

Implementasi Framework Django Dalam Sistem Penjualan Sparepart Motor Menggunakan Metode Agile

Implementation Of The Django Framework In The Motorcycle Spare Parts Sales System Using The Agile Method

Dodi Irawan^{1*}, Putra Edi Mujahid²

AFFILIATIONS

- [1] Akutansi Syariah,
Fakultas Ekonomi dan
Bisnis Islam, UIN
Sulthan Thaha Saifuddin
Jambi, Indonesia
- [2] Ekonomi Syariah,
Fakultas Ekonomi dan
Bisnis Islam, UIN
Sulthan Thaha Saifuddin
Jambi, Indonesia

*Correspondence:

dodiirawan@uinjambi.ac.id

Submitted: 2025-10-15

Revised: 2025-11-17

Accepted: 2025-12-30

Copyright © 2025 by
Author(s)

This is an open access article
under the terms and
conditions of the Creative
Commons 4.0 International
(CC-BY-SA)

ABSTRAK

Bengkel Kevin Motor merupakan usaha yang bergerak di bidang penjualan suku cadang dan layanan servis motor. Dalam menjalankan kegiatan operasionalnya, bengkel ini masih menggunakan sistem penjualan dan pencatatan servis secara manual, sehingga menimbulkan berbagai permasalahan seperti duplikasi data, kesulitan dalam pencarian informasi, serta keterlambatan dalam pembuatan laporan. Tujuan dari penelitian ini adalah menganalisis dan memahami permasalahan pada sistem penjualan yang berjalan di Bengkel Kevin Motor, serta merancang sistem informasi penjualan suku cadang dan servis motor berbasis web menggunakan bahasa pemrograman Python dengan framework Django. Metode pengembangan sistem yang digunakan adalah *Agile* dengan pendekatan pemodelan *Unified Modeling Language* (UML) yang meliputi diagram aktivitas dan diagram kelas. Hasil penelitian ini berupa sistem informasi yang mempermudah pencatatan transaksi, pengelolaan stok, serta pembuatan laporan sehingga dapat mengurangi kesalahan dan duplikasi data.

KATA KUNCI

Python, Django, Bengkel, Sparepart, Agile

ABSTRACT

Kevin Motor Workshop is a business engaged in the sale of spare parts and motorcycle service. In carrying out its operational activities, this workshop still uses a manual sales and service recording system, resulting in various problems such as data duplication, difficulty in searching for information, and delays in reporting. The purpose of this study is to analyze and understand the problems in the sales system running at Kevin Motor Workshop, as well as to design a web-based motorcycle spare parts and service sales information system using the Python programming language with the Django framework. The system development method used is Agile with a Unified Modeling Language (UML) modeling approach that includes activity diagrams and class diagrams. The results of this study are an information system that simplifies transaction recording, stock management, and report generation so as to reduce errors and data duplication.

KEYWORDS

Python, Django, Workshop, Spare Parts, Agile

PENDAHULUAN

Teknologi dapat memberikan nilai tambah dalam peningkatan dari kualitas informasi [1], terkhusus dalam transaksi penjualan pada dunia bengkel. Penggunaan sistem berbasis teknologi memungkinkan pencatatan transaksi penjualan sparepart dan jasa servis

dilakukan secara otomatis, akurat, tersimpan pada database dan dapat mengurangi kesalahan perhitungan dan meningkatkan efisiensi dalam pelayanan.

Kevin Motor bergerak di bidang usaha penjualan Sparepart dalam proses penjualan sparepart dan jasa service akan dicatat dalam bentuk pembukuan dan akan dicatat satu persatu sesuai dengan transaksi penjualan dan service per-harinya. Adapun permasalahan yang dihadapi yaitu Dalam proses penginputan data penjualan sering terjadi kesalahan yang menyebabkan data penjualan suku cadang menjadi kurang akurat. Selain itu, kegiatan pengecekan, pembaruan, dan pencarian stok suku cadang memerlukan waktu yang cukup lama karena belum tersedianya laporan stok yang menunjukkan suku cadang yang hampir habis atau telah habis. Akibatnya, ketika konsumen membutuhkan suku cadang tertentu, mereka kerap harus menunggu lebih lama karena pihak bengkel masih melakukan pengecekan secara manual dengan mendatangi gudang secara langsung [2].

Sistem transaksi yang masih dilakukan secara manual perlu dilakukan perubahan sesuai dengan teknologi yang saat ini sedang berkembang sehingga pendataan persediaan *sparepart* lebih cepat, akurat, dan efisien, karena komputer mampu mengolah, menyimpan, dan menampilkan data dalam jumlah besar dengan tingkat kesalahan yang lebih rendah. Maka dari itu untuk mengatasi kendala yang ada, penulis memberikan solusi melalui penyediaan sistem informasi yang dirancang untuk mengintegrasikan seluruh data pada bengkel secara komputerisasi.

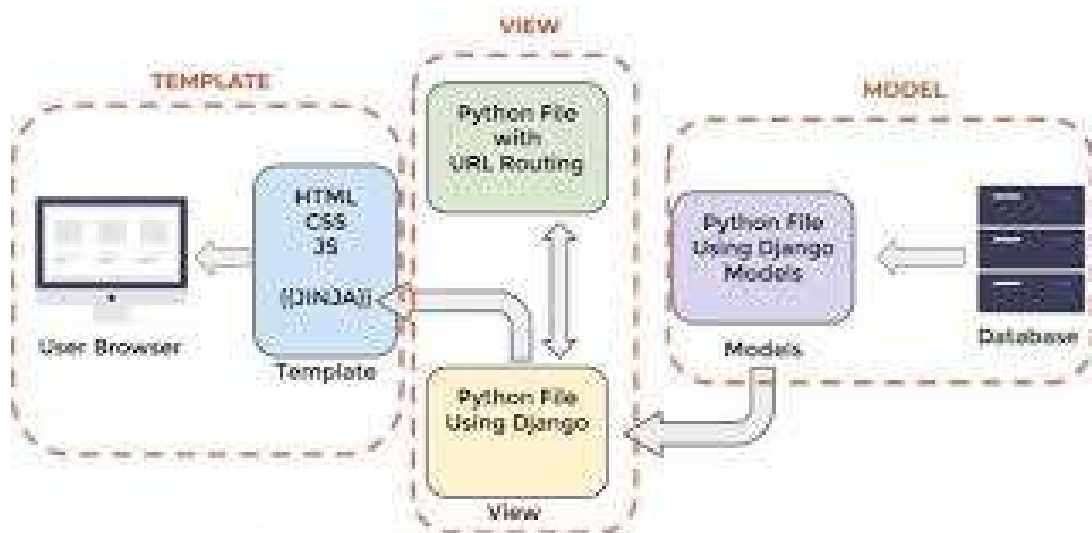
Implementasi sistem informasi berbasis *web* dengan framework Django bertujuan untuk mengoptimalkan pengelolaan data pada bengkel. Melalui sistem ini, proses seperti pendataan mekanik, pelanggan, riwayat transaksi, dan pemantauan stok sparepart dapat dilakukan dengan lebih terstruktur.

Tinjauan Pustaka

Python adalah bahasa pemrograman interpretatif yang dapat digunakan untuk berbagai tujuan, dan fokusnya pada keterbacaan kode membuat sintaks lebih mudah dipahami [3]. *Python Software Foundation* (PSF) adalah organisasi nirlaba yang secara resmi menjadi pemilik hak kekayaan intelektual Python sejak versi 2.1. Organisasi ini bertanggung jawab untuk mendukung pengembangan, distribusi, dan penggunaan Python secara global. Python telah menjadi salah satu bahasa pemrograman dengan pertumbuhan tercepat di dunia. Python adalah bahasa pemrograman interpretatif yang dapat digunakan untuk berbagai tujuan, dan fokusnya pada keterbacaan kode membuat sintaks lebih mudah dipahami [4].

Nginx merupakan platform *web server open-source* terkemuka yang banyak diimplementasikan oleh berbagai instansi maupun perusahaan sebagai infrastruktur utama dalam mengoperasikan situs dan aplikasi berbasis *web* [5].

Django merupakan kerangka kerja yang menggunakan arsitektur MVT (*Model-View-Template*) sebagai pola desain perangkat lunaknya. Pola ini berfungsi untuk memisahkan logika data, antarmuka pengguna, dan kendali aplikasi dalam proses pengembangan *web* [6]. Model View Template arsitektur django dapat dilihat pada gambar 1.



Gambar 1. Arsitektur *Model View Template* pada Django [7]

Penjelasan dari Gambar 1. Akan di jabarkan secara lebih rinci pada point-point yang dapat dilihat seperti berikut ini :

a) *Model*

Model berperan sebagai jembatan komunikasi dengan *database*. Komponen ini tidak hanya menyimpan informasi mengenai struktur objek dan keterkaitannya, tetapi juga menetapkan aturan validasi yang memastikan data yang diinputkan sesuai dengan tipe data yang dibutuhkan.

b) *View*

Komponen *View* berperan sebagai pengatur logika bisnis sekaligus pengelola permintaan (*request*) dari pengguna. *View* dapat digunakan untuk menarik data dari *Model* dan mentransformasikannya ke dalam format yang sesuai untuk disajikan pada tampilan pengguna.

c) *Template*

Tampilan akhir bagi pengguna dihasilkan melalui komponen *Template*. Melalui sintaks khusus yang terintegrasi dengan Django, data yang diproses dapat digabungkan secara baik dengan *markup* HTML, sehingga informasi yang ditampilkan bersifat fleksibel dan adaptif.

Kesenjangan Penelitian

Berdasarkan kajian literatur, sebagian besar penelitian terdahulu berfokus pada perancangan sistem informasi berbasis web, penggunaan framework tertentu, atau penerapan metode Agile secara terpisah. Namun, masih terbatas penelitian yang mengintegrasikan implementasi framework Django dengan arsitektur MVT secara rinci, penerapan metode Agile secara menyeluruh dari tahap concept hingga retired, serta pengujian fungsional sistem pada konteks bengkel sepeda motor skala kecil dan menengah.

Selain itu, penelitian sebelumnya umumnya menekankan pada hasil perancangan sistem tanpa memberikan deskripsi mendalam mengenai proses iteratif pengembangan, mekanisme pengujian berulang, serta tahapan deployment dan maintenance dalam

lingkungan operasional nyata bengkel. Oleh karena itu, diperlukan penelitian yang tidak hanya membangun sistem, tetapi juga mendokumentasikan proses pengembangan dan implementasinya secara sistematis.

Tujuan dan Kontribusi

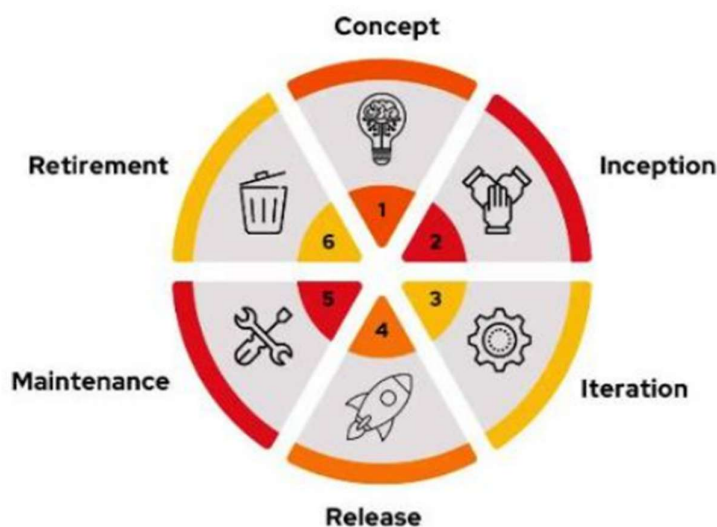
Penelitian ini bertujuan untuk menganalisis permasalahan sistem penjualan dan layanan bengkel yang berjalan secara manual, serta merancang dan mengimplementasikan sistem informasi bengkel berbasis web menggunakan Python dan framework Django dengan metode Agile.

Kontribusi penelitian ini meliputi: (1) penyediaan sistem informasi bengkel yang terintegrasi untuk pengelolaan transaksi, stok, dan laporan; (2) dokumentasi implementasi arsitektur MVT dalam sistem bengkel berbasis web; dan (3) penerapan metode Agile secara end-to-end sebagai model pengembangan sistem informasi pada usaha bengkel skala kecil dan menengah.

METODE PENELITIAN

Teknik Analisis Data

Penelitian ini menerapkan metode Agile dalam pengembangan perangkat lunaknya. Agile merupakan pendekatan pengembangan *software* yang mengandalkan serangkaian proses iteratif berskala kecil yang dilakukan secara berulang untuk menjamin fleksibilitas dan kualitas hasil akhir [8]. Agile mendukung pengembangan yang adaptif melalui penyampaian fitur secara bertahap dalam setiap *sprint*. Fleksibilitas ini memungkinkan pengembang untuk merespons perubahan kebutuhan proyek secara lebih responsif dibandingkan metode tradisional.



Gambar 2. Metode Pengembangan *Agile* [9]

- a) *Concept*: Menentukan tujuan serta kebutuhan sistem yang akan dirancang dan dikembangkan.

- b) *Inception*: Menyusun perencanaan awal proyek yang mencakup penentuan teknologi serta sumber daya yang dibutuhkan.
- c) *Iteration*: Pengembangan dilaksanakan melalui beberapa siklus, di mana setiap iterasi dievaluasi dan disempurnakan berdasarkan umpan balik yang diperoleh.
- d) *Release*: Sistem dinyatakan siap digunakan oleh pengguna setelah melalui proses pengujian dan mendapatkan persetujuan.
- e) *Maintenance*: Menjaga kinerja sistem tetap optimal serta menyesuaikan perubahan proses bisnis dan penurunan performa.
- f) *Retired*: Sistem akan dihentikan atau digantikan dengan versi terbaru apabila sudah tidak dibutuhkan atau tidak lagi digunakan.

Etika Penelitian

Penelitian ini menerapkan prinsip etika penelitian yang meliputi persetujuan partisipan (*informed consent*), perlindungan kerahasiaan data, dan integritas akademik. Peneliti memperoleh izin dari pihak bengkel sebelum melakukan observasi dan pengumpulan data, menjaga anonimitas data pelanggan dan karyawan, serta memastikan bahwa seluruh data digunakan hanya untuk kepentingan penelitian dan dilaporkan secara jujur tanpa manipulasi

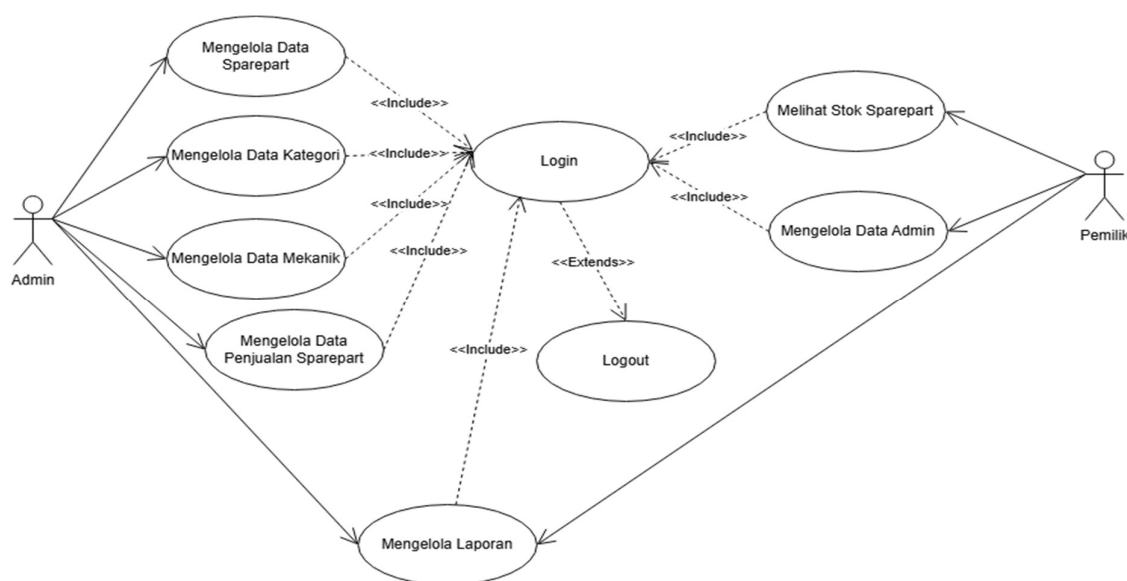
HASIL DAN PEMBAHASAN

Concept

Tahapan ini mengidentifikasi kebutuhan dengan menganalisis sistem penjualan sparepart dan service motor yang masih dilakukan secara manual melalui pembukuan. Tujuan dari tahap ini adalah mengetahui kebutuhan sistem berdasarkan permasalahan yang terjadi agar dapat dirancang sistem yang lebih sesuai dengan kebutuhan bengkel. Hasil dari tahap ini menunjukkan adanya kebutuhan akan sistem terkomputerisasi untuk menghindari kerangkapan data, mempermudah pencarian stok dan service, serta mempercepat pembuatan laporan. Dalam *Agile*, kebutuhan ini dicatat sebagai *product backlog* untuk dikembangkan secara bertahap.

Inception

Tahapan ini merupakan penyusunan rencana awal proyek yang mencakup pemilihan teknologi, perencanaan sumber daya, dan perancangan sistem menggunakan UML. UML digunakan untuk memvisualisasikan struktur dan alur kerja sistem agar pengembangan lebih terarah, mudah dipahami, dan meminimalkan kesalahan implementasi.



Gambar 3. *Use Case Diagram*

Dari gambar 3. Dapat di jelaskan lebih lanjut tentang *Use Case Diagram* yang akan di jabarkan secara rinci pada tabel berikut ini.

Tabel 1. Definisi *Use Case*

No	Use Case	Aktor	Deskripsi
1.	<i>Login</i>	Admin dan Pemilik	Melalui <i>Use Case</i> ini, admin dan pemilik diberikan akses untuk masuk ke sistem dengan memvalidasi identitas pengguna menggunakan <i>username</i> dan <i>password</i> yang terdaftar
2.	Mengelola Data Sparepart	Admin	<i>Use case</i> ini merincikan kegiatan ketika mengelola data Sparepart
3.	Mengelola Data Kategori	Admin	<i>Use case</i> ini merincikan kegiatan ketika mengelola data kategori
4.	Mengelola Data Mekanik	Admin	<i>Use case</i> ini merincikan kegiatan ketika a data mekanik
5.	Mengelola Data Penjualan Sparepart	Admin	<i>Use case</i> ini merincikan kegiatan ketika data kategori
6.	Mengelola Laporan	Admin dan Pemilik	<i>Use case</i> ini memungkinkan seorang admin dan pemilik untuk mengelola laporan seperti laporan perhari, perbulan maupun pertahun
7.	Melihat Stok Sparepart	Pemilik	<i>Use case</i> ini menjelaskan proses dari kegiatan pemilik ketika

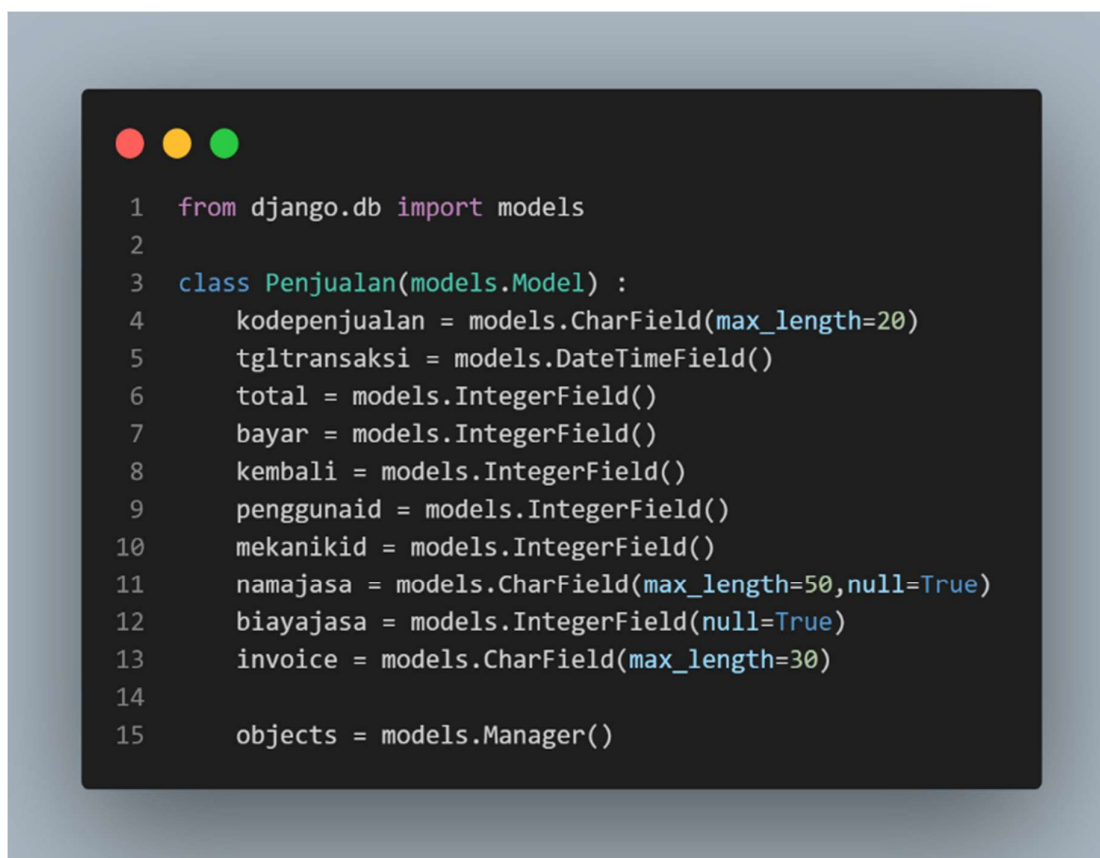
			melihat stok sparepart
8. Mengelola Data Admin	Pemilik		<i>Use case</i> ini merincikan kegiatan ketika pemilik mengelola data admin.
9. <i>Logout</i>	Admin dan Pemilik		<i>Use case</i> ini menggambarkan kegiatan admin dan pemilik ketika keluar dari aplikasi

Iteration

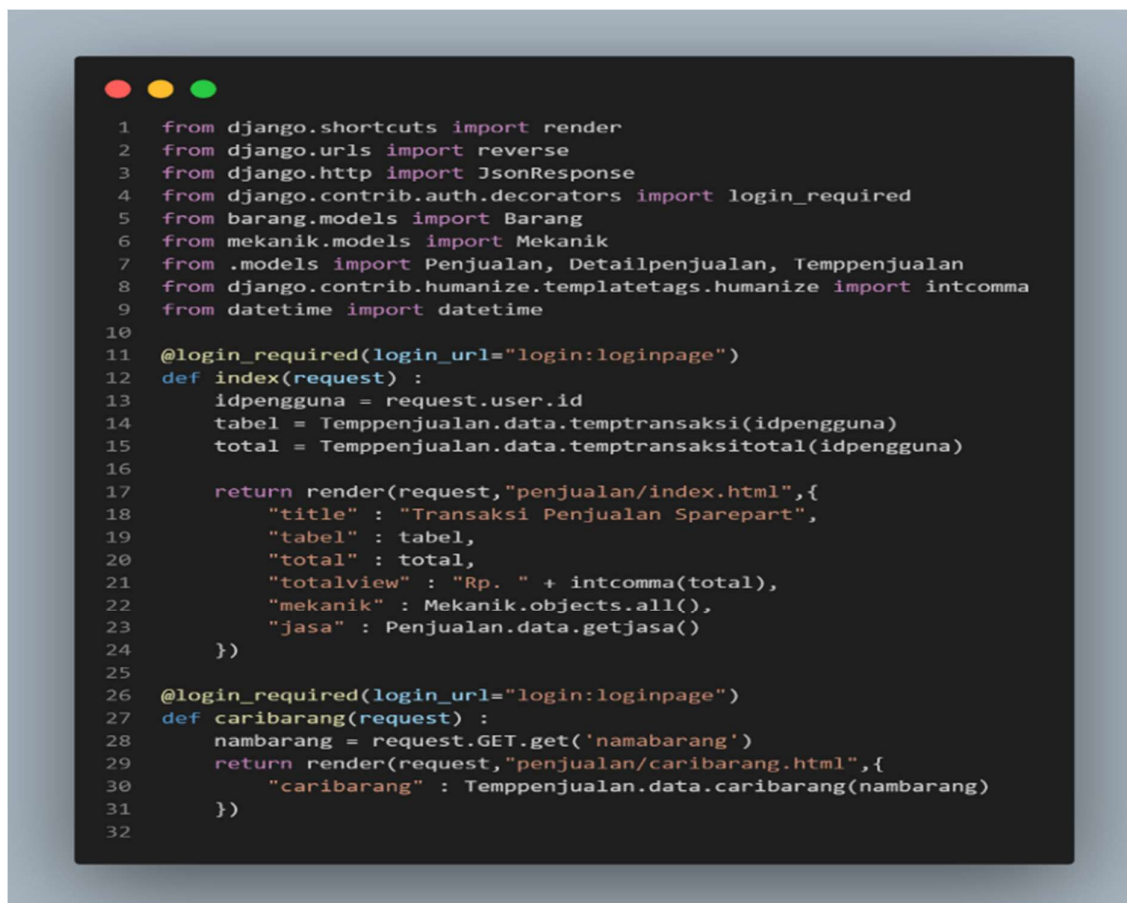
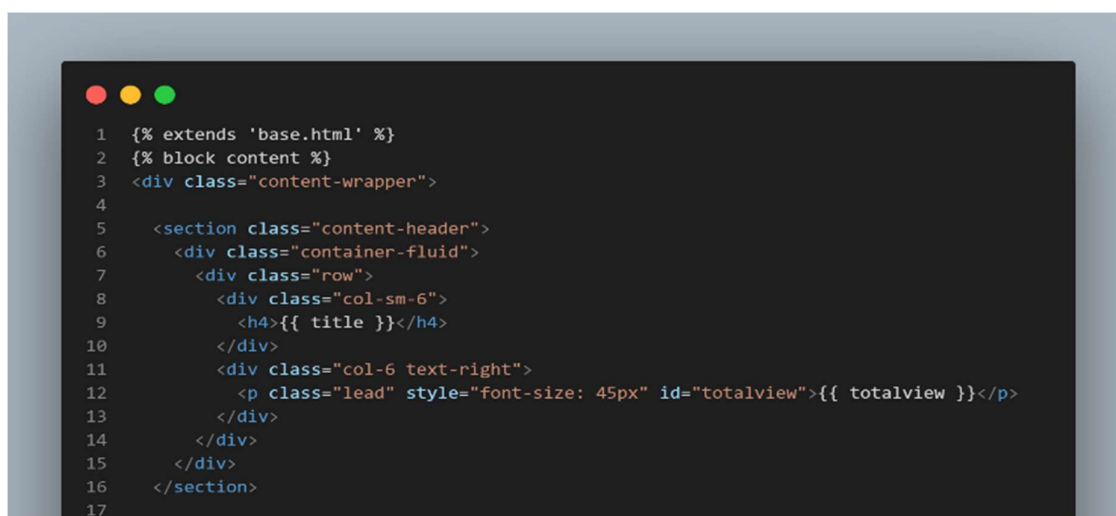
Tahapan ini mencakup proses pengembangan perangkat lunak yang dijalankan secara iteratif menggunakan metodologi *Agile*. Implementasi teknisnya menggunakan bahasa pemrograman Python dengan *framework* Django yang berbasis arsitektur MVT (*Model-View-Template*), serta didukung oleh Bootstrap untuk perancangan antarmuka pengguna. Guna menjamin kualitas perangkat lunak, dilakukan pengujian *whitebox* dan *blackbox* pada setiap akhir iterasi. Proses ini dilengkapi dengan sesi demo guna mendapatkan masukan langsung dari pengguna, sehingga pengembangan dapat mengoptimalkan fungsi sistem secara berkelanjutan sebelum memulai fase selanjutnya.

Implementasi MVT (*Model, View, Template*)

Implementasi dari tahapan MVT *framework* Django untuk penerapan model, view dan template dapat dilihat pada gambar 4, 5, dan 6.



Gambar 4. Implementasi Model Penjualan

Gambar 5. Implementasi *View* Penjualan

Gambar 6. Implementasi Template Penjualan

```

1  {% extends 'base.html' %}
2  {% block content %}
3  <div class="content-wrapper">
4
5      <section class="content-header">
6          <div class="container-fluid">
7              <div class="row">
8                  <div class="col-sm-6">
9                      <h4>{{ title }}</h4>
10                 </div>
11                 <div class="col-6 text-right">
12                     <p class="lead" style="font-size: 45px" id="totalview">{{ totalview }}</p>
13                 </div>
14             </div>
15         </div>
16     </section>
17
18     <section class="content">
19         <div class="container-fluid">
20             <div class="row">
21                 <div class="col">
22                     <div class="invoice p-3 mb-3">
23                         {% include 'penjualan/formtransaksi.html' %}
24                         <hr />
25                         <div id="loadtabeltemptrx">
26                             {% include "penjualan/tabeltransaksi.html" %}
27                         </div>
28                         <hr />
29                     <div class="row">
30                         <div class="col-md-6">
31                             <a
32                                 href="javascript:void()"
33                                 data-toggle="modal"
34                                 data-target="#modalPembayaran"
35                                 class="btn btn-success"
36                                 >PROSES PEMBAYARAN {F7}</a>
37                             </div>
38                         </div>
39                         <div class="col-md-6 text-right">
40                             <a href="{% url 'penjualan:riwayattransaksi' %}">Riwayat Transaksi</a>
41                         </div>
42                     </div>
43                 </div>
44             </div>
45         </div>
46     </section>
47 </div>
48 {% include "penjualan/modalbarang.html" %}
49 {% include "penjualan/modalbayar.html" %}
50 {% endblock %}

```

Gambar 7. Sambungan bagian Implementasi Template Penjualan

Selanjutnya, berdasarkan hasil proses pengkodean dan penerapan arsitektur MVT, menghasilkan tampilan yang dapat di akses melauai web browser yang dapat dilihat tampilan antar muka yang ditunjukkan pada gambar 8 dan 9.

KEVIN MOTOR

ADMIN BENGKEL

Dashboard

Transaksi Penjualan

Master Data

Laporan

Transaksi Penjualan Sparepart

Rp. 30,000

Kode Nama Harga Jumlah Simpan

Kode Nama Harga Jumlah + Simpan

Aksi	Kode	Nama	Satuan	Harga	Jumlah	Sub Total
	14	BOLA LAMPU DEPAN	PCS	15,000	1	15,000
	77	OLI GARDAN	PCS	15,000	1	15,000

PROSES PEMBAYARAN (F7)

Riwayat Transaksi

Copyright © KEVIN MOTOR

Gambar 8. Hasil Implementasi Template Penjualan

KEVIN MOTOR

ADMIN BENGKEL

Dashboard

Transaksi Penjualan

Master Data

Laporan

Total Transaksi

Mekanik: Irawan

Jasa Service: GANTI OLI (Rp. 5,000)

Total Transaksi Barang	Jasa	Total Transaksi Barang + Jasa
Rp. 30,000	Rp. 5,000	Rp. 35,000

Jumlah Bayar: 50.000

Sisa Bayar: 15.000

PROSES TRANSAKSI

PROSES PEMBAYARAN (F7)

Riwayat Transaksi

Copyright © KEVIN MOTOR

Gambar 9. Hasil Implementasi Template Penjualan Form Pembayaran

Pada gambar 8 dan 9 pengguna dapat melakukan transaksi penjualan dengan memasukkan data sparepart dengan mengklik icon search (pencarian) kemudian masukan nama barang yang dijual dan memasukkan jumlah barang yang di beli klik simpan untuk menyimpan sementara kedalam tabel transaksi sistem secara otomatis menghitung total bayar berdasarkan harga dan jumlah yang di inputkan. Proses pembayaran dapat dilakukan dengan menklik tombol proses pembayaran setelah itu sistem akan menampilkan modal untuk melakukan proses pembayaran dengan menginputkan data yang sesuai pada modal tersebut.

Pengujian *Blackbox*

Pengujian *black box* dilakukan untuk memvalidasi bahwa sistem yang dibuat beroperasi sesuai dengan fitur yang telah ditetapkan tanpa melibatkan analisis terhadap kode program. Metode ini, juga dikenal sebagai *functional* atau *behavioral testing*, menitik

beratkan evaluasi pada aspek fungsionalitas perangkat lunak guna memastikan *output* yang dihasilkan sesuai dengan *input* yang diberikan [10]. Metode ini berfokus pada pengujian fungsi sistem untuk mengidentifikasi kesalahan operasional serta memastikan stabilitas dan performa sistem di berbagai kondisi penggunaan, baik melalui validasi data masukan maupun hasil keluaran.

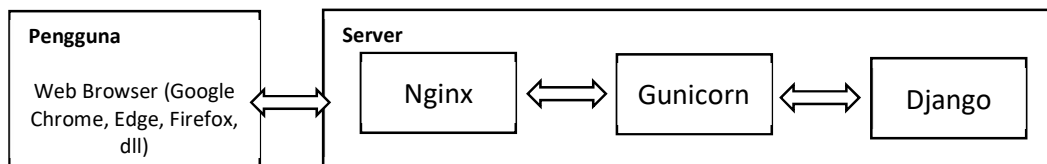
Tabel 2. Pengujian *Black Box*

No	Fitur yang Pengujian	Skenario Pengujian	Kriteria Keberhasil	Hasil Pengujian
1.	<i>Login</i>	Pengguna memasukan username dan password	Sistem menampilkan halaman utama setelah berhasil melakukan login	Berfungsi
2.	Mengelola Data Sparepart	Pengguna melakukan tambah, ubah dan hapus Sparepart	Sistem dapat melakukan tambah, ubah dan hapus data Sparepart	Berfungsi
3.	Mengelola Data Kategori	Pengguna melakukan tambah, ubah dan hapus Kategori	Sistem dapat melakukan tambah, ubah dan hapus data Kategori	Berfungsi
4.	Mengelola Data Mekanik	Pengguna melakukan tambah, ubah dan hapus Mekanik	Sistem dapat melakukan tambah, ubah dan hapus data Mekanik	Berfungsi
5.	Mengelola Data Penjualan Sparepart	Pengguna melakukan transaksi penjualan sparepart	Sistem dapat melakukan transaksi penjualan sparepart dan dapat mencetak laporan transaksi penjualan	Berfungsi
6.	Mengelola Laporan	Pengguna melakukan pengelolaan laporan	Sistem dapat melakukan laporan seperti laporan perhari, perbulan maupun pertahun	Berfungsi
7.	Melihat Stok Sparepart	Pengguna melihat stok sparepart	Sistem dapat menampilkan data stok sparepart	Berfungsi
8.	Mengelola Data Admin	Pengguna melakukan tambah, ubah dan hapus Admin	Sistem dapat melakukan tambah, ubah dan hapus data Admin	Berfungsi
9.	<i>Logout</i>	Pengguna <i>logout</i> dari sistem	Sistem berhasil menutup akses pengelolaan data dan	Berfungsi

kembali ke halaman
login

Release

Setelah melalui serangkaian pengujian dan evaluasi, untuk memastikan bahwa seluruh fitur berfungsi sesuai spesifikasi dan algoritma berjalan dengan benar. Adapun tahapan release ini yaitu diawali dengan proses deploy yaitu melakukan konfigurasi konfigurasi yang dibutuhkan agar sistem dapat berjalan semestinya. Pada sistem yang dibangun menggunakan Django maka peneliti menggunakan gunicorn sebagai WSGI HTTP Server untuk aplikasi web Python yang bertindak sebagai jembatan antara server web (seperti Nginx). Yang dapat dilihat dengan gambar berikut ini.



Gambar 10. Proses Kerja Aplikasi Django

Maintenance

Maintenance sistem merupakan tahap akhir pengembangan aplikasi yang bertujuan menjaga agar sistem yang dibangun tetap berjalan dengan baik. Kegiatan *maintenance* meliputi pengecekan file system, kapasitas server dan storage, database, serta keamanan aplikasi. Maintenance dilakukan untuk menyesuaikan perubahan proses bisnis dan meningkatkan performa sistem yang menurun akibat bertambahnya data, kerusakan file.

Retired

Apabila sistem dinilai tidak lagi relevan dengan kebutuhan pengguna, maka akan dilakukan perubahan ke versi yang terbaru. Proses penghentian ini didasarkan pada hasil evaluasi menyeluruh terhadap aspek performa, keamanan, dan efektivitas, dengan tetap mengutamakan migrasi data penting ke platform yang baru demi menjaga keberlangsungan layanan serta mengadaptasi perkembangan teknologi terkini.

KESIMPULAN

Berdasarkan analisis dan pembahasan yang telah dipaparkan sebelumnya, dapat disimpulkan bahwa penerapan sistem bengkel berbasis web mampu meningkatkan efektivitas pengelolaan penjualan sparepart dan service motor. Sistem ini mempermudah pencatatan transaksi, pengelolaan stok, serta pembuatan laporan sehingga mengurangi kesalahan dan duplikasi data. Selain itu, sistem bengkel dikembangkan menggunakan bahasa pemrograman Python dengan framework Django yang menerapkan konsep Model View Template (MVT) sehingga pengelolaan data dan tampilan sistem menjadi lebih terstruktur. Dengan adanya sistem ini, waktu kerja pemilik dan karyawan bengkel menjadi lebih efisien karena seluruh proses dilakukan secara terkomputerisasi.

REFERENSI

- [1] S. Tasya Zhawani and M. Irwan Padli Nasution, "Implementasi Teknologi Informasi dalam Pertumbuhan Efektivitas, Efisiensi dan Produktivitas Perusahaan," *Kohesi J. Multidisiplin Saintek*, vol. 02, no. 01, pp. 21–30, 2023.
- [2] D. Wahyudi and A. Nur Rahmi, "Pembuatan Sistem Informasi Bengkel Berbasis Website Pada Bengkel Ferdi Motor," *Inf. Syst. J.*, vol. 5, no. 2, pp. 1–6, 2022, doi: <https://doi.org/10.24076/infosjournal.2022v5i2.932>.
- [3] A. K. Putri and D. I. Nur, "Penggunaan Bahasa Python Untuk Analisis Dan Visualisasi Data Penduduk Di Desa Sumberjo, Nganjuk," *J. Pengabd. Kpd. Masy.*, vol. 3, no. 3, pp. 206–217, 2023, [Online]. Available: https://jurnal.fkip.samawa-university.ac.id/karya_jpm/index
- [4] M. Hikmal Maulana, "Python Bahasa Pemrograman Yang Ramah Bagi Pemula," *JISCO (Journal Inf. Syst. Comput.)*, vol. 2, no. 2, pp. 73–78, 2024, doi: <https://doi.org/10.30631/jisco.v2i2.105>.
- [5] P. Satya Saputra, P. Aditya Pratama, and L. Putu Ary Sri Tjahyanti, "Perancangan Dan Komparasi Web Server Nginx Dengan Web Server Apache Serta Pemanfaatan Reverse Proxy Server Pada Nginx," *J. Komput. dan Teknol. Sains*, vol. 2, no. 1, pp. 16–21, 2023.
- [6] S. Retnowati and D. D. Nugroho, "Implementasi Restful Web Service pada Aplikasi Surat Keterangan Pendamping Ijazah (SKPI) Berbasis Web," *J. Kridatama Sains Dan Teknol.*, vol. 6, no. 01, pp. 247–262, 2024, doi: 10.53863/kst.v6i01.1117.
- [7] Gat, "Pemanfaatan Python dan Framework Django Sebagai Dashboard Sistem Informasi Pengelolaan Skripsi Pada STMIK Pontianak," *Semin. Nas. Corisindo*, pp. 128–133, 2023, [Online]. Available: <https://stmikpontianak.org/ojs/index.php/corisindo/article/download/154/24>
- [8] N. Etrariadi and E. Sarah Permata A'inunisya, "Pengembangan Website Manajemen Proyek Menggunakan Metode Agile Scrum (Studi Kasus Diskopindag Kota Malang)," *J. Nas. Teknol. dan Sist. Inf.*, vol. 9, no. 1, pp. 55–66, 2023.
- [9] A. Rahman Ismail, "Penerapan Metode Agile Pada Perancangan Sistem Informasi Pengajuan Nomor Surat di Pemerintahan Desa," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 7, no. 2, pp. 285–290, 2025.
- [10] M. Maulida, F. Zahro, R. Hakim, and M. S. Akbar, "Pengujian Black Box Testing Pada Sistem Website Pemesanan Online Toko Ayam Krispy," *J. Media Akad.*, vol. 3, no. 5, pp. 3031–5220, 2025.
- [11] Al Baiti, M., Mizani Putri, U. and Kurniawan, H. (2025) "SISTEM INFORMASI PELAYANAN ADMINISTRASI TERPADU BERBASIS WEB PADA BALAI PENJAMINAN MUTU PENDIDIKAN PROVINSI JAMBI MENGGUNAKAN FRAMEWORK CODEIGNITER", *Jurnal Mnemonic*, 8(2), pp. 193-202. doi: 10.36040/mnemonic.v8i2.14949.

- [12] Arliana Fadillah Putri Hari Wijaya, Mutamassikin Mutamassikin, & Heru Kurniawan. (2025). Perancangan Sistem Informasi Akademik Berbasis Web Pada SMAS Islam Darul Fikri Muaro Jambi. *Jurnal Publikasi Ilmu Komputer Dan Multimedia*, 4(3), 24–38. <https://doi.org/10.55606/jupikom.v4i3.4859>.
- [13] Nabila Putri Mareta, Andreo Yudertha, & Heru Kurniawan. (2025). Perancangan Sistem Informasi Pemesanan Jasa Wedding Organizer Berbasis Web di Ranni Gallery Palembang. *Jurnal Teknik Informatika Dan Teknologi Informasi*, 5(1), 472–487. <https://doi.org/10.55606/jutiti.v5i1.5390>.
- [14] Nashrullah, A. C. N., Mutamassikin, & Heru Kurniawan. (2025). Perancangan Sistem Informasi Manajemen Penjualan dan Jasa Berbasis Web Pada PT Ranah Sahabat Mandiri Kota Jambi. *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, 4(2), 1186–1198. <https://doi.org/10.62712/juktisi.v4i2.610>.
- [15] Widya Silviana, Elfitra, & Heru Kurniawan. (2025). Perancangan Desain UI/UX Aplikasi Penjualan Batik Berbasis Website Pada Toko Batik Jambi Ariny Menggunakan Metode User Centered Design (UCD). *Jurnal Komputer Teknologi Informasi Sistem Informasi (JUKTISI)*, 4(2), 687–698. doi.org/10.62712/juktisi.v4i2.486.