

Perancangan Sistem Informasi Inventory (SIVEN)

Eka Puspita Sari¹, Eni Pudjiarti²

¹Universitas Bina Sarana Informatika
Jl. Kramat Raya No. 98, Indonesia
e-mail: eka.eps@bsi.ac.id

²Universitas Nusa Mandiri
Jl. Raya Jatiwaringin No. 2, Indonesia
e-mail: eni.epr@nusamandiri.ac.id

Abstrak - Teknologi berkembang dengan begitu pesat serta kebutuhan informasi pada saat ini merupakan kebutuhan mutlak bagi suatu organisasi atau perusahaan agar mempermudah dalam menyelesaikan sebuah pekerjaan pada bidang tertentu. Dimana informasi juga merupakan pendukung dalam pengambilan keputusan, sehingga dibutuhkan penyampaian informasi yang akurat, tepat dan cepat dalam mendapatkannya agar tidak menimbulkan masalah-masalah yang akan menghambat sebuah pekerjaan. Dalam penelitian ini pengumpulan data dilakukan dengan beberapa metode, yaitu dengan menggunakan metode wawancara (*Interview*), pengamatan (*Observation*), studi pustaka. Sistem informasi *inventory* yang dilakukan secara manual, mulai dari proses penerimaan material, pengecekan material, *update* stok material, sampai pembuatan laporan sehingga memungkinkan pada saat proses berlangsung terjadi kesalahan dalam pencatatan, kurang akuratnya laporan yang dibuat dan keterlambatan dalam pencarian data-data yang diperlukan. Untuk mengetahui permasalahan yang terdapat pada sistem informasi *inventory* yang masih secara manual maka metode yang digunakan yaitu metode *waterfall* (model air terjun), Model *Waterfall* adalah strategi SDLC paling awal yang dimanfaatkan untuk peningkatan perangkat lunak. Dengan diterapkannya perancangan sistem *inventory* diharapkan dapat membantu dan mempermudah dalam proses pengelolaan *inventory* baik dalam pengelolaan material masuk, material keluar dan juga stok material, sehingga dalam proses kerja dapat menghasilkan sebuah laporan yang lebih akurat dan juga lebih efisien dan efektif dalam pengelolaan *inventory*.

Kata Kunci : Sistem, Informasi, *Inventory*

Abstracts – *Technology is developing so rapidly and the need for information at this time is an absolute necessity for an organization or company to make it easier to complete a job in a particular field. Where information is also a support in decision making, so it takes the delivery of information that is accurate, precise and fast in getting it so as not to cause problems that will hinder a job. In this study, data collection was carried out by several methods, namely by using the method of interview (Interview), observation (Observation), literature study. Inventory information system that is carried out manually, starting from the process of receiving materials, checking materials, updating material stock, to making reports so that it is possible during the process there are errors in recording, less accurate reports are made and delays in finding the required data. To find out the problems that exist in the inventory information system that is still manually, the method used is the waterfall method (waterfall model), the Waterfall Model is the earliest SDLC strategy used for software improvement. With the implementation of the inventory system design, it is expected to help and simplify the inventory management process both in the management of incoming materials, outgoing materials and also material stock, so that in the work process can produce a report that is more accurate and also more efficient and effective in inventory management.*

Keywords : System, Information, *Inventory*

PENDAHULUAN

Kemajuan teknologi informasi dipengaruhi adanya keinginan akan teknologi dan sistem informasi yang eksplisit, efektif dan efisien. Meningkatnya teknologi komputer mempengaruhi kemampuan manusia sebagai



operasional sistem sehingga perubahan ke arah sistem informasi yang berbasis komputer semakin bertambah. Dimana informasi juga merupakan sebuah pendukung dalam pembuat keputusan, sehingga dibutuhkan penyajian informasi yang akurat, tepat dan cepat dalam mendapatkannya agar tidak menimbulkan sebuah masalah-masalah yang akan menghalangi sebuah pekerjaan. Informasi tersebut berisi kondisi yang terjadi di dalam maupun di luar perusahaan yang kemudian akan digunakan sebagai landasan untuk perencanaan strategi dan program kerja.

Persediaan dalam perusahaan pengertian atau prosesnya terkait dari jenis perusahaan tersebut. Jika perusahaan tergolong dalam kelompok perusahaan manufaktur berarti persediaan yang akan diatur mencakup persediaan hasil jadi, persediaan hasil dalam proses, persediaan bahan baku, persediaan bahan pendukung dan lainnya (Astuti, 2017). Persediaan adalah aset yang akan diperdagangkan dan dijual untuk menghasilkan tingkat pengembalian yang diperlukan (Susanto, 2017).

Adapun tujuan pengelolaan persediaan adalah sebagai berikut :

1. Untuk melengkapi kebutuhan atau permintaan konsumen dengan cepat (memuaskan konsumen)
2. Untuk menjaga kesinambungan atau menjaga agar perusahaan tidak menghadapi kekurangan persediaan yang akan menimbulkan terhentinya proses, hal ini dikarenakan alasan :
 - a. Kemungkinan bahan menjadi jarang sehingga sukar untuk didapatkan.
 - b. Kemungkinan pemasok terlambat mengirimkan bahan yang dipesan.
3. Untuk mempertahankan dan jika dapat meningkatkan penjualan dan keuntungan perusahaan.
4. Menjaga agar pemesanan dalam skala kecil dapat dikurangi, karena dapat menyebabkan biaya pesan menjadi bertambah.
5. Menjaga agar penyimpanan dalam lokasi tidak besar-besaran, karena akan mengakibatkan biaya lebih naik.

Menurut Maniah, S.Kom., MT dalam (Rakhmah et al., 2018) mendefinisikan Sistem bahwa Sistem didefinisikan sebagai kumpulan dari elemen-elemen berupa data, jaringan kerja dari prosedur-prosedur yang saling berhubungan, sumber daya manusia, teknologi baik hardware maupun software yang saling berinteraksi sebagai satu kesatuan untuk mencapai tujuan/sasaran tertentu yang sama.

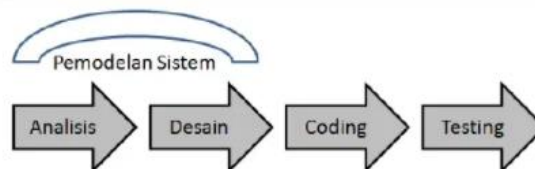
Sistem persediaan merupakan salah satu sistem yang mendukung operasional perusahaan. Kegiatan utama dari suatu perusahaan yang melakukan penjualan barang harus mampu mencukupi stok barang. Keterjaminan stok barang yang dimiliki oleh perusahaan dalam sistem persediaan akan meningkatkan kepuasan pelanggan yang berbelanja karena stok selalu tersedia. Kepuasan pelanggan merupakan hal yang paling penting untuk diperhatikan oleh perusahaan. Untuk itu penting bagi suatu perusahaan meningkatkan operasional kegiatan persediaan barangnya. Sehingga dengan adanya sistem persediaan barang (SIVEN) akan menunjang kinerja sistem penjualan yang pada akhirnya meningkatkan pendapatan perusahaan.

METODE PENELITIAN

Model *Waterfall* yang terkadang juga dikenal sebagai model air terjun adalah model sistem pertama yang diperkenalkan. Model *Waterfall* sangat mudah untuk dimengerti dan digunakan.(Ali Edward, 2019). Dalam model *Waterfall*, setiap tahapan harus dilalui sebelum tahapan selanjutnya bisa dimulai dan tidak ada tahapan yang saling menumpuk. Model *Waterfall* adalah strategi SDLC paling awal yang dimanfaatkan untuk peningkatan perangkat lunak.(Ali Edward, 2019).

Pada pendekatan *Waterfall* , semua proses peningkatan perangkat lunak dirinci menjadi tahapan terpisah. Hasil dari satu tahapan berperan sebagai masukan untuk tahapan selanjutnya secara konsekutif. Ini bermakna bahwa setiap tahapan dalam proses peningkatan dimulai hanya jika tahapan sebelumnya tuntas (Ali Edward, 2019). Model *Waterfall* adalah teknik rancangan berurutan di mana perkembangan kegiatan dilihat sebagai bentuk aliran dari atas terus ke bawah (seperti air terjun melalui beberapa tahapan).

Gambar berikut menggambarkan sekuensial linier untuk rekayasa perangkat lunak, yang sering disebut dengan siklus kehidupan klasik atau model air terjun.



Sumber : (Ali Edward, 2019)

Gambar 1. Model Sekuensial Linier (*Waterfall Model*)

Sekuensial linier menerapkan sebuah pendekatan kepada peningkatan perangkat lunak yang sistematis dan sekuensial yang mulai pada tingkat dan evolusi sistem pada seluruh analisis, desain, kode, pengujian (tes), dan pemeliharaan.(Ali Edward, 2019)

Berikut ini uraian yang dapat diberikan untuk setiap tahap menurut (Ali Edward, 2019):

- a) Analisis kebutuhan perangkat lunak

Analisis kebutuhan merupakan prosedur awal untuk mendefinisikan uraian perangkat lunak yang akan didapatkan ketika pengembang melakukan sebuah proyek penyusunan perangkat lunak. Perangkat lunak yang baik dan sesuai dengan keperluan pengguna sangat terkait pada keberhasilan dalam melakukan analisis kebutuhan. Untuk proyek-proyek perangkat lunak yang besar, analisis kebutuhan dilakukan setelah aktivitas sistem *information engineering* dan *software project planning*. Analisa kebutuhan yang efisien dan efektif belum tentu membentuk perangkat lunak yang baik, tetapi analisa kebutuhan yang tidak tepat dapat menghasilkan perangkat yang tidak berguna. Keberhasilan mengenal adanya kesalahan pada analisis kebutuhan di langkah awal memang jauh lebih baik, tapi kesalahan analisis kebutuhan yang diketahui ketika sudah memasuki penulisan kode atau pengujian, terlebih hampir masuk dalam langkah penyelesaian menjadikan kesalahan besar bagi pengembang perangkat lunak. Biaya dan waktu yang diperlukan akan menjadi percuma. Agar proses dokumentasi data kebutuhan berjalan dengan baik, perancang perangkat lunak (analisis) harus menguasai persis domain permasalahan (*problem domain*), tingkah laku, unjuk kerja dan antarmuka (*interface*) yang dibutuhkan. Kebutuhan yang terkait dengan sistem atau perangkat lunak didokumentasikan dan dipantau lagi oleh klien.

- b) Desain
Desain perangkat lunak sebenarnya adalah proses banyak tahapan yang berpusat pada empat atribut sebuah rancangan yang berbeda (struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (algoritma) prosedural. Proses desain menerjemahkan syarat atau keperluan ke dalam sebuah gambaran perangkat lunak yang dapat diprediksi demi kualitas sebelum dimulai pembuatan kode (*coding*). Sebagaimana analisis, desain ini juga didokumentasikan.
- c) Menghasilkan kode
Desain yang telah dihasilkan harus diterjemahkan ke dalam bentuk bahasa mesin yang dapat dibaca oleh perangkat keras. Tahapan pembentukan kode melingkupi pekerjaan dalam tahapan ini, dan bisa dilakukan secara otomatis.
- d) Pengujian (*Testing*)
Sekali kode dibuat, pengujian program sudah dapat dilakukan. Proses pengujian berpusat pada logika internal perangkat lunak menentukan bahwa semua *statement* sudah diuji, dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk mendeteksi kesalahan-kesalahan dan menentukan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang diinginkan.
- e) Pemeliharaan (*Maintenance*)
Perangkat lunak akan menghadapi perubahan dan penyesuaian setelah diberikan kepada pengguna. Perubahan akan terjadi karena kesalahan-kesalahan pada perangkat lunak harus disesuaikan untuk beradaptasi terhadap perubahan-perubahan di dalam lingkungan eksternalnya atau karena pengguna membutuhkan pengembangan aspek fungsional atau cara kerja. Pemeliharaan perangkat lunak mengaplikasikan lagi setiap tahapan program sebelumnya dan tidak membuatnya dari awal lagi.

Web atau website adalah kumpulan dari halaman situs dan dokumen yang tersebar di beberapa komputer server yang berada di seluruh penjuru dunia dan terhubung menjadi satu jaringan melalui jaringan yang disebut internet (Batubara, 2018). Web Browser adalah suatu perangkat lunak yang digunakan untuk menjelajahi internet atau mencari informasi dari suatu web (Suri, 2019). Web Server adalah perangkat lunak yang memberikan layanan yang berbasis data dan berfungsi mentransfer berkas permintaan dari HTTP atau HTTPS untuk mengirimkan kembali yang hasilnya dalam bentuk halaman web dan berbentuk dokumen HTML (Suri, 2019). Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis (Fadallah & Rosyida, 2018).

HASIL DAN PEMBAHASAN

Berdasarkan hasil dari penelitian didapatkan bahwa dalam pembuatan sistem informasi *inventory* dibutuhkan beberapa tahapan antara lain:

Analisa kebutuhan diperoleh berdasarkan kebutuhan *user* (pengguna) dan kebutuhan dari sistem *inventory* :

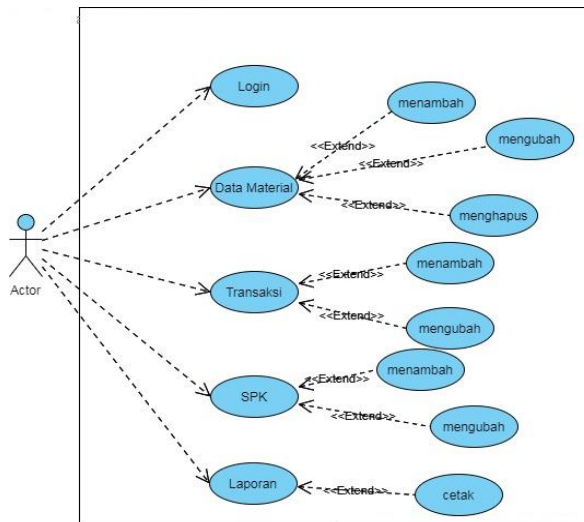
1. Kebutuhan *User* (Pengguna)

Dalam sistem *inventory* terdapat satu *user* (pengguna) yang dapat berinteraksi dalam lingkup sistem secara menyeluruh yaitu admin. *User* (pengguna) tersebut memiliki karakteristik dalam berinteraksi dengan sistem dan memiliki kebutuhan informasi sebagai berikut :

A. Skenario *User* (pengguna)

- a) Mengakses data *inventory*, data spk, data transaksi dan laporan *inventory*.
 - b) Mengelola data *inventory*, data spk, data transaksi dan data laporan *inventory* (mengubah, menambah, menghapus, mencetak).
- B. Kebutuhan Sistem
- a) *User* (pengguna) harus login terlebih dahulu adar dapat mengakses dan mengelola data *inventory*, data transaksi dan laporan yang terdapat dalam sistem dengan memasukkan username dan password agar privasi dan keamanan *User* (pengguna) dan data tersebut tetap terjaga keamanannya.
 - b) *User* (pengguna) wajib melakukan *log out* setelah mengakses ataupun mengelola data *inventory*, data transaksi dan laporan yang terdapat pada sistem.
2. Rancangan Diagram *Usecase*

Menurut Sukanto dan M.Shalahuddin dalam (Mauludin, 2019) “Diagram use case atau *Use Case Diagram* merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat”.*Use Case* diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dibawah ini *use case* diagram dari rancangan sistem yang menjelaskan mengenai proses yang dilakukan user.



Sumber : Hasil Penelitian (2021)

Gambar 2. *Usecase Diagram*

Tabel 1. Deskripsi *Use Case Diagram Login*

<i>Use Case Name</i>	<i>Login</i>
<i>Requirements</i>	Foreman dapat mengakses web inventory
<i>Goal</i>	Foreman dapat menambah, megubah, dan menghapus data material, spk dan laporan
<i>Pre-Conditions</i>	Foreman memasukkan username dan password
<i>Post-Conditions</i>	Data material, spk, dan laporan tersimpan, terupdate, terhapus atau tercetak
<i>Failed end Condition</i>	Gagal masuk ke web inventory
<i>Actors</i>	Foreman
<i>Main Flow/ Basic Path</i>	<ol style="list-style-type: none"> 1. Foreman membuka web inventory 2. Forman memasukkan usersname dan password 3. Foreman dapat mengakses halaman utama web sistem inventory
<i>Alternate Flow/Invariant A</i>	<ol style="list-style-type: none"> A1. Foreman melakukan login A2. Foreman salah memasukkan username dan password A3. Foreman tidak dapat masuk web inventory
<i>Invariant B</i>	-

Sumber : Hasil Penelitian (2021)

Tabel 2. Deskripsi *Use Case Diagram Data material*

<i>Use Case Name</i>	Mengakses dan mengelola data material
<i>Requirements</i>	Foreman dapat mengelola data material

<i>Goal</i>	Foreman dapat menambah, menghapus, mengubah, dan mencari data material.
<i>Pre-Conditions</i>	Foreman sudah melakukan <i>login</i>
<i>Post-Conditions</i>	Data material tersimpan, terupdate atau terhapus
<i>Failed end Condition</i>	Gagal menyimpan, mengupdate, atau menghapus
<i>Actors</i>	Foreman
<i>Main Flow/ Basic Path</i>	<ol style="list-style-type: none"> 1. Foreman memilih tambah data material 2. Sistem akan menampilkan form tambah data material 3. Foreman menginput kode material 4. Foreman menginput nama supplier 5. Foreman menginput nama material 6. Foreman menginput qty material 7. Foreman menginput tanggal material masuk 8. Foreman menginput satuan material 9. Foreman dapat menyimpan data material 10. Sistem menampilkan data material
<i>Alternate Flow/Invariant A</i>	<ol style="list-style-type: none"> A1. Foreman mengubah data material A2. Sistem menampilkan data material yang akan diubah A3. Foreman menginput data perubahan data material A4. Foreman menyimpan data material A5. Sistem menampilkan data material yang sudah diubah
<i>Invariant B</i>	<ol style="list-style-type: none"> B1. Foreman menghapus data material B2. Sistem menampilkan konfirmasi B3. Foreman menghapus data

Sumber : Hasil Penelitian (2021)

Tabel 3. Deskripsi *Use Case* Diagram Transaksi

<i>Use Case Name</i>	Mengakses dan mengelola Transaksi
<i>Requirements</i>	Foreman dapat mengelola data transaksi
<i>Goal</i>	Foreman dapat menambah, dan mengubah data transaksi
<i>Pre-Conditions</i>	Foreman sudah melakukan <i>login</i>
<i>Post-Conditions</i>	Transaksi tersimpan, atau terupdate
<i>Failed end Condition</i>	Gagal menyimpan, mengupdate, atau menghapus
<i>Actors</i>	Foreman
<i>Main Flow/ Basic Path</i>	<ol style="list-style-type: none"> 1. Foreman memilih tambah transaksi 2. Sistem akan menampilkan form tambah data transaksi 3. Foreman menginput id transaksi 4. Foreman menginput id foreman 5. Foreman menginput kode spk 6. Foreman menginput tanggal transaksi 7. Foreman menginput qty 8. Foreman menginput satuan 9. Foreman menginput jenis transaksi 10. Foreman dapat menyimpan data transaksi 11. Sistem menampilkan transaksi
<i>Alternate Flow/Invariant A</i>	<ol style="list-style-type: none"> A1. Foreman mengubah transaksi A2. Sistem menampilkan data transaksi yang akan diubah A3. Foreman menginput data perubahan transaksi A4. Foreman dapat menyimpan data transaksi A5. Sistem menampilkan data transaksi yang sudah diubah
<i>Invariant B</i>	-

Sumber : Hasil Penelitian (2021)

Tabel 4. Deskripsi *Use Case* Diagram Surat Perintah Kerja

<i>Use Case Name</i>	Mengakses dan mengelola SPK
<i>Requirements</i>	Foreman dapat mengelola data SPK
<i>Goal</i>	Foreman dapat menambah, mengubah, data SPK
<i>Pre-Conditions</i>	Foreman sudah melakukan <i>login</i>
<i>Post-Conditions</i>	Data SPK tersimpan, atau terupdate

<i>Failed end Condition</i>	Gagal menyimpan, atau mengupdate
<i>Actors</i>	Foreman
<i>Main Flow/ Basic Path</i>	<ol style="list-style-type: none"> 1. Foreman memilih tambah spk 2. Sistem akan menampilkan form tambah spk 3. Foreman menginput kode spk 4. Foreman menginput tanggal 5. Foreman menginput no po 6. Foreman menginput customer 7. Foreman menginput nama material 8. Foreman menginput qty 9. Foreman menginput satuan 10. Foreman dapat menyimpan spk 11. Sistem menampilkan spk
<i>Alternate Flow/Invariant A</i>	<ol style="list-style-type: none"> A1. Foreman mengubah spk A2. Sistem menampilkan data spk yang akan diubah A3. Foreman menginput data perubahan spk A4. Foreman dapat menyimpan data spk A5. Sistem menampilkan data spk yang sudah diubah
<i>Invariant B</i>	-

Sumber : Hasil Penelitian (2021)

Tabel 5. Deskripsi Use Case Diagram Laporan

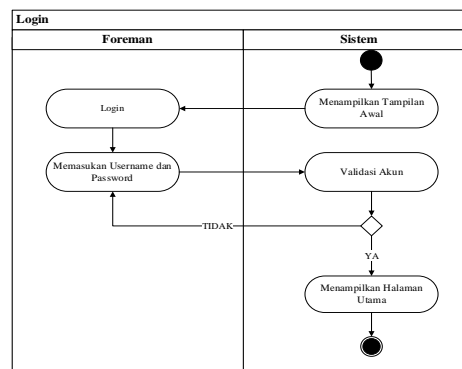
Use Case Name	Mengakses Laporan dan mencetak laporan
Requirements	Foreman mengakses data laporan
Goal	Foreman dapat mengakses dan mencetak laporan
Pre-Conditions	Foreman sudah <i>Login</i>
Post-Conditions	Laporan tercetak
Failed end Condition	Gagal mengakses dan mencetak laporan
Actors	Foreman
Main Flow/ Basic Path	<ol style="list-style-type: none"> 1. Foreman memilih cetak laporan 2. Sistem menampilkan data laporan 3. Foreman memilih periode laporan yang akan dicetak 4. Sistem menampilkan data laporan yang akan dicetak
Alternate Flow/Invariant A	-
Invariant B	-

Sumber : Hasil Penelitian (2021)

3. Rancangan Diagram *Activity*

Diagram *Activity* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem yaitu proses bisnis atau menu yang ada pada perangkat lunak. Diagram *activity* dibawah ini menggambarkan proses foreman melakukan *login* (masuk) ke dalam perancangan sistem informasi *inventory*.

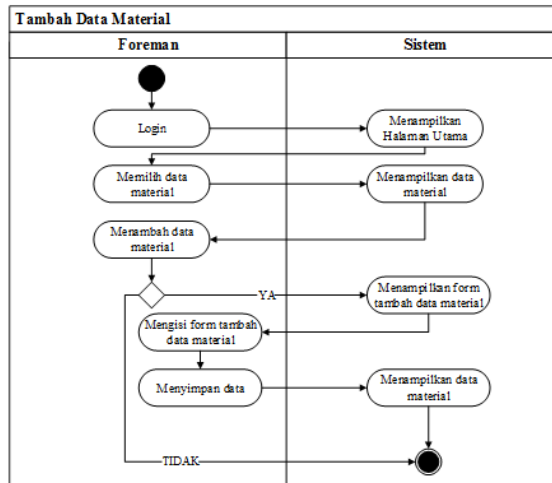
a. Activity Diagram Login



Sumber : Hasil Penelitian (2021)

Gambar 3. Activity Diagram Login

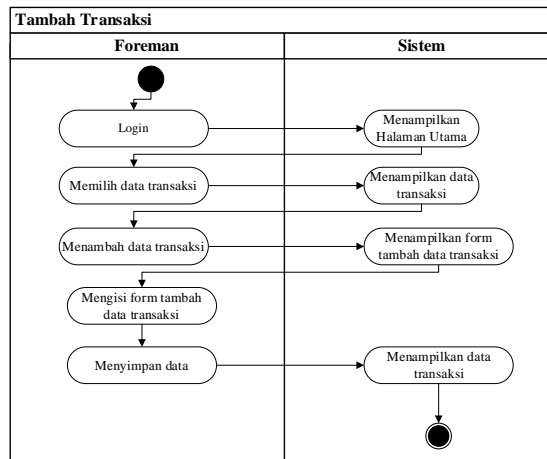
b. Activity Diagram Tambah Data Material



Sumber : Hasil Penelitian (2021)

Gambar 4. Activity Diagram Tambah Data Material

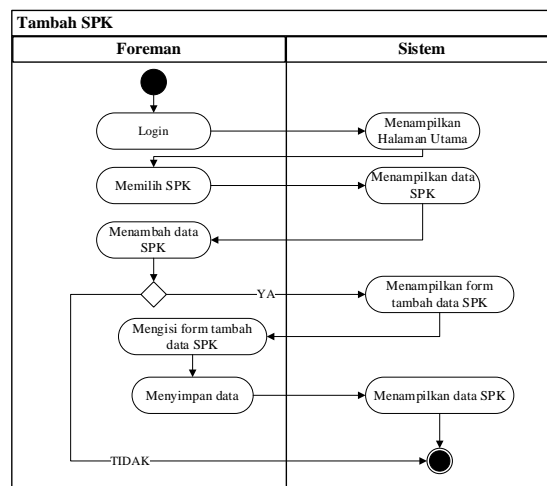
c. Activity Diagram Tambah Transaksi



Sumber : Hasil Penelitian (2021)

Gambar 5. Activity Diagram Tambah Transaksi

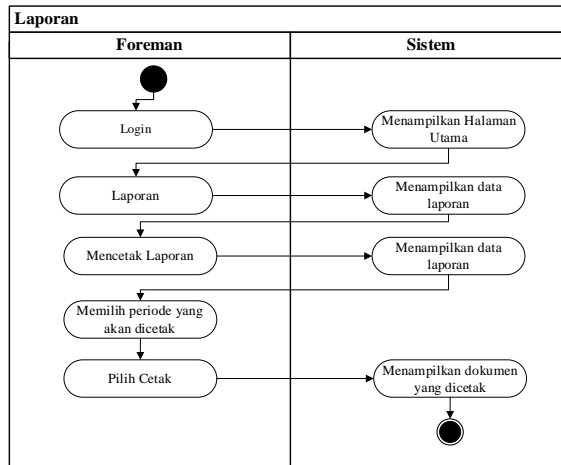
d. Activity Diagram Tambah SPK



Sumber : Hasil Penelitian (2021)

Gambar 6. Activity Diagram Tambah SPK

e. Activity Diagram Laporan

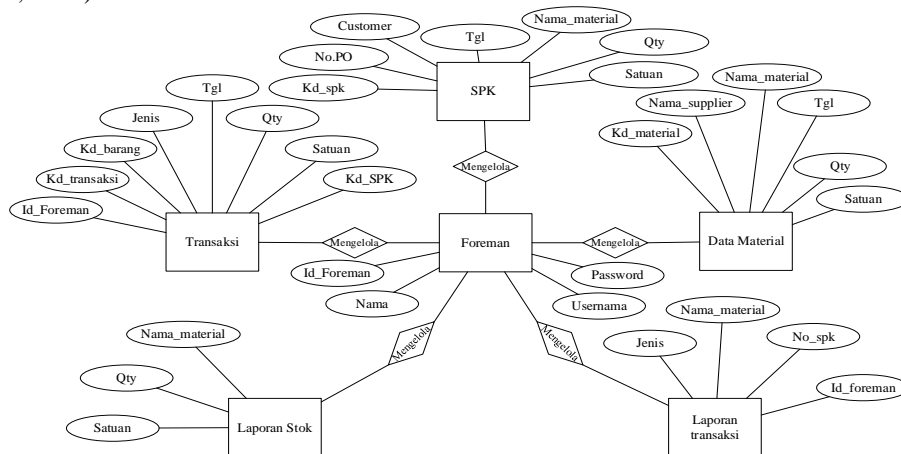


Sumber : Hasil Penelitian (2021)

Gambar 7. Activity Diagram Laporan

4. Entity Relationship Diagram (ERD)

ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional. Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan Entity Relationship Diagram (ERD)”. (sukamto & Shalahuddin, 2018)






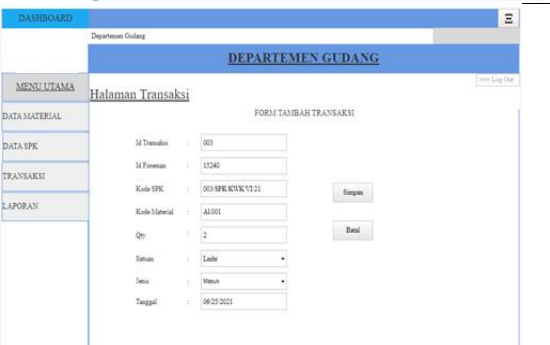

Sumber : Hasil Penelitian (2021)

Gambar 8. Entity Relationship Diagram (ERD)

5. User Interface

Tabel 6. User Interface

Nama Form	Gambar	Keterangan
Login		Pada saat sistem dijalankan, pengguna wajib melakukan login terlebih dahulu untuk bisa masuk ke sistem.

<p>Menu Utama</p>		<p>Setelah pengguna berhasil login, sistem akan menampilkan menu utama yang berisikan Data Material, Data SPK, Transaksi dan Laporan.</p>
<p>Data Material</p>		<p>Pada menu Data Material, pengguna dapat melakukan tambah data material, update data material dan hapus data material.</p>
<p>Data SPK</p>		<p>Pada menu Data SPK, pengguna dapat melakukan tambah data SPK, update data SPK dan hapus data SPK.</p>
<p>Transaksi</p>		<p>Pada menu Transaksi, pengguna dapat melakukan tambah data transaksi, dan hapus data transaksi.</p>
<p>Laporan</p>		<p>Pada menu Laporan, pengguna dapat melakukan cetak laporan transaksi berdasarkan Jenis Laporan, Periode dan Tanggal Laporan yang diperlukan.</p>

Sumber : Hasil Penelitian (2021)

KESIMPULAN

Berdasarkan hasil penelitian, didapatkan bahwa dengan diterapkannya perancangan sistem inventory yang diusulkan dapat menyelesaikan masalah dan mempermudah dalam proses pengelolaan inventory atau persediaan baik dalam pengelolaan material masuk, material keluar dan juga stok material, sehingga dalam proses kerja dapat menghasilkan sebuah keefisienan dan ke efektifan dalam pengelolaan persediaan atau inventory. Implementasi sistem informasi inventory dapat menyesuaikan, dengan melakukan pendataan ulang terhadap stok-stok material yang saat ini sudah masuk gudang dan dapat meminimalisir kesalahan yang terjadi dalam pengolahan data inventory.

Untuk pengembangan selanjutnya, sistem informasi inventory dapat dikembangkan menggunakan aplikasi yang berbasis web agar lebih fleksibel saat pengaksesan.

REFERENSI

- Ali Edward. (2019). Buku Ajar Rekayasa Perangkat Lunak. In *Journal of Chemical Information and Modeling* (Vol. 53, Issue 9).
- Astuti, W. Y. (2017a). *Perancangan sistem informasi penjualan skincare berbasis web*. <https://repository.bsi.ac.id/index.php/repo/viewitem/27480>
- Astuti, W. Y. (2017b). Perancangan sistem informasi penjualan skincare berbasis web. *Journal of Physics A: Mathematical and Theoretical*, 44(8), 8–24. <https://doi.org/10.1088/1751-8113/44/8/085201>
- Batubara, H. H. (2018). *Pembelajaran Berbasis Web Dengan Moodle Versi 3.4* (01 ed.). Deepublish.
- Fadallah, M. F., & Rosyida, S. (2018). Program Pemesanan Percetakan Berorientasi Objek dengan Pemodelan Unified Modeling Language. *Jurnal Sistem Informasi Stmik Antar Bangsa*, VII(1).
- Mauludin, Y. (2019). PERANCANGAN INVENTORY PACKAGING PADA PT.SANTOS JAYA ABADI. <https://doi.org/10.31227/osf.io/n4f68>
- Rakhmah, Astri Hijratul, & Alrasyid, M. I. (2018). *Perancangan Sistem Informasi Seleksi Karyawan*. 4(1), 21–33.
- Sukanto, & Shalahuddin. (2018). Rekayasa Perangkat Lunak. Bandung: Informatika. *Jurnal Manajemen Sistem Informasi*, 2(2).
- Suri, N. (2019). Bab II Landasan Teori. *Journal of Chemical Information and Modeling*, 53(9), 1689–1699.
- Susanto, A. (2017). Sistem Informasi Manajemen Konsep dan Pengembangan Secara Terpadu. In *Bandung: Linggar Jaya*.