



BAB 2 LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian Terdahulu merupakan sarana untuk menunjukkan keaslian penelitian. Peneliti melakukan peneliti yang tidak jauh dari hasil penelitian yang sudah pernah dilakukan, dengan melihat dan meneliti sebagai bahan referensi perbandingan kajian materi.

Penelitian pertama, (Denny Rusdianto & Angeu Nurdesni, 2020) tentang Perancangan Sistem Informasi Persediaan Obat Berbasis *Web* Pada Apotek Andir Farma, penelitian tersebut menjelaskan bahwa apotek Andir farma yang terletak di jalan Raya NO. 54 Ciparay masih memiliki sebuah kendala dalam hal pergudangan, misalnya sering terjadinya keterlambatan pembuatan laporan persediaan obat yang disebabkan oleh penumpukan data transaksi barang masuk dan data barang keluar. Dan untuk itu penulis berinisiatif membuat sistem informasi persediaan obat dengan menggunakan metodologi *ModelDriven Development* (MDD) yang memanfaatkan alat bantu seperti *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.

Penelitian kedua, (Wawan Alakel, Imam Ahmad & Eli Budi Santoso, 2019) tentang Sistem Informasi Akuntansi Persediaan Obat Metode *First In First Out* (Studi Kasus: Rumah Sakit Bhayangkara Polda Lampung), pada penelitian yang dilakukan menjelaskan bahwa Rumah Sakit ini masih terdapat sebuah masalah yang dihadapi yakni proses pencatatan sisa obat dan stok tidak sesuai dengan kondisi persediaan dan masih sangat acak acakan dalam persediaan obat. Maka dalam hal ini penulis memiliki sebuah solusi dengan membangun sebuah sistem akuntansi persediaan obat dengan metode *First In First Out* yang bertujuan meningkatkan ketepatan dalam

persediaan obat serta efisiensi waktu yang dibutuhkan dalam pelayanan.

Penelitian ketiga, (Hari Sugiarto & Ronaldo Sibarani, 2019) tentang Rancang Bangun Aplikasi Persediaan Obat Pada Klinik Umum Grace Medika Cikarang, pada penelitian yang telah dilakukan menjelaskan bahwa pada Klinik tersebut terdapat sebuah masalah yakni masih menggunakan cara manual dalam pengumpulan data, sehingga keakuratan data masih minim dan kesulitan saat pencarian data histori stok. Dengan permasalahan tersebut, penulis berinisiatif membuat sebuah aplikasi persediaan obat dengan menggunakan metode *Waterfall Model* yang bertujuan untuk menanggulangi masalah yang dihadapi oleh Klinik Umum Grace Medika Cikarang.

Penelitian keempat, (Miwan Kurniawan Hidayat, 2019) tentang Rancang Bangun Sistem Informasi Persediaan Obat Pada Puskesmas Babelan I Kabupaten Bekasi, dari penelitian yang telah dilakukan peneliti menemukan jika Puskesmas tersebut masih menggunakan sistem konvensional, yaitu data dicatat dalam sebuah buku dan *file Microsoft Office*. Permasalahan utama yang terjadi yaitu belum ada suatu sistem informasi yang berbasis komputer untuk menangani pengolahan data, sehingga menyebabkan informasi dan pembuatan laporan yang diperlukan tidak bisa tepat waktu. Maka dari itu penulis berinisiatif membuat sebuah sistem informasi yang mencakup sistem konvensional dengan menggunakan metode *Waterfall* sehingga nantinya diharapkan pada sistem ini diperoleh integrasi antar bidang dan terciptanya keakuratan data.

Tabel 2.1 penelitian terdahulu

No	Judul	Nama Peneliti	Persamaan	Perbedaan
1	Perancangan Sistem Informasi Persediaan Obat	(Denny Rusdianto & Angeu	Problem yang diangkat sama – sama tentang	Penelitian menggunakan an metodologi <i>ModelDriven</i>

No	Judul	Nama Peneliti	Persamaan	Perbedaan
	Berbasis Web Pada Apotek Andir Farma	Nurdesni, 2020)	sistem manual yang belum terdigitalisasi	<i>Development (MDD)</i>
2	Sistem Informasi Akuntansi Persediaan Obat Metode First In First Out (Studi Kasus: Rumah Sakit Bhayangkara Polda Lampung)	(Wawan Alakel, Imam Ahmad & Eli Budi Santoso, 2019)	Problem yang diangkat sama – sama tentang sistem manual yang belum terdigitalisasi	Penelitian menggunakan Metode <i>First In First Out</i>
3	Rancang Bangun Aplikasi Persediaan Obat Pada Klinik Umum Grace Medika Cikarang	(Hari Sugiarto & Ronaldo Sibarani, 2019)	Studi kasus sama – sama di bidang pelayanan dan menggunakan metode <i>Waterfall</i>	Implementasi program yang digunakan pada sistem informasi persediaan obat menggunakan bahasa pemrograman <i>Visual Basic 6.0</i>
4	Rancang Bangun Sistem	(Miwan Kurniawan	Studi kasus sama – sama di	Sistem ini terdapat halaman

No	Judul	Nama Peneliti	Persamaan	Perbedaan
	Informasi Persediaan Obat Pada Puskesmas Babelan I Kabupaten Bekasi	Hidayat, 2019)	bidang pelayanan dan menggunakan metode <i>Waterfall</i>	resep obat untuk pasien

2.2 Kajian Pustaka

Berikut ini adalah kajian pustaka yang dilakukan oleh penulis guna memperkaya pengetahuan dan membantu penulis dalam melakukan perancangan sistem.

2.2.1 Aplikasi

Di era globalisasi saat ini sebuah instansi, lembaga maupun sebuah organisasi tidak lepas dengan yang namanya proses komputerisasi, karena itu jika sebuah instansi, lembaga serta organisasi tidak mengikuti arus globalisasi dibidang teknologi, maka mereka akan tertinggal dari yang lain. Hal ini dibuktikan bahwa seluruh proses yang ada sudah menggunakan bantuan *software* berbentuk aplikasi dalam menjalankan alur proses bisnisnya. Aplikasi adalah penggunaan atau penerapan suatu konsep yang menjadi suatu pokok pembahasan. Aplikasi dapat diartikan juga sebagai program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu. (Noviansyah, 2008)



2.2.2 Sistem Informasi

Dalam sebuah organisasi diperlukan sebuah sistem baik dari segi *software* maupun *hardware* yang mampu mengatur jalannya sebuah alur proses pada sebuah divisi tiap organisasi maupun lembaga yang bertujuan untuk mempermudah serta membantu pekerjaan agar lebih efektif dan efisien. Menurut Mulyanto dalam Kuswara dan Kusmana (2017), menyatakan bahwa sistem informasi adalah suatu sistem yang terdiri dari kumpulan komponen sistem, yaitu *software*, *hardware* dan *brainware* yang memproses informasi menjadi sebuah *output* yang berguna untuk mencapai suatu tujuan tertentu dalam suatu organisasi.

2.2.3 Manajemen Persediaan

Pada sebuah manajemen keluar masuk suatu barang diperlukan perencanaan mengenai kebutuhan mendatang yang dilakukan dengan cara mengatur suku cadang dalam beberapa tertentu sehingga bisa memaksimalkan proses manajemen stok di masa mendatang. Menurut Ahmad (2018) manajemen persediaan adalah proses penyimpanan bahan atau barang untuk memenuhi tujuan tertentu seperti, penggunaan untuk proses produksi atau perakitan yang nantinya akan dijual kembali atau penggunaan suku cadang dari suatu peralatan atau mesin.

2.2.4 Basis Data

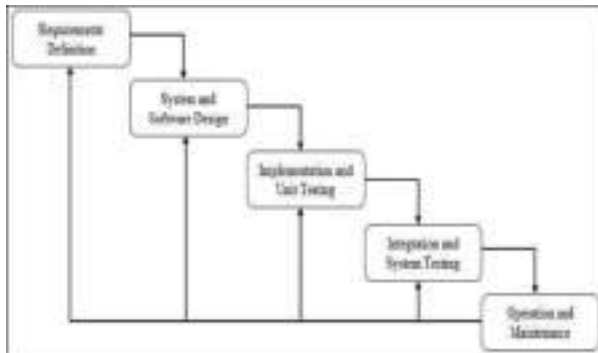
Pada perancangan sebuah aplikasi, sebuah *database* atau basis data tentunya sangat dibutuhkan guna mendukung proses organisasi agar sebuah informasi bisa terintegrasi dan sanggup selaras dengan tujuan yang ingin dicapai sebuah organisasi. Menurut Jayanti & Sumiari (2018), “Basis Data merupakan data yang terintegrasi, yang diorganisasi untuk memenuhi kebutuhan para pemakai di dalam suatu organisasi”.



Berdasarkan diatas, dapat disimpulkan bahwa basis data merupakan data yang dapat didesain dan berintegrasi sehingga dapat memenuhi kebutuhan *User* dalam perusahaan atau organisasi.

2.2.5 Metode *Waterfall*

Dalam sebuah perancangan aplikasi diperlukan sebuah metode yang digunakan dengan tujuan untuk mengurangi resiko kesalahan dan pengambilan jalan pintas serta menjamin adanya konsistensi proses, Ian Sommerville (2011) menjelaskan bahwa ada lima tahapan pada Metode *Waterfall* yang digunakan dalam proses perancangan sebuah aplikasi, yakni *Requirements Analysis and Definition, System and Software Design, Implementation and Unit Testing, Integration and System Testing, dan Operation and Maintenance*.



Gambar 2.1 alur waterfall (sumber : Ian Sommerville, 2011)

Berdasarkan Gambar diatas, adapun tahapan dari metode *Waterfall* adalah :

a. *Requirements analysis and definition*

Tahap analisis dilakukan untuk memenuhi kebutuhan sistem. Pengumpulan data dalam penelitian dengan

meminta data ke tempat studi kasus Dinas Kesehatan Kabupaten Jombang, selanjutnya kepada Petugas PSC 119 melalui wawancara dan observasi dari data yang didapatkan.

b. *System and software design*

Tahap desain dilakukan untuk membuat simulasi rancangan yang siap untuk diimplementasikan, berdasarkan tahap-tahap sebelumnya. Pada tahap ini akan dibuat rancangan sistem seperti *UML (Unified Modelling Language)*, *desain CDM (Conceptual Data Model)*, dan *PDM (Physical Data Model)* serta desain antar muka pengguna.

c. *Implementation and unit testing*

Pada tahap ini, dilakukan pengembangan simulasi perangkat lunak sistem berdasarkan desain sistem yang dihasilkan. Desain harus diterjemahkan dalam bentuk yang dapat dimengerti oleh mesin dengan menggunakan Bahasa pemrograman yang telah ditentukan.

d. *Integration and system testing*

Setelah diimplementasikan, akan dilakukan pengujian menggunakan *blackbox testing* terhadap aplikasi tersebut pada pengguna, sehingga akan diperoleh kesesuaian hasil implementasi dengan hasil analisis, serta harapan dan tujuan. Pada tahap ini juga akan diperoleh kelebihan dan kekurangan yang dimiliki oleh sistem ini.

e. *Operation and maintenance*

Pada langkah terakhir, program sudah bisa digunakan. Selain itu, pada tahap ini dapat dilakukan untuk pengembangan sistem seperti menambah modul atau fitur baru.

2.2.6 MySQL

Aplikasi *Web Server* diperlukan dalam proses pengembangan sebuah sistem, hal ini dikarenakan sebuah sistem akan berjalan jika terdapat *server* yang sanggup



mengatur akses dari basis data dalam hal jaringan lokal maupun yang lainnya. *MySQL* adalah sistem basis data yang efisien, bebas, dan banyak digunakan yang mengimplementasikan *SQL*. *Server database MySQL* mendukung akses lokal dan jaringan. Ini mendukung sistem hak istimewa dan kata sandi untuk menentukan siapa yang dapat mengakses atau memodifikasi apa yang ada di sistem basis data (Sebesta, 2014).

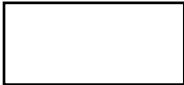
2.2.7 Usability

Sebuah produk perangkat lunak atau sistem yang sudah dirancang perlu diuji untuk mengetahui kualitas serta pengalaman penggunaan ketika berinteraksi dengan perangkat tersebut. Menurut ISO 9241 (1999), *usability* adalah tingkat dimana produk bisa digunakan oleh pengguna tertentu untuk mencapai tujuannya dengan lebih efektif, efisien, dan memuaskan dalam ruang lingkup penggunanya.


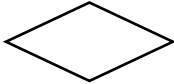

2.2.8 Entity Relationship Diagram (ERD)

Pada perancangan sebuah sistem diperlukan sebuah Gambaran suatu database yang menunjukkan entitas beserta atributnya yang bertujuan untuk mengetahui relasi antar objek. Menurut Rusmawan (2019) *Entity Relationship Diagram (ERD)* adalah Gambaran suatu model data yang menjelaskan seluruh hubungan, entitas, dan batasan untuk menyelesaikan pengembangan *system*. Berikut simbol dari *ERD* tersebut.

Table 2.2 simbol ERD (sumber : Rusmawan, 2019)

Simbol	Nama	Keterangan
	Entitas	Berisi tentang Tabel-Tabel yang terdapat dalam sistem.



	Atribut	Menjelaskan tentang <i>field</i> atau isi dari Tabel yang terdapat didalam sistem.
	<i>Relationship</i>	Mendeskripsikan hubungan antar Tabel yang terdapat didalam sistem.
	<i>Arrow</i>	Garis yang menghubungkan antara himpunan entitas, atribut, dan himpunan relasi.

2.2.9 UML (*Unified Modelling Language*)

Menurut Rosa dan Shalahuddin (2014), berpendapat bahwa *UML (Unified Modeling Language)* adalah “Salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisa & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”. Sedangkan Mulyani (2016) mengatakan *UML (Unified Modeling Language)* adalah “Sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem”.

Dari beberapa penjelasan teori tersebut dapat disimpulkan bahwa *UML (Unified Modeling Language)* adalah bahasa yang sering digunakan untuk membangun sebuah sistem perangkat lunak dengan melakukan penganalisaan desain dan spesifikasi dalam pemrograman berorientasi objek.





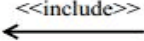
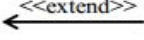
UML (Unified Modeling Language) memiliki diagram-diagram yang digunakan dalam pembuatan aplikasi berorientasi objek, diantaranya (Rosa dan Shalahuddin, 2014):



1. Use Case Diagram

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Berikut simbol yang terdapat *pada use case diagram*.

Tabel 2.3 simbol use case (sumber : <https://www.dicoding.com>)

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi








2. Activity Diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Perlu diperhatikan bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut simbol yang ada di *activity diagram*.

3. Class Diagram

Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Berikut simbol yang terdapat pada *class diagram*.

Tabel 2.4 class diagram (sumber : sugiarti, 2018)




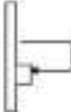


NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	<u>Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor</u>
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	<u>Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri</u>
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

4. Sequence Diagram



Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram sekuen maka harus diketahui objek- objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Berikut simbol yang ada pada *Sequence diagram*.

Tabel 2.5 *sequence diagram*

Gambar	Nama	Keterangan
	Entity Class	Gambaran sistem sebagai landasan dalam menyusun basis data
	Boundary Class	Menangani komunikasi antar lingkungan sistem
	Control Class	Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika
	Recursive	Pesan untuk dirinya
	Activation	Mewakil proses durasi aktivasi sebuah operasi
	Life Line	Komponen yang digambarkan garis putus terhubung dengan objek



2.2.10 Pengujian *Black Box*

Black Box Testing adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan dan keluar dari perangkat lunak sesuai dengan aplikasi yang dibutuhkan. Pengujian *Black Box* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan (Muharni, 2021). Kelebihan *Black Box Testing* : (1) dapat memilih subset tes secara efektif dan efisien, (2) dapat menemukan cacat, (3) memaksimalkan *testing investment*. Kekurangannya adalah tester tidak pernah yakin apakah perangkat lunak tersebut benar-benar lulus uji.

