

# Perancangan Sistem *Inventory* Dengan *Barcode Scanner* Pada Toko Berkah Sejahtera Menggunakan Metode *Rapid Application Development*

1<sup>st</sup> Fadhia Nadhira Putri Yulianto  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

fadhianadhira@student.telkomuniversity.ac.id

2<sup>nd</sup> Augustina Asih Rumanti  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

augustinaar@telkomuniversity.ac.id

3<sup>rd</sup> Afrin Fauzya Rizana  
Fakultas Rekayasa Industri  
Universitas Telkom  
Bandung, Indonesia

afrinfauzya@telkomuniversity.ac.id

## Abstrak

Toko Berkah Sejahtera merupakan toko elektronik yang terletak di Kota Kediri, Jawa Timur. Toko ini masih melakukan proses pendataan barang masuk dan keluar menggunakan *Microsoft Excel* dan kartu stok. Sistem pendataan ini mengakibatkan terjadinya ketidaksesuaian jumlah barang di catatan dengan yang ada di gudang. Berdasarkan masalah tersebut, penelitian ini bertujuan untuk merancang sistem *inventory* berbasis *website* yang dapat dikontrol oleh *barcode scanner* sehingga diharapkan dapat mengurangi kemungkinan terjadinya kesalahan dalam proses pendataan. Perancangan sistem *inventory* dilakukan dengan mengikuti tahapan dalam metode *Rapid Application Development* (RAD) dan diuji menggunakan metode *black box testing* dan *user acceptance test* (UAT). Hasil yang didapat adalah rancangan sistem *inventory* dapat diterima oleh pengguna dengan rentang persentase penilaian UAT di antara 81%-100%. Dengan adanya sistem *inventory*, diharapkan dapat membantu pemilik toko dan pegawai toko dalam melakukan pendataan barang masuk dan keluar, mengecek stok barang, memberikan laporan penjualan, laporan barang terlaris, dan laporan barang yang cepat habis.

## I. PENDAHULUAN

Toko Berkah Sejahtera merupakan sebuah toko elektronik yang berdiri sejak akhir tahun 2019 dan

**Kata Kunci:** *Sistem Inventory, Website, Rapid Application Development, Barcode Scanner*

## Abstract

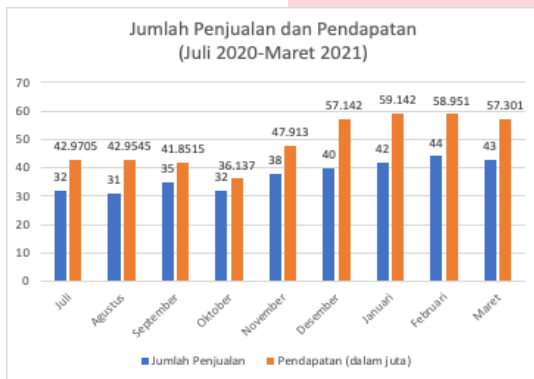
*Toko Berkah Sejahtera is an electronic store located in Kediri City, East Java. This store is still doing the data collection process for incoming and outgoing goods using Microsoft Excel and stock cards. This data collection process resulted in a discrepancy in the number of items in the records with those in the warehouse. Based on these problems, this study aims to design a website-based inventory system that can be controlled by a barcode scanner so that it is expected to reduce the possibility of errors in the data collection process. The design of the inventory system is carried out by following the steps in the Rapid Application Development (RAD) method and tested using the black box testing and user acceptance test (UAT) methods. The results obtained are the design of the inventory system can be accepted by the user with a percentage range of UAT assessment between 81%-100%. With the inventory system, it is hoped that it can help the user in conducting data collection of incoming and outgoing goods, checking stock of goods, providing sales reports, reports on best-selling items, and reports on items that run out quickly.*

**Keywords:** *Inventory System, Website, Rapid Application Development, Barcode Scanner*

berlokasi di Kota Kediri, Jawa Timur. Toko ini menawarkan produk elektronik yang dibutuhkan sehari-hari, seperti TV LED, mesin cuci, *freezer*

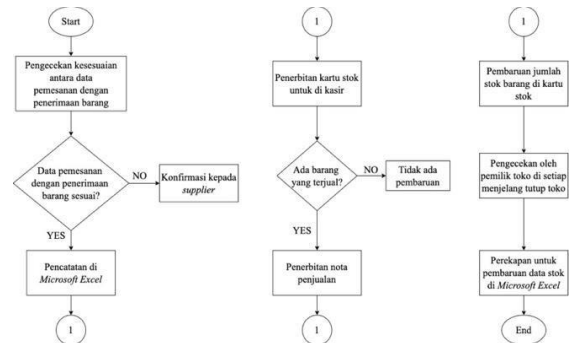
box, kulkas, speaker, dan showcase dari merek terkenal dengan harga yang bersaing.

Pemberian harga yang bersaing ini menjadikan Toko Berkah Sejahtera sebagai toko rujukan konsumen dalam mencari barang elektronik. Selain itu toko ini terletak di lingkungan perumahan dan merupakan toko elektronik satu-satunya di kecamatan tersebut menjadikan toko dapat bersaing dengan kompetitor. Dengan keunggulan tersebut, Toko Berkah Sejahtera berpotensi untuk berkembang lebih besar yang ditunjukkan dengan peningkatan penjualan barang selama bulan Juli 2020-Maret 2021, ditunjukkan pada Gambar 1.



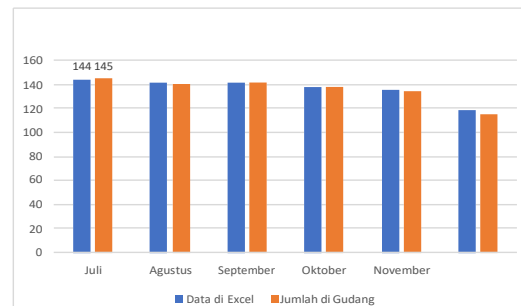
Gambar 1 Data pendapatan dan penjualan Juli 2020-Maret 2021

Gambar 2 menunjukkan alur proses *inventory* di Toko Berkah Sejahtera berdasarkan hasil wawancara dengan pemilik toko. Proses pendataan dimulai dari kedatangan pesanan dari pemasok. Sebelum disimpan di gudang, dilakukan pengecekan untuk mengetahui kecocokan jumlah dan jenis barang yang dipesan. Jika terdapat ketidaksesuaian, pemilik toko akan melakukan konfirmasi ulang kepada pemasok dan dilanjutkan dengan proses pendataan barang. Data barang disimpan dalam *file Spreadsheet* berbasis aplikasi *Microsoft Excel*. Berdasarkan data ini, kartu stok akan dikeluarkan dan disimpan di kasir. Setiap ada penjualan barang, kasir akan mengeluarkan nota penjualan dan memperbarui jumlah barang yang ada di kartu stok. Menjelang penutupan toko, pemilik toko melakukan pengecekan dan merekap data stok di *Microsoft Excel*.



Gambar 2 Flowchart proses pendataan *inventory* toko

Dengan menerapkan proses pendataan eksisting, pemilik toko menemukan masalah ketidaksesuaian jumlah barang di catatan dengan yang ada di gudang. Ketidaksesuaian data ini dapat dilihat di Gambar 3.



Gambar 3 Ketidaksesuaian Jumlah Barang (Juli-Desember 2020)

Pemilik toko juga kesulitan dalam mengetahui data stok yang valid karena perlu dilakukan penyesuaian data terlebih dahulu yang menjadi penghambat saat akan menambah stok. Hal ini juga menjadi kendala bagi pemilik pada saat akan berjualan secara *online* yang harus mulai dirintis dalam mengantisipasi kondisi pandemi Covid-19 berkepanjangan, seperti saat ini.

Alternatif solusi yang dipilih untuk menyelesaikan masalah ini adalah merancang sistem *inventory* berbasis *website* yang dapat dikontrol oleh *barcode scanner*. Sistem *inventory* ini dapat digunakan untuk mendata barang masuk dan keluar dengan cepat dan tepat dengan memanfaatkan *barcode SKU*. Selain itu sistem *inventory* dapat menampilkan data stok secara

valid dan *real time* sehingga dapat mempercepat pengambilan keputusan untuk penambahan stok.

Metode yang akan digunakan dalam merancang sistem *inventory* adalah metode *Rapid Application Development* (RAD), yaitu metode pengembangan sistem yang cocok untuk proyek dengan waktu yang singkat dan melibatkan *stakeholder* selama proses perancangan berlangsung. Metode ini terdiri dari empat tahap, yaitu *requirement planning*, *user design*, *construction*, dan *cutover*.

Dengan dirancangnya sistem *inventory*, diharapkan dapat memudahkan proses pendataan barang masuk dan keluar, mengecek stok barang, memberikan laporan penjualan, laporan barang terlaris, dan laporan barang yang cepat habis.

## II. KAJIAN TEORI

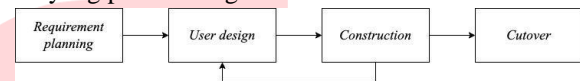
### A. Sistem *Inventory*

Sistem *inventory* adalah sistem yang mengelola proses keluar dan masuknya barang. Proses tersebut meliputi pendataan barang masuk, pendataan barang keluar atau terjual, data pengembalian barang, dan data stok barang yang tersedia untuk memudahkan mengetahui barang yang masuk maupun yang keluar [1]. Pengertian lain dari sistem *inventory* adalah aktivitas mengolah data barang yang tersedia di gudang. Sistem *inventory* berperan penting dalam suatu instansi karena dapat mendukung penyelesaian masalah pengolahan data barang dan mempermudah membuat laporan data barang yang tersedia [2]. Berdasarkan kedua definisi tersebut, dapat disimpulkan bahwa sistem *inventory* adalah sistem yang membantu mengelola *inventory* dengan melakukan pendataan barang. Pendataan ini meliputi barang masuk, barang keluar, pengembalian barang, penggunaan barang, dan stok barang. Pendataan barang berperan penting dalam menyelesaikan masalah data barang dan memudahkan pelaporan barang. Di era teknologi sekarang ini semakin memudahkan pemilik toko dalam melakukan manajemen *inventory* dengan adanya sistem manajemen *inventory*.

### B. *Rapid Application Development* (RAD)

RAD merupakan metode pengembangan sistem *user oriented* yang melibatkan pengguna selama tahap perancangan dan memiliki tahapan

yang lebih ringkas dari seluruh SDLC [3]. RAD disebut sebagai metode yang dapat mempercepat proses pengembangan sistem informasi dan sangat bergantung pada timbal balik pengguna. Pengguna dapat memeriksa model sejak awal perancangan, menentukan apakah model yang dirancang telah memenuhi kebutuhan, dan memberikan saran perubahan apabila diperlukan. Komunikasi dengan pengguna terus berlanjut hingga sistem telah selesai dikembangkan dan pengguna puas dengan hasil rancangan. RAD memiliki empat siklus hidup yang paralel dengan fase SDLC.



Gambar 4 Tahapan RAD

Berikut penjelasan untuk setiap tahapan RAD [3].

#### a. *Requirement planning*

Fase *requirement planning* mengkombinasikan komponen perancangan dan analisis sistem berdasarkan fase SDLC. Hal yang didiskusikan dalam fase ini adalah keperluan bisnis, ruang lingkup proyek, pembatas, dan keperluan sistem.

#### b. *User design*

Dalam fase *user design*, pengguna diizinkan untuk berperan dalam analisis sistem dan menguraikan model dan *prototype* yang menggambarkan keseluruhan *input*, *process*, dan *output* sistem. *User design* merupakan fase yang kontinu dan interaktif, dimana pengguna dapat memahami, mengubah, dan menyetujui model sistem untuk memenuhi kebutuhan pengguna.

#### c. *Construction*

Fase *construction* berfokus pada program dan tugas pengembangan aplikasi. Dalam RAD, pengguna dapat berpartisipasi dan memberikan saran perubahan atau peningkatan selama sistem sedang dikembangkan.

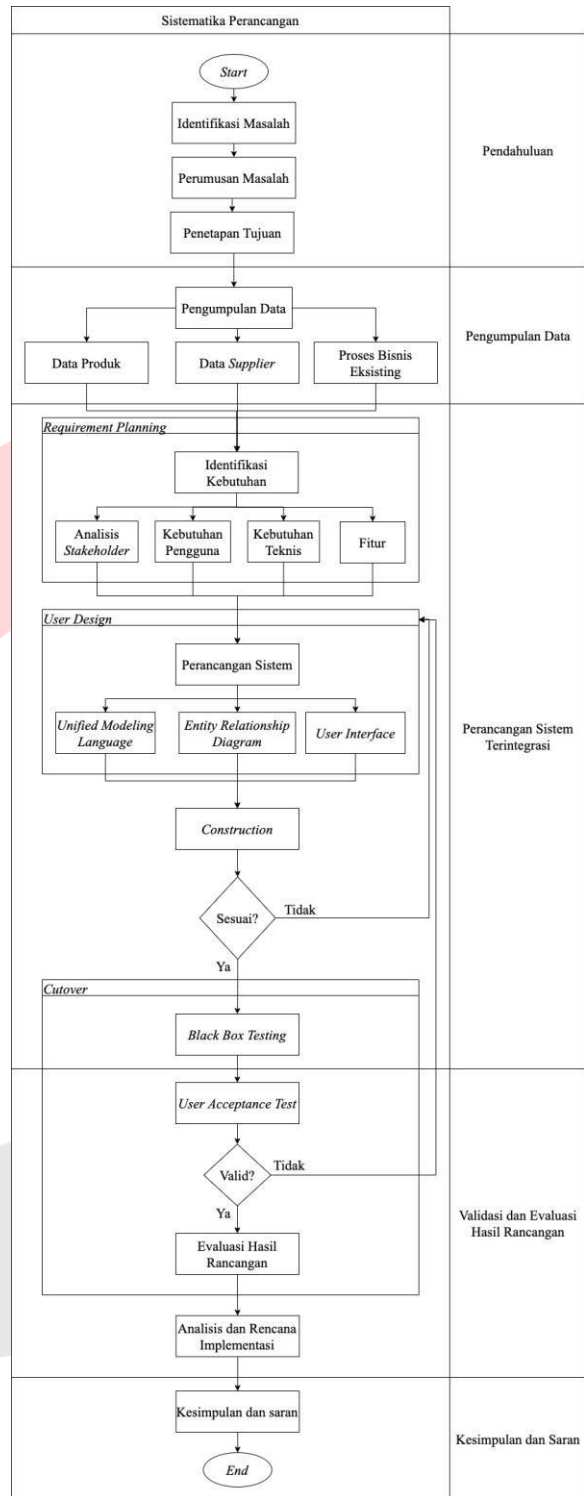
#### d. *Cutover*

Fase *cutover* serupa dengan fase implementasi di SDLC, meliputi transformasi data, pengujian, pergantian ke sistem baru, dan pelatihan untuk pengguna. Apabila dibandingkan dengan model

tradisional, proses di fase ini dipersingkat. Hasil yang diraih adalah sistem dibangun, disajikan, dan dioperasikan lebih cepat.

### III. METODE

Berikut merupakan sistematika perancangan penelitian ini.



Gambar 5 Sistematika Perancangan

#### A. Pengumpulan Data

Sebelum memasuki tahap pengembangan sistem, dilakukan pengumpulan data yang meliputi data produk, data *supplier*, dan proses

bisnis eksisting toko. Data produk dan data *supplier* merupakan data sekunder yang didapatkan dari pemilik toko dalam bentuk dokumen. Kedua data tersebut akan menjadi *input* sistem yang akan dikembangkan. Sedangkan proses bisnis eksisting toko merupakan data primer yang didapatkan dari melakukan wawancara dengan pemilik toko. Pemilik toko menjelaskan bagaimana alur proses penerimaan barang dari *supplier*, pendataan barang masuk, dan pendataan barang keluar atau terjual. Identifikasi proses bisnis eksisting dilakukan untuk menguraikan peran dan aktivitas operasional toko.

**3.1 Tahapan Perancangan**

Sistem dirancang dengan mengikuti tahapan dalam metode *Rapid Application Development* (RAD) yang terdiri dari tahap *requirement planning, user design, construction, dan cutover*. Dalam *requirement planning*, hal yang dilakukan adalah identifikasi kebutuhan yang meliputi analisis *stakeholder*, kebutuhan pengguna, kebutuhan teknis, dan hak akses dalam sistem. Dalam *user design*, hal yang dilakukan adalah proses pendesainan sistem menggunakan *unified modeling language* (UML), pendesainan basis data menggunakan *entity relationship diagram* (ERD), dan pendesainan *user interface* yang menggambarkan tampilan sistem yang akan dibangun. Selanjutnya dilakukan *construction*, yaitu tahap pengkodean sistem sesuai dengan desain yang telah dikembangkan di tahap *user design* dengan menggunakan *framework* dan bahasa pemrograman yang telah ditentukan. Dalam tahap ini, pengguna dapat memberikan *feedback* kepada pengembang sistem sehingga memungkinkan adanya perubahan, baik dari menu atau tampilan *user interface*. Setelah tidak ada perubahan, dilakukan tahap *cutover* yaitu pengujian sistem yang terdiri dari verifikasi dan validasi.

**B. Verifikasi Rancangan**

Verifikasi adalah proses pengecekan internal yang dilakukan oleh penulis menggunakan metode *black box testing*, yaitu pengujian yang memperhatikan fungsionalitas sistem. Pengujian ini dilakukan dengan melakukan *test case*

pengujian setiap menu yang telah disiapkan untuk mengetahui apakah sistem berhasil melakukan *test case* tersebut. Apabila telah lolos verifikasi, dilanjutkan dengan validasi.

**C. Validasi Rancangan**

Validasi adalah proses pengecekan eksternal yang dilakukan oleh pemilik toko dan satu pegawai toko untuk mendapatkan *feedback* terhadap sistem *inventory* yang telah dikembangkan. Proses validasi diawali dengan pengoperasian sistem secara langsung oleh pemilik toko dan satu pegawai toko untuk mendapatkan pengalaman pengguna. Setelah itu pengguna akan diminta untuk mengisi kuesioner *User Acceptance Test* (UAT) yang berisi pertanyaan dan nilai dari pengguna. Berdasarkan hasil kuesioner dapat diketahui apakah sistem telah sesuai dengan kebutuhan pengguna. Instrumen yang digunakan dalam UAT adalah *functional suitability, performance efficiency, operability, dan reliability* yang merupakan empat karakteristik penilaian kinerja perangkat lunak dari ISO 25010.

**IV. HASIL DAN PEMBAHASAN**

**A. Requirement Planning**

Dalam *requirement planning*, hal yang dilakukan adalah identifikasi kebutuhan yang meliputi analisis *stakeholder*, kebutuhan pengguna, kebutuhan teknis, dan hak akses dalam sistem.

**B. Identifikasi Stakeholder**

Identifikasi *stakeholder* dilakukan dengan mengikuti teori stakeholder oleh [4] yang terdiri dari *problem owner, problem user, problem customer, dan problem analyst*. Berikut merupakan *stakeholder* dalam sistem.

Tabel 1 Identifikasi Stakeholder

No.	Stakeholder	Penjelasan dalam Sistem
1.	<i>Problem owner</i>	Pemilik Toko Berkah Sejahtera
2.	<i>Problem user</i>	Pegawai
3.	<i>Problem customer</i>	Pegawai
4.	<i>Problem analyst</i>	Penulis



C. Identifikasi Kebutuhan Pengguna

Informasi mengenai kebutuhan pengguna didapat dari pemilik dan pegawai Toko Berkah Sejahtera selaku pengguna dari sistem yang akan dirancang. Hal ini dilakukan untuk mengetahui fitur yang dibutuhkan dalam sistem. Berikut merupakan kebutuhan pengguna yang didapatkan.

Tabel 2 Identifikasi Kebutuhan Pengguna

Pengguna	Kebutuhan User	Fitur Sistem
Pemilik toko	1. Sistem dapat melakukan pendataan barang masuk dan keluar dengan <i>barcode scanner</i> .	1. Sistem memiliki fitur <i>barcode reader</i> . 2. Sistem memiliki fitur diagram batang untuk menampilkan stok barang terkini.
	2. Sistem dapat menampilkan diagram stok barang.	3. Sistem memiliki fitur <i>download</i> dan <i>print</i> laporan penjualan.
	3. Sistem memiliki laporan penjualan.	4. Sistem memiliki menu untuk mengelola <i>user</i> .
	4. Sistem dapat membuat akun baru.	
Pegawai	1. Sistem dapat melakukan pendataan barang masuk dan keluar dengan <i>barcode scanner</i> .	1. Sistem memiliki fitur <i>barcode reader</i> . 2. Sistem memiliki fitur data stok barang.
	2. Sistem dapat menampilkan jumlah stok barang.	

D. Identifikasi Kebutuhan Teknis

Kebutuhan teknis menjelaskan spesifikasi kebutuhan yang diperlukan supaya sistem dapat berjalan baik. Berikut merupakan kebutuhan teknis untuk sistem *inventory*.

Tabel 3 Identifikasi Kebutuhan Teknis

Platform	Berbasis <i>website</i>
Server	XAMPP
Framework	CodeIgniter
Bahasa Pemrograman	PHP
Database	MySQL
Browser	Google Chrome, Safari
Keamanan	Hak otoritas diberikan kepada pengguna yang memiliki akun dan dapat <i>login</i>
Pengguna	Pemilik toko (admin) Pegawai
Konten	Sistem <i>inventory</i> untuk pendataan barang masuk dan keluar Toko Berkah Sejahtera.

E. Identifikasi Hak Akses

Tahap ini dilakukan untuk menguraikan menu apa saja yang dapat diakses oleh masing-masing peran dalam sistem. Terdapat dua peran dalam sistem ini, yaitu admin (pemilik toko) dan pegawai. Sistem *inventory* nantinya akan dapat menampilkan menu sesuai dengan peran yang sedang *login*. Berikut merupakan hak akses pengguna dalam sistem *inventory*.

Tabel 4 Identifikasi Hak Akses

Menu	Fungsi	Hak Akses
<i>Login</i>	<i>Login</i> adalah laman yang pertama muncul bila pengguna membuka <i>website</i> -nya. Pengguna akan diminta untuk memasukkan <i>username</i> dan <i>password</i> untuk dapat mengakses <i>website</i> .	<ul style="list-style-type: none"> <li>• Admin</li> <li>• Pegawai</li> </ul>
<i>Dashboard</i>	Laman <i>dashboard</i> akan menampilkan jumlah item yang dijual, jumlah <i>supplier</i> , jumlah pengguna dalam sistem, barang yang stoknya menipis, dan diagram batang yang menampilkan jumlah stok item terkini. Data yang ditampilkan di <i>dashboard</i> adalah data dinamis.	<ul style="list-style-type: none"> <li>• Admin</li> <li>• Pegawai</li> </ul>
<i>Supplier</i>	Laman ini akan menampilkan informasi <i>supplier</i> , meliputi nama, nomor telepon, asal kota, dan deskripsi. Pengguna dapat menambahkan, memperbarui, dan menghapus data.	<ul style="list-style-type: none"> <li>• Admin</li> <li>• Pegawai</li> </ul>
<i>Products</i>	Menu <i>products</i> memiliki dua submenu, yaitu submenu <i>categories</i> dan <i>items</i> . Submenu <i>categories</i> akan menampilkan informasi kategori jenis produk yang dijual. Sedangkan submenu <i>items</i> akan menampilkan informasi model produk yang dijual. Pengguna dapat menambahkan, memperbarui, dan menghapus data.	<ul style="list-style-type: none"> <li>• Admin</li> <li>• Pegawai</li> </ul>
<i>Transaction</i>	Menu <i>transaction</i> memiliki tiga submenu, yaitu submenu <i>sales</i> , <i>stock</i>	<ul style="list-style-type: none"> <li>• Admin</li> <li>• Pegawai</li> </ul>

Menu	Fungsi	Hak Akses
	<i>in</i> , dan <i>stock out</i> . Submenu <i>sales</i> berfungsi sebagai <i>point of sales</i> untuk mencatat produk yang akan dibeli oleh pelanggan dan dapat mencetak nota. Submenu <i>stock in</i> akan menampilkan informasi produk yang baru diterima dari <i>supplier</i> . Sedangkan submenu <i>stock out</i> akan menampilkan produk yang keluar namun bukan karena terjual, contohnya seperti produk yang rusak. Pengguna dapat menambahkan, memperbarui, dan menghapus data.	

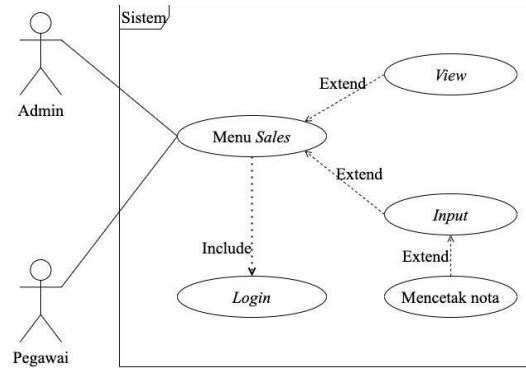
F. User Design

Dalam *user design*, hal yang dilakukan adalah proses pendesainan sistem menggunakan *unified modeling language* (UML), pendesainan basis data menggunakan *entity relationship diagram* (ERD), dan pendesainan *user interface* yang menggambarkan tampilan sistem yang akan dibangun.

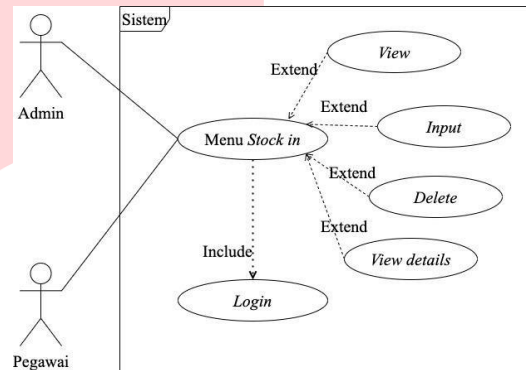
G. Desain Sistem

*User design* merupakan tahap dalam metode RAD untuk melakukan perancangan sistem dengan diagram UML dan perancangan *user interface*. Diagram UML yang digunakan adalah *use case diagram*, *activity diagram*, dan *sequence diagram* karena berkaitan erat dengan *user* dan perangkat.

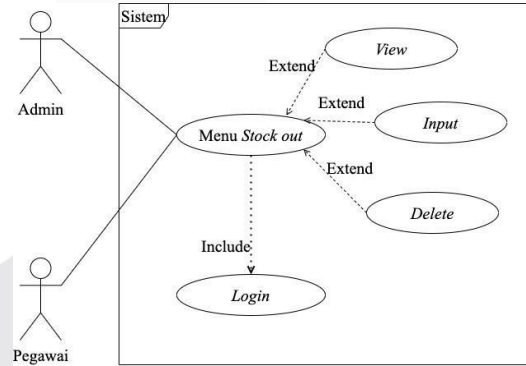
*Use case diagram* adalah diagram yang mengilustrasikan keterkaitan antara aktor dengan *use case* [5]. Diagram ini dapat menunjukkan apa yang dapat dikerjakan oleh pengguna dalam sistem secara umum. Berikut merupakan *use case diagram* sistem ini.



Gambar 6 Use case diagram menu sales

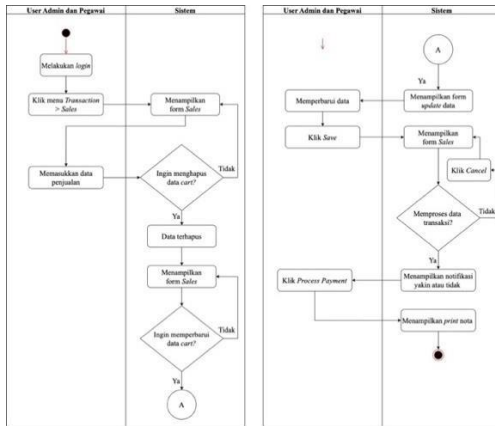


Gambar 7 Use case diagram menu stock in



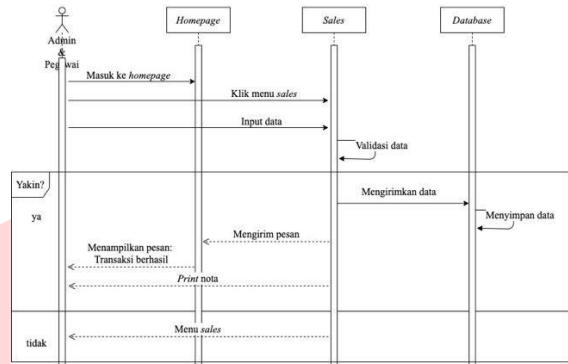
Gambar 8 Use case diagram menu stock out

*Activity diagram* adalah diagram yang menunjukkan alur aktivitas dan peran yang menjalankannya. Berikut merupakan *activity diagram* sistem ini.

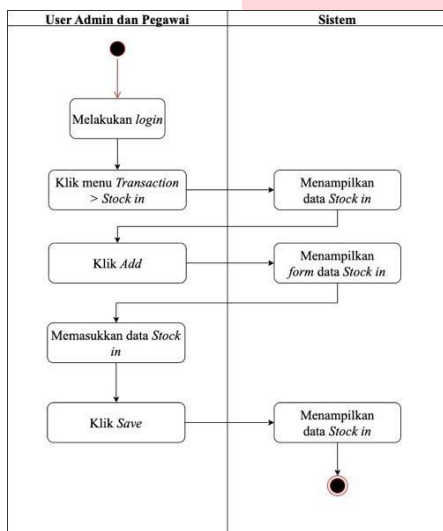


Gambar 9 Activity diagram menu sales

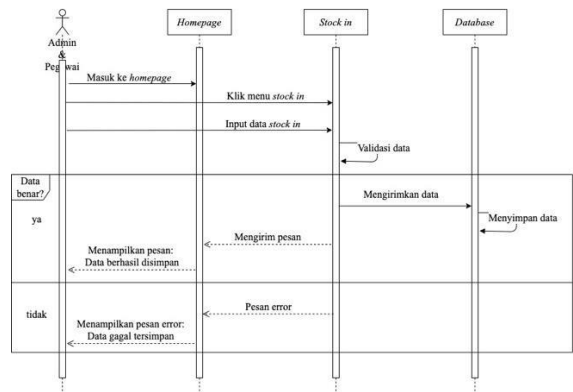
Sequence diagram merupakan diagram yang mengilustrasikan interaksi antar komponen berdasarkan waktu [6]. Berikut merupakan sequence diagram sistem ini.



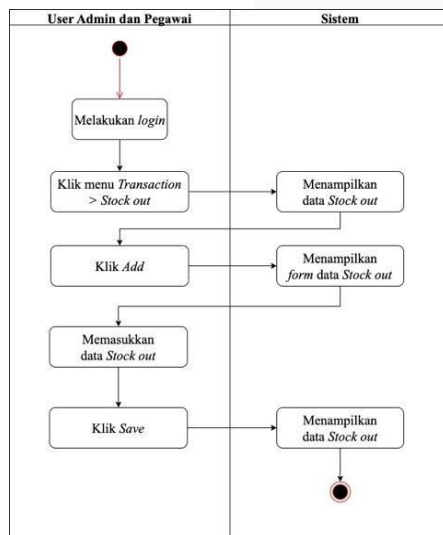
Gambar 12 Sequence diagram menu sales



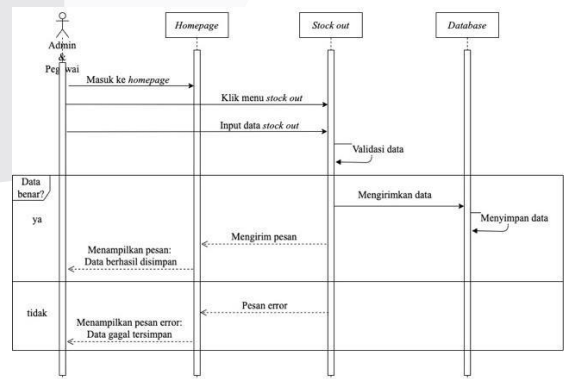
Gambar 10 Activity diagram menu stock in



Gambar 13 Sequence diagram menu stock in



Gambar 11 Activity diagram menu stock out

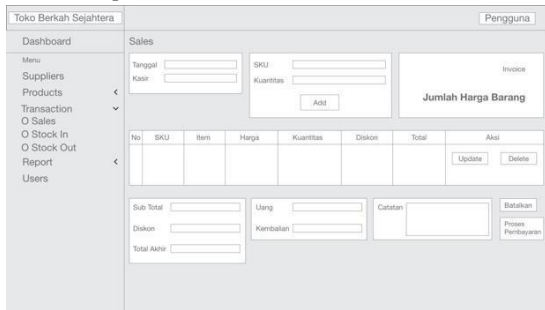


Gambar 14 Sequence diagram menu stock out

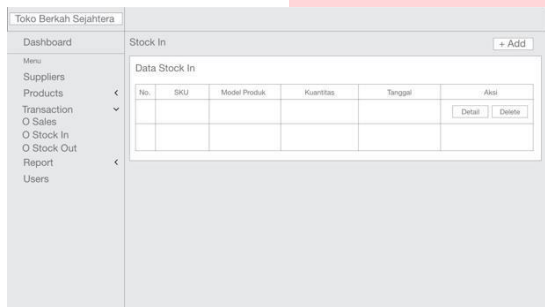


H. Mockup

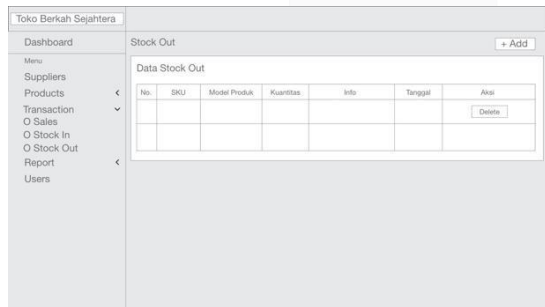
Mockup adalah gambaran tampilan dari sistem yang akan dirancang. Berikut merupakan mockup sistem ini.



Gambar 15 Mockup menu sales



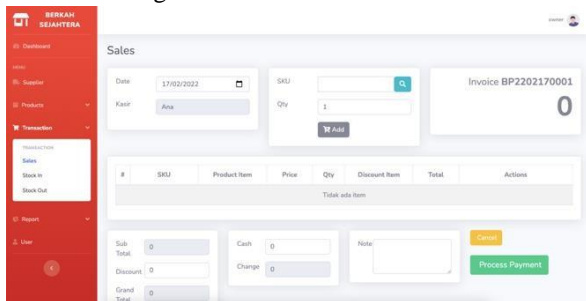
Gambar 16 Mockup menu stock in



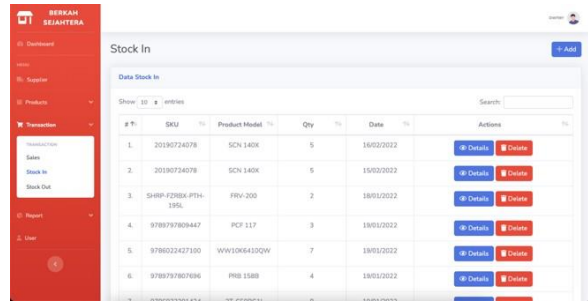
Gambar 17 Mockup menu stock out

I. Construction

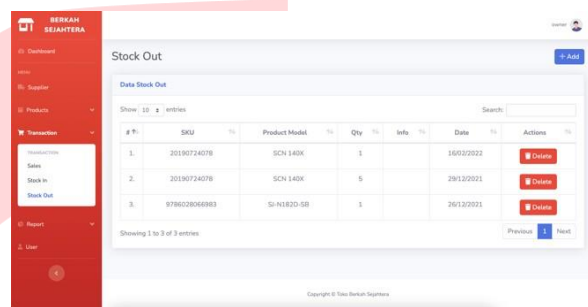
Pada tahap ini dilakukan proses penyusunan source code sistem berdasarkan desain yang telah dikembangkan sebelumnya. Berikut merupakan hasil rancangan sistem ini.



Gambar 18 Tampilan menu sales



Gambar 19 Tampilan menu stock in



Gambar 20 Tampilan menu stock out

J. Cutover

Pengujian pertama yang dilakukan adalah menggunakan metode black box testing. Pengujian ini dilakukan oleh penulis dengan menyediakan beberapa test case untuk menilai apakah sistem telah berjalan sesuai dengan yang diharapkan. Berikut merupakan hasil pengujian dengan black box testing.

Tabel 5 Black box testing

Menu	Test case	Hasil yang Diharapkan	Status Pengujian
Login	Memasukkan username dan password yang benar	Menampilkan dashboard dengan menu sesuai dengan hak akses peran	BERHASIL
	Memasukkan username dan password yang salah	Menampilkan alert bahwa login gagal dan kembali ke login page	BERHASIL
Logout	Klik logout	Menampilkan login page	BERHASIL
Dashboard	Menambahkan atau mengurangi data jumlah item, supplier, user, dan stok	Menampilkan data sesuai dengan jumlahnya	BERHASIL
Supplier	Menambahkan data supplier baru	Data tersimpan di basis data	BERHASIL

Menu	Test case	Hasil yang Diharapkan	Status Pengujian
Categories	Memperbarui data <i>supplier</i>		BERHASIL
	Menghapus data <i>supplier</i>	Data terhapus dari basis data	BERHASIL
	Menambahkan data kategori baru	Data tersimpan di basis data	BERHASIL
	Memperbarui data kategori		BERHASIL
	Menghapus data kategori	Data terhapus dari basis data	BERHASIL
	Menambahkan data item baru dengan <i>barcode scanner</i>	Data tersimpan di basis data	BERHASIL
	Memperbarui data item		BERHASIL
Items	Menghapus data item	Data terhapus dari basis data	BERHASIL
	Memasukkan nilai <i>random</i> di kolom <i>barcode</i>	Dapat menampilkan <i>barcode</i> sesuai nilai yang dimasukkan	BERHASIL
	Klik tombol <i>print barcode</i>	Dapat mencetak <i>barcode</i>	BERHASIL
	Memasukkan <i>barcode</i> di kolom <i>search</i> dengan <i>barcode scanner</i>	Dapat menampilkan data item sesuai <i>barcode</i>	BERHASIL
Sales	Menambahkan data item yang ingin dibeli dengan <i>barcode scanner</i>	Data tersimpan di basis data	BERHASIL
Stock in	Menambahkan data <i>stock in</i> dengan <i>barcode scanner</i>	Data tersimpan di basis data	BERHASIL
	Melihat detail data <i>stock in</i>	Menampilkan detail data	BERHASIL
	Menghapus data <i>stock in</i>	Data terhapus dari basis data	BERHASIL
Stock out	Memasukkan <i>barcode</i> di kolom <i>search</i> dengan <i>barcode scanner</i>	Dapat menampilkan data item sesuai <i>barcode</i>	BERHASIL
	Menambahkan data <i>stock out</i> dengan <i>barcode scanner</i>	Data tersimpan di basis data	BERHASIL
	Menghapus data <i>stock out</i>	Data terhapus dari basis data	BERHASIL
Users	Memasukkan <i>barcode</i> di kolom <i>search</i> dengan <i>barcode scanner</i>	Dapat menampilkan data item sesuai <i>barcode</i>	BERHASIL
	Menambahkan data <i>user</i> baru	Data tersimpan di basis data	BERHASIL
	Memperbarui data <i>user</i>		BERHASIL
	Menghapus data <i>user</i>	Data terhapus dari basis data	BERHASIL

Berdasarkan hasil dari Tabel 5, semua *test case* sesuai dengan hasil yang diharapkan. Maka dari itu dapat disimpulkan bahwa sistem *inventory* yang dikembangkan telah lolos pengujian.

Selanjutnya dilakukan validasi kepada pengguna sistem yang terdiri dari satu pemilik toko dan satu pegawai. Pengguna dipersilakan untuk mengoperasikan sistem, kemudian dapat menilai sistem menggunakan kuesioner UAT. Berikut merupakan hasil rekapitulasi kuesioner UAT.

Tabel 6 Hasil Rekapitulasi Kuesioner UAT

Karakteristik	Nomor	Frekuensi Jawaban				
	Pertanyaan	1	2	3	4	5
Functional suitability	1	0	0	0	1	1
	2	0	0	0	0	2
	3	0	0	0	0	2
Performance efficiency	1	0	0	0	0	2
	2	0	0	0	1	1
Operability	1	0	0	0	0	2
	2	0	0	0	0	2
	3	0	0	0	2	0
Reliability	4	0	0	0	2	0
	5	0	0	0	1	1
	6	0	0	0	0	2
Reliability	1	0	0	0	0	2
	2	0	0	0	0	2

Berdasarkan hasil kuesioner penilaian UAT, dilakukan pengolahan data untuk mengetahui besar persentase penilaian masing-masing karakteristik. Berikut merupakan hasil perhitungan UAT.

Tabel 7 Pehitungan UAT

Karakteristik	Nomor Pertanyaan	Frekuensi Jawaban					Skor	Total Skor	%
		STS (1)	TS (2)	KS (3)	S (4)	SS (5)			
Functional suitability	1	0	0	0	1	1	9		
	2	0	0	0	0	2	10	29	96.67
	3	0	0	0	0	2	10		
Performance efficiency	1	0	0	0	0	2	10		
	2	0	0	0	1	1	9	19	95
Operability	1	0	0	0	0	2	10	55	91.67

Karakteristik	Nomor Pertanyaan	Frekuensi Jawaban					Total % Skor
		STS (1)	TS (2)	KS (3)	S (4)	SS (5)	
	2	0	0	0	0	2	10
	3	0	0	0	2	0	8
	4	0	0	0	2	0	8
	5	0	0	0	1	1	9
	6	0	0	0	0	2	10
	1	0	0	0	0	2	10
Reliability	1	0	0	0	0	2	10
	2	0	0	0	1	1	9
						20	100

Berdasarkan hasil perhitungan UAT, diketahui bahwa karakteristik yang mendapatkan persentase penilaian tertinggi adalah *reliability*, yaitu 100%. Sedangkan karakteristik *functional suitability* mendapatkan penilaian 96.67%, *performance efficiency* mendapatkan penilaian 95%, dan karakteristik *operability* mendapatkan penilaian 91.67%. Selanjutnya hasil perhitungan UAT diinterpretasikan menurut kriteria interpretasi skor berikut [7].

Tabel 8 Kriteria Interpretasi Skor

Rentang Persentase	Keterangan
0% - 20%	Sangat Tidak Setuju
21% - 40%	Tidak Setuju
41% - 60%	Kurang Setuju
61% - 80%	Setuju
81% - 100%	Sangat Setuju

Berdasarkan hasil perhitungan di Tabel 7, persentase penilaian dari responden berada di rentang 81%-100%. Maka dari itu, dapat disimpulkan bahwa sistem *inventory* dapat diterima oleh pengguna.

V. KESIMPULAN

Kesimpulan dari penelitian ini adalah menghasilkan rancangan sistem *inventory* yang

dikontrol oleh *barcode scanner* untuk memperbaiki dan mengembangkan sistem *inventory* di Toko Berkah Sejahtera. Sistem *inventory* ini dapat digunakan untuk melakukan pendataan barang masuk dan keluar, mengecek stok barang, memberikan laporan penjualan, laporan barang terlaris, dan laporan barang yang cepat habis.

REFERENSI

- [1] R. A. Wibowo, "Sistem Informasi Persediaan Keluar Masuk Barang Pada Inside Distro Jakarta," *Journal Speed - Sentra Penelitian Engineering dan Edukasi*, vol. 1, no. 4, pp. 19-24, 2009.
- [2] D. Kurniawan, R. Andrian and N. Y. Utami, "Sistem Inventory Jurusan Ilmu Komputer di Universitas Lampung," *Jurnal Komputasi*, vol. 2, no. 2, pp. 18-26, 2014.
- [3] G. B. Shelly and H. J. Rosenblatt, *Systems Analysis and Design*, Boston: Cengage Learning, 2012.
- [4] H. G. Daellenbach and D. C. McNickle, *Management science, Decision making through systems thinking*, Palgrave Macmilan, 2005.
- [5] H. Kaur and P. Singh, "UML (Unified Modeling Language): Standard Language for Software Architecture Development," *International Symposium on Computing, Communication, and Control*, pp. 118-125.
- [6] S. S. Alhir, "Understanding the Unified Modeling Language (UML)," Martinig & Associates, 1999.
- [7] B. Priyatna, A. L. Hananto and M. Nova, "Application of UAT (User Acceptance Test) Evaluation Model in Minggon E-Meeting Software Development," *SYSTEMATICS*, pp. 110-117, 2020.