

Rancang Bangun Sistem Informasi Manajemen Keuangan Menggunakan Teknologi *Stack Mern* Dengan Metode *Prototyping* (Studi Kasus: Royal Tanjung)

1st Rochmat Wahyu Prayogi
Sistem Informasi
Telkom University
Surabaya, Indonesia

rochmatwahyu@student.telkomuniversity.ac.id

2nd Agus Sulistya
Sistem Informasi
Telkom University
Surabaya, Indonesia

sulistya@telkomuniversity.ac.id

3rd Tita Ayu Rospricilia
Sistem Informasi
Telkom University
Surabaya, Indonesia

titaayu@telkomuniversity.ac.id

Abstrak — Sistem informasi manajemen keuangan merupakan kebutuhan penting dalam mendukung pencatatan dan pelaporan data keuangan secara terorganisir. CV Royal Tanjung, perusahaan properti yang berdiri sejak 2020, masih menggunakan pencatatan manual berbasis *Microsoft Excel* yang berisiko menimbulkan kesalahan, kehilangan data, dan lambatnya pemantauan arus kas. Penelitian ini bertujuan merancang sistem manajemen keuangan berbasis web menggunakan teknologi *MERN Stack* (*MongoDB*, *Express.js*, *React.js*, dan *Node.js*) dengan metode *prototyping*. Sistem dirancang untuk mencatat transaksi, menyusun laporan, dan memantau arus kas secara otomatis dan terstruktur. Penerapan metode *prototyping* memungkinkan perancangan sistem yang responsif terhadap kebutuhan pengguna. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan mampu menggantikan proses manual, meningkatkan efisiensi, akurasi, dan transparansi dalam pencatatan keuangan. Sistem ini juga memberikan kemudahan bagi *owner* dalam memantau kondisi keuangan perusahaan secara *real-time*. Evaluasi yang dilakukan menunjukkan sistem diterima dengan baik oleh pengguna dan memberikan dampak positif terhadap pengambilan keputusan keuangan yang lebih tepat dan cepat.

Kata kunci— *MERN*, *Properti*, *Prototipe*, *Sistem Informasi Keuangan*

I. PENDAHULUAN

Dalam sebuah perusahaan, termasuk UMKM, penerapan sistem informasi manajemen keuangan menjadi kebutuhan penting untuk mendukung pencatatan data keuangan secara terorganisir [1]. Sayangnya, banyak pelaku usaha, khususnya skala kecil, masih menganggap pencatatan keuangan sebagai aspek sekunder karena lebih fokus pada kegiatan operasional lain seperti pemasaran dan penjualan. Minimnya pengetahuan akuntansi serta keterbatasan anggaran menjadi alasan utama mereka menghindari pencatatan formal [2]. Padahal, sistem informasi keuangan yang terstruktur mampu meningkatkan transparansi, efisiensi, serta kinerja keuangan secara keseluruhan [3].

CV Royal Tanjung adalah perusahaan properti yang telah beroperasi sejak tahun 2020, dengan lingkup usaha penjualan tanah kavling dan rumah. Selama ini, pencatatan transaksi keuangan perusahaan, seperti pembelian kavling, angsuran pelanggan, dan pengeluaran operasional masih dilakukan secara manual menggunakan *Microsoft Excel*. Proses ini menyebabkan berbagai permasalahan, antara lain keterlambatan informasi keuangan, kesalahan *input* data, kesulitan dalam pemantauan arus kas secara *real-time*, serta kendala dalam pelaporan keuangan yang akurat [4]. Masalah ini diperparah dengan kurangnya integrasi sistem, sehingga menghambat pengambilan keputusan oleh *owner* dan memperbesar potensi kesalahan administratif [5].

Penggunaan sistem informasi manajemen keuangan berbasis web menjadi solusi yang menjanjikan untuk mengatasi kendala tersebut. Sistem berbasis website memungkinkan pencatatan transaksi dilakukan secara otomatis, terstruktur, dan dapat diakses kapan pun secara *real-time*, baik oleh admin keuangan maupun *owner* [6]. Beberapa penelitian sebelumnya juga telah menunjukkan keberhasilan implementasi sistem serupa, seperti dalam penelitian Nur Sarfiah yang merancang aplikasi laporan keuangan untuk UMKM agar lebih mudah diakses melalui perangkat digital. Studi tersebut menunjukkan bahwa sistem digital dapat meningkatkan efisiensi, meminimalkan kesalahan *input*, dan mendukung pengambilan keputusan yang lebih cepat dan akurat.

Penelitian ini bertujuan untuk merancang dan membangun sistem informasi manajemen keuangan berbasis web menggunakan teknologi *MERN Stack* (*MongoDB*, *Express.js*, *React.js*, *Node.js*). Pendekatan *prototyping* digunakan untuk memfasilitasi pengembangan sistem yang responsif terhadap kebutuhan pengguna, dengan memungkinkan adanya *feedback* berkelanjutan selama proses perancangan [7]. *Stack MERN* dipilih karena menawarkan solusi *full-stack* yang konsisten berbasis *JavaScript*, memudahkan iterasi, serta mendukung pengembangan

aplikasi web yang efisien dan fleksibel. Dengan adanya sistem ini, diharapkan CV Royal Tanjung dapat meminimalkan kesalahan pencatatan, mempercepat proses pelaporan, dan memberikan akses informasi keuangan secara transparan dan *real-time* kepada pihak manajemen. Diharapkan juga operasional keuangan perusahaan menjadi lebih efisien, dan terkontrol.

II. KAJIAN TEORI

A. Sistem Informasi

Sistem informasi merupakan penerapan teknologi informasi dan komunikasi dalam perusahaan untuk mengelola data menjadi informasi yang bermanfaat [8]. Sistem ini terdiri dari perangkat keras, perangkat lunak, jaringan, dan basis data, serta berperan penting dalam meningkatkan efisiensi operasional, menyelesaikan masalah, dan mendukung pengambilan keputusan [9].

B. Akuntansi Dan Manajemen Keuangan

Manajemen keuangan mencakup perencanaan hingga pengawasan keuangan untuk meningkatkan efisiensi dan mendukung pengambilan keputusan yang rasional. Menurut Annisa Meilani & Isnri Andriana, manajemen ini penting bagi UMKM karena membantu pemisahan dana, memantau arus kas, dan merancang strategi bisnis. Akuntansi berperan sebagai sistem informasi yang menyediakan data keuangan melalui proses pencatatan dan pelaporan [10], yang menjadi dasar pengambilan keputusan keuangan. Seluruh keputusan investasi dan pendanaan membutuhkan informasi akuntansi [5]. Dalam sistem informasi manajemen keuangan berbasis teknologi, integrasi proses keuangan dan akuntansi penting untuk otomatisasi pencatatan dan pelaporan sesuai prinsip akuntansi.

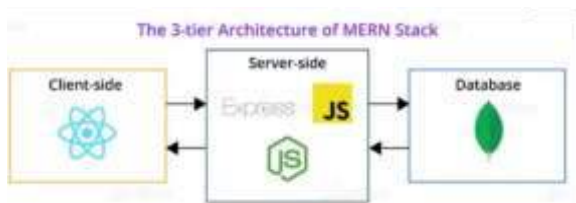
C. Metode Prototype

Metode *Prototype* merupakan model pengembangan perangkat lunak yang fokus pada pembuatan awal sistem (*Prototype*) untuk memperoleh masukan pengguna secara cepat dan iteratif. Proses ini memfasilitasi komunikasi antara pengembang dan pengguna guna meminimalkan kesalahpahaman dan menyempurnakan kebutuhan sistem. Menurut [11], *Prototype* disempurnakan secara bertahap hingga sesuai dengan harapan pengguna. Metode ini sangat efektif untuk proyek dengan kebutuhan yang belum jelas atau berpotensi berubah [7].



GAMBAR 1
(TAHAPAN METODE PROTOTYPE)

D. MERN Stack



GAMBAR 2
(ARSITEKTUR MERN STACK)

MERN adalah singkatan dari *MongoDB*, *Express.js*, *React.js*, dan *Node.js* empat teknologi terintegrasi untuk membangun aplikasi web *full-stack* [12]. *MongoDB* berfungsi sebagai database *NoSQL* berbasis dokumen; *Express.js* adalah *framework backend* minimalis untuk menangani *routing* dan *middleware*; *React.js* digunakan untuk membangun antarmuka pengguna yang interaktif di sisi *frontend*; dan *Node.js* memungkinkan eksekusi *JavaScript* di sisi server. Arsitektur *MERN* mengikuti pola 3-tier: *React.js* sebagai *frontend* untuk UI dan interaksi pengguna, *Express.js* dan *Node.js* sebagai *backend* untuk logika aplikasi dan *API*, serta *MongoDB* sebagai penyimpanan data. *Stack* ini mendukung pengembangan aplikasi yang efisien dan responsif dari ujung ke ujung.

E. MongoDB

MongoDB adalah sistem manajemen basis data *NoSQL* yang menyimpan data dalam format dokumen *JSON-like (BSON)*, memungkinkan fleksibilitas skema antar dokumen dalam satu koleksi [12].

F. Desain Antarmuka Pengguna

User Interface (UI) adalah bagian dari desain sistem yang mengatur tampilan visual dan interaksi pengguna dengan aplikasi. UI bertujuan menciptakan antarmuka yang menarik, mudah dipahami, dan nyaman digunakan [13]. Perkembangan teknologi mendorong munculnya berbagai jenis antarmuka, yang menuntut penyesuaian desain agar pengalaman pengguna tetap optimal [14].

G. Black Box Testing

Black box testing adalah metode pengujian perangkat lunak yang menilai fungsionalitas sistem tanpa melihat struktur internal atau kode sumber [15]. Pengujian dilakukan dari sudut pandang pengguna berdasarkan spesifikasi yang ditetapkan, mencakup pendekatan fungsional, uji integrasi antar modul, dan verifikasi penanganan kesalahan. Kelebihannya, penguji tidak perlu memahami implementasi teknis, cukup fokus pada *output* aplikasi. Hasil pengujian kemudian dianalisis dan didokumentasikan [16].

H. Usability Testing

Usability Testing adalah metode evaluasi untuk menilai kemudahan penggunaan sistem oleh pengguna akhir melalui skenario nyata. Menurut Jakob Nielsen, pengujian ini mencakup lima aspek: *learnability*, *efficiency*, *memorability*, *error*, dan *satisfaction* [17]. Dalam sistem informasi manajemen keuangan, aspek efisiensi menjadi fokus utama, seperti kecepatan navigasi dan akses laporan. Evaluasi dilakukan melalui kuesioner berbasis persentase setelah pengguna mencoba aplikasi [9]. Umpan balik ini membantu pengembang meningkatkan performa dan pengalaman pengguna secara berkelanjutan [18].

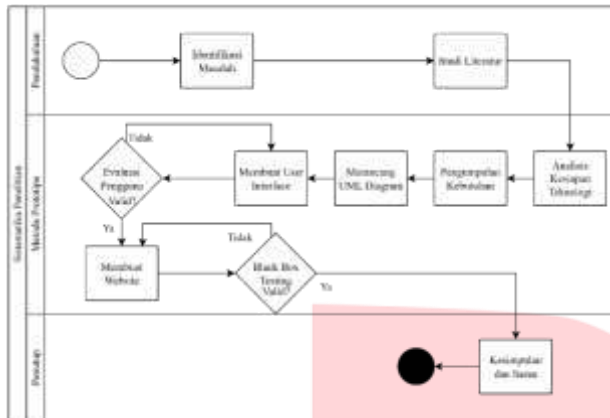
I. UML (Unified Modeling Language)

Unified Modeling Language (UML) adalah bahasa visual standar untuk memodelkan, merancang, dan mendokumentasikan sistem perangkat lunak, khususnya dalam pendekatan berorientasi objek [19]. UML membantu pengembang memahami struktur dan perilaku sistem dari berbagai sudut pandang secara terstruktur.

III. METODE

Bab ini menjelaskan metode yang digunakan dalam merancang dan mengembangkan sistem. Proses dimulai dengan identifikasi masalah melalui observasi dan wawancara, dilanjutkan dengan analisis sistem berjalan,

perancangan antarmuka dan database, hingga implementasi. Pengujian dilakukan dengan metode black-box untuk memastikan fungsionalitas sistem berjalan sesuai harapan. Alur penelitian secara lengkap ditunjukkan pada gambar di bawah ini



GAMBAR 3
(SISTEMATIKA PENELITIAN)

A. Identifikasi Masalah

Pada tahap awal, penelitian dimulai dengan observasi langsung untuk mengidentifikasi permasalahan. Ditemukan bahwa proses pencatatan masih manual, berisiko menimbulkan kesalahan, sulit diakses, tidak terpusat, sering terjadi kelalaian penagihan, dan laporan kurang lengkap. Hal ini menunjukkan perlunya sistem pencatatan yang lebih terstruktur dan otomatis.

B. Studi Literatur

Langkah berikutnya adalah studi literatur untuk menelusuri penelitian terdahulu, mengevaluasi metode yang telah digunakan, serta mengidentifikasi solusi yang relevan. Studi ini juga membantu menemukan celah penelitian yang belum terjawab, sehingga penelitian dapat memberikan kontribusi nyata pada bidang terkait.

C. Analisis Kesiapan Implementasi Teknologi

Agar CV Royal Tanjung siap mengimplementasikan sistem manajemen keuangan berbasis *MERN* stack, dibutuhkan infrastruktur teknologi yang memadai, pelatihan SDM, serta dukungan kebijakan internal dan sosialisasi kepada *Owner* dan admin. Implementasi ini bertujuan untuk meningkatkan pengelolaan keuangan, pelacakan data, dan pengambilan keputusan berbasis data.

D. Pengumpulan Kebutuhan

Tahap ini merupakan tahap awal dari metode *Prototype Model*, yaitu pengumpulan kebutuhan, yang dilakukan melalui wawancara dengan *owner*, admin keuangan, dan admin IT CV Royal Tanjung. Wawancara difokuskan pada identifikasi permasalahan dalam pencatatan dan pengelolaan keuangan, serta kebutuhan sistem yang mendukung penyusunan laporan keuangan. Hasil wawancara menjadi dasar dalam perancangan prototipe sistem yang sesuai dengan kebutuhan perusahaan.

E. Merancang UML Diagram

Setelah wawancara dan identifikasi kebutuhan fungsional, tahap selanjutnya adalah perancangan sistem menggunakan diagram UML seperti *Use Case*, *Activity*, *Sequence*, dan *Class Diagram*. Perancangan ini bertujuan menetapkan struktur dan alur kerja sistem agar sesuai dengan kebutuhan pengguna, serta membantu memahami interaksi komponen dan struktur data yang dibutuhkan.

F. Pembuatan User Interface

Tahap ini berfokus pada perancangan antarmuka pengguna (UI) yang menampilkan elemen dasar seperti tombol, menu, dan kolom informasi. UI membantu menyusun tata letak sistem secara logis dan memudahkan tim pengembang memvisualisasikan alur interaksi pengguna serta memastikan fitur utama mudah diakses.

G. Evaluasi Pengguna

Pengujian UI dilakukan melalui evaluasi pengguna, di mana *owner*, admin keuangan, dan admin IT mencoba prototipe dan memberikan masukan terkait tampilan dan kemudahan penggunaan. Masukan ini digunakan untuk analisis dan revisi sebelum tahap pengembangan aplikasi.

H. Membuat Website

Setelah evaluasi pengguna selesai dengan hasil memuaskan, proses berlanjut ke tahap implementasi menggunakan *Framework MERN* (*MongoDB*, *Express.js*, *React.js*, dan *Node.js*). *MERN* dipilih karena kemampuannya membangun aplikasi web modern yang cepat dan dinamis. *MongoDB* digunakan untuk menyimpan data secara fleksibel, *Express.js* sebagai server ringan, *React.js* untuk UI interaktif, dan *Node.js* untuk server *real-time* yang efisien.

I. Black Box Testing

Evaluasi sistem dilakukan dengan metode *black box* testing untuk memastikan fungsi sistem berjalan sesuai kebutuhan tanpa melihat kode. Proses ini mencakup identifikasi fitur, penyusunan kasus uji, dan pengujian terhadap berbagai skenario. Hasil diuji dibandingkan dengan *output* yang diharapkan untuk menemukan *bug*. Setelah perbaikan, pengujian diulang guna memastikan tidak muncul masalah baru. Evaluasi ini menjamin sistem siap digunakan dengan performa optimal.

IV. HASIL DAN PEMBAHASAN

Pada bab sebelumnya, telah dijelaskan mengenai metode dan prosedur penelitian, termasuk penerapan *Metode Prototype* yang digunakan dalam penelitian ini. Pada bab ini, akan diuraikan secara detail implementasi metodologi penelitian.

A. Pengumpulan Kebutuhan

Dalam pengumpulan data, didapatkan hasil dari wawancara untuk keperluan mengenai kebutuhan sistem dan fitur yang harus dikembangkan untuk mempermudah pencatatan, meminimalkan kesalahan *input*, dan memfasilitasi pemantauan keuangan di CV Royal Tanjung. Tabel berikut merangkum data yang diperlukan untuk setiap aktor.

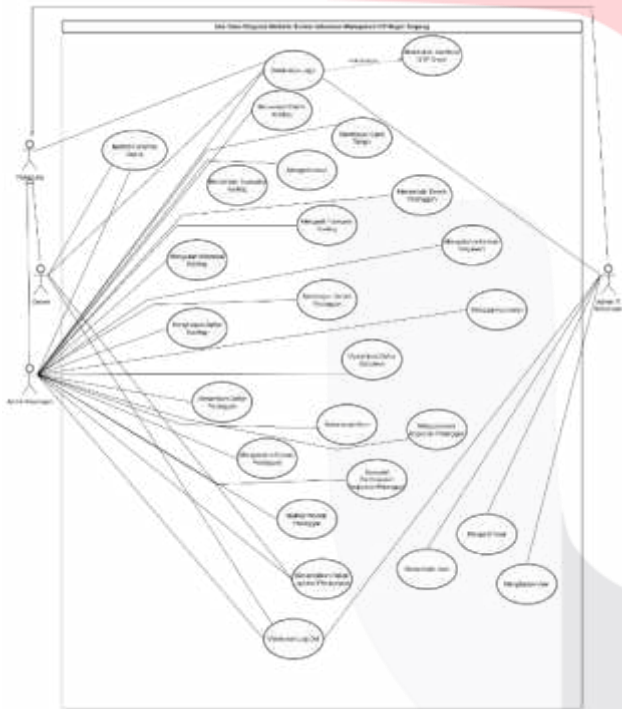
TABEL 1
(KEBUTUHAN PENGGUNA)

Aktor	Data Yang Diperlukan	Keterangan
<i>Owner</i>	Data Laporan Keuangan, Data Ringkasan Keuangan	Bertanggung jawab memantau proses keluar masuk keuangan perusahaan
Admin Keuangan	Data Angsuran, Data Pelanggan, Data Kavling, Data Denda, Data	Bertanggung jawab mengelola proses keluar masuk keuangan perusahaan

	Cash/Tempo, Data Laporan Keuangan, Data Transaksi, dan Data Pengeluaran	
Admin IT	Data Pengguna	Bertanggung jawab mengelola kebutuhan teknologi dan pengguna aplikasi

B. Merancang UML Diagram

Pada tahap ini dilakukan perancangan diagram UML untuk menggambarkan struktur dan alur kerja sistem secara visual. Diagram yang digunakan meliputi *Use Case Diagram*, *Use Case Spesifikasi*, *Class Diagram*, *Activity Diagram*, *Sequence Diagram* dan *Robustness Diagram*. Perancangan UML ini bertujuan untuk memberikan gambaran yang jelas mengenai sistem yang akan dibangun serta menjadi acuan dalam tahap implementasi.



GAMBAR 4
(USE CASE DIAGRAM)

Gambar di atas menunjukkan *Use Case Diagram* dari *Website* Sistem Informasi Manajemen CV Royal Tanjung yang menggambarkan interaksi antara aktor seperti *Owner*, *Admin Keuangan*, *Admin IT*, dan *Pelanggan* dengan sistem. Diagram ini mencakup berbagai fungsi penting seperti login dan verifikasi OTP *Gmail*, pengelolaan akun dan data pengguna, pencatatan transaksi kavling, pengelolaan pembayaran angsuran dan denda, manajemen data karyawan, serta penyajian laporan dan rekapitulasi keuangan. *Use case* diagram ini membantu memberikan gambaran umum mengenai fungsionalitas sistem yang dikembangkan serta peran masing-masing aktor dalam proses pengelolaan informasi.

C. Membuat User Interface

Pada tahap ini dilakukan perancangan dan pembuatan antarmuka pengguna (*user interface*) untuk aplikasi manajemen keuangan CV Royal Tanjung. Tujuan utama dari proses ini adalah untuk menghadirkan tampilan yang sederhana, intuitif, dan mudah digunakan, serta mendukung alur kerja sesuai dengan peran masing-masing pengguna seperti *owner*, admin keuangan, dan admin IT. Gambar berikut menunjukkan prototipe *high-fidelity* dari Sistem Informasi Manajemen Keuangan.



GAMBAR 5
(PROTOTYPE SISTEM INFOMASI MANAJEMEN KEUANGAN)

Setelah dilakukan desain *high fidelity* dikembangkan hingga menjadi prototipe, langkah selanjutnya adalah melakukan evaluasi pengguna untuk memastikan apakah *Prototype* yang dihasilkan telah sesuai dengan kebutuhan pengguna.

TABEL 2
(FEEDBACK USABILITY PENGGUNA)

Desain	<i>Feedback</i> Pengguna	Kesimpulan
Halaman Login	Tampilan sudah sesuai, simpel, dan tidak membingungkan	Diterima
Halaman Dashboard	Sudah cukup lengkap, namun untuk grafik pemasukan bisa ditampilkan per tahun	Perbaikan
Halaman Transaksi	tampilan sudah cukup bagus, tombol pagination ditaruh bawah.	Perbaikan
Halaman Penggajian	tampilan sudah cukup bagus, tombol pagination ditaruh bawah.	Perbaikan
Halaman Laporan Keuangan	Fitur laporan sudah sesuai dengan kebutuhan	Diterima

Setelah evaluasi dilakukan, proses dilanjutkan pada iterasi kedua dengan melakukan penyesuaian dan revisi terhadap *Prototype* berdasarkan masukan yang diperoleh dari pengguna pada tahap evaluasi sebelumnya. Revisi ini bertujuan untuk menyempurnakan tampilan antarmuka dan

alur sistem agar lebih sesuai dengan kebutuhan serta meningkatkan kenyamanan penggunaan.

D. Membuat Website

Pembuatan *website* merupakan tahapan mengubah ide dan rancangan (desain) menjadi sebuah aplikasi perangkat lunak yang fungsional. Setelah selesai dikembangkan, aplikasi kemudian memasuki tahap *deployment* agar dapat diakses oleh pengguna akhir melalui internet.

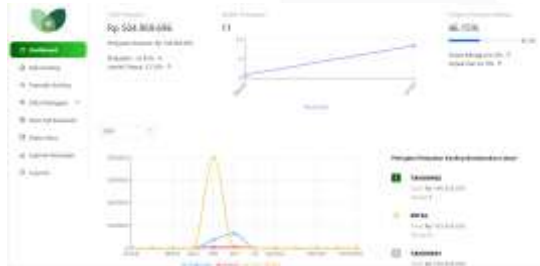
1. Halaman Login



GAMBAR 6
(HALAMAN LOGIN)

Gambar di atas menunjukkan tampilan halaman login. Pada bagian kiri terdapat form login dengan kolom *input* email dan *password*, serta tombol "Log in" berwarna hijau., di mana pengguna memasukkan email dan *password*, lalu menekan tombol Masuk. Setelah diverifikasi, sistem akan mengarahkan pengguna ke halaman utama sesuai perannya.

2. Halaman Dashboard



GAMBAR 7
(HALAMAN DASHBOARD)

Gambar di atas menunjukkan halaman dashboard sistem informasi penjualan kavling yang menyajikan ringkasan metrik penting seperti total penjualan, jumlah pelanggan, dan progres penjualan. Tersedia grafik pemasukan bulanan dari angsuran, denda, dan cash tempo, serta peringkat penjualan kavling berdasarkan lokasi. Desain yang sederhana dan informatif memudahkan admin memantau aktivitas penjualan secara *real-time* dan mendukung pengambilan keputusan yang cepat dan tepat.

3. Halaman Transaksi

No	Tgl Transaksi	Jumlah	Status	Lokasi	Detail
1	2024-01-01	1000000	Selesai	Surabaya	
2	2024-01-02	2000000	Pending	Jakarta	
3	2024-01-03	1500000	Selesai	Bandung	
4	2024-01-04	3000000	Pending	Yogyakarta	
5	2024-01-05	2500000	Selesai	Medan	
6	2024-01-06	1800000	Pending	Semarang	
7	2024-01-07	2200000	Selesai	Palembang	
8	2024-01-08	1200000	Pending	Manado	
9	2024-01-09	2800000	Selesai	Padang	
10	2024-01-10	1600000	Pending	Malang	

GAMBAR 8
(HALAMAN TRANSAKSI)

Gambar diatas menampilkan halaman Transaksi Kavling yang menyajikan data penjualan kavling secara lengkap, termasuk nama pelanggan, detail kavling, tanggal, jenis pembelian, total transaksi, dan status pembayaran. Admin dapat menambahkan atau mengedit data transaksi melalui tombol Tambah dan Edit yang tersedia. Halaman ini dirancang untuk memudahkan pemantauan dan pencatatan transaksi secara *real-time* agar proses penjualan tetap rapi dan terstruktur.

4. Halaman Penggajian

No	Nama Karyawan	Jabatan	Gaji Pokok	Tambahan	Total Gaji
1	Andi Pratomo	Manajemen	5000000	500000	5500000
2	Budi Santia	Manajemen	4500000	450000	4950000
3	Citra Lestari	Manajemen	4000000	400000	4400000
4	Dani Kusuma	Manajemen	3500000	350000	3850000
5	Eva Nurhidayah	Manajemen	3000000	300000	3300000

GAMBAR 9
(HALAMAN PENGGAJIAN)

Gambar diatas menampilkan informasi gaji karyawan perusahaan dalam satu tahun tertentu, dengan rincian, seperti nama karyawan, posisi jabatan, jumlah gaji pokok, tambahan gaji, dan total gaji yang diterima. Pengguna dapat memfilter data berdasarkan tahun tertentu untuk memudahkan pencarian dan analisis data secara lebih spesifik. Selain itu, pengguna juga dapat melakukan aksi seperti menambah data gaji baru atau mengedit data gaji yang sudah ada jika terjadi perubahan atau koreksi.

5. Halaman Laporan Keuangan

No	Tgl Laporan	Kategori	Saldo Awal	Saldo Akhir
1	2024-01-01	Saldo Awal	Rp 1.000.000,00	0,00
2	2024-01-02	Saldo Akhir	0,00	Rp 1.000.000,00
3	2024-01-03	Saldo Awal	0,00	0,00
4	2024-01-04	Saldo Akhir	0,00	Rp 1.000.000,00
5	2024-01-05	Saldo Awal	0,00	0,00
6	2024-01-06	Saldo Akhir	0,00	Rp 1.000.000,00
7	2024-01-07	Saldo Awal	0,00	0,00
8	2024-01-08	Saldo Akhir	0,00	Rp 1.000.000,00
9	2024-01-09	Saldo Awal	0,00	0,00
10	2024-01-10	Saldo Akhir	0,00	Rp 1.000.000,00

GAMBAR 10
(HALAMAN LAPORAN KEUANGAN)

Gambar diatas menampilkan halaman "Neraca Keuangan" yang menyajikan posisi keuangan perusahaan berdasarkan kategori *Asset*, *Liability*, dan *Ekuitas*, lengkap dengan rincian nilai Debit dan Kredit. Pengguna juga dapat memilih jenis laporan lain serta menentukan rentang tanggal, menjadikan halaman ini alat analisis keuangan yang fleksibel dan mendukung pengambilan keputusan.

E. Black Box Testing

Pada tahap ini dilakukan pengujian black-box untuk mengevaluasi apakah fungsionalitas aplikasi telah sesuai dengan ekspektasi pengguna. Pengujian ini bertujuan untuk memastikan bahwa setiap fitur yang dikembangkan dapat digunakan dengan baik tanpa adanya kesalahan pada sisi tampilan maupun proses Pengujian melibatkan beberapa aktor,

yaitu *Owner*, Admin Keuangan, dan Admin IT. Untuk aktor *Owner*, pengujian dilakukan oleh Mas Dani, pengujian untuk aktor Admin Keuangan dilakukan Mbak Febby, dan aktor Admin IT dilakukan oleh Mas Kiki. Hasil pengujian dari ketiga aktor menunjukkan bahwa sistem telah memenuhi kebutuhan. Hal ini menunjukkan bahwa sistem siap untuk diimplementasikan secara penuh.

TABEL 3
(BLACK BOX TESTING)

No.	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil
1	Login	Pengguna mencoba login dengan email dan <i>password</i> yang salah.	Muncul pesan error: "Email atau <i>password</i> salah."	Valid
2		Pengguna mencoba login dengan email dan <i>password</i> yang benar.	Login berhasil, diarahkan ke halaman verifikasi kode.	Valid
3	Verifikasi Kode Login	Pengguna mencoba memasukkan kode verifikasi yang salah.	Muncul pesan error: "Kode verifikasi salah atau kadaluarsa."	Valid
4		Pengguna mencoba memasukkan kode verifikasi yang benar.	Verifikasi berhasil, pengguna diarahkan ke halaman dashboard.	Valid
5	Dashboard	Pengguna masuk ke halaman dashboard	Halaman dashboard ditampilkan dengan data ringkasan yang sesuai.	Valid
6	Laporan Keuangan	Pengguna masuk ke halaman laporan keuangan	Halaman laporan keuangan ditampilkan dengan opsi filter laporan dan rentang tanggal.	Valid
7		Pengguna memilih laporan keuangan	Menampilkan sesuai laporan yang dipilih	Valid

Hasil pengujian menggunakan metode *black box* menunjukkan bahwa seluruh fungsi sistem telah berjalan sesuai dengan spesifikasi yang ditetapkan. Setiap fitur berhasil merespons *input* pengguna dengan *output* yang benar tanpa ditemukan kesalahan fungsional. Validasi ini membuktikan bahwa sistem layak digunakan dan telah memenuhi kebutuhan pengguna secara optimal.

V. KESIMPULAN

Berdasarkan hasil penelitian, sistem informasi manajemen keuangan berbasis website berhasil dikembangkan untuk menggantikan pencatatan manual di CV Royal Tanjung Menganti. Sistem ini dirancang menggunakan pemodelan UML (*Use Case*, *Activity*, *Robustness*, *Sequence*, dan *Class Diagram*) serta prototipe awal dari Figma yang dievaluasi oleh *Owner*, Admin Keuangan, dan Admin IT melalui dua iterasi. Perbaikan pada iterasi pertama divalidasi dan diterima pada iterasi kedua. Sistem dikembangkan dengan teknologi *MERN Stack* dan diuji menggunakan black box testing, dengan hasil menunjukkan seluruh fungsionalitas berjalan baik dan sesuai kebutuhan. Sistem ini dinyatakan berhasil digunakan sebagai solusi pencatatan keuangan perusahaan.

REFERENSI

- [1] A. Rohman, F. Indra Pratama, and M. X. Tengah, "Sistem Informasi Manajemen Keuangan pada Herosoftmedia Berbasis Web," *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 4, no. 1, pp. 72–78, 2022.
- [2] Y. Fitriani, R. Pakpahan, O. Pahlevi, and P. Studi Sistem Informasi, "Sistem Informasi Keuangan Menggunakan Pendekatan Pengembangan Sistem Extreme Programming (XP)," 2024.
- [3] A. Mahira, S. Billah1, M. Fikriyah2, S. Anisa3, and D. Sutriyono4, "Perancangan Sistem Manajemen Keuangan Berbasis Web di CV. Tigana Sukses," *JORAPI: Journal of Research and Publication Innovation*, vol. 1, no. 3, 2023.
- [4] Sunardi, Nihayatut Tasliyah, and Khalidatus Sa'adah, "Manfaat Penerapan Sistem Informasi Akuntansi Terhadap Kualitas Laporan Keuangan Bank Jatim Syariah Cabang Jember," *Mazinda Jurnal Akuntansi, Keuangan, dan Bisnis*, vol. 1, pp. 42–48, 2023, [Online]. Available: <https://journal.ibrahimy.ac.id/index.php/mazinda/index>
- [5] I. Novita, U. Kuningan, H. Wiharno, and D. Djuniardi, "Pengaruh Demografi dan Literasi Keuangan terhadap Praktik Manajemen Keuangan Personal (Studi Kasus pada Guru Pegawai Negeri Sipil di Kabupaten Kuningan)," *Indonesian Journal of Strategic Management*, vol. 5, no. 2, 2022, [Online]. Available: <https://journal.uniku.ac.id/index.php/ijsm>
- [6] H. C. Ambarwati, A. Safira, P. Palevy, and I. A. Kustiwi, "Analisis Pengaruh Sistem Informasi Akuntabilitas Terhadap Kualitas Laporan Keuangan Perusahaan," *Jurnal Ekonomi Manajemen Dan Bisnis*, vol. 1, no. 3, pp. 1–6, 2024, doi: 10.62017/jemb.
- [7] S. Pressman, *Software Engineering: A Practitioner's Approach*. McGraw-Hil, 2010. [Online]. Available: www.mhhe.com/pressman.

- [8] E. P. Primawanti and H. Ali, "Pengaruh Teknologi Informasi, Sistem Informasi Berbasis Web, dan Knowledge Management terhadap Kinerja Karyawan (Literature Review Executive Support System (ESS) for Business)," *Jurnal Ekonomi Manajemen Sistem Informasi (JEMSI)*, vol. 3, no. 3, 2022, doi: 10.31933/jemsi.v3i3.
- [9] M. Maisura, D. Aminatus Saleha, and U. Islam Negeri Ar-Raniry Banda Aceh, "Rancang Bangun E-Roster Berbasis Web Dengan Menerapkan Aspek Usability Testing," *CIRCUIT: Jurnal Ilmiah Pendidikan Teknik Elektro*, vol. 6, no. 2, pp. 196–209, 2022, doi: 10.22373/crc.v6i2.13726.
- [10] A. A. Pramesti and N. Anisyah, "Implementasi Sistem Informasi Akuntansi Terintegrasi untuk Meningkatkan Efisiensi Pelaporan Keuangan," *Jurnal Riset Ilmu Akuntansi*, vol. 3, no. 1, pp. 28–38, 2024, doi: 10.55606/akuntansi.v2i4.1453.
- [11] M. I. Hossain, "Software Development Life Cycle (SDLC) Methodologies for Information Systems Project Management," 2023. [Online]. Available: www.ijfmr.com
- [12] S. Hoque, *Full-Stack React Projects Second Edition*. Packt Publishing Ltd., 2020.
- [13] L. Akbar, A. Usman, and A. Budiman, "Rancang Bangun Desain UI/UX Pada Pembuatan Startup Aplikasi Selfcare Berbasis Website," *Jurnal Ilmu Komputer dan Sistem Informasi (JIRSI)*, vol. 2, no. 1, pp. 158–172, 2023.
- [14] Dario. Calonaci, *Designing user interfaces exploring user interfaces, UI elements, design Prototypes and the Figma UI design tool*. BPB Publications, 2021.
- [15] A. Maspupah, "Literature Review: Advantages and Disadvantages of Black Box and White Box Testing Methods," *Jurnal Techno Nusa Mandiri*, vol. 21, no. 2, pp. 151–162, Sep. 2024, doi: 10.33480/techno.v21i2.5776.
- [16] A. Fahrezi, F. N. Salam, G. M. Ibrahim, R. R. Syaiful, and A. Saifudin, "Pengujian Black Box Testing pada Aplikasi Inventori Barang Berbasis Web di PT. AINO Indonesia," *LOGIC: Jurnal Ilmu Komputer dan Pendidikan*, vol. 1, no. 1, pp. 1–5, 2022, [Online]. Available: <https://journal.mediapublikasi.id/index.php/logic>
- [17] N. J. Suryono, J. E. Samodra, and D. B. Setyohadi, "Pengujian Antarmuka Web Perpustakaan Universitas Atma Jaya Yogyakarta Menggunakan Usability Testing," *Jurnal Informatika Atma Jogja*, vol. 3, no. 2, pp. 34–41, 2022.
- [18] D. Kurniawan and F. Yuamita, "Usability Testing Penggunaan Menu Kartu Hasil Studi Di Website Sistem Informasi Akademik Universitas Teknologi Yogyakarta," *Jurnal Teknologi dan Manajemen Industri Terapan (JTMIT)*, vol. 2, no. 1, pp. 41–52, 2023, [Online]. Available: <https://sia.uty.ac.id/std>.
- [19] L. ,Jeffrey Whitten and D. ,Lonney Bentley, *System Analysis and Design Methods 7th W_copressed*. McGraw-Hil, 2007.