

BAB II

TINJAUAN PUSTAKA

2.1 Sistem

Sistem dapat didefinisikan sesuatu yang terdiri dari kesatuan yang utuh dari bagian tertentu yang saling berhubungan untuk mencapai tujuan tertentu. Seperti sistem lainnya, sebuah sistem informasi terdiri atas input (data, instruksi) dan output laporan (Wijoyo *et al*, 2021)

2.2 Monitoring

Monitoring merupakan proses pengumpulan, peninjauan ulang laporan, dan tindakan atas informasi suatu proses yang sedang diimplementasikan. Sehingga mendapatkan data yang bersifat *real time* agar kebenaran data dapat dipercaya dan dipertanggung jawabkan jika terjadi suatu kesalahan (Azhar, Wina, & Asep, 2020).

2.3 Sistem Monitoring

Sistem *Monitoring* adalah proses pemantauan untuk pengumpulan data yang bersifat *real time* sehingga informasi yang dihasilkan dapat menjadi suatu pengambil keputusan agar kebenarannya dapat dipercaya. Sistem *monitoring* juga dapat meminimalisir masalah yang timbul agar dapat langsung diatasi secara efektif (Sulasno dan Saleh, 2020).

2.4 Ketersediaan Barang

Ketersediaan barang yaitu kemudahan yang diberikan oleh penjual kepada pembeli untuk mendapatkan barang yang dibutuhkan. Indikator ketersediaan barang layout penjualan (tata letak barang di display), persediaan barang selalu ada, distribusi barang merata tersedia, kemudahan melakukan pembelian, kelengkapan barang yang tersedia (Sugiharto, 2020).

2.5 Pemesanan

Pemesanan merupakan suatu aktivitas yang dilakukan oleh pemesan sebelum membeli. Untuk mewujudkan kepuasan pelanggan maka perusahaan harus mempunyai sebuah sistem pemesanan yang baik (Simatupang, 2020).

Menurut Kamus Besar Bahasa Indonesia yang dimaksud pemesanan adalah proses, perbuatan, cara memesan tempat atau barang dan sebagainya kepada orang lain.

2.6 Website

Website merupakan halaman yang menampilkan informasi data teks, gambar, suara, video atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis. Halaman pertama sebuah *website* disebut *homepage*. *Website/Situs* merupakan kumpulan informasi atau kumpulan halaman/*page* yang bisa diakses lewat jalur *internet*. Setiap orang di berbagai tempat dan segala waktu bisa menggunakannya selama terhubung secara *online* (Harianto *et al*, 2019).

2.7 PHP

PHP (*Personal Home Page*) adalah pemrograman yang melakukan proses penerjemahan baris sumber menjadi kode mesin yang dimengerti oleh komputer secara dinamis. Pengertian PHP juga merupakan singkatan dari *Hypertext Preprocessor* dengan bahasa yang berbentuk skrip yang bersifat *server side* yang dimana proses pengerjaan kode program dilakukan di *server*, dan hasilnya akan ditampilkan di *browser* (Sihombing, 2022).

2.8 MySql

MySQL merupakan sebuah manajemen basis data menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. Database *Management System* (DBMS) MYSQL multi pengguna dan bersifat gratis. Mysql digunakan sebagai wadah dalam mengelola data yang dapat disimpan dan digunakan kembali dengan cara yang lebih efisien (Setyawan dan Pratiwi, 2019).

2.9 Prototype

Prototype merupakan metode yang digunakan untuk mendapatkan kebutuhan pengguna lebih spesifik dalam segi teknis. Model *prototype* dapat digunakan untuk menyambungkan ketidakpahaman pengguna mengenai teknis dan memperjelas spesifikasi kebutuhan yang diinginkan pengguna kepada pengembang perangkat lunak (Rosa dan Shalahuddin, 2019)

Prototype dapat dimulai dengan mengumpulkan kebutuhan pengguna terhadap perangkat lunak yang akan dibuat, kemudian dirancang menggunakan model *prototype* dalam bentuk *mockup* untuk mempermudah pengguna mengevaluasi *prototype* tersebut. Sehingga hasil dari sistem yang dikembangkan telah sesuai dengan permintaan dan kebutuhan pengguna.



Gambar 2. 1 Metode *Prototype*

1. Kelebihan *Prototype*

- a. Menghemat waktu dan biaya pengembangan
- b. Adanya keterlibatan pemilik sistem sehingga kesalahan sistem bisa diminimalisir dari awal proses
- c. Membantu anggota tim untuk berkomunikasi secara efektif
- d. Klien memiliki kepuasan tersendiri karena sudah memiliki gambaran dari sistem yang akan dibuat
- e. Implementasi atau penggunaan sistem lebih mudah karena klien sudah tahu gambaran sistem sebelumnya

- f. Kemudahan dalam memperkirakan pengembangan sistem selanjutnya
- g. Memungkinkan klien untuk mempersiapkan perangkat lunak yang cocok dengan sistem yang akan dibuat

2. Kelemahan *Prototype*

- a. *Prototype* adalah metode yang menghabiskan banyak waktu jika klien kurang puas ditahapan awal.
- b. Klien terus menerus menambah *requirement* dari sistem, ingin dibuatkan yang seperti inilah seperti itulah sehingga menambah kompleksitas pembuatan sistem.
- c. Sistem akan terhambat jika komunikasi kedua belah pihak tidak berjalan secara efektif.

2.9.1 Tahapan Penelitian

Tahapan dalam penelitian sebagai langkah-langkah penelitian yang harus dikerjakan, berikut adalah tahapan penelitian menggunakan *prototype*.


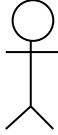

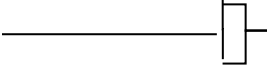
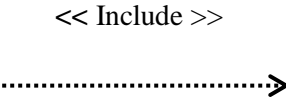
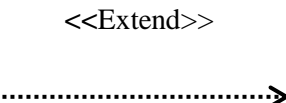
1. Mendengarkan pelanggan
Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, semua kebutuhan dan garis besar sistem yang akan dibuat
2. Membangun dan Memperbaiki *mockup*
Dengan membuat perancangan sementara seperti *mockup* yang berfokus pada penggambaran terhadap sistem yang dibangun kepada pengguna (Misal membuat rancangan input dan output).
3. Pelanggan melihat atau menguji *mockup*
Penggambaran sistem yang digunakan seperti *mockup* perlu di uji kepada pengguna untuk mendapatkan kebutuhan yang sesuai dengan keinginannya, selanjutnya pada sistem yang telah dibangun perlu diuji untuk mengetahui kelayakan terhadap fungsi sistem.

2.10 Unified Modelling Language (UML)

2.10.1 Use Case Diagram

Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Rosa dan Shalahuddin, 2019).



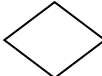

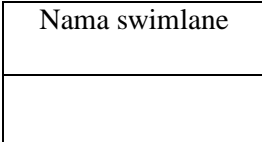

Tabel 2. 1 Simbol *Use Case Diagram*

| No | Simbol | Deskripsi |
|----|---|---|
| 1. |  | <i>Usecase</i> Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal <i>frase</i> nama <i>use case</i> . |
| 2. |  | Aktor Aktor seseorang/sesuatu yang berinteraksi dengan yang akan dibuat. diluar sistem informasi. Biasanya dinyatakan menggunakan kata benda |
| 3. |  | Asosiasi/ <i>association</i> merupakan komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor. |
| 4. |  | Generalisasi (<i>generalization</i>) merupakan hubungan (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum |
| 5. |  | Include berarti <i>use case</i> yang ditambahkan akan dipanggil saat <i>use case</i> tambahan dijalankan. |
| 6. |  | Ekstensi (<i>extend</i>) merupakan <i>use case</i> tambahan ke sebuah <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu. |

2.10.2 Activity Diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis yang menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2019).

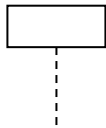

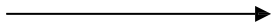
Tabel 2. 2 Simbol *Activity Diagram*

| No. | Simbol | Keterangan |
|-----|---|---|
| 1. |  | Status awal aktivitas sitem, sebuah diagram aktivitas memiliki sebuah status awal. |
| 2. |  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| 3. |  | Percabangan (<i>Decision</i>) merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| 4. |  | Penggabungan (<i>Join</i>) merupakan asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5. |  | Swimlane Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas. |
| 6. |  | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |

2.10.3 Sequence Diagram

Diagram rangkaian menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada eksekusi sebuah *use case* atau operasi (Rosa dan Shalahuddin, 2019).



Tabel 2. 3 Simbol *Sequence Diagram*

| No. | Simbol | Deskripsi |
|-----|---|---|
| 1. | <p><i>Object lifeline</i></p>  | Menggambarkan panjang kehidupan suatu objek selama scenario sedang di buat contohnya |
| 2. | <p><i>Activation</i></p>  | Dimana proses sedang dilakukan oleh <i>object</i> atau <i>class</i> untuk memenuhi pesan atau perintah |
| 3. | <p><i>Message</i></p>  | Sebuah anak panah yang mengindikasikan pesan diantara objek. Dan objek dapat mengirimkan pesan ke dirinya sendiri |


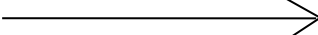
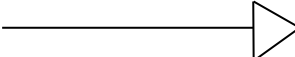
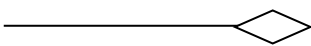
2.10.4 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa dan Shalahuddin, 2019).

Tabel 2. 4 Simbol *Class Diagram*

| No. | Simbol | Deskripsi |
|-----|---|---|
| 1. |  | Kelas pada struktur sistem. |
| 2. | <p><i>Antar Muka/Interface</i></p>  <p>Nama_<i>Interface</i></p> | Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek. |

Tabel 2.4 Simbol *Class Diagram* (Lanjutan)

| No. | Simbol | Deskripsi |
|-----|---|--|
| 3. | <p style="text-align: center;"><i>Asosiasi / Association</i></p>  | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan simbol |
| 4. | <p style="text-align: center;"><i>Asosiasi Berarah / Directed Association</i></p>  | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan symbol. |
| 5. | <p style="text-align: center;">Generalisasi</p>  | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus) |
| 6. | <p style="text-align: center;"><i>Agregasi / aggregation</i></p>  | Relasi antar kelas dengan maksna semua bagian (<i>whole-part</i>) |

2.11 Pengujian Black Box Testing

Black Box Testing merupakan pengujian fungsional struktur program tidak dipertimbangkan. Kasus uji diputuskan berdasarkan persyaratan atau spesifikasi program atau modul lain dan internal yang diuji kepada user. Metode *Black Box Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang output pakai. Program *Black Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan (Ningrum *et al*, 2019).

2.12 Penelitian Terdahulu

Dalam penelitian ini akan digunakan lima penelitian terdahulu yang nantinya dapat mendukung penelitian, berikut ini merupakan penelitian terdahulu yang diambil yaitu:

Tabel 2. 5 Penelitian Terdahulu

| Nama Peneliti | Judul | Masalah | Hasil Penelitian |
|--|--|--|---|
| Mochammad Arief Hermawan Sutoyo, Agnes Patricia (2022) | Perancangan Sistem Informasi Penjualan dan Persediaan Barang Berbasis Web Pada Toko Wonder Paint | Masih melakukan pencatatan di buku besar sesuai kebutuhan setiap penjualannya, persediaan barang juga masih melakukan pencatatan secara manual sehingga akan sulit untuk mengecek barang yang tersedia. | Dengan sistem informasi ini pemakai dapat melakukan pencatatan, pemesanan, persediaan barang tanpa harus menanyakan melalui kontak. Dengan adanya aplikasi ini dapat mempermudah pengguna karena pengguna dapat menggunakan sistem kapan saja dan dimana saja |
| Mohammad Suhatsyah (2020) | Perancangan Sistem Informasi Pemesanan Kamar Hotel Horizon di Kabupaten Karimun Berbasis Web | Proses Pengelolaan data transaksi dan pembuatan laporan keuangan kurang rapi. Kesulitan dalam membuat informasi atau laporan, dikarenakan dalam menyusun laporan harus dicatat dan direkap secara manual maka penyajian laporan menjadi kurang efektif dan efisien | Dengan sistem informasi ini pemakai dapat melakukan pemesanan kamar secara online tanpa harus datang langsung kehotel. Dengan adanya aplikasi ini dapat mempermudah pengguna karena pengguna dapat memesan kamar kapan saja dan dimana saja |
| Priyo Nugraha (2020) | Rancangan Aplikasi Persediaan dan Pemesanan Bahan Kimia Berbasis Java | Selama ini PT. Nusantara Parkerizing menjual kembali | Sistem mempermudah dari segi informasi dan pelayanan |

Tabel 2.5 Penelitian Terdahulu (Lanjutan)

| Nama Peneliti | Judul | Masalah | Hasil Penelitian |
|---------------|--------------------------------|---|---|
| | Pada PT. Nusantara Parkerizing | barang dari beberapa supplier, kemudian melakukan pencatatan transaksi penjualan dan data stok barang masih dengan cara manual pada buku besar sesuai nota yang diterima oleh bagian admin untuk di input ke dalam komputer dengan menggunakan Microsoft Excel. | pemesanan terhadap konsumen. Dengan menyediakan media sistem informasi pemesanan berbasis Java yang dirancang untuk meminimalkan waktu proses pemesanan dengan tujuan meningkatkan pelayanan bagi konsumen, meningkatkan volume pemesanan sehingga pendapatan perusahaan dapat meningkat. |