

# Pengembangan Sistem Informasi Bengkel Otomotif dengan Metode *Agile Scrum*

Glorian Hilarius Kiantin Beda<sup>1</sup>, Adli Farizi<sup>2</sup>, Icha Maulidya<sup>3</sup>, Cynthia Octavania Putri Salma<sup>4</sup>, Aditya Wicaksono<sup>5,\*</sup>, Muhammad Nasir<sup>6</sup>

<sup>1,2,3,4,5,6</sup> Sekolah Vokasi, Teknologi Rekayasa Perangkat Lunak, IPB University, Bogor, Indonesia

Email: <sup>1</sup>glorianbeda@apps.ipb.ac.id, <sup>2</sup>adlifarizi@apps.ipb.ac.id, <sup>3</sup>ichamaulidya@apps.ipb.ac.id, <sup>4</sup>octvncynthia@apps.ipb.ac.id, <sup>5</sup>adityawicaksono@apps.ipb.ac.id, <sup>6</sup>m\_nasir@apps.ipb.ac.id

<sup>\*</sup> Email Corresponding Author

**Abstrak**—Transformasi digital telah menjadi kebutuhan mendesak dalam berbagai sektor industri, termasuk layanan bengkel otomotif yang selama ini masih banyak menggunakan sistem manual. Permasalahan umum yang dihadapi meliputi pencatatan data pelanggan yang tidak efisien, manajemen keuangan yang rentan kesalahan, serta kurangnya transparansi dan kenyamanan dalam pelayanan pelanggan. Agar dapat mengatasi tantangan tersebut, penelitian ini bertujuan untuk mengembangkan sistem informasi bengkel otomotif berbasis web dengan menerapkan metodologi *Agile Scrum*. Metode ini dipilih karena bersifat iteratif dan adaptif, memungkinkan pengembangan sistem secara bertahap melalui sprint yang disusun berdasarkan prioritas dan kebutuhan pengguna. Proses pengembangan dilakukan dalam empat *sprint*, dengan masing-masing *sprint* mencakup perencanaan, implementasi, evaluasi, dan pengujian. Fitur yang berhasil dikembangkan antara lain sistem autentikasi, *form* pengecekan progres perbaikan, manajemen entri servis, ulasan pelanggan, *pop-up chat real-time*, laporan keuangan, serta pengelolaan data lokasi dan kontak bengkel. Setiap tahapan pengembangan dilakukan secara kolaboratif oleh tim *Scrum* yang terdiri dari *Scrum Master*, *Product Owner*, dan *Developer*, yang secara aktif berpartisipasi dalam perencanaan backlog dan pembuatan user story untuk memastikan hasil yang lebih sesuai dengan kebutuhan pelanggan. Evaluasi terhadap sistem dilakukan menggunakan metode *Black Box Testing*, yang menunjukkan bahwa semua fitur telah berjalan sesuai dengan spesifikasi dan tanpa kesalahan berarti, menjamin kualitas dan kehandalan sistem. Hasil penelitian menunjukkan bahwa penerapan *Agile Scrum* tidak hanya meningkatkan efisiensi internal operasional bengkel, tetapi juga mendukung pelayanan yang lebih transparan dan interaktif kepada pelanggan melalui berbagai fitur seperti chat secara *real-time* dan sistem ulasan. Fleksibilitas metode *Agile* dalam menanggapi perubahan kebutuhan pengguna secara cepat juga menjadi nilai tambah penting dalam proses pengembangan ini, serta memungkinkan sistem untuk beradaptasi dengan dinamika kebutuhan yang berkembang. Dengan demikian, sistem informasi yang dibangun mampu menjadi solusi praktis dan efektif dalam mendukung digitalisasi layanan bengkel otomotif. Penelitian ini diharapkan dapat menjadi referensi bagi pengembangan sistem serupa di sektor layanan lain yang membutuhkan pendekatan fleksibel dan kolaboratif dalam menghadapi tantangan perkembangan teknologi yang cepat.

**Kata Kunci:** Bengkel Otomotif, Layanan Pelanggan, Pengembangan Perangkat Lunak, Scrum, Sistem informasi, Transformasi Digital

**Abstract**—Digital transformation has become an urgent need in various industrial sectors, including automotive repair shop services that have so far primarily used manual systems. Common problems faced include inefficient customer data recording, error-prone financial management, and a lack of transparency and convenience in customer service. To overcome these challenges, this study aims to develop a web-based automotive repair shop information system by implementing the Agile Scrum methodology. This method was chosen because it is iterative and adaptive, allowing for gradual system development through sprints arranged based on user priorities and needs. The development process was carried out in four sprints, including planning, implementation, evaluation, and testing. Features successfully developed include an authentication system, a repair progress check form, service entry management, customer reviews, real-time chat pop-ups, financial reports, and repair shop location and contact data management. Each stage of development was carried out collaboratively by a Scrum team consisting of a Scrum Master, Product Owner, and Developer, who actively participated in backlog planning and user story creation to ensure results that were more in line with customer needs. The system was evaluated using the Black Box Testing method, which showed that all features ran according to specifications and without significant errors, ensuring the quality and reliability of the system. The study results showed that the implementation of Agile Scrum not only increased the internal efficiency of workshop operations but also supported more transparent and interactive services to customers through various features such as real-time chat and review systems. The flexibility of the Agile method in responding to changes in user needs quickly is also a significant added value in this development process, and allows the system to adapt to the dynamics of evolving needs. Thus, the information system built can become a practical and effective solution in supporting the digitalisation of automotive workshop services. This study is expected to be a reference for developing similar systems in other service sectors that require a flexible and collaborative approach in facing the challenges of rapid technological developments.

**Keywords:** Automotive Workshop, Customer Service, Digital Transformation, Information Systems, Scrum, Software Development

## 1. PENDAHULUAN

Transformasi digital merupakan kebutuhan yang mendesak bagi sektor industri di era revolusi industri 4.0 [1]. Salah satu sektor yang turut terdampak adalah layanan bengkel otomotif. Saat ini, sebagian besar bengkel otomotif masih mengandalkan sistem manual dalam menjalankan kegiatan operasional harian seperti pencatatan data pelanggan, manajemen keuangan, dan penjadwalan servis kendaraan [2], [3], [4], [5]. Kondisi ini menimbulkan berbagai permasalahan seperti kurangnya efisiensi waktu, risiko kehilangan data, serta penurunan kualitas layanan kepada pelanggan.

Untuk mengatasi permasalahan tersebut, sejumlah penelitian telah dilakukan guna menemukan solusi yang tepat dalam meningkatkan efisiensi dan kualitas layanan bengkel otomotif [6]. Menurut Fitri Yunita et al. dalam penelitiannya pada Bengkel Kakella Motor, menunjukkan bahwa penerapan sistem informasi berbasis web dapat meningkatkan efisiensi dalam pengelolaan data transaksi, servis, dan *sparepart* dibandingkan metode manual yang tidak efisien terhadap waktu dan berisiko tinggi terhadap kehilangan data [7]. Penelitian lain oleh Dwi Ningsih et al. pada Bengkel Bimo Motor juga menyoroti permasalahan dalam pelayanan pelanggan akibat sistem reservasi manual, sehingga mereka mengembangkan sistem reservasi berbasis web dengan metode *prototyping* guna mengelola jadwal servis secara terkomputerisasi [8].

Meskipun kedua penelitian tersebut telah memberikan kontribusi yang positif, pendekatan pengembangan sistem yang digunakan masih memiliki keterbatasan. Pendekatan linear seperti *Waterfall*, dan pendekatan iteratif seperti *Prototyping* cenderung kurang mampu merespons perubahan kebutuhan pengguna secara cepat. Di sisi lain, metode *Agile*, yang juga bersifat iteratif, menawarkan tingkat fleksibilitas yang lebih tinggi. Dengan melibatkan pengguna dalam setiap iterasi, metode ini memungkinkan pengembangan sistem yang dapat menyesuaikan terhadap perubahan kebutuhan yang dinamis [9], [10]. Dalam penelitiannya, Cut Asiana et al. membuktikan bahwa penerapan pendekatan *Agile* dalam perancangan sistem penjualan dan pelayanan pada Bengkel Hiba Mobil mampu meningkatkan efisiensi transaksi serta kepuasan pelanggan secara signifikan [4].

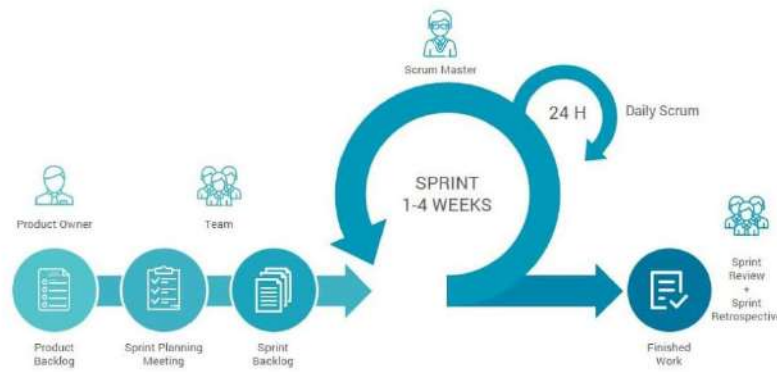
Oleh karena itu, penelitian ini bertujuan untuk merancang sistem bengkel otomotif berbasis web dengan menerapkan metode *Agile* sebagai pendekatan pengembangannya. Kontribusi utama dari penelitian ini adalah menghadirkan sistem digital yang tidak hanya mengotomatisasi proses layanan bengkel, tetapi juga mendukung fleksibilitas dalam pengembangan sesuai kebutuhan pengguna melalui iterasi dan umpan balik secara berkala. Dengan mengacu pada penelitian terdahulu dan mengintegrasikan metode *Agile*, penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam pengembangan sistem informasi di sektor layanan otomotif.

## 2. METODE PENELITIAN

### 2.1 Skema Alur Penelitian

Penelitian ini menggunakan metode pengembangan perangkat lunak secara iteratif dengan pendekatan *Agile*. *Agile* merupakan pendekatan yang berfokus pada iterasi singkat, kolaborasi tim, serta respons cepat terhadap perubahan kebutuhan pengguna. Pendekatan ini memungkinkan sistem dikembangkan secara bertahap (*incremental*) dengan evaluasi berkelanjutan antara tim pengembang dan pemangku kepentingan, sehingga hasil akhir dapat disesuaikan dengan kebutuhan [11], [12]. Salah satu implementasi *Agile* yang digunakan dalam penelitian ini adalah *Scrum*, yaitu kerangka kerja yang membagi proses pengembangan ke dalam siklus sprint yang mencakup tahap perencanaan, implementasi, dan evaluasi secara berkala [13].

Keunggulan utama pendekatan *Agile* terletak pada fleksibilitasnya dalam mengakomodasi perubahan kebutuhan secara dinamis, sehingga dapat mengurangi risiko ketidaksesuaian produk dan mendorong efisiensi melalui siklus pengembangan singkat [11], [13]. Selain itu, kolaborasi intensif dan umpan balik cepat meningkatkan kualitas serta kemudahan pemeliharaan sistem, menjadikannya ideal untuk proyek yang membutuhkan adaptasi cepat terhadap dinamika pasar [2]. Tahapan-tahapan pada pendekatan *Agile Scrum* disajikan pada Gambar 1.



Gambar 1. Tahapan Agile Scrum

## 2.2 Requirement Analysis

Pada tahap ini, pengembangan dimulai dengan melakukan analisis kebutuhan, yang bertujuan untuk memahami secara menyeluruh permasalahan yang dihadapi oleh pihak bengkel dan menentukan solusi yang tepat. Proses ini dilakukan dengan mengidentifikasi permasalahan utama dalam proses bisnis yang berjalan serta menggali informasi penting dari para *stakeholder* untuk menyusun kebutuhan fungsional dan non-fungsional dari sistem yang akan dikembangkan [14].

Selanjutnya, dibentuk scrum team yang terdiri dari *Scrum Master*, *Product Owner*, dan *Developers*, sesuai dengan peran dan keahlian masing-masing. Tim ini menyusun User Story sebagai acuan pengembangan, dengan struktur format seperti sebagai apa, saya ingin sehingga bagaimana.

## 2.2 Product Backlog

*Product backlog* merupakan daftar terurut dari semua fitur, fungsi, persyaratan, perbaikan, dan penyempurnaan yang harus diterapkan pada produk [15]. *Product backlog* disusun berdasarkan hasil analisis kebutuhan yang telah dilakukan sebelumnya [16]. *Product backlog* bersifat dinamis dan terus berkembang sesuai dengan kebutuhan dan umpan balik dari *stakeholder*.

## 2.4 Sprint Planning

*Sprint Planning* merupakan sesi perencanaan di awal setiap *sprint* yang bertujuan untuk menentukan prioritas dan memilih item dari *Product Backlog* yang akan dikerjakan. Kegiatan ini melibatkan *Scrum Master*, *Product Owner*, dan *Developers* untuk menyepakati *Sprint Backlog* sebagai acuan tugas selama *sprint* berlangsung [17].

## 2.2 Sprint

*Sprint* adalah periode waktu terbatas yang digunakan oleh tim pengembang untuk membangun bagian fungsional dari sistem secara insentif berdasarkan *product backlog* dan *sprint backlog* yang telah disepakati [14]. Pada tahap ini, pengembangan perangkat lunak dilakukan secara iteratif dan kolaboratif, dengan dukungan pertemuan harian (*daily scrum*) untuk memantau perkembangan, mengatasi hambatan, dan menyelaraskan aktivitas tim [18]. Tahapan dalam *sprint* ini terdiri dari beberapa aktivitas yang dilakukan secara teratur, yaitu:

### a. Daily Scrum

Pada tahap ini, dilakukan evaluasi terhadap tugas-tugas yang telah diselesaikan selama *sprint* sebelumnya guna memastikan koordinasi tim tetap terjaga dan setiap kendala dapat diatasi secara efektif.

### b. Sprint Review

*Sprint Review* adalah pertemuan di akhir *sprint* yang dilaksanakan untuk mengevaluasi fitur yang dihasilkan. *Development Team* mendemonstrasikan fitur yang telah selesai kepada *Product Owner* dan *stakeholder*, kemudian menerima umpan balik untuk perbaikan. Hasilnya adalah *Product Backlog* yang direvisi untuk *sprint* berikutnya.

### c. Sprint Retrospective

*Sprint Retrospective* merupakan sesi evaluasi yang dilaksanakan setelah *Sprint Review* dan sebelum *Sprint Planning* berikutnya, dengan tujuan meninjau proses kerja selama *sprint* sebelumnya serta merumuskan langkah-langkah perbaikan untuk meningkatkan efektivitas *sprint* selanjutnya.

### 3. HASIL DAN PEMBAHASAN

Pada bagian ini, menyajikan hasil dari proses pengembangan sistem yang dilakukan dengan pendekatan *Agile Scrum*. Pengembangan sistem dibagi ke dalam beberapa *sprint*, yang meliputi perencanaan, pelaksanaan, dan evaluasi. Dengan menggunakan *Scrum*, tim dapat bekerja secara kolaboratif dan adaptif untuk menghasilkan solusi yang sesuai dengan kebutuhan pengguna. Penjelasan ini akan menggambarkan bagaimana setiap elemen dalam metode *Scrum* berperan dalam mendukung keberhasilan pengembangan sistem.

#### 3.1 Requirement Analysis

Tahap analisis kebutuhan dilakukan untuk mengidentifikasi anggota *scrum* dan perannya, seperti yang ditunjukkan pada Tabel 1.

Tabel 1. *Scrum Team*

Nama	Peran
Rizky Dwi Prasetya	<i>Product Owner</i>
Glorian Hilarius Kiantin Beda	<i>Scrum Master</i>
Icha Maulidya	<i>Developer</i>
Adli Farizi	<i>Developer</i>
Cynthia Octavania Putri Salma	<i>Developer</i>

Setelah penetapan anggota *scrum*, langkah selanjutnya menyusun tabel cerita pengguna, sebagai acuan untuk proses pengembangan sistem dalam format yang sesuai standar yang diharapkan. Tabel *user stories* ditunjukkan pada Tabel 2.

Tabel 2. *User Stories*

No	Cerita Pengguna ( <i>User Story</i> )	Kriteria Penerimaan ( <i>Acceptance Criteria</i> )
1	Sebagai admin, saya ingin dapat melakukan <i>login</i> agar bisa mengakses fitur manajemen sistem.	<ul style="list-style-type: none"> <li>- Tersedia <i>form login</i> dengan validasi.</li> <li>- Hanya akun admin yang bisa <i>login</i>.</li> <li>- Tampil pesan <i>error</i> jika <i>login</i> gagal.</li> </ul>
2	Sebagai pelanggan, saya ingin mengecek progres perbaikan kendaraan agar tahu sejauh mana kendaraan saya ditangani.	<ul style="list-style-type: none"> <li>- Pelanggan dapat melihat status dan progres kendaraan melalui <i>form</i> dengan input Nomor Plat kendaraan.</li> </ul>
3	Sebagai admin, saya ingin mencatat entri servis agar data kendaraan pelanggan dapat tersimpan dan dikelola.	<ul style="list-style-type: none"> <li>- Admin dapat input jenis servis, tanggal, estimasi biaya, dan keterangan lainnya.</li> <li>- Data tersimpan di basis data.</li> </ul>
4	Sebagai pelanggan atau admin, saya ingin bisa menggunakan fitur <i>pop-up chat</i> agar dapat bertanya atau menjawab pertanyaan secara langsung.	<ul style="list-style-type: none"> <li>- Tersedia fitur chat <i>real-time</i>.</li> <li>- Pesan langsung terkirim dan tampil sesuai pengirim.</li> <li>- Admin dapat membalas pertanyaan pelanggan.</li> </ul>
5	Sebagai pelanggan, saya ingin melihat layanan dan fasilitas bengkel agar tahu layanan apa saja yang tersedia.	<ul style="list-style-type: none"> <li>- Tersedia halaman daftar fasilitas dan layanan.</li> <li>- Info ditampilkan lengkap dan rapi.</li> </ul>
6	Sebagai pelanggan, saya ingin melihat lokasi dan kontak bengkel agar mudah dihubungi dan dikunjungi.	<ul style="list-style-type: none"> <li>- Ada informasi alamat, no. telepon, jam buka, dan tautan ke Google Maps.</li> </ul>

7	Sebagai admin, saya ingin dapat mengatur dan memperbarui data lokasi dan kontak bengkel agar informasi tetap <i>up-to-date</i> .	- Admin dapat CRUD ( <i>Create, Read, Update, Delete</i> ) data lokasi dan kontak.
8	Sebagai pelanggan dan admin, saya ingin dapat memberi dan melihat <i>feedback</i> serta <i>rating</i> agar kualitas layanan bisa diketahui.	- Pelanggan dapat memberi <i>rating</i> dan komentar. - Admin dapat melihat dan menanggapi <i>feedback</i> .
9	Sebagai pelanggan, saya ingin mendapat estimasi kunjungan kembali agar tahu kapan kendaraan perlu servis berikutnya.	- Sistem menampilkan estimasi tanggal servis berikutnya berdasarkan riwayat sebelumnya.
10	Sebagai admin, saya ingin melihat laporan keuangan bengkel agar dapat mengevaluasi pemasukan dan pengeluaran.	- Laporan ditampilkan berdasarkan periode tertentu (harian/bulanan).
11	Sebagai admin, saya ingin sistem mencatat keuangan secara otomatis agar akurat dan mengurangi <i>human error</i> .	- Setiap transaksi keuangan (servis, pembelian, dsb.) tercatat otomatis.
12	Sebagai admin, saya ingin mengelola data keuangan secara manual jika ada pembetulan atau penyesuaian.	- Admin dapat melakukan CRUD data keuangan.

### 3.2 Product Backlog

Langkah selanjutnya adalah pembuatan *product backlog*. *Product Backlog* merupakan daftar lengkap fitur yang diperlukan sebagai acuan bagi *developer* dalam menentukan prioritas pengembangan sistem. Tabel *product backlog* ditunjukkan pada Tabel 3.

**Tabel 3.** *Product Backlog*

No	<i>Product Backlog Item</i>	Prioritas
1	Pembuatan Gambaran Sistem	Tinggi
2	Autentikasi & Sistem <i>Login</i>	Tinggi
3	<i>Form</i> Cek Progres Perbaikan (Pelanggan)	Sedang
4	Manajemen Entri Servis (Admin)	Tinggi
5	<i>Pop-up Chat</i> dengan Bengkel (Pelanggan & Admin)	Sedang
6	Fasilitas & Layanan Bengkel (Pelanggan)	Rendah
7	Lokasi dan Kontak Bengkel (Pelanggan)	Rendah
8	Manajemen Lokasi & Kontak Bengkel (Admin)	Sedang
9	<i>Feedback &amp; Rating</i> (Pelanggan & Admin)	Sedang
10	Otomatisasi Estimasi Kunjungan Kembali Bengkel (Pelanggan)	Tinggi
11	Laporan Keuangan Bengkel (Admin)	Tinggi
12	Otomatisasi Pencatatan Keuangan	Tinggi
13	CRUD Data Keuangan	Tinggi
14	Persiapan Akhir & <i>Deployment</i>	Tinggi

### 3.3 Sprint Planning

Setelah menentukan prioritas, tahapan selanjutnya yaitu merencanakan aktivitas pengembangan sistem. Setiap aktivitas dibagi menjadi empat iterasi. Perencanaan *Sprint* dilakukan untuk memastikan bahwa pengembangan sistem berjalan secara terstruktur, efisien, dan sesuai dengan kebutuhan. Tabel 4 menyajikan iterasi pertama pada proses perencanaan pengembangan sistem.

**Tabel 4.** *Sprint Planning 1*

<b>ID</b>	<b>Item Backlog</b>	<b>Task</b>	<b>Estimasi (Hari)</b>
1	Pembuatan Gambaran Sistem	Membuat <i>use case</i> diagram	1
		Membuat <i>class</i> diagram	1
		Setup proyek di Microsoft Project	2
		Menentukan <i>sprint backlog</i>	1
2	Autentikasi & Sistem <i>Login</i>	Desain UI Halaman Login Admin	1
		Integrasi login dengan <i>database</i>	2
		Implementasi <i>backend</i> login admin	2
		<i>Testing</i> Fitur <i>Login/Logout</i>	1

Pada tabel 4, total estimasi waktu pengerjaan *Sprint 1* adalah 11 hari kerja. Selanjutnya, perencanaan berlanjut ke iterasi kedua yang ditampilkan pada Tabel 5.

**Tabel 5.** *Sprint Planning 2*

<b>ID</b>	<b>Item Backlog</b>	<b>Task</b>	<b>Estimasi (Hari)</b>
3	Cek <i>Form</i> Progres Perbaikan (Pelanggan)	Desain UI <i>form</i> cek progres	1
		Integrasi dengan <i>database</i> servis	2
		Implementasi <i>backend</i> cek progres ( <i>query</i> berdasarkan plat kendaraan)	2
		<i>Testing</i> fitur cek progres	1
4	Manajemen Entri Servis (Admin)	Desain UI Halaman Manajemen entri servis	1
		Implementasi <i>backend</i> untuk menampilkan daftar servis	2
		CRUD entri servis (tambah, ubah status, hapus)	2
		<i>Testing</i> fitur manajemen servis	1
5	<i>Pop-up Chat</i> dengan Bengkel (Pelanggan & Admin)	Desain UI <i>pop-up chat</i> pelanggan	1
		Implementasi <i>backend</i> chat pelanggan-admin	3
		Desain UI halaman balasan chat admin	1
		<i>Testing</i> fitur chat	1

Pada tabel 5, total estimasi waktu pengerjaan *Sprint 2* adalah 18 hari kerja. Selanjutnya, perencanaan pengembangan pada iterasi ketiga ditampilkan pada Tabel 6, mencakup aktivitas lanjutan sesuai dengan kebutuhan *backlog*.

**Tabel 6.** *Sprint Planning 3*

ID	Item Backlog	Task	Estimasi (Hari)
6	Fasilitas & Layanan Bengkel (Pelanggan)	Desain UI halaman fasilitas & layanan	1
7	Lokasi dan Kontak Bengkel (Pelanggan)	Desain UI halaman lokasi dan kontak bengkel	1
		Integrasi dengan <i>database</i>	1
		Implementasi <i>backend</i> untuk menampilkan lokasi dan kontak bengkel	2
		<i>Testing</i> fitur kontak dan lokasi bengkel	1
8	Manajemen Lokasi & Kontak Bengkel (Admin)	Desain UI halaman manajemen lokasi	1
		Implementasi <i>backend</i> Manajemen Informasi lokasi	2
		<i>Testing</i> Fitur manajemen Lokasi	1
		Desain UI halaman manajemen kontak	1
		Implementasi <i>backend</i> Manajemen Informasi Kontak	2
		<i>Testing</i> Fitur manajemen Kontak	1
9	<i>Feedback &amp; Rating</i> (Pelanggan & Admin)	Desain UI halaman <i>feedback</i> pelanggan	1
		Desain UI halaman manajemen <i>feedback</i> pelanggan	1
		Integrasi manajemen <i>feedback</i> (CRUD) untuk admin	2
		Implementasi <i>backend</i> pengiriman <i>feedback</i>	2
		<i>Testing</i> fitur manajemen <i>feedback</i>	1
10	Otomatisasi Estimasi Kunjungan Kembali Bengkel (Pelanggan)	Persiapan Data model prediksi estimasi	1
		Implementasi <i>Backend</i> Otomatisasi Estimasi Kunjungan Kembali Bengkel	4
		<i>Testing</i> fitur Otomatisasi Estimasi Kunjungan Kembali Bengkel	1
11	Laporan Keuangan Bengkel (Admin)	Desain UI laporan keuangan (pemasukan dan pengeluaran)	1
		Implementasi <i>backend</i> laporan keuangan	1

12	Otomatisasi Pencatatan Keuangan	Implementasi otomatisasi penambahan transaksi saat servis selesai	2
----	---------------------------------	---	---

Pada tabel 6, total estimasi waktu pengerjaan *Sprint* 3 adalah 31 hari kerja. Terakhir iterasi keempat, proses pengembangan sistem tahap akhir ditampilkan, yang dapat dilihat pada Tabel 7.

**Tabel 7.** Sprint Planning 4

ID	Item Backlog	Task	Estimasi (Hari)
13	CRUD Data Keuangan	Desain UI perubahan & penghapusan data keuangan	1
		Implementasi <i>backend</i> perubahan keuangan	3
		Testing Fitur CRUD Data Keuangan	1
14	Persiapan Akhir & Deployment	Perbaiki dan <i>Testing</i> secara menyeluruh	2
		Pull file project ke Layanan Hosting	2
		Deploy Web ke Server	2
		Dokumentasi Proyek	1

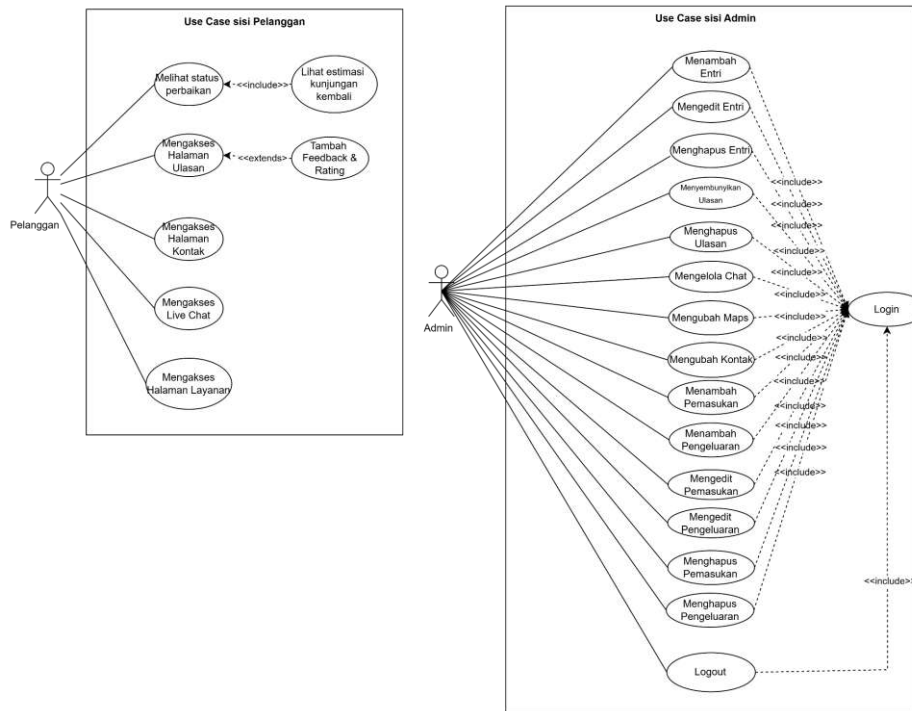
Tabel 7 menunjukkan bahwa estimasi waktu pengerjaan untuk *Sprint* 4 adalah 12 hari kerja. Dengan demikian, total estimasi waktu dari keseluruhan aktivitas dalam empat *Sprint* mencapai 72 hari kerja. Setiap anggota tim dilibatkan secara aktif dalam perencanaan, dengan pembagian tugas yang disesuaikan berdasarkan keahlian masing-masing. Proses *Sprint Planning* ini tidak hanya memastikan pemanfaatan sumber daya secara optimal, tetapi juga menjadi wadah untuk menyamakan persepsi tim, mengantisipasi kendala sejak awal, serta menyusun langkah kerja yang realistis dan terstruktur.

### 3.4 Hasil Sprint

Hasil pengembangan sistem disusun berdasarkan *product backlog* dan *sprint planning* yang telah dirancang di awal. Setiap *sprint* dijalankan dengan pendekatan iteratif dan kolaboratif melalui *Daily Scrum* untuk memastikan kemajuan harian dan mengatasi hambatan secara langsung. Setelah masing-masing *sprint* selesai, dilakukan *Sprint Review* untuk meninjau hasil implementasi serta *Sprint Retrospective* sebagai evaluasi proses kerja tim. Pendekatan ini memastikan bahwa setiap fitur dikembangkan secara bertahap, terukur, dan sesuai dengan kebutuhan pengguna. Dokumentasi hasil dari tiap *sprint* disajikan pada bagian berikut sebagai bentuk pencapaian dalam proses pengembangan.

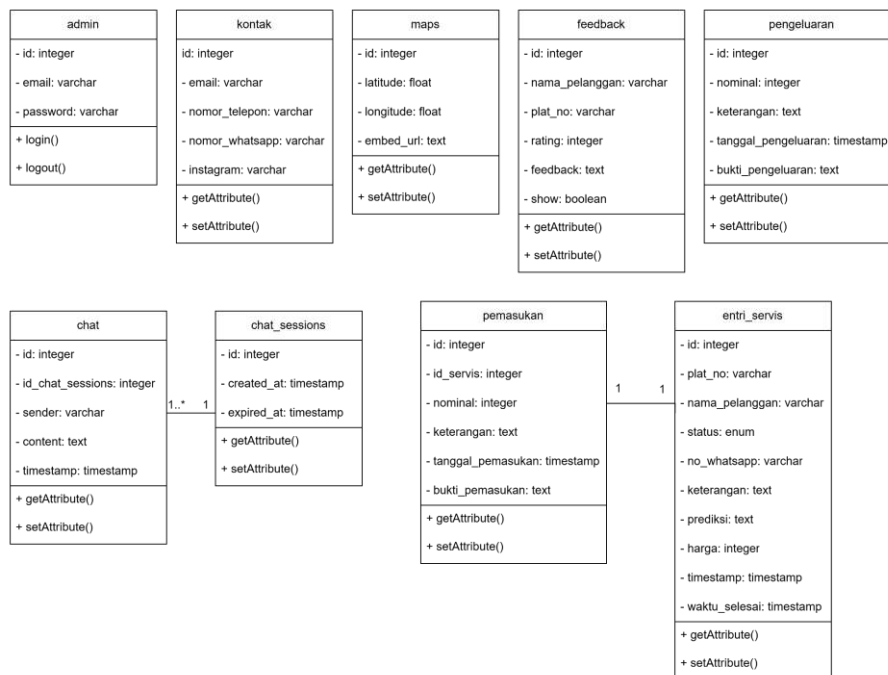
#### 3.4.1 Sprint 1

Pada tahapan ini, menghasilkan dua *item backlog*, yaitu gambaran sistem dan pengembangan fitur *login*. Hasil yang diperoleh meliputi dokumen *use case* diagram yang disajikan pada Gambar 2.



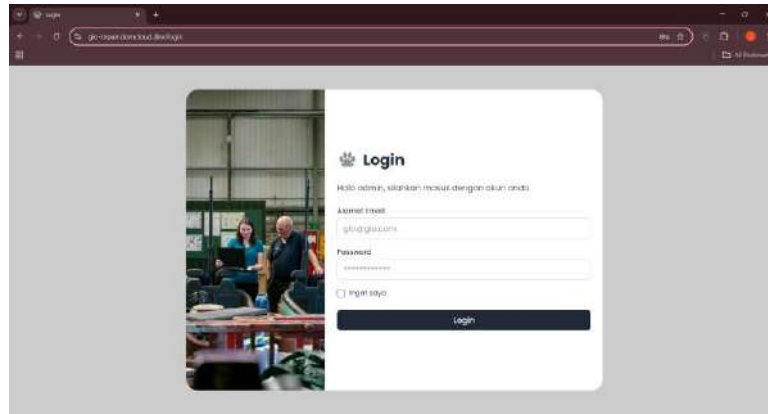
Gambar 2. Use Case Diagram

Dari use case diagram yang telah disusun, dapat diidentifikasi kebutuhan fungsional sistem dan interaksi antara aktor dengan fitur yang tersedia. Informasi tersebut kemudian digunakan sebagai dasar untuk menyusun class diagram sebagai pelengkap, guna menggambarkan struktur data dan hubungan antar entitas dalam sistem. Class diagram yang dihasilkan disajikan pada Gambar 3.



Gambar 3. Class Diagram

Setelah dokumen disusun, selanjutnya proses implementasi halaman *login* admin dilakukan pada *sprint* 1. Berikut merupakan tampilan halaman *login* admin yang disajikan pada Gambar 4.



Gambar 4. Halaman *Login Admin*

### 3.4.2 *Sprint 2, Sprint 3, dan Sprint 4*

Pelaksanaan *sprint* dalam pengembangan sistem dilakukan secara bertahap melalui implementasi kode program untuk merealisasikan fungsionalitas sesuai dengan rancangan yang telah disusun sebelumnya. Seluruh aktivitas *sprint* mengikuti kerangka kerja *Scrum* dan mengacu pada *Definition of Done (DoD)*. Hasil dari setiap *sprint* diwujudkan dalam bentuk demonstrasi fitur yang telah dikembangkan dan diuji. Untuk mendokumentasikan dan memvalidasi proses implementasi, hasil dari masing-masing *sprint* disajikan pada bagian berikut.

a. Halaman Beranda

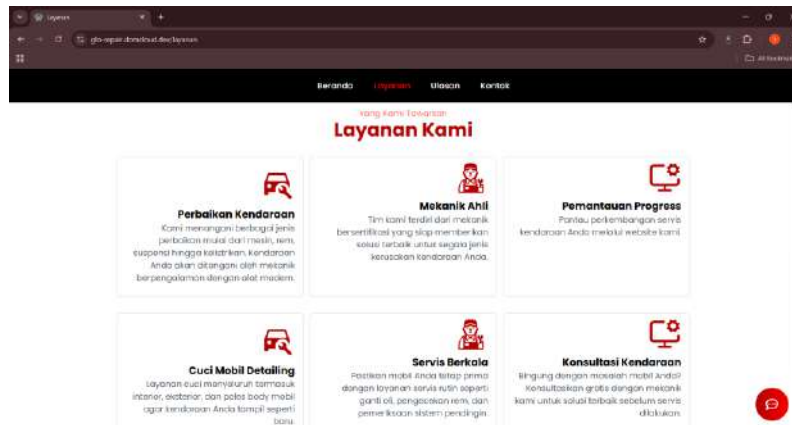
Halaman beranda menampilkan informasi umum mengenai layanan bengkel serta menyediakan fitur untuk memantau status perbaikan secara *real-time*. Di halaman beranda ini juga terdapat *form* cek progres, yang memungkinkan pengguna untuk menginput nomor plat kendaraan guna memantau status perbaikan. Fitur ini merupakan realisasi dari *Sprint 2* dan dapat dilihat pada Gambar 5.



Gambar 5. Halaman Beranda

b. Halaman Layanan

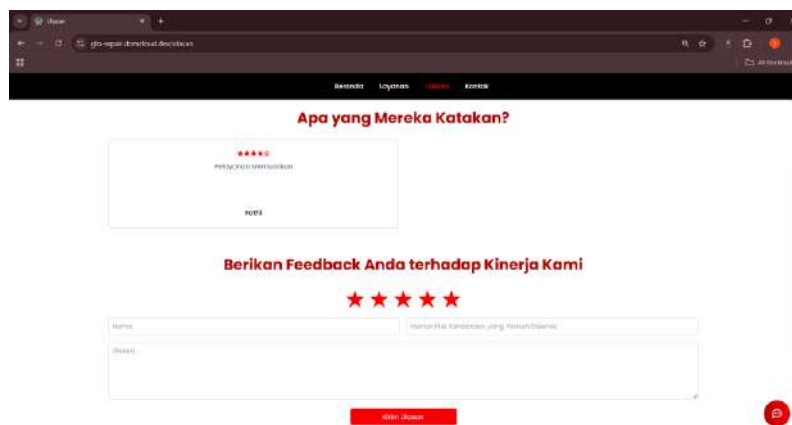
Halaman layanan menampilkan informasi mengenai berbagai layanan yang tersedia di bengkel. Halaman ini dirancang pada tahapan *sprint 3* dan dapat dilihat pada Gambar 6.



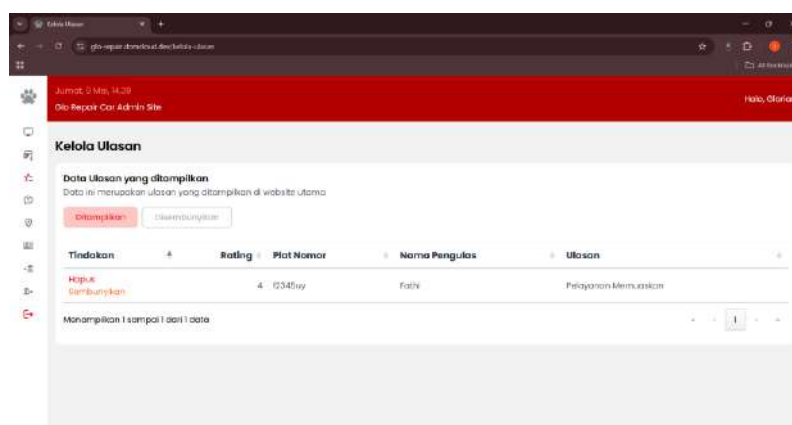
Gambar 6. Halaman Layanan Pelanggan

c. Halaman Ulasan

Halaman Ulasan ini terdiri dari dua bagian, yaitu untuk pelanggan dan admin. Pada bagian pelanggan, pengguna dapat melihat ulasan dan mengisi *form feedback* terkait layanan bengkel, yang ditampilkan pada Gambar 7. Sedangkan pada bagian admin, fitur ini memungkinkan pengelolaan ulasan, seperti menghapus atau menyembunyikan ulasan yang tidak sesuai, yang ditampilkan pada Gambar 8. Fitur ini merupakan realisasi dari *Sprint 3*, dengan *task Feedback & Rating* (Pelanggan & Admin).



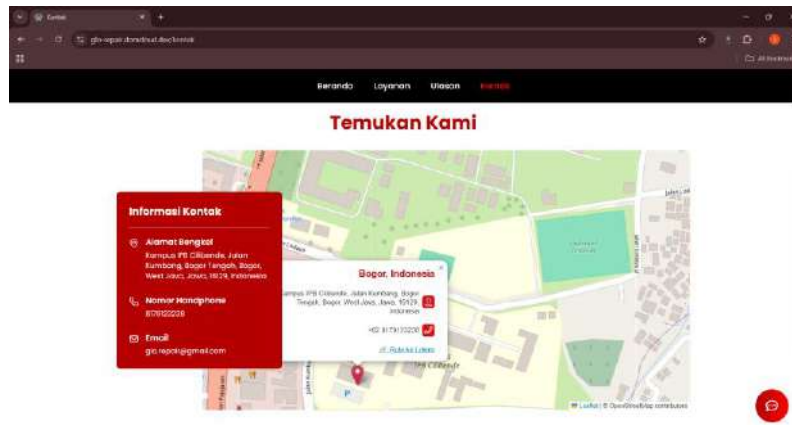
Gambar 7. Halaman Ulasan



Gambar 8. Halaman Kelola Ulasan Admin

d. Halaman Kontak dan Lokasi Bengkel

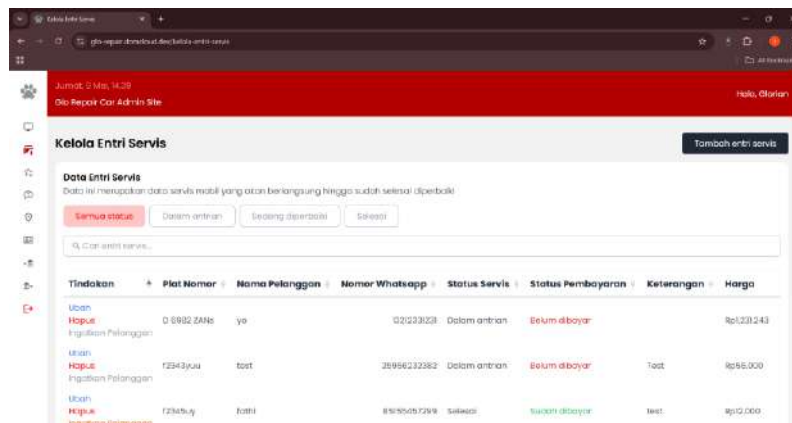
Halaman Kontak dan Lokasi Bengkel menampilkan informasi kontak yang dapat dihubungi serta lokasi bengkel. Fitur ini merupakan realisasi dari *Sprint 3* dan dapat dilihat pada Gambar 9.



Gambar 9. Halaman Kontak

e. Halaman Kelola Entri Servis

Halaman Kelola Entri Servis merupakan fitur yang digunakan oleh admin untuk menambahkan, mengedit, dan menghapus data servis yang terkait dengan perbaikan kendaraan. Admin juga dapat mengedit progres kendaraan, yang terhubung dengan fitur cek progres untuk memantau status perbaikan secara *real-time*. Setelah servis dinyatakan selesai, sistem akan secara otomatis mencatat data ke dalam pemasukan dan mengirim notifikasi kepada pelanggan sebagai pengingat kunjungan kembali. Fitur ini merupakan realisasi dari *Sprint 2* dan *Sprint 3*, dan dapat dilihat pada Gambar 10.



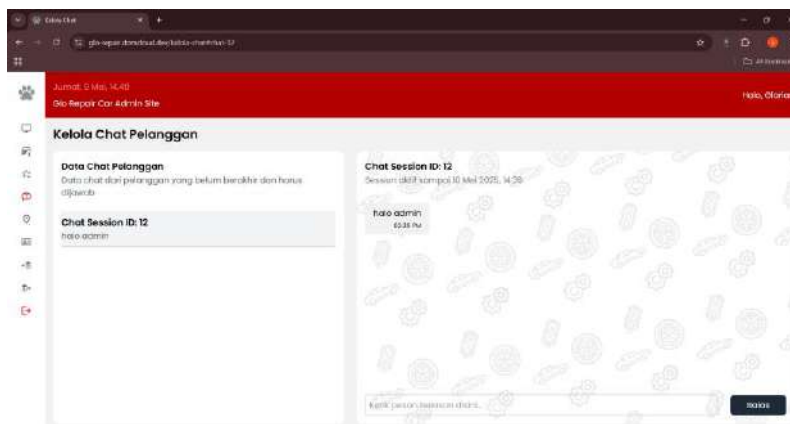
Gambar 10. Halaman Kelola Entri Servis

f. Halaman Chat

Halaman *Chat* terdiri dari dua bagian, yaitu *pop-up chat* untuk pelanggan yang digunakan untuk menanyakan status servis yang dapat dilihat pada Gambar 11, dan halaman kelola *chat* untuk admin yang menampilkan sesi *chat* masuk dan membalas *chat* pelanggan seperti pada Gambar 12. Fitur ini direalisasikan pada *Sprint 3* melalui task "Pop-up Chat dengan Bengkel (Pelanggan & Admin)".



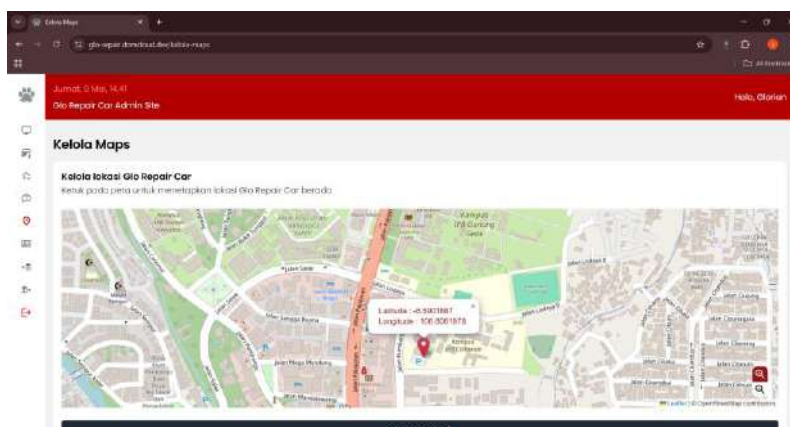
Gambar 11. Halaman Chat



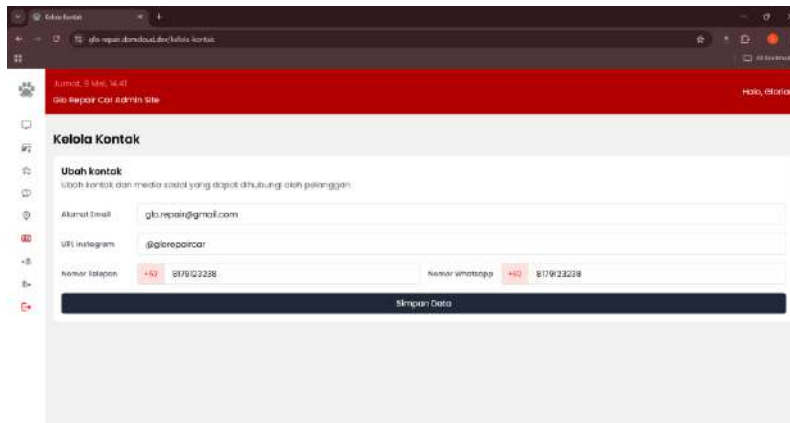
Gambar 12. Halaman Kelola Chat Pelanggan

g. Halaman Manajemen Lokasi dan Kontak Bengkel

Halaman Manajemen Lokasi dan Bengkel memungkinkan admin untuk mengedit informasi lokasi bengkel. Halaman Kelola *Maps*, yang menampilkan peta lokasi, dapat dilihat pada Gambar 13, sedangkan Kelola Kontak, yang menampilkan informasi kontak bengkel, ditampilkan pada Gambar 14. Fitur ini merupakan realisasi dari *Sprint 3*, dengan tugas Manajemen Lokasi & Kontak Bengkel (Admin).



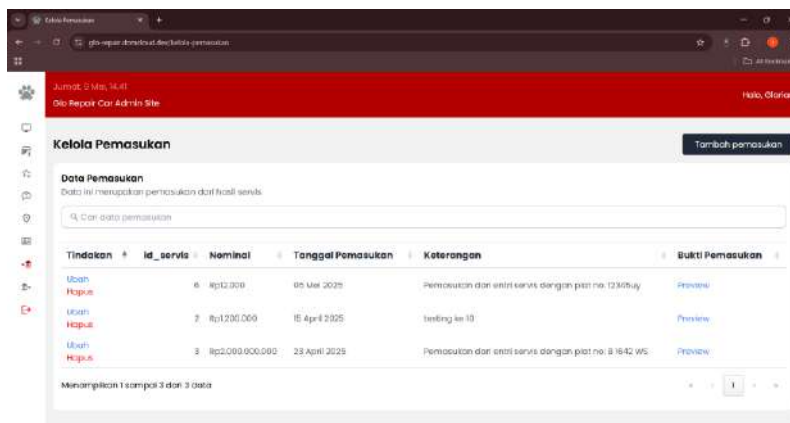
Gambar 13. Halaman Kelola Maps



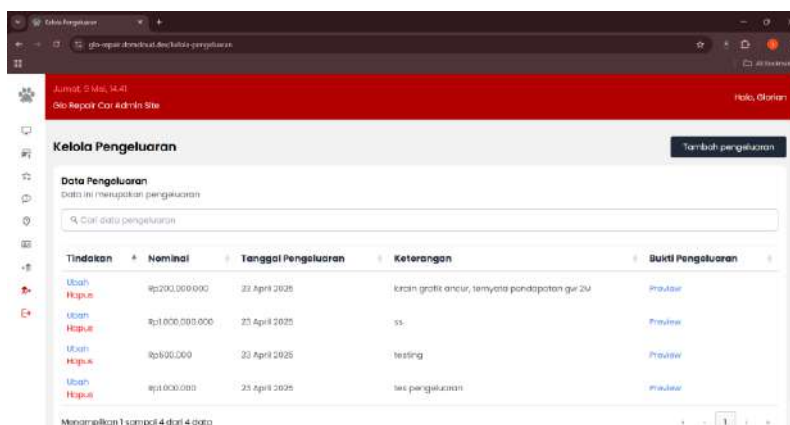
Gambar 14. Halaman Kelola Kontak

h. Halaman Kelola Pemasukan & Pengeluaran

Halaman Kelola Pemasukan & Pengeluaran terdiri dari dua bagian terpisah, yaitu pemasukan yang ditampilkan pada Gambar 15 dan pengeluaran pada Gambar 16. Fitur ini merupakan realisasi dari *Sprint* 3 dan *Sprint* 4. Pada bagian pemasukan, data akan tercatat secara otomatis ketika suatu layanan servis dinyatakan selesai. Selain itu, baik pada pemasukan maupun pengeluaran, admin memiliki akses untuk menambahkan, mengedit, dan menghapus data sesuai kebutuhan.



Gambar 15. Halaman Kelola Pemasukan



Gambar 16. Halaman Kelola Pengeluaran

### 3.5 Pengujian

Setelah penyelesaian setiap *sprint*, dilakukan proses pengujian untuk memastikan bahwa fitur-fitur yang telah dikembangkan berfungsi sesuai dengan spesifikasi dan kebutuhan pengguna. Pengujian ini menjadi bagian dari *Definition of Done* (DoD) dan bertujuan untuk memverifikasi kualitas hasil *sprint* secara menyeluruh. Metode

yang digunakan adalah *Black Box Testing*, yang mengevaluasi sistem dari sisi pengguna berdasarkan masukan dan keluaran tanpa memeriksa struktur internal program. Hasil pengujian disajikan pada bagian ini sebagai bentuk validasi terhadap fitur-fitur yang telah direalisasikan dalam *sprint* sebelumnya. Komponen pengujian disajikan pada Tabel 7.

**Tabel 7.** Pengujian Sistem Manajemen Bengkel

Komponen Uji	Teknik Pengujian	Hasil Pengujian
Status Perbaikan	<i>Black Box Testing</i>	Berhasil
Ulasan Pelanggan	<i>Black Box Testing</i>	Berhasil
Estimasi Kunjungan Kembali	<i>Black Box Testing</i>	Berhasil
Mengakses <i>Live Chat</i>	<i>Black Box Testing</i>	Berhasil
<i>Login Admin</i>	<i>Black Box Testing</i>	Berhasil
Menambah Entri	<i>Black Box Testing</i>	Berhasil
Mengedit Entri	<i>Black Box Testing</i>	Berhasil
Menghapus Entri	<i>Black Box Testing</i>	Berhasil
Menyembunyikan Ulasan	<i>Black Box Testing</i>	Berhasil
Mengelola <i>Chat</i>	<i>Black Box Testing</i>	Berhasil
Mengubah <i>Maps</i>	<i>Black Box Testing</i>	Berhasil
Mengubah Kontak	<i>Black Box Testing</i>	Berhasil
Menambah Pemasukan	<i>Black Box Testing</i>	Berhasil
Menambah Pengeluaran	<i>Black Box Testing</i>	Berhasil
Mengedit Pemasukan	<i>Black Box Testing</i>	Berhasil
Mengedit Pengeluaran	<i>Black Box Testing</i>	Berhasil
Menghapus Pemasukan	<i>Black Box Testing</i>	Berhasil
Menghapus Pengeluaran	<i>Black Box Testing</i>	Berhasil

Berdasarkan Tabel 7, hasil pengujian menunjukkan bahwa seluruh fitur dari setiap *task* pada masing-masing *sprint* telah berjalan dengan baik tanpa kendala, dan sesuai dengan spesifikasi yang telah ditetapkan dalam perancangan sistem.

Secara keseluruhan, penerapan metodologi *Agile Scrum* pada pengembangan sistem informasi bengkel otomotif menunjukkan efektivitas yang tinggi dalam menyelesaikan kebutuhan fungsional secara bertahap dan terstruktur. Dengan siklus *sprint* yang berulang, tim pengembang dapat menyesuaikan fitur sistem berdasarkan masukan pengguna, sehingga solusi yang dihasilkan bersifat adaptif terhadap dinamika kebutuhan yang berubah. Temuan ini selaras dengan permasalahan awal yang diidentifikasi dalam penelitian, yaitu rendahnya efisiensi dan fleksibilitas sistem manual, serta menunjukkan bahwa pendekatan metodologi *Agile Scrum* dalam pengembangan sistem mampu mengatasi tantangan tersebut secara efektif dan adaptif.

#### 4. KESIMPULAN

Berdasarkan hasil pengembangan yang telah dilakukan, dapat disimpulkan bahwa penerapan metodologi *Agile Scrum* pada sistem informasi bengkel otomotif memberikan hasil yang efektif dan adaptif dalam menyelesaikan permasalahan manajemen layanan bengkel konvensional. Sistem yang dibangun berhasil mengintegrasikan fitur-fitur penting seperti autentikasi pengguna, manajemen entri servis, pelacakan status kendaraan, sistem ulasan pelanggan, *pop-up chat* untuk komunikasi *real-time* antara pelanggan dan admin, serta pencatatan laporan

keuangan. Seluruh fitur ini dikembangkan secara bertahap melalui pendekatan sprint yang terbagi menjadi empat tahap, dengan aktivitas perencanaan, implementasi, dan evaluasi yang dilakukan secara berkala.

Keunggulan utama dari penerapan *Agile Scrum* terlihat pada fleksibilitasnya dalam merespons kebutuhan pengguna yang dapat berubah sewaktu-waktu, serta adanya evaluasi dan umpan balik pada setiap akhir *sprint* yang mendorong perbaikan berkelanjutan. Kolaborasi antara tim pengembang, yang terdiri dari *Scrum Master*, *Product Owner*, dan *Developer*, juga terbukti berjalan efektif dengan pembagian peran yang jelas serta pembuatan *backlog* dan *user story* yang terstruktur. Setiap tahapan pengembangan didokumentasikan dengan baik, termasuk perencanaan *backlog*, desain antarmuka, hingga pengujian fungsional sistem. Pengujian sistem dengan metode *Black Box Testing* menunjukkan bahwa seluruh fitur telah berjalan sesuai spesifikasi tanpa ditemukan *error* yang signifikan. Selain meningkatkan efisiensi dan akurasi pencatatan data, sistem ini juga berhasil memberikan kemudahan akses informasi bagi pelanggan serta meningkatkan transparansi layanan bengkel. Dengan demikian, pengembangan sistem informasi bengkel berbasis web menggunakan metodologi *Agile Scrum* tidak hanya mampu meningkatkan kinerja operasional *internal* bengkel, tetapi juga memperkuat kualitas layanan yang diberikan kepada pelanggan. Penerapan metode ini juga dapat dijadikan model dalam pengembangan sistem serupa di sektor lain yang membutuhkan kecepatan adaptasi dan kolaborasi berkelanjutan.

## REFERENCES

- [1] I. Maulidya *et al.*, 'Pengembangan Sistem Pencatatan Pesanan Berbasis Website dengan Pendekatan Equivalence Partitioning pada Tahap Pengujian Black Box', *BINA INSANI ICT J.*, vol. 11, no. 2, p. 154, Dec. 2024, doi: 10.51211/biict.v11i2.3233.
- [2] D. Humaedi, A. P. Putra, A. Saifudin, and Yulianti, 'Pengembangan Aplikasi Rental Mobil Berbasis Web Menggunakan Model Agile di CV. Endang Jaya', *J. Inform. Univ. Pamulang*, vol. 6, no. 3, 2021.
- [3] W. Mulyana, M. Ridhwan, C. N. Y. Sari, and L. Tobing, 'SISTEM INFORMASI MANAJEMEN KEUANGAN BERBASIS WEB BENGKEL MOBIL CV. ANEKA BINTANG SERVICE'.
- [4] C. A. Gemawaty and Y. Yuliani, 'Rancangan Sistem Informasi Penjualan Suku Cadang dan Pelayanan pada Bengkel Hiba Mobil', *remik*, vol. 7, no. 1, pp. 123–133, Jan. 2023, doi: 10.33395/remik.v7i1.11963.
- [5] M. F. Aji and S. Hidayat, 'Pengembangan Business Intelligence pada Sistem Informasi Bengkel (Studi Kasus Bengkel Ban Motor XYZ)'.
- [6] A. Y. Pratama, N. Anwar, A. Ichwani, and Hendry Gunawan, 'Pengembangan Sistem Pemesanan dan Konsultasi Bengkel Bubut Motor Berbasis Scrum untuk Efisiensi Layanan', *IKRA-ITH Inform. J. Komput. Dan Inform.*, vol. 9, no. 1, 2025.
- [7] F. Yunita, B. Rianto, and A. Andilau, 'SISTEM INFORMASI BENGKEL BERBASIS WEB STUDI KASUS KAKELLA MOTOR DI TEMBILAHAN', *Selodang Mayang J. Ilm. Badan Perenc. Pembang. Drh. Kabupaten Indragiri Hilir*, vol. 8, no. 2, pp. 154–165, Aug. 2022, doi: 10.47521/selodangmayang.v8i2.260.
- [8] D. Ningsih, N. T. Rasyid, A. A. Rosdianti, K. Ikhsan, and W. Abdullah, 'Perancangan Sistem Reservasi pada Bengkel Bimo Motor Berbasis', 2023.
- [9] Z. Ramadhan, 'Mengoptimalkan Pengembangan Aplikasi Mobile Melalui Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, Mobile-D, Agile, RAD)', *SUBMIT J. Ilm. Teknol. Infomasi Dan Sains*, vol. 3, no. 2, pp. 13–19, Dec. 2023, doi: 10.36815/submit.v3i2.2993.
- [10] N. Ahmad *et al.*, *ANALISA & PERANCANGAN SISTEM INFORMASI BERORIENTASI OBJEK*. Penerbit Widina, 2022. [Online]. Available: <https://books.google.co.id/books?id=wSSFEAAAQBAJ>
- [11] S. Khoirunnisa, M. A. E. Fatih, and B. Sadjiwo, 'Perancangan Sistem Informasi Dan Pendaftaran Online Di SMK IQRO Berbasis Web Dengan Metode Agile', vol. 2, no. 1, 2024.
- [12] H. Handayani, A. M. Ayulya, K. U. Faizah, D. Wulan, and M. F. Rozan, 'Perancangan Sistem Informasi Inventory Barang Berbasis Web Menggunakan Metode Agile Software Development', *J. Test. Dan Implementasi Sist. Inf.*, vol. 1, no. 1, pp. 29–40, Mar. 2023, doi: 10.55583/jtisi.v1i1.324.
- [13] P. N. Aditya and Bagus Mulyawan, 'Perancangan Website Travel Menggunakan Metode Agile', *J. Ilmu Komput. Dan Sist. Inf.*, vol. 13, no. 1, Jan. 2025, doi: 10.24912/jiksi.v13i1.32890.
- [14] F. Hardiansyah, A. Rizal, and I. Purnamasari, 'IMPLEMENTASI METODE AGILE SCRUM DALAM PENGEMBANGAN APLIKASI PEMBELAJARAN OLAHRAGA', *JATI J. Mhs. Tek. Inform.*, vol. 7, no. 2, pp. 1242–1247, Sep. 2023, doi: 10.36040/jati.v7i2.6734.
- [15] R. Noveandini, M. S. Wulandari, and A. Hakim, 'PENERAPAN METODE SCRUM PADA RANCANG BANGUN SISTEM INFORMASI PENJUALAN TOKO SEPATU RABBANI SHOES', *Semin. Nas. Teknol. Inf. Dan Komun. STIK SeNTIK*, vol. 7, no. 1, 2023.
- [16] M. Hilmyansyah, M. Malabay, H. Simorangkir, and Y. Yulhendri, 'Implementasi Metode Scrum Pada Pembangunan Sistem Informasi Monitoring Progress Proyek Berbasis Web (Studi Kasus: PT Quatra Engineering Mandiri)', *Ikraith-Infom.*, vol. 6, no. 3, Nov. 2022, doi: 10.37817/ikraith-informatika.v6i3.2198.
- [17] P. N. K. Wiharadhita, I. G. P. K. Juliharta, and I. G. J. Eka Putra, 'Sistem Informasi Geografis Pemetaan Hasil Penjualan Sales Lapangan Dengan Leaflet.js dan OpenStreetMap', *J. Tekno Kompak*, vol. 17, no. 1, p. 39, Feb. 2023, doi: 10.33365/jtk.v17i1.2110.
- [18] R. Hartanto and D. Mahdiana, 'Implementasi Metodologi Scrum Dalam Pengembangan Vehicle Inspection Mobile Application Divisi CIST PT. Serasi Autoraya', vol. 11, no. 4, 2024.