



Implementasi Konsep *Maximum Capacity* pada *Warehouse Information System* Berbasis Web

Asri Mulyani¹, Ridwan Setiawan², Sahdan Hadianto³

Jurnal Algoritma
Institut Teknologi Garut
Jl. Mayor Syamsu No. 1 Jayaraga Garut 44151 Indonesia
Email : jurnal@itg.ac.id

¹asri.mulyani@itg.ac.id
²ridwan.setiawan@itg.ac.id
³1706067@itg.ac.id

Abstrak – Peran gudang sangat penting terhadap berjalannya sebuah usaha, dengan pengelolaan gudang yang baik diharapkan mampu memberikan keuntungan yang lebih maksimal kepada pengusaha. Salah satunya pengaturan tata letak dan kondisi yang baik dalam gudang yaitu dengan memanfaatkan ruang penyimpanan secara maksimum meskipun barang disimpan secara *vertical* atau *horizontal*. Tujuan penelitian ini untuk merancang Sistem Informasi Gudang Berbasis Web yang dapat menyajikan informasi mengenai *volume* rak yang tersedia untuk memaksimalkan rak yang ada, membantu pengguna dalam proses pencarian barang dan memberikan informasi mengenai stok barang yang akan habis atau kosong. Metodologi penelitian yang digunakan adalah *Rational Unified Process* (RUP) yang pelaksanaannya hanya meliputi tahapan *Inception*, *Elaboration*, dan *Construction* dengan pemodelan menggunakan *Unified Modelling Language* (UML), selanjutnya pengujian *Alpha* dengan menggunakan metode *Black-Box Testing*. Hasil penelitian ini berupa rancangan Sistem Informasi Gudang dengan hak akses multi *user* dan sistem berbasis web dengan menerapkan konsep *Maximum Capacity*. Sistem yang dirancang dapat memberikan informasi mengenai *volume* rak yang tersedia, memberikan informasi mengenai stok barang yang akan habis atau kosong, dan sistem yang dihasilkan juga telah dilengkapi dengan fitur pencarian untuk memudahkan *user* dalam proses pencarian posisi barang.

Kata Kunci – *Information System; Maximum Capacity; Warehouse.*

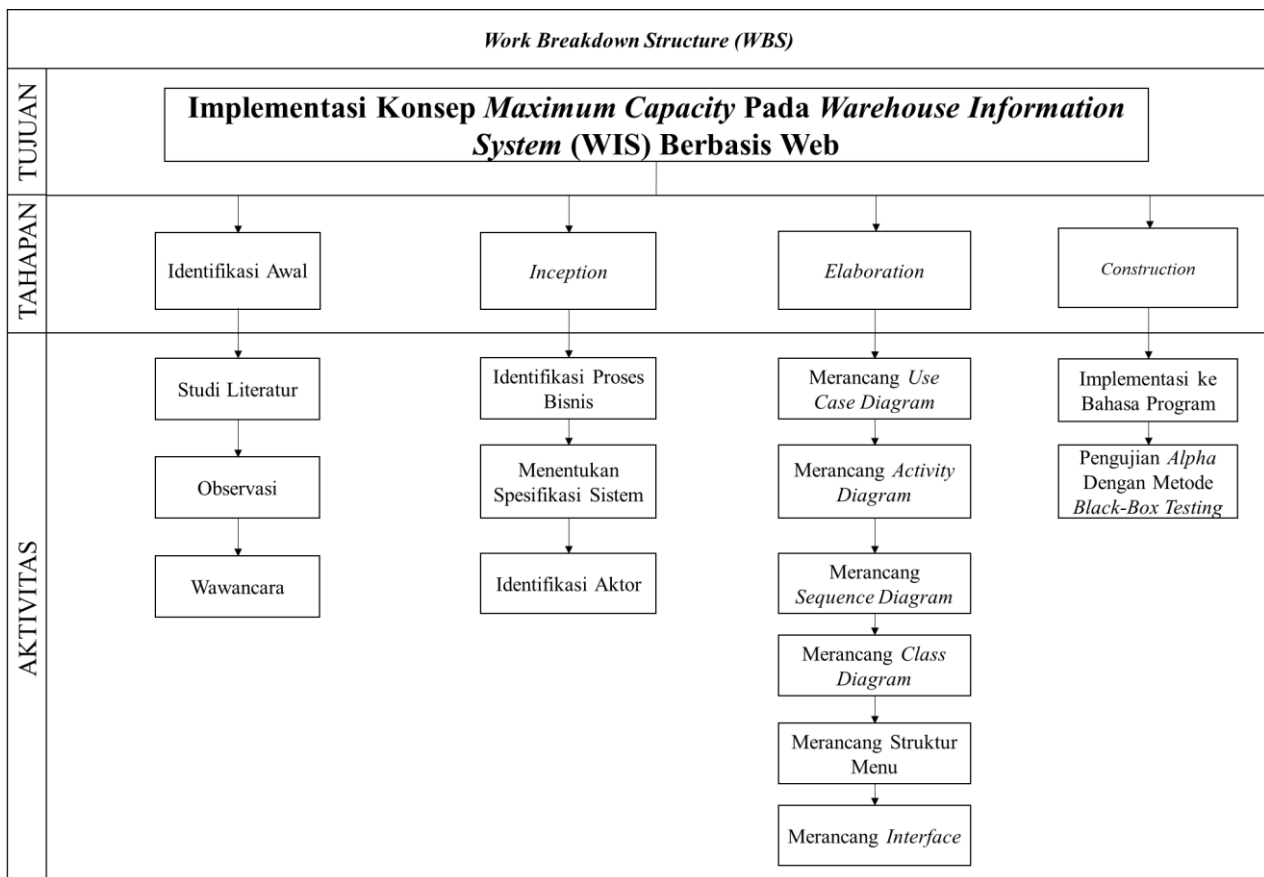
I. PENDAHULUAN

Gudang merupakan suatu bagian dalam sebuah pabrik yang berfungsi untuk menyimpan berbagai jenis barang dalam berbagai ukuran mulai dari yang kecil sampai barang yang berukuran besar yang dapat disimpan dalam jangka waktu tertentu. Pemanfaatan kapasitas ruangan yang kurang maksimum dapat menyebabkan banyaknya ruang yang tidak terpakai sehingga dapat membuat peran gudang kurang optimal. Hal ini terjadi pada gudang penyimpanan salah satu Toko Perlengkapan Olahraga di Garut, hasil dari observasi dan wawancara dengan pengelola toko, kurangnya pemanfaatan ruang gudang secara maksimum menyebabkan banyaknya ruang yang tidak terisi dan barang tidak tertampung dalam rak-rak penyimpanan sehingga barang tersebut disimpan di ruas-ruas jalan menuju gudang. Selain itu kebijakan penyimpanan barang secara *randomized* yaitu setiap barang yang datang akan diletakkan secara acak pada lokasi penyimpanan mana pun dalam gudang yang tidak diikuti dengan sistem informasi yang mendukung sehingga saat pencarian barang pun memakan waktu lama karena tertumpuk dengan barang lain dan terkadang lupa posisi menyimpan barang berada di sebelah mana, tidak adanya informasi mengenai stok yang akan habis atau kosong, hingga laporan barang masuk dan barang keluar yang masih *manual* ditulis dibuku yang sangat riskan memungkinkan untuk datanya hilang dan rusak serta harus merekap ulang data barang fisik untuk laporannya.

Hal inilah yang mendasari perlunya Sistem Informasi Gudang yang mampu memberikan solusi terhadap permasalahan yang ada, dengan menerapkan konsep *maximum capacity* yang merupakan konsep untuk memaksimalkan muatan ruang penyimpanan yang tersedia, *volume* dalam maksimum wadah atau kemasan termasuk wadah besar menengah dan kemasan besar yang dinyatakan dalam ukuran meter kubik [1]. Adapun beberapa penelitian sebelumnya telah membahas mengenai sistem informasi gudang, di antaranya penelitian pertama yang hanya berfokus untuk mengelola dan melihat informasi aliran barang masuk dan keluar, mengontrol pergerakan dan penyimpanan barang di gudang dan memproses transaksi terkait [2], [3]. Penelitian yang lainnya berfokus pada perbaikan tata letak gudang yang lebih efektif [4]–[6]. Selanjutnya penelitian yang bertujuan hanya untuk dapat mengelola dan melakukan pencarian data barang dengan cepat serta dapat memberikan informasi yang akurat [7]–[9]. Kemudian penelitian yang hanya berfokus untuk mengetahui barang yang tersedia, pengelolaan stok barang, menentukan stok barang yang optimal, dan meningkatkan efisiensi fungsi gudang [10], [11]. Terakhir penelitian yang bertujuan hanya untuk pengelolaan barang inventaris sehingga pembuatan laporan data barang inventaris menjadi lebih mudah [12]. Berdasarkan penelitian sebelumnya, fitur yang disajikan di dalam pengembangan sistem nya belum terdapat sebuah konsep yang dengan khusus membahas bagaimana dalam proses pengelolaan ruang penyimpanan barangnya. Maka dari itu, tujuan dari penelitian ini yaitu mengembangkan sebuah sistem informasi gudang dengan menambah fitur yang belum ada pada penelitian sebelumnya. Dimana pada penelitian ini menerapkan sebuah konsep *maximum capacity* yang disajikan dibagian dashboard untuk mengelola ruang penyimpanan.

II. URAIAN PENELITIAN

Pada penelitian ini menggunakan metodologi *Rational Unified Process* [13] dan pemodelan untuk penelitian ini menggunakan *Unified Modeling Language* [14]. Tahapan yang akan digambarkan pada perancangan sistem informasi ini mencakup pembuatan *diagram* yang menggambarkan dan membantu menjelaskan tahapan-tahapan di dalam sistem informasi. Langkah kerja yang dilakukan selama penelitian ini di gambarkan dengan menggunakan *Work Breakdown Structure* (WBS).



Gambar 1: *Work Breakdown Structure* Penelitian

A. Identifikasi Awal

Tahapan pertama pada WBS ini adalah identifikasi awal, kegiatan yang dilakukannya meliputi studi literatur, observasi dan wawancara. Hasil dari tahapan ini berupa penerapan konsep *Maximum Capacity* pada sistem yang akan dibangun, judul penelitian, dan metodologi yang akan dilakukan pada penelitian ini.

B. Inception

Tahapan yang kedua adalah *inception*, pada aktivitas ini di fokuskan untuk kegiatan pengumpulan data yang berkaitan dengan kebutuhan sistem yang akan dibuat, kegiatan yang dilakukan dimulai dari identifikasi proses bisnis, menentukan spesifikasi sistem, dan mengidentifikasi aktor.

C. Elaboration

Tahapan yang ketiga adalah *elaboration*, pada aktivitas ini kegiatan yang dilakukan ialah merancang *use case diagram*, merancang *activity diagram*, merancang *sequence diagram*, merancang *class diagram*, merancang struktur menu, dan merancang *interface*.

D. Construction

Tahapan yang keempat adalah *construction*, pada aktivitas ini melakukan implementasi teori konsep *maximum capacity* dan perancangan sistem ke dalam bahasa pemrograman dan pengujian *alpha* dengan menggunakan metode *Black-Box Testing*.

III. HASIL DAN DISKUSI

A. Hasil Penelitian

1. Identifikasi Awal

Pada tahapan ini meliputi studi literatur, observasi dan wawancara. Studi literatur ini merujuk pada penelitian sebelumnya dan pemahaman teori dari konsep *maximum capacity*, selanjutnya observasi ke salah satu Toko Perlengkapan Olahraga untuk analisis kebutuhan perancangan sistem, aktivitas ini melakukan pengamatan dan analisa yang dilakukan dengan melihat dari sistem informasi yang sudah ada, serta mencari informasi dengan melakukan wawancara bersama pengelola toko.

2. Inception

Tahapan pertama yang dilaksanakan adalah tahapan *inception* yang melalui beberapa aktivitas sebagai berikut:

2.1 Identifikasi Proses Bisnis

Analisis aktivitas identifikasi proses bisnis dilakukan untuk mengetahui bagaimana proses bisnis yang sedang berlangsung di Toko Perlengkapan Olahraga, sehingga dapat memberikan gambaran untuk sistem yang akan dirancang. Identifikasi proses bisnis ini merupakan hasil dari aktivitas studi literatur, observasi dan wawancara yang sudah dilakukan.

2.2 Menentukan Spesifikasi Sistem

Dalam menentukan spesifikasi sistem maka terdapat rincian persyaratan suatu sistem yang akan dirancang. Rincian persyaratan suatu sistem menjelaskan mengenai bagaimana sistem yang akan dibangun berdasarkan kebutuhan adalah sebagai berikut:

a) Persyaratan Tampilan Sistem

Persyaratan tampilan sistem, ini harus memiliki tampilan awal yang mengandung bagian untuk admin dan *user* yaitu tampilan *login*, *dashboard* yang menyediakan informasi secara singkat dari seluruh barang yang tersedia, menyediakan tabel, dan *form* yang dapat berjalan dengan

- baik.
- b) **Persyaratan Sistem**
 Berdasarkan data yang telah di dapatkan dalam memenuhi kebutuhan dari pengguna, maka penelitian dengan judul *Implementasi Konsep Maximum Capacity Pada Warehouse Information System (WIS) Berbasis Web* yang akan dibangun memiliki spesifikasi sebagai berikut:
- 1) Spesifikasi Fungsional Sistem, untuk admin menyediakan fitur untuk pembuatan akun *user*, menghapus *token user*, mengelola barang, barang masuk, barang keluar, mencetak laporan, mengelola rak, rak yang terisi, *supplier*, dan kategori barang. Sistem untuk *user* menyediakan fitur untuk mengelola barang, barang masuk, barang keluar, mencetak laporan, mengelola rak, rak yang terisi, *supplier*, dan kategori barang.
 - 2) Spesifikasi Non Fungsional Sistem, *warehouse information system* ini berfokus pada perekaman barang baik dari barang masuk maupun barang keluar yang memudahkan *user* atau pekerja gudang untuk menyimpan barang bebas di mana saja mau sesuai kategori ataupun tidak mau *vertical* atau *horizontal* tidak masalah. Konsep yang diterapkan pada *warehouse information system* ini ialah memaksimalkan kapasitas rak yang tersedia. Sehingga pada konsep ini mengharuskan agar rak bisa terisi penuh (*maximum*) atau tidak ada ruang yang kosong. *Warehouse information system* ini dibuat dengan *platform website* menggunakan bahasa pemrograman *JavaScript* dengan *library ReactJs* dan *android* untuk membantu proses perekaman sebagai piranti *scan* dengan menggunakan *ReactNative*.

2.3. Mengidentifikasi Aktor

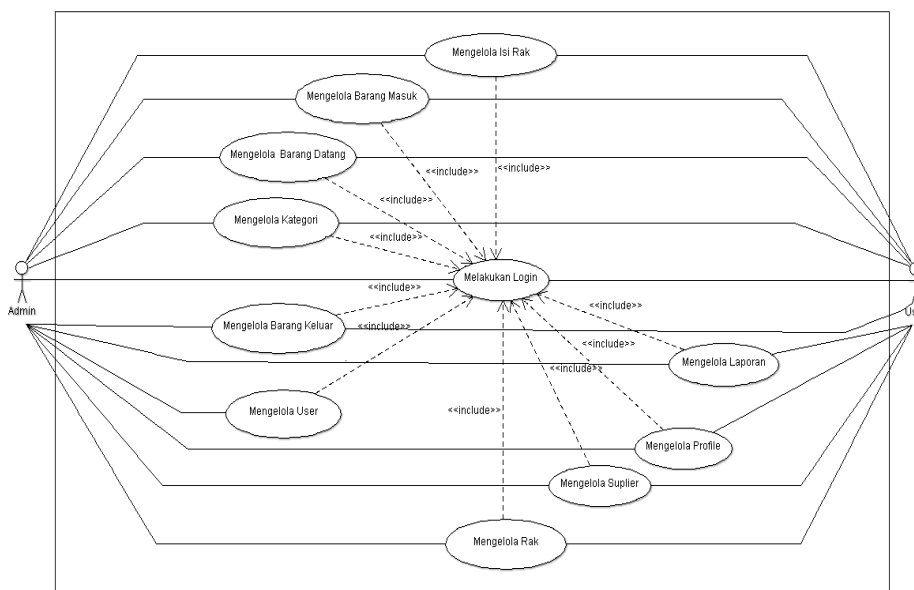
Menentukan aktor dalam membangun sebuah sistem merupakan suatu bagian penting yang harus dilakukan, karena penentuan aktor dalam sebuah sistem yang akan dibangun merupakan peran yang dimainkan oleh pengguna suatu sistem untuk dapat berinteraksi serta berkaitan dengan sistem yang akan digunakan. Adapun aktor yang ada di dalam *warehouse information system* ini adalah admin dan *user*. Berikut spesifikasi aktor yang disajikan pada Gambar 2.

3. Elaboration

Pada tahapan *elaboration* ini merancang arsitektur sistem sesuai dengan aktivitas yang terdapat di WBS yaitu merancang *use case diagram*, merancang *activity diagram*, merancang *sequence diagram*, merancang *class diagram*, merancang struktur menu, dan merancang *interface* sebagai lanjutan dari tahapan sebelumnya.

3.1. Merancang Use Case Diagram

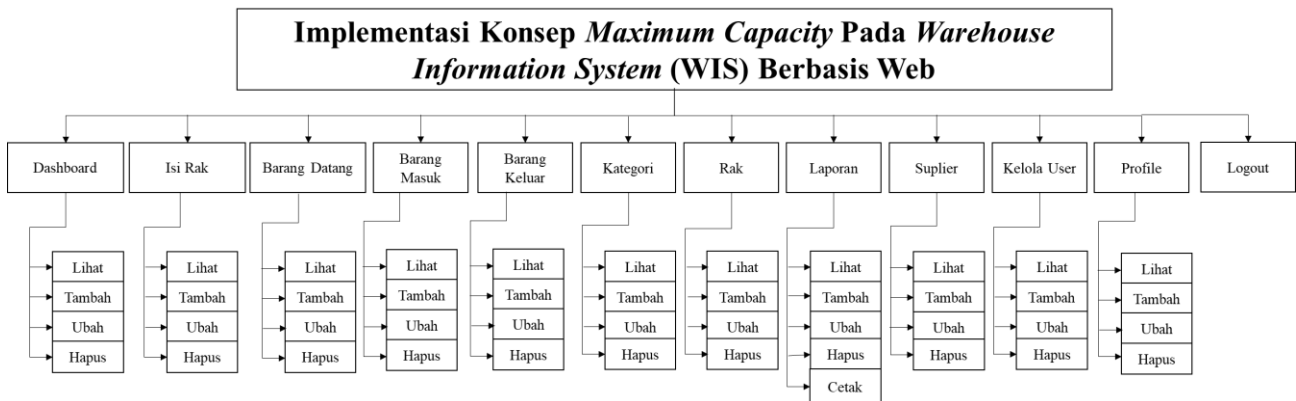
Perancangan *Use case diagram* bertujuan untuk mengidentifikasi aktor yang terlibat dalam sistem. Hasil dari identifikasi aktor tersebut adalah sebagai berikut:



Gambar 2: Use Case Diagram Penelitian

3.2. Merancang Struktur Menu

Struktur menu merupakan salah satu dari tahapan *elaboration* dalam kebutuhan membangun suatu sistem. Struktur menu menggambarkan bagian-bagian menu yang terdapat dalam sistem yang dirancang. Adapun untuk struktur menu *Warehouse Information System* Berbasis Web adalah sebagai berikut:



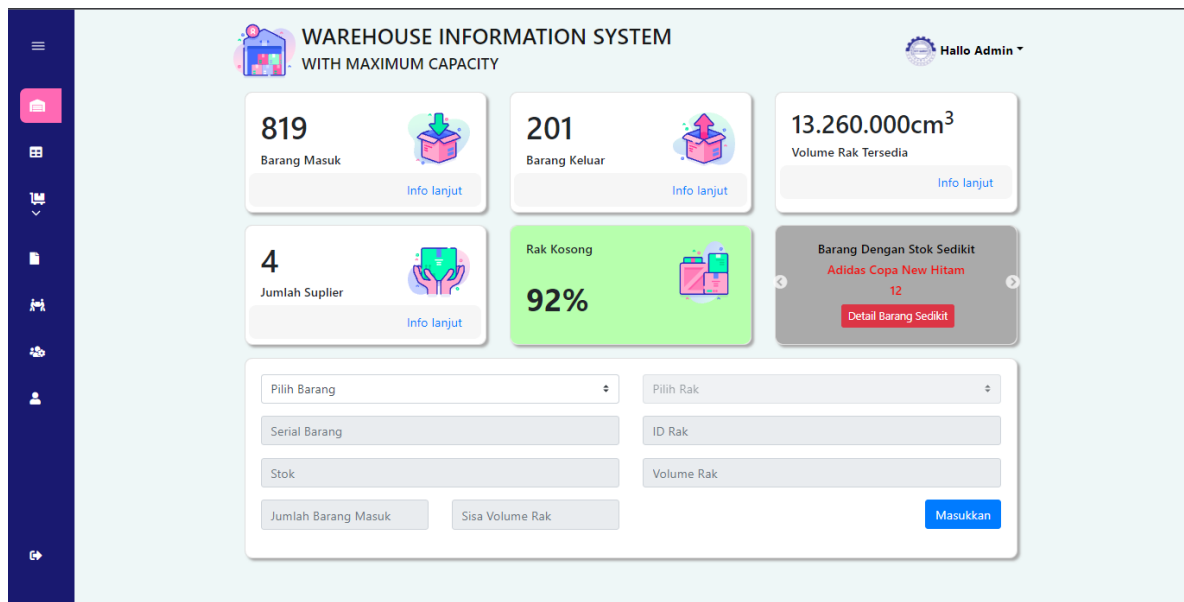
Gambar 3: Struktur Menu Admin

4. Construction

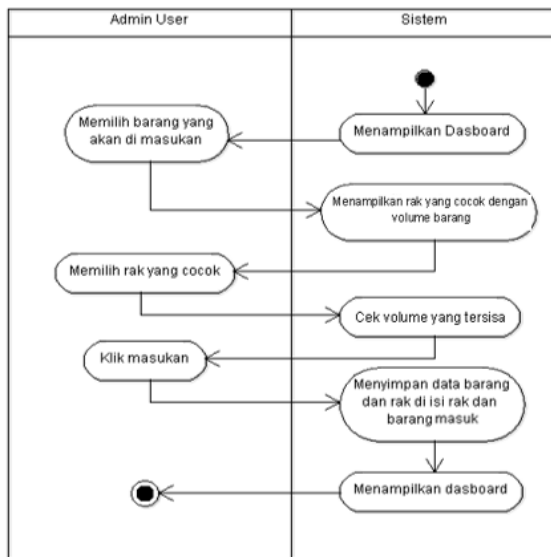
Dalam tahapan *construction* sebagai lanjutan dari tahapan sebelumnya. Implementasi rancangan perangkat lunak pada kode program dan pengujian *alpha* dengan metode *black-box testing* dilakukan pada tahapan *construction* ini.

4.1. Implementasi ke Bahasa Pemrograman

Setelah merancang *Interface* selanjutnya adalah implementasi konsep *maximum capacity* dan rancangan sistem pada kode program.



Gambar 4: Dashboard Warehouse Information System



Gambar 5: Proses Bisnis Sistem

Berikut *detail* aktivitas yang ada pada Gambar 5:

Tabel 1: *Detail* Aktivitas Proses Bisnis Sistem

No	Aktivitas	Deskripsi
1	Menampilkan halaman <i>Dashboard</i>	Admin atau <i>User</i> mempunyai status <i>login</i> , sehingga sistem menampilkan halaman <i>dashboard</i> .
2	Memilih barang yang akan di masukan	Admin atau <i>User</i> memilih barang yang akan di masukan ke dalam rak.
3	Menampilkan rak yang cocok dengan <i>volume</i> barang	Dengan menerapkan konsep <i>maximum capacity</i> sistem secara otomatis memproses data barang yang akan di masukan ke dalam rak, kemudian sistem menampilkan daftar rak yang masih memiliki ruang penyimpanan yang sesuai dengan <i>volume</i> barang yang akan dimasukkan ke dalam rak.
4	Memilih rak yang cocok	Setelah sistem menampilkan pilihan rak yang memiliki <i>volume</i> sesuai dengan barang yang ingin dimasukkan, maka Admin atau <i>User</i> memilih rak yang cocok dengan data <i>volume</i> barang yang akan di masukan.
5	Cek <i>volume</i> yang tersisa	Sistem memproses hasil dari inputan data yang telah admin masukan dan memberikan informasi mengenai sisa ruang rak yang tersisa yang dapat di isi oleh barang lainnya yang memiliki <i>volume</i> sama dengan isi rak, sehingga ruang rak dapat di maksimumkan kapasitas penyimpanannya.
6	Klik masukan	Admin memilih tombol masukan untuk di menyimpan datanya.
7	Menyimpan data barang dan data rak di isi rak dan barang masuk	Sistem menyimpan data barang dan data rak di halaman isi rak dan di halaman barang masuk.

4.2. Pengujian *Alpha* dengan Metode *Black Box Testing*

Pengujian *alpha* dengan menggunakan metode *black-box testing* merupakan suatu pengujian terhadap sistem informasi yang telah dirancang untuk mengetahui dan memenuhi apakah sistem informasi yang telah dirancang telah berjalan dengan sebagaimana mestinya [15].

Tabel 2: Deskripsi Pengujian Kelas-kelas Diagram

No	Aktivitas	Kelas Uji	Skenario Uji	Hasil yang diharapkan	Keterangan
1	<i>Login</i>	Melakukan <i>login</i> yang benar	Memasukkan <i>email</i> dan <i>password</i>	<i>Login</i> berhasil dan masuk ke halaman awal	Sesuai
		Melakukan <i>login</i> yang salah	Memasukkan <i>email</i> dan <i>password</i>	<i>Login</i> gagal dan kembali ke halaman <i>login</i>	Sesuai
2	Mengelola barang datang	Menambah barang	Memasukkan data ke <i>form</i> barang	Data barang berhasil di simpan ke <i>database</i>	Sesuai
		Mengubah barang	Memilih data barang yang akan diubah dan memasukkan data baru ke <i>form</i>	Data barang di <i>database</i> berhasil diubah	Sesuai
		Melihat barang	Data barang di tampilkan di halaman barang	Data barang berhasil tampil di halaman barang	Sesuai
		Mencari barang	Memasukkan kata kunci barang di <i>form</i> pencarian	Data barang berhasil dicari dan di tampilkan	Sesuai
		Menghapus barang	Memilih data barang yang akan dihapus	Data barang di <i>database</i> berhasil dihapus	Sesuai
3	Mengelola rak	Menambah rak	Memasukkan data ke <i>form</i> rak	Data rak berhasil di simpan ke <i>database</i>	Sesuai
		Mengubah rak	Memilih data rak yang akan diubah dan memasukkan data baru ke <i>form</i>	Data rak di <i>database</i> berhasil diubah	Sesuai
		Melihat rak	Data rak di tampilkan di halaman rak	Data rak berhasil tampil di halaman rak	Sesuai
		Mencari rak	Memasukkan kata kunci rak di <i>form</i> pencarian	Data rak berhasil dicari dan di tampilkan	Sesuai
		Menghapus rak	Memilih data rak yang akan dihapus	Data rak di <i>database</i> berhasil dihapus	Sesuai
4	Mengelola kategori	Menambah kategori	Memasukkan data ke <i>form</i> kategori	Data kategori berhasil di simpan ke <i>database</i>	Sesuai
		Mengubah	Memilih data	Data kategori di	Sesuai

No	Aktivitas	Kelas Uji	Skenario Uji	Hasil yang diharapkan	Keterangan
		kategori	kategori yang akan diubah dan memasukkan data baru ke <i>form</i>	<i>database</i> berhasil diubah	
		Melihat kategori	Data kategori di tampilkan di halaman kategori	Data kategori berhasil tampil di halaman kategori	Sesuai
		Menghapus kategori	Memilih data kategori yang akan dihapus	Data kategori di <i>database</i> berhasil dihapus	Sesuai
5	Mengelola <i>suplier</i>	Menambah <i>suplier</i>	Memasukkan data ke <i>form suplier</i>	Data <i>suplier</i> berhasil di simpan ke <i>database</i>	Sesuai
		Mengubah <i>suplier</i>	Memilih data <i>suplier</i> yang akan diubah dan memasukkan data baru ke <i>form</i>	Data <i>suplier</i> di <i>database</i> berhasil diubah	Sesuai
		Melihat <i>suplier</i>	Data <i>suplier</i> di tampilkan di halaman <i>suplier</i>	Data <i>suplier</i> berhasil tampil di halaman <i>suplier</i>	Sesuai
		Mencari <i>suplier</i>	Memasukkan kata kunci <i>suplier</i> di <i>form</i> pencarian	Data <i>suplier</i> berhasil dicari dan di tampilkan	Sesuai
		Menghapus <i>suplier</i>	Memilih data <i>suplier</i> yang akan dihapus	Data <i>suplier</i> di <i>database</i> berhasil dihapus	Sesuai
6	Mengelola isi rak	Menambah isi rak	Memasukkan data barang dan rak dengan menerapkan konsep <i>maximum capacity</i>	Data barang dan rak berhasil di simpan ke <i>database</i> isi rak	Sesuai
		Melihat isi rak	Data isi rak di tampilkan di halaman isi rak	Data isi rak berhasil tampil di halaman isi rak	Sesuai
		Mencari isi rak	Memasukkan kata kunci isi rak di <i>form</i> pencarian	Data isi rak berhasil dicari dan di tampilkan	Sesuai
		Mengeluarkan isi rak	Memilih data isi rak yang akan dikeluarkan	Data isi rak di <i>database</i> berhasil dikeluarkan	Sesuai
7	Mengelola barang masuk	Menambah barang masuk	Otomatis bertambah saat menambah data isi rak	Data barang masuk berhasil di simpan ke <i>database</i>	Sesuai
		Mengubah barang masuk	Memilih data barang masuk yang akan diubah dan memasukkan data baru ke <i>form</i>	Data barang masuk di <i>database</i> berhasil diubah	Sesuai
		Melihat barang	Data barang masuk	Data barang	Sesuai

No	Aktivitas	Kelas Uji	Skenario Uji	Hasil yang diharapkan	Keterangan
		masuk	di tampilkan di halaman barang masuk	masuk berhasil tampil di halaman barang masuk	
		Mencari barang masuk	Memasukkan kata kunci barang masuk di <i>form</i> pencarian	Data barang masuk berhasil dicari dan di tampilkan	Sesuai
		Menghapus barang masuk	Memilih data barang masuk yang akan dihapus	Data barang masuk di <i>database</i> berhasil dihapus	Sesuai
8	Mengelola barang keluar	Menambah barang keluar	Otomatis bertambah saat mengeluarkan data isi rak	Data barang keluar berhasil di simpan ke <i>database</i>	Sesuai
		Mengubah barang keluar	Memilih data barang keluar yang akan diubah dan memasukkan data baru ke <i>form</i>	Data barang keluar di <i>database</i> berhasil diubah	Sesuai
		Melihat barang keluar	Data barang keluar di tampilkan di halaman barang keluar	Data barang keluar berhasil tampil di halaman barang keluar	Sesuai
		Mencari barang keluar	Memasukkan kata kunci barang keluar di <i>form</i> pencarian	Data barang masuk keluar dicari dan di tampilkan	Sesuai
		Menghapus barang keluar	Memilih data barang keluar yang akan dihapus	Data barang keluar di <i>database</i> berhasil dihapus	Sesuai
9	Mengelola <i>user</i> (admin)	Menambah <i>user</i>	Memasukkan data ke <i>form user</i>	Data berhasil di simpan ke <i>database</i>	Sesuai
		Melihat <i>user</i>	Data akun di tampilkan di halaman <i>user</i>	Data akun berhasil tampil di halaman <i>user</i>	Sesuai
10	Mengelola laporan	Melihat laporan	Data barang masuk, data barang keluar, dan data setiap barang ditampilkan di halaman laporan	Data barang masuk, data barang keluar, dan data setiap barang ditampilkan di halaman laporan	Sesuai
		Mencetak laporan	Data barang masuk, data barang keluar, dan data setiap barang di cetak	Data barang masuk, data barang keluar, dan data setiap barang berhasil di cetak	Sesuai

No	Aktivitas	Kelas Uji	Skenario Uji	Hasil yang diharapkan	Keterangan
11	Mengelola <i>profile</i>	Mengubah <i>password user</i>	Memasukkan <i>password</i> lama dan baru di <i>form</i> ubah <i>password</i>	<i>Password</i> berhasil di diubah ke <i>database</i>	Sesuai
12	<i>Logout</i>	Melakukan <i>logout</i>	Memilih menu <i>logout</i>	<i>Logout</i> berhasil dan kembali ke halaman <i>login</i>	Sesuai

Hasil dari pengujian *alpha* dengan menggunakan metode *Black Box Testing*, *Warehouse Information System* dengan Konsep *Maximum Capacity* semua fitur yang ada telah berjalan dengan baik dan sesuai fungsi.

IV. KESIMPULAN

Berdasarkan dari hasil dan pembahasan penelitian yang dilakukan maka dapat disimpulkan, pada penelitian ini menghasilkan Sistem Informasi Gudang Berbasis Web yang berfokus pada proses barang masuk dan barang keluar yang dilengkapi dengan konsep *maximum capacity*, sehingga semua ruang rak yang ada dapat dimaksimalkan. Kemudian sistem yang dihasilkan juga telah di lengkapi dengan fitur pencarian yang dapat memberikan informasi mengenai lokasi barang yang ada di rak. Serta dilengkapi fitur informasi mengenai stok barang dalam rak yang akan habis atau kosong, sehingga memudahkan pengguna dalam mengelola persediaan barang. Pada penelitian ini penulis memberikan beberapa saran untuk penelitian selanjutnya yang dapat meningkatkan kinerja sistem. Pada sistem ini dapat digunakan untuk produk makanan atau obat-obatan dengan menambahkan fitur masa kadaluwarsa. Penambahan fitur *invoice* untuk mempermudah dalam proses penentuan pesanan kepada *supplier*. Sistem ini dapat dikembangkan dengan menambahkan *tools* tambahan seperti lampu pada setiap raknya, untuk mempermudah dalam proses pencarian barang.

DAFTAR PUSTAKA

- [1] COTIF, "RID Convention concerning International Carriage by Rail (COTIF) Appendix C- Regulations concerning International Carriage of Dangerous Goods by Rail," no. July 2012, 2015.
- [2] D. Hamidin, Santoso, and P. Mutianingsih, "Rancang Bangun Aplikasi Warehouse Berbasis Web Terintegrasi Dengan Qrcode," *J. Tek. Inform.*, vol. 10, no. 3, pp. 24–30, 2018.
- [3] R. Fauzan, M. F. Shiddiq, and N. R. Raddlya, "The Designing of Warehouse Management Information System," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 879, no. 1, 2020, doi: 10.1088/1757-899X/879/1/012054.
- [4] I. Sukoco, "Perancangan Tata Letak Gudang Di PT . Panatrade dengan menggunakan metode shared storage," pp. 1–75, 2017.
- [5] R. B. M. De Koster, A. L. Johnson, and D. Roy, "Warehouse design and management," *Int. J. Prod. Res.*, vol. 55, no. 21, pp. 6327–6330, 2017, doi: 10.1080/00207543.2017.1371856.
- [6] N. Ling, X. Wei, M. M. Ren, and S. H. Fan, "The design and development of warehouse management information system on Hongxing logistics," 2017, doi: 10.1109/CSA.2015.76.
- [7] A. Budiman and A. Mulyani, "Rancang Bangun Aplikasi Sistem Informasi Persediaan Barang di TB. Indah Jaya Berbasis Desktop," *J. Algoritma*, vol. 13, no. 2, pp. 374–378, 2017, doi: 10.33364/algoritma/v.13-2.374.
- [8] B. S. S. Tejesh and S. Neeraja, "Warehouse inventory management system using IoT and open source framework," *Alexandria Eng. J.*, vol. 57, no. 4, 2018, doi: 10.1016/j.aej.2018.02.003.
- [9] I. P. A. Diana and I. P. A. Pratama, "Designing Inventory Information Systems at UD. Miasa Desktop-Based," *ACSIE (International J. Appl. Comput. Sci. Inform. Eng.)*, vol. 1, no. 1, 2019, doi: 10.33173/acsie.36.
- [10] W. Baswardono, "Analisa dan Perancangan Warehouse Inventory System Untuk UMKM Berbasis

- Multi Tenant,” *J. Algoritm.*, vol. 15, no. 2, pp. 67–78, 2019, doi: 10.33364/algoritma/v.15-2.67.
- [11] R. Caridade, T. Pereira, L. Pinto Ferreira, and F. J. G. Silva, “Analysis and optimisation of a logistic warehouse in the automotive industry,” *Procedia Manuf.*, vol. 13, pp. 1096–1103, 2017, doi: 10.1016/j.promfg.2017.09.170.
- [12] R. Setiawan and A. Ikhwana, “Pengembangan Aplikasi Pengelolaan Inventaris Barang Berbasis Web Di Sekolah Tinggi Teknologi Garut,” *J. Algoritm.*, vol. 14, no. 2, pp. 452–462, 2015, doi: 10.33364/algoritma/v.14-2.452.
- [13] C. Péraire, M. Edwards, A. Fernandes, E. Mancin, and Kathy, *Front cover The IBM Rational Unified Process*. 2007.
- [14] R. A. Sukamto and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur*. 2018.
- [15] B. Beizer and J. Wiley, “Black Box Testing: Techniques for Functional Testing of Software and Systems,” *IEEE Softw.*, vol. 13, no. 5, 2005, doi: 10.1109/ms.1996.536464.