

PERANCANGAN SISTEM PENJUALAN BERBASIS WEB PADA PERUSAHAAN SELANG INDUSTRI DENGAN INTEGRASI OTOMATISASI SALES ORDER

Alfredo Da Gonza, Johaness Fernandes Andry
Sistem Informasi, Universitas Bunda Mulia
Jl. Lodan Raya No.2, Ancol, Jakarta Utara, Indonesia
dagonzaalfredo@gmail.com

ABSTRAK

Perkembangan teknologi telah mendorong perusahaan untuk beralih ke sistem digital guna meningkatkan efisiensi operasional, terutama dalam proses penjualan dan manajemen *sales order*. Permasalahan yang sering muncul dalam sistem konvensional adalah pencatatan transaksi yang masih manual, keterlambatan dalam pemrosesan pesanan, serta kurangnya transparansi antara *dealer* dan *admin*. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan sistem penjualan berbasis *web* dengan integrasi otomatisasi *sales order*, yang memungkinkan proses pemesanan, pengelolaan harga, dan pengiriman *email* konfirmasi berjalan lebih cepat dan efisien. Metode Scrum digunakan dalam pengembangan sistem ini, yang dibagi ke dalam beberapa *sprint*, sehingga setiap fitur dapat dikembangkan secara bertahap sesuai dengan kebutuhan pengguna. Pemodelan menggunakan *class diagram* dan *entity relationship diagram*. Pengujian dilakukan menggunakan *black box testing* untuk memastikan bahwa semua fitur utama, seperti autentikasi pengguna, manajemen produk, sistem *cart*, dan otomatisasi *email*, berfungsi sesuai dengan spesifikasi yang telah ditentukan. Hasil penelitian menunjukkan bahwa sistem ini berhasil mempercepat proses transaksi, mengurangi kesalahan pencatatan, serta meningkatkan kemudahan akses bagi *dealer* dalam melihat produk dan melakukan pemesanan secara *real-time*. Total Waktu sebelum implementasi aplikasi ini berkisar 8 jam hingga 57 jam sedangkan setelah implementasi menjadi berkurang drastis yaitu hanya sekitar 10 menit hingga 76 menit. Dengan pengembangan berkelanjutan, sistem ini diharapkan dapat terus beradaptasi dengan kebutuhan bisnis yang dinamis.

Kata kunci : Sistem Penjualan Berbasis Web, Sales order, Scrum, Email Otomatis, Inventori

1. PENDAHULUAN

Industri e-commerce terus berkembang pesat, sehingga perusahaan dituntut untuk beradaptasi dengan perkembangan teknologi melalui perencanaan strategis sistem informasi/teknologi informasi (SI/TI) guna mendukung proses bisnis [1].

TI tidak hanya mengubah cara kerja, tetapi juga membuka peluang baru untuk meningkatkan efisiensi dan daya saing di pasar global [2].

Maka diperlukan pemikiran baru untuk melakukan modernisasi dalam bisnis di era digital, TI menjadi bagian integral dalam operasional perusahaan karena mampu mengumpulkan, mengolah, dan mendistribusikan informasi secara cepat dan akurat [3].

Sistem penjualan terintegrasi dengan inventori memungkinkan pemantauan stok secara *real-time*, mempercepat pemrosesan pesanan, dan mengoptimalkan ketersediaan barang, sehingga meningkatkan efisiensi pengelolaan barang. Penggunaan sistem berbasis *web* memberikan fleksibilitas akses dari berbagai perangkat, mendukung komunikasi bisnis, mempercepat pengambilan keputusan, serta mengurangi biaya operasional. Penelitian sebelumnya menunjukkan bahwa sistem berbasis *web* meningkatkan efisiensi dan akurasi data inventori dengan menghapus metode manual yang rentan kesalahan dan keterlambatan [4], mempermudah pemantauan penjualan dan persediaan [5].

Penerapan sistem berbasis *web* yang dirancang dengan metode *scrum* dengan kerangka berbasis *agile*, memungkinkan pengembangan perangkat lunak dilakukan secara iteratif melalui *sprint*, yang meningkatkan efisiensi pengembangan dan memastikan sistem dapat beradaptasi dengan kebutuhan pengguna [6].

Sebuah perusahaan yang bergerak di bidang penyediaan selang, *fitting*, dan produk terkait lainnya untuk sektor konstruksi, minyak dan gas, serta transportasi. Perusahaan ini beroperasi di pasar regional asia pasifik. Namun, sistem manajemen inventori yang digunakan saat ini masih bergantung pada *oracle*, dengan keterbatasan akses *admin* yang menyebabkan inefisiensi dalam pengelolaan stok dan pembuatan *sales order*.

Dealer merupakan *reseller* produk sebelum di berikan ke *End Customer*. Proses *sales order* yang digunakan saat ini masih menggunakan memo sebagai permintaan oleh *dealer* yang kemudian harus di proses secara manual oleh *sales* atas persetujuan manajer kemudian baru dikirim *email* ke masing-masing divisi. Produk selang dengan spesifikasi rumit juga dapat menyulitkan pengguna yang kurang berpengalaman dalam memahami produk. Proses manual ini sering kali memperlambat operasional dalam menginput data [7], meningkatkan risiko kesalahan pencatatan, seperti yang umum terjadi dalam sistem manual [8] hingga masalah kehilangan atau kerusakan kertas dokumen [9].

Implementasi sistem berbasis *web* diharapkan dapat memfasilitasi *dealer* dan sales untuk memesan barang secara langsung dan cepat dari berbagai lokasi, memantau ketersediaan stok real-time, serta mengakses informasi produk secara mudah, sehingga mempercepat pemrosesan pesanan dan mengatasi tantangan pengelolaan inventori manual seperti lambatnya proses dan kesalahan pelaporan.

2. TINJAUAN PUSTAKA

2.1. Sistem sales order

Sistem *sales order* adalah dokumen proses internal perusahaan yang mengelola pesanan pelanggan mulai dari pemesanan, detail produk atau jasa, harga, penjualan dan pengiriman. Sistem ini memastikan bahwa setiap pesanan dicatat, diproses, dan dipenuhi sesuai dengan ketentuan yang telah disepakati, sehingga mendukung efisiensi operasional dan komunikasi antar divisi. Implementasi sistem manajemen penjualan dapat mempercepat proses transaksi penjualan [12].

2.2. Penggunaan Metode Scrum

Metode *scrum* adalah kerangka kerja dalam pengembangan perangkat lunak yang menggunakan pendekatan iteratif dan inkremental untuk mengelola proyek secara teratur. Dalam *scrum*, proyek akan dipecah menjadi siklus-siklus pendek yang disebut *sprint*, biasanya berlangsung antara dua hingga empat minggu [13].

Setiap *sprint* mencakup perencanaan, pengembangan, pengujian, dan peninjauan, dengan tujuan menghasilkan peningkatan produk yang dapat diuji dan dievaluasi [14].

2.3. Sistem e-commerce dalam memudahkan transaksi

Sistem *e-commerce* menyediakan tempat interaksi antara penjual dan pembeli secara digital. Dengan adanya *e-commerce*, pelanggan dapat melakukan pembelian kapan saja dan dari mana saja, tanpa harus mengunjungi toko fisik, sehingga meningkatkan efisiensi, kecepatan dan kenyamanan dalam bertransaksi [16].

Implementasi sistem *e-commerce* juga memberikan kemudahan dalam pengelolaan produk, stok, dan pemrosesan pesanan secara otomatis, sehingga mendukung pertumbuhan bisnis di era digital [18].

3. METODE PENELITIAN

3.1. Metode Pengumpulan data

- Observasi: Peneliti melakukan pengamatan langsung terhadap proses bisnis yang sedang berjalan di perusahaan, termasuk alur komunikasi antara *sales*, *admin*, dan *dealer*.
- Wawancara: Wawancara dilakukan secara langsung dengan Senior manajer, *Business Development manager*, *admin* dan *marketing*

yang terlibat dalam proses operasional untuk mendapatkan data yang lebih mendalam.

3.2. Metode Perancangan

Metode perancangan dalam penelitian ini menggunakan pendekatan berbasis pemodelan untuk menggambarkan struktur dan alur kerja sistem yang dikembangkan. Proses bisnis saat ini akan dideskripsikan dan akan dibandingkan dengan proses bisnis setelah adanya aplikasi tersebut. Diagram yang digunakan meliputi *class diagram* untuk memodelkan hubungan antar entitas dalam sistem, serta *Entity Relationship Diagram* (ERD) untuk mendefinisikan struktur database.

3.3. Metode Pengembangan

Metode penelitian ini menggunakan framework *scrum* sebagai pendekatan iteratif dalam pengelolaan proyek perangkat lunak. Langkah-langkah dalam *scrum* adalah sebagai berikut:

- Product backlog refinement*: Identifikasi kebutuhan sistem untuk disusun ke dalam *product backlog*.
- Sprint planning*: Tim memilih item penting dari *product backlog* yang akan dikerjakan selama satu *sprint*.
- Sprint dan Daily scrum*: Pelaksanaan kegiatan *scrum* dengan penetapan cara dan batas waktu pengerjaan, serta *daily scrum* untuk melaporkan perkembangan produk yang dipandu oleh *scrum master*.
- Sprint review*: Evaluasi hasil kerja dan presentasi ke *Stake holder* untuk mendapatkan umpan balik.
- Sprint retrospective*: Analisis kinerja *sprint* untuk mengetahui kelebihan dan area yang perlu ditingkatkan guna perbaikan pengembangan berikutnya.

3.4. Metode Testing

Metode pengujian dalam penelitian ini menggunakan pendekatan *Black Box Testing* untuk memastikan sistem bekerja sesuai dengan spesifikasi dan kebutuhan pengguna, dengan memvalidasi input dan output tanpa memperhatikan struktur kode internal.

4. HASIL DAN PEMBAHASAN

4.1. Proses bisnis saat ini

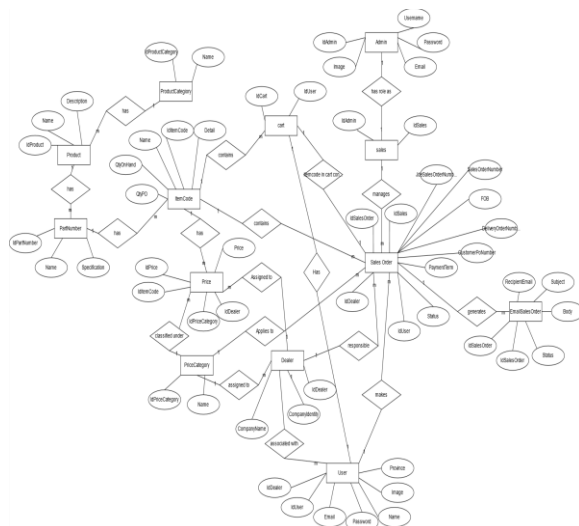
Proses bisnis saat ini dalam sistem penjualan berbasis *dealer* masih mengandalkan prosedur manual, di mana *dealer* harus menghubungi perusahaan atau mengunjungi untuk mengajukan pesanan melalui memo atau *purchase order* (PO). *sales* kemudian melakukan pemeriksaan stok, kemudian pesanan akan dikonfirmasi, lalu diteruskan melalui *email* untuk persetujuan transaksi oleh *sales manager*. Setelah transaksi disetujui, pesanan dimasukkan ke dalam sistem aplikasi (*JDE Oracle* dan *Excel*) secara manual untuk pembuatan *sales order* oleh *sales support*. Selanjutnya, salinan *sales order*

dikirim dengan *email* satu per satu ke divisi keuangan untuk pencatatan invoice dan ke divisi inventori untuk proses persiapan barang. Setelah barang siap, dilakukan pengiriman ke *dealer*. Proses ini masih memerlukan banyak interaksi manual, yang dapat memperlambat alur kerja terutama saat memproses *sales order* dan meningkatkan potensi kesalahan administratif.

4.2. Proses bisnis rancangan

Setelah implementasi sistem penjualan berbasis *web*, proses bisnis mengalami otomatisasi yang signifikan. *dealer* kini dapat melakukan pemesanan langsung melalui platform digital, memilih produk, dan melakukan *checkout* yang secara otomatis menghasilkan *sales order* tanpa perlu menghubungi perusahaan secara manual. Sistem juga secara otomatis mengecek ketersediaan stok, dan *sales manager* dapat langsung menyetujui atau menolak transaksi melalui *dashboard admin*. Setelah pesanan dikonfirmasi, sistem akan mengirimkan *email* otomatis ke tim inventory dan finance, sehingga proses penyimpanan, perakitan barang, serta pembuatan *invoice* menjadi lebih efisien dan cepat. Setelah itu kemudian barang akan langsung dikirimkan ke daerah masing-masing dan dapat dilacak melalui status pada aplikasi yang berubah untuk memberi tahu proses kepada *dealer*. Hal ini mengurangi ketergantungan pada komunikasi manual dan mempercepat penyelesaian transaksi.

4.3. Entity Relationship Diagram untuk hubungan entitas

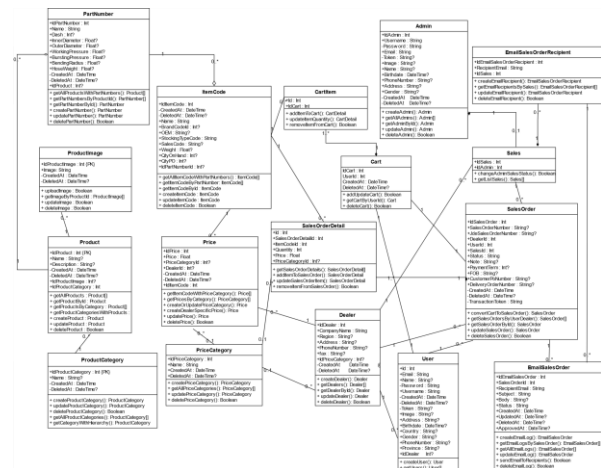


Gambar 1. ERD

ERD pada gambar 1 menggambarkan struktur data dan hubungan antar entitas dalam Sistem Penjualan berbasis *web* untuk *dealer* pada industri selang. Diagram ini menunjukkan bagaimana *admin*, *sales*, *dealer*, *user*, relasi kompleks “*ProductCategory-Product-PartNumber-ItemCode*”, *cart*, dan *sales order* saling terhubung dalam proses

transaksi. Setiap entitas memiliki atribut yang mencerminkan data yang dikelola dalam sistem, serta relasi yang menggambarkan alur bisnis, seperti pengelolaan produk, pemesanan barang oleh *dealer*, *checkout* melalui *cart*, hingga otomatisasi *email* untuk konfirmasi *sales order*.

4.4. Class diagram untuk pemodelan



Gambar 2. Class Diagram

Class diagram pada gambar 2 digunakan untuk memodelkan struktur sistem *sales order* dengan menampilkan entitas, atribut, dan relasi antar kelas. Diagram ini membantu dalam memahami hierarki objek, asosiasi antar kelas, serta bagaimana data diorganisir dalam sistem mulai dari bagaimana sistem inventori bekerja, proses melalui sistem *order*, siapa saja yang bisa mengakses hingga sistem *email sales order* tersebut. Dalam sistem ini, *user* berperan sebagai entitas utama dan pelanggan yang dapat melakukan transaksi dan memiliki hubungan langsung dengan “*Cart*” untuk penampungan sementara serta “*Salesorder*” untuk transaksi.

Sebelum dipesan pelanggan tentu harus memilih produk dengan spesifikasi yang cukup rumit dimana ada relasi “*ProductCategory-Product-PartNumber-ItemCode*”. Tentu saja sebagai produk selang industri pasti memiliki spesifikasi khusus seperti diameter selang yang berbeda (disebut “*PartNumber*”) untuk dibedakan. Setiap “*PartNumber*” juga memiliki produksi pabrikan yang berbeda dalam merek dan label sehingga memiliki turunan yang disebut “*ItemCode*”. Setelah “*User*” memilih maka baru disimpan dalam “*Cart*”, yang berisi “*CartItem*” sebagai representasi “*ItemCode*” (basis data utama untuk hubungan “*Product-PartNumber-ItemCode-Price*”) yang dipilih oleh pengguna. Sedangkan *dealer* berfungsi sebagai perusahaan dimana banyak *user* terkait. Skema harga yang dikendalikan oleh “*PriceCategory*” terikat terhadap setiap “*Dealer*” untuk mengkategorikan setiap harga bagi “*User*”. “*Salesorder*” menyimpan data transaksi dan detail pesanan melalui “*Salesorderdetail*”, yang mencatat “*ItemCode*” dan harga dalam setiap pembelian.

Kemudian “Sales” bertindak sebagai pengelola yang bertugas untuk melakukan *approve* setiap pesanan dari *user* sebelum dikirim ke *email*. Setiap “*ItemCode*” dalam sistem mewakili produk yang tersedia, yang memiliki relasi dengan harga melalui “*Price*” dan juga dapat ditambahkan ke keranjang belanja atau pesanan. Selain itu, sistem juga memiliki “*Emailsalesorder*” yang menangani data kumpulan *order* yang dikirimkan melalui *email*.

4.5. Tahapan Metode Scrum

Tahap ini dilakukan penerapan metode *Scrum* dalam proses pengembangan aplikasi *sales order* otomatis yang terintegrasi dengan inventori ini:

4.5.1. Requirement Product Backlog Aplikasi

Mencaup penentuan modul atau fitur yang diperlukan dan dibagi menjadi dua bagian utama berdasarkan peran pengguna, yaitu halaman *admin* dan halaman *dealer*. Untuk halaman *admin*, modul yang diperlukan meliputi autentikasi *login*, manajemen produk, manajemen harga, manajemen *sales order*, serta manajemen dan otomatisasi *email*, yang memungkinkan *admin* mengelola data transaksi secara komprehensif guna memastikan kelancaran operasional. Sebaliknya, halaman *dealer* difokuskan pada pemesanan dan akses *read-only*, sehingga *dealer* dapat melihat katalog produk, memilih item, mengajukan *sales order*, dan memantau status transaksi serta *email* konfirmasi. Pembagian ini memastikan setiap peran memiliki akses sesuai dengan kebutuhannya, sekaligus terintegrasi dalam satu database yang sama..

4.5.2. Tahap Sprint

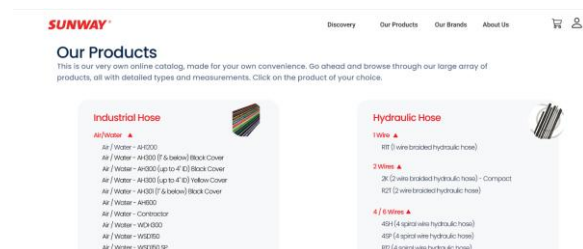
Pengembangan sistem dilakukan dalam enam *sprint* yang mencakup modul untuk *admin* dan *dealer*. *Sprint 1* fokus pada *login* dan manajemen produk *admin*, termasuk desain antarmuka dan fitur CRUD. *Sprint 2* dan *3* mencakup manajemen harga, *sales order*, serta otomatisasi *email* yang dapat mengirim konfirmasi secara otomatis. *Sprint 4* mengembangkan fitur *login dealer* dan katalog produk dengan pencarian. *Sprint 5* mencakup halaman detail produk dan sistem cart untuk pengelolaan item pesanan. *Sprint 6* menyelesaikan proses transaksi *dealer*, termasuk konversi *cart* menjadi *sales order* dan status *email*. Setiap *sprint* mencakup desain antarmuka, pengembangan API, database, dan fitur pendukung otomatisasi sistem.

4.5.3. Daily Scrum

Setiap satu kali *sprint* dilakukan selama seminggu sehingga waktu pengerjaan total kurang lebih 6 minggu. Setiap *Sprint* berfokus pada pengembangan fitur, perancangan *database* dan juga alur sistem menggunakan *API*. *Daily Scrum* rutin dilaksanakan setiap satu kali akhir pada *Sprint* untuk membahas progress, hambatan serta evaluasi untuk solusi.

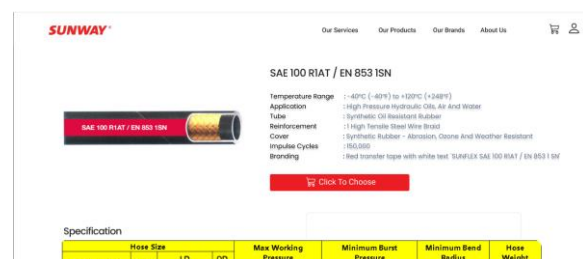
4.5.4. Development

Tahap ini merupakan implementasi pembuatan aplikasi penjualan yang sudah terintegrasi dengan *sales order* dan Inventori. Pengembangan ini menggunakan framework *Nextjs* untuk *frontend*, *ExpressJs* untuk *Backend* dan *PostgreSQL* untuk *Database* penyimpanan data dan informasi. Berikut antarmuka aplikasi penjualan ini:



Gambar 3. Katalog Produk

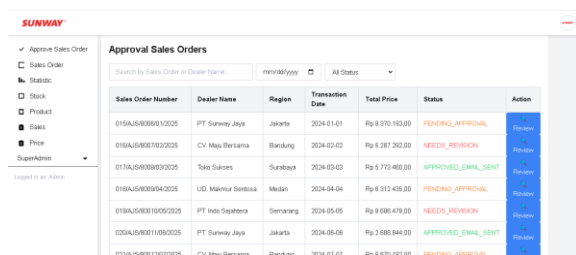
Pada gambar 3 menampilkan halaman katalog produk yang berfungsi sebagai tampilan awal bagi *dealer* untuk melihat berbagai jenis produk yang tersedia. Halaman ini dirancang untuk mempermudah *dealer* dalam menelusuri dan memilih produk berdasarkan kategori produk yang telah dikelompokkan dengan jelas. Pada tampilan ini, terdapat dua kategori utama, yaitu *Industrial Hose* dan *Hydraulic Hose* (Disebut “*Product Category*”), yang masing-masing mencakup berbagai jenis produk. Dalam kategori produk pun memiliki turunan lagi yaitu misalkan pada *Hydraulic Hose* dimana memiliki spesifikasi khusus lagi seperti *1wire*, *2wire* atau *4/6wire* (disebut “*Product*”). *Dealer* dapat mengklik salah satu produk yang tersedia untuk melihat informasi lebih detail terkait spesifikasi teknis, fitur, serta opsi pembelian yang lebih lengkap. Tampilan ini juga dilengkapi dengan navigasi yang memudahkan *dealer* dalam berpindah antar kategori produk serta mengakses informasi lainnya yang dibutuhkan sebelum melakukan pemesanan.



Gambar 4. Detail Produk

Halaman detail produk ini pada gambar 4 menampilkan informasi lengkap mengenai sebuah produk, termasuk spesifikasi teknis seperti rentang suhu, aplikasi, material, dan daya tahan. Pengguna dapat melihat gambar, deskripsi rinci, serta tabel spesifikasi produk secara detail. Tersedia tombol “*Click To Choose*” yang memungkinkan pengguna untuk memilih sub produk (disebut “*Part Number*”) yang kemudian akan bercabang kembali menjadi

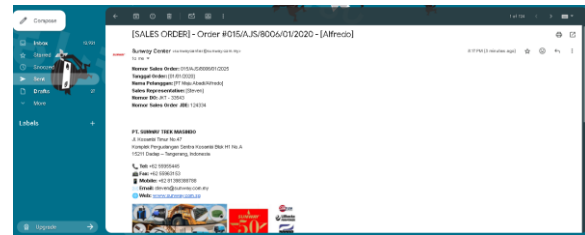
beberapa “*ItemCode*” yang sama tetapi memiliki proses pabrikan dan label yang berbeda (disebut “*ItemCode*”). Setelah pemilihan *PartNumber* dan *ItemCode* maka *user* baru bisa memasukan ke dalam *cart*. Basis utama detail dalam *cart* bukan dari produk tetapi merupakan turunan dari produk yaitu *ItemCode*. Hal ini dilakukan karena kompleksitas spesifikasi dari produk tetapi tetap menjaga halaman sistem agar tetap sederhana. Harga yang diberikan juga merupakan harga yang sudah sesuai dengan yang sudah ditentukan sesuai kategori harga *user* yang sedang melakukan *checkout* (“*User-Price-ItemCode*”). Setelah dilakukan *submit* maka data dari *cart* tersebut akan dikonversi otomatis ke dalam *database admin* untuk segera dilakukan persetujuan oleh manajer yang bersangkutan dengan *user/dealer*.



Gambar 5. Menu approval sales order

Halaman pada gambar 5 merupakan proses dimana *sales* dapat melakukan *approval* terhadap pesanan yang sudah masuk. Jika ada perubahan pada pesanan maka *sales* juga dapat melakukan *review* dan mengubahnya. Proses penjualan yang sudah dilakukan *approve* maka akan berubah statusnya menjadi “*APPROVED*” kemudian akan disimpan di menu

modul manajemen *sales order* agar tetap masuk di riwayat dan dapat diubah kembali statusnya apabila barang sudah siap dikirim atau barang sudah sampai.



Gambar 6. Hasil pengiriman email otomatis

Gambar 6 merupakan proses *email* yang diterima secara otomatis setelah *sales* melakukan *approval* terhadap *order*. Pengiriman *email* pun dilakukan otomatis ke daftar yang sudah ditentukan oleh *sales* yang bersangkutan. *Email* ini juga sudah terformat dengan baik untuk memudahkan organisir. Format pun sudah dibuat secara otomatis oleh sistem sehingga tidak perlu mengetik ataupun merapikan format email kembali.

4.6. Testing Sistem

Metode testing yang digunakan dalam pengujian ini adalah *black box testing*, yang fokus pada validasi output berdasarkan berbagai skenario input tanpa memeriksa kode sumber secara langsung. Pengujian dilakukan langsung melalui tes fungsi pada setiap fitur agar berjalan sesuai harapan.

Setiap pengujian dilakukan berdasarkan tahap *sprint* yang telah direncanakan, dengan hasil yang dibandingkan terhadap harapan sistem.

Tabel 1. Sprint 1

Modul	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan	Tahap Proses
Autentikasi Login admin	Mengisi <i>login</i> dengan salah satu data yang salah	Sistem menampilkan 'Email atau Password Salah!'	Berhasil menampilkan pemberitahuan	Sesuai	Sprint 1
	Mengisi <i>email</i> dan password dengan benar	Sistem menerima <i>login</i> dan mengarahkan ke halaman <i>dashboard</i>	Berhasil mengarahkan ke halaman <i>dashboard</i>	Sesuai	Sprint 1
Manajemen Produk	Menambahkan produk baru di manajemen produk	Sistem menyimpan produk dan menampilkan dalam daftar produk	Produk berhasil ditambahkan dan ditampilkan	Sesuai	Sprint 1
	Mengedit produk yang telah ditambahkan	Sistem memperbarui data produk di daftar produk	Berhasil memperbarui data produk	Sesuai	Sprint 1
	Menghapus produk dari daftar	Produk dihapus dari daftar produk	Produk tidak muncul setelah dihapus	Sesuai	Sprint 1

Pada tabel 1, terlihat bahwa hasil pengujian *sprint 1* menunjukkan bahwa pada modul autentikasi *login admin*, sistem menampilkan pesan “*Email atau Password Salah!*” saat data *login* salah dimasukkan dan mengarahkan pengguna ke *dashboard* saat data benar dimasukkan, sementara pada modul manajemen produk, sistem berhasil menambahkan produk baru,

memperbarui data produk yang ada, dan menghapus produk sesuai dengan spesifikasi, yang semuanya mengonfirmasi bahwa fungsionalitas kedua modul tersebut berjalan optimal sesuai dengan rencana pada *sprint 1*.

Tabel 2. *Sprint 2 hingga sprint 5*

Modul	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan	Tahap Proses
Manajemen Harga	Menambahkan kategori harga baru	Sistem menyimpan kategori harga dan menampilkannya	Kategori harga berhasil ditampilkan	Sesuai	<i>Sprint 2</i>
	Mengedit kategori harga yang sudah ada	Sistem memperbarui data kategori harga	Data kategori harga diperbarui dengan sukses	Sesuai	<i>Sprint 2</i>
	Menghapus kategori harga	Sistem menghapus kategori harga	Kategori harga tidak muncul di daftar setelah dihapus	Sesuai	<i>Sprint 2</i>
	Menambahkan harga baru	Sistem melakukan prioritas terhadap harga khusus dibandingkan harga kategori kemudian menyimpan harga	Sistem berhasil menampilkan harga khusus dibandingkan harga kategori	Sesuai	<i>Sprint 2</i>
	Mengedit harga yang sudah ada	Sistem memperbarui data harga	Data harga diperbarui dengan sukses	Sesuai	<i>Sprint 2</i>
	Menghapus harga	Sistem menghapus harga	Harga tidak muncul di daftar setelah dihapus	Sesuai	<i>Sprint 2</i>
Manajemen <i>sales order</i>	Menambahkan <i>sales order</i> baru	Sistem menyimpan data <i>sales order</i>	<i>sales order</i> berhasil tersimpan	Sesuai	<i>Sprint 2</i>
	Mengedit status <i>sales order</i>	Status berubah sesuai update yang dilakukan	Status berhasil diperbarui	Sesuai	<i>Sprint 2</i>
Manajemen dan Otomatisasi Email	Konversi <i>sales order</i> menjadi PDF/Excel	Sistem menghasilkan file yang bisa diunduh	Berhasil mengonversi dan mengunduh file	Sesuai	<i>Sprint 3</i>
	Mengirim email konfirmasi <i>sales order</i>	Sistem mengirim email ke penerima yang ditentukan	Email berhasil dikirim	Sesuai	<i>Sprint 3</i>
	Melihat status email yang telah dikirim	Status email tampil sesuai dengan pengiriman	Status email sesuai dengan hasil pengiriman	Sesuai	<i>Sprint 3</i>
Autentikasi Login dealer	dealer login dengan email & password benar	dealer berhasil masuk ke sistem	Berhasil login	Sesuai	<i>Sprint 4</i>
	dealer login dengan email atau password salah	Sistem menampilkan 'Email atau Password Salah'	Berhasil menampilkan pemberitahuan	Sesuai	<i>Sprint 4</i>
Katalog Produk	Melihat katalog produk dealer	Semua produk yang tersedia ditampilkan	Katalog berhasil ditampilkan	Sesuai	<i>Sprint 4</i>
	Mencari produk di katalog	Sistem menampilkan hasil pencarian yang sesuai	Produk ditemukan sesuai kata kunci pencarian	Sesuai	<i>Sprint 4</i>
Detail Produk	dealer melihat detail produk	Informasi lengkap tentang produk ditampilkan	Berhasil menampilkan detail produk	Sesuai	<i>Sprint 5</i>
	dealer menambahkan produk ke cart	Produk muncul di dalam cart dealer	Produk berhasil ditambahkan ke cart	Sesuai	<i>Sprint 5</i>
Sistem Cart	dealer mengubah jumlah produk di cart	Jumlah produk berubah sesuai input	Berhasil memperbarui jumlah produk	Sesuai	<i>Sprint 5</i>
	dealer menghapus produk dari cart	Produk tidak lagi muncul di dalam cart	Berhasil menghapus produk dari cart	Sesuai	<i>Sprint 5</i>

Pada tabel 2 terlihat proses *sprint 2* hingga *sprint 5*. Pada *sprint 2*, pengujian pada modul manajemen harga meliputi penambahan, pengeditan, dan penghapusan kategori harga serta penambahan, pengeditan, dan penghapusan harga, dimana semua fitur berjalan sesuai spesifikasi, dilanjutkan dengan pengujian pada modul manajemen *sales order* yang menunjukkan bahwa penambahan *sales order* baru dan

pengeditan status berjalan optimal. Selanjutnya pada *sprint 3*, pengujian manajemen dan otomatisasi email mencakup konversi *sales order* menjadi file PDF/Excel, pengiriman email konfirmasi, dan pengecekan status email, dengan semua proses berhasil dilakukan sesuai harapan. Pada *sprint 4*, pengujian autentikasi *login dealer* memastikan bahwa dealer dapat login dengan benar, serta pengujian pada

katalog produk menunjukkan bahwa seluruh produk dapat ditampilkan dan dicari dengan tepat. Terakhir, pada *sprint* 5, pengujian detail produk memastikan bahwa informasi produk ditampilkan lengkap dan penambahan produk ke cart berhasil, sedangkan

pengujian Sistem cart membuktikan bahwa perubahan jumlah produk dan penghapusan produk dari cart dilakukan sesuai dengan input yang diberikan.

Tabel 3. *Sprint* 6

Modul	Deskripsi Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesi mpul an	Tahap Proses
Halaman Transaksi sales order	Dealer melihat daftar transaksi	Semua transaksi yang telah dilakukan ditampilkan	Berhasil menampilkan daftar transaksi	Sesuai	<i>Sprint</i> 6
	Cart otomatis dikonversi menjadi sales order setelah checkout	sales order baru muncul di daftar setelah checkout	sales order berhasil dibuat	Sesuai	<i>Sprint</i> 6

Pada tabel 3 terdapat detail *sprint* 6, pengujian pada halaman transaksi sales order memastikan bahwa seluruh transaksi yang telah dilakukan oleh dealer ditampilkan dengan benar, sehingga dealer dapat memantau status pesanan secara menyeluruh. Selain itu, fitur konversi otomatis cart menjadi sales

order juga diuji, dengan hasil bahwa setelah proses checkout, sales order langsung berhasil dibuat.

4.7. Waktu perbandingan perubahan sebelum dan sesudah implementasi sistem

Berikut merupakan perbandingan waktu proses estimasi sebelum dan sesudah implementasi sistem:

Tabel 4. Perubahan waktu transaksi

Tahapan Proses	Sebelum Sistem	Sesudah Sistem
1. Pengajuan Pesanan oleh dealer	Pengiriman PO secara manual. ± 1 jam–4 jam	Dealer transaksi pesan via web. ± 5 menit–1 jam
2. Pengecekan stok	Cek stok via Oracle/Excel ± 5–30 menit	Otomatis saat dealer checkout. ± 0.1–1 detik
3. Email dan approval dari sales manager	Email ke manager, tunggu approval. ± 6–48 jam	Manager approve via dashboard. ± 5–15 menit
4. Input Sales order ke Oracle/Excel	Manual input. ± 1–3 jam	Otomatis dari sistem. ± 1–3 detik
5. Pengiriman Email ke Divisi Terkait (termasuk gudang)	Tunggu perintah manual masuk ± 30 menit–2 jam	Email otomatis setelah approve. ± 1–3 detik

Pada tabel 4 ini, sebelum implementasi sistem, proses transaksi dimulai dengan pengajuan pesanan oleh dealer yang dilakukan secara manual dengan pengiriman PO selama 1–4 jam, dilanjutkan dengan pengecekan stok menggunakan Oracle/Excel selama 5–30 menit, kemudian pengiriman email dan menunggu approval dari sales manager selama 6–48 jam, diikuti dengan input sales order secara manual ke Oracle/Excel selama 1–3 jam, dan akhirnya pengiriman email ke divisi terkait yang memakan waktu 30 menit–2 jam, sehingga total estimasi waktu berkisar antara 515 menit (sekitar 8 jam 35 menit atau 0,36 hari) hingga 3450 menit (sekitar 57,5 jam atau 2,4 hari); sedangkan sesudah implementasi, pengajuan pesanan dilakukan melalui web selama 5 menit–1 jam, pengecekan stok terjadi otomatis saat checkout dalam waktu 0,1–1 detik, approval dari sales manager melalui dashboard hanya memerlukan 5–15 menit, serta input sales order dan pengiriman email berlangsung otomatis masing-masing hanya dalam 1–3 detik, sehingga total waktu transaksi menjadi berkisar antara sekitar 10 hingga 76 menit (kurang dari 1,3 jam), menunjukkan peningkatan efisiensi yang sangat signifikan.

5. KESIMPULAN DAN SARAN

Berdasarkan penelitian dan implementasi sistem penjualan berbasis web dengan integrasi otomatisasi sales order, dapat disimpulkan bahwa sistem ini berhasil meningkatkan efisiensi operasional, mempercepat proses transaksi dari proses manual ke proses digital serta mengurangi kesalahan dalam pencatatan manual. Dengan metode *scrum*, pengembangan dilakukan secara bertahap sehingga fitur seperti penjualan digital, manajemen produk, pengelolaan harga, sistem cart hingga otomatisasi pengiriman email dapat berjalan optimal sesuai kebutuhan pengguna dan sales. Pengujian sistem menggunakan *blackbox* menunjukkan bahwa fitur-fitur utama seperti e-commerce serta otomatisasi email berfungsi sesuai dengan yang diharapkan. Namun peneliti menyarankan bahwa skalabilitas pada aplikasi masih diperlukan untuk fitur-fitur lainnya. Masih terdapat beberapa aspek yang dapat ditingkatkan, seperti integrasi dengan sistem pembayaran digital untuk mendukung transaksi otomatis, fitur kredit, serta pengembangan fitur analitik untuk membantu pengambilan keputusan berbasis data.

DAFTAR PUSTAKA

- [1] J. F. Andry, D. Y. Bernanda, Honni, K. Christianto, and A. Andriani, "Analysis of information systems strategic planning using ward and peppard framework case e-commerce company," *Int. J. Adv. Appl. Sci.*, vol. 12, no. 2, pp. 179–187, 2023, doi: 10.11591/ijaas.v12.i2.pp179-187.
- [2] D. Wang, "Research on the Influence of Information Technology on Industrial Competitiveness," *J. Phys. Conf. Ser.*, vol. 1815, 2021, doi: 10.1088/1742-6596/1815/1/012031.
- [3] S. Triantafyllou, "Use of Business Information Systems to Achieve Competitiveness," *2022 13th Natl. Conf. with Int. Particip.*, pp. 1–4, 2022, doi: 10.1109/ELECTRONICA55578.2022.9874433.
- [4] M. Tanaman, "Web-based Inventory Management System," *Int. J. Sci. Appl. Inf. Technol.*, 2023, doi: 10.30534/ijisait/2023/021252023.
- [5] C. C. Utami, R. I. Desanti, and F. A. Halim, "The Development of Web-based Sales and Inventory System for a Stationary Store," *Ultim. InfoSys J. Ilmu Sist. Inf.*, 2023, doi: 10.31937/si.v14i2.3358.
- [6] F. A. Dzaky and D. Kurniawan, "Implementation of Scrum Framework Agile Method to Develop Integrated Asset Management Information System at Universitas Diponegoro Inventory Module," *J. Masy. Inform.*, 2023, doi: 10.14710/jmasif.14.1.52605.
- [7] H. Tannady, S. L. Felix, K. Christianto, F. S. Lee, and F. Isputrawan, "Aplikasi Persediaan, Penjualan, Dan Pencatatan Piutang Pada Pt. Sultana Agro Lestari," *JBASE - J. Bus. Audit Inf. Syst.*, vol. 5, no. 2, pp. 1–8, 2022, doi: 10.30813/jbase.v5i2.3775.
- [8] R. A. Saputra, E. B. Sambani, R. Rachman, D. Latif, and Y. P. Putra, "Inventory Information System Using LIFO Method," *Informatics Manag. Eng. Inf. Syst. J.*, 2024, doi: 10.56447/imeisj.v1i2.253.
- [9] V. Dylen, F. S. Lee, and M. Geasela, "Aplikasi Inventaris Berbasis Website Pada Toko Naomi Wig Menggunakan Metode Extreme Programing," *J. Teknol. Dan Sist. Inf. Bisnis-JTEKSIS*, vol. 6, no. 2, p. 339, 2024, [Online]. Available: <http://jurnal.unidha.ac.id/index.php/jteksishttps://doi.org/10.47233/jteksis.v6i2.1214>
- [10] L. Hidayati, "Evaluasi Sistem Informasi Manajemen Terhadap Pengguna Dalam Mendukung Pengelolaan Obat Dipuskesmas Kabupaten Bantul," vol. 2, pp. 116–124, 2021, doi: 10.36760/JP.V2I1.168.
- [11] C. A. Herdian, "Analisis dan Desain Sistem Informasi Pelatihan Dalam Mewujudkan Good Governance di Kabupaten Subang (Studi Kasus: Lembaga Ilmu Pengetahuan Indonesia Pusat Pengembangan Teknologi Tepat Guna)," *World Public Adm. J.*, 2022, doi: 10.37950/wpaj.v4i1.1343.
- [12] S. Ilyas, A. A. Shah, and A. Sohail, "Order Management System for Time and Quantity Saving of Recipes Ingredients Using GPS Tracking Systems," *IEEE Access*, vol. 9, pp. 100490–100497, 2021, doi: 10.1109/ACCESS.2021.3090808.
- [13] F. H. Nyembe, J. Van Der Poll, and H. H. Lotriet, "Formal Methods for an Agile Scrum Software Development Methodology," *Proc. Int. Conf. Adv. Technol.*, 2023, doi: 10.58190/icat.2023.35.
- [14] M. Hagal and H. M. Mftuh, "Improving Scrum Methodology Management using the Kano Model," 2024, doi: 10.37376/fesj.vi15.5633.
- [15] N. T. Rachman and D. Sushandoyo, "Analysis of Scrum Implementation in Digital Startup Product Development," *J. Econ. Bussines Account.*, 2021, doi: 10.31539/costing.v5i1.
- [16] M. M. Agustian, R. I. Desanti, and W. Wella, "E-Commerce System for Media Group Cooperative," *Ultim. J. Tek. Inform.*, 2023, doi: 10.31937/ti.v15i1.3202.
- [17] S. Tyagi, S. Yadav, U. Singhal, and H. Chaudhary, "Analysis and Development of E-Commerce Web Application," *2022 Fifth Int. Conf. Comput. Intell. Commun. Technol.*, pp. 65–72, 2022, doi: 10.1109/CCICT56684.2022.00024.
- [18] R. T. Amanda and R. A. Putri, "Application of User-Centered Design Method in E-Commerce Sales System," *SISTEMASI*, 2024, doi: 10.32520/stmsi.v13i3.4145.