

Perancangan dan Pembuatan Aplikasi Inventaris Berbasis Web pada Toko Nusantara Menggunakan Next.js

Andy Wijaya Nusantara¹⁾ Desi Arisandi²⁾ Novario Jaya Perdana³⁾

^{1) 2) 3)} Teknik Informatika, FTI, Universitas Tarumanagara Jl. Letjen S Parman no 1, Jakarta 11440 Indonesia
email: andy.825210077@stu.untar.ac.id¹⁾, desia@fti.untar.ac.id²⁾, novariojp@fti.untar.ac.id³⁾

ABSTRACT

Toko Nusantara merupakan sebuah toko bahan bangunan di Pematangsiantar, Sumatera Utara yang sudah berdiri sejak tahun 1993. Dari awal berdiri sampai dengan sekarang, Toko Nusantara masih mengandalkan sistem inventaris *periodic* yang dimana pencatatan stok barang akan dihitung di akhir bulan. Sistem pencatatan stok barang rentan terhadap kesalahan dan kehilangan data dikarenakan sistem pencatatan masih dilakukan secara manual menggunakan buku. Sistem yang sudah berjalan selama bertahun-tahun ini, kerap menimbulkan kerugian dikarenakan sering terjadinya kendala seperti kekurangan atau kelebihan stok dikarenakan kesalahan pencatatan. Tujuan dari pembuatan skripsi ini adalah menghasilkan sebuah aplikasi inventaris berbasis web yang dapat digunakan oleh Toko Nusantara untuk memudahkan dalam pencatatan barang, menyediakan informasi stok barang dengan *real-time*, mendigitalisasi data inventaris Toko Nusantara. Sedangkan manfaat yang diperoleh adalah meningkatkan efektivitas dan efisiensi pengelolaan inventaris pada Toko Nusantara, mengurangi resiko kehilangan data yang dapat terjadi apabila bergantung pada media penyimpanan secara fisik seperti buku, mempermudah proses pencatatan dan pengawasan stok barang, dan Membantu pengambilan keputusan bisnis. Perancangan dan pembuatan aplikasi berbasis web ini dilakukan menggunakan metodologi *System Development Life Cycle (SDLC) waterfall*. Bahasa pemrograman yang digunakan yaitu HTML, CSS, JavaScript, dan dengan menggunakan *framework* Next.js. Basis data yang digunakan untuk menyimpan data inventaris adalah MongoDB.

Key words

Inventaris, *Waterfall*, JavaScript, Next.js

1. Pendahuluan

1.1. Latar Belakang

Toko Nusantara, sebuah toko perseorangan yang telah berdiri sejak tahun 1993 di Pematangsiantar, Sumatera Utara, telah menjadi langganan bagi masyarakat sekitar dalam memenuhi kebutuhan alat dan bahan bangunan. Dengan beragam produk yang ditawarkan, mulai dari pasir, semen, hingga kayu-kayuan, toko ini

telah menjadi bagian dari aktivitas pembangunan di wilayah tersebut. Namun, di balik kesuksesannya, Toko Nusantara masih menghadapi tantangan dalam pengelolaan data inventarisnya. Sejak awal beroperasi, Toko Nusantara mengandalkan sistem inventaris *periodic* untuk mengelola data barang inventarisnya. Semua transaksi, mulai dari penerimaan hingga penjualan barang, dicatat secara tertulis dalam buku inventaris dan kemudian di akhir bulan, pemilik Toko Nusantara akan melakukan penghitungan fisik dari stok barang untuk memastikan jumlah dan kondisi dari barang tersebut.

Sistem inventaris yang sedang digunakan oleh Toko Nusantara, meskipun sederhana, tetapi memiliki beberapa kelemahan yang signifikan. Salah satunya adalah risiko kehilangan data. Selain risiko kehilangan data, sistem pencatatan manual juga menyulitkan Toko Nusantara dalam mengelola volume transaksi yang tinggi, sehingga berpotensi menimbulkan kesalahan pencatatan. Akibatnya, data inventaris yang dihasilkan tidak selalu akurat dan dapat menyebabkan masalah dalam pengelolaan stok barang. Permasalahan-permasalahan tersebut berdampak buruk pada toko. Sebagai contoh, jika data inventaris tidak akurat, dapat mengakibatkan kehilangan peluang penjualan karena kehabisan stok. Data yang tidak akurat juga akan membuat pengambilan keputusan bisnis menjadi kurang tepat. Misalnya, jika saat toko salah menduga bahwa stok suatu cat sudah habis dan kemudian melakukan pemesanan tambahan, hal ini dapat mengakibatkan penumpukan stok atau kelebihan persediaan. Kondisi ini sangat merugikan, terutama untuk bahan bangunan seperti semen dan cat yang memiliki kadaluarsa. Jika barang tersebut disimpan terlalu lama, kualitas bahan bangunan akan menurun, bahkan menjadi rusak dan tidak dapat dijual. Hal ini tentu saja dapat menimbulkan kerugian finansial bagi toko.

Melihat permasalahan tersebut, penulis berinisiatif untuk mengusulkan solusi digitalisasi sistem inventaris Toko Nusantara dengan menerapkan aplikasi inventaris berbasis web, yang diharapkan dapat mengatasi berbagai kendala yang dihadapi oleh toko saat ini. Dengan sistem inventaris ini, penulis ingin menerapkan sistem inventaris *perpetual* yang dikenal lebih efektif, dan efisien untuk pengelolaan data inventaris dengan jumlah volume yang tinggi. Dengan aplikasi ini juga akan memungkinkan Toko Nusantara untuk melakukan pencatatan transaksi secara lebih efisien, akurat, dan aman. Selain itu, aplikasi ini juga dapat menghasilkan catatan atau laporan stok

barang dan juga data transaksi berupa resi yang dapat disimpan dalam penyimpanan lokal ataupun dapat disimpan secara fisik dengan cara mencetak laporan tersebut.

1.2. Batasan

Dalam penyusunan skripsi ini, penulis memberlakukan beberapa batasan yang mencakup hal-hal berikut:

1. Aplikasi yang dirancang dan dibuat merupakan aplikasi inventaris berbasis web yang dapat diakses oleh pemilik toko.
2. Aplikasi inventaris ini memiliki 1 *role* saja yaitu, *role owner*.
3. *Role owner* dapat melakukan *login*, melakukan *logout*, memiliki akses ke setiap menu aplikasi, mengelola barang, dan mengelola resi.
4. *User Acceptance Test* (UAT) dilakukan secara online, yaitu website akan di hosting secara online dan UAT akan dilakukan oleh pemilik toko.

1.3. Tujuan dan Manfaat

Perancangan skripsi ini bertujuan untuk mencapai sasaran dan memberikan dampak positif sebagai berikut:

1.3.1. Tujuan

1. Menghasilkan sebuah aplikasi inventaris berbasis web yang dapat digunakan oleh Toko Nusantara untuk memudahkan dalam pencatatan barang.
2. Menyediakan informasi stok barang dengan *real-time*.
3. Mendigitalisasi data inventaris Toko Nusantara.

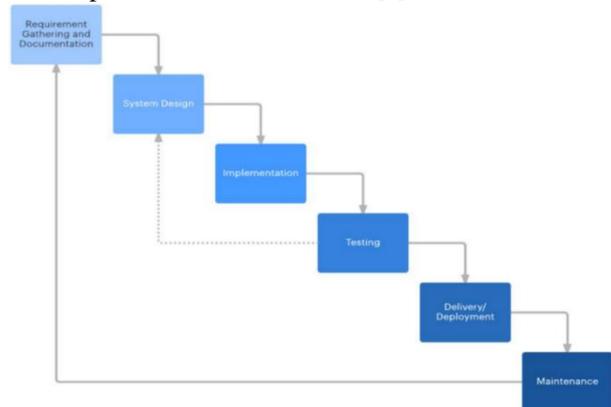
1.3.2. Manfaat

1. Meningkatkan efektivitas dan efisiensi pengelolaan inventaris pada Toko Nusantara.
2. Mengurangi resiko kehilangan data yang dapat terjadi apabila bergantung pada media penyimpanan secara fisik seperti buku.
3. Mempermudah proses pencatatan dan pengawasan stok barang.
4. Membantu pengambilan keputusan bisnis.

2. Metodologi

Metode yang digunakan oleh penulis untuk membuat aplikasi sistem informasi inventaris berbasis web ini adalah metodologi *waterfall*. Penulis memilih metodologi *waterfall* dikarenakan metode ini paling sesuai dengan cara kinerja penulis dalam membuat suatu sistem yang dimana, batasan-batasan, fitur-fitur, dan *requirement* lainnya sudah diidentifikasi dan ditetapkan

dari awal sebelum memulai proses implementasi. Metode *waterfall* merupakan salah satu metode *Software Development Life Cycle* (SDLC) yang dimana setiap tahap bergantung pada keberhasilan pada tahap sebelumnya. Metode *waterfall* sesuai dengan namanya, metode ini mirip dengan arah aliran air terjun yang mengalir dari tepi tebing, yang dimana titik akhir atau tujuan ditetapkan untuk setiap tahap pengembangan dan tidak dapat direvisi setelah selesai [1].



Gambar 1. Bagan Tahap-Tahap Metodologi *Waterfall* [1]

Gambar 1 menunjukkan tahapan pada metodologi *waterfall*. Menurut Rasch [1], metodologi *waterfall* memiliki tahapan sebagai berikut:

1. Requirement

Tahap *requirement* difokuskan pada menganalisis dan mengidentifikasi kebutuhan-kebutuhan *user* akan sistem yang akan dibangun. Pada tahap ini, penulis akan mengidentifikasi kebutuhan pemilik toko dengan memberikan pertanyaan-pertanyaan terkait kebutuhan-kebutuhan pemilik toko akan sistem yang akan dibangun. Dikarenakan jarak pemilik toko dan penulis yang tidak memungkinkan untuk wawancara secara tatap muka, maka wawancara akan dilakukan secara daring.

2. Design

Setelah memahami *requirement* dari *user*, langkah selanjutnya adalah merancang sistem secara detail. Tahap ini seperti membuat *blueprint* sebuah bangunan. Penulis akan membuat diagram alur yang menggambarkan setiap langkah dalam proses bisnis sistem inventaris, mulai dari *login*, *logout*, barang masuk, barang keluar, dan lainnya. Selain itu, penulis akan merancang tampilan antarmuka yang intuitif dan mudah digunakan. Basis data juga dirancang dengan cermat untuk menyimpan data inventaris secara terstruktur dan efisien.

3. Implementation

Tahap implementasi merupakan tahap di mana desain sistem yang telah dibuat

diwujudkan menjadi sebuah sistem yang nyata. Penulis akan menerjemahkan desain tersebut ke dalam kode program.

4. **Testing**

Sebelum sistem digunakan oleh *user*, perlu dilakukan pengujian secara menyeluruh untuk memastikan sistem dan fitur-fitur berfungsi dengan baik dan sesuai dengan harapan dan kebutuhan *user*. Pengujian ini meliputi berbagai jenis pengujian seperti *Unit Testing* dan *User Acceptance Testing (UAT)*. Tahap pengujian ini juga menguji secara menyeluruh agar sistem yang telah dibangun terhindar dari *bugs* atau *errors* saat dipakai oleh *user*.

5. **Deployment**

Tahap *deployment* merupakan tahap krusial di mana sistem yang telah dibangun dan telah melalui serangkaian tahap *testing*, akan di-*hosting* secara *online* agar dapat digunakan oleh *user*.

6. **Maintenance**

Tahap terakhir pada metodologi *waterfall* adalah tahap pemeliharaan. Pemeliharaan sistem secara berkala sangat penting untuk memastikan sistem tetap berjalan dengan baik dan relevan dengan perkembangan bisnis. Tahap pemeliharaan meliputi perbaikan *bugs* atau *issue* yang ditemukan, dan penambahan fitur-fitur baru sesuai dengan kebutuhan.

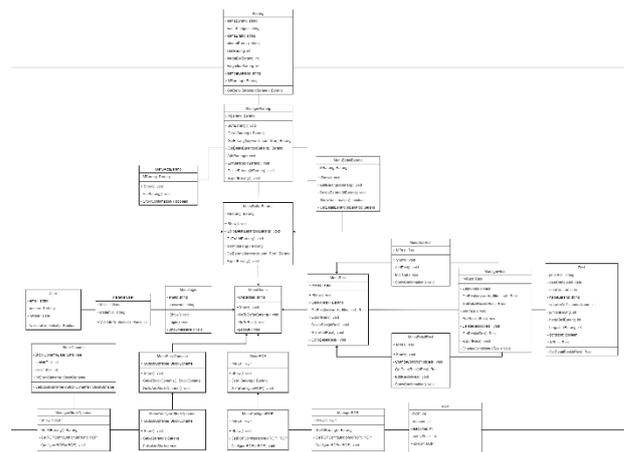
3. **Hasil Rancangan**

UML (*Unified Modeling Language*) merupakan sebuah standar yang digunakan untuk membuat model dokumentasi berorientasi objek dan merupakan dokumentasi model untuk sistem *software* [2]. UML dikembangkan oleh Grady Booch, Ivar Jacobson, dan James Rumbaugh [2]. UML merupakan semacam “*blueprint*” yang digunakan untuk membantu sistem dan *software developers* untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan sistem perangkat lunak.



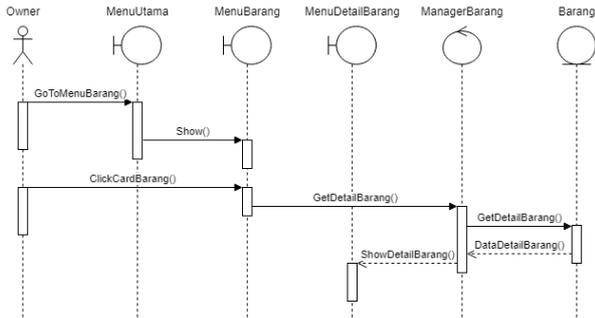
Gambar 2. Use Case Diagram

Gambar 2 menunjukkan *use case diagram* pada aplikasi inventaris Toko Nusantara. Terdapat 1 aktor yaitu *Owner* yang dapat mengakses keseluruhan dari aplikasi inventaris mencakup mengelola daftar barang, melakukan *login* dan *logout*, melakukan *reset password*, dan mengelola daftar resi.



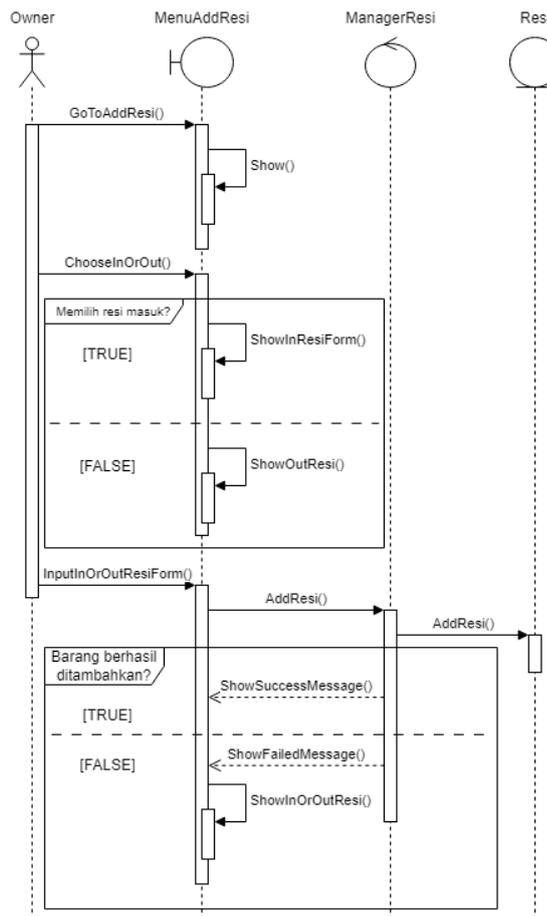
Gambar 3. Class Diagram

Gambar 3 menunjukkan *class diagram* yang menggambarkan hubungan antar kelas-kelas atau entitas pada menu-menu yang terdapat pada aplikasi inventaris seperti menu *login*, menu daftar barang, dan menu resi.



Gambar 4. Sequence Diagram Melihat Detail Barang

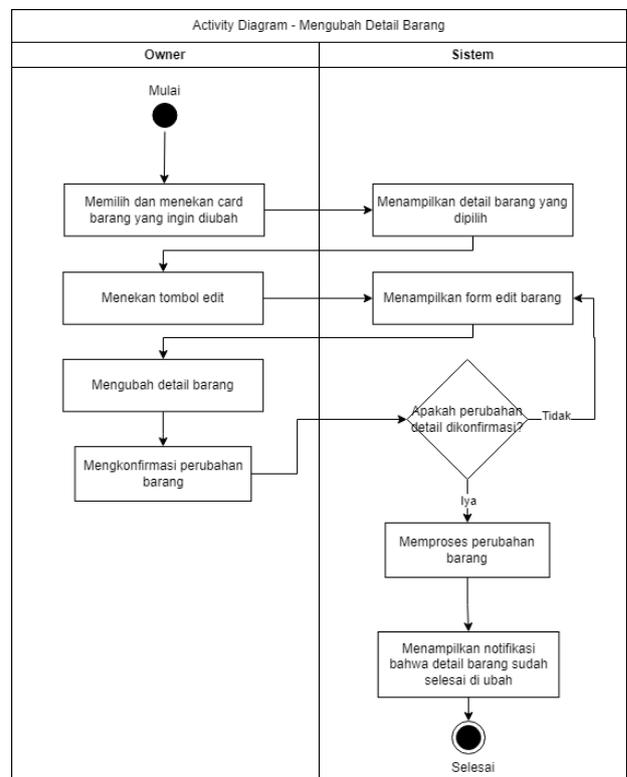
Gambar 4 menunjukkan sequence diagram saat owner melihat detail barang. Proses dimulai saat owner sudah melakukan login, lalu owner memasuki menu daftar barang. Setelah itu owner menekan card dari barang yang ingin dilihat detailnya. Kemudian, sistem akan melakukan fetch data ke database melalui ManagerBarang dengan menggunakan fungsi GetDetailBarang() yang kemudian data tersebut akan dikirimkan kembali dari database kepada ManagerBarang agar hasil dari fetch barang tersebut di tampilkan atau ditampilkan di menu detail barang.



Gambar 5. Sequence Diagram Buat Resi

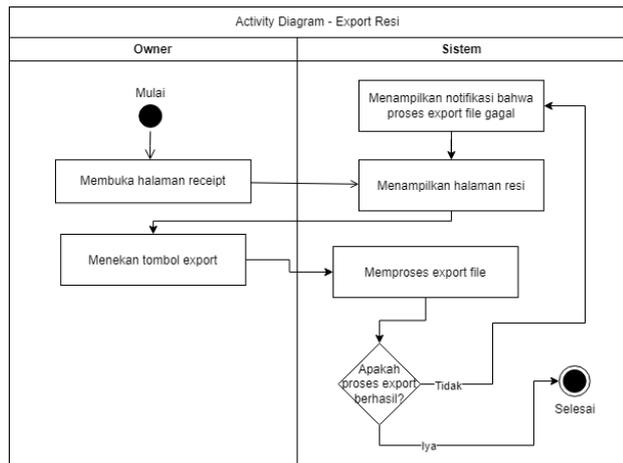
Gambar 5 menunjukkan alur sequence diagram dari proses pembuatan resi. Pembuatan resi dapat dilakukan setelah aktor sudah memasuki halaman resi.

Setelah itu aktor dapat membuat resi baru dengan menekan tombol *add* resi dengan fungsi `GoToAddResi()` yang kemudian modal pilihan antara resi masuk dan keluar akan ditampilkan. Setelah aktor selesai memilih jenis resi dengan menekan tombol, menu *Add* resi akan menampilkan *form* input untuk jenis resi yang dipilih yang kemudian dapat diisi oleh *owner*. Setelah selesai mengisi *form* dan menekan tombol *submit*, sistem akan mengirimkan data penambahan resi dengan fungsi `AddResi()` yang akan dikirimkan oleh `ManagerResi` ke *database* yang kemudian akan di proses oleh *database*. Jika proses penambahan barang di dalam *database* selesai atau sukses, maka `ManagerResi` akan menampilkan notifikasi bahwa resi selesai dibuat, dan sebaliknya jika penambahan barang gagal, maka notifikasi bahwa resi gagal dibuat akan ditampilkan.



Gambar 6. Activity Diagram Mengubah Detail Barang

Gambar 6 menunjukkan diagram alur aktivitas saat mengubah detail barang. Proses dimulai dari *owner* memilih dan menekan card dari barang yang ingin diubah detailnya. Lalu sistem akan menampilkan detail barang yang pilih dan kemudian *owner* menekan tombol *edit* barang. Setelah itu sistem akan menampilkan *form edit* barang yang dapat diinput oleh *owner*. Lalu setelah selesai, akan muncul konfirmasi untuk perubahan barang. Jika aktor menyetujui perubahan barang, maka sistem akan memproses perubahan detail barang yang akan dilanjutkan dengan menampilkan notifikasi bahwa barang sudah selesai diubah. Jika sebaliknya aktor tidak mengkonfirmasi perubahan, maka sistem tidak akan memproses perubahan barang dan akan kembali menampilkan *form edit* barang.



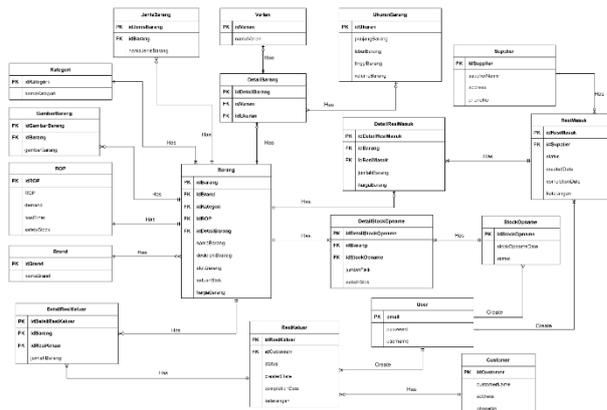
Gambar 7. Activity Diagram Export Resi

Pada Gambar 7, menunjukkan alur diagram aktivitas saat owner melakukan export resi. Proses dimulai dengan owner membuka halaman resi yang kemudian menu resi akan ditampilkan oleh sistem. Lalu owner menekan tombol export yang kemudian sistem akan melakukan proses export file. Jika proses export file berhasil, maka owner file daftar resi akan diunduh secara otomatis pada perangkat owner. Jika sebaliknya proses export file gagal, maka sistem akan menampilkan notifikasi bahwa export file gagal dan kembali menampilkan halaman resi.

Tabel 1. Entity Relationship

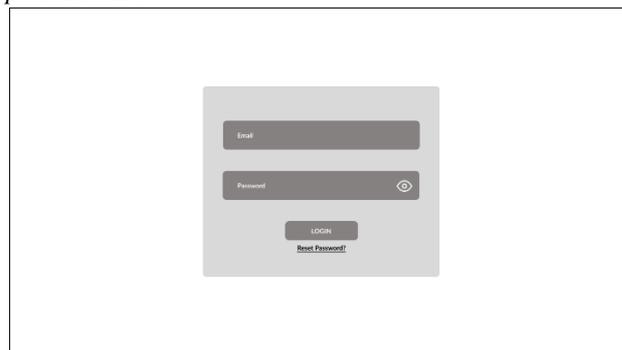
Entity	Multiplicity	Relationship	Multiplicity	Entity
Barang	1..1	Has	0..*	GambarBarang
Barang	1..*	Has	1..1	Kategori
Barang	0..*	Has	0..*	UkuranBarang
Barang	1..*	Has	0..1	Brand
Barang	0..*	Has	0..*	Varian
Barang	1..1	Has	1..1	ROP
Barang	1..*	Has	1..*	StockOpname
Barang	1..1	Has	0..*	JenisBarang
ResiMasuk	0..*	Has	1..1	Supplier
ResiMasuk	0..*	Has	1..*	Barang
ResiKeluar	0..*	Has	1..1	Customer
ResiKeluar	0..*	Has	1..*	Barang
User	1..1	Create	1..*	ResiMasuk
User	1..1	Create	1..*	ResiKeluar
User	1..1	Create	1..*	StockOpname

Tabel 1 menunjukkan entity relationship antar entitas yang digunakan untuk mendesain ER diagram database.

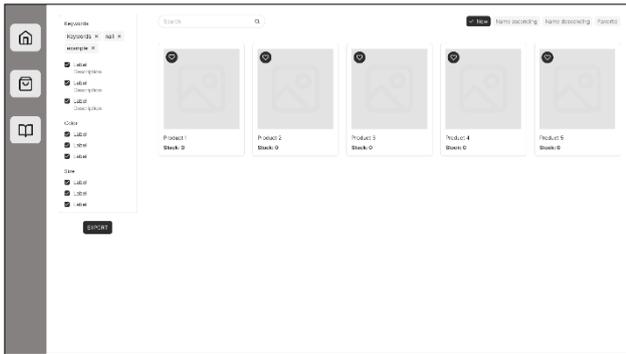


Gambar 8. ER Diagram

Menurut Yusri et al. [3], Wireframe adalah kerangka kerja desain yang digunakan untuk menyusun elemen-elemen pada halaman aplikasi sebelum proses desain utama dimulai. Wireframe dapat dikatakan merupakan gambaran kerangka dari sebuah aplikasi, website, atau produk lainnya. Wireframe dibagi menjadi 2 kategori, yaitu wireframe dengan fidelitas rendah dan wireframe dengan fidelitas tinggi. Kedua jenis wireframe ini digunakan sebagai dasar untuk pembuatan prototipe. Wireframe dengan fidelitas tinggi atau high-fidelity (Hi-Fi) mensimulasikan seperti apa produk akhir yang dihasilkan untuk mendukung pengujian produk. Sedangkan, wireframe dengan fidelitas rendah atau low-fidelity (Lo-Fi) berisikan sedikit atau tidak ada konten dan cenderung dibuat dalam skala abu-abu (grayscale). Desain fidelitas rendah berisikan teks dan gambar placeholder.

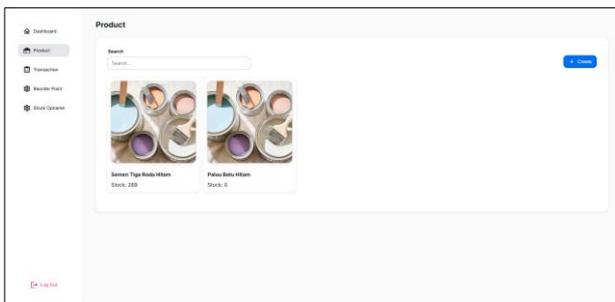


Gambar 9. Wireframe Login

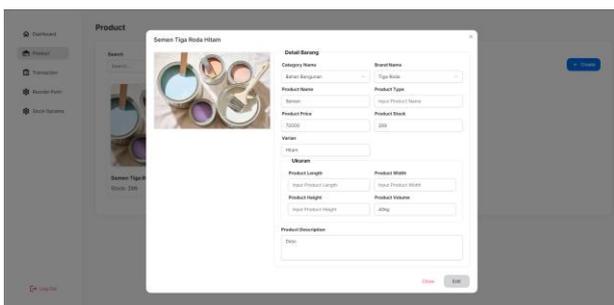


Gambar 10. Wireframe Menu Daftar Barang

Hasil dari rancangan merupakan aplikasi inventaris berbasis web yang hanya memiliki 1 user yaitu dengan role owner. Owner dapat mengakses semua menu dan juga dapat menggunakan semua fungsionalitas yang terdapat pada aplikasi inventaris. Menu-menu yang terdapat pada aplikasi inventaris terdiri dari, menu dashboard, menu daftar barang, menu transaksi, menu ROP (ReOrder Point), dan menu Stock Opname. Gambar 11 menunjukkan hasil rancangan menu daftar barang. Sedangkan Gambar 12 menunjukkan hasil rancangan modal untuk melakukan proses create, read, dan update barang.

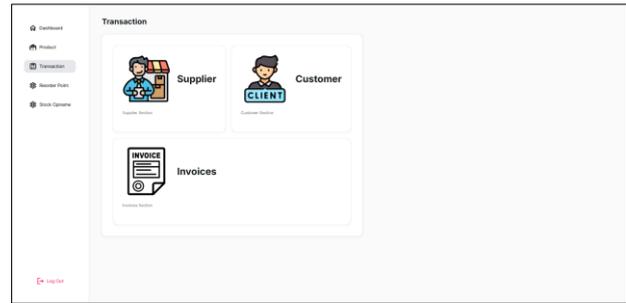


Gambar 11. Hasil Rancangan Menu Daftar Barang

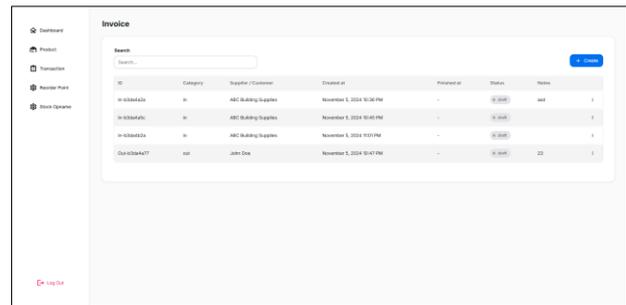


Gambar 12. Hasil Rancangan Modal Create, Read, dan Update Barang

Gambar 13 menunjukkan menu transaksi yang terbagi menjadi 3 sub-menu yaitu menu invoices, supplier, dan customer. Pada tiap menu tersebut, user dapat melakukan proses create, read, dan update.

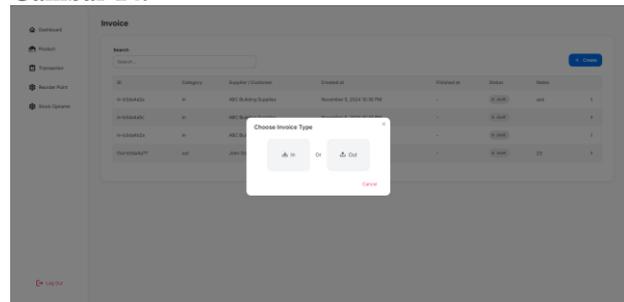


Gambar 13. Hasil Rancangan Menu Transaksi

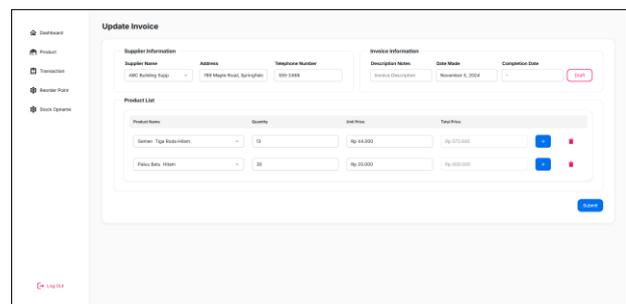


Gambar 14. Hasil Rancangan Menu Invoices / Resi

Hasil rancangan dari menu invoices, menu customer dan menu supplier merupakan list-list yang ditampilkan dengan menggunakan tabel yang dapat dilihat pada Gambar 14.



Gambar 15. Hasil Rancangan Modal Opsi Masuk atau Keluar



Gambar 16. Hasil Rancangan Form Create, Read, dan Update Invoice / Resi

Gambar 15 menunjukkan hasil rancangan modal untuk pilihan masuk atau keluar sebuah resi yang akan dibuat. Setelah user memilih antara masuk atau keluar, sistem akan menampilkan form sesuai dengan pilihan user dengan perbedaan pengisian field Customer atau Supplier. Gambar 16 menunjukkan hasil rancangan form pada proses create, read, dan update pada menu invoice / resi.

4. Kesimpulan

Kesimpulan yang dapat diambil dari perancangan dan pembuatan aplikasi inventaris berbasis web ini adalah efektivitas dan efisiensi pengelolaan inventaris pada Toko Nusantara meningkat dikarenakan digitalisasi dari pengelolaan inventaris yang masih manual. Risiko kehilangan data juga berkurang dikarenakan proses pencatatan dan pengawasan stok barang akan dilakukan dan disimpan secara online dan juga dapat diunduh menjadi berkas yang dapat diakses secara offline.

REFERENSI

- [1] F. A. Rasch, *Methodologies In Project Management*. Amazon Digital Services LLC - Kdp, 2019.
- [2] A. Nordeen, *Learn UML in 24 Hours*. Guru99, 2020.
- [3] A. R. Yusri, I. F. Hanif, M. D. Al-farel, N. Zaandami, and M. Yasin, "Perancangan Desain UI/UX Berbasis Scan Barcode Dengan Metode Design Thinking Untuk Pemesanan Makanan," *Bulletin of Information Technology (BIT)*, vol. 5, no. 2, pp. 102–113, 2024.

Andy Wijaya Nusantara, saat ini sebagai Mahasiswa S1 Program Studi Sistem Informasi Fakultas Teknologi Informasi Universitas Tarumanagara.

Desi Arisandi, memperoleh gelar S.Kom. dari Universitas Tarumanagara tahun 2004. Kemudian memperoleh gelar M.T.I. dari Universitas Indonesia pada tahun 2007. Saat ini aktif sebagai Dosen Tetap program studi Teknik Informatika Universitas Tarumanagara.

Novario Jaya Perdana, memperoleh gelar S.Kom. dari Institut Teknologi Sepuluh Nopember tahun 2011. Kemudian memperoleh gelar M.T. dari Universitas Indonesia pada tahun 2016. Saat ini aktif sebagai Dosen Tetap program studi Teknik Informatika Universitas Tarumanagara.