, pengendalian produksi, pengadaan bahan baku, manajemen persediaan, transportasi, dan penyaluran produk (Mentzer, et al., 2011). Perencanaan produksi melibatkan penentuan jadwal produksi, kapasitas produksi, dan alokasi sumber daya. Pengadaan bahan baku melibatkan identifikasi pemasok, negosiasi kontrak, dan pemantauan kualitas. Manajemen persediaan mencakup pengendalian stok, peramalan permintaan, dan pengaturan reorder. Transportasi melibatkan pengiriman barang dari pemasok ke produsen, dari produsen ke distributor, dan dari distributor ke konsumen. Penyaluran produk melibatkan distribusi fisik dan manajemen gudang.

Proses dalam SCM meliputi perencanaan, pengadaan, produksi, pengiriman, dan pengembalian. Perencanaan melibatkan peramalan permintaan, penentuan kebutuhan persediaan, dan perencanaan kapasitas (Mentzer, et al., 2011). Pengadaan melibatkan pemilihan pemasok, negosiasi kontrak, dan pengadaan bahan baku. Produksi melibatkan manajemen operasi, kontrol kualitas, dan pengelolaan tenaga kerja. Pengiriman melibatkan logistik, transportasi, dan manajemen persediaan. Pengembalian melibatkan penanganan produk yang dikembalikan, perbaikan, dan pengembalian dana.

Manfaat SCM meliputi peningkatan efisiensi operasional, pengurangan biaya, peningkatan kepuasan pelanggan, peningkatan kualitas produk dan peningkatan responsibilitas rantai pasokan. Dengan menerapkan SCM yang baik, perusahaan dapat mengoptimalkan proses bisnis, mengurangi waktu siklus, menghindari kekurangan stok atau kelebihan persediaan, dan meningkatkan koordinasi antara mitra bisnis.

Strategi dalam SCM mencakup pengelolaan risiko, kolaborasi dengan mitra bisnis, peningkatan fleksibilitas, penggunaan teknologi informasi, dan peningkatan keberlanjutan (Asyahdina, Krisnanik, & Wirawan, 2021). Pengelolaan risiko melibatkan identifikasi, penilaian, dan mitigasi risiko yang mungkin timbul dalam rantai pasokan. Kolaborasi dengan mitra bisnis melibatkan kerjasama dalam perencanaan, pengadaan, produksi, dan distribusi. Peningkatan fleksibilitas melibatkan adaptasi cepat terhadap perubahan permintaan atau kondisi pasar. Penggunaan teknologi informasi melibatkan implementasi sistem informasi terintegrasi untuk meningkatkan visibilitas dan koordinasi. Peningkatan keberlanjutan melibatkan penerapan praktik yang ramah lingkungan dan sosial.

Praktik dalam SCM meliputi pemantauan kinerja dan pengukuran hasil. Pemantauan kinerja melibatkan pengukuran dan evaluasi terhadap kinerja operasional dan kepatuhan terhadap target. Pengukuran hasil melibatkan pengukuran efektivitas dan efisiensi dari implementasi SCM.

2.1.1. *Supply Chain Management (SCM) Barang Rongsok*

Industri barang rongsok, juga dikenal sebagai industri daur ulang atau limbah, telah menjadi salah satu sektor yang berkembang pesat dalam beberapa tahun terakhir. Industri ini melibatkan pengumpulan, pengolahan, dan penjualan kembali barang rongsok seperti logam, kertas, plastik, kaca, dan bahan lainnya. Dalam konteks *supply chain management*, industri barang rongsok merupakan bagian penting dari rantai pasok yang bertujuan untuk mengoptimalkan penggunaan kembali dan daur ulang bahan-bahan tersebut.

Dalam konteks industri barang rongsok, SCM menjadi krusial untuk mengelola pergerakan barang rongsok dari sumber-sumbernya hingga ke pelanggan akhir dengan cara yang efektif dan efisien. SCM dalam industri barang rongsok memperhatikan aspek pengadaan barang rongsok dari pengepul, klasifikasi dan penilaian barang rongsok, manajemen persediaan, serta pendistribusian barang rongsok kepada pelanggan atau pihak ketiga. Melalui penerapan SCM yang baik, diharapkan dapat mencapai efisiensi, transparansi, dan memenuhi kebutuhan dalam rantai pasok barang rongsok.

Supply chain dalam industri barang rongsok dimulai dengan pengumpulan barang rongsok dari berbagai sumber, termasuk pengepul, pedagang, perusahaan, dan individu. Proses pengumpulan ini melibatkan kegiatan seperti pemilahan, pemisahan, dan pengangkutan barang rongsok ke pusat pengolahan. Pusat pengolahan berperan dalam menjalankan proses pengolahan, yaitu membersihkan, mendaur ulang, dan mengubah barang rongsok menjadi bahan baku yang dapat digunakan kembali.

Selanjutnya, bahan rongsok yang telah diolah akan didistribusikan ke berbagai pelanggan atau pabrik yang membutuhkan. Pada tahap ini, sistem supply chain management berperan penting dalam memastikan kelancaran dan efisiensi proses distribusi. SCM barang rongsok dengan fokus pada pengumpulan, pengolahan, dan distribusi barang rongsok memiliki peran penting untuk keberlangsungan industri barang rongsok. Penelitian ini akan menghasilkan sistem *supply chain management* yang diharapkan memberikan peningkatan proses pengumpulan dan pendistribusian, dengan cara memberikan sebuah wadah kepada aktor-aktor dalam rantai pasok barang rongsok memenuhi kebutuhan pasok mereka. Menghubungkan satu aktor dengan aktor lainnya dalam suatu wadah agar bisa saling memenuhi pasokan yang mereka punya dengan cara jual-beli pasok untuk dijadikan sebuah produk daur ulang.

Manajemen rantai pasok yang efektif dan berkelanjutan menjadi kunci untuk mengoptimalkan penggunaan kembali dan daur ulang bahan rongsok. Pengelolaan yang baik mengakibatkan industri barang rongsok dapat berkontribusi pada pengurangan limbah, penghematan sumber daya, dan pembangunan ekonomi yang berkelanjutan.

2.2. Sistem Informasi

Sistem Informasi (SI) merupakan suatu sistem yang terdiri dari elemen-elemen yang saling berinteraksi dan bekerja bersama-sama untuk mengumpulkan, mengelola, menyimpan, mengolah, dan menyajikan informasi guna mendukung pengambilan keputusan dan kegiatan operasional suatu organisasi (Wijoyo, Ariyanto, Sudarsono, & Dwi, 2021).

Sistem informasi terdiri dari beberapa komponen penting, yaitu input (masukan), proses, output (keluaran), dan mekanisme umpan balik (feedback). Komponen input merupakan data dan informasi yang diambil dari berbagai sumber, baik itu dari pengguna, perangkat keras, atau sistem lainnya. Proses merupakan tahapan di mana data diolah dan diubah menjadi informasi yang berguna melalui penggunaan algoritma, logika, dan perhitungan. Keluaran adalah hasil yang dihasilkan dari proses tersebut, berupa laporan, grafik, atau tampilan informasi yang dapat dimanfaatkan oleh pengguna.

Manfaat sistem informasi berupa peningkatan efisiensi operasional, pengambilan keputusan, dan komunikasi. Operasional yang dapat ditingkatkan seperti melakukan otomatisasi dalam suatu proses bisnis untuk mengurangi kesalahan yang dilakukan oleh manusia. Pengambilan keputusan berdasarkan data yang ada didalam sistem informasi adalah data yang relevan dan akurat untuk pengambilan keputusan yang lebih baik dan cepat. Komunikasi tim antar departemen pun menjadi lebih baik mengenai proses bisnis yang mereka jalankan.

2.3. Sistem Informasi Berbasis *Web*

Sistem Informasi Berbasis Web adalah suatu sistem yang menggunakan teknologi web untuk mengumpulkan, mengelola, menyimpan, dan menyajikan informasi secara efisien dan terintegrasi (Ahmad, 2018).

2.3.1. Teknologi *Web*

Teknologi *web* merujuk pada kumpulan alat, protokol, bahasa pemrograman, dan standar yang digunakan untuk membangun, mengembangkan, dan menyajikan aplikasi dan konten melalui internet (Asyahdina, Krisnanik, & Wirawan, 2021). Teknologi *web* memainkan peran penting dalam memfasilitasi interaksi dan pertukaran informasi antara pengguna yang terhubung ke jaringan. Ini mencakup berbagai komponen dan konsep, termasuk bahasa *markup*, *scripting*, protokol komunikasi, dan platform pengembangan.

Bahasa markup seperti *HyperText Markup Language* (HTML) digunakan untuk membuat struktur dan tampilan halaman *web*. Selain itu, teknologi *web* juga melibatkan protokol komunikasi seperti *HyperText Transfer Protocol* (HTTP) dan Secure HTTP (HTTPS) yang digunakan untuk mentransfer data antara *server web* dan klien *web*. Protokol ini memungkinkan permintaan dan respon antara *browser web* pengguna dan *server* yang menyimpan dan mengelola konten *web*.

2.3.1.1. HTML

HTML (*HyperText Markup Language*) adalah bahasa markup standar yang digunakan untuk membangun struktur dan konten halaman *web* (Ahmad, 2018). Sebagai bahasa markup, HTML memungkinkan pengembang *web* untuk menentukan elemen-elemen dan hierarki yang membentuk halaman *web*.

Setiap elemen dalam HTML dikelilingi oleh tag, yang memberi tahu *browser web* cara menampilkan dan memanipulasi konten tersebut. HTML menggunakan sintaks yang terdiri dari elemen, atribut, dan nilai.

Elemen HTML digunakan untuk mengatur dan menampilkan konten seperti teks, gambar, video, tautan, tabel, formulir, dan banyak lagi. Setiap elemen memiliki struktur dan makna tersendiri, seperti elemen `<p>` untuk paragraf, `<h1>` hingga `<h6>` untuk judul dengan tingkat kepentingan yang berbeda, `` untuk gambar, `<a>` untuk tautan, dan seterusnya.

HTML juga menggunakan atribut yang memberikan informasi tambahan tentang elemen tersebut. Atribut digunakan untuk mengatur sifat dan perilaku dari elemen HTML. Contoh atribut yang umum digunakan adalah atribut "src" untuk menentukan sumber gambar pada elemen ``, atribut "href" untuk menentukan URL tujuan pada elemen `<a>`, dan atribut "class" atau "id" untuk memberikan identifikasi atau gaya khusus pada elemen.

2.3.1.2. Bahasa Pemrograman

Bahasa pemrograman dalam konteks teknologi web adalah sekumpulan instruksi yang digunakan untuk mengembangkan aplikasi web dan memberikan fungsionalitas pada halaman web (Gusti, Akbar, & Indah, 2019). Ada berbagai bahasa pemrograman yang digunakan dalam pengembangan *web*, dan setiap bahasa memiliki fitur, sintaksis, dan kegunaan yang berbeda.

Salah satu bahasa pemrograman yang paling umum digunakan dalam pengembangan web adalah *JavaScript*. *JavaScript* adalah bahasa pemrograman yang berjalan di sisi klien (*client-side*) dan digunakan untuk memberikan interaktivitas pada halaman *web* (Sianipar, 2015). Dengan *JavaScript*, pengembang *web* dapat membuat efek visual yang dinamis, mengontrol perilaku elemen, mengirim permintaan ke *server* secara *asynchronous*, serta mengelola dan memanipulasi data. Selain *JavaScript*, ada juga bahasa pemrograman seperti *PHP*, *Python*, dan *Ruby* yang sering digunakan untuk mengembangkan sisi *server* (*server-side*) dari aplikasi *web*. Bahasa-bahasa ini digunakan untuk memproses permintaan dari klien, mengakses basis data, melakukan logika bisnis, dan menghasilkan halaman *web* dinamis yang dikirimkan kepada pengguna.

Selain bahasa pemrograman yang spesifik untuk sisi klien atau sisi *server*, ada juga kerangka kerja (*framework*) yang mempermudah pengembangan *web* dengan menyediakan fungsi dan komponen yang siap pakai. Contohnya adalah React.js dan Angular.js yang menggunakan bahasa JavaScript, Django yang menggunakan bahasa Python, atau Laravel yang menggunakan bahasa PHP.

Kerangka kerja ini mempercepat proses pengembangan, memungkinkan pengembang untuk bekerja dengan konvensi yang sudah mapan, dan menyediakan fitur-fitur yang berguna seperti manajemen keadaan (*state management*), routing, validasi formulir, dan banyak lagi. Dalam penelitian ini Bahasa pemrograman yang digunakan yaitu JavaScript dan kerangka kerja React.js.

2.3.1.2.1. Javascript

JavaScript adalah bahasa pemrograman tingkat tinggi yang berjalan di sisi klien (*client-side*) dan digunakan untuk mengembangkan aplikasi web yang interaktif. Dengan *JavaScript*, pengembang dapat mengontrol perilaku elemen *HTML*, mengatur logika, dan berkomunikasi dengan server secara asinkron (Sianipar, 2015). Menurut Sianipar (2015) *JavaScript* pertama kali diperkenalkan oleh Brendan Eich di Netscape Communications Corporation pada tahun 1995, pada awalnya bahasa tersebut disebut "*LiveScript*," tetapi kemudian diubah namanya menjadi "*JavaScript*" untuk mencerminkan popularitas bahasa pemrograman *Java* pada saat itu setelah itu pada tahun 1996, *JavaScript* diserahkan ke European Computer Manufacturers Association (ECMA) untuk diadopsi sebagai standar internasional.

Standar tersebut kemudian dikenal sebagai ECMAScript, ECMAScript dalam sejarahnya memiliki banyak versi rilis seperti ES3, ES5, ES6 setelah ES6 ECMAScript mulai mengadopsi siklus tahunan dengan masing masing versi sesuai tahun rilisnya seperti ECMAScript 2016, 2017, dan 2018.

2.3.1.2.2. React.js

React.js adalah *library JavaScript* yang digunakan untuk membangun antarmuka pengguna (UI) dalam aplikasi *web*. Dengan menggunakan konsep komponen, React memungkinkan pengembang untuk membagi tampilan aplikasi menjadi bagian-bagian terpisah yang dapat diatur dan dikelola secara mandiri. *React* juga menggunakan *Virtual DOM (Document Object Model)* untuk mengoptimalkan efisiensi pembaruan tampilan (Ahmad, 2018).

Keunggulan *React.js* yaitu menyediakan pendekatan yang terstruktur dan efisien dalam mengembangkan aplikasi *web*. Dengan menggunakan komponen yang dapat digunakan ulang, pengembang dapat memecah tampilan menjadi bagian-bagian yang lebih kecil, membuat kode lebih mudah dikelola dan dipelihara.

2.3.1.2.3. Node.js

Node.js adalah platform *runtime* yang berbasis *JavaScript* yang memungkinkan pengembang untuk menjalankan kode *JavaScript* di sisi *server* (Wijaya & Akbar, 2019). *Node.js* memanfaatkan model I/O non-blokir, yang memungkinkan banyak permintaan dapat ditangani secara efisien dalam satu proses, sehingga meningkatkan throughput dan kinerja aplikasi.

Node.js juga dilengkapi dengan kumpulan modul bawaan yang dikenal sebagai "*Node.js Core Modules*", yang menyediakan berbagai fungsi dan fitur yang dapat digunakan dalam pengembangan aplikasi, seperti modul HTTP untuk membangun *server web*, modul *File System* untuk mengelola *file*, modul *Path* untuk manipulasi jalur *file*, dan banyak lagi.

2.3.1.3. Sistem Interaksi

Sistem interaksi dalam teknologi web merujuk pada cara pengguna berinteraksi dengan halaman web dan bagaimana halaman web memberikan respons terhadap tindakan pengguna (Setiyani, 2021).

Sistem interaksi ini memainkan peran penting dalam menciptakan pengalaman pengguna yang baik dan fungsionalitas yang optimal pada aplikasi web.

Salah satu aspek penting dari sistem interaksi adalah antarmuka pengguna (*user interface*) yang memungkinkan pengguna berinteraksi dengan halaman *web*. Antarmuka pengguna melibatkan elemen-elemen seperti tombol, formulir, tautan, menu, dan elemen interaktif lainnya yang memungkinkan pengguna memasukkan data, melakukan aksi, atau berpindah antara halaman. Desain antarmuka pengguna yang baik mempertimbangkan kejelasan, usability, dan kesesuaian dengan tujuan aplikasi *web*. Selain antarmuka pengguna, sistem interaksi juga melibatkan interaksi yang terjadi antara halaman web dan pengguna melalui perangkat input seperti *keyboard*, *mouse*, layar sentuh, dan perangkat masukan lainnya. Pengguna dapat berinteraksi dengan elemen halaman *web*, memasukkan data, memilih pilihan, mengklik tautan, atau menggulir halaman menggunakan perangkat *input* tersebut. Sistem interaksi harus responsif dan memberikan umpan balik yang jelas kepada pengguna agar mereka dapat memahami dan mengendalikan aksi yang mereka lakukan.

Selain itu, sistem interaksi juga melibatkan interaksi antara halaman *web* dengan *server* atau sumber daya eksternal. Misalnya, pengiriman data melalui formulir, permintaan data ke *server* melalui permintaan HTTP, pengiriman *email*, atau integrasi dengan layanan pihak ketiga. Sistem interaksi yang baik harus dapat mengelola dan memproses interaksi ini dengan aman, efisien, dan handal.

2.3.1.4. Database Management System (DBMS)

Sistem manajemen basis data (*Database Management System/DBMS*) berperan dalam mengelola dan menyimpan data yang dibutuhkan oleh sistem. Data-data tersebut disimpan dalam basis data yang terentralisasi, yang dapat diakses oleh pengguna melalui antarmuka web. DBMS menyediakan mekanisme untuk melakukan manipulasi data, pemeliharaan, serta mengelola keamanan dan integritas data.

DBMS (*Database Management System*) adalah perangkat lunak yang digunakan untuk mengelola dan mengorganisir data dalam sebuah sistem informasi. DBMS bertanggung jawab atas penyimpanan, pengambilan, pembaruan, dan penghapusan data dari database (Christono & Sama, 2020). Dalam konteks sistem informasi berbasis *web*, DBMS memiliki peran yang sangat penting dalam memfasilitasi akses dan manipulasi data secara efisien.

Salah satu fungsi utama DBMS adalah menyediakan struktur penyimpanan yang terorganisir, seperti tabel, relasi, atau objek, yang memungkinkan data disimpan dengan format yang konsisten dan terstruktur. DBMS juga menawarkan kemampuan pengindeksan, pengaturan keamanan, dan konstrain data untuk memastikan integritas data dan akses yang aman.

DBMS menyediakan antarmuka atau bahasa yang memungkinkan pengguna atau aplikasi berinteraksi dengan database. Bahasa ini biasanya berupa SQL (*Structured Query Language*), yang digunakan untuk mengambil data, memasukkan data baru, memperbarui data yang ada, dan menghapus data. DBMS juga mendukung transaksi, yang memungkinkan operasi pengolahan data yang kompleks dan aman, termasuk rollback dan commit.

2.3.1.4.1. PostgreSQL

PostgreSQL adalah sebuah DBMS (*Database Management System*) yang populer dan open-source yang menawarkan kemampuan canggih untuk mengelola data dalam sistem informasi berbasis *web*. PostgreSQL didasarkan pada model basis data relasional, yang memungkinkan penyimpanan dan pengambilan data dengan menggunakan tabel, relasi, dan kueri SQL (*Structured Query Language*) (Christono & Sama, 2020).

Salah satu keunggulan utama PostgreSQL adalah kemampuan skala dan performanya yang tinggi. PostgreSQL dirancang untuk mengelola *volume* data yang besar dengan efisiensi dan ketahanan tinggi. Dalam konteks sistem informasi berbasis web, hal ini sangat penting untuk mendukung operasi yang cepat dan responsif, terutama dalam pengelolaan data supply chain management barang rongsok yang melibatkan berbagai entitas seperti pemasok, pelanggan, dan transaksi.

PostgreSQL menyediakan fitur-fitur canggih yang mendukung kebutuhan pengembangan aplikasi web yang kompleks. Fitur-fitur ini mencakup dukungan untuk transaksi ACID (*Atomicity, Consistency, Isolation, Durability*) yang memastikan integritas dan konsistensi data, kemampuan indexing yang kuat untuk pencarian dan pengurutan data, dan dukungan untuk kueri kompleks yang memungkinkan manipulasi dan analisis data yang fleksibel.

PostgreSQL juga menawarkan keamanan yang tinggi melalui fitur autentikasi, otorisasi, dan enkripsi data. DBMS ini mendukung pengelolaan pengguna, peran, dan hak akses yang terperinci, sehingga memungkinkan pengaturan tingkat keamanan yang tepat untuk melindungi data sensitif dalam sistem supply chain management barang rongsok.

2.3.1.4.2. SQL

SQL (*Structured Query Language*) adalah bahasa pemrograman yang digunakan untuk mengelola dan mengoperasikan basis data dalam sistem informasi berbasis web (Christono & Sama, 2020). SQL adalah bahasa standar yang digunakan secara luas untuk berinteraksi dengan DBMS (*Database Management System*), termasuk DBMS seperti PostgreSQL, MySQL, dan Oracle.

SQL memungkinkan pengguna atau pengembang aplikasi untuk melakukan berbagai operasi terhadap basis data, seperti pengambilan data, pembaruan data, penghapusan data, dan penambahan data baru.

Bahasa ini menyediakan sintaks yang konsisten dan terstruktur untuk menyusun perintah-perintah yang dieksekusi oleh DBMS. Penggunaan SQL dalam sistem informasi berbasis web memungkinkan manipulasi dan pengambilan data dengan cara yang efisien dan fleksibel. Penggunaan SQL sistem informasi berbasis web dapat melakukan operasi pengambilan data yang kompleks, pembaruan data, dan manipulasi data lainnya dengan mudah dan efisien. Bahasa ini merupakan fondasi penting dalam pengembangan aplikasi web yang berhubungan dengan basis data.

2.4. Barang Rongsok

2.4.1. Definisi

Barang rongsok, juga dikenal sebagai barang bekas atau limbah, merujuk pada barang atau material yang sudah tidak lagi digunakan, dianggap tidak bernilai, dan akan dibuang atau didaur ulang (Pratama, Sholva, & Azhar, 2023). Barang rongsok dapat berasal dari berbagai sumber, seperti rumah tangga, industri, atau lembaga lainnya. Contoh barang rongsok meliputi kertas bekas, logam bekas, elektronik bekas, plastik bekas, dan banyak lagi.

2.4.2. Masalah

Pengelolaan barang rongsok dalam konteks supply chain management menghadapi beberapa masalah yang perlu diatasi. Salah satu masalah yang dihadapi adalah kurangnya infrastruktur yang memadai untuk pengumpulan, pengolahan, dan distribusi barang rongsok. Seringkali, barang rongsok masih terfragmentasi dan kurang terorganisir, sehingga menghambat efisiensi dan efektivitas dalam proses pengelolaan.

Masalah lainnya adalah kurangnya kesadaran dan partisipasi masyarakat dalam mengelola barang rongsok. Banyak masyarakat yang belum sepenuhnya memahami pentingnya mendaur ulang atau memanfaatkan kembali barang rongsok, sehingga cenderung membuangnya ke tempat pembuangan akhir yang

tidak tepat. Kurangnya kesadaran ini juga berdampak pada pemilahan dan pemisahan limbah, sehingga menghambat proses pengolahan dan pengumpulan yang lebih efisien.

2.4.3. Potensi Ekonomi

Kota Bandar Lampung memiliki potensi ekonomi yang signifikan terkait dengan pengelolaan barang rongsok. Sebagai salah satu kota besar di Indonesia, Kota Bandar Lampung menghasilkan jumlah barang rongsok yang cukup besar setiap harinya yaitu sebesar 0,45 kilogram/hari, pada tahun 2020 tercatat pada Dinas Lingkungan Hidup Provinsi Lampung timbunan sampah di Lampung mencapai 1.630.317,05 ton/tahun (Rahman, 2021). Barang rongsok yang dihasilkan meliputi berbagai jenis, seperti kertas bekas, logam bekas, plastik bekas, dan elektronik bekas. Potensi ekonomi barang rongsok dapat dimanfaatkan melalui beberapa aspek. Pertama, ada peluang untuk melakukan pengumpulan dan pemilahan barang rongsok yang dilakukan secara efisien dan terorganisir. Dengan pendekatan yang tepat, barang rongsok dapat dikumpulkan dari sumber-sumber yang berbeda, seperti rumah tangga, industri, atau lembaga lainnya. Pemilahan juga penting untuk memisahkan jenis barang rongsok sesuai dengan nilai ekonominya.

Selanjutnya, terdapat peluang dalam proses pengolahan atau daur ulang barang rongsok. Barang rongsok yang telah dikumpulkan dan dipilah dapat diolah menjadi bahan baku yang bernilai ekonomi, seperti kertas daur ulang, logam daur ulang, atau plastik daur ulang. Proses pengolahan yang efisien dan berkualitas dapat meningkatkan nilai ekonomi barang rongsok dan membuka peluang bisnis dalam industri daur ulang.

Selain itu, potensi ekonomi juga terkait dengan pemanfaatan kembali barang rongsok dalam bentuk yang masih dapat digunakan. Beberapa barang rongsok dapat mengalami perbaikan atau pemulihan sehingga dapat digunakan kembali.

Misalnya, perbaikan dan *refurbishing* pada barang elektronik bekas dapat menghasilkan produk yang siap digunakan kembali dengan nilai ekonomi yang lebih rendah daripada barang baru.

Potensi ekonomi barang rongsok di Kota Bandar Lampung dapat memberikan manfaat yang signifikan. Selain memberikan peluang usaha dan lapangan kerja, pengelolaan barang rongsok yang baik juga dapat mengurangi beban lingkungan dan membantu mengurangi penggunaan sumber daya alam yang terbatas. Dalam konteks sistem *supply chain management* barang rongsok berbasis web, potensi ekonomi ini dapat dioptimalkan melalui penggunaan teknologi informasi untuk menghubungkan berbagai pelaku dalam rantai pasokan yaitu meningkatkan efisiensi operasional.

2.4.4. Manfaat

Pengelolaan barang rongsok di Kota Bandar Lampung memiliki sejumlah manfaat yang dapat dirasakan oleh berbagai pihak. Pengelolaan yang baik dan efektif terhadap barang rongsok dapat memberikan manfaat ekonomi yang signifikan. Dengan memanfaatkan potensi barang rongsok yang dihasilkan setiap harinya, pelaku bisnis dan industri daur ulang dapat menciptakan lapangan kerja, menghasilkan produk yang bernilai ekonomi, serta meningkatkan pertumbuhan ekonomi lokal.

Selain manfaat ekonomi, pengelolaan barang rongsok juga berdampak positif terhadap lingkungan. Dalam Kota Bandar Lampung, pengurangan limbah melalui pengelolaan barang rongsok dapat membantu mengurangi beban pada tempat pembuangan akhir dan mengurangi dampak negatif terhadap lingkungan. Proses daur ulang barang rongsok juga membantu mengurangi penggunaan sumber daya alam yang terbatas dan mengurangi polusi yang dihasilkan dari produksi barang baru. Manfaat lainnya adalah aspek sosial dan masyarakat. Pengelolaan barang rongsok yang melibatkan partisipasi aktif masyarakat dapat meningkatkan kesadaran akan pentingnya mendaur ulang dan menjaga kebersihan lingkungan. Selain itu, dengan adanya pengelolaan yang terorganisir, masyarakat dapat memiliki akses yang lebih mudah untuk memanfaatkan barang rongsok yang masih bisa digunakan, sehingga mengurangi beban finansial dalam memenuhi kebutuhan sehari-hari.

2.5. Penelitian Terdahulu

Penelitian terdahulu merupakan penelitian yang pernah dilakukan sebelumnya, berikut beberapa penelitian terkait pengembangan Sistem *Supply Chain Management* (SCM) sebagai bentuk kajian dasar yang memperkuat rangka penyusunan dengan penelitian yang dilakukan:

Tabel 1. Penelitian Terdahulu

No	Peneliti	Judul Penelitian	Metode Pengembangan	Hasil Penelitian
1.	(Septiar, Fajar, & Darmawan, 2020)	Implementasi <i>Supply Chain Management</i> (SCM) Toko Alat dan Bahan Bangunan Berbasis <i>Web</i> (Studi Kasus: TB. Bojong Indah)	<i>Waterfall</i>	Sistem Supply Chain Management (SCM) Toko Alat dan Bangunan Berbasis Web
2.	(Asyahdina, Krisnanik, & Wirawan, 2021)	Rancang Bangun <i>Supply Chain Management</i> Budidaya Jamur Berbasis <i>Web</i> (Studi Kasus: Budidaya Jamur Jatayutm)	<i>Prototyping</i>	Sistem Supply Chain Management Budidaya Jamur Berbasis Web

Tabel 1. (Lanjutan)

3. (Pratama, Sholva, & Azhar, 2023)	Aplikasi Jual Beli Barang Rongsok Berbasis Jarak Menggunakan Fitur <i>Location Based Service</i>	<i>Waterfall</i>	Aplikasi Jual Beli Barang Rongsok dengan Fitur <i>Location Based Service</i>
4. (Dicky & Kurniawan, 2022)	Sistem Informasi Pemantauan Penjualan Barang Rongsok Menggunakan <i>Framework CI.4</i> Pada CV. Sumber Baja	<i>Agile</i>	Sistem Informasi Pemantauan Penjualan Barang Rongsok Berbasis Web
5. (Supardiyono, Pertiwi, & Eridani, 2022)	Perancangan Sistem Informasi Pengelolaan Barang Rongsok Berbasis Web Menggunakan Kerangka Kerja CodeIgniter	<i>Waterfall</i>	Sistem Informasi Pengelolaan Barang Rongsok Berbasis Web

2.5.1. Implementasi Supply Chain Management (SCM) Toko Alat dan Bahan Bangunan Berbasis Web (Studi Kasus: TB. Bojong Indah)

Penelitian yang dilakukan Septiar, Fajar, dan Darmawan (2020) melakukan implementasi *Supply Chain Management* ke dalam basis *web* dan menggunakan model pengembangan *Waterfall* yang berfokus pada pengelolaan dan komunikasi pada supplier seperti informasi mengenai stock barang.

Penelitian ini bertujuan untuk meningkatkan efisiensi dan memaksimalkan pengelolaan dan komunikasi mengenai barang seperti alat dan bahan bangunan pada supplier dengan cepat antara keduanya.

2.5.2. Rancang Bangun Supply Chain Management Budidaya Jamur Berbasis Web (Studi Kasus: Budidaya Jamur Jatayutm)

Penelitian yang dilakukan oleh Asyahdina, Krisnanik, dan Wirawan (2021) merancang dan membangun Supply Chain Management (SCM) budidaya jamur berbasis web menggunakan model pengembangan *Prototyping* yang befokus pada pendataan panen jamur dan pencatatan pesanan jamur. Penelitian ini bertujuan untuk membantu membudidaya jamur tersebut kepada konsumen dan juga bisa mengintegrasikan supplier, distributor, dan pelanggan akhir dalam suatu sistem berbasis website.

2.5.3. Aplikasi Jual Beli Barang Rongsok Berbasis Jarak Menggunakan Fitur *Location Based Service*

Penelitian yang dilakukan oleh Pratama, Sholva, dan Azhar (2023) membangun aplikasi jual beli barang rongsok yang mengedepankan fitur *location based service* menggunakan metode pengembangan *waterfall* yang dimana tujuan dibangunnya aplikasi ini untuk mendekatkan penjual barang rongsok dan pembeli barang rongsok. Penelitian ini berfokus untuk mendekatkan penjual dan pembeli barang rongsok untuk memudahkan proses transaksi barang rongsok yang akan dijual.

2.5.4. Sistem Informasi Pemantauan Penjualan Barang Rongsok Menggunakan *Framework CI.4* Pada CV. Sumber Baja

Penelitian yang dilakukan Dicky dan Kurniawan (2022) membangun sistem informasi pemantauan penjualan barang rongsok menggunakan metode pengembangan *agile* berfokus pada pemantauan penjualan barang rongsok yang ada di CV. Sumber Baja.

Penelitian ini mengedepankan pencatatan penjualan barang rongsok yang ada di CV. Sumber Baja agar semua barang rongsok yang terjual dapat di pantau sehingga memudahkan untuk melihat barang rongsok yang sudah terjual.

2.5.5. Perancangan Sistem Informasi Pengelolaan Barang Rongsok Berbasis Web Menggunakan Kerangka Kerja CodeIgniter

Penelitian yang dilakukan Supardiyono, Pertiwim dan Eridani (2022) merancang sistem informasi pengelolaan barang rongsok menggunakan metode pengembangan *waterfall* mengedepankan pengelolaan barang rongsok yang sudah ada pengelolaan berupa klasifikasi barang atau pengelompokan barang. Penelitian ini berfokus untuk pengelolaan barang rongsok agar mudah dalam pendataan dan pendistribusian yang akan dilakukan, dibuktikan meningkatkan efisiensi dalam pengelolaan dan pendataan melalui sistem informasi ini.

2.6. Pengembangan Perangkat Lunak

Pengembangan perangkat lunak merupakan suatu proses yang terstruktur dan sistematis dalam menciptakan sebuah sistem atau aplikasi yang berfungsi sesuai dengan kebutuhan pengguna. Salah satu pendekatan yang sering digunakan dalam pengembangan perangkat lunak adalah *Software Development Life Cycle (SDLC)*, yang merupakan suatu kerangka kerja yang terdiri dari serangkaian tahapan yang harus dilalui dalam siklus hidup pengembangan perangkat lunak.

SDLC terdiri dari beberapa fase utama, yaitu analisis kebutuhan, perancangan, pengkodean, pengujian, dan implementasi. Pada fase analisis kebutuhan, tim pengembang perangkat lunak berinteraksi dengan pengguna atau pemangku kepentingan untuk memahami kebutuhan fungsional dan nonfungsional sistem yang akan dikembangkan.

Contoh metode SDLC yang umum di gunakan adalah, Model Waterfall, Model Agile, Model Spiral, *Rapid Application Development* (RAD), dan Prototyping. Masing masing metode digunakan dan ditujukan dengan kebutuhan dan tujuan tertentu dengan karakteristik masing masing metode.

Dalam pengembangan perangkat lunak pada penelitian ini digunakan metode Model Waterfall dikarenakan pendekatan yang secara linier dan berurutan dalam pengembangannya. Kebutuhan yang sudah dipastikan dari hasil identifikasi masalah dan perencanaan yang dilakukan membuat Model Waterfall ini cocok untuk pengembangan perangkat lunak dalam penelitian ini.

2.6.1. Metode *Waterfall*

Metode *waterfall* (air terjun) adalah salah satu pendekatan tradisional untuk pengembangan perangkat lunak yang mengikuti urutan langkah-langkah yang terstruktur dan linier (Wahid, 2020). Metode ini menggambarkan alur kerja pengembangan perangkat lunak sebagai air terjun yang mengalir secara berurutan dari satu fase ke fase berikutnya.

Metode ini melakukan pengembangan perangkat lunak yang berdasarkan tahap-tahap berurutan yang harus diselesaikan secara linear. Setiap tahap dalam metode ini harus selesai sebelum tahap berikutnya dimulai. Menurut Wahid (2020) tahap-tahap tersebut umumnya mencakup:

1. Analisa Kebutuhan:

Pada fase ini, kebutuhan pengguna dicatat dan dianalisis dan persyaratan yang harus dipenuhi oleh perangkat lunak yang dikembangkan ditentukan.

2. Perencanaan:

Pada fase ini, perancangan sistem secara keseluruhan dan arsitektur perangkat lunak dilakukan berdasarkan persyaratan yang telah ditetapkan. Hal ini meliputi perancangan struktur data, antarmuka pengguna, dan logika bisnis.

3. Implementasi:

Tahap ini melibatkan penerjemahan perancangan ke dalam kode program yang dapat dieksekusi oleh komputer. Program-program ini dikembangkan dan dites secara individu.

4. Pengujian:

Setelah implementasi, tahap pengujian dilakukan untuk memverifikasi dan validasi perangkat lunak. Berbagai jenis pengujian seperti pengujian fungsionalitas, integrasi, dan performa dilakukan untuk memastikan bahwa perangkat lunak bekerja dengan baik.

5. Pemeliharaan:

Setelah perangkat lunak selesai dan digunakan, tahap pemeliharaan dilakukan untuk memperbaiki kesalahan (bug) yang mungkin ditemukan, memberikan perbaikan, dan menghadapi perubahan atau perluasan kebutuhan pengguna.

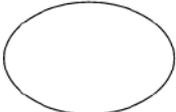
2.6.2. *Unified Modelling Language*

Unified Modeling Language (UML) adalah sebuah bahasa standar untuk mendokumentasikan, merancang, dan memodelkan sistem perangkat lunak (Oktaviani & Sauda, 2019). UML menyediakan notasi grafis yang kaya untuk menggambarkan struktur, perilaku, dan interaksi sistem. Berikut adalah beberapa jenis-jenis diagram UML yang akan digunakan di penelitian ini :

2.6.2.1. *Use Case Diagram*

Use case merupakan deskripsi fungsi dari sebuah sistem dari perspektif atau sudut pandang para pengguna sistem, juga mendefinisikan apa yang akan diproses oleh sistem dan komponen – komponennya (Setiyani, 2021). Menurut Setiyani (2021) *use case* bekerja dengan menggunakan *scenario* yang merupakan deskripsi dari urutan atau langkah – langkah yang menjelaskan apa yang dilakukan oleh user terhadap sistem maupun sebaliknya. Berikut elemen *use case diagram* yang dipakai dalam pengembangan sistem ini dapat dilihat pada Tabel 2.

Tabel 2 Elemen *Use Case Diagram*

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Segala sesuatu yang akan berinteraksi dengan sistem
2		<i>Use Case</i>	Deskripsi dari fungsionalitas suatu sistem
3		<i>System</i>	Menyatakan batasan sistem dalam relasi dengan <i>actor</i> yang menggunakannya
4		<i>Association</i>	Komunikasi antar <i>actor</i> dan <i>use case</i> yang berpartisipasi pada <i>use case</i>
5		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri tanpa <i>use case</i> tambahan itu

2.6.2.2. *Class Diagram*

Class Diagram merupakan diagram untuk mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka (Setiyani, 2021). Menurut Setiyani (2021) *class diagram* juga menunjukkan properti dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut. Berikut adalah simbol simbol *class diagram* yang digunakan di penelitian ini pada Tabel 3.

Tabel 3 Elemen *Use Case Diagram*

No	Simbol	Nama	Keterangan
1		<i>Class</i>	Memiliki fungsi untuk memetakan himpunan dari objek-objek yang berbagai atribut serta operasi
2		<i>Association</i>	Memetakan apa yang menghubungkan antara objek satu dengan objek lainnya

2.6.3. Pengujian Sistem

Pengujian sistem adalah proses pengujian dan evaluasi sistem perangkat lunak untuk memastikan bahwa sistem memenuhi persyaratan yang ditentukan. Tujuan pengujian sistem adalah untuk menemukan bug, error, atau masalah lain pada sistem sehingga dapat diperbaiki sebelum sistem diimplementasikan secara penuh.

Pengujian sistem bisa juga dibilang sekumpulan aktivitas yang meliputi perancangan, perancangan skenario pengujian, pengumpulan data pengujian, pelaksanaan pengujian, analisis hasil pengujian dan pelaporan. Kegiatan ini dilakukan dengan menggunakan berbagai teknik dan metode pengujian untuk memastikan kualitas dan keandalan sistem. Pengujian sistem dapat mencakup berbagai aspek seperti fungsionalitas sistem, performa, beban kerja, dan keamanan (Hidayat & Muttaqin, 2018).

2.6.3.1. Pengujian Fungsionalitas Sistem

2.6.3.1.1. Black Box Testing

Black Box Testing adalah metode pengujian perangkat lunak yang fokus pada pengujian fungsionalitas sistem tanpa memperhatikan struktur internal atau implementasi detail dari perangkat lunak yang sedang diuji (Hidayat & Muttaqin, 2018). Black Box Tester hanya memiliki akses ke antarmuka eksternal perangkat lunak, dan tidak memiliki pengetahuan tentang bagaimana perangkat lunak tersebut bekerja di dalamnya. Penggunaan Black Box Testing, tester tidak memiliki pengetahuan tentang bagaimana perangkat lunak diimplementasikan atau bagaimana kode programnya bekerja di balik layar.

Fokus utama adalah menguji apakah perangkat lunak berperilaku sesuai dengan harapan, mengikuti spesifikasi, dan memenuhi persyaratan yang ditetapkan. Menurut Hidayat dan Muttaqin (2018) metode ini melibatkan pembuatan kasus uji berdasarkan analisis spesifikasi dan kondisi pengujian yang mungkin terjadi, serta pengujian *input-output*, pengujian batas, pengujian kesalahan, dan pengujian skenario.

2.6.3.2. Pengujian Performa dan Beban Sistem

2.6.3.2.1. K6

K6 merupakan salah satu alat pengujian performa dan beban yang *open-source*, dirancang untuk memberikan solusi yang efisien dan kuat dalam melaksanakan uji performa serta uji beban pada sistem atau aplikasi (2023). K6 dapat dijelaskan sebagai alat pengujian beban berbasis skenario yang memungkinkan pengembang untuk mengevaluasi sejauh mana sistem atau aplikasi dapat menanggung beban yang tinggi. K6 memudahkan pengembang dengan memperkenankan penulisan skenario pengujian menggunakan sintaks *JavaScript*.

Penggunaan K6 melibatkan beberapa langkah, mulai dari merancang skenario pengujian, menentukan jumlah pengguna bersamaan, hingga melaksanakan uji kinerja pada sistem atau aplikasi target. Fokus utama K6 adalah pada uji kinerja yang realistis dan mendalam.

Alat ini memberikan perhatian khusus terhadap perubahan beban yang dapat terjadi pada aplikasi, memungkinkan identifikasi masalah yang mungkin muncul pada tahap pengembangan atau perubahan sistem.

Proses pengujian yang dilakukan oleh K6 memiliki beberapa langkah-langkah yang dilakukan seperti, pengaturan skenario pengujian yang mencerminkan perilaku pengguna nyata, mengukur dan merekam respon dari sistem atau aplikasi lalu menyajikan hasil dengan metrik performa. K6 memungkinkan pengguna untuk menganalisis hasil pengujian seperti waktu respons, tingkat kesalahan, dan ketersediaan sistem, memberikan pandangan yang jelas terhadap performa aplikasi di bawah beban tertentu.

2.6.3.3. Pengujian Intuitivitas Antarmuka Sistem

2.6.3.3.1. Skala Likert

Skala Likert adalah skala psikometrik yang banyak digunakan di banyak bidang penelitian yang berbeda untuk mengevaluasi kuesioner. Pertanyaan yang dirancang dengan menggunakan skala Likert di dalam kuesioner harus di antara setuju atau tidak setuju. Metode ini dapat di gunakan untuk memperoleh pengukuran validitas suatu fenomena. Skala Likert juga memiliki beberapa kekurangan, termasuk ke dalamnya yaitu keterbatasan karena kejujuran responden dalam menjawab pertanyaan dan validitas hasilnya (Hery, 2020)

III. METODOLOGI PENELITIAN

3.1. Waktu dan Tempat Penelitian

Penelitian ini dilakukan di Kota Bandar Lampung yang terletak di Provinsi Lampung. Penelitian ini dilakukan pada Semester Genap TA 2022/2023 s.d Semester Genap TA 2023/2024.

3.2. Perangkat Penelitian

Penelitian ini menggunakan dua jenis perangkat, yaitu perangkat keras dan perangkat lunak. Berikut merupakan spesifikasi alat yang digunakan selama penelitian.

3.2.1. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam penelitian ini adalah:

1. Sistem Operasi MacOS Sonoma 14.4.
2. Visual Studio Code versi 1.87.1
3. React.js versi 18.2.0.
4. Node.js versi 19.0.0.
5. PostgreSQL versi 12.0.0
6. Web *Browser* Mozilla Firefox.
7. Figma.
8. Draw.io.
9. Microsoft Office 2016 (Word dan Powerpoint)

3.2.2. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan penelitian ini adalah sebuah laptop dengan spesifikasi sebagai berikut.

1. *Manufacturer*: Apple.
2. *System Model*: Macbook Air.
3. *Processor*: Apple Chip M2
4. *Installed RAM*: 8.00 GB.
5. *System Type*: arm64-based processor

3.3. Jenis dan Sumber Data

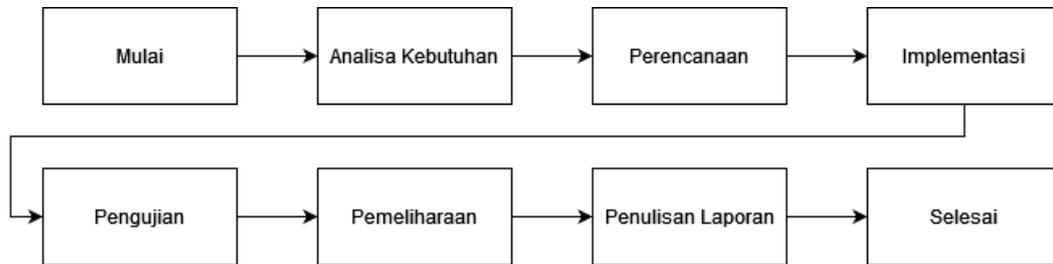
3.3.1. Data Primer

Data primer didapatkan dari hasil observasi dan wawancara pada pelaku industri barang rongsok di Kota Bandar Lampung mengenai pengumpulan, pengelolaan, dan pendistribusian barang rongsok. Pelaku industri barang rongsok yaitu pemilik barang rongsok dan penjual, pembeli, pengelola barang rongsok atau biasa disebut pengepul barang rongsok

3.3.2. Data Sekunder

Data sekunder yaitu data yang diperoleh dari studi literatur seperti buku, jurnal, artikel, dan situs resmi. Situs resmi berisikan informasi tentang penerapan *library* React.js, Node.js, *Github*, dan penelitian serupa.

3.4. Tahapan Penelitian



Gambar 1. Tahapan Penelitian

Pada penelitian ini terdapat 6 tahapan yang dilakukan yaitu Analisa Kebutuhan, Perencanaan, Implementasi, Pengujian, Pemeliharaan, dan Penulisan Laporan.

3.4.1. Analisa Kebutuhan

Analisa kebutuhan merupakan tahap yang dilakukannya observasi dan identifikasi mengenai kebutuhan terhadap permasalahan yang ada. Pada tahapan ini dihasilkan kebutuhan-kebutuhan sistem yang nantinya digunakan dalam perencanaan.

Berikut merupakan hasil analisa kebutuhan yang sudah di dapatkan berdasarkan hasil observasi dan wawancara singkat kepada pelaku industri barang rongsok yang telah dilakukan.

3.4.1.1. Analisis Kebutuhan Bisnis

1. Sistem harus mampu mengelola proses pengadaan barang rongsok dari berbagai sumber, termasuk pengepul, pengelola, dan individu.
2. Sistem harus memungkinkan pengguna untuk mengelola persediaan barang rongsok yang masuk dan keluar.
3. Sistem harus mendukung proses pengolahan seperti klasifikasi.
4. Sistem harus memungkinkan pengguna untuk mendistribusikan barang rongsok kepada pelanggan atau pihak ketiga.

3.4.1.2. Analisis Kebutuhan Fungsional

1. Sistem harus memiliki fitur manajemen pemasok untuk mengelola informasi pemasok, termasuk nama, alamat, kontak, dan catatan lainnya.
2. Sistem harus memiliki fitur manajemen persediaan untuk memantau stok barang rongsok yang tersedia, termasuk jumlah, dan lokasi penyimpanan.
3. Sistem harus memiliki fitur ruang diskusi untuk memungkinkan menemukan pemasok atau pelanggan baru.
4. Sistem harus memiliki fitur pembelian barang rongsok antar pelanggan dan mitra.

3.4.1.3. Analisis Kebutuhan Non-Fungsional

1. Sistem harus dapat menangani beban yang tinggi, terutama saat mengelola persediaan dalam jumlah yang besar.
2. Sistem harus mudah digunakan dan memiliki antarmuka pengguna yang intuitif.

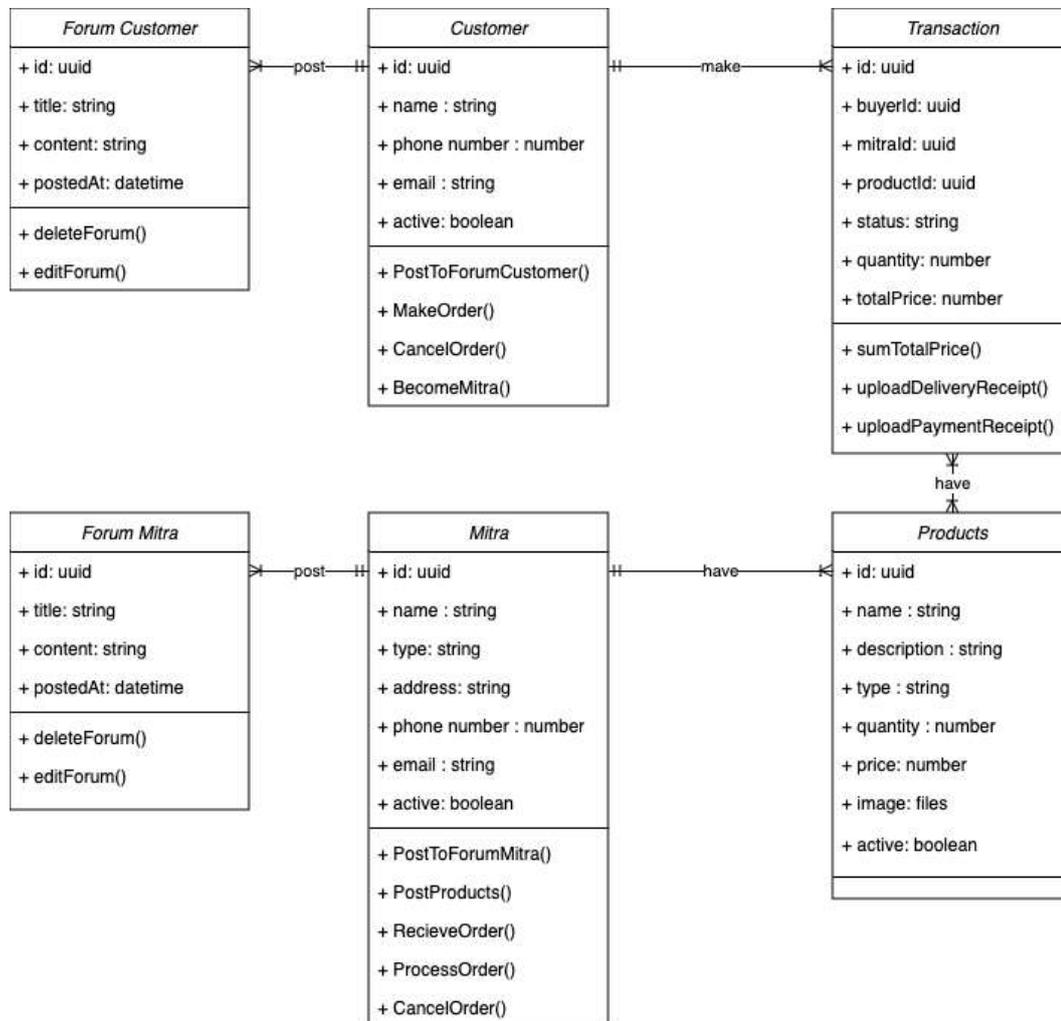
3.4.2. Perencanaan

Perencanaan merupakan tahap yang dimana hasil analisa kebutuhan dikembangkan menjadi rancangan seperti model tampilan sistem (*interface*), model *class*, model *activity*, model *database*, model *use case*.

3.4.2.1. Rancangan *Class Diagram*

Rancangan *class diagram* pada Gambar 2 didapatkan berdasarkan hasil dari tahapan analisis kebutuhan yang telah dilakukan sebelum tahap perencanaan ini. Gambar 2 menjelaskan ada delapan objek yang akan di implementasikan ke dalam sistem

supply chain management ini dan juga berbagai hubungan terhadap masing masing objek.

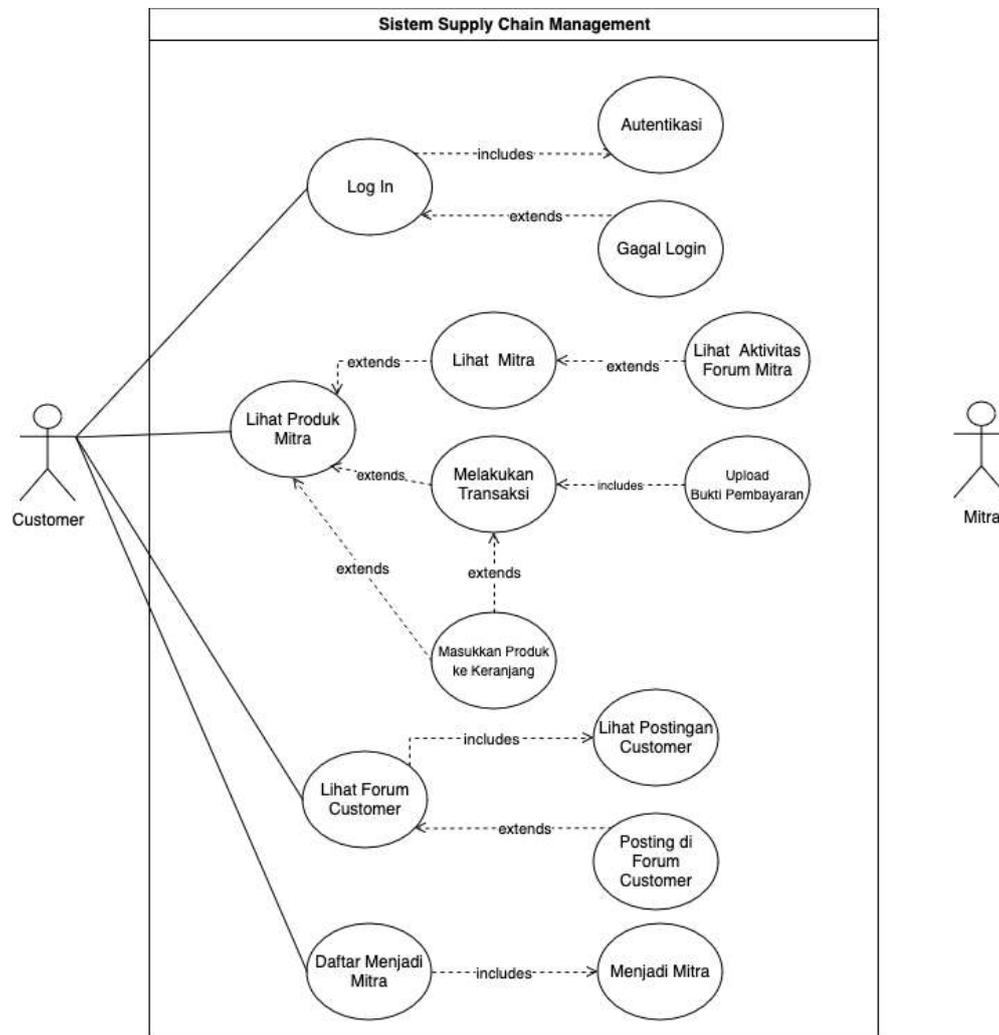


Gambar 2. Rancangan *Class Diagram*

3.4.2.2. Rancangan *Use Case Diagram*

Rancangan *use case* diagram didapatkan berdasarkan hasil tahapan analisis kebutuhan yang telah dilakukan untuk memenuhi kebutuhan sistem yang ingin dibangun. Gambar 3 yaitu *use case* diagram untuk menjelaskan fungsi dari

sistem *supply chain management* dari prespektif atau sudut pandang pengguna dalam Gambar 3 pengguna atau aktor yang di maksud yaitu *customer*.



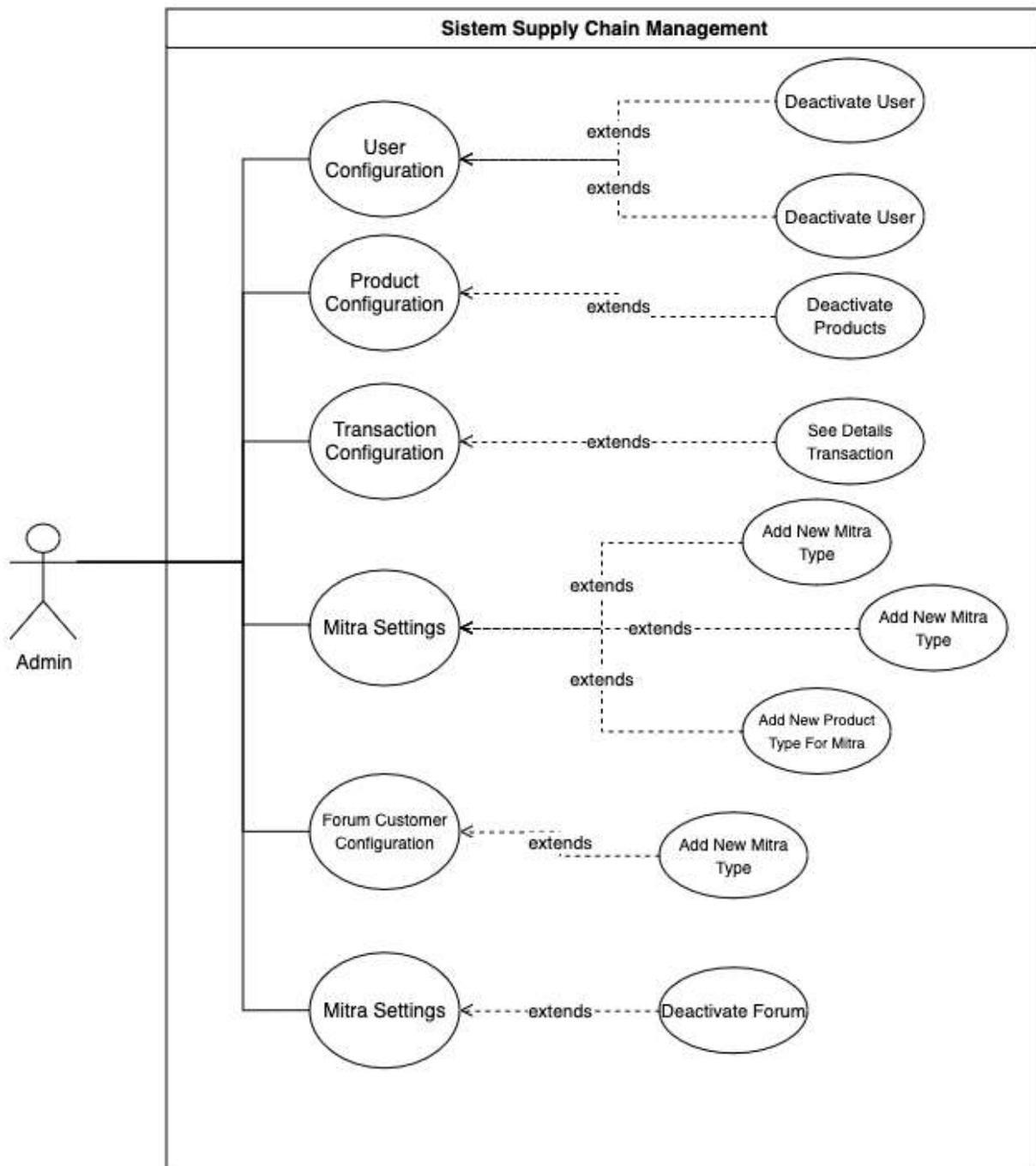
Gambar 3. Rancangan *Use Case Diagram* (Customer)

Gambar 4 menjelaskan fungsi sistem dari prespektif atau sudut pandang pengguna dalam Gambar 4 di jelaskan pengguna atau aktor yang dimaksud yaitu mitra.



Gambar 4. Rancangan *Use Case Diagram* (Mitra)

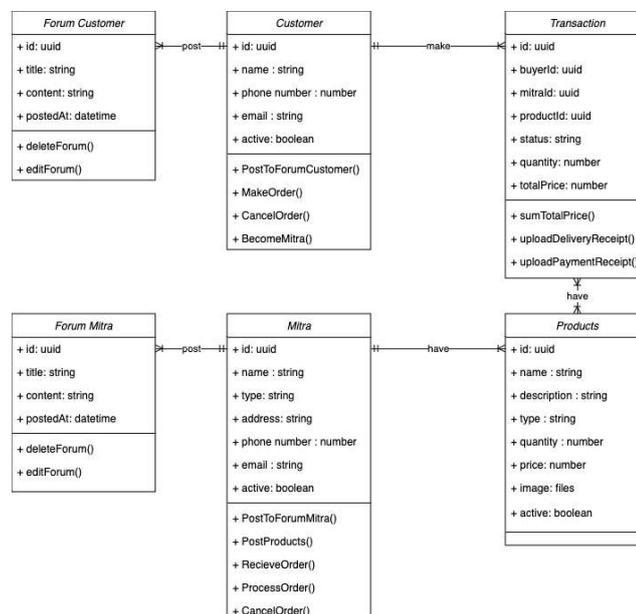
Gambar 5 menjelaskan fungsi sistem dari prespektif atau sudut pandang admin, Gambar 5 menjelaskan aktor atau pengguna adalah admin.



Gambar 5. Rancangan *Use Case* Diagram (Admin)

3.4.2.3. Rancangan ERD (*Entity Relationship Diagram*)

Rancangan *database* yang berupa *entity relationship* diagram ini didapatkan dari hasil tahapan analisis kebutuhan yang telah dilakukan sebelum tahapan ini. Gambar 5 yaitu *ER* diagram menjelaskan bahwa ada 5 entitas yang akan di implementasikan kedalam sistem *supply chain management*, ada entitas *customer*, mitra, forum *customer*, forum mitra, dan *product*. Entitas ini masing masing memiliki hubungan yang berbeda beda menyesuaikan dengan hasil tahapan analisis kebutuhan.



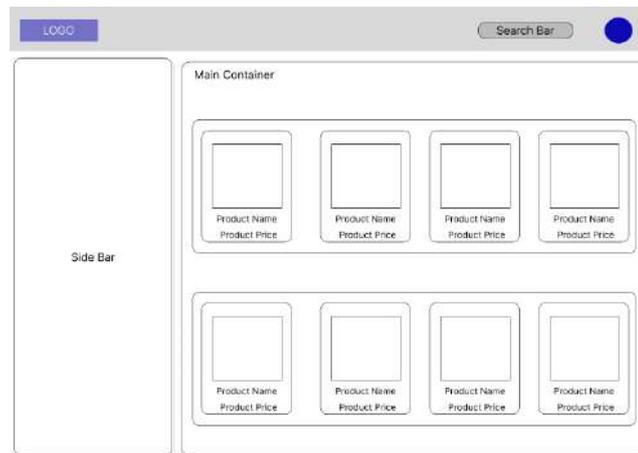
Gambar 6. Rancangan ER Diagram

3.4.2.4. Rancangan Tampilan Sistem (*Interface*)

Berikut merupakan rancangan tampilan sistem yang mengacu kepada analisis kebutuhan sistem.

1. Rancangan Tampilan *Homepage Screen* Sistem

Tampilan ini merupakan halaman yang pertama kali akan tampil saat pengguna mengakses sistem. Halaman ini merupakan *homepage* sekaligus *landing page* yang menandakan user sudah memasuki sistem. Berikut rancangan tampilan halaman *Homepage Screen* pada Gambar 6.



Gambar 7. Rancangan Tampilan *Homepage*

2. Rancangan Tampilan *Register Screen* Sistem

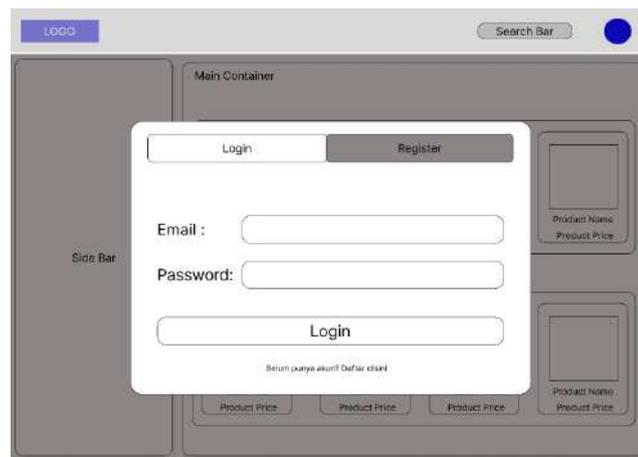
Tampilan ini merupakan halaman dimana pengguna sistem ingin mendaftar sebagai *customer* di sistem. Berikut rancangan tampilan *Register Screen* pada Gambar 7.



Gambar 8. Rancangan Tampilan *Register Screen*

3. Rancangan Tampilan *Login Screen* Sistem

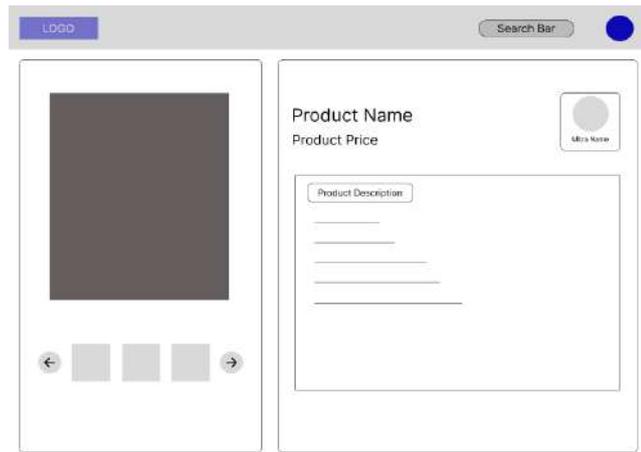
Tampilan ini merupakan halaman dimana seorang *customer* atau mitra ingin melakukan autentikasi untuk masuk kedalam sistem. Pada halaman ini pengguna diminta memasukan kredensial akun yang telah di buatnya pada halaman *register screen*. Berikut rancangan tampilan *Login Screen* dapat dilihat pada Gambar 8.



Gambar 9. Rancangan Tampilan *Login Screen*

4. Rancangan Tampilan *Product Screen* Sistem

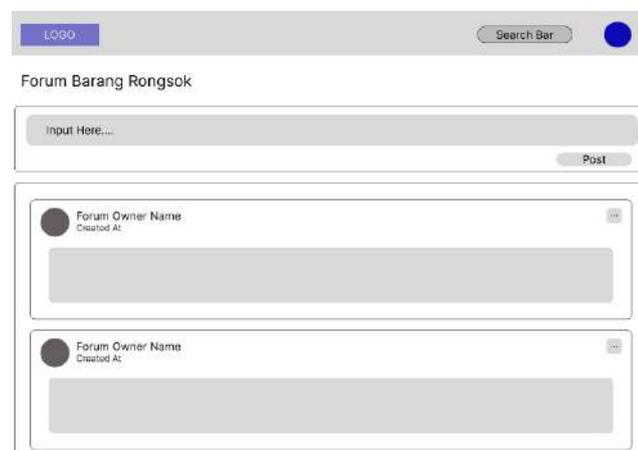
Tampilan ini merupakan halaman dimana seorang *customer* atau mitra ingin melihat detail suatu produk informasi yang didapat dari produk tersebut seperti harga, deskripsi, dan kuantitas produk. Berikut rancangan tampilan *Product Screen* pada sistem pada Gambar 9.



Gambar 10. Rancangan Tampilan *Product Screen*

5. Rancangan Tampilan Sistem Forum *Screen* Barang Rongsok

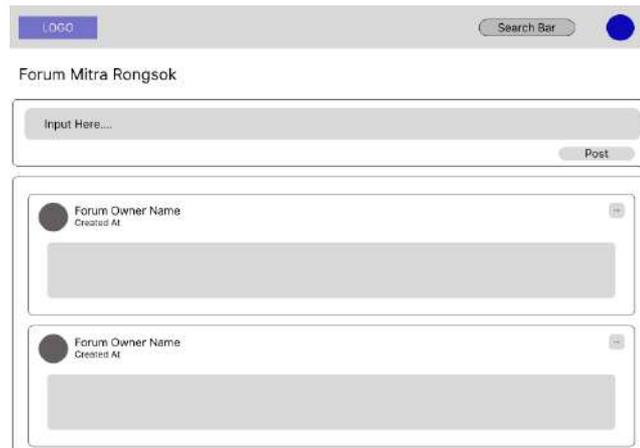
Tampilan ini merupakan tampilan yang digunakan untuk seorang *customer* mengunggah kebutuhan dan keinginan juga mencari mitra untuk bekerja sama. Berikut rancangan tampilan Forum *Screen* Barang Rongsok pada Gambar 10.



Gambar 11. Rancangan Tampilan Forum *Screen* Barang Rongsok

6. Rancangan Tampilan Forum *Screen* Mitra Barang Rongsok

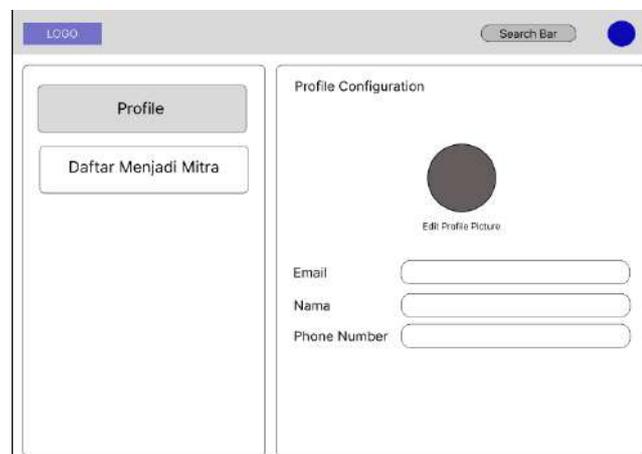
Tampilan ini merupakan tampilan yang dipergunakan oleh mitra untuk mengunggah kebutuhan bisnis mereka ke sesama mitra untuk memenuhi kebutuhan bisnis mereka. Berikut rancangan tampilan Forum *Screen* Mitra Barang Rongsok pada Gambar 11.



Gambar 12. Rancangan Tampilan Forum *Screen* Mitra Barang Rongsok

7. Rancangan Tampilan Konfigurasi Profil *Customer*

Tampilan ini merupakan tampilan yang dipergunakan oleh *customer* untuk mengkonfigurasi data data di profil atau akun mereka seperti email, nomor handphone, dan nama *customer*. Berikut rancangan tampilan Konfigurasi Profil *Customer* sistem *supply chain management* barang rongsok pada Gambar 12.



Gambar 13. Rancangan Tampilan Konfigurasi Profil *Customer*

8. Rancangan Tampilan Konfigurasi Mendaftar Menjadi Mitra *Customer*

Tampilan ini merupakan tampilan yang dipergunakan jika seorang *customer* ingin menjadi mitra di sistem *supply chain management* barang rongsok. Pada halaman ini *customer* dibutuhkan melengkapi informasi terlebih

dahulu sebelum mendaftar menjadi mitra. Berikut merupakan rancangan tampilan Konfigurasi Mendaftar Menjadi Mitra *Customer* pada Gambar 13.

LOGO Search Bar

Profile

Daftar Menjadi Mitra

Daftar Menjadi Mitra

Email

Nama

Phone Number

Alamat

Gambar 14. Rancangan Tampilan Konfigurasi Mendaftar Menjadi Mitra Customer

9. Rancangan Tampilan Konfigurasi Profil Mitra

Tampilan ini merupakan tampilan yang dipergunakan oleh mitra untuk mengkonfigurasi data di profil atau akun mitra seperti email, nama mitra, nomor handphone, dan alamat mitra. Berikut merupakan rancangan tampilan Konfigurasi Profil Mitra pada Gambar 14.

LOGO Search Bar

Profile

Katalog

Product

Profile Configuration

Edi Profile Picture

Email

Nama

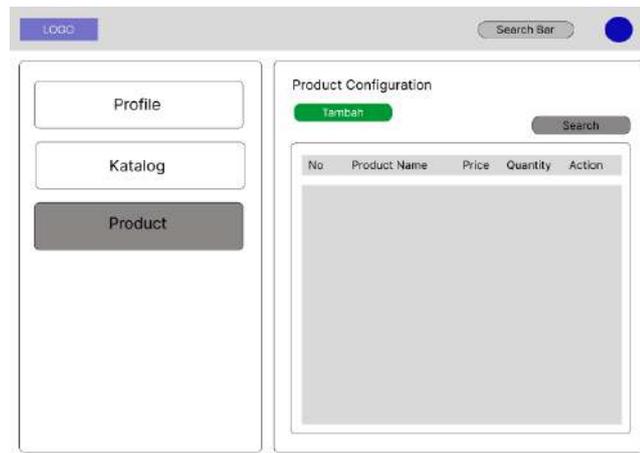
Alamat

Phone Number

Gambar 15. Rancangan Tampilan Konfigurasi Profil Mitra

10. Rancangan Tampilan Konfigurasi Produk Mitra

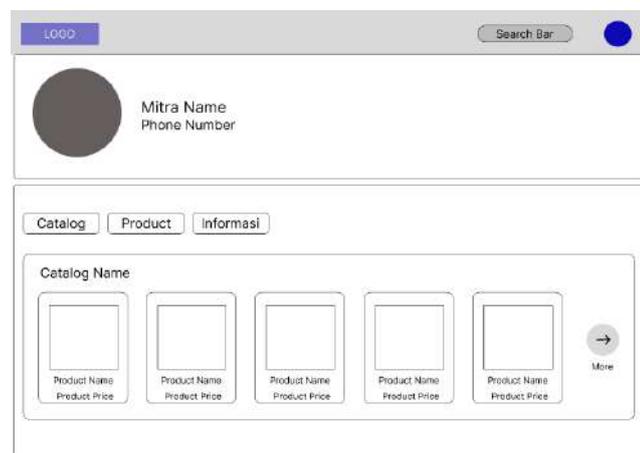
Tampilan ini merupakan tampilan yang dipergunakan oleh mitra untuk mengkonfigurasi produk yang dimiliki mitra seperti nama, deskripsi, kuantitas, kondisi, dan harga produk mitra. Berikut merupakan rancangan tampilan Konfigurasi Produk Mitra pada Gambar 15.



Gambar 16. Rancangan Tampilan Konfigurasi Produk Mitra

11. Rancangan Tampilan Profil Mitra

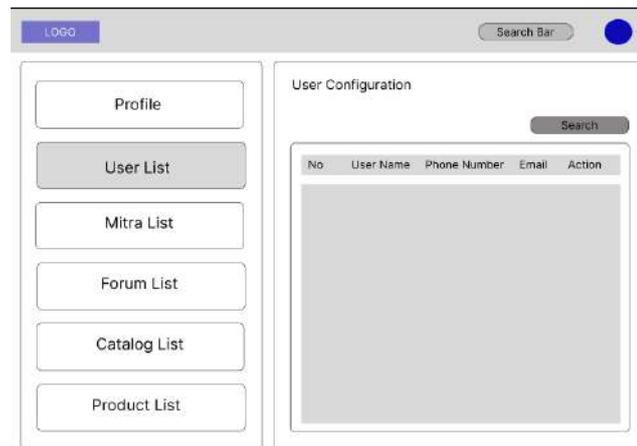
Tampilan ini merupakan tampilan yang menampilkan profil mitra beserta katalog maupun produk yang ada di dalam sistem. Pada halaman ini pengguna bisa melihat secara detail informasi mengenai mitra seperti nomor handphone, alamat, katalog, dan produk mitra tersebut. Berikut merupakan rancangan tampilan profil mitra pada Gambar 16.



Gambar 17. Rancangan Tampilan Profil Mitra

12. Rancangan Tampilan Konfigurasi *User*

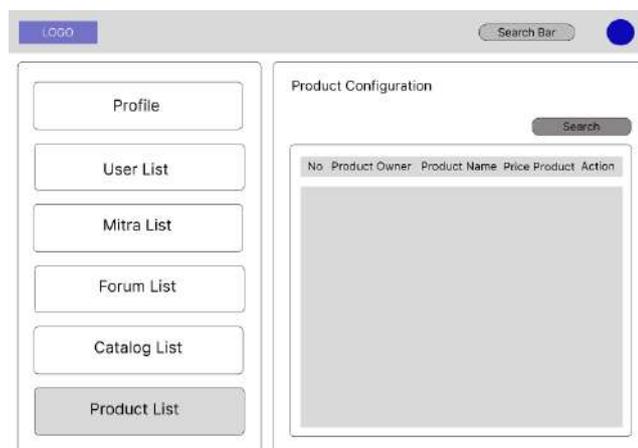
Tampilan ini merupakan tampilan yang menampilkan konfigurasi admin terhadap daftar pengguna atau *user*. Pada halaman ini admin bisa melakukan deaktivasi akun *user* atau menjadikan akun *user* menjadi *superuser*. Berikut merupakan rancangan tampilan konfigurasi *User* pada Gambar 18.



Gambar 18. Rancangan Tampilan Konfigurasi *User*

13. Rancangan Tampilan Konfigurasi *Products*

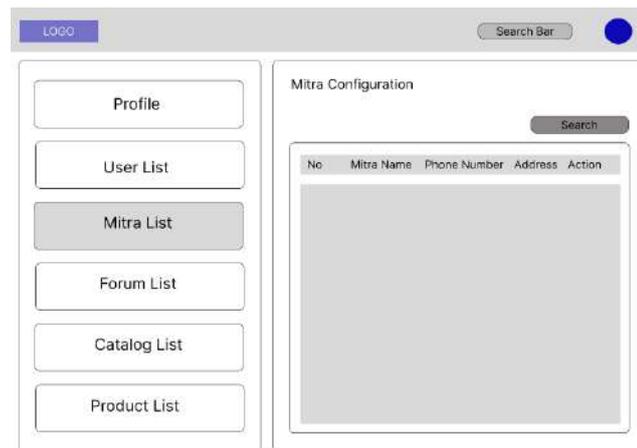
Tampilan ini merupakan tampilan yang menampilkan konfigurasi *products* mitra. Pada halaman ini admin bisa deaktivasi sebuah *products* sehingga tidak dapat diperjual belikan didalam sistem. Berikut merupakan rancangan tampilan konfigurasi *Products* pada Gambar 19



Gambar 19. Rancangan Tampilan Konfigurasi *Products*

14. Rancangan Tampilan Konfigurasi Mitra

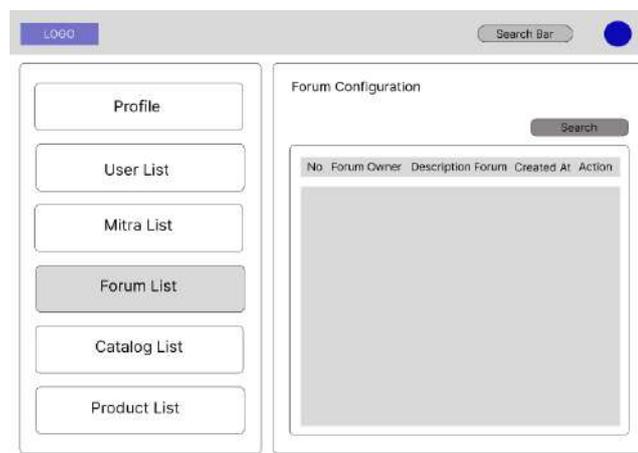
Tampilan ini merupakan tampilan yang menampilkan konfigurasi mitra. Pada halaman ini admin bisa menambahkan tipe mitra, tipe produk mitra, dan satuan unit produk mitra. Berikut merupakan rancangan tampilan konfigurasi Mitra pada Gambar 20.



Gambar 20. Rancangan Tampilan Konfigurasi Mitra

15. Rancangan Tampilan Konfigurasi Forum

Tampilan ini merupakan tampilan yang menampilkan konfigurasi dari postingan yang ada di forum. Pada halaman ini admin bisa deaktivasi suatu postingan. Berikut merupakan rancangan tampilan konfigurasi Forum pada Gambar 21.



Gambar 21. Rancangan Tampilan Konfigurasi Forum

3.4.3. Implementasi

Implementasi merupakan tahapan dimana semua rancangan yang sudah di hasilkan di tahap perencanaan mulai di implementasikan kedalam sistem sesuai dengan analisis kebutuhan yang di dapat pada tahap analisis.

Terdapat dua kali tahapan implementasi yang dilakukan menggunakan dua alat yang berbeda dengan tujuan yang berbeda juga.

3.4.3.1. Implementasi *Library* React.js

Implementasi menggunakan *library* React.js dilakukan untuk menerjemahkan perencanaan tampilan ke dalam basis web. Tahapan yang dilakukan dalam implementasi ini yaitu dengan membuat komponen-komponen yang digunakan dalam hasil rancangan tampilan sistem pada tahap perencanaan menjadi komponen yang dapat digunakan di dalam web. Komponen yang akan dibuat harus bisa menerima dan menampilkan informasi yang dikirim atau dipanggil untuk diletakan di dalam komponen tersebut.

3.4.3.2. Implementasi Node.js

Implementasi menggunakan Node.js dilakukan untuk mendukung kebutuhan informasi yang akan digunakan dari komponen yang telah dihasilkan pada tahapan implementasi *library* React.js. Dukungan yang diberikan seperti pencarian informasi yang ingin ditampilkan pada komponen tersebut dan memenuhi permintaan komponen tersebut. Proses bisnis yang dilakukan di dalam sistem seperti menulis, mengubah, dan menghapus data juga merupakan dukungan yang akan dipenuhi oleh hasil tahapan implementasi ini.

3.4.4. Pengujian

Tahapan pengujian adalah tahapan yang dilakukan setelah semua tahapan implementasi selesai. Tahapan ini bertujuan agar sistem berjalan dengan sesuai

yang diharapkan. Tahapan pengujian akan dilakukan dengan cara *Functional Testing* untuk memastikan semua fungsi sistem berjalan dengan baik.

3.4.4.1. Fungsionalitas Sistem

Pengujian fungsi dilakukan dengan metode *Black-Box Testing* dilakukan untuk memastikan fungsional sistem berjalan dengan skenario yang diharapkan. Skenario yang dari pengujian yang dilakukan sebagai berikut.

1. Skenario *Testing* Registrasi

Test Case Description : *Testing* Registrasi Sistem

Tabel 4. Skenario *Testing* Registrasi

No	<i>Test Case ID</i>	<i>Test Scenario</i>
1.	REG_001	Pengguna melakukan registrasi.

2. Skenario *Testing* Autentikasi dan Otorisasi

Test Case Description : *Testing* Autentikasi dan Otorisasi Sistem

Tabel 5. Skenario *Testing* Autentikasi

No	<i>Test Case ID</i>	<i>Test Scenario</i>
1.	AUT_001	<i>Customer</i> /Mitra melakukan login ke sistem

3. Skenario *Testing* Fitur Sistem *Supply Chain Management* Barang Rongsok

Test Case Description : *Testing* Fitur Sistem

Tabel 6. Skenario Testing Fitur Sistem

No	Test Case ID	Test Scenario
1.	USR_001	Customer/Mitra masuk ke <i>Homepage</i> sistem
2.	USR_002	Customer/Mitra melihat detail produk
3.	USR_003	Customer/Mitra mencari produk
4.	USR_004	Customer/Mitra melihat profil mitra
6.	USR_006	Customer/Mitra melihat unggahan di forum barang rongsok
7.	CUS_001	Customer melakukan konfigurasi profil
8.	CUS_002	Customer melakukan registrasi menjadi mitra
9.	CUS_003	Customer <i>posting</i> ke forum barang rongsok
10.	CUS_004	Customer membuat pembelian produk mitra
11.	CUS_005	Customer membatalkan pembelian produk mitra
12.	MIT_001	Mitra melakukan konfigurasi profil
13.	MIT_002	Mitra melakukan konfigurasi produk
14.	MIT_003	Mitra memproses pembelian produk customer
15.	MIT_004	Mitra membatalkan pembelian produk customer
16.	MIT_005	Mitra melakukan <i>posting</i> ke forum mitra barang rongsok

3.4.4.2. Performa dan Beban Sistem

Pengujian performa dan beban dilakukan dengan menggunakan alat atau perangkat lunak K6, untuk mengetahui kapasitas performa dan beban dari sistem. Pengujian yang dilakukan menggunakan 4 tipe pengujian yang sudah di sediakan oleh K6, antara lain :

1. *Average-load Test* yaitu untuk menilai bagaimana performa sistem dalam kondisi normal yang diharapkan.
2. *Stress Test* yaitu untuk menilai bagaimana performa sistem ketika beban melebihi rata-rata yang di ekspektasikan.
3. *Spike Test* yaitu menilai bagaimana performa sistem ketika terjadi peningkatan aktivitas secara tiba-tiba dan besar.

Perancangan skenario untuk melakukan pengujian performa dan beban pada sistem di tampilkan pada Tabel 7.

Tabel 7. Perancangan Pengujian Performa dan Beban Sistem

No.	<i>Virtual User</i>	Durasi(menit)	Skenario Pengujian
1.	1.000	15	<i>Average-load, Stress, Spike</i>

3.4.4.3. Intuitivitas Antarmuka Sistem

Pengujian Intuitivitas Antarmuka Sistem dilakukan untuk mengukur seberapa intuitif hasil antar muka yang telah di kembangkan dengan cara responden mencoba sistem yang telah di implementasikan, setelah itu responden mengisi kuesioner megenai intuitivitas antarmuka sistem. Data hasil rekapitulasi kuisisioner kemudian di analisis menggunakan skala likert. Tabel 8 menunjukkan rancangan pertanyaan untuk pengujian intuitivitas antarmuka sistem.

Tabel 8. Rancangan Pertanyaan Kuesioner Pengujian Intuitivitas Antarmuka Sistem

No.	Pertanyaan
1.	Anda setuju bahwa navigasi antarmuka sistem ini mudah dipahami?
2.	Anda setuju tentang keterbacaan teks dan informasi pada antarmuka ini?
3.	Anda setuju ikon dan simbol yang digunakan dalam antarmuka sistem?
4.	Anda setuju antarmuka sistem menyediakan bantuan yang cukup untuk memahami fungsionalitasnya?
5.	Anda setuju mudah untuk menemukan fitur-fitur utama dalam antarmuka sistem ini?
6.	Anda setuju merasa nyaman menggunakan antarmuka sistem ini tanpa bantuan tambahan?
7.	Anda setuju terhadap kejelasan tata letak antarmuka sistem?

8. Anda setuju tentang responsifitas antarmuka sistem terhadap tindakan pengguna?
 9. Anda setuju antarmuka sistem ini meminimalkan kebingungan pengguna?
-

Responden diberikan pilihan yang dapat dipilih atas pertanyaan yang ada di kuesioner berdasarkan beberapa kriteria. Kriteria tersebut ditentukan menggunakan Skala Likert yang dapat dilihat di Tabel 9.

Tabel 9. Bobot Jawaban Kuisisioner

No	Pilihan Jawaban	Bobot
1.	Sangat Setuju	5
2.	Setuju	4
3.	Cukup	3
4.	Tidak Setuju	2
5.	Sangat Tidak Setuju	1

Setelah mendapatkan jawaban dari responden selanjutnya akan dilakukan perhitungan dengan menggunakan beberapa rumus sebagai berikut.

$$P = \frac{x}{\text{skor ideal}} \times 100\%$$

dimana

$$X = \sum (N \times R)$$

Skor ideal = nilai likert tertinggi \times jumlah responden

Keterangan:

P = nilai presentase yang dicari

X = jumlah dari hasil perkalian nilai setiap jawaban dengan responden

N = nilai dari setiap jawaban

R = jumlah responden

Tabel interval penilaian untuk menentukan kategori penilaian yang diperoleh melalui perhitungan jawaban responden dapat dilihat di Tabel 10.

Tabel 10. Kriteria Interpretasi Interval

No	Pilihan Jawaban	Kategori Penilaian
1.	Sangat Setuju	80% - 100%
2.	Setuju	60% - 79,99%
3.	Cukup	40% - 59,99%
4.	Tidak Setuju	20% - 39,99%
5.	Sangat Tidak Setuju	0% - 19,99%

Pengujian intuitivitas antarmuka sistem ini dilakukan oleh sepuluh (10) pelaku industri barang rongsok yaitu pengepul maupun pengelola yang ada di Kota Bandar Lampung dan sepuluh (10) individu yang pernah menjual barang rongsok ke pengepul. Mekanisme pengujian dilakukan secara daring menggunakan *Google Formulir* untuk mengisi kuisionernya. Jawaban pertanyaan Tabel 9 kemudian di analisis menggunakan skala likert yang dapat dilihat pada Tabel 10.

3.4.5. Pemeliharaan

Tahapan pemeliharaan adalah tahapan dimana setelah melakukan proses pengujian sistem dan mendapat sebuah hasil yang diinginkan maka harus di pertahankan. Hasil yang tidak diinginkan harus di sesuaikan dengan hasil yang diinginkan, disini tahap pemeliharaan menjadi tahap yang penting untuk sebuah keberlangsungan sistem.

3.4.6. Penulisan Laporan

Penulisan laporan adalah tahapan terakhir di penelitian ini, penulisan ini bertujuan untuk bahan referensi penelitian yang akan datang. Laporan ini juga selain menjadi referensi juga merupakan dokumentasi terhadap penelitian ini.

V. SIMPULAN DAN SARAN

5.1. Simpulan

Penelitian yang telah dilakukan memiliki hasil penelitian yang dapat disimpulkan sebagai berikut.

1. Telah di kembangkan dan di implementasikan Node.js sebagai *server-side* dan *API Provider* untuk sistem *supply chain management* barang rongsok dengan studi kasus Kota Bandar Lampung.
2. Telah di kembangkan dan di implementasikan React.js sebagai *client-side* untuk sistem *supply chain management* barang rongsok dengan studi kasus Kota Bandar Lampung.
3. Sistem *supply chain management* barang rongsok berbasis web sudah melalui 3 tahapan pengujian sistem yaitu, pengujian fungsionalitas sistem, pengujian performa dan beban sistem juga pengujian intuitivitas antarmuka sistem. Pengujian fungsionalitas sistem dilakukan menggunakan metode *black-box testing* yang memiliki hasil sesuai yang diharapkan. Pengujian performa dan beban dilakukan menggunakan alat atau perangkat lunak k6 yang memiliki hasil sesuai yang cukup memuaskan. Pengujian intuitivitas antarmuka sistem diujikan menggunakan metode likert yang menghasilkan nilai indeks penilaian 80%-100%, sehingga dapat disimpulkan responden setuju dengan antarmuka sistem yang sudah intuitiv.

5.2. Saran

Berdasarkan proses pengembangan , pengujian sistem, dan penilaian dari responden yang telah dilakukan, maka saran yang dapat direkomendasikan untuk penelitian masa yang akan datang antara lain adalah sebagai berikut :

1. Membuat desian sistem yang kompatibel dengan berbagai perangkat agar pengguna tidak hanya dapat menggunakan sistem dengan perangkat laptop atau desktop.
2. Membuat dokumentasi atau *user guide* sistem yang dikembangkan agar pengguna dapat lebih mudah mengenal dan menggunakan sistem, sehingga pengguna tidak harus didampingi saat mengoperasikan sistem.
3. Mengembangkan fitur *payment-gateway* untuk mendukung transaksi di dalam jaringan, agar memudahkan pengguna transaksi dalam nominal uang yang besar.

DAFTAR PUSTAKA

- React.js. (2023). *Getting Started with React.js*. Diambil kembali dari React.js:
<https://reactjs.org/docs/getting-started.html>
- Joyent. (2023). *Jooyent*. Diambil kembali dari Node.js Production:
<https://www.joyent.com/node-js/production>
- Ahmad, F. (2018). Perancangan Sistem Informasi Berbasis Web Menggunakan Framework Laravel dan React.js. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 39-46.
- Wijaya, S., & Akbar, M. R. (2019). A Comparative Study of Node.js and Java for Backend Development. *International Journal of Computer Science and Network Security*, 23-28.
- Septiar, T. S., Fajar, T. N., & Darmawan, E. (2020). Implementasi Supply Chain Management (SCM) Toko Alat dan Bahan Bangunan Berbasis Web (Studi Kasus: TB. Bojong Indah). *Infortech Journal*, 44-50.
- Asyahdina, A., Krisnanik, E., & Wirawan, R. (2021). Rancang Bangun Supply Chain Management Budidaya Jamur Berbasis Web (Studi Kasus: Budidaya Jamur Jatayumn). *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, 91-98.
- Hasiholan, R. S. (2015). *Pemrograman Javascript: Teori dan Implementasi*. INFORMATIKA.
- Oktaviani, N., & Sauda, S. (2019). Pemodelan dan Implementasi Aplikasi Mobile Umrah Guide Menggunakan Unified Modeling Language. *Jurnal Sains dan Informatika*, 177-186.
- Setiyani, L. (2021). Desain Sistem : Use Case Diagram. *Seminar Nasional: Inovasi & Adopsi Teknologi*, 246-260.
- Hidayat, T., & Muttaqin, M. (2018). Pengujian Sistem Informasi Pendaftaran Pembayaran Wisuda Online menggunakan Black Box Testing dengan

- Metode Equivalence Partitioning dan Boundary Value Analysis. *Jurnal Teknik Informatika UNIS*, 25-29.
- Wahid, A. A. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi. *Jurnal Ilmu-Ilmu Informatika dan Manajemen STMIK*, 1-5.
- Wijoyo, H., Ariyanto, A., Sudarsono, A., & Dwi, K. W. (2021). *Sistem Informasi Manajemen*. Sumatra Barat: Insan Cendikia Mandiri.
- Gusti, N. A., Akbar, R. H., & Indah, P. C. (2019). AJARINCODE : APLIKASI PEMBELAJARAN BAHASA PEMROGRAMAN BERBASIS WEB. *Garba Rujukan Digital*, 21-23.
- Christono, K., & Sama, H. (2020). Studi Komparasi Database Management System Antara Maria DB dan PostgreSQL Terhadap Efisiensi Penggunaan Sumber Daya Komputer. *Conference on Business, Social Sciences and Innovation Technology*, 573-579.
- Pratama, A., Sholva, Y., & Azhar, M. I. (2023). Aplikasi Jual Beli Barang Rongsok Berbasis Jarak Menggunakan Fitur Location Based Service. *Jurnal Riset Sains dan Teknologi Informatika*, 93-99.
- Dicky, & Kurniawan. (2022). Sistem Informasi Pemantauan Penjualan Barang Rongsokan Menggunakan Framework CI.4 Pada CV. Sumber Baja. *Jurnal Sistem Informasi*, 37-50.
- Supardiyono, D., Pertiwi, I. W., & Eridani, D. (2022). Perancangan Sistem Informasi Pengelolaan Barang Bagi Pengepul Rongsok Berbasis Web Menggunakan Kerangka Kerja CodeIgniter. *Jurnal Teknik Komputer*, 90-100.
- Rahman, A. C. (2021, June 9). *Harus Ada Solusi, Penanganan Sampah Plastik di Lampung*. Diambil kembali dari MONGABAY Situs Berita Lingkungan: <https://www.mongabay.co.id/2021/06/09/harus-ada-solusi-penanganan-sampah-plastik-di-lampung/>
- Mentzer, J. T., DeWitt, W., S. Keebler, J., Min, S., W. Nix, N., D. Smith, C., & G. Zacharia, Z. (2011). Defining Supply Chain Management. *Journal of Bussiness Logistics*, 1-25.
- Grafana Labs. (2023, 11 25). *What is K6 ?* Diambil kembali dari <https://k6.io/docs/#what-is-k6>: <https://k6.io>

Hery, A. S. (2020). SKALA LIKERT UNTUK PENELITIAN PARIWISATA;
BEBERAPA CATATAN UNTUK MENYUSUNNYA DENGAN BAIK.
Jurnal Kepariwisata, 29-40.