

BAB II

LANDASAN TEORI

2. 1. Penelitian terdahulu

Sebelumnya terdapat beberapa penelitian yang meneliti tentang penelitian seperti ini dan dijadikan sebagai bahan referensi. Adapun kajian penelitian akan dijabarkan sebagai berikut.

2.1.1 Sistem Informasi Inventory Obat Pada Apotek Tritunggal Farma Jakarta 2020

Menurut (Rifky, Nia, Tika, 2020) Di penelitian ini terdapat informasi dimana pencatatan order obat, penerimaan obat, pengeluaran obat sampai penyimpanan data-data lainnya yang berhubungan dengan proses persediaan obat sampai dengan pembuatan laporan, pencatatan data, juga kurangnya tingkat keakuratan laporan yang dibuat dan keterlambatan dalam pencarian data-data yang diperlukan. Maka itu dapat disimpulkan perlunya dibuat sistem persediaan obat yang terkomputerisasi untuk mengurangi terjadinya kesalahan dalam penginputan data keluar masuknya obat pada apotek dan untuk menyediakan informasi yang lebih akurat tentang stok barang yang tersedia.

2.1.2 Pengembangan Sistem Informasi Inventory Obat Dan CRM Pada Apotek Sentosa Dan Klinik Pranata 2023

Menurut (Tobias, Eka, 2023) Pada penelitian ini dapat ditarik informasi bahwa Penanganan data dengan sistem manual ini mempunyai beberapa kendala, diantaranya menyebabkan terjadinya kesalahan dalam pengecekan stok obat, dan setiap laporannya terjadi redundancy data, serta kurang telitinya dalam pencatatan stok obat yang masuk maupun keluar sehingga terjadinya kekeliruan dalam pencatatan stok akhir. Maka dengan dirancangnya sebuah sistem informasi, diharapkan semua obat-obat yang masuk dan keluar yang ada di gudang dapat diketahui dengan jelas tanpa adanya kesalahan-kesalahan yang sifatnya mengganggu dalam proses penyediaan obat nantinya. Dan dengan menggunakan web, akan lebih mudah dalam menyajikan informasi dan dapat memonitor keadaan persediaan obat-obatan

2.1.3 Sistem Inventory Obat Pada Puskesmas Tapalang 2021

Menurut (Zaenuddin, Assidiq, Edy, 2021) Dalam penelitian ini dapat ditarik informasi bahwa proses mengetahui jumlah stok obat kadang akan mengalami kesulitan jika harus melakukan pengecekan secara manual menggunakan berkas fisik. Apabila dilihat dari sisi efisiensi kerja, tentu saja hal ini membuat operasional terkesankurang handal dan memiliki banyak resiko, karena berkas-berkas ini mudah rusak dan jika rusak tidak terdapat data cadangan, maka perlu dibuat sebuah sistem inventori.

2.1.4 Perancangan Sistem Informasi Inventori Stok Obat Studi Kasus: Apotek Mitra Manakarra 2020

Menurut (Herlina, 2020) Dalam penelitian ini dapat ditarik kesimpulan yaitu dalam tata kelola apoteknya adalah banyak obat yang terdeteksi kadaluarsa (expired) di gudang obat, dan jika persediaan obat overstock dalam jumlah besar, dapat mengakibatkan modal usaha menjadi menganggur (idle money) serta tingginya resiko kerusakan obat (expired date). Tetapi sebaliknya, jika stok obat terlalu sedikit akan terjadi stockout dan mengakibatkan kerugian penjualan obat bagi rumah sakit. Dengan adanya Sistem Informasi Inventory Stok Obat yang berbasis komputer akan memudahkan pihak gudang apotek untuk melakukan pengecekan stok obat, sehingga obat yang expired dapat terkontrol. Sistem juga sangat membantu dalam pembuatan laporan bulanan, sehingga laporan bisa terselesaikan tepat waktu, dan dengan informasi laporan bulanan yang akurat akan memudahkan level top manager dalam pengambilan keputusan terkait stok obat.

2.1.5 Sistem Informasi Inventori Obat Berbasis Web di Rumah Sakit Universitas Riau 2019

Menurut (Salhazan, Rihhadatul, 2019) dapat disimpulkan informasi yaitu Pengelolaan data persediaan obat di rumah sakit ini masih sederhana dengan melakukan pencatatan obat di kartu stok dan Ms. Excel. Hal ini dapat menyulitkan petugas jika ingin melakukan pengecekan terhadap obat yang menuju masa expired, maupun melihat stok obat secara realtime karena banyaknya jenis obat yang ada. Solusi dari masalah tersebut adalah dengan membangun sistem informasi inventori obat berbasis web yang digunakan untuk mengelola data obat dan transaksi di

apotek. Sistem yang dibangun dapat melakukan penyimpanan, pengolahan dan pencarian data obat serta laporan bulanan. Sistem juga dilengkapi dengan fitur pemberitahuan obat yang akan kadaluarsa dan stok obat yang menipis melalui notifikasi email. Pengujian sistem telah dilakukan oleh lima orang pegawai pada Bagian Farmasi Rumah Sakit Universitas Riau. Dari hasil kuesioner pengujian sistem didapatkan hasil akhir 80.8% dengan kategori “sangat memuaskan”.

Tabel 1. 2 Penelitian Sebelumnya

Judul Penelitian	Peneliti & tahun Penelitian
Sistem Informasi Inventori Obat Pada Apotek Berbasis Web (Studi Kasus: Apotek Asembagus Situbondo)	Penelitian dilakukan oleh M.Firmansyah Alnaufal G pada tahun 2019
Tujuan	Ruang lingkup
Tujuan Penelitian pada Apotek Asembagus adalah untuk membuat sebuah sistem informasi inventori berbasis web.	Ruang Lingkup penelitian terletak di Apotek Asembagus
Hasil Penelitian	
Hasil dari penelitian yang dibuat adalah sebuah sistem informasi yang dapat membantu mengontrol stok obat sehingga mempermudah untuk melakukan pemesanan. Tetapi masih memiliki kekurangan dimana belum terdapat format master satuan serta belum ada sistem purchasing untuk supplier.	
Perbedaan dengan penelitian yang akan di lakukan	
Sistem yang akan penulis buat akan mengembangkan sistem dan menambahkan fitur master satuan yang belum dibuat di sistem sebelumnya.	

2.2 Pengertian Sistem Informasi

Menurut (Romney, Marshall B, & Steinbart, 2015), sistem informasi adalah rangkaian dari dua atau lebih komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiri dari subsistem yang lebih kecil yang mendukung sistem yang lebih besar. Sedangkan, Sistem Informasi (SI) adalah kombinasi dari teknologi informasi dan aktifitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen (Nugroho, 2016).

2.3 Framework Codeigniter (CI)

Framework adalah sekumpulan perintah atau fungsi dasar yang membentuk aturan – aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi *website*, kita harus mengikuti aturan dari *framework* tersebut (Wardana, 2010). CodeIgniter adalah sebuah *framework* untuk web yang dibuat dalam format PHP. *Codeigniter* dapat mempercepat proses pembuatan web , karena semua *class* dan modul yang dibutuhkan sudah ada dan programmer hanya tinggal menggunakannya kembali pada aplikasi web yang akan dibuat (Prabowo, 2015).

2.4 Pengertian Sublime Text Editor

Sublime Text Editor adalah editor *text* untuk berbagai bahasa pemrograman termasuk pemrograman PHP. *Sublime Text Editor* juga mendukung banyak bahasa pemrograman dan bahasa *markup*, dan fungsinya dapat ditambah dengan plugin, dan *Sublime Text Editor* tanpa lisensi perangkat lunak (Pasaribu, 2017)

2.5 Pengertian Apotek

Apotek adalah toko tempat meramu dan menjual obat berdasarkan resep dokter serta dokter serta memperdagangkan barang medis (Andita, et al., 2016). Berdasarkan Peraturan Pemerintah No. 25 Tahun 1980 dan Permenkes No. 922/MenKes/X/1993, Apotek adalah suatu tempat tertentu dimana dilakukan pekerjaan kefarmasian dan penyaluran obat kepada masyarakat.

2.6 Pengertian Inventori

Persediaan atau inventory adalah bahan atau barang yang disimpan yang akan digunakan untuk memenuhi tujuan tertentu. Setiap perusahaan yang melakukan kegiatan usaha umumnya memiliki persediaan. Keberadaannya tidak saja dianggap sebagai beban (*liability*) karena merupakan pemborosan (*waste*), tetapi sekaligus juga dapat dianggap sebagai kekayaan (*asset*) yang dapat segera dicairkan dalam bentuk uang tunai (*cash*). (Riadi, 2018)

2.7 Pengertian Basis Data (*Database*)

Database adalah kumpulan data yang terhubung dan disimpan di dalam suatu media, yang tidak memerlukan kerangkapan data, jikapun ada maka kerangkapan data tersebut harus seminimal mungkin. Dan disimpan dengan berbagai cara sehingga proses penambahan, modifikasi, dan pengambilan data dapat dilakukan dengan mudah dan terkontrol (Sibi, M., & Rafie, 2019)

2.8 Unified Modelling Language (UML)

Unified Modelling Language atau UML merupakan suatu metode untuk menganalisa proses bisnis dan kemudian memodelkannya dalam notasi-notasi tertentu. Notasi-notasi tersebut mempunyai aturan-aturan khusus yang mencerminkan suatu pengertian tertentu (Nugroho J. , 2006)

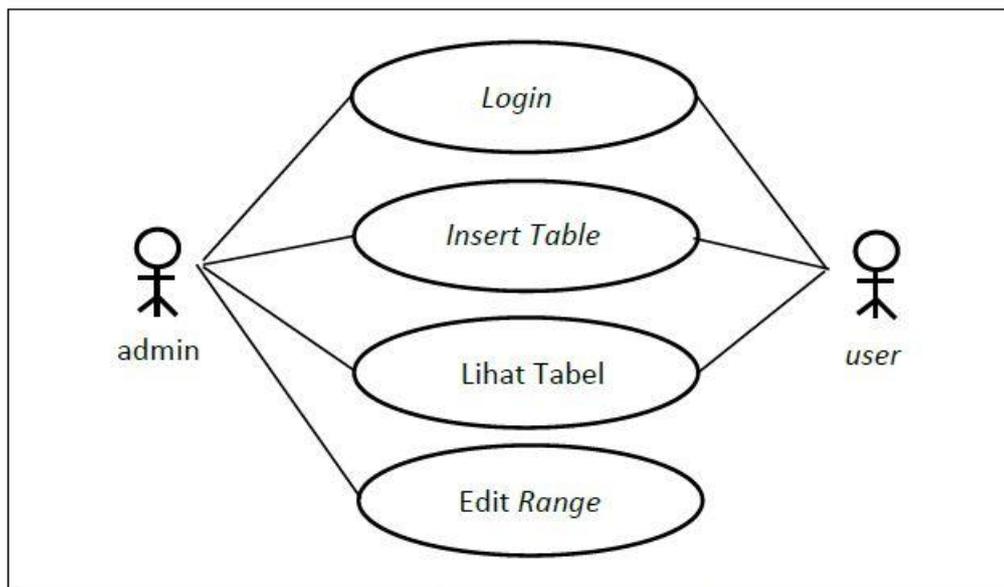
2.8.1 Use Case Diagram

Use Case Diagram adalah *Diagram* yang penting untuk memvisualisasikan, menspasifikasikan, dan mendokumentasikan kebutuhan kebutuhan dan perilaku sistem. *Use Case Diagram* terdiri dari *actor*, *use case*, dan hubungannya. *Use Case Diagram* digunakan untuk menjelaskan kegiatan apa saja yang dapat dilakukan oleh pengguna atau bisa disebut dengan *user* yang sedang berjalan (Novita & Sari, 2015). Notasi *Use Case Diagram* dapat dilihat di Tabel 2.1 dan Gambar 2.1

Tabel 2. 1 Notasi *Use Case Diagram*

Simbol	Keterangan
	<p>Aktor: merupakan pemakai sistem, dapat berupa manusia atau sistem terotomatisasi lain. Aktor seseorang yang berinteraksi dengan sistem, yaitu siapa dan apa yang terdapat didalam sistem. Aktor menjelaskan peran bukan pemakai individu dari sistem. Aktor mempunyai nama yang menyatakan peran aktor.</p>
	<p>Use Case: merupakan cara spesifik penggunaan didalam sistem oleh seorang aktor. <i>Use Case</i> melihat interaksi antara aktor yang ada dengan</p>

	sistem. <i>Use Case</i> mengemukakan suatu kerja yang tampak.
—————	Assosiation: hubungan antara aktor dan <i>use case</i>
<pre> <<include>> → </pre>	Include: perilaku <i>use case</i> merupakan bagian dari <i>use case</i> lain
<pre> <<extend>> ← </pre>	Extend: perilaku <i>use case</i> memperluas <i>use case</i> yang lain



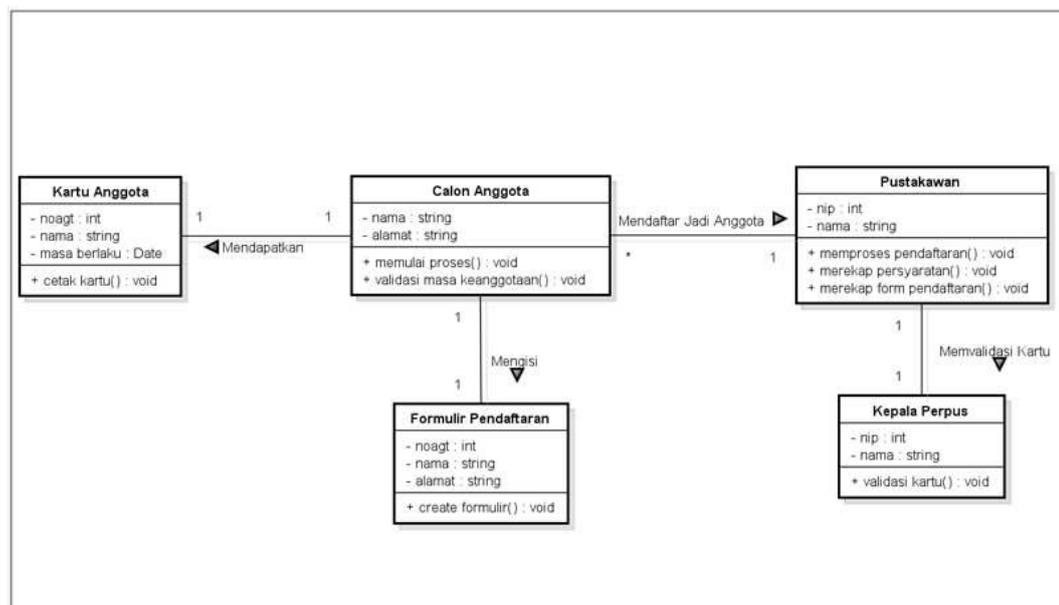
Gambar 2.1 Use Case Diagram

2.8.2 Class Diagram

Class Diagram adalah *Diagram* yang menunjukkan *class-class* yang ada di sistem dan hubungannya secara *logic* (Novita & Sari, 2015). *Notasi Class Diagram* dapat dilihat di Gambar 2.2 dan Gambar 2.3

Simbol	Deskripsi
<p>Kelas</p> <pre> classDiagram class nama_kelas { + atribut + operasi } </pre>	kelas pada struktur sistem
<p>Antarmuka / interface</p> <pre> classDiagram class nama_interface { } </pre>	sama dengan konsep interface dalam pemrograman berorientasi objek

Gambar 2. 2 Notasi Class Diagram



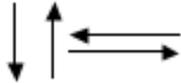
Gambar 2. 3 Class Diagram

2.8.3 Activity Diagram

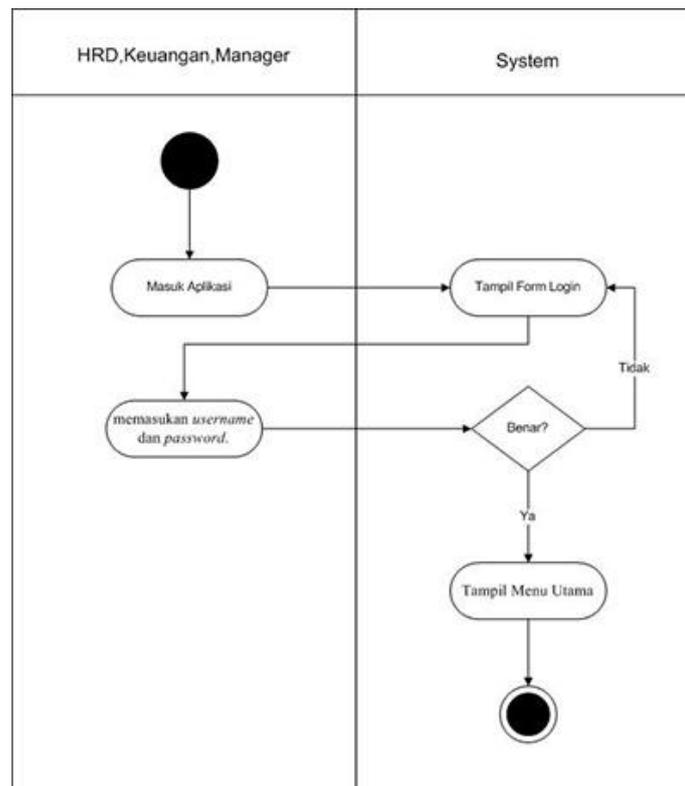
Activity Diagram adalah sebuah diagram yang memiliki berbagai elemen yang diantara sebagai berikut :

- a. Sebuah keadaan awal (*start state*) dan keadaan akhir (*end state*).
- b. Aktivitas-aktivitas yang menggambarkan satu tahapan dalam *workflow* tersebut.
- c. Transisi yang menggambarkan keadaan apa yang mengikuti suatu keadaan lainnya.
- d. Keputusan (*decision*), elemen yang menyediakan pilihan alur dalam *workflow*.
- e. Batang penyelaras (*synchronization bar*) memperlihatkan sub alur paralel.
- f. *Swimlane* yang menjelaskan pemeran bisnis yang bertanggung jawab terhadap aktivitas yang dikandungnya.

Notasi *Activity Diagram* dapat dilihat di Gambar 2.4 dan Gambar 2.5

NO	GAMBAR	NAMA	KETERANGAN
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk atau diawali.
4		Activity Final Node	Bagaimana objek dibentuk dan diakhiri
5		Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
6		Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

Gambar 2. 4 Notasi Activity Diagram



Gambar 2. 5 Activity Diagram

2.8.4 Sequence Diagram

Sequence Diagram adalah sebuah *Diagram* yang digunakan guna menjelaskan dan menampilkan interaksi antar objek di dalam sebuah sistem secara rinci. Selain itu, *Sequence Diagram* pun juga akan menampilkan pesan atau perintah yang dikirim, bersamaan dengan waktu pelaksanaannya. Objek yang berhubungan dengan proses operasi yang berjalan biasanya diurut dari kiri ke kanan.

Notasi *Sequence Diagram* dapat dilihat di Gambar 2.6

No	Simbol	Nama	Fungsi
1		Object	Komponen utama Sequence Diagram
2		Actor	Menggambarkan orang yang sedang berinteraksi dengan sistem
3		Entity Class	Menggambarkan hubungan kegiatan yang akan dilakukan
4		Boundary Class	Menggambarkan sebuah penggambaran dari form
5		Control Class	Menggambarkan penghubung antara boundary dengan tabel
6		Life Line	Menggambarkan tempat mulai dan berakhirnya sebuah message
7		Message	Menggambarkan pengiriman pesan

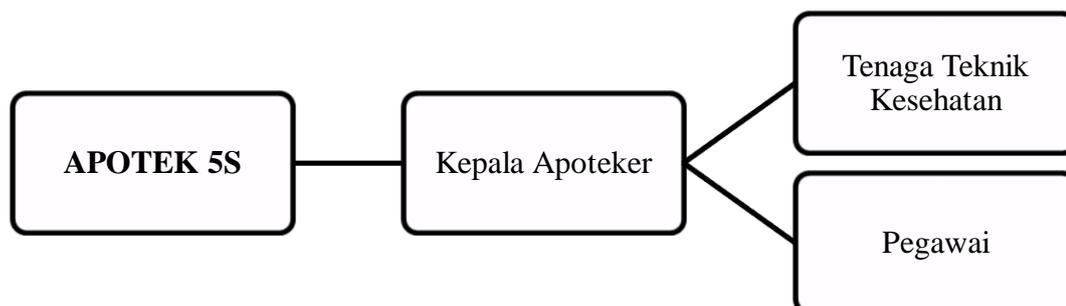
Gambar 2.6 *Sequence Diagram*

2.9 Gambaran umum perusahaan

2.9.1 Profil Studi Kasus

Apotek 5S adalah sebuah perusahaan swasta yang bergerak di bidang kesehatan, Didirikan pada Bulan Oktober 2014 yang beralamatkan di Daerah Kepuh di Kecamatan Sukun Kota Malang. Apotek 5S sendiri adalah sebuah singkatan dari 5 kata yang berlawanan huruf S yaitu Selalu Siap Saat Sehat & Sakit yang berarti Apotek 5S ini selalu siap melayani pelanggan yang sehat maupun sakit.

2.9.2 Struktur Organisasi Perusahaan



Gambar 2.7 Struktur Organisasi Perusahaan

Struktur Organisasi Perusahaan menjelaskan orang-orang yang bekerja di Apotek 5S sesuai dengan jabatannya masing-masing. Apotek 5S di pimpin oleh seorang Apoteker yang merupakan pemilik dari Apotek 5S tersebut yang memiliki 2 bawahan yaitu Tenaga Teknik Kesehatan dan Pegawai. Berikut adalah penjelasan dari tugas-tugas jabatan di Apotek 5S :

Tabel 2. 2 Tugas Jabatan

Jabatan	Tugas – Tugasnya
Kepala Apoteker	<ol style="list-style-type: none"> 1. Menentukan Keputusan tertinggi di Apotek 2. Bertanggung Jawab keuntungan dan kerugian Apotek 3. Mengontrol keluar masuknya pasokan obat di Apotek
Tenaga Teknik Kesehatan	<ol style="list-style-type: none"> 1. Meracik obat 2. Melakukan transaksi dagang dengan pelanggan 3. Menerima <i>supply</i> barang dari <i>supplier</i>
Pegawai	<ol style="list-style-type: none"> 1. Melakukan transaksi dagang dengan pelanggan 2. Menerima <i>supply</i> barang dari <i>supplier</i>

2.9.3 Visi dan Misi Perusahaan

a. Visi :

Menjadi pelayan masyarakat yang selalu memberikan pelayanan terbaik di bidang farmasi dan alat-alat kesehatan kepada masyarakat yang sehat maupun sakit

b. Misi :

Menyediakan obat-obat dan alat-alat kesehatan yang dapat membantu masyarakat serta melayani konsultasi kesehatan terhadap masyarakat