

BAB II

TINJAUAN PUSTAKA

3.1 Penelitian Terkait

Penelitian yang telah dilakukan oleh Wardinasahira et al., dengan judul “Pengembangan Aplikasi *Website* untuk Penyewaan dan Pembuatan Kostum Menggunakan Metode *Prototype* (Studi Kasus: Ghaniart Kostum)” bertujuan untuk merancang dan membangun sistem informasi berbasis *website* guna mengotomasi proses penyewaan kostum yang sebelumnya masih dilakukan secara manual. Sistem ini dikembangkan menggunakan metode *prototype* dengan pendekatan iteratif yang meliputi tahapan komunikasi, perancangan cepat, pembuatan *prototype*, serta evaluasi dan perbaikan berdasarkan umpan balik pengguna. Pengumpulan data dilakukan melalui wawancara dengan pemilik usaha untuk mengetahui kebutuhan sistem, kemudian dilakukan pemodelan menggunakan *use case diagram*, *activity diagram*, dan *sequence diagram*. Sistem yang dihasilkan memiliki fitur utama seperti katalog dan pemesanan kostum *online*, pemesanan pembuatan kostum, manajemen stok dan pesanan oleh *admin*, integrasi pembayaran, halaman status pesanan, serta laporan keuangan. Hasil penelitian menunjukkan bahwa sistem yang dibangun mampu mengatasi permasalahan proses manual, meningkatkan efisiensi operasional, akurasi data, serta kualitas pelayanan, yang dibuktikan melalui pengujian *black box* dimana seluruh fungsi sistem berjalan dengan baik sesuai kebutuhan pengguna..[4]

Penelitian yang telah dilakukan oleh Mubarok et al. (2025), dengan judul “Sistem Informasi Penyewaan Kebaya, Jasa *Make Up*, Dekorasi, Fotografer Berbasis *Website*”. Penelitian ini bertujuan untuk mengembangkan sebuah sistem informasi berbasis *website* yang mampu mempermudah proses pencarian informasi, pemesanan, serta pengelolaan penyewaan kebaya dan layanan pendukung lainnya seperti jasa *make up*, dekorasi, dan fotografer secara *online* tanpa harus datang langsung ke tempat penyedia jasa. Sistem ini dibangun menggunakan bahasa pemrograman *Hypertext Preprocessor* (PHP) dengan dukungan basis data *Structured Query Language* (MySQL) serta dirancang menggunakan metode pengembangan sistem *waterfall* yang meliputi tahapan analisis kebutuhan, desain sistem, implementasi, pengujian, hingga pemeliharaan. Dengan adanya sistem ini, pelanggan dapat dengan mudah melihat katalog produk, melakukan pemesanan, serta mengetahui ketersediaan layanan secara *real-time*, sedangkan pemilik usaha dapat lebih efektif dalam mengelola data inventaris, transaksi, serta memantau permintaan pelanggan. Selain itu, sistem ini juga dirancang dengan tampilan antarmuka yang *user-friendly* sehingga memberikan kenyamanan bagi pengguna. Hasil dari penelitian ini menunjukkan bahwa sistem yang dikembangkan telah berjalan dengan baik dan sesuai dengan kebutuhan pengguna, dibuktikan melalui pengujian *black box* yang memperoleh tingkat validitas mencapai 100%, sehingga sistem ini dapat menjadi solusi yang efektif dalam meningkatkan efisiensi operasional serta kualitas pelayanan pada bisnis penyewaan kebaya berbasis digital.[5]

Penelitian yang telah dilakukan oleh Prasetyo dan Prasetya (2024) dengan judul “Sistem Informasi Penyewaan Baju pada Vendor Shasa Organizer Berbasis Website” bertujuan untuk merancang sistem informasi berbasis *website* guna mengatasi proses penyewaan yang masih dilakukan secara manual. Sistem ini dikembangkan menggunakan metode *waterfall* dengan tahapan identifikasi masalah, analisis kebutuhan, perancangan, hingga implementasi. Dalam pengembangannya, sistem menggunakan teknologi berbasis *website* seperti bahasa pemrograman PHP, *database* MySQL, serta tools pendukung XAMPP (*Cross-Platform, Apache, MySQL, PHP, Perl*). Hasil penelitian menunjukkan bahwa sistem mampu menyediakan fitur pengelolaan data baju, pemesanan, pembayaran DP dan pelunasan, pengiriman oleh kurir, serta pembuatan laporan secara otomatis, sehingga dapat meningkatkan efisiensi pengelolaan data, meminimalisir kesalahan pencatatan, dan mempermudah pelanggan dalam melakukan penyewaan tanpa harus datang langsung ke lokasi.[6]

3.2 Landasan Teori

Adapun landasan teori pada “Sistem Informasi Penyewaan Kostum dan Kebaya Luwunragi Berbasis *Website*” berpacuan dengan berbagai penelitian terdahulu di antaranya :

2.2.1 Waterfall

Metode penelitian *Waterfall* adalah konsep pengembangan sistem yang menekankan langkah-langkah kerja secara sistematis dan berurutan, dimulai dari identifikasi kebutuhan hingga proses pemeliharaan sistem. Setiap tahap harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya.

2.2.2 Penyewaan

Penyewaan adalah proses, cara, atau perbuatan menyewa suatu barang atau tempat dengan memberikan sejumlah pembayaran dalam jangka waktu tertentu. Penyewaan terjadi ketika seseorang (penyewa) menggunakan barang milik pihak lain (pemilik) berdasarkan kesepakatan harga dan durasi pemakaian. Definisi ini menunjukkan bahwa penyewaan melibatkan transaksi antara dua pihak dengan ketentuan pembayaran dan batas waktu yang jelas.[7]

2.2.3 Website

Website, atau sering disebut situs *web* atau *portal*, merupakan sekumpulan halaman web yang saling terhubung satu sama lain. Halaman utama dari sebuah *website* disebut *home page*, sedangkan halaman lainnya dikenal sebagai *web page*. Secara sederhana,

website adalah sebuah situs yang dapat diakses dan ditampilkan oleh pengguna internet dari berbagai belahan dunia. [8]

2.2.4 Laravel



Gambar 2. 1 Logo Laravel

Laravel adalah sebuah *framework* (kerangka kerja) berbasis PHP yang bersifat gratis (*open-source*). *Framework* ini dirancang untuk mempermudah pembuatan aplikasi web agar menghasilkan perangkat lunak yang berkualitas tinggi. Dengan penulisan kode (*sintaks*) yang ringkas dan mudah dipahami, Laravel membuat proses pengembangan sistem menjadi lebih cepat, efisien, serta dapat menghemat biaya pembuatan maupun perawatannya. [9]

2.2.5 Visual Studio Code



Gambar 2. 2 Logo Visual Studio Code

Visual Studio Code (VS Code) merupakan editor teks lintas platform yang kuat dan serbaguna, dikembangkan Microsoft. Selain dapat digunakan pada Windows, VS Code juga tersedia untuk sistem operasi Linux dan macOS. Editor ini mendukung berbagai bahasa pemrograman, termasuk *JavaScript*, *TypeScript*, dan *Node.js*. [10]

2.2.6 Bootstrap



Gambar 2. 3 Logo Bootstrap

Bootstrap adalah sebuah *framework* (kerangka kerja) gratis yang digunakan untuk membuat tampilan *website* agar terlihat rapi dan pas di berbagai layar, terutama ponsel (*mobile-first*). *Framework* ini menyediakan berbagai komponen desain siap pakai, sehingga proses pengerjaan *website* bisa jadi jauh lebih cepat dan mudah.

2.2.7 HTML



Gambar 2. 4 Logo HTML

HTML atau *Hypertext Markup Language* merupakan bahasa *markup* yang digunakan untuk membangun struktur halaman pada sebuah *website*. Melalui HTML, halaman *website* dapat disusun menjadi tampilan yang menarik, saling terhubung antarhalaman, serta dapat diakses oleh pengguna melalui jaringan internet.[11]

2.2.8 JavaScript



Gambar 2. 5 Logo JavaScript

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat *website* menjadi lebih dinamis dan interaktif. Dengan JavaScript, konten di *website* bisa berubah, merespon aksi pengguna, dan memperbaharui tampilan tanpa perlu memuat ulang seluruh halaman.[12]

2.2.9 Cascading Style Sheets



Gambar 2. 6 Logo Cascading Style Sheets

CSS (*Cascading Style Sheets*) merupakan aturan yang dipakai untuk mengatur gaya dan desain tampilan *website*. Fungsi utamanya adalah menentukan warna, ukuran teks, dan tata letak halaman agar terlihat menarik dan pas di layar pengguna. Kode CSS biasanya dipisah dari HTML supaya desain *website* lebih mudah diperbarui dan dirawat.[13]

2.2.10 Laragon



Gambar 2. 7 Logo Laragon

Laragon adalah aplikasi gratis dan *open-source* yang digunakan untuk membuat *server* lokal di komputer berbasis Windows. Aplikasi ini sudah dilengkapi dengan berbagai komponen penting seperti Apache/Nginx, PHP, dan MySQL sehingga memudahkan proses pembuatan, pengujian, hingga pengelolaan *website* secara lebih cepat dan praktis. [14]

2.2.11 Database

Database (basis data) adalah sistem yang digunakan untuk menyimpan dan mengelola sekumpulan data secara teratur. Setiap *database* menyediakan *API* (*Application Programming Interface*) atau antarmuka pemrograman yang memudahkan pengguna untuk membuat, mengambil, mengatur, mencari, hingga mencadangkan data. Di dalamnya, data disusun dalam *file* yang saling terhubung agar bisa digunakan oleh berbagai aplikasi dengan mudah.[15]

2.2.12 UML (Unified Modeling Language)

UML (*Unified Modeling Language*) merupakan bahasa visual standar yang digunakan untuk merancang dan menggambarkan struktur sistem dalam pengembangan perangkat lunak berorientasi objek. Melalui diagram dan notasi standar, UML memudahkan

pengembang, analis, dan tim proyek dalam memvisualisasikan alur sistem, meminimalkan kesalahpahaman, serta menjadi acuan utama pada tahap implementasi dan dokumentasi. [16]

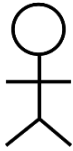
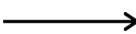
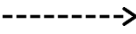
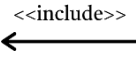
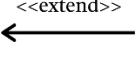
Dalam pendekatan *Object-Oriented Analysis and Design* (OOAD), setiap komponen sistem dianggap sebagai objek yang memiliki atribut dan perilakunya masing-masing. UML digunakan untuk memodelkan hubungan, struktur, dan interaksi antarobjek tersebut secara visual agar sistem lebih mudah dianalisis dan dikembangkan. [17]

Perancangan berorientasi objek berbasis UML pada penelitian ini menggunakan alat bantu (diagram) sebagai berikut:

a. Use Case Diagram

Use Case Diagram merupakan alat untuk memodelkan perilaku dan fungsionalitas yang diinginkan dari suatu sistem informasi. Diagram ini menekankan pada “apa” yang dilakukan sistem, bukan pada cara pelaksanaannya. Setiap *use case* menggambarkan interaksi antara aktor dengan sistem, misalnya proses login, pembuatan daftar belanja, dan aktivitas lainnya.[18]

Tabel 2. 1 Use Case Diagram

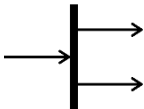
No	Gambar	Simbol	Keterangan
1.		<i>Actor</i>	Mewakili peran seorang sistem yang lain, atau alat ketika berkomunikasi dengan <i>Use Case</i> .
2.		<i>Use Case</i>	Mewakili Abstraksi dan interaksi antara sebuah sistem dan <i>Actor</i> .
3.		<i>Association</i>	Abstraksi penghubung antar <i>Actor</i> dengan <i>Use Case</i> .
4.		<i>Generalization</i>	Menunjukkan spesialisasi <i>Actor</i> yang terlibat <i>Use Case</i>
5.		<i>Include</i>	Menunjukkan bahwa suatu <i>Use Case</i> seluruhnya merupakan fungsionalitas dari <i>Use Case</i> lainnya.
6.		<i>Extend</i>	Menunjukkan bahwa suatu <i>Use Case</i> merupakan tambahan fungsional dari <i>Use Case</i> lainnya jika suatu kondisi terpenuhi.

b. Activity Diagram

Activity diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana alur tersebut berakhir. Berikut adalah simbol-simbol

dari *activity diagram*: [19]

Tabel 2. 2 Activity Diagram

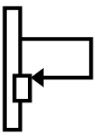
No	Gambar	Simbol	Keterangan
1.		<i>Initial node</i>	Merupakan awal aktivitas yang akan dibuat.
2.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi.
3.		<i>Decicion</i>	Menggambarkan suatu keputusan atau tindakan yang harus diambil pada kondisi tertentu.
4.		<i>Fork Join node</i>	Digunakan untuk menunjukan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
5.		<i>Fork node</i>	Satu aliran pada tahap yang berubah menjadi beberapa aliran.
6.		<i>Activity Final node</i>	Merupakan akhir aktivitas yang telah dibuat.

c. Sequence Diagram

Sequence diagram adalah diagram yang menggambarkan interaksi dan urutan pertukaran pesan antarobjek berdasarkan rentang waktu. Dimensi horizontal menunjukkan objek-objek yang terlibat (disusun dari kiri ke kanan), sedangkan dimensi

vertikal menunjukkan garis waktu (*lifeline*). Pesan atau perintah digambarkan sebagai garis panah yang menghubungkan satu *lifeline* dengan *lifeline* lainnya.[20]

Tabel 2. 3 Sequence Diagram

No	Gambar	Simbol	Keterangan
1.		<i>Entity Class</i>	Menggambarkan awal dari sebuah sistem, menjadi dasar penyusunan basisdata.
2.		<i>Boundary Class</i>	Menggambarkan komunikasi antara sistem seperti <i>Form</i> .
3.		<i>Control Class</i>	Penghubung antara <i>boundary</i> dengan <i>table</i> .
4.		<i>Recursive</i>	Mengirimkan pesan yang akan dikirim ke dirinya sendiri.
5.		<i>Activation</i>	Menghubungkan proses durasi aktivasi sebuah proses.
6.		<i>Life Line</i>	Menghubungkan tiap-tiap objek yang mana terdapat <i>activation</i> .
7.		<i>Message</i>	Sebagai pengirim pesan ke tiap-tiap <i>Class</i> .

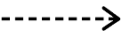
d. Class Diagram

Class diagram adalah salah satu diagram struktur dalam UML yang berfungsi untuk menampilkan detail susunan sistem,

meliputi *class* (kelas), atribut, *method* (metode), serta hubungan antarobjek. Dalam perancangannya, diagram ini umumnya terbagi menjadi dua bagian utama: bagian pertama menggambarkan struktur *database* (basis data), sedangkan bagian kedua menggambarkan komponen utama sistem yang terdiri dari *class interface*, *class control*, dan *class entity*..[21]

Tabel 2. 4 Class Diagram

No	Gambar	Simbol	Keterangan
1.		<i>Class</i>	Menggambarkan kumpulan objek-objek dengan atribut dan operasi yang sama.
2.		<i>Nary Association</i>	Sebagai upaya menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Collaboration</i>	Urutan aksi-aksi yang ditampilkan oleh sistem yang menghasilkan suatu hasil yang tertukar bagi suatu <i>actor</i> .
4.		<i>Generalization</i>	Menggambarkan dimana hubungan objek <i>descendent</i> berperilaku dan struktur data objek yang di atasnya objek induk <i>ancestor</i> .

No	Gambar	Simbol	Keterangan
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri akan mempengaruhi elemen yang tidak mandiri.