

**PERANCANGAN *BACKEND WEBSITE* ERP (*ENTERPRISE RESOURCE
PLANNING*) PADA INDUSTRI KELAPA SAWIT**

(Skripsi)

**Oleh
FRANS ALDI GURNING
2015031087**



**PRODI TEKNIK ELEKTRO FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

ABSTRAK

PERANCANGAN *BACKEND WEBSITE* ERP (*ENTERPRISE RESOURCE PLANNING*) PADA INDUSTRI KELAPA SAWIT

OLEH

FRANS ALDI GURNING

Penelitian ini bertujuan untuk merancang dan mengembangkan sistem manajemen rantai pasok berbasis *Enterprise Resource Planning* (ERP) yang mengintegrasikan teknologi *blockchain* untuk meningkatkan transparansi, keterlacakan, dan keberlanjutan dalam industri sawit. Dengan memanfaatkan konsep *blockchain*, sistem ini memastikan keamanan dan transparansi dalam setiap transaksi, sementara fitur *traceability* memungkinkan pemantauan yang akurat terhadap asal-usul dan perjalanan produk sawit. Pengembangan sistem dilakukan menggunakan metodologi Scrum, dengan pemanfaatan Python dan Flask sebagai *framework backend* serta PostgreSQL sebagai basis data. Pengujian sistem dilakukan untuk memastikan bahwa semua endpoint dapat merespons permintaan dengan baik, khususnya untuk fitur *traceability*. Hasil penelitian menunjukkan bahwa implementasi Python dan PostgreSQL pada sistem ini berfungsi dengan baik pada entitas yang ada di dalam sistem. Dengan *database* PostgreSQL, data dapat disimpan dan diakses secara cepat serta akurat, sementara Python memungkinkan implementasi logika bisnis yang fleksibel dan skalabel. Dengan konsep *blockchain*, seluruh data dasar dan data transaksi terekam dan dapat diakses oleh setiap *stakeholder*, sehingga meningkatkan *visibility*, transparansi, dan akuntabilitas proses bisnis perusahaan.

Kata kunci: *Enterprise Resource Planning*, Industri Sawit, *Traceability*, *Blockchain*. *transparansi*.

ABSTRACT

DESIGNING THE BACKEND OF AN ERP (ENTERPRISE RESOURCE PLANNING) WEBSITE FOR THE PALM OIL INDUSTRY

BY

FRANS ALDI GURNING

This research aims to design and develop a supply chain management system based on Enterprise Resource Planning (ERP) that integrates blockchain technology to enhance transparency, traceability, and sustainability in the palm oil industry. By utilizing blockchain concept, the system ensures security and transparency in every transaction, while the traceability feature allows accurate monitoring of the origin and journey of palm oil products. The system development is carried out using the Scrum methodology, with Python and Flask as the backend framework and PostgreSQL as the database. System testing is conducted to ensure that all endpoints respond properly, particularly for the traceability feature. The results of this research indicate that the implementation of Python and PostgreSQL in the system functions well within the existing entities. With PostgreSQL, data can be stored and accessed quickly and accurately, while Python allows the implementation of flexible and scalable business logic. With the blockchain concept, all basic data and transaction data are recorded and accessible by every stakeholder, thus improving visibility, transparency, and accountability in the company's business processes.

Keywords: *Enterprise Resource Planning, Palm Oil Industry, Traceability, Blockchain, Transparency.*

Judul Skripsi : **PERANCANGAN BACKEND WEBSITE ERP
(ENTERPRISE RESOURCE PLANNING)
PADA INDUSTRI KELAPA SAWIT**

Nama Mahasiswa : Frans Aldi Gurning

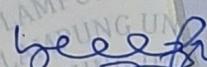
Nomor Pokok Mahasiswa : 2015031087

Program Studi : Teknik Elektro

Fakultas : Teknik

MENYETUJUI

1. Komisi Pembimbing

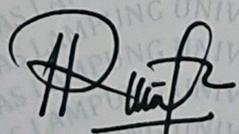

Yetti Yuniati, S.T., M.T.

NIP 198001132009122002

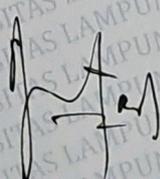

Dr. -Ing. Ardian Ulvan, S.T., M.Sc.

NIP 197311281999031005

2. Mengetahui


Herlinawati, S.T., M.T.

NIP 197103141999032001


Sumadi, S.T., M.T.

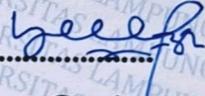
NIP 197311042000031001

MENGESAHKAN

1. Tim Penguji

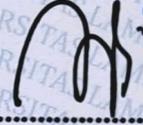
Ketua

Yeti Yuniati, S.T., M.T.



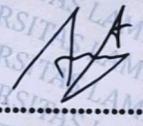
Sekretaris

Dr. -Ing. Ardian Ulvan, S.T., M.Sc.

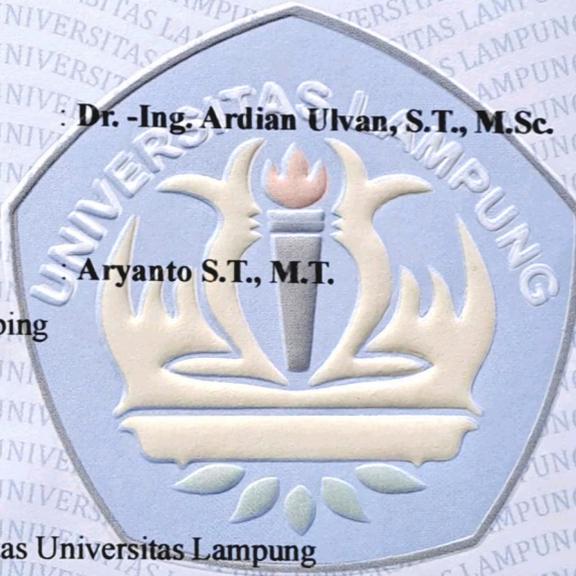


Penguji

Aryanto S.T., M.T.



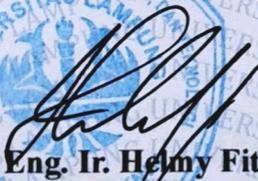
Bukan pembimbing



2. Dekan Fakultas Universitas Lampung

Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc.

NIP 197509282001121002



Tanggal Lulus Ujian Skripsi: 5 Juni 2025

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini:

Nama : Frans Aldi Gurning

NPM : 2015031087

Dengan ini saya menyatakan bahwa skripsi saya yang berjudul “Perancangan *Backend Website ERP (Enterprise Resource Planning)* Pada Industri Kelapa Sawit” merupakan asli penelitian saya. Pada skripsi ini tdiak terdapat karya yang pernah dilakukan orang lain dan sepengetahuan saya tidak terdapat atau diterbitkan oleh orang lain, kecuali secara tertulis diacu dalam naskah ini sebagaimana yang disebutkan dalam daftar pustaka. Selain itu, saya menyatakan bahwa skripsi ini dibuat oleh saya sendiri.

Apabila pernyataan saya tidak benar, maka saya bersedia dikenai sanksi sesuai dengan hukum yang berlaku.

Bandar Lampung 28 Juni 2025



Frans Aldi Gurning

NPM. 2015031087

RIWAYAT HIDUP



Penulis lahir di Tiga Binanga, Sumatera Utara, Pada tanggal 6 September 2002. Penulis merupakan anak ke dua dari 5 bersaudara, anak dari pasangan Alm Bapak Albert Gurning dan Ibu Nurliana Br Ginting. Penulis menyelesaikan pendidikan sekolah dasar di SDN 040572 Tiga Binanga pada tahun 2015, sekolah menengah pertama di SMP Swasta HKBP Medan pada tahun 2017, sekolah menengah atas di SMAN 1 Kabanjahe pada tahun 2020

Penulis terdaftar sebagai mahasiswa Jurusan Teknik Elektro Universitas Lampung pada tahun 2020 melalui jalur SBMPTN. Selama menjadi mahasiswa penulis aktif mengikuti organisasi Himpunan Mahasiswa Teknik Elektro sebagai Anggota Divisi Minat dan Bakat selama dua periode kepengurusan. Penulis mengambil konsentrasi Telekomunikasi dan Teknik Informasi pada semester 5. Penulis juga merupakan asisten Laboratorium Pengukuran Besaran Listrik pada semester 5. Kemudian, pada Tahun 2022 dan 2023, Penulis mengikuti Studi Independen di *Orbit Future Academy* yang memperdalam keterampilan bahasa pemrograman python.

PERSEMBAHAN

**Biarlah pengharapan di dalam Kristus menjadi jangkar yang kokoh bagi
jiwamu, di tengah badai kehidupan**

Ku persembahkan karya ini kepada

Ayah dan Ibu Tercinta

Albert Gurning dan Nurliana Br Ginting

Saudara-saudaraku tercinta

Mariati Br Gurning

Irna Sari Br Gurning

Sandro Janesra Gurning

Ruth Efenty Br Gurning

Terima kasih untuk dukungan dan doa yang diberikan sehingga saya dapat
menyelesaikan tugas akhir ini

MOTTO HIDUP

"Serahkanlah perbuatanmu kepada TUHAN, maka terlaksanalah segala rancanganmu."

(Amsal 16:3)

"Percayalah kepada TUHAN dengan segenap hatimu, dan janganlah bersandar kepada pengertianmu sendiri. Akuilah Dia dalam segala lakumu, maka Ia akan meluruskan jalanmu."

(Amsal 3:5-6)

"I can do all things through Christ who strengthens me."

(Philippians 4:13)

SAWACANA

Puji syukur penulis ucapkan kehadiran Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya skripsi ini dapat diselesaikan. Skripsi yang berjudul “*Perancangan Backend Website Erp (Enterprise Resource Planning) Pada Industri Kelapa Sawit*” adalah salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung.

Pada kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Ibu Prof. Dr. Ir. Lusmeilia Afriani, D.E.A., I.P.M., selaku Rektor Universitas Lampung.
2. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung.
3. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
4. Bapak Sumadi, S.T., M.T selaku Kepala Program Studi Teknik Elektro Universitas Lampung.
5. Segenap Dosen di Jurusan Teknik Elektro yang telah memberikan ilmu yang bermanfaat, wawasan, dan pengalaman bagi Penulis.
6. Ibu Yetti Yuniati S.T., M.T. selaku Dosen Pembimbing Utama yang telah memberikan ilmu, bimbingan, bantuan, masukan, dan pandangan serta berbagai pola pikir kepada penulis disetiap kesempatan dengan baik dan menarik.
7. Bapak Dr. Ing. Ardian Ulvan, S.T., M.Sc. selaku Dosen Pembimbing Pendamping yang telah memberikan ilmu, bimbingan, masukan, motivasi, dan sudut pandang lain serta berbagai pola pikir kepada penulis disetiap kesempatan dengan baik dan menarik.
8. Bapak Aryanto, S.T., M.T. selaku Dosen Penguji yang telah memberikan kritik dan saran serta penyelesaian masalah yang membangun kepada penulis;

9. Mbak Nurul dan Segenap Staff di Jurusan Teknik Elektro dan Fakultas Teknik yang telah sangat membantu penulis baik dalam hal administrasi dan hal-hal lainnya.
10. Kepada kedua orang tua dan kakak yang telah memberikan motivasi dalam hidup.

Penulis mengucapkan banyak terima kasih kepada semua pihak yang telah membantu penulis. Semoga Tuhan membalas segala bentuk kebaikan yang telah Bapak, Ibu, dan rekan-rekan semua berikan. Penulis berharap semoga skripsi ini dapat bermanfaat untuk kita semua.

Bandar Lampung, 28 Juni 2025

Penulis,

Frans Aldi Gurning

NPM 2015031087

DAFTAR ISI

	Halaman
ABSTRAK	ii
ABSTRACT	iii
LEMBAR PERSETUJUAN.....	iv
LEMBAR PENGESAHAN.....	v
SURAT PERNYATAAN	vi
RIWAYAT HIDUP.....	vii
PERSEMBAHAN.....	viii
MOTTO HIDUP.....	ix
SAWACANA	x
DAFTAR ISI	xii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	2
1.5 Manfaat Penelitian	3
1.6 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu.....	5
2.2 Python	10
2.3 Sawit.....	10
2.4 <i>Backend</i>	12
2.5. Flask	12
2.6 PostgreSQL	12
2.7 <i>Enterprise Resource Planning (ERP)</i>	14
2.8 <i>Scrum Method</i>	17
2.9. Postman	17
2.10 Pengujian <i>Website</i>	19
2.11 Indikator Keberhasilan	20
2.12 <i>Stakeholder</i>	20

BAB III METODOLOGI PENELITIAN	25
3.1 Waktu dan Tempat Penelitian.....	25
3.2 Alat dan Bahan	25
3.3 <i>Capstone Project</i>	26
3.4 Metode Penelitian.....	28
3.4.1 Studi Literatur	29
3.4.2 Analisis Kebutuhan	29
3.4.3 Proses Perancangan Sistem	29
3.5 Sistem <i>Enterprise Resource Planning</i> (ERP).....	31
3.6 Pembuatan <i>Backend</i>	33
3.6.1 <i>Login</i>	33
3.6.2 Perancangan Modul Super Admin	34
3.6.3 Perancangan Modul Petani.....	35
3.6.4 Proses Perancangan Modul Kebun.....	35
3.6.5 Perancangan Modul Logistik	36
3.6.6 Perancangan Modul Pabrik	37
3.6.7 Proses Pengujian	38
3.7 Struktur <i>Database</i>	40
3.7.1 Data Primer	40
3.7.2 Data Transaksional	41
3.8 Parameter Keberhasilan	43
BAB HASIL DAN PEMBAHASAN	45
4.1 Proses Pengembangan Sistem	45
4.2 Pembuatan <i>Database</i>	47
4.2.1 Registrasi <i>server</i> PostgreSQL	48
4.2.2 Create <i>database</i>	50
4.2.3 Konfigurasi <i>database</i> dengan flask.....	51
4.2.4. Pembuatan tabel <i>database</i>	52
4.2.5 Meng- <i>install</i> library	53
4.2.6 Relasi <i>Database</i>	54
4.3. Keamanan Data	55
4.3.1. <i>Hashing Password</i>	55
4.3.2 Autentikasi dan Otorisasi	56
4.3.3 Konfigurasi <i>Session</i>	57
4.4 Pengujian Lokal	58
4.5 <i>Deployment Backend</i>	59
4.5.1 <i>Upload File</i> ke <i>Repository</i> Github	59
4.5.2 <i>Clone repository</i> Github ke dalam <i>Visual Studio Code</i>	62
4.5.3 Konfigurasi Nginx.....	62
4.5.5 Penyelesaian Error Ketika <i>Deployment</i>	65
4.5.6 Penerapan Konsep <i>Blockchain</i>	67
4.6 Pengujian Setelah <i>Deployment</i>	68

4.6.1 Tamu (<i>Guest</i>).....	68
4.6.2 Petani.....	69
4.6.3 Logistik	71
4.6.4 Pabrik	73
4.6.5 Kebun	75
4.6.6 Super Admin	79
4.6.7 Pengujian 100 <i>Virtual User</i>	79
4.7 Analisis Pencapaian.....	80
4.7.1 Sistem <i>Login</i>	81
4.7.2 <i>Traceability</i>	83
4.7.3 <i>Update</i> Pengiriman.....	85
4.7.4 Pelacakan Pengiriman	87
4.7.5 Transparansi harga	88
BAB V KESIMPULAN DAN SARAN	91
5.1 Kesimpulan	91
5.2 Saran.....	91
DAFTAR PUSTAKA.....	93
Lampiran 1	98

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Python	10
Gambar 2.2 Kelapa Sawit	11
Gambar 2.3. Flask Python.....	12
Gambar 2.4 PostgreSQL	13
Gambar 2.5 Sistem ERP.....	14
Gambar 2.6 Arsitektur ERP yang akan dibuat	15
Gambar 2.7 Postman	18
Gambar 2.8 Diagram <i>Use Case</i> Super Admin	21
Gambar 2.9 Diagram <i>Use Case</i> Kebun	21
Gambar 2.10 Diagram <i>Use Case</i> Petani.....	22
Gambar 2.11 Diagram <i>Use Case</i> Pabrik	23
Gambar 2.12 Diagram <i>Use Case</i> Logistik	24
Gambar 3.1 Diagram Keseluruhan Pengembangan <i>Website</i>	27
Gambar 3.2 Diagram Alir Penelitian.....	28
Gambar 3.3 Perancangan <i>Scrum Method</i>	30
Gambar 3.4 <i>Flowchart</i> Perancangan sistem.....	31
Gambar 3.5 Sistem ERP.....	32
Gambar 3.6 Diagram Alir Perancangan Modul <i>Login</i>	33
Gambar 3.7 Diagram Alir Modul Super Admin.....	34
Gambar 3.8 Diagram Alir Modul Petani	35
Gambar 3.9 Diagram Alir Modul Kebun	36
Gambar 3.10 Diagram Alir Modul Logistik.....	37
Gambar 3.11 Diagram Alir Modul Pabrik.....	38
Gambar 3.12 Proses Pengujian <i>Endpoint</i>	39
Gambar 4.1 Proses Pengembangan Sistem	45
Gambar 4.2 Struktur Folder	47
Gambar 4.3 registrasi <i>Server</i>	48
Gambar 4.4. Kredensial <i>Login</i>	49
Gambar 4.5 Kredensial <i>Login</i>	49
Gambar 4.6 Membuat <i>Database</i>	50
Gambar 4.7 Membuat <i>Database</i>	50
Gambar 4.8 Tampilan <i>Database</i>	51
Gambar 4.9 Konfigurasi <i>Database</i>	52

Gambar 4.10 Models.py	53
Gambar 4.12 <i>Hashing Password</i> Menggunakan SHA-256	55
Gambar 4.13 Hasil <i>Hashing Password</i>	56
Gambar 4.14 Autentikasi dan Otorisasi	57
Gambar 4.15 Konfigurasi <i>Session</i>	58
Gambar 4.16 Membuat <i>Repository</i> Github	60
Gambar 4.17 <i>Git Bash Here</i>	60
Gambar 4.18 Terminal Gituhub	61
Gambar 4.19 Perintah Untuk Meng- <i>upload File</i>	61
Gambar 4.20 <i>Deployment</i> Aplikasi	64
Gambar 4.21 pesan error 502	65
Gambar 4.22 Mengecek <i>Error</i>	66
Gambar 4.23 Memulai Kembali <i>Server</i>	67
Gambar 4.24 Tampilan Sukses	67
Gambar 4.25 Pengujian <i>login</i> dengan 100 <i>Virtual User</i>	80
Gambar 4.26 Tampilan <i>Login</i> berhasil	81
Gambar 4.28 Percobaan <i>Login</i> Dengan <i>Password/Username</i> Salah	83
Gambar 4.29 Percobaan <i>Login</i> Dengan <i>Role</i> Salah	83
Gambar 4.30 Fitur <i>Traceability</i>	84
Gambar 4.31 Fitur <i>Traceability</i>	85
Gambar 4.32 Memperoleh kode <i>driver</i>	86
Gambar 4.33 Meng- <i>update</i> pengiriman	86
Gambar 4.34 Kode unik tidak Valid	87
Gambar 4.35 Fitur Pelacakan Pengiriman	88
Gambar 4.36 Harga sawit	89
Gambar 4.37 <i>Update</i> Berat akhir Sawit	89
Gambar 4.38 Riwayat Transaksi Pabrik	90

DAFTAR TABEL

	Halaman
Tabel 2. 1 Analisis dan hasil penelitian terkait.....	7
Tabel 3.2 Data Primer	40
Tabel 3.3 Data Transaksional	42
Tabel 4.1 Pengujian Tamu (<i>Guest</i>).....	69
Tabel 4.2 Pengujian Petani.....	69
Tabel 4.2. Pengujian Petani (Lanjutan).....	70
Tabel 4.2 Pengujian Petani (Lanjutan).....	71
Tabel 4.3 Pengujian Logistik.....	71
Tabel 4.3 Pengujian Logistik (Lanjutan).....	72
Tabel 4.3 Pengujian Logistik (Lanjutan).....	73
Tabel 4.4 Pengujian Pabrik	73
Tabel 4.4 Pengujian Pabrik (Lanjutan).....	74
Tabel 4.4 Pengujian Pabrik (Lanjutan).....	75
Tabel 4.5 Pengujian Kebun	75
Tabel 4.5 Pengujian Kebun (Lanjutan)	76
Tabel 4.5 Pengujian Kebun (Lanjutan)	77
Tabel 4.5 Pengujian Kebun (Lanjutan)	78
Tabel 4.6 Pengujian Super Admin.....	79

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kelapa sawit (*Elaeis guineensis*) adalah tanaman tropis yang berasal dari Afrika Barat dan kini menjadi salah satu komoditas penting di dunia. Tumbuhan ini menghasilkan buah yang kaya akan minyak, yang diekstrak menjadi minyak kelapa sawit dan minyak inti sawit [1]. Indonesia dan Malaysia adalah dua negara produsen terbesar minyak kelapa sawit di dunia. Secara total, keduanya menghasilkan sekitar 85% dari total produksi minyak kelapa sawit global. Indonesia sendiri merupakan produsen dan eksportir terbesar minyak kelapa sawit. Budidaya kelapa sawit telah memberikan manfaat ekonomi signifikan, terutama di daerah pedesaan dengan menyediakan lapangan kerja dan pendapatan bagi banyak petani. [2]

Enterprise Resource Planning (ERP) adalah sistem perangkat lunak yang mengintegrasikan berbagai fungsi bisnis dalam satu *platform*, memungkinkan aliran informasi yang efisien di seluruh departemen. Sistem ini terdiri dari modul-modul seperti akuntansi, manajemen inventaris, sumber daya manusia, dan penjualan, yang dapat disesuaikan dengan kebutuhan organisasi [3]. Menggunakan *database* tunggal, ERP mengurangi redundansi data dan meningkatkan akurasi informasi. Dengan otomatisasi proses bisnis, ERP mengurangi kesalahan manusia dan meningkatkan efisiensi operasional. ERP menyediakan data *real-time* untuk mendukung pengambilan keputusan yang cepat dan berbasis data [4]. Penerapan sistem *Enterprise Resource Planning* (ERP) dalam industri kelapa sawit bertujuan agar semua proses bisnis, mulai dari pembibitan, penanaman, pemanenan, hingga pengolahan dan distribusi, memastikan data dan informasi dapat mengalir dengan lancar antara berbagai departemen.

Berdasarkan hal tersebut, penelitian ini merancang *Backend website* ERP pada kelapa sawit menggunakan python dan PostgreSQL sebagai basis data dengan tujuan perusahaan kelapa sawit dapat meningkatkan produktivitas, mengurangi biaya operasional, dan meningkatkan daya saing di pasar. Implementasi ERP juga mendukung praktik bisnis berkelanjutan dengan menyediakan alat untuk pemantauan dan pelaporan yang lebih baik.

1.2 Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut:

1. Bagaimana merancang *Backend* menggunakan Python dan PostgreSQL untuk mengelola berbagai proses bisnis utama dalam industri kelapa sawit?
2. Bagaimana cara mengimplementasikan konsep teknologi *blockchain* pada model manajemen rantai pasok perusahaan kelapa sawit untuk meningkatkan ketelusuran (*traceability*) produk *Crude Palm Oil* (CPO) dan limbah yang dihasilkan, serta memastikan transparansi dan akuntabilitas dalam seluruh proses bisnis perusahaan?

1.3 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah:

1. Penelitian ini hanya berfokus pada pembuatan *Backend* dari *website Enterprise Resource Planning*.
2. Penelitian ini hanya menggunakan pemrograman python sebagai bahasa utama dan PostgreSQL sebagai *database*.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu:

1. Merancang *Backend* menggunakan Python dan PostgreSQL untuk mengelola berbagai proses bisnis utama dalam industri kelapa sawit.
2. Mengetahui cara mengimplementasikan teknologi *blockchain* pada model manajemen rantai pasok perusahaan kelapa sawit untuk meningkatkan

ketelusuran (*traceability*) produk *Crude Palm Oil* (CPO) dan limbah yang dihasilkan, serta memastikan transparansi dan akuntabilitas dalam seluruh proses bisnis perusahaan.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah :

1. Membantu perusahaan sawit untuk menghemat biaya transaksi sawit
2. Memudahkan para petani sawit dalam menjual sawit
3. Transparansi transaksi sehingga tidak ada pihak yang dirugikan dalam transaksi sawit

1.6 Sistematika Penulisan

Adapun sistematika penulisan dari penelitian ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini menjelaskan terkait latar belakang, tujuan penelitian, manfaat penelitian, rumusan masalah, batasan masalah, dan sistematika penulisan Penelitian ini.

BAB II TINJAUAN PUSTAKA

Pada bab ini menjelaskan tentang penelitian terdahulu dan teori dasar terkait penelitian yang didapatkan dari berbagai sumber seperti jurnal, penelitian ilmiah, dan buku yang digunakan oleh penulis untuk penulisan laporan penelitian ini.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan terkait waktu dan tempat, pelaksanaan penelitian, metode penelitian, proses perancangan, proses pengujian, dan parameter keberhasilan penelitian.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan dan menganalisis hasil dari data yang diperoleh yaitu perhitungan dan analisis sebagai pembahasan dari penelitian ini.

BAB V KESIMPULAN DAN SARAN

Pada bab ini berisi tentang kesimpulan dari hasil penelitian serta saran untuk penelitian selanjutnya

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian sebelumnya berjudul "Implementasi *Enterprise Resource Planning* (ERP) Pada Usaha Pithik Sambel Ndeso Berbasis Odoo" yang dilakukan oleh Suminten, Penelitian ini berfokus pada penyelesaian masalah operasional proses bisnis yang masih dilakukan secara manual dan belum terkomputerisasi di usaha Pithik Sambel Ndeso, sebuah usaha kuliner yang spesialisasi dalam olahan ayam organik. Masalah-masalah tersebut termasuk proses bisnis yang tidak efisien dan kesulitan dalam mengelola informasi tentang pembelian, stok bahan baku, supplier, penjualan, karyawan, dan pelanggan. Metodologi yang digunakan adalah penerapan teknologi *Enterprise Resource Planning* (ERP) untuk menyelesaikan masalah-masalah tersebut. *Software* yang dipilih untuk implementasi sistem ERP adalah Odoo (*open ERP*), yang dipilih karena bersifat *open source*, mudah digunakan, dapat beradaptasi dengan perubahan, dan mampu terintegrasi secara menyeluruh dengan kebutuhan operasional perusahaan. Tujuannya adalah untuk membuat kegiatan operasional bisnis menjadi lebih efektif dan efisien melalui sistem ERP [5].

Penelitian selanjutnya berjudul "Perancangan dan Pembuatan Web ERP untuk PT Cipta Tekno Mandiri Menggunakan *Framework Yii 2*" oleh Taufik Hidayat dan Rasyidah. Penelitian ini berfokus pada pengembangan sistem *Enterprise Resource Planning* (ERP) berbasis web untuk PT Cipta Tekno Mandiri, sebuah *Software house* yang menawarkan jasa pengembangan *Software*, layanan domain dan hosting, serta kursus programming. Sistem ERP ini dirancang untuk mengintegrasikan setiap departemen agar dapat saling bertukar informasi secara efisien. Metode yang digunakan dalam penelitian ini adalah model *waterfall*, yang

melibatkan tahapan pengumpulan data, desain, pengkodean, pengujian, dan pemeliharaan. *Framework* yang digunakan untuk pengembangan ERP ini adalah Yii 2. Tujuan penelitian ini adalah untuk mengembangkan sistem ERP berbasis web menggunakan *Framework* Yii 2 bagi PT Cipta Tekno Mandiri. Sistem ini diharapkan dapat mempercepat dan mempermudah proses berbagi informasi antar departemen, meningkatkan efisiensi operasional, dan memfasilitasi integrasi yang lebih baik di dalam perusahaan [6].

Penelitian selanjutnya berjudul “Rancangan Implementasi *Enterprise Resource Planning* (ERP) pada Sistem Pengelolaan *Sales Order* PT Jaya Mandiri Indotech” oleh Triana Fatmawati. Penelitian ini berfokus pada integrasi dan peningkatan efisiensi pengelolaan *Sales order* di PT Jaya Mandiri Indotech, sebuah perusahaan manufaktur komponen berbahan dasar logam, melalui implementasi sistem ERP Odoo modul *Sales*. Metode pengembangan sistem yang digunakan adalah model incremental. Penelitian ini bertujuan untuk mengintegrasikan aliran data antara bagian yang terlibat dalam pengelolaan *order customer* dengan konfigurasi sistem ERP Odoo modul *Sales*. Tujuannya adalah untuk mengurangi redundansi dan perbedaan data, memungkinkan pembuatan laporan yang akurat, serta mendukung pengontrolan dan pengambilan keputusan yang lebih efisien dan efektif dalam pengelolaan *sales order* [7].

Penelitian selanjutnya berjudul Perancangan Sistem *Enterprise Resource Planning* (ERP) Inventory di Toko Bangunan Sederhana Mandiri Jaya Sejahtera. Penelitian ini berfokus pada pengembangan sistem ERP untuk mengatasi masalah pengelolaan data barang, transaksi pembelian, penjualan, dan pelaporan di Toko Bangunan Sederhana Mandiri Jaya Sejahtera, yang saat ini masih menggunakan sistem manual. Perancangan Sistem *Enterprise Resource Planning* (ERP) Inventory di Toko Bangunan Sederhana Mandiri Jaya Sejahtera. Metode yang digunakan dalam penelitian ini adalah model *waterfall*, dengan penggunaan modul *inventory* ERP untuk mengelola data barang dan transaksi, serta Bootstrap dan CSS untuk pengembangan tampilan web. Sistem ini juga mencakup tiga hak akses pengguna: admin, pekerja, dan pemilik toko. Penelitian ini bertujuan untuk mengembangkan sistem ERP yang mempermudah pengelolaan informasi secara real-time,

mengurangi kesalahan dalam pembelian barang, mengelola penjualan dengan lebih baik, dan mendapatkan laporan stok barang yang akurat [8].

Penelitian selanjutnya berjudul “Rancang Bangun *Sistem Enterprise Resource Planning (ERP)* Berbasis Web pada PT Sainsgo Karya Indonesia Menggunakan Metode Scrum”. Penelitian ini berfokus pada pengembangan sistem ERP untuk mengatasi masalah operasional di PT Sainsgo Karya Indonesia, yang sebelumnya mengelola proses bisnis secara manual menggunakan Microsoft Excel sehingga menimbulkan berbagai kendala seperti pengelolaan pembelian dan persediaan yang tidak terintegrasi. Metode pengembangan yang digunakan adalah Scrum dengan teknologi yang mencakup Golang untuk *backend*, JavaScript untuk *frontend*, dan PostgreSQL sebagai basis data. Sistem ini dirancang sebagai solusi *Software as a Service (SaaS)* yang mengintegrasikan modul pembelian, persediaan, produksi, penjualan, dan keuangan. Tujuan penelitian ini adalah untuk menciptakan sistem yang efisien dan terintegrasi guna memperbaiki pengelolaan proses bisnis, serta menawarkan solusi ERP yang terjangkau bagi Usaha Mikro Kecil dan Menengah (UMKM) binaan perusahaan [9].

Tabel analisis dan hasil penelitian terkait dapat dilihat pada Tabel. 2.1.

Tabel 2. 1 Analisis dan hasil penelitian terkait.

Judul	Tahun	Metode	Analisis
Implementasi <i>Enterprise Resource Planning (ERP)</i> Pada Usaha Pithik Sambel Ndeso Berbasis Odoo	2019	Studi kasus dan implementasi sistem ERP menggunakan <i>software</i> Odoo	Implementasi Odoo berhasil mengatasi masalah pada pembelian, stok bahan baku, supplier, penjualan, karyawan, dan customer. Sistem ERP ini membuat operasional usaha lebih efektif dan efisien.

Tabel 2. 1 Analisis dan hasil penelitian terkait.

Judul	Tahun	Metode	Analisis
Perancangan dan Pembuatan Web ERP untuk PT Cipta Tekno Mandiri Menggunakan <i>Framework Yii 2</i>	2021	Metode <i>waterfall</i> , yang meliputi tahap pengumpulan data, desain, pengkodean, pengujian, dan pemeliharaan	Sistem ERP berbasis web berhasil diimplementasikan, memungkinkan integrasi antar departemen dan mempercepat proses berbagi informasi. <i>Framework Yii 2</i> digunakan untuk mempercepat pembuatan aplikasi web
Rancangan Implementasi <i>Enterprise Resource Planning (ERP)</i> pada Sistem Pengelolaan Sales Order PT Jaya Mandiri Indotech	2022	Model incremental, konfigurasi sistem ERP Odoo modul Sales	Konfigurasi sistem ERP doo modul Sales Sistem ERP berhasil mengintegrasikan aliran data antara bagian yang terlibat dalam pengelolaan order customer, mengurangi redundansi dan perbedaan data, serta memungkinkan pembuatan laporan yang akurat. Implementasi ini membuat pengelolaan sales order lebih efisien dalam waktu dan sumber daya, serta mengurangi risiko kehilangan data
Perancangan <i>Sistem Enterprise Resource Planning (ERP) Inventory</i> di Toko	2023	Metode <i>waterfall</i> , menggunakan Bootstrap untuk tampilan web dan CSS untuk desain	Sistem ERP berbasis web berhasil diimplementasikan, mengatasi masalah pengelolaan data barang, transaksi pembelian dan

Bangunan Sederhana Mandiri Jaya Sejahtera			penjualan, serta laporan yang sebelumnya tidak akurat. Sistem ini meningkatkan efisiensi dan akurasi pengelolaan inventaris
Rancang Bangun Sistem <i>Enterprise Resource Planning</i> (ERP) Berbasis Web pada PT Sainsgo Karya Indonesia Menggunakan Metode Scrum	2024	Metode Scrum	Sistem ERP yang dibangun dengan Scrum berhasil mengintegrasikan proses bisnis manual (pembelian, persediaan, produksi, penjualan, keuangan) menjadi lebih efisien. Sistem ini juga dirancang sebagai solusi <i>Software as a Service</i> (SaaS) yang terjangkau bagi UMKM binaan.

Berdasarkan penelitian-penelitian sebelumnya, terdapat perbedaan yang signifikan dengan penelitian yang dilakukan oleh peneliti, terutama pada metode dan *Backend* yang digunakan dalam pembuatan sistem *Enterprise Resource Planning* (ERP.) Penelitian sebelumnya didominasi oleh penggunaan *software* Odoo dengan metode *waterfall*. Namun, penulis menggunakan bahasa pemrograman Python dengan PostgreSQL sebagai basis data dan Flask sebagai API. Selain itu, penulis menggunakan metode *Scrum*, yang berfokus pada iterasi dan peningkatan berkelanjutan, untuk memastikan fleksibilitas dan *responseivitas* sistem terhadap perubahan kebutuhan. Dengan pendekatan ini, penulis berharap dapat mengembangkan sistem *Enterprise Resource Planning* yang lebih fleksibel dibandingkan solusi tradisional.

2.2 Python

Python adalah bahasa pemrograman tingkat tinggi yang diciptakan oleh Guido van Rossum dan pertama kali dirilis pada tahun 1991. Dikenal karena sintaksisnya yang bersih dan mudah dibaca, Python telah berkembang pesat dan menjadi salah satu bahasa pemrograman paling populer di dunia. Python mendukung berbagai paradigma pemrograman seperti berorientasi objek, fungsional, dan imperatif, serta dilengkapi dengan perpustakaan standar yang luas [10]. Bahasa python digunakan dalam berbagai aplikasi termasuk pengembangan web dengan *Framework* seperti Django dan Flask, analisis data dan ilmu data dengan perpustakaan seperti NumPy dan Pandas, serta pembelajaran mesin dengan TensorFlow dan *scikit-learn*. Logo python dapat dilihat pada Gambar 2.1



Gambar 2.1 Python

(Sumber: <https://www.python.org/community/logos/>)

Kelebihan Python meliputi sintaksis yang mudah dipelajari, produktivitas tinggi, dan fleksibilitas yang luas. Namun, Python memiliki beberapa kekurangan seperti kinerja yang lebih lambat dibandingkan bahasa pemrograman yang lain [11].

2.3 Sawit

Kelapa sawit (*Elaeis guineensis*) adalah tanaman tropis yang berasal dari Afrika Barat, terutama di wilayah antara Angola dan Gambia. Pada abad ke-19, kelapa sawit diperkenalkan ke Asia Tenggara oleh para penjajah Belanda dan Inggris. Di Indonesia, penanaman komersial kelapa sawit dimulai pada awal abad ke-20, dengan perkebunan pertama didirikan di Deli, Sumatera Utara, pada tahun 1911 [12]. Sejak itu, industri kelapa sawit berkembang pesat dan menjadi salah satu

sektor penting dalam perekonomian Indonesia. Pemanfaatan kelapa sawit di Indonesia sangat beragam. Minyak sawit mentah (CPO) digunakan sebagai bahan baku untuk berbagai produk seperti minyak goreng, margarin, sabun, dan deterjen. Selain itu, minyak sawit juga digunakan dalam industri kosmetik dan farmasi, serta sebagai bahan bakar nabati (*biofuel*) [13]. Industri kelapa sawit berperan dalam menciptakan lapangan pekerjaan dan menggerakkan perekonomian di banyak daerah pedesaan di Indonesia. Industri kelapa sawit di Indonesia memainkan peran vital dalam perekonomian, khususnya dalam menciptakan lapangan pekerjaan dan mendorong pertumbuhan ekonomi di daerah pedesaan. Perkebunan kelapa sawit menyediakan pekerjaan bagi jutaan orang, mulai dari pekerja di kebun hingga tenaga kerja di sektor pengolahan. Pendapatan dari sektor kelapa sawit juga menjadi sumber utama bagi banyak pekerja di daerah penghasil sawit.

Berdasarkan data Dinas Perkebunan Provinsi Lampung tahun 2021, perkebunan sawit rakyat di Provinsi Lampung diantaranya tersebar di Kab. Tulang Bawang dengan Luas 18.922 Ha (47.140 ton), Kab. Mesuji dengan luas 21.600 Ha (37.151 ton) dan Kab. Lampung Tengah dengan Luas 19.179 Ha (43.149 ton) [14]. Bentuk kelapa sawit dapat dilihat pada Gambar 2.2.



Gambar 2.2 Kelapa Sawit

(Sumber: <https://disbun.jabarprov.go.id/page/view/54-id-kelapa-sawit>)

2.4 Backend

Backend adalah bagian dari sistem aplikasi atau website yang menjalankan proses di balik layar, termasuk pengolahan data, manajemen server, dan interaksi dengan database. *Backend* bertanggung jawab untuk menambahkan, mengubah, menghapus, dan mengelola data yang tidak langsung terlihat oleh pengguna. Dalam arsitektur *Model-View-Controller (MVC)*, *backend* biasanya mencakup bagian model dan controller yang mengatur logika bisnis dan komunikasi dengan database [15].

2.5. Flask

Flask adalah sebuah *framework web* yang ditulis dalam Python. Dikenal sebagai *micro-framework* karena tidak memerlukan alat atau pustaka khusus dan memiliki *database* bawaan [16]. Flask menyediakan alat dasar untuk *routing* URL, manajemen permintaan dan *response* HTTP, serta *template rendering* menggunakan Jinja2. Kelebihan utama Flask terletak pada kesederhanaan dan fleksibilitasnya, menjadikannya pilihan populer untuk proyek kecil hingga menengah serta aplikasi prototipe [17]. Dengan dokumentasi yang lengkap dan komunitas yang aktif, Flask berguna untuk membangun aplikasi *web* yang *scalable* dan mudah dikelola. Logo Flask dapat dilihat pada Gambar 2.3.



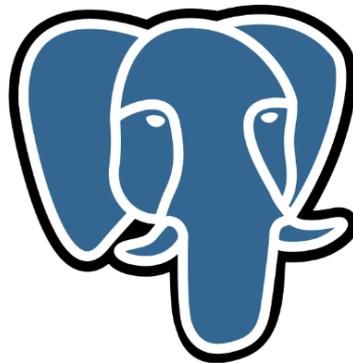
Gambar 2.3. Flask Python

(Sumber: https://flask.palletsprojects.com/en/stable/_images/flask-horizontal.svg)

2.6 PostgreSQL

PostgreSQL adalah sistem manajemen basis data relasional yang dikenal karena kehandalan, kekayaan fitur, dan skalabilitasnya. Dikembangkan sejak 1986 di

Universitas California, Berkeley [18]. PostgreSQL telah berkembang menjadi salah satu RDBMS *open-source* paling populer di dunia [19]. Fitur-fitur utama PostgreSQL mencakup dukungan untuk transaksi ACID (*Atomicity, Consistency, Isolation, Durability*), kemampuan untuk menangani beban kerja yang besar dan kompleks, serta kompatibilitas dengan standar SQL. Selain itu, PostgreSQL mendukung tipe data yang luas, termasuk JSON, XML, dan *hstore*, yang memungkinkannya menangani data semi-terstruktur dengan efisien. PostgreSQL juga memiliki sistem ekstensi yang sangat fleksibel, memungkinkan pengguna untuk menambah fungsionalitas tambahan tanpa mengubah kode sumber inti. Ekstensi ini mencakup PostGIS untuk data spasial dan *hstore* untuk penyimpanan *key-value*. Logo PostgreSQL dapat dilihat pada Gambar 2.4.



Gambar 2.4 PostgreSQL

(Sumber: <https://www.postgresql.org/media/img/about/press/elephant.png>)

PostgreSQL mendukung replikasi, baik replikasi *master-slave* untuk tujuan *failover* dan ketersediaan tinggi, maupun replikasi multi-master untuk distribusi beban kerja [20]. Dengan keamanan yang ketat, dukungan untuk otentikasi berbagai metode, dan kontrol akses yang rinci, PostgreSQL adalah pilihan yang kuat untuk aplikasi-aplikasi yang memerlukan manajemen data yang aman dan dapat diandalkan. Selain itu, PostgreSQL menyediakan alat-alat pengelolaan yang kuat seperti pgAdmin, serta kemampuan untuk integrasi dengan berbagai bahasa pemrograman seperti Python, Java, dan C++ [21]. Dengan komunitas yang aktif dan dukungan berkelanjutan dari pengembang di seluruh dunia, PostgreSQL terus berevolusi dan memperkenalkan fitur-fitur baru untuk memenuhi kebutuhan berbagai jenis aplikasi modern. Implementasi prosedur penyimpanan (*stored procedures*), *triggers*, dan

pandangan (*views*) memungkinkan pengembang untuk menciptakan logika aplikasi yang kompleks langsung di dalam basis data, yang mengurangi beban aplikasi klien dan meningkatkan efisiensi.

2.7 Enterprise Resource Planning (ERP)

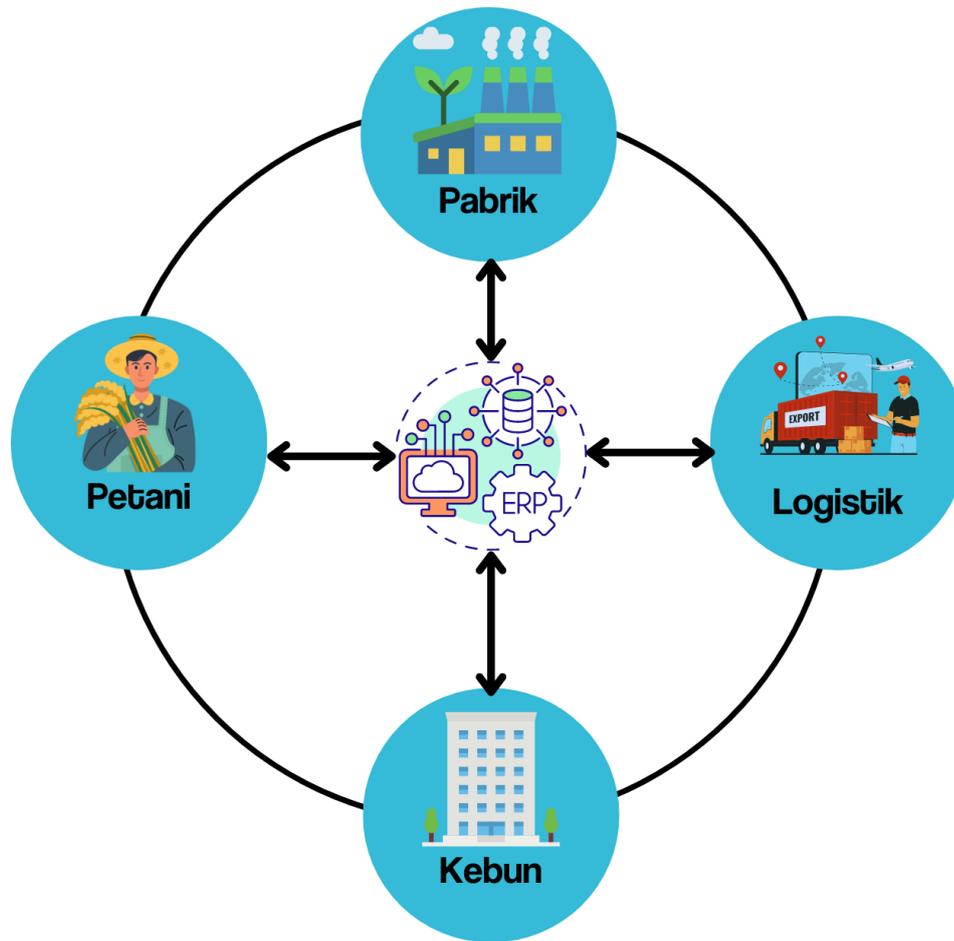
Enterprise Resource Planning (ERP) adalah metode pengelolaan sumber daya perusahaan melalui penggunaan Teknologi Informasi (TI). *Enterprise Resource Planning* (ERP) merupakan sistem perangkat lunak yang terintegrasi untuk mengelola dan mengoordinasikan berbagai fungsi bisnis dalam organisasi. ERP berfokus pada integrasi dan otomatisasi proses bisnis utama seperti manajemen keuangan, sumber daya manusia, produksi, rantai pasokan, penjualan, dan layanan pelanggan [22]. Dengan menggunakan basis data terpusat, ERP memungkinkan informasi yang relevan untuk diakses secara *real-time* oleh semua departemen, yang meningkatkan efisiensi operasional dan akurasi data. Sistem ini mengurangi duplikasi data dan mempercepat Alir kerja, serta memberikan kemampuan pengambilan keputusan yang lebih baik melalui akses informasi yang komprehensif dan terstruktur. Sistem ERP dapat dilihat pada Gambar 2.5.



Gambar 2.5 Sistem ERP

(Sumber: <https://killuazoldyck10.wordpress.com/wp-content/uploads/2013/07/ec-erp-system-overview.jpg>)

Arsitektur ERP yang akan dibuat pada penelitian ini dapat dilihat pada Gambar 2.6.



Gambar 2.6 Arsitektur ERP yang akan dibuat

Berdasarkan arsitektur diatas terdapat entitas seperti pabrik, petani, perusahaan, dan logistik yang memiliki peran yang penting dalam rantai pasokan dan manajemen sumber daya. Arsitektur ERP yang ditampilkan menunjukkan bagaimana berbagai entitas dalam rantai produksi dan distribusi terhubung melalui sistem ERP (*Enterprise Resource Planning*). Berikut adalah penjelasan singkat mengenai masing-masing entitas:

1. Petani

Petani Sawit adalah individu atau kelompok yang menanam dan mengelola kebun sawit untuk memproduksi buah sawit. Dalam konteks ERP, modul yang terkait dengan Petani Sawit sering kali merupakan bagian dari *Supply Chain Management* (SCM). ERP membantu petani sawit dalam

merencanakan penanaman, pemeliharaan kebun sawit, pengelolaan panen, dan pemantauan hasil produksi. Sistem ERP memungkinkan Petani Sawit untuk meningkatkan produktivitas lahan, mengelola inventaris tanaman dengan lebih baik, dan merencanakan distribusi hasil panen secara efisien.

2. Kantor Kebun

Kantor kebun adalah entitas yang memiliki dan mengoperasikan sistem ERP untuk mengelola operasional bisnis mereka, termasuk pengelolaan pabrik sawit dan koordinasi dengan petani sawit. Dalam konteks ERP, peran kantor mencakup manajemen keuangan, manajemen persediaan, penjualan, dan mungkin juga modul CRM (*Customer Relationship Management*) untuk hubungan dengan pelanggan. ERP membantu perusahaan sawit untuk mengelola operasi bisnis mereka dengan lebih efisien, meningkatkan visibilitas atas rantai pasokan sawit, dan meningkatkan layanan kepada pelanggan.

3. Pabrik

Pabrik adalah unit pengolahan yang menerima buah sawit dari petani sawit, mengolahnya, dan menghasilkan produk akhir seperti minyak kelapa sawit (cpo), minyak inti sawit, atau produk turunan lainnya. Dalam sistem erp, modul yang terkait dengan pabrik sawit mencakup *Manufacturing Resource Planning* (MRP) yang membantu dalam perencanaan produksi, manajemen inventaris bahan baku (buah sawit), pengelolaan proses pengolahan, dan kontrol kualitas produk. ERP membantu pabrik sawit untuk meningkatkan efisiensi operasional, mengoptimalkan penggunaan bahan baku, dan memastikan kualitas produk yang dihasilkan sesuai dengan standar yang ditetapkan.

4. Logistik

Logistik mencakup pengelolaan aliran barang, informasi, dan sumber daya dari titik asal (petani sawit) hingga titik konsumsi atau penggunaan (pabrik sawit). Dalam konteks ERP, modul logistik membantu dalam perencanaan transportasi, manajemen gudang, pengiriman, dan pengelolaan rantai

pasokan secara keseluruhan. ERP memungkinkan perusahaan untuk mengoptimalkan rute pengiriman produk sawit, mengelola inventaris dengan efisien, dan memastikan pengiriman tepat waktu kepada pabrik sawit.

2.8 Scrum Method

Scrum adalah sebuah metode dalam pengembangan perangkat lunak yang merupakan bagian dari *framework Agile* [24]. Metode ini dirancang untuk membantu tim bekerja lebih efisien dan efektif dengan berfokus pada kolaborasi, fleksibilitas, dan kemampuan untuk *meresponse* perubahan dengan cepat. Dalam *Scrum*, pekerjaan dilakukan dalam iterasi yang disebut *Sprint*, biasanya berlangsung selama 2-4 minggu. Setiap *Sprint* diawali dengan *Sprint Planning*, dimana tim merencanakan pekerjaan yang akan dilakukan selama *Sprint*. Peran-peran khusus dalam *Scrum* meliputi *Scrum Master* yang bertanggung jawab untuk memastikan proses berjalan lancar dan mengatasi hambatan, serta *Product Owner* yang bertugas memprioritaskan backlog produk berdasarkan nilai bisnis. Di tengah-tengah *Sprint*, tim mengadakan *Daily Standup*, sebuah pertemuan singkat untuk menyelaraskan pekerjaan dan mengidentifikasi hambatan. Pada akhir *Sprint*, dilakukan *Sprint Review* untuk menilai hasil kerja dan menerima masukan dari pemangku kepentingan, diikuti dengan *Sprint Retrospective* untuk refleksi dan peningkatan proses. *Scrum* menekankan pada transparansi, inspeksi, dan adaptasi, dengan tujuan membantu tim tetap fokus pada tujuan akhir sambil fleksibel dalam menghadapi perubahan dan tantangan. Selain itu, *Scrum* mendorong kolaborasi erat antara tim pengembang dan pemangku kepentingan melalui komunikasi terus-menerus dan keterlibatan aktif. Dengan struktur yang jelas namun fleksibel, *Scrum* memungkinkan tim mengelola proyek kompleks dengan lebih baik, meningkatkan kepuasan pelanggan, dan menghasilkan produk dengan nilai bisnis tinggi.

2.9. Postman

Postman adalah alat yang digunakan oleh pengembang perangkat lunak untuk mengembangkan, menguji, dan mendokumentasikan *Application Programming*

Interfaces (API). Alat ini menyediakan antarmuka pengguna grafis yang memungkinkan pengembang untuk mengirim permintaan HTTP ke *server* dan menganalisis *response* yang diterima, yang sering kali berupa data dalam format JSON atau XML. Logo Postman dapat dilihat pada Gambar 2.7.



Gambar 2.7 Postman

(Sumber: <https://www.postman.com/legal/logo-usage/>)

Salah satu fitur utama Postman adalah kemampuannya untuk membuat dan mengelola koleksi permintaan API, yang memungkinkan pengembang untuk menyimpan dan mengorganisasi berbagai jenis permintaan dalam satu tempat. Koleksi ini memudahkan kolaborasi antara anggota tim, yang memungkinkan pengujian dan pemeliharaan API lebih efisien. Postman juga mendukung pengujian otomatis menggunakan skrip pengujian untuk memverifikasi apakah API berfungsi sesuai yang diinginkan, yang sangat berguna dalam pengembangan perangkat lunak modern yang menggunakan metodologi DevOps dan CI/CD (*Continuous Integration/Continuous Deployment*) [25]. Selain itu, Postman menyediakan fitur monitoring, yang memungkinkan pengujian berkala terhadap API untuk memastikan kinerja dan stabilitasnya dalam jangka panjang. Meskipun Postman sangat berguna, tantangan tetap ada dalam pengujian API yang lebih kompleks, seperti pengujian beban atau pengujian terhadap API yang sangat terdistribusi, dimana alat lain seperti JMeter atau Newman dapat digunakan untuk melengkapi fungsionalitas Postman

2.10 Pengujian *Website*

Pengujian *website* adalah langkah penting dalam proses pengembangan aplikasi web untuk memastikan bahwa semua fungsi bekerja sesuai dengan yang diharapkan dan pengalaman pengguna optimal. Pengujian ini umumnya dilakukan melalui tiga cara utama: uji fungsionalitas, uji *database*, dan integrasi dengan *front-end*.

Uji Fungsionalitas adalah tahap pengujian yang fokus pada memastikan bahwa setiap fitur dan fungsi dari *website* berjalan dengan benar. Dalam tahap ini, penguji akan memeriksa berbagai aspek seperti formulir pendaftaran, fitur pencarian, proses checkout, dan validasi data. Uji fungsionalitas bertujuan untuk memastikan bahwa semua elemen interaktif di *website* berfungsi sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini mencakup skenario penggunaan yang berbeda-beda untuk menilai bagaimana sistem merespons berbagai *input* dan tindakan pengguna.

Uji *database* adalah pengujian yang memastikan bahwa interaksi antara aplikasi dan basis data berjalan lancar dan benar. Pengujian ini mencakup verifikasi terhadap operasi CRUD (*Create, Read, Update, Delete*) untuk memastikan bahwa data disimpan, diambil, diperbarui, dan dihapus dengan benar dalam *database*. Selain itu, uji *database* juga memastikan bahwa integritas data terjaga, misalnya dengan melakukan uji konsistensi data, uji keutuhan transaksi, serta uji keamanan data untuk melindungi terhadap akses yang tidak sah atau pelanggaran data. *Database testing* juga mengecek performa *query* untuk memastikan bahwa waktu *response* tetap optimal, bahkan dengan beban data yang besar.

Integrasi dengan *frontend* adalah proses pengujian yang memastikan bahwa back end dan front end aplikasi terintegrasi dengan baik. Pengujian ini melibatkan pemeriksaan bagaimana data dari *server* ditampilkan di antarmuka pengguna dan sebaliknya, bagaimana input pengguna dikirim ke *server* untuk diproses. Ini termasuk pengujian API, layanan web, dan koneksi antara berbagai lapisan aplikasi. Penguji akan mengecek bahwa semua *endpoint* API bekerja dengan benar, data ditransfer dengan aman, dan *response server* ditangani dengan tepat oleh *frontend*. Integrasi ini juga mencakup pengujian antarmuka pengguna untuk memastikan bahwa elemen visual seperti tombol, *form*, dan navigasi bekerja tanpa masalah dan tampil sesuai dengan desain yang diinginkan.

Pengujian melalui ketiga cara ini sangat penting untuk menjamin kualitas dan keandalan sebuah *website*. Setiap metode pengujian memberikan perspektif yang berbeda namun saling melengkapi, memastikan bahwa *website* tidak hanya berfungsi dengan baik tetapi juga aman dan memberikan pengalaman pengguna yang optimal.

2.11 Indikator Keberhasilan

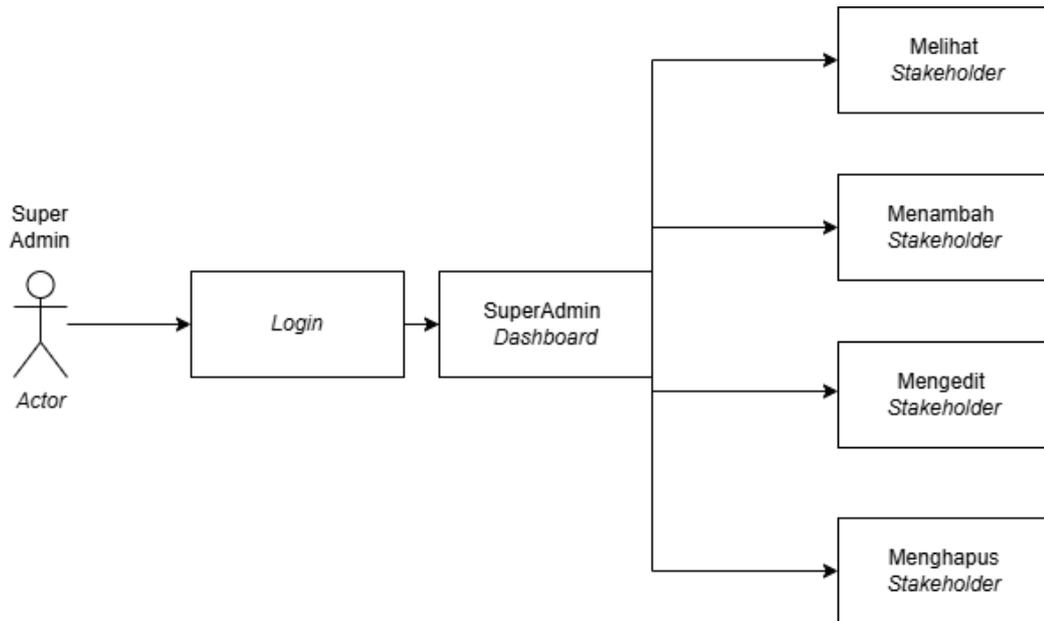
Pengujian yang digunakan dalam implementasi sistem *traceability* untuk industri kelapa sawit adalah *black-box testing*. Metode ini fokus pada pengujian persyaratan fungsional program tanpa memperhatikan struktur internal dari aplikasi. Pengujian black-box bertujuan untuk menemukan bug atau masalah yang ada pada program dengan mengevaluasi fungsionalitas berdasarkan input yang diberikan dan output yang dihasilkan. Teknik *black-box testing* yang akan diterapkan dalam penelitian ini adalah *all-pair testing*. Metode ini melibatkan pengujian seluruh kemungkinan kombinasi parameter input aplikasi, dengan tujuan untuk mengidentifikasi potensi masalah yang mungkin terjadi akibat interaksi antar bagian aplikasi. Jika hasil pengujian menunjukkan bahwa sebagian besar kombinasi input menghasilkan nilai yang sesuai dengan yang diharapkan, maka hasil pengujian dapat dianggap baik, yang menandakan bahwa aplikasi berfungsi dengan optimal.

2.12 Stakeholder

Pada penelitian ini, membangun sebuah sistem dengan proses komunikasi antar *stakeholder*. *Stakeholder* terkait yaitu super admin, petani, kebun, logistik, dan pabrik. Pada tiap-tiap *stakeholder* memiliki master data, dan *session* (*objective* dan *features*) dalam sebuah sistem aplikasi seperti pada analisis kebutuhan atau *requirements project*.

1. Super Admin

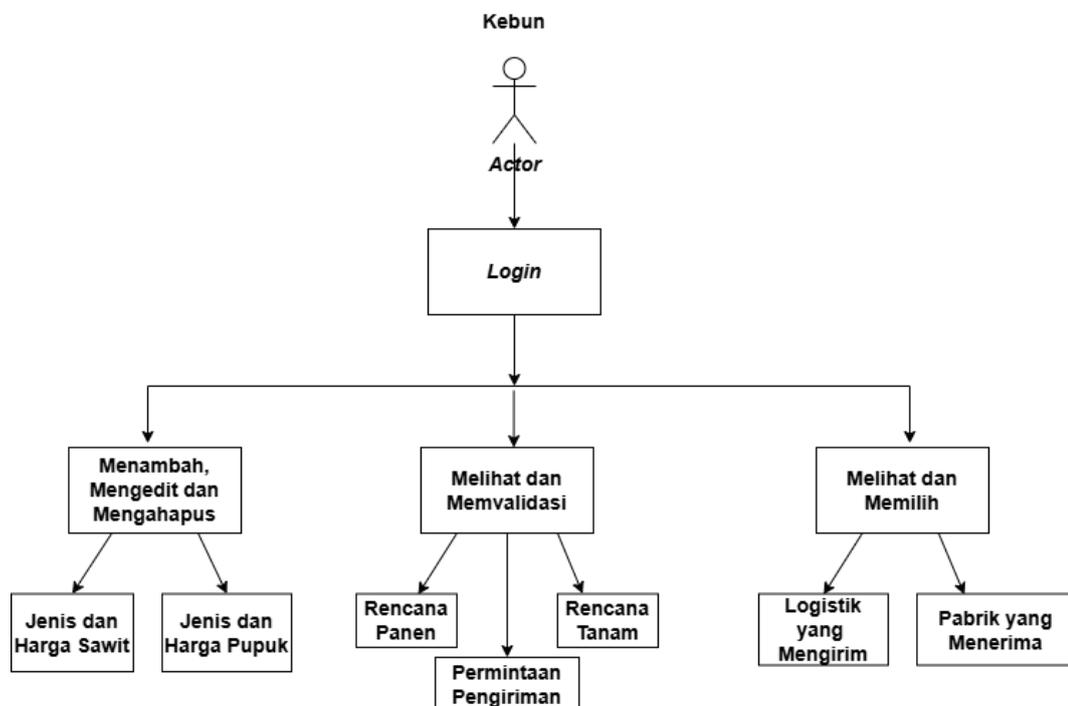
Super admin entitas yang mengatur *stakeholder*, superadmin dapat menambah, mengedit dan menghapus *stakeholder*. Bagan super admin dapat dilihat pada Gambar 2.8.



Gambar 2.8 Diagram Use Case Super Admin

2. Kebun

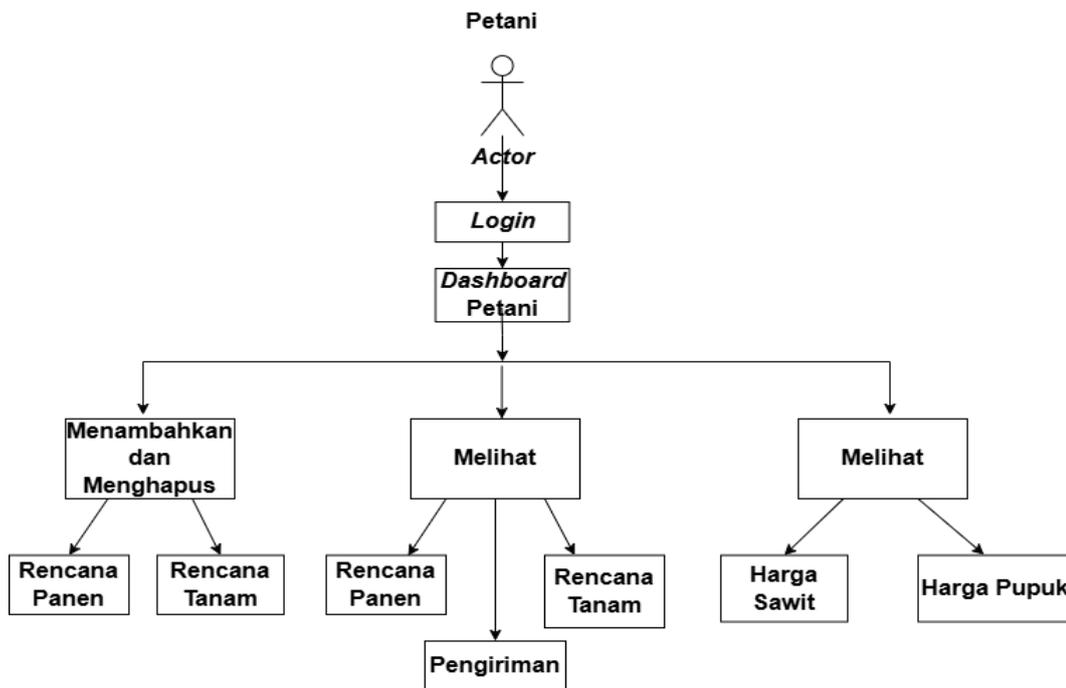
Kantor kebun adalah entitas yang mengoperasikan sistem ERP untuk mengelola operasional bisnis mereka, termasuk pengelolaan pabrik sawit dan koordinasi dengan petani sawit. Diagram *use case* kebun dapat dilihat pada Gambar 2.9.



Gambar 2.9 Diagram Use Case Kebun

3. Petani

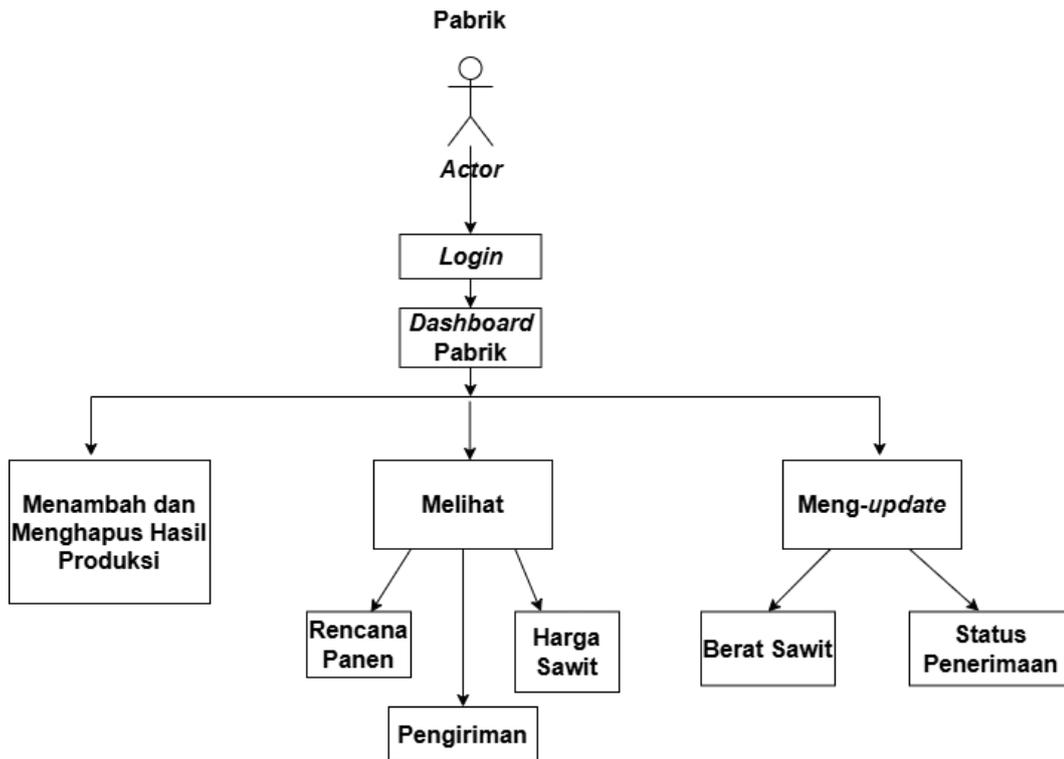
Petani Sawit adalah individu yang menanam dan mengelola kebun sawit untuk memproduksi buah sawit. Diagram *use case* Petani dapat dilihat pada Gambar 2.10



Gambar 2.10 Diagram *Use Case* Petani

4. Pabrik

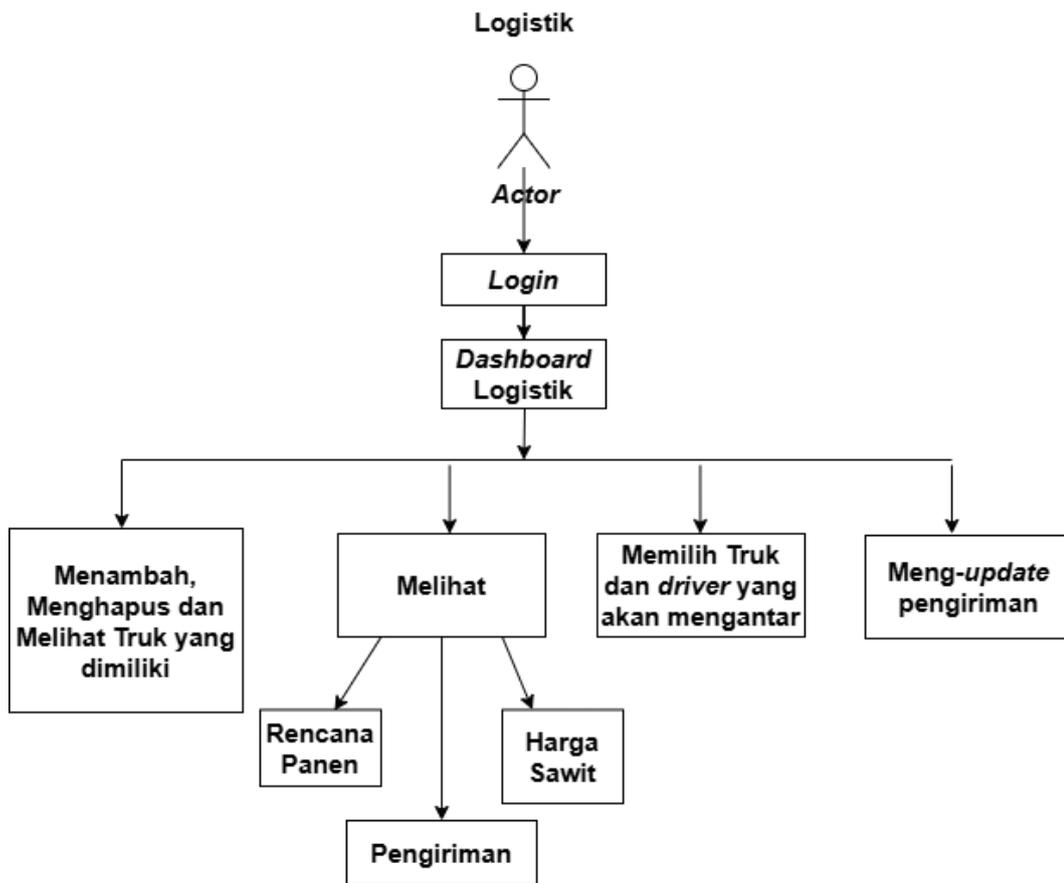
Pabrik adalah unit pengolahan yang menerima buah sawit dari petani sawit, mengolahnya, dan menghasilkan produk akhir seperti minyak kelapa sawit (cpo), minyak inti sawit, atau produk turunan lainnya Diagram *use case* Pabrik dapat dilihat pada Gambar 2.11.



Gambar 2.11 Diagram *Use Case* Pabrik

5. Logistik

Logistik mencakup pengelolaan aliran barang, informasi, dan sumber daya dari titik asal (petani sawit) hingga titik penggunaan (pabrik sawit). Diagram *use case* Petani dapat dilihat pada Gambar 2.12.



Gambar 2.12 Diagram *Use Case* Logistik

BAB III

METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Pada pelaksanaan penelitian ini dilakukan pada:

Waktu : Agustus 2024 - Januari 2025

Tempat : Laboratorium Pengukuran Besaran Listrik, Fakultas Teknik,
Universitas Lampung.

3.2 Alat dan Bahan

Alat dan bahan merupakan dua komponen penting yang digunakan untuk melakukan penelitian. Keduanya memiliki peran yang berbeda namun sama-sama penting dalam memastikan keberhasilan dan validitas penelitian.

Alat dan bahan yang digunakan dalam pengerjaan penelitian ini adalah sebagai berikut :

1. *Hardware*

Adapun *Hardware* yang digunakan dalam penelitian ini yaitu:

- a) Laptop, dengan spesifikasi :
 - Mikroprosesor: core i3-6006u 2 GHz
 - Memori: 4 GB
 - Media Penyimpanan: SSD 256 GB

b) *Keyboard*

c) *Mouse*

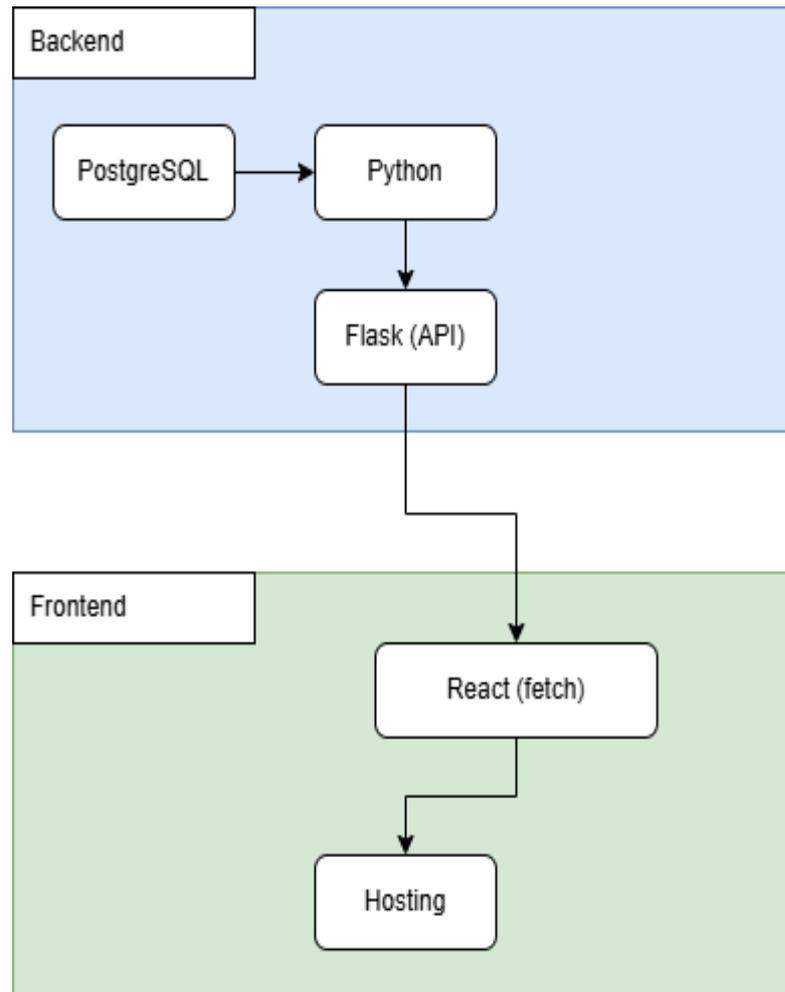
2. *Software*

- a) *Software Visual Studio Code* sebagai *text editor* dan *running script* program

- b) PostgreSQL sebagai basis data yang akan digunakan
- c) Postman sebagai media untuk menguji coba program
- d) GitHub sebagai *Platform* untuk menyimpan repositori kode secara *online*.

3.3 Capstone Project

Pada *project* yang dikerjakan terdapat beberapa bagian yang dikerjakan secara Bersama-sama yaitu membuat *Software Design Document* (SDD) dan *Software Requirement Specification* (SRS), dan bagian yang dikerjakan terpisah yaitu *Backend* dan *frontend*. Tujuan dari penulisan *Software Design Document* ini adalah untuk memberikan penjelasan mengenai hasil proses perancangan, termasuk perbaikan hasil rancangan tersebut untuk merepresentasikan rancangan pembuatan sistem ERP baik berupa gambaran umum maupun penjelasan detil dan menyeluruh. *Software Design Document* ini akan digunakan sebagai acuan informasi serta bahan analisis untuk perbaikan atau pengembangan sistem lebih lanjut. SRS memberikan definisi yang jelas dan rinci mengenai fungsionalitas sistem, berfungsi sebagai pedoman bagi tim pengembangan, memfasilitasi komunikasi antara pemangku kepentingan, menyediakan dasar untuk pengujian dan validasi, dan membantu manajemen perubahan. Dengan adanya SRS, *project* ini diharapkan lebih terarah dan sukses mencapai tujuannya. Pengembangan *website* ini dibagi menjadi dua bagian utama: pengembangan *Backend* dan *frontend*. Peneliti berfokus pada pengembangan *Backend*, sementara anggota tim lainnya bertanggung jawab atas pengembangan *frontend*. *Backend* dikembangkan menggunakan Python sebagai bahasa pemrograman utama. Python dipilih karena fleksibilitasnya, ekosistemnya yang kaya, serta dukungannya terhadap berbagai *library* dan *framework* yang memudahkan pengembangan aplikasi web. Untuk penyimpanan data, proyek ini menggunakan PostgreSQL, sebuah sistem manajemen basis data relasional yang terkenal dengan keandalan, kekayaan fitur, dan skalabilitasnya yang tinggi. PostgreSQL mendukung transaksi ACID (*Atomicity, Consistency, Isolation, Durability*) dan memiliki kemampuan untuk menangani beban kerja yang besar dan kompleks. Adapun *capstone project* dapat dilihat pada Gambar 3.1.



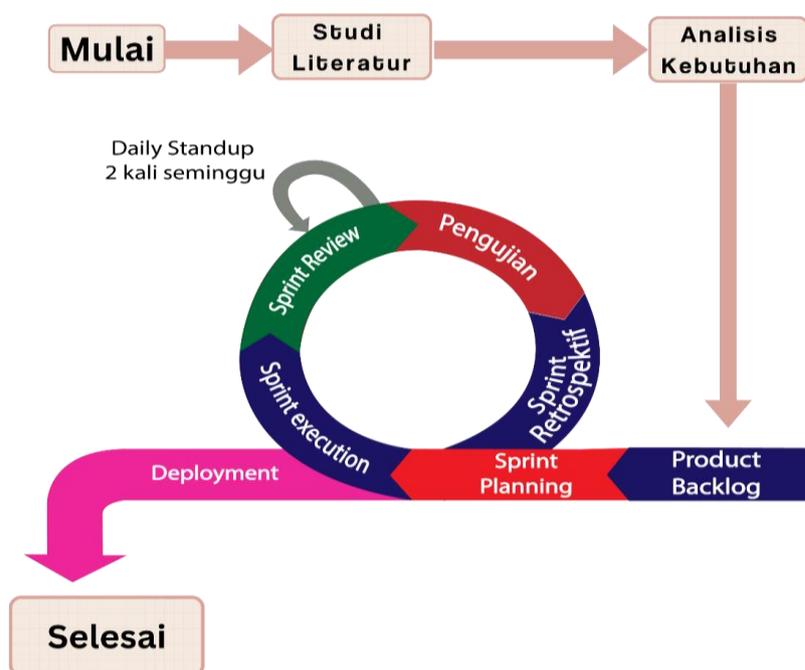
Gambar 3.1 Diagram Keseluruhan Pengembangan *Website*

Dalam membangun *Application Programming Interface* (API), digunakan Flask, sebuah *micro-framework* untuk Python yang memudahkan pembuatan aplikasi web dan API. Flask dipilih karena kesederhanaannya, kemudahan penggunaannya, serta kemampuannya untuk menangani berbagai permintaan HTTP dengan efisien. Flask juga memiliki dukungan komunitas yang luas dan banyak ekstensi yang bisa digunakan untuk memperkaya fungsionalitas aplikasi. Flask API yang dibangun akan menyediakan berbagai *endpoint* yang memungkinkan *frontend* untuk berinteraksi dengan *Backend* secara *seamless*. *Endpoint* ini mencakup operasi CRUD (*Create, Read, Update, Delete*) yang dibutuhkan oleh *website*, serta mekanisme otentikasi dan otorisasi untuk memastikan keamanan data pengguna. Dengan menggunakan PostgreSQL, proyek ini diharapkan dapat mengelola data

dengan lebih terstruktur dan mendukung berbagai jenis operasi yang kompleks, menjadikan aplikasi lebih handal dan efisien.

3.4 Metode Penelitian

Metode penelitian yang digunakan adalah metode *scrum*. Metode *scrum* adalah kerangka kerja manajemen proyek yang membantu tim untuk bekerja secara kolaboratif dan iteratif dalam mengembangkan produk atau menyelesaikan proyek. Adapun diagram alir dari penelitian ini dapat dilihat pada Gambar 3.2.



Gambar 3.2 Diagram Alir Penelitian

Metode Scrum dipilih karena dapat meningkatkan efisiensi dan produktivitas tim melalui pembagian tugas yang jelas. Setiap anggota tim, seperti backend dan frontend, fokus pada bidang keahliannya masing-masing. Dengan penerapan sprint secara berkala, tim dapat memantau progres secara rutin dan menyesuaikan prioritas sesuai kebutuhan. Metode ini sangat sesuai untuk tim kecil karena mendukung kolaborasi yang efektif serta memungkinkan adaptasi yang cepat terhadap perubahan.

3.4.1 Studi Literatur

Pada Tahapan ini, Peneliti mencari dan memahami referensi yang berkaitan dengan tujuan agar dapat memahami penelitian yang akan dilakukan. Studi literatur dilakukan dengan cara mencari penelitian terdahulu yang berkaitan dengan topik ini dengan tujuan untuk menemukan *state of the art* dari penelitian ini.

3.4.2 Analisis Kebutuhan

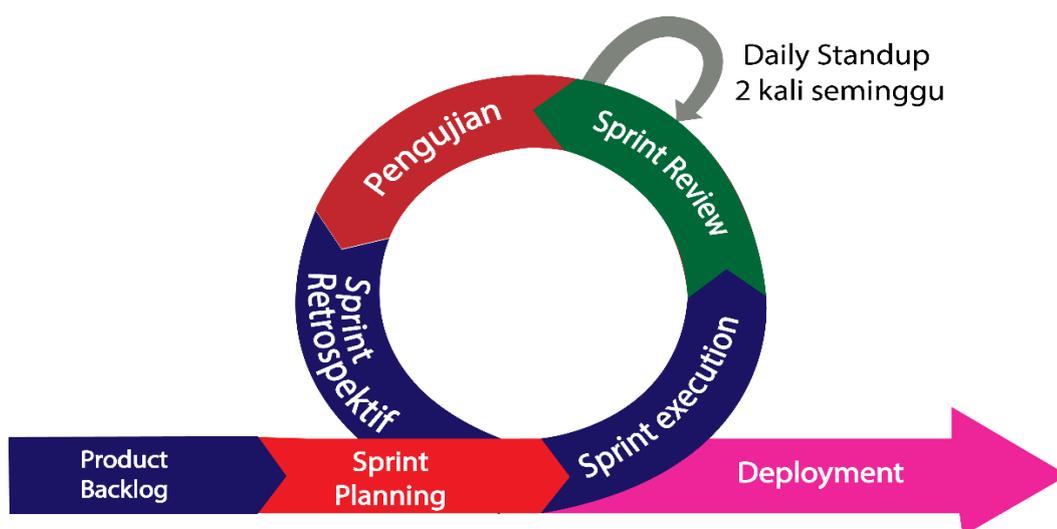
Peneliti mengidentifikasi kebutuhan utama dari perangkat yang akan dibuat serta menetapkan tujuan spesifik dari penelitian ini. Analisis kebutuhan dalam pembuatan sistem ERP untuk industri kelapa sawit melibatkan identifikasi dan pemahaman mendalam mengenai berbagai aspek operasional yang kritis bagi perusahaan di sektor ini. Sistem ERP mencakup modul-modul utama seperti manajemen perkebunan, produksi, logistik, dan distribusi untuk mengoptimalkan efisiensi dan produktivitas. Pada modul manajemen perkebunan, sistem harus mampu mengelola informasi mengenai lokasi, ukuran, dan status lahan, serta aktivitas penanaman, dan pemanenan. Dalam hal produksi, sistem ERP harus menyediakan fasilitas untuk mengelola pabrik pengolahan, mencatat proses pengolahan buah sawit menjadi produk akhir seperti minyak sawit mentah (CPO), dan memastikan manajemen kualitas pada setiap tahap produksi. Modul logistik dan distribusi harus mampu mengatur *transportasi* dari petani ke pabrik, memastikan pengiriman tepat waktu dan efisien. Integrasi yang mulus antara berbagai modul ini sangat penting untuk menciptakan sistem ERP yang mampu mendukung pertumbuhan dan berkelanjutan bisnis kelapa sawit.

3.4.3 Proses Perancangan Sistem

Proses perancangan sistem merupakan tahapan awal dalam membuat suatu penelitian. Pada penelitian ini digunakan metode *scrum* sebagai metode perancangan pembuatan *website*. Metode penelitian *Scrum* untuk pengembangan *Backend website* ERP sawit dapat diuraikan sebagai berikut. Pengembangan *Backend* dimulai dengan mendefinisikan *product backlog*, yaitu daftar semua fitur dan tugas *Backend* yang perlu diselesaikan, seperti setup lingkungan Flask, desain

dan implementasi *database*, sistem autentikasi pengguna, operasi CRUD (*Create, Read, Update, Delete*), validasi data, dokumentasi API, pengelolaan kesalahan, pengujian, dan *deployment*.

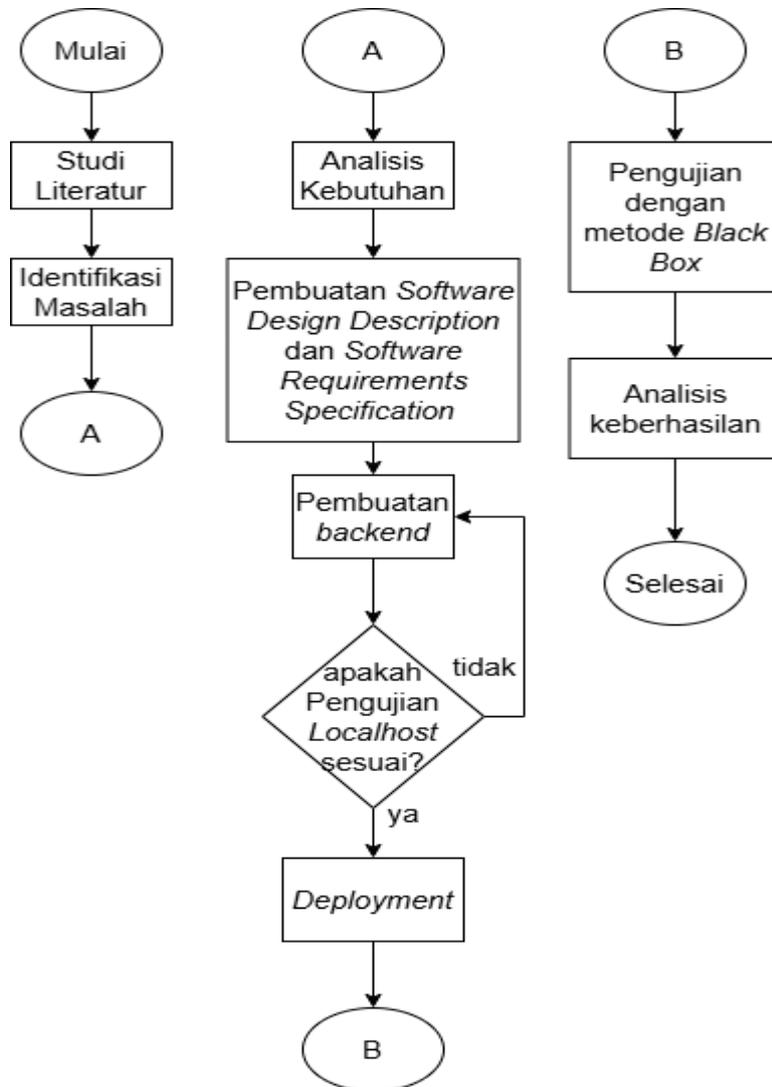
Proses perancangan Sistem dengan menggunakan *scrum method* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Perancangan *Scrum Method*

Selanjutnya, dalam setiap *Sprint planning*, tim memilih tugas-tugas prioritas dari *product backlog* untuk dimasukkan ke dalam *Sprint backlog*, memastikan tugas-tugas ini dapat diselesaikan dalam satu *Sprint*. *Sprint execution* adalah fase dimana pengembang *Backend* bekerja pada tugas-tugas yang telah dipilih, berpartisipasi dalam *daily standups*, yaitu pertemuan 2x seminggu untuk melaporkan kemajuan dan hambatan, serta berkolaborasi erat dengan pengembang *frontend* untuk memastikan integrasi yang mulus. Pada akhir setiap *Sprint*, hasil pekerjaan dipresentasikan dalam *Sprint review* untuk mendapatkan *feedback*. Setelah itu, dilakukan *Sprint retrospective* untuk merefleksikan dan mengidentifikasi area perbaikan dalam proses pengembangan. *Deployment* adalah proses penyebaran aplikasi ke *server* produksi setelah semua tugas dalam *Sprint* selesai dan diuji. Dengan pendekatan ini, pengembangan *Backend website* ERP sawit menggunakan metode *Scrum* memastikan bahwa semua aspek kritis seperti fungsionalitas, integrasi, kinerja, keamanan, ketersediaan, dan integrasi *frontend* dapat dikelola dan

dioptimalkan secara efektif, menghasilkan sistem yang andal dan berkualitas tinggi. *Flowchart* perancangan sistem dapat dilihat pada Gambar 3.4



Gambar 3.4 *Flowchart* Perancangan sistem

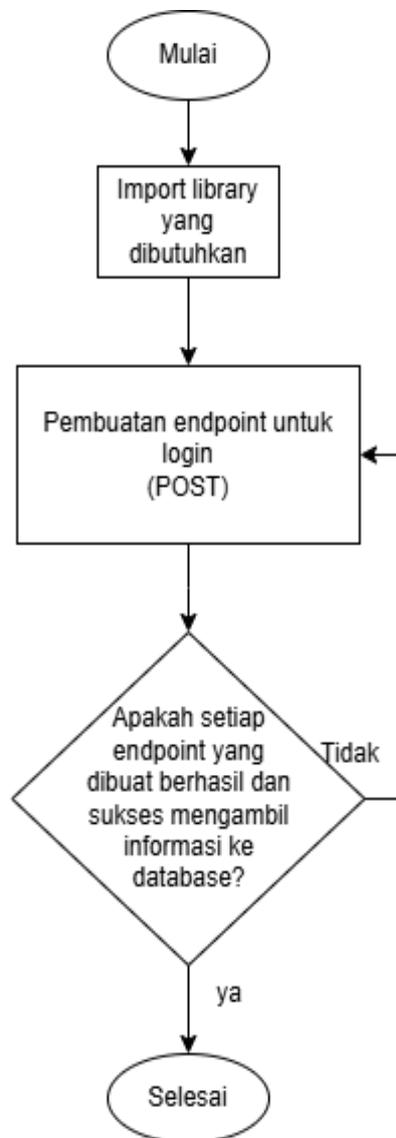
3.5 Sistem *Enterprise Resource Planning* (ERP)

Sistem *Enterprise Resource Planning* (ERP) yang dirancang bertujuan untuk mengintegrasikan seluruh proses bisnis dalam suatu organisasi ke dalam satu platform yang terpusat. Dengan menggunakan sistem ERP, data dan informasi dari berbagai modul dapat diproses secara otomatis dan efisien, sehingga memudahkan pengambilan keputusan yang lebih cepat dan tepat. Implementasi ERP memungkinkan perusahaan sawit untuk meningkatkan efisiensi operasional,

3.6 Pembuatan *Backend*

3.6.1 *Login*

Tahap ini bertujuan untuk membangun sistem otentikasi dan manajemen pengguna yang menjadi inti dari keamanan aplikasi. *Backend* modul *login* akan menangani proses *login*, dan pengelolaan sesi. Modul ini juga mencakup pengaturan peran (seperti petani, pabrik, logistik, kebun, dan superadmin) memberikan hak akses berbeda kepada pengguna. Diagram Alir perancangan modul *login* dapat dilihat pada Gambar 3.6.



Gambar 3.6 Diagram Alir Perancangan Modul *Login*

3.6.2 Perancangan Modul Super Admin

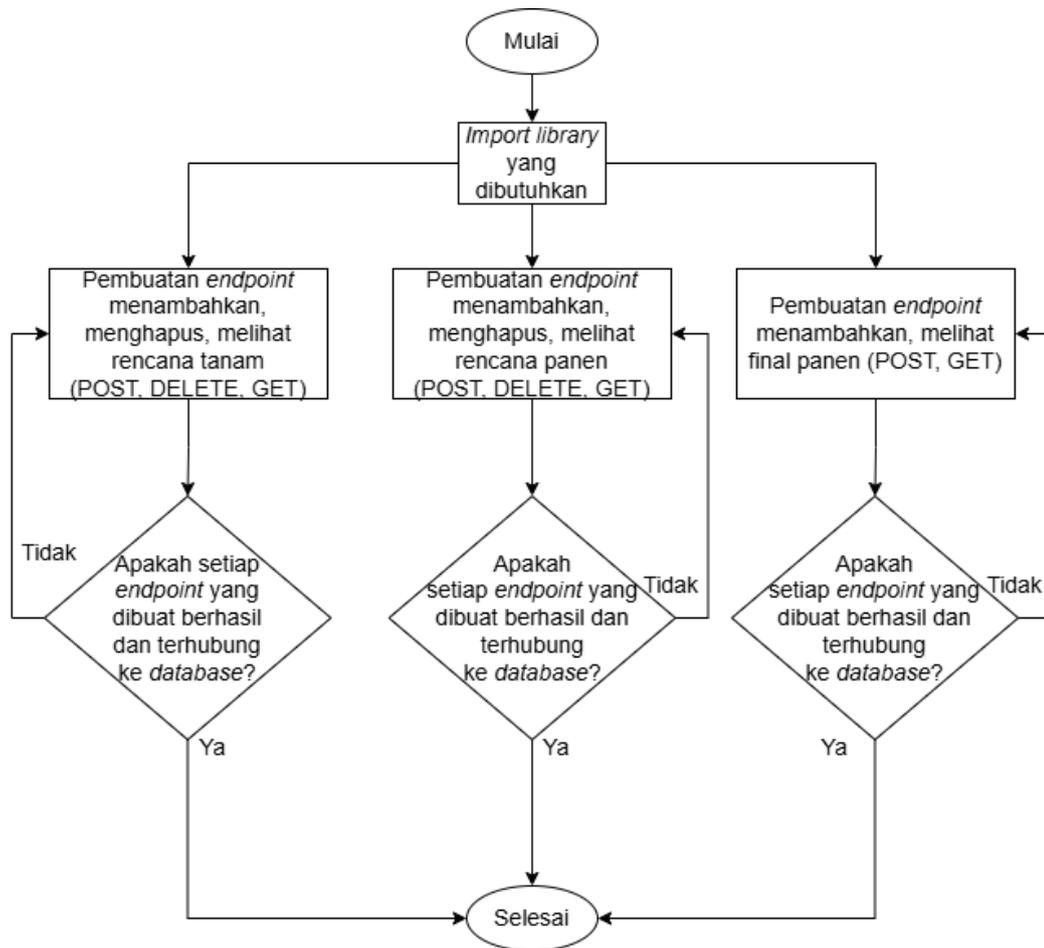
Modul "superadmin" menyediakan kontrol penuh atas seluruh sistem. *Backend* ini dirancang untuk memungkinkan superadmin mengelola pengguna, mengatur peran dan izin, serta memantau aktivitas dalam aplikasi. Modul ini juga memungkinkan superadmin untuk melakukan administrasi tingkat tinggi, seperti manajemen *database*, pengaturan konfigurasi sistem, dan pengawasan operasional keseluruhan. Superadmin memiliki akses penuh ke semua modul dan dapat mengelola integrasi antar modul untuk memastikan keseluruhan sistem berfungsi dengan baik. Diagram Alir perancangan modul super admin dapat dilihat pada Gambar 3.7.



Gambar 3.7 Diagram Alir Modul Super Admin

3.6.3 Perancangan Modul Petani

Modul "petani" menyediakan fungsionalitas untuk mengelola aktivitas pertanian, seperti pencatatan hasil panen, pengelolaan lahan, dan interaksi dengan pabrik atau logistik. *Backend* ini dirancang untuk memungkinkan petani mengelola data mereka sendiri dan berinteraksi dengan pengguna lain dalam sistem, memastikan koordinasi yang efektif. Diagram alir perancangan modul petani dapat dilihat pada Gambar 3.8.

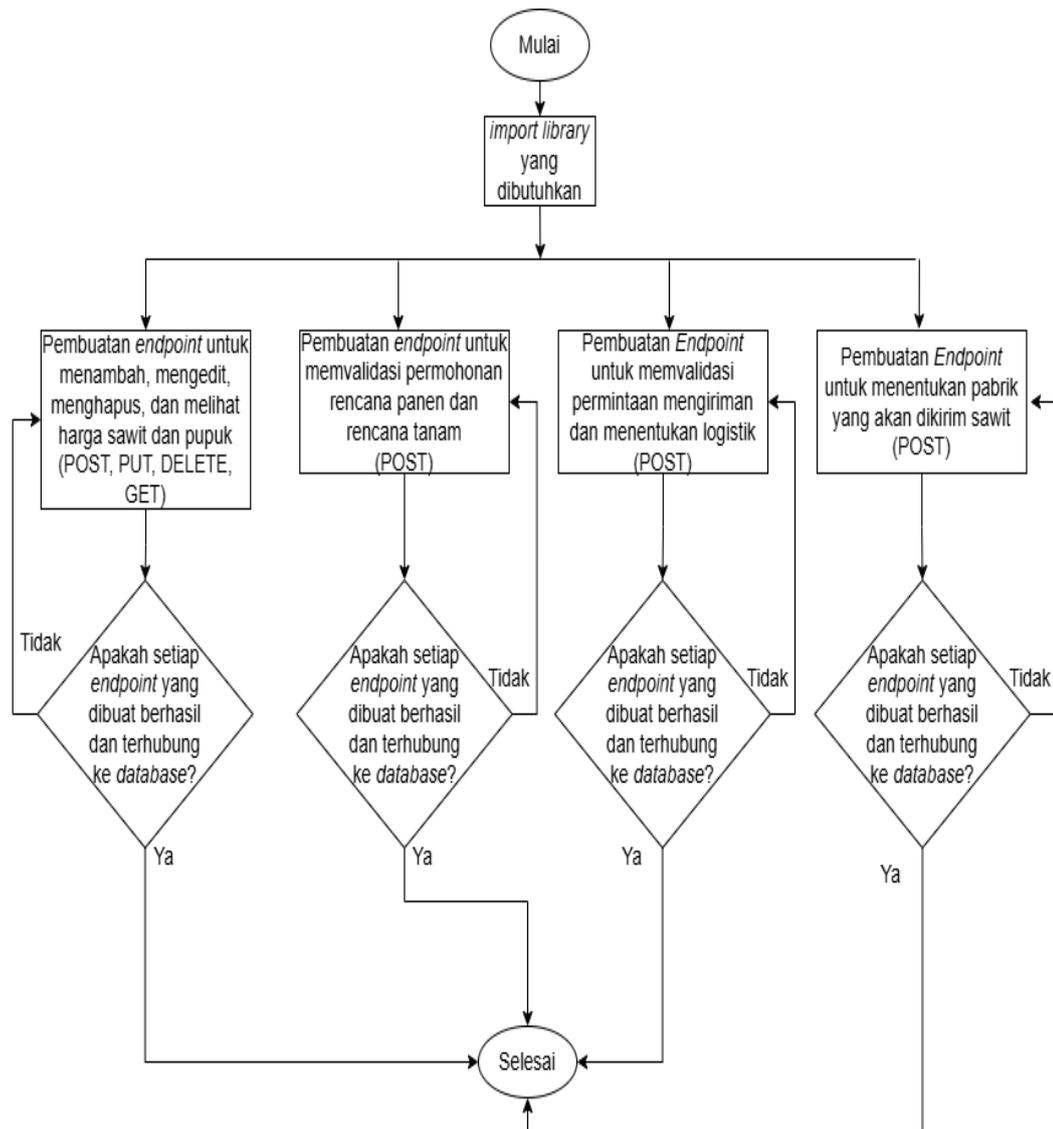


Gambar 3.8 Diagram Alir Modul Petani

3.6.4 Proses Perancangan Modul Kebun

Modul kebun pada perusahaan, pemilik, atau kantor yang memiliki kendali penuh atas beberapa aspek penting dalam rantai produksi. *Backend* ini memungkinkan kebun untuk mengelola informasi terkait harga dan jenis sawit, memvalidasi

rencana tanam dan panen, menentukan pabrik mana yang akan memproses hasil panen, dan memutuskan pengiriman. Modul ini juga berfungsi sebagai penghubung utama antara petani, pabrik, dan logistik, memastikan bahwa semua proses berjalan sesuai dengan rencana yang telah ditentukan. Diagram Alir perancangan modul kebun dapat dilihat pada Gambar 3.9.

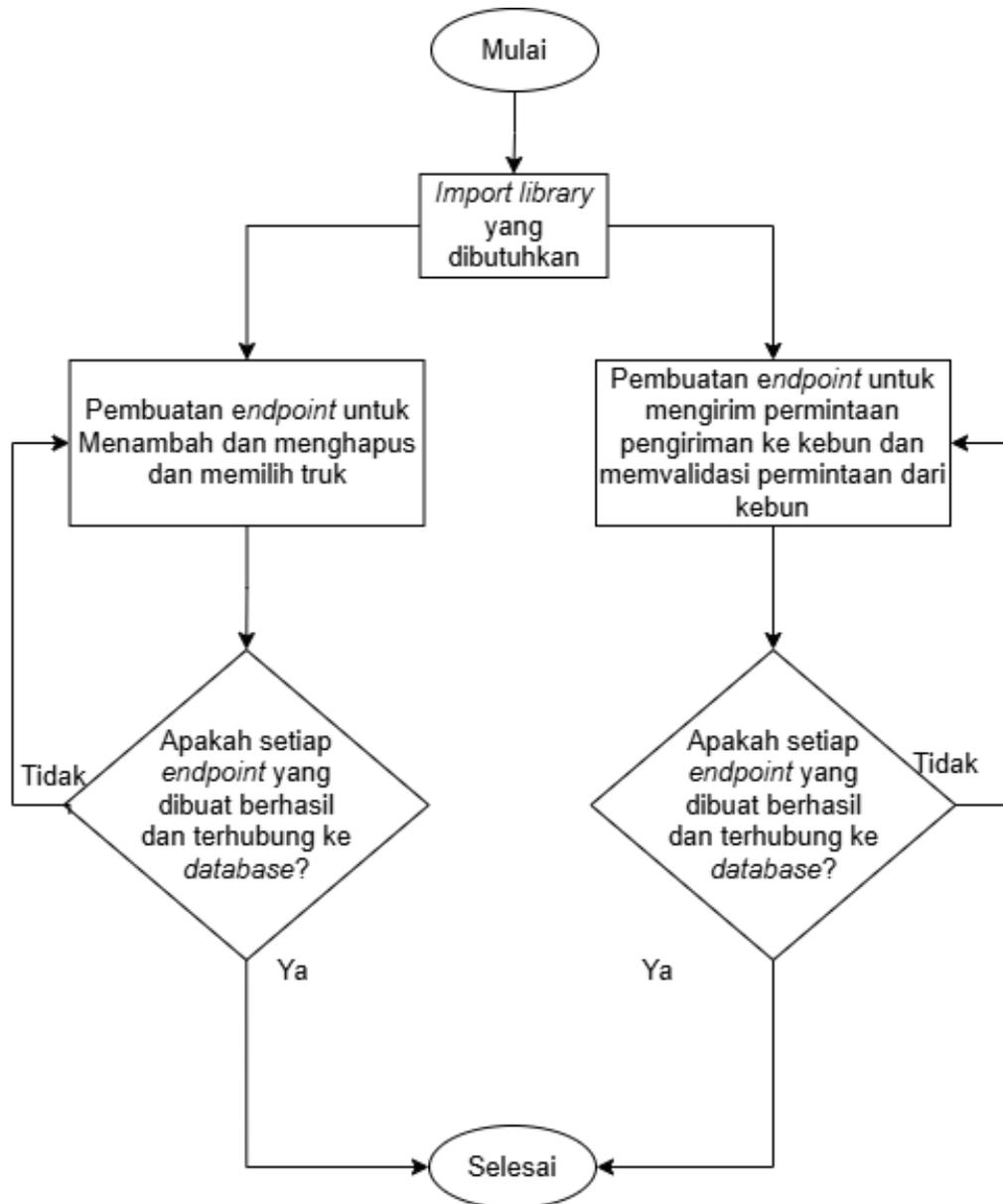


Gambar 3.9 Diagram Alir Modul Kebun

3.6.5 Perancangan Modul Logistik

Modul "logistik" mengelola proses pengiriman dan distribusi hasil panen serta produk olahan. *Backend* ini dirancang untuk melacak pergerakan barang dari kebun

atau pabrik ke tujuan akhir, seperti pasar atau konsumen. Modul ini juga mengatur penjadwalan pengiriman, pelacakan status, dan berkolaborasi dengan pabrik dan kebun untuk memastikan pengiriman yang tepat waktu dan efisien. Diagram alir perancangan modul logistik dapat dilihat pada Gambar 3.10.

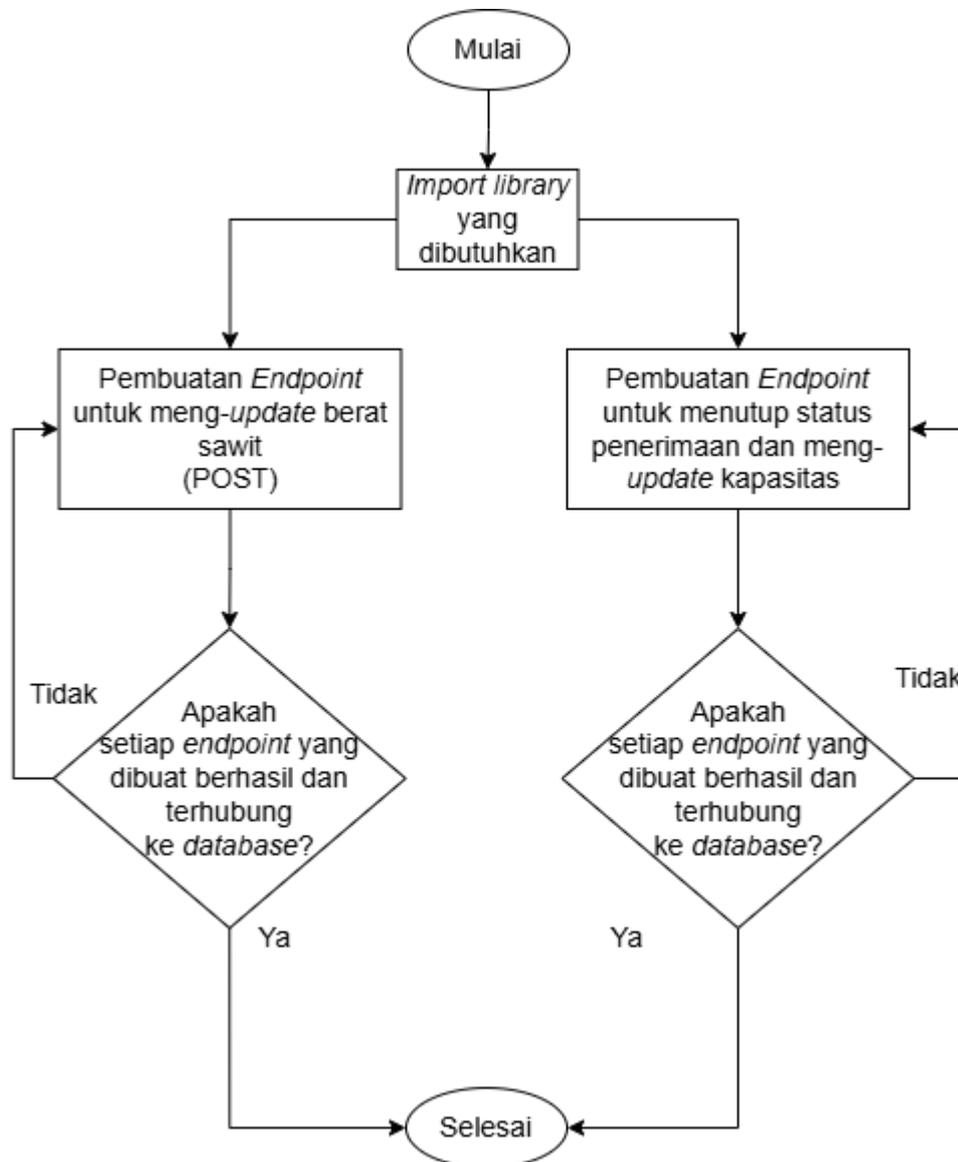


Gambar 3.10 Diagram Alir Modul Logistik

3.6.6 Perancangan Modul Pabrik

Modul "pabrik" mengelola proses pengolahan hasil pertanian. *Backend* ini menangani data penerimaan barang dari petani, status pengolahan, dan pengiriman

produk akhir. Modul ini juga menghubungkan petani dengan logistik, memastikan Alir produksi dan distribusi berjalan dengan lancar dan efisien. Diagram Alir perancangan modul pabrik dapat dilihat pada Gambar 3.11.

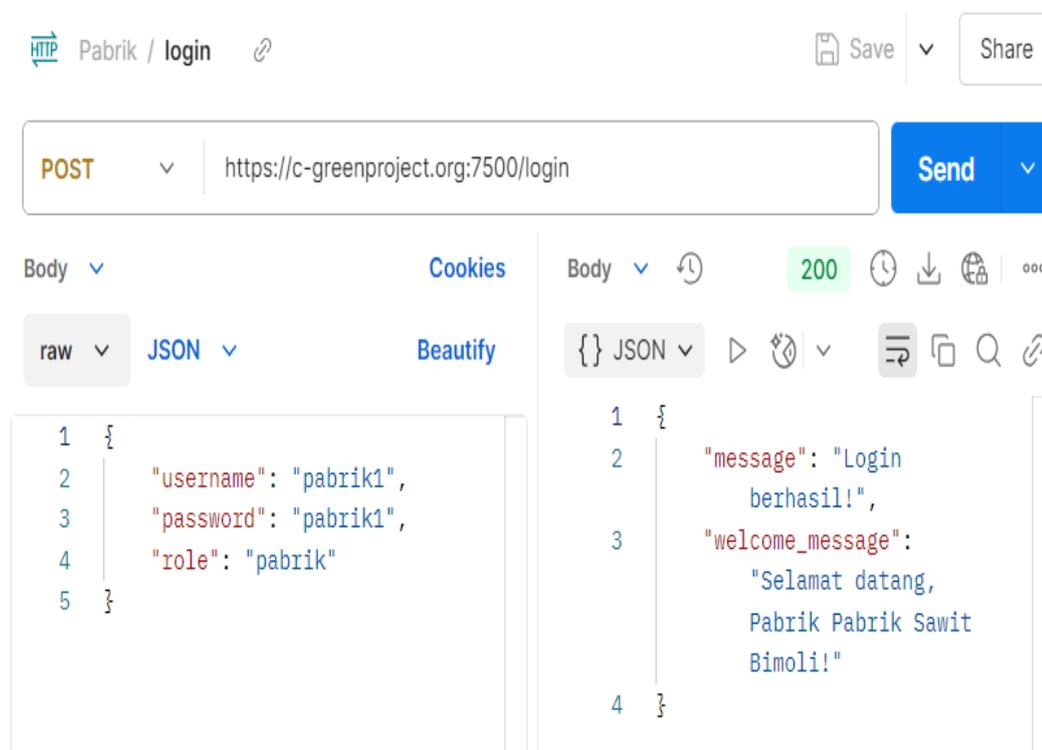


Gambar 3.11 Diagram Alir Modul Pabrik

3.6.7 Proses Pengujian

Proses pengujian merupakan tahapan penting dalam menguji *Backend* yang telah dibuat sebelumnya untuk memastikan fungsionalitas dan stabilitasnya. Dalam penelitian ini, proses pengujian dilakukan untuk mengidentifikasi dan memperbaiki potensi bug atau kesalahan. Pengujian ini melibatkan berbagai metode, termasuk

uji fungsionalitas, uji *database*, dan uji integrasi antar komponen *backend* untuk memastikan bahwa setiap elemen *backend* berfungsi dengan baik secara individual. Pengujian dilakukan menggunakan *software Postman* yang merupakan alat untuk menguji API dengan mengirimkan *request* ke setiap *endpoint* yang telah disediakan. Postman memungkinkan pengujian *endpoint* secara terpisah untuk memastikan *response* yang tepat sesuai dengan spesifikasi yang telah ditentukan. Salah satu contoh pengujian *endpoint* dapat dilihat pada gambar 3.12.



Gambar 3.12 Proses Pengujian *Endpoint*

Gambar 3.12 menggambarkan salah satu langkah dalam proses pengujian local *backend* menggunakan *software Postman*. Pengujian ini dilakukan untuk memastikan *endpoint* API dapat meresponse sesuai dengan parameter yang dimasukkan. Dalam pengujian ini, *endpoint* API *login* diuji dengan metode POST, dimana data pengguna dikirim dalam format JSON melalui *Body request*. Data tersebut mencakup tiga atribut utama: *Username*, *password*, dan *role*. Setelah permintaan dikirim, *server* meresponse dengan kode status 200 OK, yang menunjukkan bahwa permintaan berhasil diproses. Hal ini menunjukkan bahwa

API *login* bekerja dengan baik, memverifikasi kredensial pengguna, dan mengembalikan *response* sesuai dengan peran pengguna.

Tujuan dari pengujian ini adalah untuk memvalidasi apakah sistem *backend* dapat memproses autentikasi pengguna berdasarkan data yang diberikan. Dengan menggunakan Postman, setiap elemen API diuji secara terpisah, memastikan bahwa API dapat memberikan *response* sesuai dengan spesifikasi yang telah ditentukan sebelumnya.

3.7 Struktur *Database*

Dalam pembuatan ERP ini, peneliti akan membuat data yang banyak sehingga memerlukan *database* yang dapat menampung data dalam skala besar seperti *database* PostgreSQL. PostgreSQL adalah sebuah sistem manajemen basis data relasional (RDBMS) yang bersifat open-source dan sangat cocok untuk aplikasi yang membutuhkan transaksi tingkat tinggi dan integritas data yang kuat. Data dalam PostgreSQL disimpan dalam format tabel yang terstruktur, dengan dukungan penuh untuk *query* SQL yang kompleks dan kemampuan skalabilitas yang tinggi.

3.7.1 Data Primer

Data primer adalah data yang dikumpulkan oleh setiap stakeholder yang terlibat dalam sistem untuk tujuan pembuatan akun dan interaksi lebih lanjut dengan aplikasi. Data ini sangat penting untuk memvalidasi identitas pengguna dan memastikan kelancaran proses pendaftaran serta pengelolaan akun dalam sistem. Data primer dapat dilihat pada Tabel 3.2.

Tabel 3.2 Data Primer

Semua stakeholder	
<i>Username</i>	<i>String</i>
<i>Password</i>	<i>String</i>
<i>Role</i>	<i>String</i>
<i>e-mail</i>	<i>String</i>

Tabel 3.2 Data Primer (lanjutan)

Petani	
Nama Petani	<i>String</i>
Alamat Petani	<i>String</i>
Nomor Telepon	<i>String</i>
Alamat Lahan	<i>text</i>
Luas Lahan	<i>Float</i>
Status Sertifikasi Lahan	<i>String</i>
Kebun	
Nama Kantor Kebun	<i>String</i>
Alamat Kantor Kebun	<i>String</i>
Nomor Telepon	<i>String</i>
Logistik	
Nama Perusahaan Logistik	<i>String</i>
Alamat Perusahaan	<i>String</i>
Nomor Telepon	<i>String</i>
Pabrik	
Nama Pabrik	<i>string</i>
Alamat Pabrik	<i>string</i>
Nomor Telepon	<i>string</i>
Kapasitas Penyimpanan	<i>float</i>
Kapasitas Produksi	<i>float</i>

3.7.2 Data Transaksional

Data transaksional adalah data yang terkait dengan aktivitas transaksi yang dilakukan oleh pengguna dalam sistem. Data ini mencakup informasi mengenai setiap tindakan atau interaksi yang terjadi dalam aplikasi, seperti pembelian, pembayaran, pengiriman, atau transaksi lainnya. Data transaksional sangat penting untuk memantau status dan riwayat transaksi, serta untuk analisis dan pelaporan

yang diperlukan oleh super admin atau pihak yang berwenang. Data transaksional dapat dilihat pada Tabel 3.3.

Tabel 3.3 Data Transaksional

Kebun	
Jenis Sawit	<i>String</i>
Jenis Pupuk	<i>String</i>
Harga Sawit	<i>Float</i>
Harga Pupuk	<i>Float</i>
Petani	
Nama Petani	<i>String</i>
Nomor Telepon	<i>String</i>
Tanggal Rencana Tanam	<i>Date</i>
Tanggal Rencana Panen	<i>Date</i>
Berat sawit awal dan akhir	<i>Float</i>
Total Pembayaran	<i>Float</i>
Jenis Sawit yang ditanam/panen	<i>String</i>
Jumlah panen	<i>int</i>
Jumlah tanam	<i>int</i>
Alamat lahan	<i>String</i>
Sertifikasi Lahan	<i>String</i>
Logistik	
Nama Logistik	<i>String</i>
Nomor Telepon	<i>String</i>
Nama <i>Driver</i>	<i>String</i>
Plat Nomor Truk	<i>String</i>
Status Pengiriman (diterima/dikirim)	<i>String</i>
Tanggal Pengiriman	<i>Datetime</i>
Tanggal Penerimaan	<i>Datetime</i>

Tabel 3.3 Data Transaksional (Lanjutan)

Pabrik	
Nama Pabrik	<i>String</i>
Nomor Telepon	<i>String</i>
Status Penerimaan Sawit (buka/tutup)	<i>String</i>
Hasil Produksi sawit	<i>Float</i>
Waktu Produksi Sawit	<i>Date time</i>
Sumber Produksi Sawit	<i>String</i>
Ketersediaan Kapasitas pabrik	<i>Float</i>

3.8 Parameter Keberhasilan

Untuk mengevaluasi keberhasilan, pengujian dilakukan menggunakan Postman dengan menguji beberapa parameter kunci yaitu *login*, fitur *traceability*, *update* pengiriman, lacak pengiriman, dan transparansi harga. Berikut adalah ringkasan capaian dari setiap parameter yang diuji:

1. *Login*

Pengujian *login* dikatakan berhasil mengelola kredensial valid (*response 200*) dan menolak kredensial yang salah (*response 401 Unauthorized*).

2. Fitur *traceability*

Sistem mampu memberikan kemampuan pelacakan yang komprehensif terhadap alur sawit yang diproses, mulai dari asal (bibit) hingga tujuan akhir (CPO dan limbah). Fitur *traceability* memungkinkan pengguna untuk memantau pergerakan dan meminimalisir kesalahan.

3. *Update* Pengiriman

Sistem mampu melakukan pembaruan otomatis terkait status pengiriman barang secara berkala. Hal ini mencakup pemberitahuan mengenai perubahan

status pengiriman, seperti pengiriman yang telah berhasil atau dalam pengiriman

4. Lacak Pengiriman

Sistem mampu menyediakan fitur pelacakan pengiriman yang memungkinkan pengguna untuk mengetahui posisi terkini barang dalam perjalanan.

5. Transparansi Harga

Sistem mampu menampilkan harga yang di-*input* oleh perusahaan berdasarkan kategori atau grade produk secara transparan. Pengguna dapat dengan mudah mengakses informasi terkait harga yang ditawarkan, termasuk perincian biaya terkait, untuk memastikan adanya kejelasan dalam transaksi dan mengurangi potensi kesalahpahaman atau kecurangan harga di pasar.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

1. Implementasi python dan PostgreSQL pada sistem ini berfungsi dengan baik pada entitas yang ada di dalam sistem. Dengan *database* PostgreSQL, data dapat disimpan dan diakses secara akurat, sementara Python memungkinkan implementasi logika bisnis yang fleksibel dan skalabel. Hal ini akan meningkatkan transparansi, mengurangi kesalahan manual, dan mempercepat pengambilan keputusan berbasis data. Sistem ini dapat mengoptimalkan rantai pasokan agar tetap konsisten.
2. Model manajemen rantai pasok perusahaan kelapa sawit dapat ditransformasikan dari metode tradisional/konvensional menjadi model blockchain melalui pembuatan sistem *Enterprise Resource Planning* (ERP) menggunakan python. Sistem ini berfungsi sebagai media untuk menjamin ketelusuran (*traceability*) produk *Crude Palm Oil* (CPO) dan Limbah cair/padat yang dihasilkan oleh perusahaan minyak kelapa sawit. Dengan konsep *blockchain*, seluruh data dasar dan data transaksi terekam/tercatat di sistem dan dapat diakses oleh setiap *stakeholder*, sehingga *visibility*, transparansi, dan akuntabilitas proses bisnis perusahaan dapat terjamin.

5.2 Saran

Aplikasi yang dibangun melalui riset ini masih dapat terus dikembangkan dan ditingkatkan fungsi-fungsinya menjadi sebuah aplikasi ERP *dedicated*.

Pengembangan platform ini dapat dilakukan adalah menambahkan stakeholder bank untuk integrasi sistem pembayaran atau payment gateway, yang akan mempermudah dan mengamankan transaksi antara pemasok, logistik, dan produsen. Selain itu, perlu dipertimbangkan sistem perhitungan harga yang lebih fleksibel, terutama jika dalam satu truk terdapat lebih dari satu jenis sawit. Sistem ini harus mampu menghitung harga secara akurat berdasarkan berat masing-masing jenis sawit, untuk memastikan transparansi dan keadilan dalam transaksi.

DAFTAR PUSTAKA

- [1] Y. Fauzi, Y. E. Widyastuti, I. Satyawibawa and R. H. Paeru, *Kelapa Sawit*, Jakarta: Penebar Swadaya, 2012.
- [2] D. Nurhayati, H. Noviarita and M. I. Fasa, "*Cooperation Between Indonesia And Malaysia In Palm Hydrilization*," *Management Studies and Entrepreneurship Journal*, vol. 4, no. 6, pp. 9136-9141, 2023.
- [3] J. M. Fatoni and A. Nugroho, "Implementasi *Open Source Enterprise Resource Planning*," *Jurnal Teknik Informatika dan Sistem Informasi* , vol. 10, no. 2, pp. 666-676, 2023.
- [4] B. Abdillah, R. F. N.A, H. Yuliantika, I. A. R and S. R. J. Akbar, "Peran ERP dalam Meningkatkan, Transparansi dan Akuntabilitas Bisnis dan Keuangan," *Jurnal Pendidikan Tambusai*, vol. 8, no. 2, pp. 26616-26621, 2024.
- [5] Suminten, "Implementasi *Enterprise Resource Planning (Erp)* Pada Usaha Pithik Sambel Ndeso Berbasis Odoo," *Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, vol. 6, no. 1, pp. 60-68, 2019.
- [6] T. Hidayat and Rasyidah, "Perancangan dan Pembuatan Web ERP untuk PT Cipta Tekno Mandiri Menggunakan Framework Yii 2," *Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 2, no. 3, pp. 85-89, 2021.
- [7] T. Fatmawati, R. Kramanandita and R. Miza, "Rancangan Implementasi *Enterprise Resource Planning (ERP)* pada Sistem Pengelolaan Sales Order

- PT Jaya Mandiri Indotech," *Jurnal Teknologi dan Manajemen*, vol. 20, no. 1, pp. 33-44, 2022.
- [8] Liyani and E. Hadinata, "Perancangan Sistem *Enterprise Resource Planning* (ERP) *Inventory* di Toko Bangunan Sederhana Mandiri Jaya Sejahtera," *Indonesian Journal Computer Science*, vol. 2, no. 2, pp. 104-110, 2023.
- [9] R. Hanifudin, P. Rokhmayati, M. F. N. Rizqi and L. Fitriana, "Rancang Bangun Sistem *Enterprise Resource Planning* (ERP) Berbasis Web pada PT Sainsgo Karya Indonesia Menggunakan Metode Scrum," *JournalL Syntax Idea*, vol. 6, no. 6, pp. 2857-2871, 2024.
- [10] P. A. Maurya, R. Jagtap, K. Jadhav, A. Patil and M. Tripathi, "*Python: The Most Advanced Programming Language for Computer Science Applications*," *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, vol. 3, no. 2, pp. 174-179, 2023.
- [11] A. C. Rompis and R. F. Aji, "*Performance Comparison of Node.js, PHP, and Python*," *Cogito Smart Journal*, vol. 4, no. 1, pp. 171-187, 2018.
- [12] R. E. Lubis and A. Widanarko, *Buku Pintar Kelapa Sawit*, Jakarta: PT. AgroMedia Pustaka, 2011.
- [13] C. I. Pratiwi, "Analisis Permintaan Minyak Sawit Indonesia Oleh Industri Minyak Goreng, Margarin, Sabun, Dan Fatty Acid," Institut Pertanian Bogor, Bogor, 2014.
- [14] "Dinas Perkebunan Provinsi Lampung," Dinas Perkebunan Provinsi Lampung, 2022.
[Online]. Available: <https://disbun.lampungprov.go.id/pages/tata-alur-budidaya-kelapa-sawit>. [Accessed 19 06 2024].
- [15] R. Pangestika and R. T. Dirgahayu, "Pengembangan Back-end Sistem Informasi Komunitas Pendar Foundation Yogyakarta," *Automata*, vol. 1, no. 2, 2020.

- [16] M. Singh, A. Verma, A. Parasher, N. Chauhan and G. Budhiraja, "Implementation of Database Using Python Flask Framework," *International Journal Of Engineering And Computer Science*, vol. 8, no. 12, pp. 24894-24899, 2019.
- [17] Y. Zanevych, "Flask Vs. Django Vs. Spring Boot: Navigating Framework Choices For Machine Learning Object Detection Projects.," *Collection of scientific papers «ΑΙΟΓΟΣ»*, pp. 311-318, 2024.
- [18] PostgreSQL, "About Page PostgreSQL," PostgreSQL, [Online]. Available: <https://www.postgresql.org/about/>. [Accessed 03 06 2025].
- [19] B. Setiono, T. B. Adji and D. Adhipta, "Komparasi Kinerja Basis Data Oracle Dan Postgresql Berdasarkan Estimasi *Cost* Dan *Execution Time* Serta TPC-C Benchmark," Universitas Gajah Mada, Yogyakarta, 2016.
- [20] Y. J. S. Nahak and H. D. Purnomo, "Perancangan Sistem Replikasi Dan Sistem Backup Database Postgresql Menggunakan Repmgr Dan Barman," *Jurnal Ilmiah Komputer*, vol. 19, no. 2, 2023.
- [21] Z. Xi, "The Usability of PostgreSQL," Kemi-Tornio University Of Applied Sciences Technology, Kemi and Tornio, 2010.
- [22] A. Hadiana and Y. Sudaryo, FRAMEWORK: *Enterprise Resource Planning*, Penerbit Andi, 2021.
- [23] P. Ramadhani¹, Nur'Illiyyien, A. Indrijawati and M. I. Ferdiansyah, "Studi Literatur Sistematis: Faktor Penentu Keberhasilan Penerapan Erp Dalam Organisasi," *Islamic Accounting and Finance Review*, vol. 4, no. 1, 2023.
- [24] W. Zayat and O. Senvar, "Framework Study for Agile Software Development Via Scrum and Kanban," *International Journal of Innovation and Technology Management*, vol. 17, no. 4, 2020.

- [25] A. R. R. N. Chandrika, "Postman for API Testing: A Comprehensive Guide for QA Testers," *Science Data and Learning Machine ,Intelligence Artificial of Journal*, vol. 1, no. 1, 2024.