

SISTEM INFORMASI PERSEDIAAN PRODUK MAKANAN RINGAN BERBASIS WEB (STUDI KASUS: D'CARULUK BOGOR)

Hans Daniel William Tucunan, Nono Heryana

Program Studi Sistem Informasi, Fakultas Ilmu Komputer
Universitas Singaperbangsa Karawang, Jl. HS. Ronggo Waluyo Karawang, Indonesia
1910631250046@student.unsika.ac.id

ABSTRAK

D'Caruluk merupakan perusahaan penjualan makanan ringan yang berlokasi di daerah Bogor. Pada kali ini penulis menganalisis sistem persediaan barang yang D'Caruluk jalankan saat ini. Sistem persediaan produk makanan ringan yang belum terkomputerisasi, pencatatan bukti transaksi untuk penerimaan barang dan pengeluaran barang yang masih tercampur, mengakibatkan sering terjadinya kehilangan dan kerusakan barang. Metode yang digunakan adalah *waterfall*. Sistem yang dibangun memanfaatkan PHP sebagai bahasa pemrograman dan MySQL sebagai *database* dalam pembentukan sistem informasi tersebut. Hasil dari penelitian menerangkan bahwa perancangan sistem persediaan barang berbasis web dapat menolong pihak D'Caruluk dalam mendapatkan informasi terhadap transaksi barang dan laporan persediaan barang. Langkah terakhir adalah menguji aplikasi tersebut.

Kata kunci: Sistem Informasi, Persediaan Barang, Waterfall, Web

1. PENDAHULUAN

Pesatnya kemajuan ilmu pengetahuan dan teknologi sekarang memudahkan dan mempercepat untuk mendapatkan data dan informasi, namun keakuratan data juga harus diperhatikan. Sistem persediaan (*inventory*) dalam lingkungan produksi merupakan hal yang perlu diperhatikan perusahaan dalam proses pabrikasi [1]. Ketepatan informasi dan data komoditas ketika produksi menjadi catatan penting karena akan digunakan sebagai laporan informasi akuntansi.

Saat ini, bisnis di Indonesia sedang berkembang pesat. Usaha kuliner (minuman dan makanan), contohnya seperti jenis usaha makanan ringan. Akan tetapi saat ini D'Caruluk Bogor masih menerapkan sistem secara manual dalam proses pencatatan barang masuk dan keluar, dalam proses bisnis bagian gudang mencatat setiap kali penerimaan barang dan pengeluaran barang dengan surat jalan dan surat keterangan transaksi. Kemudian diringkas dan dimasukkan ke dalam *Microsoft Excel* untuk membuat laporan data persediaan bulanan, namun permasalahan yang muncul adalah sulitnya pencarian stok barang dan sulitnya pengolahan data.

Dengan perkembangan pengetahuan dan perkembangan teknologi yang begitu pesat setiap tahunnya, hal inilah yang menjadi dorongan bagi penulis untuk membangun aplikasi dimana informasi mengenai persediaan barang (*inventory*) dapat diakses secara online. Aplikasi ini juga memberikan kemudahan pada penginputan data penerimaan barang dan pengeluaran barang, yang bisa diakses kapanpun dan dimanapun, yang tentunya lebih efektif dan efisien.

2. TINJAUAN PUSTAKA

2.1. Sistem Informasi

Sistem informasi adalah sekumpulan komponen yang saling berinteraksi yang digunakan untuk mengumpulkan, menyimpan, menganalisis, dan menyebarkan informasi untuk mendukung operasi dan pengambilan keputusan dalam suatu organisasi. Sistem informasi meliputi *hardware*, *software*, data, prosedur, dan orang-orang yang digunakan untuk mengumpulkan, menyimpan, menganalisis, dan menyebarkan informasi [2].

2.2. Web

Web adalah sebuah jaringan global yang terdiri dari dokumen-dokumen yang saling terhubung melalui tautan (*hyperlink*) dan dapat diakses menggunakan browser web [3]. Dokumen-dokumen tersebut dapat berupa halaman web, gambar, video, atau file lain yang diformat dengan bahasa web seperti HTML, CSS, dan JavaScript. Web dikembangkan oleh Tim Berners-Lee pada tahun 1989 sebagai cara untuk mengakses informasi melalui internet. Saat ini, web digunakan untuk berbagai keperluan, seperti komunikasi, belanja, hiburan, pendidikan, dan lainnya [4].

2.3. Metode Waterfall

Waterfall adalah sebuah metode pengembangan perangkat lunak yang mengikuti proses linier dan berurutan. Metode ini mengikuti tahapan-tahapan yang telah ditentukan sebelumnya, dimulai dari analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan. Setiap tahap harus selesai sebelum melangkah ke tahap berikutnya [5].

Pada tahap analisis kebutuhan, akan ditentukan spesifikasi sistem yang diinginkan, yang akan menjadi

dasar untuk desain sistem. Pada tahap desain, arsitektur sistem akan ditentukan dan dirancang. Pada tahap implementasi, sistem akan dikembangkan sesuai dengan desain yang telah ditentukan. Pada tahap pengujian, sistem akan diuji untuk memastikan bahwa sistem sesuai dengan spesifikasi yang telah ditentukan dan bekerja dengan baik. Pada tahap pemeliharaan, sistem akan diperbaiki dan diperbaharui sesuai dengan perubahan yang diperlukan.

Kelebihan dari metode *waterfall* adalah proses yang jelas dan terstruktur, sehingga memudahkan perencanaan dan pengendalian proyek [6]. Namun, metode ini memiliki beberapa kelemahan seperti kesulitan dalam menangani perubahan yang diperlukan setelah tahap analisis dan desain selesai. Dan juga kesulitan dalam mengukur progres proyek selama proses pengembangan.

2.4. Hypertext Preprocessor (PHP)

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang digunakan untuk mengembangkan aplikasi web dinamis. PHP dapat digunakan untuk menulis skrip yang akan dijalankan di server dan menghasilkan output HTML, XML, atau JSON. PHP dapat digabungkan dengan HTML, CSS, dan JavaScript untuk menciptakan halaman web yang interaktif dan responsif [7]. PHP dapat digunakan dengan berbagai jenis *database*, seperti MySQL, PostgreSQL, dan Oracle, sehingga memungkinkan untuk menyimpan dan mengambil data dari *database*.

PHP juga memiliki berbagai *library* dan *framework* yang dapat digunakan untuk meningkatkan produktivitas dan mempermudah pengembangan aplikasi web, seperti Laravel, CodeIgniter, dan Yii. PHP dapat dijalankan pada berbagai sistem operasi, seperti Windows, Linux, dan macOS, dan dapat diintegrasikan dengan server web seperti Apache dan Nginx. PHP juga merupakan bahasa pemrograman yang *open-source*, artinya bahwa kode sumbernya dapat diakses dan dimodifikasi oleh siapa saja [8].

2.5. Black Box Testing

Black box testing adalah sebuah metode pengujian perangkat lunak yang hanya menguji sistem dari luar tanpa memperhatikan implementasi teknis di dalam sistem. Dalam *black box testing*, tester tidak perlu memahami kode sumber sistem atau bagaimana sistem bekerja secara internal [9]. Tester hanya perlu mengetahui spesifikasi sistem dan input-output yang diharapkan. *black box testing* digunakan untuk menguji sistem dari perspektif pengguna, yang berarti tester hanya akan menguji sistem seperti apa yang akan digunakan oleh pengguna akhir. Tester akan menguji sistem dengan memberikan input yang diharapkan dan memastikan bahwa output yang dihasilkan sesuai dengan yang diharapkan.

Black box testing dapat digunakan untuk menguji berbagai jenis sistem, seperti aplikasi desktop, aplikasi web, dan sistem embedded. Ini digunakan untuk menguji fungsionalitas sistem, kompatibilitas sistem,

keandalan sistem, dan usabilitas sistem. *Black box testing* termasuk dalam *Functional Testing*, yang menguji sistem dari sudut pandang fungsionalitas. Dan tidak termasuk dalam *Non-Functional Testing* seperti *performance*, *security*, dan lain-lain [10].

3. METODE PENELITIAN

Dalam melengkapi eksplorasi ini, penulis menerapkan dua bagian solusi yaitu pengumpulan dari data serta model dari pengembangan sistem.

3.1. Metode Pengumpulan Data

a. Penelitian Langsung

Metode yang penulis lakukan adalah pengumpulan data melalui pengamatan langsung terhadap seluruh aktivitas pada sistem *inventory* yang dijalankan oleh D'Caruluk, yang nantinya akan dikaji guna mengidentifikasi proses dari bisnis serta kekurangan pada aktivitas yang dijalankan.

b. Wawancara

Pada tahap ini, mengumpulkan data dengan melakukan percakapan langsung kepada Bapak Sutiman (pemilik D'Caruluk) untuk mengumpulkan informasi dan dokumen lengkap mengenai sistem *inventory* D'Caruluk.

c. Studi Pustaka

Tahap berikutnya, penulis juga melaksanakan penelitian kepustakaan guna menunjang serta memenuhi informasi yang diperoleh sebagai acuan dari jurnal dan buku, sehingga lebih relevan dengan pokok bahasan atau topik yang diangkat.

3.2. Metode Pengembangan Sistem

Model yang akan diterapkan dalam eksplorasi kali ini melalui penerapan model *waterfall*.

a. Analisis Kebutuhan Sistem

Pada tahap ini melakukan analisis kebutuhan dari sistem, kemudian wawancara atau lainnya untuk mengumpulkan informasi yang dibutuhkan. Informasi yang dikumpulkan dapat membantu dalam perancangan sistem.

b. Desain

Desain adalah deskripsi dari *software* yang diusulkan, seperti struktur data, arsitektur perangkat lunak, representasi antarmuka, dan proses pengkodean [11]. Dalam tahap ini, sistem dirancang dengan memanfaatkan tools dari UML (*Diagram Activity*, *Diagram Use case*, *Diagram Sequence*, *Diagram Class*, *Diagram Deployment*, dan *Diagram Component*).

c. Pembuatan Kode Program

Dalam langkah ini, penulis membangun program dengan memanfaatkan pemrograman terstruktur, phpMyAdmin sebagai basis data MySQL dan *Visual Studio Code* sebagai text editor. Pembuatan program ini didasarkan pada hasil desain tahap sebelumnya.

d. Pengujian (Testing)

Tahapan ini diterapkan guna menemukan kekurangan dari sistem yang dibangun dan untuk meyakinkan output yang dibuat sudah sesuai dengan

yang diharapkan. *Testing* kali ini menggunakan *black box testing*.

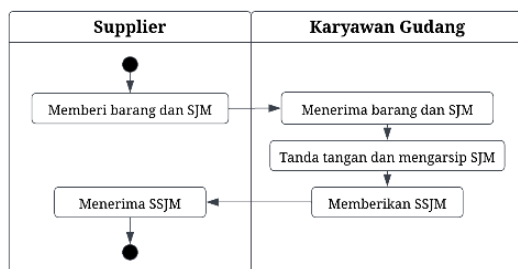
4. HASIL DAN PEMBAHASAN

4.1. Analisis Sistem Berjalan

Berdasarkan analisis terhadap sistem yang telah diterapkan sebelumnya, D'Caruluk masih menerapkan media yang manual yaitu kertas untuk semua operasi pengolahan dari data. Terdapat 3 tahap proses dari bisnis dalam sistem ini yaitu:

a. Proses ketika Barang Masuk

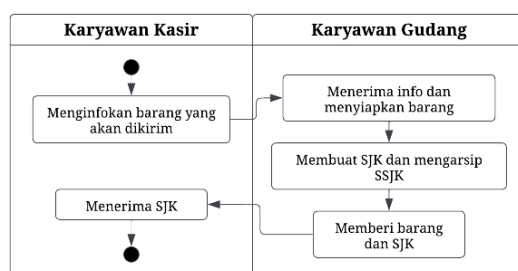
Barang beserta Surat Jalan Masuk (SJM) melalui Pemasok (*supplier*) kemudian dikirimkan kepada Karyawan Gudang yang nantinya akan ditandatangani, setelah itu Salinan Surat Jalan Masuk (SSJM) dikirim kembali kepada Pemasok, sementara Surat Jalan Masuk (SJM) disimpan.



Gambar 1. Diagram Activity Proses ketika Barang Masuk

b. Proses ketika Barang Keluar

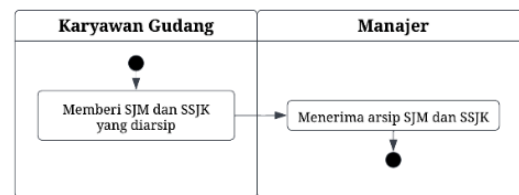
Karyawan Kasir mengabarkan kepada Karyawan Gudang barang yang akan dikirim, lalu Karyawan Gudang menyediakan produk tersebut dan meneruskannya kepada Karyawan Kasir bersama Surat Jalan Keluar (SJK). Salinan Surat Jalan Keluar (SSJK) disalin Karyawan Gudang.



Gambar 2. Diagram Activity Proses ketika Barang Keluar

c. Proses dari Pelaporan

Karyawan Gudang menyerahkan dokumen Surat Jalan Masuk (SJM) beserta Salinan Surat Jalan Keluar (SSJK) ke Manajer apabila dibutuhkan.



Gambar 3. Diagram Activity Proses dari Pelaporan

4.2. Rancangan Sistem Usulan

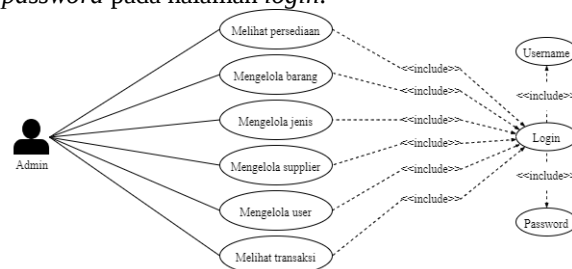
Sistem *inventory* diterapkan guna menampilkan beragam informasi tentang stok gudang. Berikut ini adalah rincian kebutuhan sistem persediaan D'Caruluk.

4.2.1. Diagram Use case

Diagram *Use case* diterapkan guna mendeskripsikan tingkah laku beserta menggambarkan hubungan antara satu atau lebih aktor dengan sistem yang dibangun. Dikawatir ini adalah *Use case* diagram dari sistem informasi persediaan D'Caruluk.

a. Admin

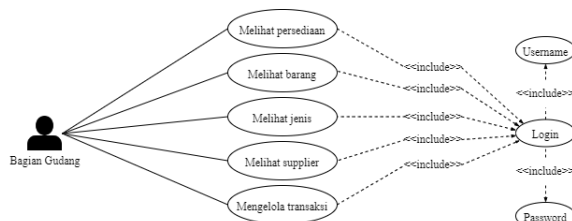
Admin sebagai *actor* dapat berinteraksi dengan *use case* seperti melihat persediaan, mengelola barang, mengelola jenis, mengelola *supplier*, mengelola user, dan melihat transaksi. Hal tersebut dapat terjadi apabila Admin sudah menginput *username* dan *password* pada halaman *login*.



Gambar 4. Diagram Use Case Admin

b. Bagian Gudang

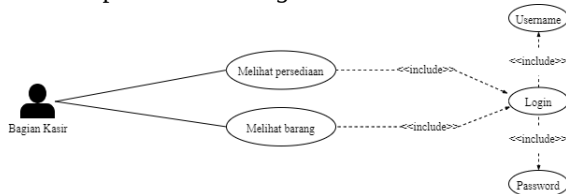
Bagian Gudang sebagai *actor* dapat berinteraksi dengan *use case* seperti melihat persediaan, melihat barang, melihat jenis, melihat *supplier*, dan mengelola transaksi. Hal tersebut dapat terjadi apabila Bagian Gudang sudah menginput *username* dan *password* pada halaman *login*.



Gambar 5. Diagram Use Case Bagian Gudang

c. Bagian Kasir

Bagian Kasir sebagai *actor* dapat berinteraksi dengan *use case* seperti melihat persediaan dan melihat barang. Hal tersebut dapat terjadi apabila Bagian Kasir sudah menginput *username* dan *password* pada halaman *login*.



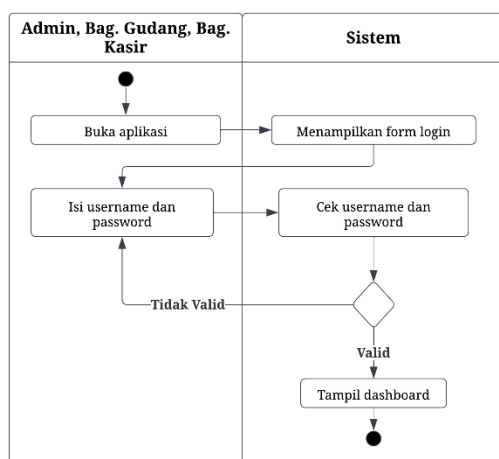
Gambar 6. Diagram Use case Bagian Kasir

4.2.2. Diagram Activity

Diagram aktivitas digunakan untuk menggambarkan aliran aktivitas dalam suatu sistem atau proses bisnis. Berikut ini merupakan kegiatan-kegiatan yang termasuk dalam sistem *inventory* D'Caruluk.

a. Login

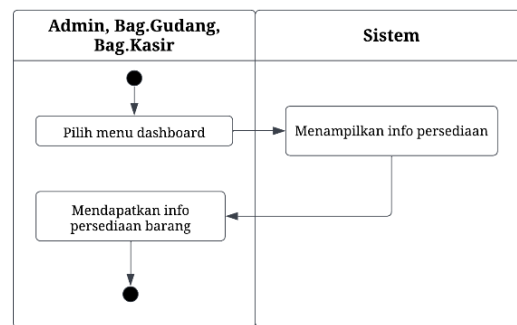
Admin, Bagian Gudang, dan Bagian Kasir dapat membuka aplikasi, selanjutnya sistem akan menampilkan form *login*. Kemudian Admin, Bagian Gudang, dan Bagian Kasir diminta untuk mengisi *username* dan *password*. Setelah itu, sistem mengecek *username* dan *password*, jika valid sistem akan menampilkan *dashboard*, dan jika tidak valid Admin, Bagian Gudang, dan Bagian Kasir diminta untuk mengisi *username* dan *password* kembali.



Gambar 7. Diagram Activity dari Login

b. Dashboard

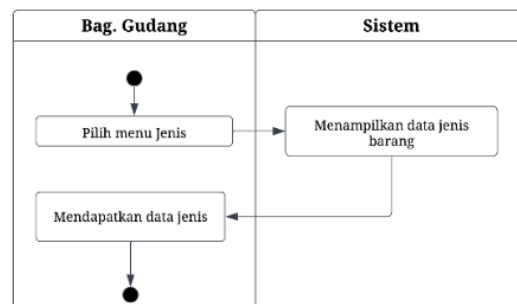
Admin, Bagian Gudang, dan Bagian Kasir bisa memilih menu *dashboard*. Setelah itu, sistem akan menampilkan info persediaan. Kemudian Admin, Bagian Gudang, dan Bagian Kasir mendapatkan info persediaan barang.



Gambar 8. Diagram Activity dari Dashboard

c. Halaman Data Jenis dari Bagian Gudang

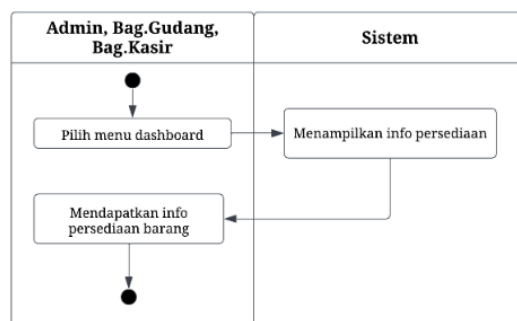
Bagian Gudang bisa memilih menu jenis, kemudian sistem pada akhirnya akan memproses permintaan dari Bagian Gudang. Setelah itu, sistem akan menampilkan data jenis barang. Bagian Gudang kemudian mendapatkan data jenis.



Gambar 9. Diagram Activity Data Jenis Bagian Gudang

d. Halaman Data Barang dari Bagian Kasir dan Bagian Gudang

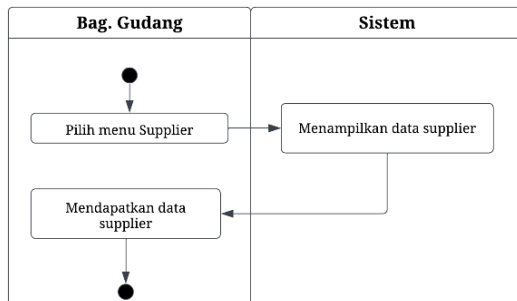
Bagian Gudang, dan Bagian Kasir dapat memilih menu *dashboard*. Setelah itu, sistem akan menampilkan info persediaan. Kemudian Bagian Gudang, dan Bagian Kasir mendapatkan info persediaan barang.



Gambar 10. Diagram Activity Data Barang Bagian Kasir dan Bagian Gudang

e. Diagram Activity Halaman Data Supplier

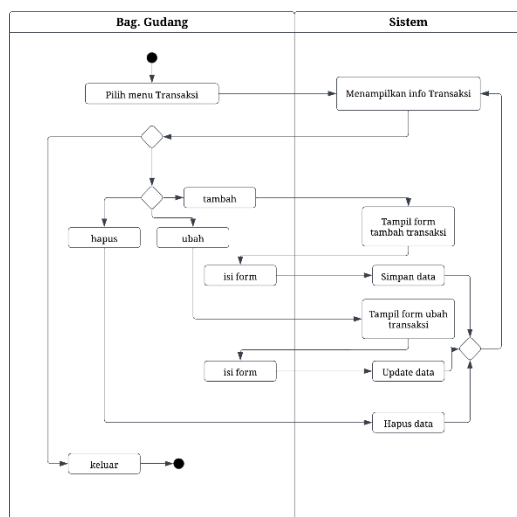
Bagian Gudang memilih menu *supplier*, kemudian sistem akan menampilkan data *supplier*. Setelah itu, Bagian Gudang mendapatkan data *supplier*.



Gambar 11. Diagram Activity Data Supplier Bagian Gudang

f. Halaman Data Transaksi dari Bagian Gudang

Bagian Gudang bisa memilih menu transaksi, kemudian sistem akan menampilkan info transaksi. Selanjutnya, Bagian Gudang akan diberikan pilihan untuk lanjut atau langsung keluar. Apabila lanjut akan diberi pilihan untuk tambah, ubah, dan hapus. Jika Bagian Gudang memilih untuk tambah, sistem akan menampilkan form untuk menambahkan transaksi, Bagian Gudang akan diberikan form pengisian dan sistem akan menyimpan data tersebut. Jika Bagian Gudang memilih untuk ubah, sistem akan menampilkan form untuk mengubah transaksi, Bagian Gudang akan diberikan form pengisian dan sistem akan mengupdate data tersebut. Terakhir jika Bagian Gudang memilih hapus, maka sistem akan menghapus data.



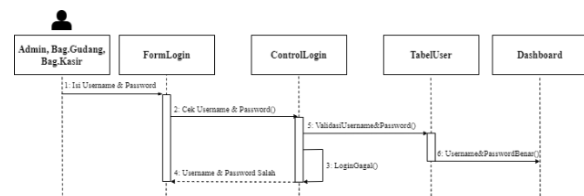
Gambar 12. Diagram Activity Data Transaksi Bagian Gudang

4.2.3. Diagram Sequence

Diagram Sequence menggambarkan interaksi antara objek atau komponen dalam suatu sistem dalam urutan waktu. Berikut ini merupakan *Diagram Sequence* yang termasuk dalam sistem inventory D'Caruluk.

a. Login

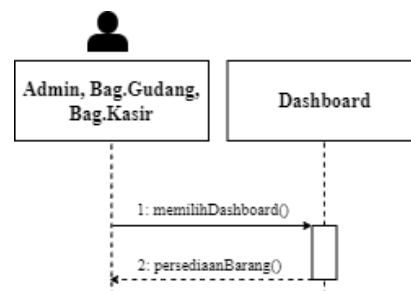
Admin, Bagian Gudang, dan Bagian Kasir dan empat objek, yaitu *form login*, *control login*, *table user*, dan *dashboard*. Pertama-tama Admin, Bagian Gudang, dan Bagian Kasir akan masuk ke tampilan *form login* dengan menggunakan *username* dan *password*. Lalu, *control login* akan mengirimkan data tersebut ke *table user* untuk divalidasi. Di dalam *table user* data Admin, Bagian Gudang, dan Bagian Kasir akan diperiksa dan divalidasi. Jika data yang dimasukan salah dan tidak valid, maka akan menampilkan pesan bahwa *username* atau *password* salah. Sedangkan jika data yang dimasukan benar dan valid, maka sistem akan menampilkan *dashboard* aplikasi.



Gambar 13. Diagram Sequence dari Login

b. Dashboard

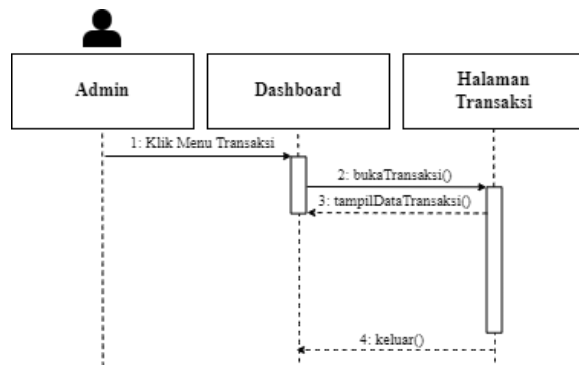
Admin, Bagian Gudang, dan Bagian Kasir memilih *dashboard*. Maka *dashboard* akan menampilkan persediaan barang.



Gambar 14. Diagram Sequence dari Dashboard

c. Halaman Data Transaksi dari Admin

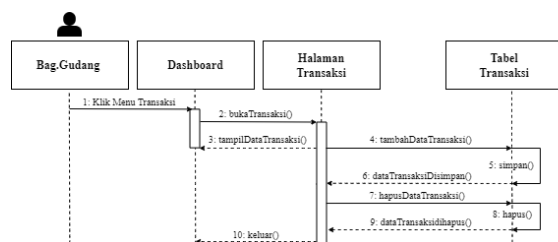
Admin memilih menu transaksi, kemudian *dashboard* akan membuka halaman transaksi dan menampilkan data transaksi pada *dashboard*.



Gambar 15. Diagram Sequence Data Transaksi Admin

d. Halaman Data Transaksi Bagian Gudang

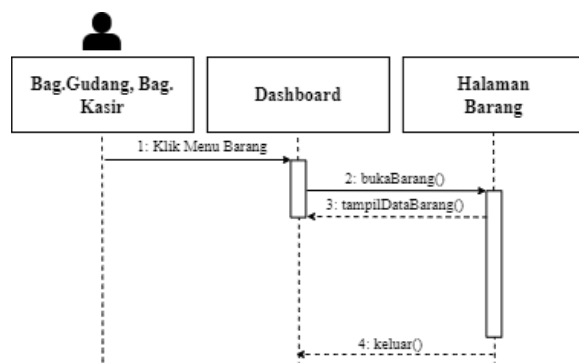
Bagian Gudang memilih menu transaksi, kemudian *dashboard* akan membuka halaman transaksi dan menampilkan data transaksi. Jika Bagian Gudang ingin menambahkan data transaksi, tabel transaksi akan menyimpan data tersebut dan menampilkan data yang telah tersimpan pada halaman transaksi. Jika Bagian Gudang ingin menghapus data transaksi, maka tabel transaksi akan menghapus data tersebut dan menampilkan data yang telah terhapus pada halaman transaksi.



Gambar 16. Diagram Sequence Data Transaksi Bagian Gudang

e. Halaman Data Barang Bagian Gudang dan Bagian Kasir

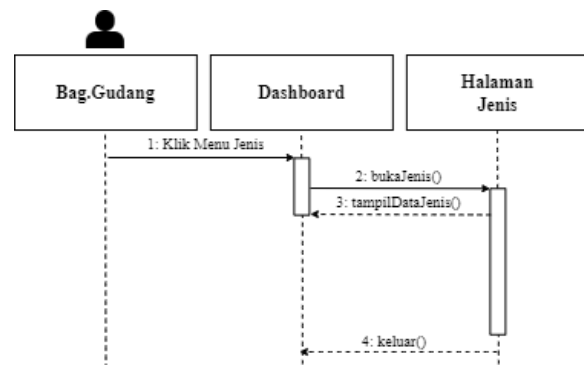
Bagian Gudang dan Bagian Kasir memilih menu barang pada *dashboard*, maka halaman barang akan menampilkan data barang pada *dashboard*.



Gambar 17. Diagram Sequence Data Barang Bagian Gudang dan Bagian Kasir

f. Halaman Data Jenis Bagian Gudang

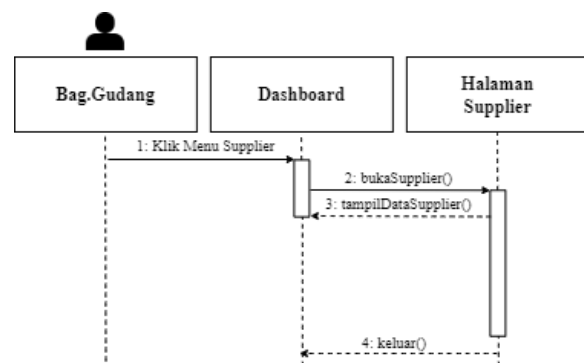
Bagian Gudang memilih menu jenis pada *dashboard*, maka halaman jenis akan menampilkan data jenis pada *dashboard*.



Gambar 18. Diagram Sequence Data Jenis Bagian Gudang

g. Halaman Data Supplier Halaman Bagian Gudang

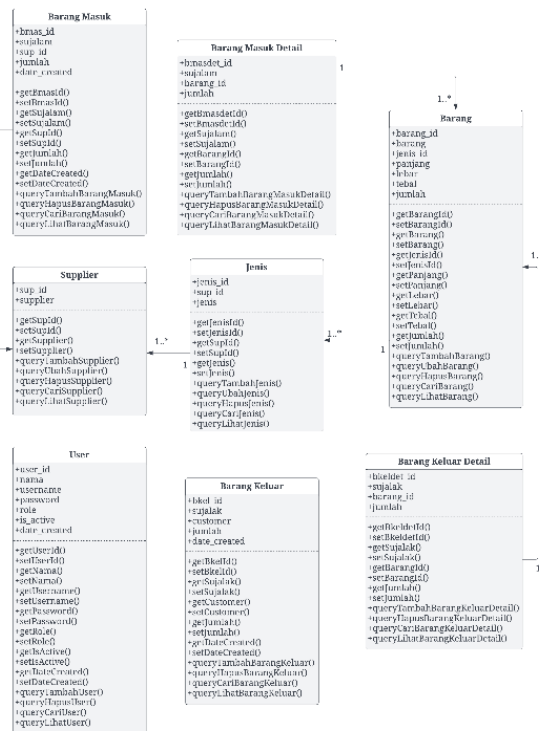
Bagian Gudang memilih menu *supplier* pada *dashboard*, maka halaman *supplier* akan menampilkan data *supplier* pada *dashboard*.



Gambar 19. Diagram Sequence Data Supplier Bagian Gudang

4.2.4. Diagram Class

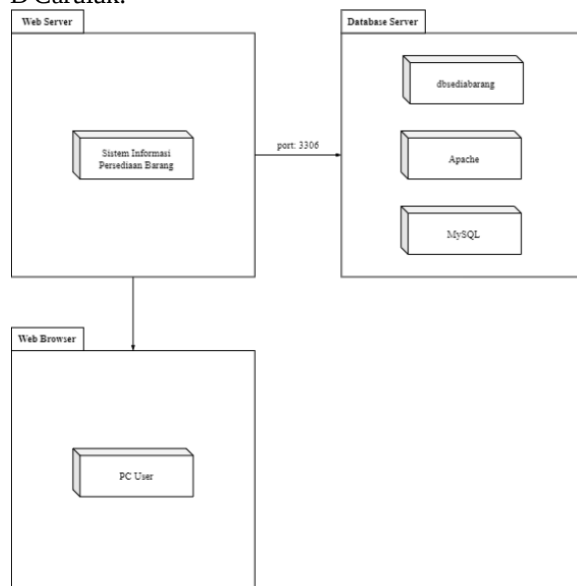
Diagram kelas memberikan gambaran mengenai struktur dari sebuah sistem atau aplikasi berbasis objek. Berikut ini merupakan Diagram *Class* dari sistem *inventory* D'Caruluk.



Gambar 20. Diagram *Class* Sistem
Inventory D'Caruluk

4.2.5. Diagram Deployment

Diagram *Deployment* menggambarkan bagaimana komponen-komponen dari suatu sistem atau aplikasi diimplementasikan dalam lingkungan fisik. Berikut ini merupakan Diagram *Deployment* dari sistem *inventory* D'Caruluk.

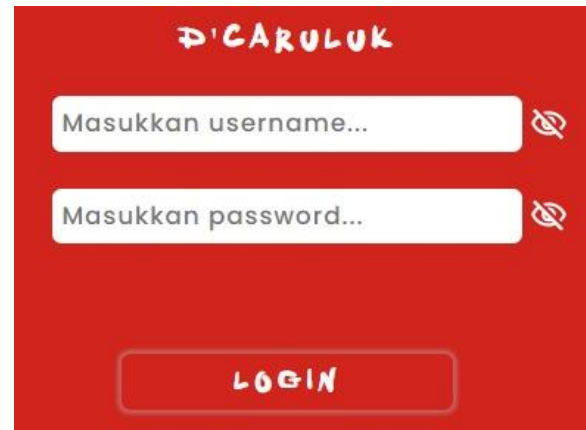


Gambar 21. Diagram *Deployment* Sistem
Inventory D'Caruluk

4.2.6. Rancangan User Interface

a. Halaman *Login*

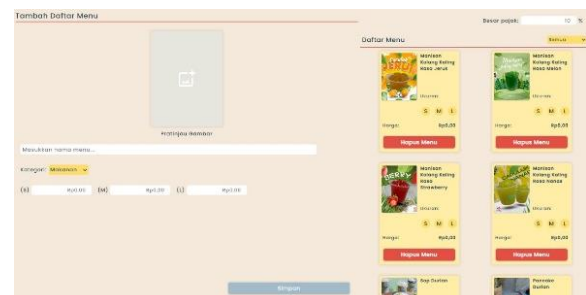
Bagian kasir dan bagian gudang dapat mengakses halaman *dashboard* dengan menginput *username* dan *password* pada halaman *login*.



Gambar 22. Tampilan Halaman dari *Login*

b. Halaman *Dashboard*

Bagian kasir, admin, dan bagian gudang memiliki akses halaman *dashboard* ini. Pada halaman *dashboard* dapat melihat informasi mengenai menu barang, menu *logout*, dan persediaan barang.



Gambar 23. Tampilan Halaman dari *Dashboard*

c. Halaman Data User

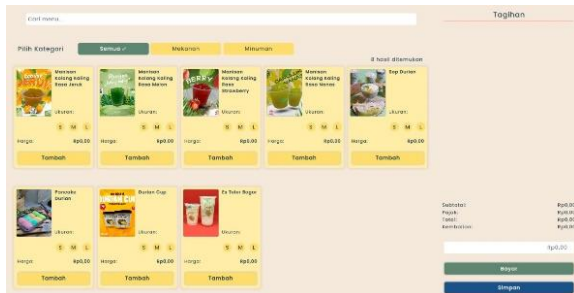
Halaman dari data pengguna khusus dimasuki oleh admin. Admin juga bisa mengubah, melihat, menambah, dan mencari data dari pengguna.



Gambar 24. Tampilan Halaman dari *User*

d. Halaman Data Barang

Bagian Gudang, bagian kasir, dan admin dapat mengakses halaman data barang. Bagian kasir dan bagian Gudang bisa melihat dan mencari barang. Sedangkan admin bisa menghapus, mengubah, melihat, mencari, serta menambah data barang.



Gambar 25. Tampilan Halaman Data Barang

e. Halaman Data Jenis

Bagian admin dan gudang dapat mengakses halaman data jenis. Bagian gudang bisa melihat dan mencari data jenis. Sedangkan admin dapat menghapus, menambah, mengubah, melihat, dan mencari data jenis.



Gambar 26. Tampilan Halaman dari Data Jenis

f. Halaman Data Supplier

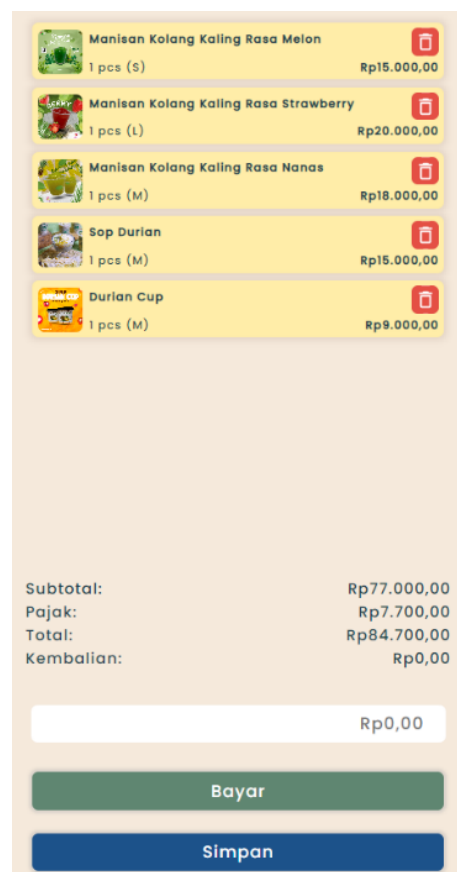
Bagian gudang dan admin dapat mengakses halaman data Supplier. Bagian gudang dapat melihat dan mencari data Supplier. Sedangkan admin dapat menghapus, menambah, mengubah, melihat, dan mencari data Supplier.



Gambar 27. Tampilan Halaman Data Supplier

g. Halaman Transaksi

Bagian gudang dan admin dapat mengakses halaman transaksi. Bagian gudang dan admin dapat mencari, menambah, dan melihat data transaksi.



Gambar 28. Tampilan Halaman dari Transaksi

5. KESIMPULAN DAN SARAN

Berdasarkan analisis yang dilaksanakan dari seluruh tahapan hasil dari pengujian sistem persediaan barang (*inventory*) D'Caruluk yang sudah dijelaskan diatas, maka penulis bisa menyimpulkan bahwa sistem yang dibangun mampu mengatasi permasalahan tersebut dan dapat diterapkan bagi sistem *inventory* D'Caruluk. Kemudian dengan dibangunnya program ini, arus penerimaan barang dan pengeluaran barang akan lebih menghemat waktu dan biaya, sehingga aktivitas yang berjalan lebih efektif dan efisien. Terakhir, sistem *inventory* juga mampu memberikan laporan stok barang secara akurat yang dulunya menjadi masalah bagi D'Caruluk.

DAFTAR PUSTAKA

- [1] Tamodia 2013, "Rancang Bangun Sistem Informasi Persediaan Barang Berbasis Web Dengan Metode Fast(Framework For The Applications)," *J. PILAR Nusa Mandiri*, vol. Vol. 13, N, no. 2, pp. 261–266, 2017, [Online]. Available: <http://ejournal.nusamandiri.ac.id/ejurnal/index.php/pilar/article/view/705>
- [2] N. H. Cahyana, B. Yuwono, and A. Y. Asmoro, "Pengembangan Sistem Informasi Persediaan Barang Berbasis Web Di Pt . Putera Agung Setia," *Pengemb. Sist. Inf. Persediaan Barang Berbas. Web Di Pt . Puter. Agung Setia*, vol. 9, no. semnasIF, pp. 252–258, 2012.
- [3] S. Suhartini, M. Sadali, and Y. Kuspani Putra, "Sistem Informasi Berbasis Web Sma Al-Mukhtariyah Mamben Lauk Berbasis Php Dan Mysql Dengan Framework Codeigniter," *Infotek J. Inform. dan Teknol.*, vol. 3, no. 1, pp. 79–83, 2020, doi: 10.29408/jit.v3i1.1793.
- [4] Y. D. Wijaya and M. W. Astuti, "Sistem Informasi Penjualan Tiket Wisata Berbasis Web Menggunakan Metode Waterfall," *Semin. Nas. Teknol. Inf. dan Komun.*, pp. 273–276, 2019.
- [5] F. Fatmawati and J. Munajat, "Implementasi Model Waterfall Pada Sistem Informasi Persediaan Barang Berbasis Web (Studi Kasus: PT.Pamindo Tiga T)," *J. Media Inform. Budidarma*, vol. 2, no. 2, pp. 1–9, 2018, doi: 10.30865/mib.v2i2.559.
- [6] A. Rifai and Y. P. Yuniar, "Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Ujian Pada SMK Indonesia Global Berbasis Web," *J. Khatulistiwa Inform.*, vol. 7, no. 1, pp. 1–6, 2019, doi: 10.31294/jki.v7i1.64.
- [7] Manurung, "Sistem Informasi Lembaga Kursus Dan Pelatihan (LKP) City Com Berbasis Web Menggunakan Php Dan Mysql," *J. Mahajana Inf.*, vol. 4, no. 1, pp. 42–50, 2019, [Online]. Available: <http://114.7.97.221/index.php/7/article/view/726>
- [8] N. Iriadi, P. Priatno, and S. Sainan, "Sistem Informasi Penjualan Makanan Olahan Berbasis Web Pada Toko Anyar Bekasi," *J. Teknol. Inform. dan Komput.*, vol. 4, no. 1, pp. 12–17, 2018, doi: 10.37012/jtik.v4i1.281.
- [9] F. N. Salamah, U., & Khasanah, "Pengujian Sistem Informasi Penjualan Undangan Pernikahan Online Berbasis Web Menggunakan Black Box Testing," *Inf. Manag. Educ. Prof.*, vol. 2, no. 1, pp. 35–46, 2017.
- [10] Z. Hakim, L. Sakuroh, and S. Awaludin, "Sistem Informasi Persediaan Barang Berbasis Web Pada CV Telaga Berkat," *J. Sisfotek Glob.*, vol. 9, no. 1, 2019, doi: 10.38101/sisfotek.v9i1.214.
- [11] R. Priskila, "Pada Perusahaan Karya Cipta Buana Sentosa," *J. Comput. Eng. Syst. Sci.*, vol. 3, no. 2, pp. 94–99, 2018.