

Penerapan Metode *Agile Scrum* Pada Pembuatan *Website* Penjualan Sembako Toko Erwin

Melanie Nurlies^{1*}, Agus Budiyantera², Irvan Lewenusa³

Sistem Informasi, Universitas Tarumanagara, Jakarta, Indonesia

¹melanie.825200053@stu.untar.ac.id, ²agusb@fti.untar.ac.id, ³irvanl@fti.untar.ac.id

*e-mail *Corresponding Author*: melanie.825200053@stu.untar.ac.id

Abstract

Toko Erwin is a grocery store with a complete selection of products. There are various types of basic food products that are sold in this shop directly. However, Toko Erwin still has shortcomings in terms of services that are still carried out conventionally and the lack of expanding customer reach. The creation of this website aims to help customers to order products online, after which the goods will be sent by the store without the customer having to go to the store. The implementation follows the Software Development Life Cycle (SDLC) method using the Agile Scrum model. The programming and markup languages used involve PHP, HTML, CSS. Testing of Toko Erwin's basic food sales website is carried out using Blackbox Testing and the test results on this basic food sales website are in accordance with the scenarios in the tests that have been carried out.

Key words: *Erwin Shop; Customer; Website; Agile Scrum*

Abstrak

Toko Erwin merupakan toko sembako dengan beragam pilihan produk yang lengkap. Terdapat berbagai jenis produk sembako yang dijual pada toko ini secara langsung. Namun, Toko Erwin ini masih memiliki kekurangan dalam hal pelayanan yang masih dilakukan secara konvensional dan kurangnya memperluas jangkauan *customer*. Dibuatnya *website* ini, bertujuan untuk membantu *customer* untuk melakukan pemesanan produk secara online, setelah itu barang akan dikirim oleh toko tanpa *customer* harus pergi ke toko. Implementasi mengikuti metode *Software Development Life Cycle* (SDLC) dengan menggunakan model *Agille Scrum*. Bahasa pemrograman dan markup yang digunakan melibatkan PHP, HTML, CSS. Pengujian *website* penjualan sembako Toko Erwin dilakukan dengan menggunakan *Blackbox Testing* dan hasil pengujian pada *website* penjualan sembako ini sesuai dengan skenario-skenario dalam pengujian yang telah dilakukan.

Kata kunci: *Toko Erwin; Customer; Website; Agile Scrum;*

1. Pendahuluan

Pengembangan *website* penjualan sembako pada Toko Erwin ini, menjadi topik penting untuk diteliti dan dikembangkan karena berkembangnya kebutuhan *customer* yang semakin meningkat dan adanya perubahan dalam sistem bisnis yang mulai merambah ke digital. Melalui sistem ini, pemilik toko dapat mencatat setiap penjualan, manajemen stok barang, serta menghasilkan laporan penjualan yang terperinci [1]. Pengembangan sistem ini juga dapat membantu *customer* dalam melakukan pemesanan barang untuk kebutuhan sehari-hari. Sesuai dengan konteks pembuatan *website* Toko Erwin, yang bertujuan untuk meningkatkan pelayanan operasional toko serta dengan adanya *website* toko ini dapat memperluas jangkauan *customer* yang lebih luas.

Kemajuan dalam sistem bisnis yang sudah merambah ke digital dapat dilihat dengan kebutuhan *customer* terhadap Toko Erwin. Terdapat perbedaan yang signifikan antara kondisi penjualan sembako secara konvensional dengan penjualan sembako dengan *website*. Saat ini, penjualan sembako secara konvensional cenderung adanya keterbatasan faktor seperti lokasi toko fisik, waktu operasional, serta keterbatasan jangkauan pasar. Dimana pelanggan harus data secara fisik ke toko yang dapat menjadi tidak nyaman terutama dalam kondisi lalu lintas atau cuaca yang buruk. Sedangkan penjualan sembako melalui *website* menawarkan kemudahan dan kenyamanan seperti pelanggan dapat mengakses toko kapan saja dan dari

mana saja. Dengan adanya *website* juga dapat memberikan informasi ke pelanggan mengenai produk, serta melakukan pemesanan jarak jauh [2]. Oleh karena itu, terdapat perbedaan besar antara kondisi penjualan sembako konvensional saat ini dengan potensi penjualan sembako melalui *website*, yang menunjukkan pentingnya adaptasi dan penerapan strategi penjualan yang lebih modern untuk meningkatkan daya saing dan relevansi bisnis.

Penelitian ini memanfaatkan *website* sebagai media yang multifungsi selain menampilkan informasi, dapat digunakan juga sebagai tempat menampilkan produk untuk toko. *Website* merupakan kumpulan halaman situs yang tergabung dalam suatu *domain* atau *subdomain*, yang ditempatkan ke dalam *World Wide Web* (WWW) di internet. Definisi dari *website* sendiri sudah mencakup halaman yang berisi berbagai data seperti teks, gambar, suara, dan elemen lainnya yang dapat diakses secara daring [3]. Dengan memanfaatkan metode *Agile Scrum* sebagai metode pengembangan *website* ini. Dapat membuat langkah-langkah kita menjadi sistematis untuk mencapai tujuan. Metode *Agile Scrum* merupakan suatu metodologi atau kerangka kerja yang terstruktur dalam pengembangan produk yang kompleks. *Scrum* menggunakan pendekatan yang bertahap dan berkelanjutan dalam mengoptimalkan kemampuan prediksi dan mengendalikan resiko [4]. Dalam pembuatan *website* tersebut diperlukannya penggunaan bahasa pemrograman dan markup seperti *Hypertext Preprocessor* (PHP), *Hypertext Markup Language* (HTML), dan *Cascading Style Sheets* (CSS) dan terintegrasi *database* dengan bantuan MySQL yang menjadi tempat penyimpanan data pelanggan, pemesanan dan produk. Dengan fokus pada pada informasi produk, serta pemesanan produk dan pembayaran yang efektif, diharapkan dapat meningkatkan pelayanan toko dan memperluas jangkauan *customer* baru.

Tujuan penulisan ini untuk dapat memberikan solusi dalam proses pengembangan *website* Toko Erwin. Dengan menggunakan metode *Agile Scrum*, yang diharapkan dapat mempermudah dalam pembuatan *website* serta meningkatkan efisiensi operasional Toko Erwin. Manfaat penelitian ini yaitu mencakup meminimalisir waktu pelanggan untuk berbelanja kebutuhan sehari-hari yang masih dilakukan secara konvensional, memperluas jangkauan *customer* baru, serta membantu operasional dari Toko Erwin sendiri.

2. Tinjauan Pustaka

Maliki [5], mendesain suatu aplikasi penjualan sembako secara grosir dan memiliki tujuan untuk memudahkan konsumen memperoleh informasi mengenai spesifikasi produk yang ditawarkan oleh Toko LA-RIS, memudahkan toko dalam menawarkan atau menjual produknya kepada konsumen yang berada di luar kota, serta memudahkan konsumen dalam proses pembayaran karena terdapat beberapa proses pembayaran. Terdapat juga sistem pengolahan atau manajemen produk yang baik yang dilengkapi keamanan, maka dapat mendukung kinerja dari sistem. Persamaan dengan penelitian ini yaitu persamaan penggunaan bahasa pemrograman PHP dan *database* yang digunakan yaitu MySQL. Selain itu juga, terdapat persamaan tujuan yang sama untuk memberikan kemudahan pada konsumen dalam melakukan pemesanan produk dan pembayaran. Adapun perbedaan pada penelitian ini, perbedaan basis yang digunakan menggunakan basis *mobile* dan menggunakan platform *Google Android*.

Aziz, Putra, dan Wahyudin [6], membuat aplikasi penjualan baju berbasis *mobile android*. Pembuatan aplikasi ini bertujuan untuk memperluas jangkauan promosi yang lebih luas karena sebelumnya hanya menggunakan promosi media sosial. Persamaan dari penelitian ini yaitu persamaan tujuan pembuatan aplikasi penjualan ini. Perbedaan pada penelitian ini yaitu basis yang digunakan berbasis *android*, bahasa pemrograman yang digunakan yaitu *java* dengan *software* pendukung *android studio*, pengujian aplikasi ini juga menggunakan metode *white box*.

Putra, Fitri, dan Hidayatullah [7], membuat aplikasi penjualan alat *music* berbasis *website*. Pembuatan aplikasi berbasis *website* ini bertujuan untuk mempermudah konsumen mendapat informasi mengenai penjualan alat musik secara *online* dengan cepat dan tepat baik mengenai produk, profil, harga dan juga cara berbelanja di toko *online* dan dapat langsung membuat transaksi penjualan. Persamaan pada penelitian ini, menggunakan metode *agile* dalam mengimplementasikan pembuatan aplikasi *e-commerce* berbasis *website* ini. Untuk perbedaan pada penelitian disini yaitu menggunakan dua metode yaitu selain *agile* disini juga menggunakan metode *waterfall* dalam implementasi pembuatan *website*.

Dahri, Hadi, dan Formis [8], membuat aplikasi penjualan *handhphone* berbasis *website* dengan menggabungkan dua toko mitra ke dalam satu *website* penjualan. Pembuatan aplikasi berbasis *website* ini bertujuan untuk meningkatkan omzet penjualan kedua toko yang bermitra, serta meningkatkan layanan informasi produk-produk pada setiap mitra dagang. Persamaan pada penelitian ini adalah menggunakan metode *System Development Life Cycle (SDLC)*, bahasa pemrograman dan *database* yang digunakan yaitu bahasa pemrograman PHP dan *database* MySQL. Untuk perbedaan penelitian ini terdapat dua toko bermitra dalam satu *website* penjualan.

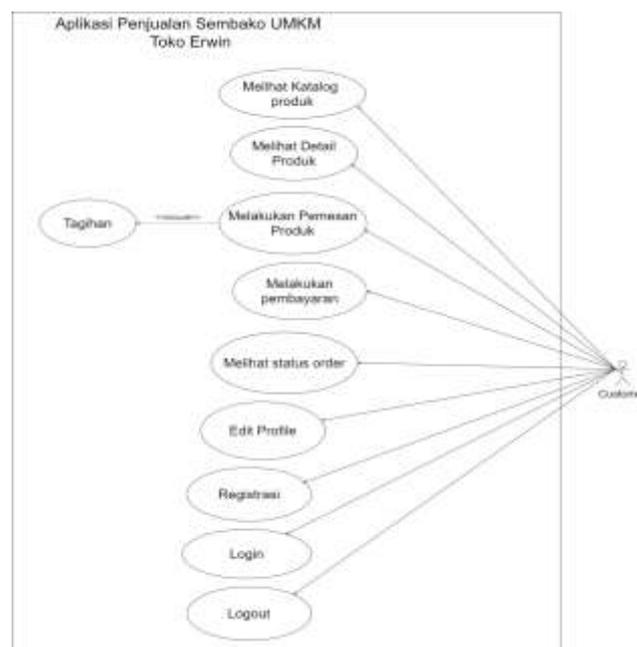
Firmansyah dan Herman [9], pembuatan aplikasi penjualan sepatu berbasis *website*. Pembuatan aplikasi berbasis *website* ini bertujuan untuk menganalisis pengaruh penggunaan *website e-commerce*, dan untuk membangun *website* yang dapat membantu konsumen memperoleh produk tanpa harus berpegangan jauh dan mengeluarkan biaya yang besar. Persamaan dari penelitian ini adalah penggunaan metode *Systems Development Life Cycle* dan persamaan tujuan dalam pembuatan *website* penjualan ini. Untuk perbedaan, terdapat perbedaan bahasa pemrograman yang digunakan yaitu menggunakan *Javascript* untuk membuat *website* lebih dinamis.

3. Metodeologi

Metode yang digunakan pada perancangan aplikasi ini adalah metode *agile scrum*, metode *agile scrum* ini merupakan salah satu metode rekayasa *software* yang menggunakan prinsip yang bertumpu pada kekuatan kolaborasi tim, *incremental* serta proses iterasi dalam mewujudkan hasil akhir. Menurut [10], *scrum* merupakan kerangka kerja yang mengatasi suatu masalah yang kompleks dan selalu berubah. *Scrum* juga dinilai dapat memberikan kualitas produk yang baik sesuai keinginan pengguna secara kreatif dan produktif. Berikut ini tahapan yang dilakukan dari metode *agile scrum* yang digunakan dalam perancangan aplikasi ini.

1) Tahapan *Product Backlog*

Pada tahap ini, berisi kumpulan informasi dari *owner* yang menghasilkan beberapa item yang ingin dirancang pada pembuatan program aplikasi penjualan ini seperti fitur apa saja yang akan ditambahkan. Pada tahap ini juga, dimulainya pembuatan diagram yang bertujuan untuk menampilkan model rancangan yang akan diimplementasikan pada *website*. Selain itu juga, bertujuan untuk mempermudah pengguna dalam memahami rancangan yang akan dibuat tim pengembang. Proses diagram menggunakan *Unified Modelling Language (UML)* yang terbagai menjadi *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*.

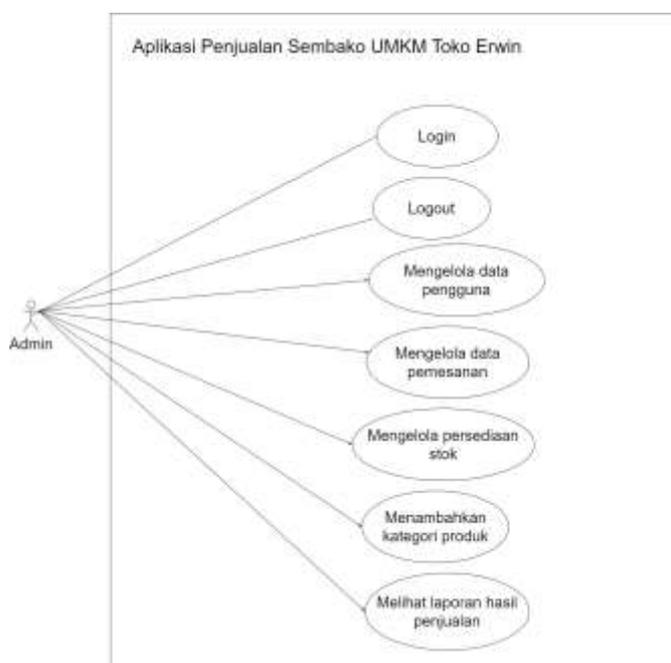


Gambar 1. Use Case Diagram Customer

Use case diagram digunakan untuk menggambarkan sesuatu yang perlu dilakukan sistem. *Use case diagram* merupakan salah satu cara untuk menggambarkan tampilan eksternal sistem dan interkasi dengan dunia luar [11]. Dalam rancangan *use case diagram* pada perancangan *website* ini, terbagi menjadi dua actor yaitu *customer* yang mewakili pengguna *website* toko ini dan admin sebagai pemilik toko atau orang yang mengurus *dashboard* toko. Perancangan *website* penjualan sembako Toko Erwin dapat dilihat pada Gambar 1 dan Gambar 2.

Pada Gambar 1 terdapat *use case diagram* dari *customer*. *Customer* dapat melihat katalog produk tanpa harus melakukan *register* atau *login*, lalu *customer* dapat melihat detail produk juga. Selanjutnya, *customer* dapat melakukan pemesanan produk namun sebelum melakukan pemesanan, *customer* diwajibkan untuk melakukan *register* atau *login* terlebih dahulu. Setelah melakukan pemesanan, *customer* akan mendapatkan tagihan. Selanjutnya *customer* akan melanjutkan dengan melakukan pembayaran dengan melakukan transfer ke rekening toko dan melakukan konfirmasi pembayaran dengan mengisi form. *Customer* dapat melihat status *order* yang berisi nomor resi pengiriman juga. *Customer* dapat melakukan *edit profile*, *register*, *login* apabila sudah memiliki akun dan *logout* akun.

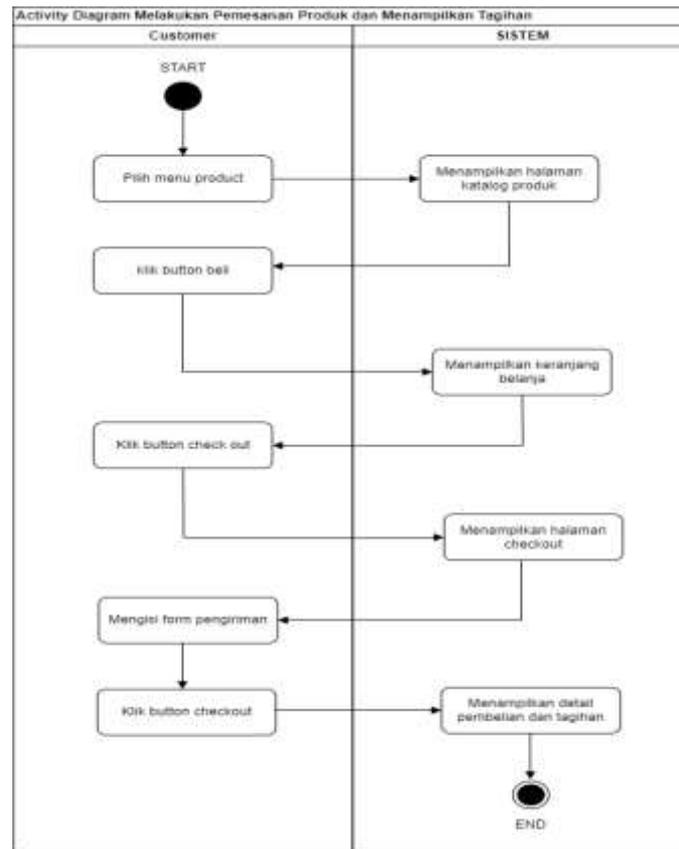
Pada Gambar 2 terdapat *use case diagram* dari admin. Disini admin dapat melakukan *login* serta *log out*, lalu admin dapat mengelola data pengguna, mengelola data pemesanan pelanggan, mengelola persediaan stok, menambahkan kategori produk serta melihat laporan penjualan. Semua ini dilakukan pada *dashboard* toko.



Gambar 2. Use Case Diagram Admin

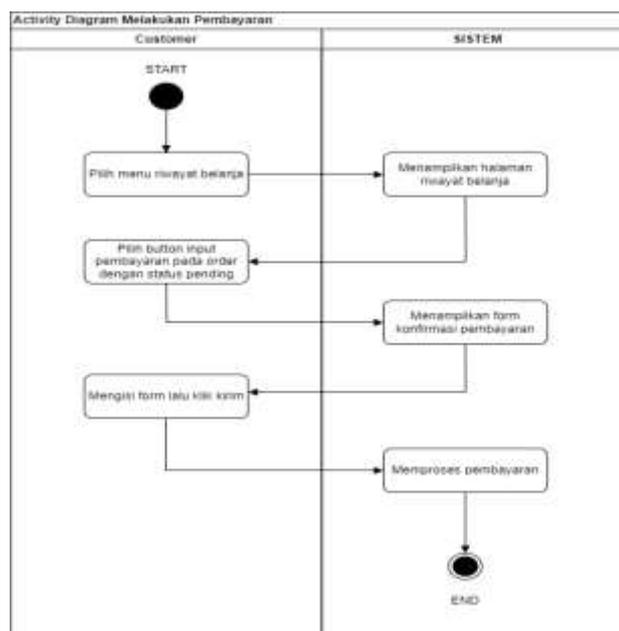
Selain *use case diagram*, adapun *activity diagram*. *Activity diagram* merupakan *workflow* dari sebuah sistem, proses bisnis, atau menu yang terdapat pada sebuah perangkat lunak yang diimplementasikan melalui *diagram*. *Activity diagram* juga menggambarkan semua ketegantungan yang menghubungkan keseluruhan aktivitas pada sistem perangkat lunak. Berikut adalah *activity diagram* dari melakukan pemesanan produk dan pembayaran.

Pada Gambar 3 untuk melakukan pemesanan barang, *customer* dapat memilih menu produk, sistem akan menampilkan katalog produk. Lalu *customer* mengklik menu beli pada produk yang ingin dibeli, sistem akan menampilkan halaman keranjang. Setelah itu, *customer* mengklik menu *checkout* pada halaman keranjang, sistem akan halaman *check out*. *Customer* mengisi *form* pengiriman sebelum klik menu *checkout*. Setelah mengisi *form* pengiriman, *customer* mengklik *button check out*. Sistem akan menampilkan detail pembelian beserta tagihan yang harus dibayar dan nomor rekening toko.



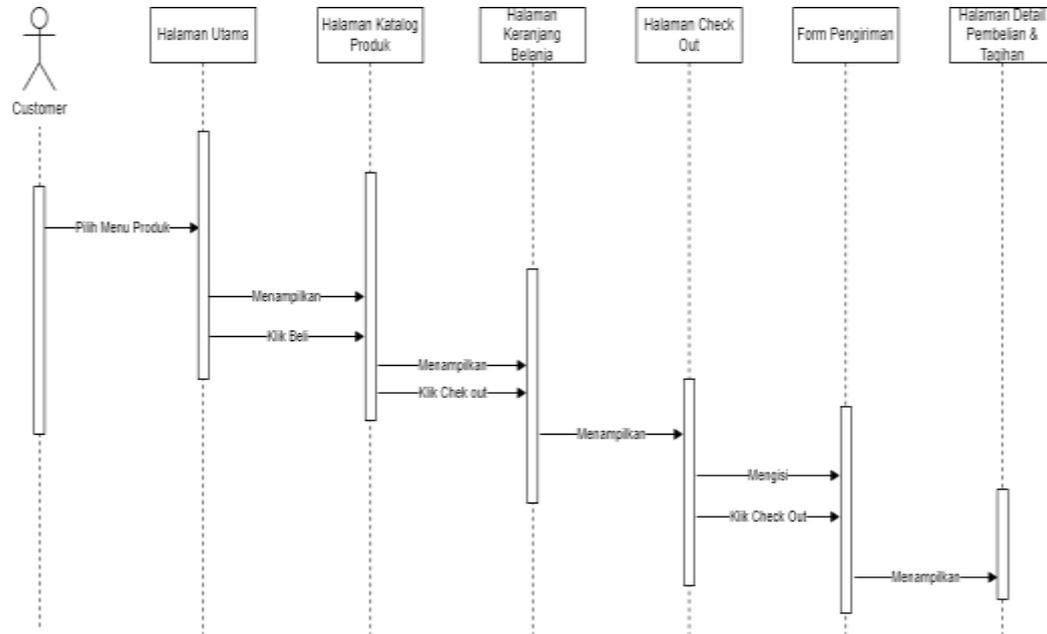
Gambar 3. Activity Diagram Melakukan Pemesanan Produk

Selain *use case diagram*, juga terdapat *activity diagram* pada gambar 4 dari pembayaran. Untuk tahap pembayaran, *customer* pilih *button riwayat belanja* dan sistem menampilkan halaman riwayat belanja. Lalu, *customer* pilih *button input pembayaran pada order dengan status pending*. Sistem menampilkan *form konfirmasi pembayaran* dan *customer* mengisi *form* pembayaran lalu klik kirim. Setelah mengirim *form* pembayaran, sistem memproses pembayaran.



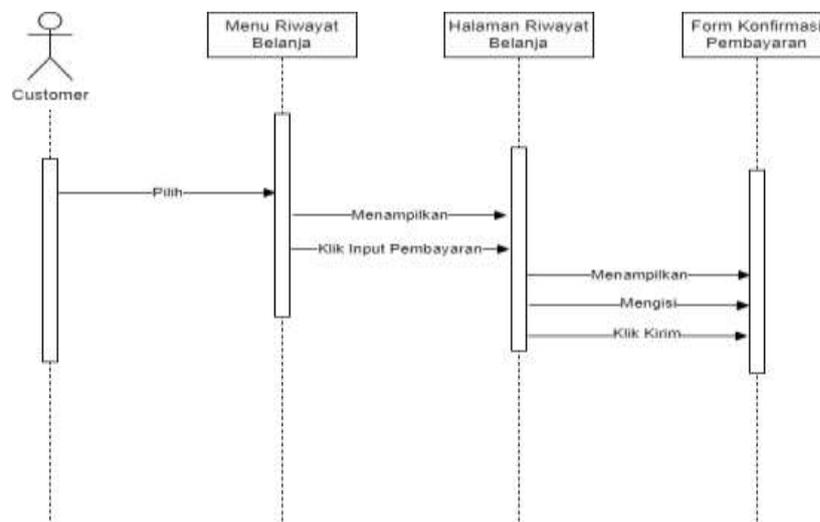
Gambar 4. Activity Diagram Pembayaran

Selanjutnya ada *sequence diagram*, *Sequence diagram* merupakan diagram yang berisi gambaran kolaborasi dari objek yang saling berinteraksi di antara komponen-komponen suatu kelas. *Sequence diagram* terdiri dari beberapa komponen *activations*, *actor*, *collaboration boundary*, *parallel vertical lines*, *processes*, *window*, dan *loop* [12]. Berikut *sequence diagram* pada perancangan *website* ini.



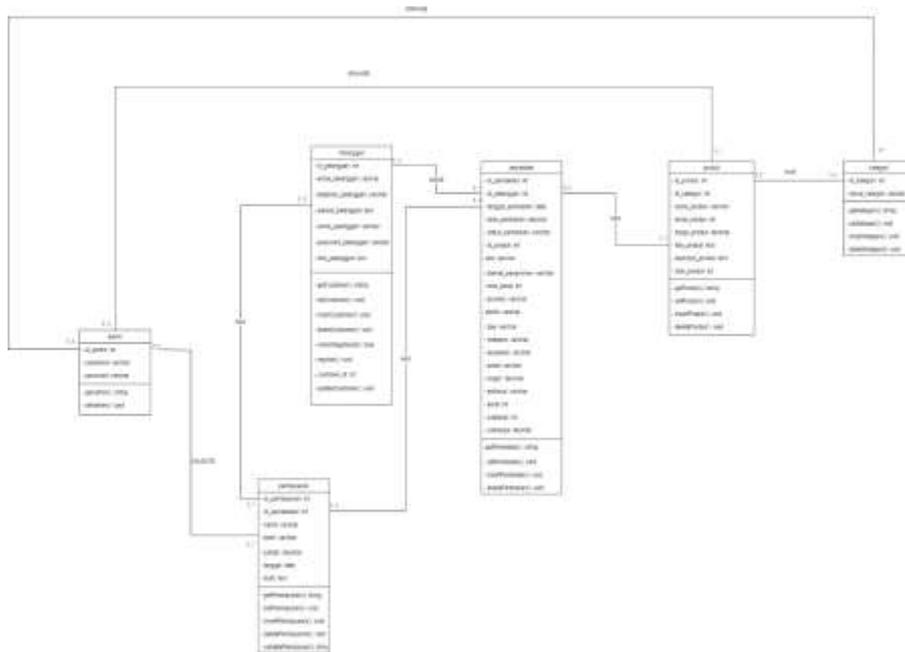
Gambar 5. *Sequence Diagram* Melakukan Pemesanan Produk

Selanjutnya, terdapat *Sequence Diagram* pembayaran pada Gambar 6 yang merupakan alur dari pelanggan saat melakukan pembayaran.



Gambar 6. *Sequence Diagram* Pembayaran

Selanjutnya, terdapat *class diagram*. *Class diagram* merupakan suatu diagram yang menggambarkan hubungan antar kelas dan memberikan penjelasan detail tiap kelasnya dalam desain dari suatu sistem, selain itu juga memperlihatkan aturan serta tanggung jawab entitas dalam menentukan perilaku sistem [13]. Berikut *class diagram* dari perancangan *website* ini.



Gambar 7. Class Diagram

2) Tahapan Sprint Backlog

Pada tahap ini, berisikan *list* tugas yang dipilih oleh tim untuk diselesaikan selama *sprint*. Ini merupakan daftar yang spesifik dan terstruktur yang digunakan sebagai panduan untuk tim tentang apa saja yang perlu mereka lakukan selama *sprint*.

Untuk *sprint* 1 dimulai dengan merancang *database*, membuat fitur registrasi, fitur *login*, dan halaman utama *website*. Lalu pembuatan *login* dan *dashboard* admin. Untuk *sprint* 2 akan merancang fitur katalog produk, detail produk, keranjang, dan *profile* pada *website*. Pada *dashboard* admin mulai menambahkan fitur tambah kategori, fitur mengelola persediaan stok dan fitur menampilkan data pelanggan. Untuk *sprint* 3 akan merancang fitur *checkout*, fitur pembayaran, fitur riwayat belanja, fitur lihat status *order* pada sisi pengguna. Pada *dashboard* admin akan dirancang fitur mengelola data pemesanan, dan fitur melihat laporan hasil penjualan.

3) Tahapan Sprint

Pada tahap ini, *developer* mulai mengerjakan program sesuai dengan *list item* yang sudah dibuat sebelumnya pada tahap *sprint backlog*. Pada tahap ini, biasanya terdapat pertemuan harian singkat yang disebut *daily scrum* untuk memperbarui kemajuan program serta mengidentifikasi masalah, dan menyesuaikan rencana jika diperlukan.

4) Tahapan Sprint Review dan Sprint Restrospective

Setelah tahapan *sprint* selesai, tahapan selanjutnya yaitu tahap *review* dan *restrospective* dimana hasil yang telah dirancang pada bagian *sprint* akan dilakukan *review* terlebih dahulu, apakah bagian tersebut sudah berhasil atau belum. Apabila masih terdapat masalah, bisa kembali pada tahap *sprint* lagi. Sedangkan tahap *restrospective* merupakan tahapan setelah tahapan *review* selesai dilakukan. Tahapan ini membantu meningkatkan kinerja tim dari *sprint* 1 ke *sprint* selanjutnya.

5) Tahapan Increment

Berikut ini tahapan terakhir dari perancangan program aplikasi ini yang menggunakan metode *agile scrum*. Pada tahapan *increment* ini, program aplikasi yang sudah tidak mengalami *error* dan sesuai dengan *sprint backlog* akan dijalankan secara *online* dan dapat digunakan oleh perusahaan maupun pengguna.

4. Hasil dan Pembahasan

4.1 Hasil Antarmuka

Antarmuka atau *User Interface* merupakan cara suatu program dan pengguna saling berinteraksi satu sama lain. *User Interface* merupakan suatu bagian dari komputer serta perangkat lunak yang dapat dirasakan dan dimengerti oleh pengguna serta salah satu pembuat tampilan dan design di dalam suatu perangkat komputer atau *software* [14]. Berikut tampilan antarmuka dari halaman pemesanan atau *checkout* dan halaman pembayaran pada *website* Toko Erwin.

1). Halaman *Checkout*

Tampilan halaman pengujian pemesanan produk atau *checkout* dapat dilihat pada Gambar 8 yaitu *form* pemesanan produk yang berfungsi untuk mengumpulkan informasi user seperti alamat lengkap, nama penerima, beserta nomor telepon. Informasi *user* ini berfungsi untuk melanjutkan proses pemesanan setelah *user* memilih produk yang ingin di beli. Selain *user* mengisi *form* untuk pengiriman, *user* juga dapat memilih nama ekspedisi dan paket nya untuk dilakukan pengiriman ke alamat *user*. Apabila *user* tidak melengkapi *form* ini, maka *user* tidak dapat melanjutkan proses pemesanan. Setelah mengisi semua yang ada pada *form*, *user* bisa mengklik *button checkout* yang akan mengarahkan ke halaman detail pembelian yang berisi tagihan yang harus dibayar dan nomor rekening toko.

No	Produk	Harga	Jumlah	Subharga
1	Beras	Rp. 30.000,00	1	Rp. 30.000,00
2	Miyak Beras	Rp. 15.000,00	1	Rp. 15.000,00
Total Belanja				Rp. 45.000,00

nama: 88753456

Alamat Lengkap Pengiriman: Jl. April No 5

Provinsi: DKI Jakarta | Daerah: Kota Jakarta Selatan | Ekspedisi: J&K | Paket: OKE 50.000 3-4

KODE: 000 | No. Telp: 88753456 | Nama: 1234 | Jml: 1000

Checkout

Gambar 8. Halaman *Checkout*

Detail Pembelian

Pembelian
No. Pembelian: 17
Tanggal: 04 March 2024
Total: Rp. 45.000,00

Pelanggan
NAMA: erwin
No. Telp: 88753456
E-mail: erwin@gmail.com

Pengiriman
ALAMAT: Kota Jakarta Selatan (DKI JAKARTA 1234)
Ongkos Kirim: Rp. 15.000,00
Dikirimkan: 04/03/2024
Alamat Pengiriman: Jl. April No 5

No	Nama Produk	Harga	Berat	Jumlah	Subberat	Subtotal
1	Miyak Beras	Rp. 15.000,00	1 Gg.	1	1 Gg.	Rp. 15.000,00
2	Gula	Rp. 15.000,00	1 Gg.	1	1 Gg.	Rp. 15.000,00

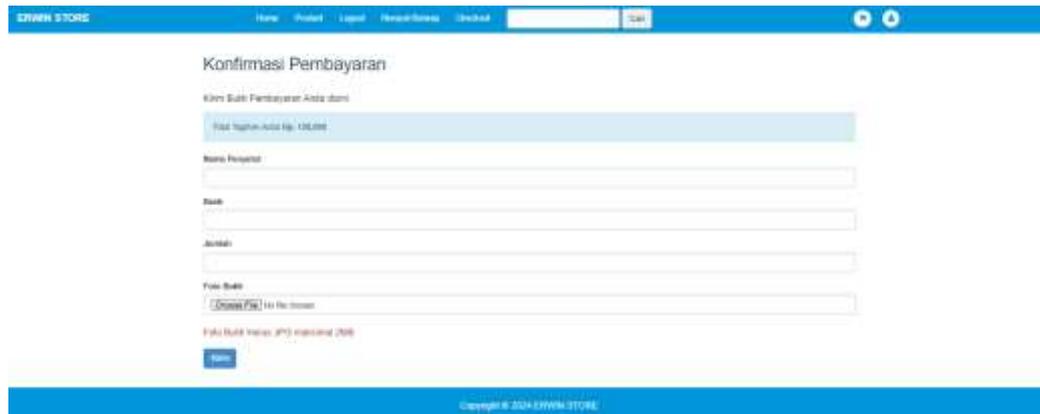
Saldo rekening pembelian Rp. 45.000,00
BANK BCA 1234567890123456

Gambar 9. Halaman Detail Pembelian

Gambar 9 merupakan halaman detail pembelian yang berisi detail pembelian *user* dan dibawah terdapat nomor rekening toko untuk melakukan pembayaran agar pemesanan dapat diproses dan segera dikirim ke alamat *user*.

2). Halaman Pembayaran

Tampilan halaman pengujian pembayaran pembelian produk dapat dilihat pada Gambar 10 dengan mekanisme *user* mengisi informasi tentang pembayaran ke *form* konfirmasi pembayaran yang berisi nama penyeter, nama *bank*, beserta total belanja sesuai dengan tagihan. Pada formulir tersebut, *user* diminta untuk menginput bukti pembayaran dan jika *user* sudah menginput bukti pembayaran maka nama *file* bukti pembayaran akan muncul. Setelah itu, *user* dapat mengklik kirim. Setelah berhasil, sistem akan mengarah ke halaman riwayat belanja dan apabila pembayaran sudah di *acc* maka akan diberikan resi pengiriman seperti pada Gambar 10.



Gambar 9. Halaman Pembayaran



Gambar 10. Halaman Riwayat Belanja

4.2 Pengujian Sistem

Tabel 1 merupakan pengujian fungsionalitas sistem menggunakan *Blackbox Testing* berdasarkan antarmuka, yang dinilai melalui hasil *output* tanpa memperlihatkan proses internal yang digunakan untuk menampilkan *output* tersebut. Pengujian *blackbox testing* merupakan metode perancangan data uji yang didasarkan pada spesifikasi perangkat lunak. Data pengujian dieksekusi pada perangkat lunak dan kemudia keluar dari perangkat lunak lalu diperiksa apakah telah sesuai dengan diharapkan atau tidak [15].

Table 1. Pengujian *Blackbox Testing*

No	Deskripsi Pengujian	Hasil yang diharapkan	Hasil
Pemesanan Produk / Checkout			
1.	Memasukkan barang ke halaman <i>checkout</i> dengan klik beli pada barang yang ingin dibeli	Barang yang ingin dibeli terdapat pada halaman <i>checkout</i>	Valid
2.	Melakukan pengisian <i>form</i> pemesanan lalu klik <i>checkout</i>	Pemesanan berhasil karena semua <i>field</i> terisi dan menampilkan halaman detail pembelian	Valid

No	Deskripsi Pengujian	Hasil yang diharapkan	Hasil
PEMBAYARAN			
1	Melakukan pengisian <i>form</i> semua <i>form</i> konfirmasi pembayaran hingga <i>upload</i> bukti pembayaran lalu klik kirim	Pembayaran berhasil karena semua <i>field</i> terisi dan dilanjutkan ke halaman riwayat belanja dan menampilkan resi pengiriman	Valid

4.3 Pembahasan

Berdasarkan permasalahan yang terdapat pada Toko Erwin untuk mempermudah pelanggan dalam pemesanan produk dan bertransaksi yang dapat dilakukan dimana saja tanpa terbatas waktu. Dengan adanya *website* Toko Erwin ini, maka pelanggan dapat melakukan pemesanan produk serta pembayaran dari jarak jauh tanpa harus datang ke toko. Pengujian pada penelitian ini sudah dilakukan dengan menggunakan *blackbox testing* pada skenario yang skenario yang diujikan dan sudah *valid* dan sesuai dengan hasil yang diharapkan.

Perancangan pada *website* Toko Erwin ini dibangun menggunakan pemodelan *System Development Life Cycle* (SDLC) seperti pada penelitian [8][9]. *Website* ini juga dibangun dengan metodeologi *Agile* seperti yang dilakukan pada penelitian [7]. Implementasi bahasa pemrograman yang digunakan ada beberapa seperti HTML, CSS, PHP, serta menggunakan *database MySQL* seperti pada penelitian [5][8]. Sistem penjualan berbasis *website* ini dapat membantu serta memudahkan pelanggan dalam melakukan pemesanan produk dari jarak jauh seperti yang dilakukan pada penelitian [5][6][7][8][9].

5. Simpulan

Berdasarkan proses hingga hasil dan pembahasan yang telah dilakukan dapat disimpulkan bahwa hasil perancangan *website* penjualan Toko Erwin dapat dilaksanakan dengan dengan baik dan berjalan lancar sesuai dengan tahapan proses metode yang diterapkan. Berdasarkan hasil pengujian yang telah dilakukan dapat disimpulkan tahap pemesanan hingga pembayaran dapat berjalan dengan lancar dan memiliki hasil yang sukses berdasarkan skenario yang diujikan sesuai kebutuhan fungsionalitas yang telah dibuat. *Website* penjualan Toko Erwin ini diharapkan dapat memudahkan pelanggan dalam melakukan pemesanan produk dan pembayaran secara *online*.

Daftar Referensi

- [1] S. B. Atim, "Permodelan Sistem Informasi Penjualan Barang Berbasis Website Menggunakan Metode Agile," *Journal of Artificial Intelligence and Technology Information*, vol. 2, no. 1, pp. 14–25, Mar. 2024, doi: <https://doi.org/10.58602/jaiti.v2i1.104>.
- [2] S. Dewi, L. Laudeciska, A. Figa, A. Auliani, D. Veronica Marbun, and W. Dwiyantri, "Perancangan Sistem Informasi Berbasis Website Pada Umkm Jaya Punggur | National Conference for Community Service Project (NaCosPro)," *ojs.digitalartisan.co.id*, vol. 3, no. 1, pp. 841–848, Oct. 2021.
- [3] A. L. Kalua, R. Mantiri, C. Rumondor, and E. Mogogibung, "Sistem Informasi Pendaftaran Beasiswa dan Jadwal Legalisir Berbasis Website Responsif (Studi Kasus: Dinas Pendidikan Sulawesi Utara)," *Journal of Information Technology, Software Engineering and Computer Science*, vol. 2, no. 2, pp. 58–74, Apr. 2024, doi: <https://doi.org/10.58602/itsecs.v2i2.108>.
- [4] W. Warkim, M. H. Muslim, F. Harvianto, and S. Utama, "Penerapan Metode SCRUM dalam Pengembangan Sistem Informasi Layanan Kawasan," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 6, no. 2, pp. 365–378, Aug. 2020, doi: <https://doi.org/10.28932/jutisi.v6i2.2711>.
- [5] M. I. Maliki, "Rancang Bangun Aplikasi Penjualan Grosir Sembako Pada Toko LA-RIS," *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 2, no. 3, pp. 304–311, Oct. 2021, doi: <https://doi.org/10.33365/jatika.v2i3.1222>.
- [6] A. Aziz, C. Galang Gemilang Putra, and W. Wahyudin, "Perancangan dan Implementasi Aplikasi Penjualan CnG Indonesian Product Berbasis Android dengan Metode White Box | JATISI (Jurnal Teknik Informatika dan Sistem Informasi)," *jurnal.mdp.ac.id*, vol. 9, no. 3, pp. 1861–1872, Sep. 2022.

- [7] W. A. Putra, I. Fitri, and D. Hidayatullah, "Implementasi Waterfall dan Agile dalam Perancangan E-Commerce Alat Musik Berbasis Website," *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 6, no. 1, pp. 56–62, Jan. 2022, doi: <https://doi.org/10.35870/jtik.v6i1.380>.
- [8] N. Dahri, H. S. Hadi, and R. R. Formis, "Perancangan Sistem E-Commerce Berbasis Kemitraan Dagang," *Jurnal Manajemen Teknologi Informatika*, vol. 1, no. 3, pp. 176–188, Dec. 2023.
- [9] Muhamad Dody Firmansyah and H. Herman, "Perancangan Web E- Commerce Berbasis Website pada Toko Ida Shoes," *Journal of Information System and Technology*, vol. 4, no. 1, pp. 361–372, May 2023, doi: <https://doi.org/10.37253/joint.v4i1.6330>.
- [10] A. A. D. Hartomo, "Rancang Bangun Aplikasi Penjualan Online Berbasis Web Menggunakan Metode Scrum | Jurnal Algoritma," *www.jurnal.itg.ac.id*, vol. 18, no. 1, pp. 161-172, Aug. 31, 2021.
- [11] S. Handayani, "Perancangan Sistem Informasi Penjualan Berbasis E-Commerce Studi Kasus Toko Kun Jakarta," *ILKOM Jurnal Ilmiah*, vol. 10, no. 2, pp. 182–189, Aug. 2018, doi: <https://doi.org/10.33096/ilkom.v10i2.310.182-189>.
- [12] T. Arianti, A. Fa'izi, S. Adam, M. Wulandari, and P. Aisyiyah Pontianak, "Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram UML (Unified Modelling Language)," *Jurnal Ilmiah Komputer Terapan dan Informasi*, vol. 1, no. 1, pp. 19–25, Feb. 2022.
- [13] M. Syarif and E. B. Pratama, "Analisis Metode Pengujian Perangkat Lunak Blackbox Testing Dan Pemodelan Diagram Uml Pada Aplikasi Veterinary Services Yang Dikembangkan Dengan Model Waterfall," *JTIK (Jurnal Teknik Informatika Kaputama)*, vol. 5, no. 2, pp. 253–258, Jul. 2021, doi: <https://doi.org/10.59697/jtik.v5i2.551>.
- [14] F. Saputra, N. Khaira, and R. Saputra, "Pengaruh User Interface dan Variasi Produk terhadap Minat Beli Konsumen (Studi Literature)," *Jurnal Komunikasi dan Ilmu Sosial*, vol. 1, no. 1, pp. 18–25, Jan. 2023, doi: <https://doi.org/10.38035/jkis.v1i1.115>.
- [15] 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, Dec. 2022.