

**PENGEMBANGAN SISTEM INFORMASI PENYEWAAN KOSTUM
COSPLAY KARAKTER ANIME BERBASIS WEBSITE DI LAMPUNG**

Oleh

Muhammad Ishann Andhika

2207051002



**PROGRAM STUDI MANAJEMEN INFORMATIKA
JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG**

2025

ABSTRAK

PENGEMBANGAN SISTEM INFORMASI PENYEWAAN KOSTUM COSPLAY KARAKTER ANIME BERBASIS WEBSITE DI LAMPUNG

Oleh

Muhammad Ishann Andhika

Perkembangan budaya pop Jepang meningkatkan popularitas *cosplay* di Lampung. Keterbatasan akses kostum *cosplay* berkualitas dengan harga terjangkau menjadi kendala bagi para penggemar *anime*. Sistem penyewaan konvensional yang manual menyebabkan ketidakefisiensi pengelolaan inventori dan layanan pelanggan. Penelitian ini bertujuan mengembangkan sistem informasi penyewaan kostum *cosplay* berbasis website untuk memudahkan akses dan pengelolaan bisnis di Lampung. Penelitian menggunakan metode *System Development Life Cycle* (SDLC) model *Waterfall* meliputi analisis kebutuhan, desain, implementasi, testing, dan maintenance. Sistem dikembangkan menggunakan *framework PHPNative*, database *MySQL*, serta *HTML*, *CSS*, dan *JavaScript* untuk menciptakan aplikasi *web* responsif. Sistem memiliki fitur katalog kostum dengan pencarian dan filter, manajemen pemesanan dan pembayaran *online*, manajemen admin, dan review pelanggan. Evaluasi dilakukan melalui pengujian fungsionalitas, usability testing, dan *User Acceptance Testing* (UAT). Hasil pengujian menunjukkan sistem berfungsi baik dan memenuhi kebutuhan pengguna. Tingkat kepuasan UAT mencapai 90% dari 23 responden, menunjukkan sistem layak digunakan. Sistem berhasil meningkatkan efisiensi pengelolaan bisnis dan kemudahan akses pelanggan. Pengembangan ini diharapkan mendukung pertumbuhan industri kreatif dan ekonomi digital di Lampung

Kata Kunci: Sistem Informasi, Penyewaan Kostum, *Cosplay*, *Anime*, *Website*, *E-commerce*, Lampung

ABSTRACT

DEVELOPMENT OF A WEBSITE-BASED ANIME COSPLAY COSTUME RENTAL INFORMATION SYSTEM IN LAMPUNG

By

Muhammad Ishann Andhika

The development of Japanese pop culture has increased the popularity of cosplay in Lampung. Limited access to good quality cosplay costumes at affordable prices is an obstacle for anime fans. Conventional manual rental systems cause inefficiencies in inventory management and customer service. This research aims to develop a website-based cosplay costume rental information system to facilitate access and business management in Lampung. The research uses the System Development Life Cycle (SDLC) Waterfall model method, including needs analysis, design, implementation, testing, and maintenance. The system was developed using the PHPNative framework, MySQL database, and HTML, CSS, and JavaScript to create a responsive web application. The system features a costume catalog with search and filter, online order and payment management, admin management, and customer reviews. The evaluation was conducted through functional testing, usability testing, and User Acceptance Testing (UAT). Test results showed the system functioned well and met user needs. The UAT satisfaction rate reached 90% among 23 respondents, indicating the system's suitability for use. The system successfully improved business management efficiency and eased customer access. This development is expected to support the growth of the creative industry and digital economy in Lampung.

Keyword : *Information System, Costume Rent, Cosplay, Anime, Website, E-Commerce, Lampung*

**PENGEMBANGAN SISTEM INFORMASI PENYEWAAN KOSTUM
COSPLAY KARAKTER ANIME BERBASIS WEBSITE DI LAMPUNG**

Oleh

Muhammad Ishann Andhika

(Tugas Akhir)

**Sebagai Salah Satu Syarat Untuk Mencapai Gelar
AHLI MADYA (A.Md)**

Pada

**Program Studi D3 Manajemen Informatika
Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2025**

LEMBAR PENGESAHAN

Judul Tugas Akhir : Pengembangan Sistem Informasi Penyewaan Kostum
Cosplay Karakter Anime Berbasis Website Di Lampung

Nama Mahasiswa : Muhammad Ishann Andhika

NPM : 2207051002

Program Studi : D3 Manajemen Informatika

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam

Pembimbing Utama

Igit Sabda Ilman, S.Kom., M.Cs.
NIK. 232111960101101



Pembimbing Kedua

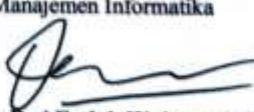
Yhoga Ismail Ibn Ibrahim, M.T.I
NIP. 199604032024061001

2. Mengetahui

Ketua Jurusan Ilmu Komputer


Dwi Sakethi, S.Si., M.Kom.
NIP. 19680611 199802 1 001

Ketua Program Studi
D3 Manajemen Informatika


Ossy Dwi Endah Wulansari, S.Si., M.T
NIP. 19740713 200312 2 002

MENGESAHKAN

1. Tim Pengaji

Pembimbing Utama : Igit Sabda Ilman, S.Kom., M.Cs.

Pembimbing Kedua : Yhoga Ismail Ibn Ibrahim, M.T.I

Pengaji / Pembahas : Rizky Prabowo, S.Kom., M.Kom.

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Bapak Heri Satria, S.Si., M.Si.
NIP. 1971001 200501 1 002

Tanggal Lulus Ujian Tugas Akhir : **8 Agustus 2025**

**PERNYATAAN MENGENAI TUGAS AKHIR
DAN SUMBER INFORMASI**

Saya yang bertanda tangan dibawah ini menyatakan bahwa tugas akhir saya yang berjudul "**Pengembangan Sistem Informasi Penyewaan Kostum Cosplay Karakter Anime Berbasis Website Di Lampung**" merupakan karya saya sendiri sesuai dengan arahan komisi pembimbing dan belum diajukan dalam bentuk apapun kepada Perguruan Tinggi manapun. Sumber Informasi yang berasal atau dikutip dari karya yang telah diterbitkan oleh penulis lain telah disebutkan dalam teks dan dicantumkan dalam "Daftar Pustaka" dibagian akhir Tugas Akhir ini.

Demikian surat pernyataan ini, Saya buat dengan sebenar-benarnya dan tanpa paksaan dari siapapun untuk digunakan sebagaimana semestinya.

Bandar Lampung, 16 September 2025



Hak Cipta Milik UNILA, Tahun 2025

Hak Cipta dilindungi Undang – Undang

Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan atau menyebutkan sumbernya. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik, atau tinjauan suatu masalah dan pengutipan tersebut tidak merugikan kepentingan yang wajar.

Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis dalam bentuk apapun tanpa izin UNILA.

RIWAYAT HIDUP



Penulis memiliki nama lengkap Muhammad Ishann Andhika, lahir di Kota Bandar Lampung pada tanggal 2 Mei 2004. Penulis merupakan anak bungsu dari Bapak Sujana dan Ibu Baharun Nisah. Penulis merupakan anak ke-5 dari 5 bersaudara. Penulis memiliki empat orang saudara kandung, yaitu Chandra Andhika, Rully Andhika, Indri Windha Prasasty, dan Sunita Agustina. Pendidikan formal penulis dimulai di SDN 1 Sidodadi dan lulus pada tahun 2016. Selanjutnya, penulis melanjutkan pendidikan di SMPN 20 Bandar Lampung dan menyelesaiannya pada Tahun 2019. Pendidikan menengah atas ditempuh di SMKN 2 Bandar Lampung dengan Jurusan yaitu Rekayasa Perangkat Lunak (RPL) dan diselesaikan pada Tahun 2022. Pada tahun yang sama, penulis diterima sebagai mahasiswa Program Studi D3 Manajemen Informatika Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

Selama menjalani masa perkuliahan penulis aktif mengikuti berbagai kegiatan diantaranya:

1. Mengikuti rangkaian kegiatan Pengenalan Kehidupan Kampus bagi Mahasiswa Baru (PKKMB) Universitas Lampung, PKKMB Fakultas Matematika dan Ilmu Pengetahuan Alam dan Program Orientasi Jurusan Ilmu Komputer (PRINTER) pada Tahun 2022.
2. Menjadi anggota Himpunan Mahasiswa Ilmu Komputer (HIMAKOM) Bidang Medinfo pada Tahun 2023.
3. Menjadi panitia Pekan Raya Jurusan dan Himakom Tournament (PRJ x HT) Ke – XI pada Tahun 2023.

4. Menjadi panitia Program Orientasi Jurusan Ilmu Komputer (PRINTER) pada Tahun 2023.
5. Menjadi panitia Mipa Olympic BEM FMIPA Universitas Lampung Tahun 2023.
6. Menjalani peran sebagai Asisten Dosen pada mata kuliah Aplikasi Pengolah Kata, Presentasi dan Multimedia di Program Studi D3 Manajemen Informatika, Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung pada tahun 2023.
7. Menjalani peran sebagai Asisten Dosen pada mata kuliah Animasi Dasar di Program Studi D3 Manajemen Informatika Jurusan Ilmu Komputer,Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung pada Tahun 2024.
8. Melaksanakan kegiatan Karya Wisata Ilmiah ke - XXXIII FMIPA Universitas Lampung pada Tahun 2022.
9. Melaksanakan Kegiatan Praktik Kerja Lapangan (PKL) di PT. Acosys Global Data pada Tahun 2024.

MOTTO

“Sesungguhnya bersama kesulitan ada kemudahan.”

(Q.S Al-Insyirah:5)

“Allah tidak akan membebani seseorang melainkan sesuai kesanggupannya”

(Q.S. Al-Baqarah: 286)

“Yang terpenting bukan cepat, tapi terus. Dalam setiap barisan kode dan ide yang sederhana, ada perjalanan panjang untuk menjadi lebih baik. Semua hal besar berawal dari niat kecil yang dikerjakan dengan konsisten.”

(Hindia Baskara Putra)

“ Terlambat bukan berarti gagal, cepat bukan berarti hebat. Terlambat bukan menjadi alasan untuk menyerah, setiap orang memiliki proses yang berbeda. Percaya Proses itu yang paling penting. Karena allah telah mempersiapkan hal baik dibalik kata proses yang dianggap rumit.”

(Edwar Satria)

PERSEMBAHAN

Alhamdulillahirabbilalamin

Puji dan syukur saya panjatkan kepada Allah Subhanahu Wa Ta'ala atas segala rahmat, karunia, dan hidayah-Nya, sehingga saya dapat menyelesaikan Tugas Akhir ini. Shalawat serta salam senantiasa tercurahkan kepada junjungan Nabi besar Muhammad SAW, beserta keluarga, sahabat, dan para pengikutnya hingga akhir zaman.

Dengan penuh rasa syukur, karya tulis ini penulis persembahkan untuk:

Kedua Orang Tua Tercinta

Yang selalu memberikan dukungan, doa, semangat, dan pengorbanan tanpa henti.
Terima kasih atas segalanya.

Kakak dan Keluarga Tercinta

Yang selalu mendukung dan mendoakan dalam setiap langkah.

Seluruh Keluarga Besar Ilmu Komputer 2022

Yang telah menjadi bagian dari perjalanan ini dengan segala tawa, kerja sama, dan dukungan moral yang tak ternilai.

Almamater Tercinta, Universitas Lampung dan Jurusan Ilmu Komputer

Tempat di mana penulis tumbuh dan belajar sebagai bekal hidup untuk masa depan.

KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat, taufik dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini dengan judul “PENGEMBANGAN SISTEM INFORMASI PENYEWAAN KOSTUM COSPLAY KARAKTER ANIME BERBASIS WEBSITE DI LAMPUNG” sebagai syarat untuk menyelesaikan studi pada Program Studi D3 Manajemen Informatika Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

Shalawat serta salam semoga senantiasa tercurah kepada Nabi Muhammad SAW,beserta Keluarga, sahabar dan seluruh umatnya yang istiqamah di jalan-Nya hingga akhir zaman.

Penulisan laporan Tugas Akhir ini tidak lepas dari bantuan, bimbingan dan dukungan dari berbagai pihak. Oleh karena itu, penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan rahmat, karunia, serta hidayah-Nya selama proses penyelesaian sistem hingga penyusunan laporan ini.
2. Kedua orang tua, dan kakak-kakak saya yang selalu memberikan doa dan dukungan kepada saya tanpa henti.
3. Bapak Dr. Eng. Heri Satria, S.Si., M.Si. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam.
4. Bapak Dwi Sakethi, S.Si., M.Kom. selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
5. Ibu Ossy Dwi Endah Wulansari, S.Si., M.Si. selaku Ketua Program Studi D3 Manajemen Informatika.

6. Bapak Igit Sabda Ilman, S.Kom., M.Cs. selaku Dosen Pembimbing Utama Tugas Akhir, yang telah memberikan arahan, ide, motivasi, serta kritik dan saran yang sangat berharga, sehingga penulis dapat menyelesaikan tugas akhir ini.
7. Bapak Yhogha Ismail Ibn Ibrahim, M.T.I selaku Dosen Pembimbing kedua, yang telah memberikan motivasi dan arahan yang sangat bermanfaat daam penyempurnaan penulisan pada tugas akhir ini.
8. Bapak Rizky Prabowo, S.Kom., M.Kom. selaku Dosen Pembahas, yang telah memberikan masukan, saran, dan motivasi yang bermanfaat untuk penyempurnaan tugas akhir ini.
9. Ibu Anie Rose Irawati, S.T., M.Cs. selaku Dosen Pembimbing Akademik, yang memberikan arahan yang bermanfaar selama menjalanin proses studi.
10. Seluruh Bapak dan Ibu Dosen Prodi D3 Manajemen Informatika Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah membagikan ilmu pengetahuan yang berharga kepada penulis.
11. PT.Acosys Global Data, sebagai tempat pelaksanaan kerja praktik penulis,yang telah memberikan kesempatan, bimbingan, serta pengalaman berharga dalam dunia kerja nyata, khususnya dalam pengembangan sistem informasi dan pemahaman proses bisnis industry.
12. Rekan-Rekan D3 Manajemen Informatika yang telah membantu dalam mengerjakan tugas akhir dan telah berjuang bersama dari awal hingga akhir.

Penulis menyadari bahwa laporan ini masih memiliki kekurangan. Meskipun demikian, besar harapan penulis agar karya ini dapat memberikan manfaat serta menjadi referensi yang bermanfaat bagi civitas akademika Jurusan Ilmu Komputer.

Bandar Lampung,8 Agustus 2025

Muhammad Ishann Andhika
NPM. 2207051002

DAFTAR ISI

DAFTAR ISI.....	xvi
DAFTAR TABEL	xviii
DAFTAR GAMBAR.....	xxix
I. PENDAHULUAN	i
1.1 Latar Belakang	2
1.2 Rumusan Masalah	2
1.3 Manfaat	2
1.4 Tujuan Penelitian	3
II. LANDASAN TEORI.....	4
2.1 Konsep Dasar Sistem Informasi	4
2.2 Teori <i>Webserver</i>	5
2.3 Konsep Penyewaan.....	6
2.4 Teknologi yang Digunakan	6
2.4.1 Bahasa Pemrograman.....	7
2.4.2 <i>Framework Web</i>	7
2.4.3 Database.....	8
2.4.4 Mailtrap	9
2.4.5 PHPMailer	10
2.4.6 <i>Sofware Development Life Cycle (SDLC)</i>	10
2.4.7 Metode <i>Waterfall</i>	12
2.4.8 Black Box Testing.....	14
2.5 <i>Unified Modeling Language (UML)</i>	15

2.6	<i>Use Case Diagram</i>	16
2.7	<i>Activity Diagram</i>	17
III.	ANALISIS DAN PERANCANGAN SISTEM	19
3.1	Analisis Kebutuhan	19
3.1.1	Kebutuhan Fungsional	19
3.1.2	Kebutuhan Non Fungsional	19
3.2	Desain Sistem	20
3.2.1	Use Case Diagram	20
3.2.2	<i>Use Case Description</i>	29
3.2.3	<i>Entity Relationship Diagram (ERD)</i>	39
3.2.4	Desain Interface	41
3.2.5	Cutover	51
V.	KESIMPULAN DAN SARAN	57
5.1	Pengujian Sistem	57
5.2	Saran	58
	DAFTAR PUSTAKA	59

DAFTAR TABEL

Tabel 1. Contoh Penerapan Sistem Informasi Penyewaan Kostum <i>Cosplay</i> Karakter Anime Berbasis <i>Website</i> Di Bandar Lampung.....	14
Tabel 2. <i>Use Case Description</i> untuk <i>Register</i>	29
Tabel 3. <i>Use Case Description</i> untuk <i>Login</i>	30
Tabel 4. <i>Use Case Description</i> untuk Verifikasi KTP	31
Tabel 5. <i>Use Case Description</i> untuk Verifikasi Pembayaran.....	32
Tabel 6. <i>Use Case Description</i> untuk Penelusuran Kostum	33
Tabel 7. <i>Use Case Description</i> untuk Melakukan Pemesanan	33
Tabel 8. <i>Use Case Description</i> untuk Komunikasi dengan Penyewa	34
Tabel 9. <i>Use Case Description</i> untuk Rating dan Ulasan.....	35
Tabel 10. <i>Use Case Description</i> untuk Manajemen Katalog.....	36
Tabel 11. <i>Use Case Description</i> untuk Manajemen Pesanan	37
Tabel 12. <i>Use Case Description</i> untuk Komunikasi dengan Pelanggan.....	38
Tabel 13. Skenario pengujian Autentikasi	51
Tabel 14. Skenario Pengujian Fitur Tambah Kostum.....	52
Tabel 15. Skenario Pengujian Fitur Pemesanan Kostum	53
Tabel 16. Skenario Pengujian <i>Wishlist</i>	53
Tabel 17. Skenario Pengujian Rating dan Ulasan	53
Tabel 18. Skenario Pengujian <i>Chat</i>	53
Tabel 19. Bobot Skor Skala Likert.....	54
Tabel 20. Kriteria Indeks Kepuasan Pengguna	55

DAFTAR GAMBAR

Gambar 1. <i>Waterfall Model</i> (Model Air Terjun) dalam pengembangan perangkat	12
Gambar 2. <i>Use Case Diagram</i> Sistem Informasi Penyewaan Kostum <i>Cosplay</i>	21
Gambar 3. <i>Activity Diagram Login</i>	22
Gambar 4. <i>Activity Diagram Register</i>	22
Gambar 5. <i>Activity Diagram</i> Penelusuran Kostum	23
Gambar 6. <i>Activity Diagram</i> dari rating dan ulasan.....	24
Gambar 7. <i>Activity Diagram</i> Melakukan Pemesanan	24
Gambar 8. <i>Activity Diagram</i> Manajemen Katalog.....	25
Gambar 9. <i>Activity Diagram</i> Manajemen Pesanan	26
Gambar 10. <i>Activity Diagram</i> Verifikasi KTP.....	26
Gambar 11. <i>Activity Diagram</i> Verifikasi Pembayaran	27
Gambar 12. <i>Activity Diagram</i> Komunikasi Pelanggan	28
Gambar 13. <i>Activity Diagram</i> Komunikasi dengan Penyewa.....	28
Gambar 14. <i>Entity Relationship Diagram</i> (ERD) Sistem Informasi Penyewaan Kostum <i>Cosplay</i>	39
Gambar 15. <i>Desain</i> Halaman Login.....	42
Gambar 16. <i>Desain</i> Halaman <i>Register</i>	43
Gambar 17. Rancangan Desain Halaman Awal	44
Gambar 18. Rancangan Desain Lupa <i>Password</i>	45
Gambar 19. Rancangan Desain <i>Reset Password</i>	45
Gambar 20. Rancangan Desain Katalog	46
Gambar 21. Rancangan Desain Detail	47
Gambar 22. Rancangan Desain <i>Order</i>	47
Gambar 23. Rancangan Desain <i>Dashboard Admin</i>	48
Gambar 24. Rancangan Desain <i>Dashboard Pelanggan</i>	49

Gambar 25. Rancangan Desain <i>Dashboard</i> Penyewa.....	50
Gambar 26. Rancangan Desain Edit Profil	51

I. PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi dan kecanggihan internet saat ini telah mendorong berbagai sektor untuk bertransformasi ke arah digital, termasuk dalam bidang penyewaan kostum *cosplay*. *Cosplay* telah menjadi bagian penting dalam budaya *pop*, terutama di kalangan penggemar anime, manga, dan video game (Paramita et al, 2025). Kostum *cosplay* memungkinkan individu untuk mengekspresikan kecintaan mereka terhadap karakter favorit dalam berbagai acara seperti konvensi, festival, dan kompetisi *cosplay*. Namun, tidak semua orang memiliki kemampuan atau keinginan untuk membeli kostum *cosplay* yang sesuai dengan karakter yang mereka pilih. Layanan penyewaan kostum *cosplay* muncul sebagai solusi yang praktis dan ekonomis bagi para penggemar *cosplay*.

Hal ini terlihat dari menjamurnya usaha rental kostum seperti Rental *Cosplay* Lombok yang berdiri sejak tahun 2022, menyediakan kostum dengan harga sewa mulai dari Rp35.000 hingga Rp110.000 per tiga hari, sebagai alternatif yang lebih terjangkau dibandingkan membeli kostum baru (NTB Satu, 2022). Di kota besar seperti Surabaya dan Jakarta, pemilik usaha rental dengan koleksi lebih dari 100 set kostum mengungkapkan bahwa banyak *cosplayer* lebih memilih menyewa karena harga sewa yang berkisar antara Rp100.000–Rp250.000 untuk durasi tiga hari, jauh lebih hemat dibandingkan membeli atau menjahit sendiri (Tempo, 2023). Sayangnya, penyedia layanan penyewaan kostum umumnya masih mengandalkan proses manual dalam menjalankan operasionalnya, seperti pencatatan pesanan, pengelolaan stok kostum, dan konfirmasi transaksi. Ketergantungan pada metode manual ini sering kali menimbulkan kendala seperti human *error*, keterlambatan informasi, duplikasi data, dan kesulitan dalam memantau status pemesanan atau pengembalian secara real-time. Akibatnya, baik pelanggan maupun penyedia jasa mengalami penurunan efisiensi layanan

pelanggan merasa tidak mendapatkan pelayanan yang cepat dan akurat, sementara penyedia mengalami kesulitan dalam menjaga konsistensi data dan kualitas operasional (Putri, 2024). Mengadopsi sistem informasi berbasis *website*, penyedia layanan penyewaan kostum *cosplay* karakter *anime* dapat meningkatkan efisiensi dan kenyamanan dalam operasionalnya. Sistem ini tidak hanya mempermudah pelanggan dalam mencari dan memesan kostum *cosplay* sesuai karakter yang diinginkan, tetapi juga memungkinkan penyedia layanan untuk mengelola inventaris kostum secara real-time (Nauli, 2020). *Website* ini dirancang dengan tampilan yang responsif, memastikan akses optimal baik di perangkat desktop maupun handphone, sehingga pelanggan dapat dengan mudah melakukan pemesanan kapan saja dan di mana saja. Pengembangan sistem informasi penyewaan kostum *cosplay* karakter *anime* berbasis *website* ini sangat diperlukan untuk memberikan layanan yang lebih baik, efisien, dan mengurangi kesalahan dalam pengelolaan penyewaan. Sistem ini diharapkan dapat meningkatkan kepuasan pelanggan serta memperkuat daya saing penyedia layanan di Bandar Lampung.

1.2 Rumusan Masalah

Berdasarkan pada latar belakang yang dikemukakan, maka dapat dirumuskan permasalahan, yaitu Bagaimana merancang sistem informasi berbasis *website* yang efektif dan efisien untuk mengelola proses penyewaan kostum *cosplay* karakter anime di Bandar Lampung

1.3 Manfaat

Manfaat dari tugas akhir ini adalah sebagai berikut :

- a. Bagi Penyewa: Mempermudah proses pencarian, pemesanan, dan pengembalian kostum, sehingga pengalaman penyewa menjadi lebih baik.
- b. Bagi Penyedia Jasa: Meningkatkan efisiensi dalam pengelolaan inventaris, meminimalkan kesalahan dalam pencatatan transaksi, dan meningkatkan produktivitas.
- c. Bagi Peneliti dan Akademisi: Menjadi referensi dalam pengembangan sistem

informasi di bidang penyewaan, serta membuka peluang untuk penelitian lebih lanjut.

- d. Bagi Masyarakat: Menyediakan akses yang lebih luas terhadap kostum yang berkualitas tanpa harus membelinya, serta mendukung kegiatan kreatif dan budaya lokal.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk:

- a. Mengembangkan sistem informasi yang terintegrasi untuk mendukung proses penyewaan kostum secara digital.
- b. Menyediakan fitur-fitur utama seperti pencarian kostum, pemesanan langsung pada sistem.
- c. Meningkatkan efisiensi operasional dalam melakukan pencatatan pesanan bagi penyewa, dan penyedia jasa dan pengalaman pengguna dalam penyewaan kostum.

II. LANDASAN TEORI

2.1 Konsep Dasar Sistem Informasi

Menurut buku "Konsep Dasar Sistem Informasi: Teori dan Implementasi" oleh Hanifah Nur Nasution, Thofik Hidayat, dan Sari Wahyuni Rozi Nasution (2025), sistem informasi terdiri dari lima komponen utama:

- a. *Hardware*: Komponen fisik seperti komputer, *server*, perangkat input/output, dan jaringan yang mendukung operasional sistem informasi. Dalam konteks penyewaan kostum, hardware mencakup perangkat yang digunakan oleh admin untuk mencatat transaksi, memeriksa ketersediaan stok kostum, dan mencetak bukti pemesanan. Selain itu, perangkat seperti kamera dan scanner dapat digunakan untuk verifikasi identitas pelanggan melalui sistem KYC.
- b. *Software*: Program atau aplikasi yang digunakan untuk memproses data, termasuk sistem operasi dan aplikasi khusus yang mendukung fungsi bisnis. Pada layanan penyewaan kostum, *software* dapat berupa sistem informasi berbasis *website* yang menyediakan fitur seperti pendaftaran akun pelanggan, katalog kostum, formulir pemesanan, notifikasi pengembalian, hingga laporan transaksi secara otomatis. Perangkat lunak ini dirancang agar dapat digunakan di berbagai perangkat, baik desktop maupun mobile, sehingga memudahkan pengguna dalam mengakses layanan kapan saja dan di mana saja.
- c. *People*: Pengguna sistem informasi, baik individu maupun organisasi, yang berinteraksi dengan sistem untuk menjalankan tugas-tugas tertentu. Dalam sistem penyewaan kostum, "*people*" mencakup pemilik usaha, admin sistem, petugas gudang, hingga pelanggan. Admin bertanggung jawab untuk mengelola data dan transaksi, sementara pelanggan memanfaatkan sistem untuk mencari dan memesan kostum.

Keberhasilan sistem informasi sangat bergantung pada kemampuan pengguna dalam memahami dan mengoperasikan sistem dengan baik, sehingga pelatihan dasar bagi admin dan tampilan antarmuka yang ramah pengguna menjadi sangat penting.

- d. *Processes*: Langkah-langkah atau aturan yang mengatur cara sistem informasi beroperasi, termasuk proses pengumpulan, pengolahan, penyimpanan, dan distribusi data. Dalam layanan penyewaan kostum, proses ini mencakup tahapan pemesanan *online* oleh pelanggan, pengecekan otomatis terhadap ketersediaan kostum, konfirmasi pembayaran, validasi identitas (KYC), pencatatan waktu sewa dan pengembalian, serta notifikasi kepada pelanggan. Proses yang dirancang secara otomatis dan terstruktur dapat mengurangi risiko kesalahan manual, meningkatkan efisiensi pelayanan, dan menciptakan pengalaman pengguna yang lebih baik.

Memahami kelima komponen ini secara menyeluruh dan menerapkannya secara terintegrasi dalam sistem informasi berbasis *website*, pengelolaan penyewaan kostum *cosplay* dapat dilakukan dengan lebih efektif, efisien, dan akurat.

2.2 Teori *Webserver*

Webserver adalah sistem perangkat lunak dan/atau perangkat keras yang dirancang untuk menangani, memproses, dan merespons permintaan klien (umumnya browser) melalui protokol komunikasi HTTP (*HyperText Transfer Protocol*) atau HTTPS (*HTTP Secure*). *Web server* berfungsi sebagai jembatan antara pengguna (*client*) dan aplikasi web atau situs web yang dihosting dalam suatu sistem *server*. Selain menyajikan konten statis seperti dokumen HTML, gambar, dan file CSS, *webserver* juga dapat mengeksekusi permintaan dinamis melalui bahasa *server-side* seperti *PHP*, *Python*, atau *Node.js*, dengan bantuan modul interpreter.

Menurut Nyabuto (2024), arsitektur *client-server* merupakan model perangkat lunak di mana sumber daya dan permintaan dilayani melalui jaringan, dengan *server* yang menerima permintaan, memprosesnya, dan merespons dengan tepat. *Server* berperan sebagai komputer kompleks dengan kekuatan pemrosesan tinggi

yang menjalankan program-program untuk menerima dan memproses permintaan klien. *Webserver* memiliki berbagai fungsi penting dalam ekosistem *web*, di antaranya:

1. Melayani permintaan klien dengan menyediakan file atau data yang diminta secara cepat dan aman.
2. Memproses skrip dinamis melalui modul seperti *mod_php* (*Apache*) atau *FastCGI*.
3. Mengelola sesi pengguna dan autentikasi akses terhadap halaman tertentu.
4. Menyediakan keamanan melalui implementasi *SSL/TLS* untuk pertukaran data terenkripsi.
5. Mencatat aktivitas sistem melalui log akses dan *error* yang digunakan untuk audit serta pemeliharaan.

2.3 Konsep Penyewaan

Penyewaan adalah suatu aktivitas yang melibatkan pemberian izin penggunaan barang oleh pemilik kepada penyewa untuk jangka waktu tertentu dengan imbalan berupa pembayaran tertentu (Mytaros, 2021). Dalam penyewaan kostum, terdapat proses utama, yaitu:

1. Pemilihan barang (kostum).
2. Pemesanan,pembayaran, dan pemberian data diri untuk jaminan via *chat*.
3. Pengambilan atau pengiriman barang.
4. Pengembalian barang setelah masa sewa selesai dan pengembalian data.

Konsep penyewaan ini menjadi solusi praktis bagi pelanggan yang membutuhkan barang hanya untuk jangka waktu tertentu, tanpa perlu membeli barang tersebut.

2.4 Teknologi yang Digunakan

Pengembangan sistem informasi penyewaan kostum *cosplay* berbasis *website* ini, digunakan beberapa teknologi untuk mendukung proses perancangan, pengembangan, dan implementasi sistem agar berjalan secara optimal. Adapun teknologi yang digunakan meliputi.

2.4.1 Bahasa Pemrograman

Bahasa pemrograman digunakan untuk mengembangkan logika, fitur, dan tampilan dalam sistem berbasis *web* (Satria et al, 2023). Dalam pengembangan sistem penyewaan kostum, kombinasi beberapa bahasa pemrograman digunakan untuk membangun antarmuka pengguna, mengelola data, dan mengatur proses bisnis. Beberapa bahasa pemrograman yang relevan untuk sistem ini antara lain:

- a. *HTML (HyperText Markup Language)*: *HTML* adalah fondasi utama dalam membangun antarmuka pengguna, seperti form pemesanan, tampilan daftar kostum, dan halaman login.
- b. *CSS (Cascading Style Sheets)*: Visual dan tata letak halaman *web* Dengan *CSS*, elemen-elemen *HTML* dapat diberi gaya agar tampil lebih menarik dan responsif di berbagai perangkat.
- c. *JavaScript*: Digunakan untuk memberikan interaktivitas pada halaman *web* di sisi klien (*frontend*), seperti validasi formulir, tampilan dinamis, dan komunikasi asinkron menggunakan *AJAX*. *JavaScript* juga dapat digunakan di sisi *server* menggunakan *Node.js*, memungkinkan pengembangan *backend* berbasis *JavaScript*.
- d. *PHP*: Merupakan bahasa pemrograman *server-side* yang banyak digunakan dalam pengembangan aplikasi *web* dinamis. *PHP* memproses permintaan dari pengguna, mengelola logika bisnis, dan berinteraksi dengan database (seperti *MySQL*) untuk menampilkan data sesuai kebutuhan.

2.4.2 Framework Web

PHP Native merujuk pada bahasa pemrograman *PHP* secara langsung tanpa menggunakan bantuan kerangka kerja tambahan. Pendekatan ini umumnya digunakan dalam pembangunan aplikasi web yang sederhana atau sebagai dasar dalam proses pembelajaran pemrograman web. Meskipun tidak menggunakan framework, struktur kode tetap dapat diatur secara sistematis dengan menerapkan pola arsitektur *Model-View-Controller* (MVC).

Pola MVC membantu memisahkan bagian-bagian penting dari aplikasi menjadi tiga komponen utama:

- a. *Model*: Berfungsi untuk mengelola data dan logika bisnis aplikasi.
- b. *View*: Bertanggung jawab terhadap tampilan antarmuka pengguna.
- c. *Controller*: Menangani interaksi pengguna, memproses masukan, serta menjembatani komunikasi antara model dan view.

Menggunakan pola MVC dalam pengembangan *PHP Native*, kode menjadi lebih terstruktur dan mudah dipelihara. Pemisahan antara logika dan tampilan ini juga mempermudah proses pengembangan lanjutan di masa depan. Panduan dan tutorial mengenai penerapan arsitektur MVC dengan *PHP Native* tersedia luas di berbagai sumber pembelajaran daring.

2.4.3 Database

Database adalah sistem yang dirancang untuk menyimpan, mengelola, dan mengambil data dengan cepat dan efisien. Sistem penyewaan kostum, pemilihan teknologi database yang tepat akan membantu menjaga konsistensi data serta mempercepat proses transaksi dan pencarian informasi. Beberapa teknologi database yang digunakan dalam sistem ini antara lain:

- a. *MySQL*: Sistem manajemen basis data relasional (RDBMS) *open-source* yang mendukung bahasa *SQL (Structured Query Language)*. *MySQL* digunakan untuk menyimpan data yang terstruktur seperti informasi pengguna, data kostum, transaksi penyewaan, dan histori pengembalian. *MySQL* dikenal karena kecepatan aksesnya dan kompatibilitas tinggi dengan berbagai platform.
- b. *InnoDB*: Merupakan *storage engine default* pada *MySQL* yang mendukung fitur transaksi dan integritas data. *InnoDB* mematuhi prinsip *ACID (Atomicity, Consistency, Isolation, Durability)*, yang menjamin setiap transaksi data tereksekusi dengan aman dan konsisten. Fitur ini penting dalam sistem penyewaan, seperti saat proses booking dan pembayaran dilakukan secara bersamaan.

- c. *UTF-8 Character Encoding*: Sistem mendukung input teks multibahasa seperti nama pelanggan, deskripsi kostum, atau alamat digunakan *encoding* *UTF-8*. Karakter *encoding* ini memungkinkan penyimpanan karakter dari berbagai bahasa dengan efisien dan aman tanpa kehilangan informasi.
- d. *Optimized Indexes*: Sistem dapat memberikan respon yang cepat terhadap kueri yang sering digunakan (seperti pencarian kostum berdasarkan kategori, tanggal ketersediaan, atau nama pelanggan), maka digunakan indeks yang dioptimalkan pada kolom-kolom penting. Indeks ini mempercepat proses pencarian dan pengambilan data dari tabel-tabel besar dalam database.

2.4.4 Mailtrap

Mailtrap adalah sebuah layanan *Email* testing berbasis *cloud* yang dirancang untuk menangkap *Email* yang dikirim dari aplikasi selama tahap pengembangan dan pengujian (Alfarizi, 2024). Layanan ini memungkinkan pengembang untuk memverifikasi proses pengiriman *Email* tanpa harus benar-benar mengirimkannya ke alamat *Email* asli, sehingga mengurangi risiko kesalahan dan menjaga kerahasiaan data.

Menurut dokumentasi resmi *Mailtrap* (Mailtrap.io, 2024), sistem ini menyediakan inbox virtual tempat *Email* hasil pengujian ditampung. Pengguna dapat memeriksa isi, *header*, lampiran, serta struktur *Email* untuk memastikan bahwa pesan yang dikirim sesuai dengan format dan tujuan yang diharapkan. Hal ini sangat berguna dalam pengembangan fitur-fitur seperti verifikasi akun, reset kata sandi, maupun notifikasi transaksi.

Mailtrap mendukung integrasi dengan berbagai bahasa pemrograman dan framework seperti *PHP*, *Laravel*, *Node.js*, *Python*, dan lainnya. Konfigurasi *SMTP* (*Simple Mail Transfer Protocol*) yang fleksibel memudahkan pengembang untuk menerapkannya dalam proyek tanpa perubahan besar pada arsitektur sistem.

Proses pengujian *Email* menjadi lebih aman, efisien, dan terkendali, terutama pada sistem berbasis *web* yang melibatkan komunikasi melalui *Email* secara aktif.

2.4.5 PHPMailer

PHPMailer adalah pustaka *PHP* yang dirancang untuk mempermudah proses pengiriman *Email* melalui protokol *SMTP* (*Simple Mail Transfer Protocol*). Dibandingkan dengan fungsi *mail()* bawaan *PHP*, *PHPMailer* menawarkan fitur yang lebih lengkap dan fleksibel, seperti autentikasi *SMTP*, enkripsi *SSL/TLS*, dukungan pengiriman dalam format *HTML* dan teks biasa, pengelolaan lampiran, serta pengaturan penerima seperti *CC* dan *BCC*.

Fitur-fitur tersebut, *PHPMailer* menjadi pilihan yang lebih andal dalam membangun sistem pengiriman *Email* yang aman, profesional, dan mudah disesuaikan dengan kebutuhan aplikasi *webmodern* (Mailtrap, 2025; UniOne, 2025; Cloudways, 2025)

Menurut dokumentasi resmi *PHPMailer* (*PHPMailer GitHub*, 2024), pustaka ini dirancang untuk memudahkan integrasi pengiriman *Email* dalam aplikasi *web* tanpa harus menangani detail teknis protokol *Email* secara langsung. *PHPMailer* dapat digunakan untuk berbagai kebutuhan seperti notifikasi sistem, pengiriman verifikasi akun, laporan otomatis, maupun komunikasi berbasis *Email* lainnya.

Library ini mendukung berbagai fitur tambahan seperti pengaturan *header* kustom, pengiriman ke banyak penerima, serta penyisipan gambar inline. *PHPMailer* juga kompatibel dengan berbagai *framework PHP* seperti *Laravel*, *CodeIgniter*, dan *Symfony*, sehingga memudahkan integrasi dalam berbagai jenis proyek pengembangan web. Menggunakan *PHPMailer*, proses pengiriman *Email* menjadi lebih aman, terstruktur, dan dapat dikontrol dengan lebih baik selama proses pengembangan maupun produksi (Pahrizal et al, 2022).

2.4.6 Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC) atau Siklus Hidup Pengembangan Perangkat Lunak merupakan proses terstruktur yang digunakan untuk merancang dan membangun sistem informasi, termasuk sistem informasi penyewaan kostum *cosplay* berbasis *website*. Proses ini mencakup beberapa tahapan yang saling berurutan untuk memastikan sistem dikembangkan secara sistematis dan sesuai kebutuhan pengguna. Tahapan pertama adalah perencanaan, di mana peneliti atau pengembang menetapkan tujuan utama sistem, menyusun jadwal pengembangan,

serta memperkirakan kebutuhan sumber daya. Tahap perencanaan mencakup identifikasi kebutuhan penyewa dan penyedia kostum, serta permasalahan yang ingin diselesaikan melalui digitalisasi proses penyewaan.

Tahap analisis kebutuhan dilakukan untuk mengumpulkan informasi dari pengguna potensial, seperti penyewa kostum dan pengelola penyewaan, guna merumuskan kebutuhan fungsional dan non-fungsional dari sistem yang akan dibangun. Tahapan ini penting untuk memastikan bahwa fitur-fitur sistem, seperti pencarian kostum, verifikasi identitas (KYC), dan manajemen transaksi, benar-benar dibutuhkan oleh pengguna.

Tahap berikutnya adalah perancangan sistem (desain), yang mencakup perancangan antarmuka pengguna, struktur database, dan alur interaksi sistem. Pada sistem informasi ini, desain meliputi tampilan halaman utama, halaman katalog kostum, dashboard pengguna, dan sistem administrasi untuk verifikasi data penyewa.

Kemudian masuk ke tahap pengembangan, yaitu proses implementasi desain menjadi kode program menggunakan bahasa pemrograman tertentu. Dalam kasus ini, sistem dikembangkan menggunakan *PHP Native* dengan pola arsitektur *Model-View-Controller* (MVC) untuk menjaga struktur program tetap rapi dan mudah dipelihara.

Setelah selesai dikembangkan, sistem diuji pada tahap pengujian untuk memastikan bahwa semua fungsi berjalan sesuai harapan. Pengujian dilakukan mulai dari fungsionalitas dasar, seperti login dan registrasi, hingga pengujian sistem transaksi dan fitur pencarian kostum.

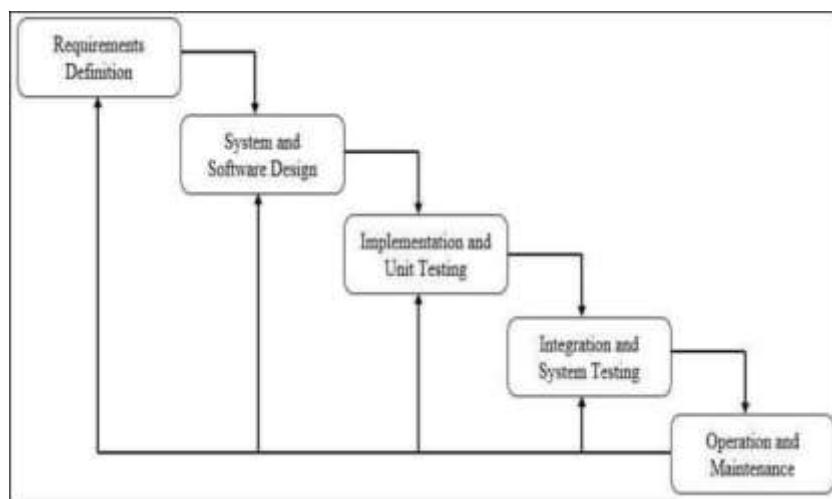
Jika sistem telah lolos uji, maka dilakukan tahap implementasi, yaitu proses peluncuran sistem agar dapat digunakan oleh pengguna secara langsung. Pengguna dapat mulai menyewa dan mengelola kostum secara *online* melalui platform yang telah dikembangkan.

Tahap terakhir adalah pemeliharaan, yang melibatkan perbaikan bug, peningkatan fitur, serta penyesuaian terhadap kebutuhan pengguna yang mungkin berubah di masa mendatang. Tahapan ini penting untuk menjaga sistem tetap relevan dan optimal dalam mendukung proses penyewaan kostum secara digital.

Mengikuti proses SDLC secara menyeluruh, pengembangan sistem informasi penyewaan kostum *cosplay* ini dapat dilakukan secara lebih terstruktur, minim risiko, dan menghasilkan sistem yang sesuai dengan kebutuhan pengguna.

2.4.7 Metode Waterfall

Metode *Waterfall* merupakan salah satu model pengembangan perangkat lunak klasik yang bersifat linear dan sekuensial. Proses pengembangan dalam model ini dilakukan secara bertahap, dimulai dari analisis kebutuhan hingga pemeliharaan, setiap tahap diselesaikan sepenuhnya sebelum melanjutkan ke tahap berikutnya menyerupai aliran air terjun yang mengalir dari atas ke bawah (Baeldung, 2023).



Gambar 1. *Waterfall Model* (Model Air Terjun) dalam pengembangan perangkat lunak.

Panah dari atas ke bawah ($\rightarrow \downarrow$):

Kegunaan: Menunjukkan alur kerja normal/maju (*forward flow*)

Makna: Proses berlanjut ke tahap berikutnya setelah tahap sebelumnya selesai

Contoh: Setelah *Requirements Definition* selesai, dilanjutkan ke *System and Software Design*

Panah dari bawah ke atas ($\uparrow \leftarrow$):

Kegunaan: Menunjukkan alur balik/mundur (*feedback flow*)

Makna: Kemungkinan untuk kembali ke tahap sebelumnya jika ditemukan

masalah atau perlu perbaikan

Contoh: Jika pada tahap Testing ditemukan bug, bisa kembali ke tahap *implementation* untuk perbaikan

1. Analisis Kebutuhan (*Requirement Analysis*)

Tahap ini dilakukan pengumpulan dan analisis kebutuhan sistem dari pengguna atau *stakeholder*. Semua kebutuhan dicatat dan dijadikan dasar dalam merancang sistem. Contoh pada sistem penyewaan kostum: kebutuhan login, verifikasi KTP, fitur penyewa dan pelanggan, dll.

2. Perancangan Sistem (*System Design*)

Berdasarkan kebutuhan yang telah ditentukan, perancangan dilakukan dalam bentuk diagram seperti:

- a. *ERD (Entity Relationship Diagram)*
- b. *DFD (Data Flow Diagram)*
- c. *UML (Use Case, Activity Diagram)*
- d. Struktur database

Tujuannya agar sistem memiliki kerangka yang jelas sebelum diimplementasikan.

3. Implementasi (*Coding*)

Tahap ini adalah proses pengubahan desain sistem menjadi bentuk nyata dengan menggunakan bahasa pemrograman (seperti *PHP*, *HTML*, *SQL*, dll). Contoh: membuat modul *login*, halaman *dashboard*, katalog penyewaan, *form* pemesanan, dll.

4. Pengujian (*Testing*)

Setelah sistem selesai dibuat, dilakukan pengujian untuk memastikan fungsionalitas sistem berjalan dengan baik dan sesuai dengan kebutuhan awal. Contohnya: menggunakan *Black Box Testing* untuk mengecek apakah fitur login, verifikasi, pemesanan, dll bekerja sesuai harapan.

5. Pemeliharaan (*Maintenance*)

Tahap ini dilakukan setelah sistem digunakan. Jika ditemukan *bug* atau diperlukan perubahan kecil, maka dilakukan perbaikan.

2.4.8 Black Box Testing

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus pada fungsionalitas sistem tanpa memeriksa struktur internal kode program. Pengujian ini dilakukan dengan memberikan input dan mengevaluasi output apakah sudah sesuai dengan yang diharapkan (Nurfauziah et al, 2022).

Penguji *Black Box Testing* tidak mengetahui bagaimana logika dan algoritma dalam sistem bekerja. Yang diuji hanya apa yang terlihat oleh pengguna akhir (*end user*), seperti form input, tampilan halaman, tombol, dan hasil proses. Tujuan utama *black box testing* adalah untuk:

1. Memastikan bahwa fungsi sistem bekerja sesuai spesifikasi.
2. Menemukan kesalahan pada antarmuka (UI), fungsionalitas, dan output.
3. Meningkatkan kualitas perangkat lunak dari sisi pengguna.

Tabel 1 menampilkan contoh penerapan Sistem Informasi Penyewaan Kostum *Cosplay* Karakter Anime Berbasis *Website* Di Bandar Lampung.

Tabel 1. Contoh Penerapan Sistem Informasi Penyewaan Kostum *Cosplay*
Karakter Anime Berbasis *Website* Di Bandar Lampung

Fitur yang Diuji	Input	Proses	Output yang Diharapkan
Login	<i>Email</i> dan <i>Password</i> valid	Validasi user	Masuk ke dashboard
Login	<i>Email</i> salah / <i>Password</i> salah	Validasi user	Muncul pesan <i>Error</i>
Tambah Kostum	<i>Form input</i> lengkap dan valid	Simpan ke database	Data tersimpan, muncul notifikasi sukses
Tambah Kostum	<i>Form input</i> kosong	Validasi form	Muncul peringatan <i>Error</i>
Pemesanan Kostum	Pilih kostum dan isi form penyimpanan	Proses	Pesanan berhasil dicatat

	penyewaan	
Pemesanan	Tidak memilih	Validasi
Kostum	kostum	Tampil pesan <i>Error</i>

2.5 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan visual standar yang digunakan untuk memvisualisasikan, mendokumentasikan, dan merancang sistem perangkat lunak. UML dikembangkan oleh Object Management Group (OMG) dan digunakan secara luas dalam pengembangan sistem berbasis objek karena mampu menggambarkan struktur dan perilaku sistem dari berbagai sudut pandang (Ruang Developer, 2023).

Menurut Object Management Group (OMG), UML dirancang untuk menjadi bahasa pemodelan umum yang dapat digunakan oleh pengembang perangkat lunak untuk menggambarkan struktur dan perilaku sistem dari berbagai perspektif. UML mendukung berbagai jenis diagram, yang secara umum dibagi menjadi tiga kategori utama:

1. Diagram Struktur: Menggambarkan aspek statis dari sistem, seperti *Class Diagram*, *Component Diagram*, dan *Deployment Diagram*.
2. Diagram Perilaku: Menggambarkan aspek dinamis dari sistem, termasuk *Use Case Diagram*, *Activity Diagram*, dan *State Machine Diagram*.
3. Diagram Interaksi: Fokus pada aliran kontrol dan data antara komponen sistem, seperti *Sequence Diagram*, *Communication Diagram*, dan *Timing Diagram*.

UML digunakan secara luas dalam berbagai metodologi pengembangan perangkat lunak, termasuk *Agile*, *RUP*, dan *Model-Driven Architecture (MDA)*. Penggunaan UML membantu tim pengembang dalam berkomunikasi secara efektif, mengidentifikasi kebutuhan sistem, dan merancang solusi yang sesuai dengan kebutuhan bisnis.

2.6 Use Case Diagram

Use Case adalah teknik pemodelan dalam rekayasa perangkat lunak yang digunakan untuk menggambarkan interaksi antara aktor (pengguna atau sistem eksternal) dengan sistem yang dikembangkan, dengan tujuan mencapai suatu hasil yang bernilai bagi aktor tersebut. Model ini membantu dalam memahami kebutuhan fungsional sistem dari perspektif pengguna, sehingga memudahkan dalam perancangan dan pengujian sistem. Menurut Visual Paradigm (2023), *Use Case* terdiri dari beberapa komponen utama:

1. Aktor: Entitas eksternal yang berinteraksi dengan sistem, bisa berupa pengguna manusia, sistem lain, atau perangkat keras eksternal.
2. *Use Case*: Deskripsi fungsi atau layanan yang disediakan oleh sistem untuk aktor.
3. Hubungan: Interaksi antara aktor dan *use case*, yang dapat berupa asosiasi, inklusi (*include*), perluasan (*extend*), atau generalisasi.

Pada Tabel 2 menjelaskan tentang simbol dan deskripsi pada *Use Case Diagram*

Tabel 2. Use Case Diagram

No	Simbol	Deskripsi
1.		<i>Use case</i> : Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.		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.

Diagram *Use Case* digunakan untuk memvisualisasikan hubungan ini, memberikan gambaran umum tentang fungsi-fungsi yang disediakan oleh sistem

dan bagaimana aktor berinteraksi dengan fungsi-fungsi tersebut. Diagram ini sangat berguna dalam tahap awal pengembangan sistem untuk menangkap dan mengklarifikasi kebutuhan pengguna.

2.7 Activity Diagram

Activity Diagram adalah salah satu jenis behavioral diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan alur kendali (*control flow*) dan aktivitas dalam suatu sistem, proses, atau prosedur bisnis. Diagram ini menunjukkan bagaimana suatu proses dimulai, apa saja aktivitas yang terjadi, keputusan yang harus diambil, dan bagaimana proses tersebut berakhir.

Menurut Firmansyah (2025), *Activity Diagram* mampu menangkap kompleksitas proses sistem secara visual, termasuk mendukung kondisi percabangan (*branching*), pengulangan (*looping*), serta alur paralel (*parallel processing*) menjadikannya alat penting dalam pengembangan sistem informasi. Menurut Firmansyah (2025), terdapat beberapa fungsi utama dari *Activity Diagram*, yaitu:

1. Memvisualisasikan Urutan Aktivitas dalam Sistem

Activity Diagram membantu menggambarkan proses secara runut dan terstruktur, dari aktivitas pertama hingga terakhir. Ini memudahkan developer maupun stakeholder dalam memahami *flow* atau urutan eksekusi dari suatu proses, seperti alur registrasi, proses pemesanan, hingga konfirmasi pembayaran.

2. Mendokumentasikan Proses Bisnis atau Sistem

Diagram ini sangat berguna sebagai alat dokumentasi sistem, karena dapat mencatat proses dan alur keputusan yang ada di dalam sistem. Dokumentasi ini dapat digunakan untuk kebutuhan analisis, pengujian, maupun pelatihan pengguna sistem.

3. Mengidentifikasi Aktivitas Paralel dan Keputusan

Sistem kompleks, terdapat aktivitas yang berjalan secara bersamaan (paralel) dan percabangan logika yang menentukan alur eksekusi selanjutnya. *Activity Diagram*

membantu mengidentifikasi bagian-bagian ini dengan simbol *fork*, *join*, dan *decision node*, sehingga mempermudah analisis efisiensi proses.

4. Menyediakan Pemahaman Awal yang Mudah bagi *Stakeholder* Non-Teknis

sifatnya visual dan tidak memerlukan pengetahuan teknis yang tinggi, *Activity Diagram* sangat cocok untuk menjelaskan alur sistem kepada pengguna akhir, manajer proyek, atau pihak non-programmer lainnya. Diagram ini menjadi *jembatan komunikasi* antara tim teknis dan non-teknis.

Pada Tabel 3 adalah simbol simbol yang digunakan pada *activity diagram* dan juga penjelasannya

Tabel 3. Simbol activity diagram

No.	Simbol	Keterangan
1.		Status awal aktivitas sistem, 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.

III. ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi fitur dan spesifikasi yang diperlukan dalam pengembangan sistem informasi penyewaan kostum *cosplay*. Kebutuhan ini dibagi menjadi dua kategori utama, yaitu kebutuhan fungsional dan kebutuhan non-fungsional (Putra et al, 2020).

3.1.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan yang berkaitan dengan fungsi-fungsi utama yang harus dimiliki oleh sistem agar dapat berjalan sesuai tujuan. Berikut adalah kebutuhan fungsional sistem:

- a. Pendaftaran Pengguna : Sistem harus menyediakan fitur pendaftaran untuk penyewa dan penyedia jasa dengan validasi data.
- b. Pencarian Kostum : Sistem memungkinkan pengguna untuk mencari kostum berdasarkan kategori, tema, ukuran, atau ketersediaan.
- c. Pemesanan Kostum : Sistem menyediakan fitur untuk memesan kostum yang terhubung dengan penjual
- d. Pengelolaan Inventaris : Sistem memungkinkan penyedia jasa untuk, atau memperbarui katalog kostum.

3.1.2 Kebutuhan Non Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang berkaitan dengan aspek kualitas dan kinerja sistem. Berikut adalah kebutuhan non-fungsional sistem:

- a. Keamanan Data : Sistem harus menjamin keamanan data pengguna dengan enkripsi dan autentikasi.

- b. Kemudahan Penggunaan : Antarmuka sistem harus *user-friendly* dan mudah digunakan oleh pengguna dengan tingkat kemampuan yang berbeda
- c. Skalabilitas : Sistem harus dirancang agar dapat diperluas untuk menampung peningkatan jumlah pengguna atau fitur di masa depan.
- d. Ketersediaan Sistem : Sistem harus memiliki *uptime* minimal 99% agar dapat diakses kapan saja oleh pengguna.
- e. Waktu Respon : Sistem harus mampu merespons permintaan pengguna dengan cepat.

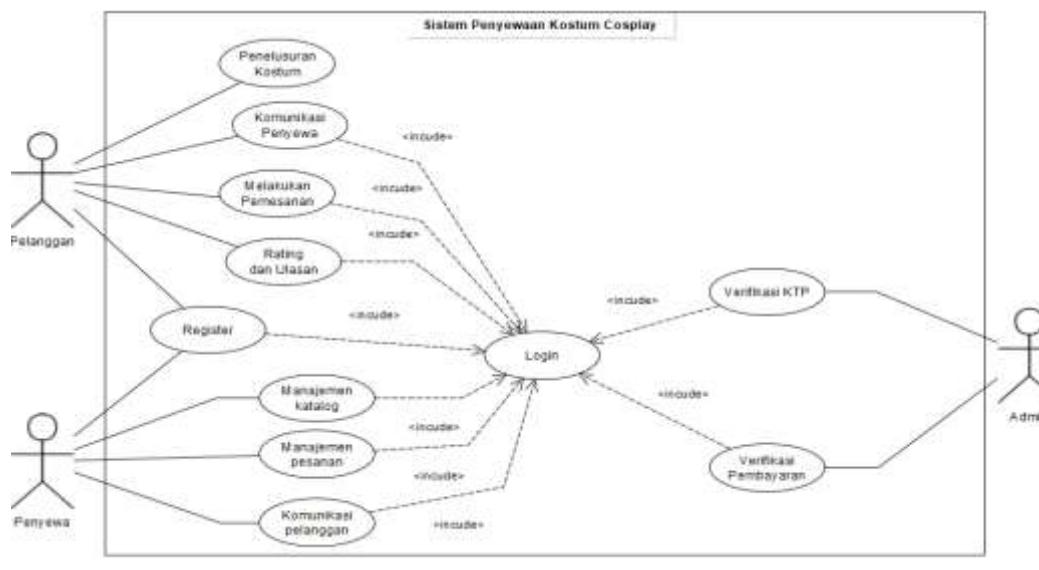
3.2 Desain Sistem

Desain sistem merupakan tahapan penting dalam pengembangan sistem informasi, karena menjadi dasar dalam implementasi sistem yang akan dibangun. Pada tahap ini, dilakukan perancangan terhadap alur kerja, struktur data, serta interaksi antar pengguna dan sistem.

Desain sistem untuk aplikasi penyewaan kostum *cosplay* berbasis *website* ini mencakup beberapa aspek utama, yaitu:

3.2.1 Use Case Diagram

Use Case Diagram menggambarkan interaksi antara aktor (Pelanggan, Pemilik Sewa, dan Admin) dengan sistem, termasuk fitur-fitur utama seperti *login*, registrasi, pencarian kostum, pengelolaan inventaris, dan pengelolaan pengguna (Yudhistira dan Endah, 2024). Gambar 4 menunjukkan *use case diagram* sistem informasi penyewaan kostum.



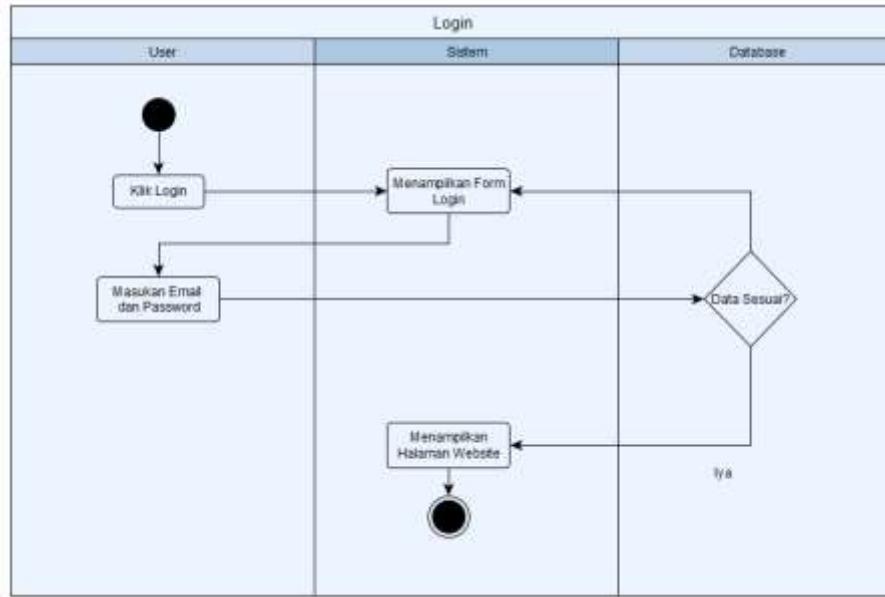
Gambar 2. *Use Case Diagram* Sistem Informasi Penyewaan Kostum Cosplay

Activity Diagram merupakan salah satu jenis *behavioral diagram* dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan alur kerja (*workflow*) atau proses bisnis dalam suatu sistem. Diagram ini menggambarkan urutan aktivitas dari awal hingga akhir, termasuk keputusan (*decision*), pengulangan (*looping*), dan alur paralel (*concurrent flow*), yang sangat berguna dalam memahami dinamika sistem sebelum implementasi teknis dilakukan.

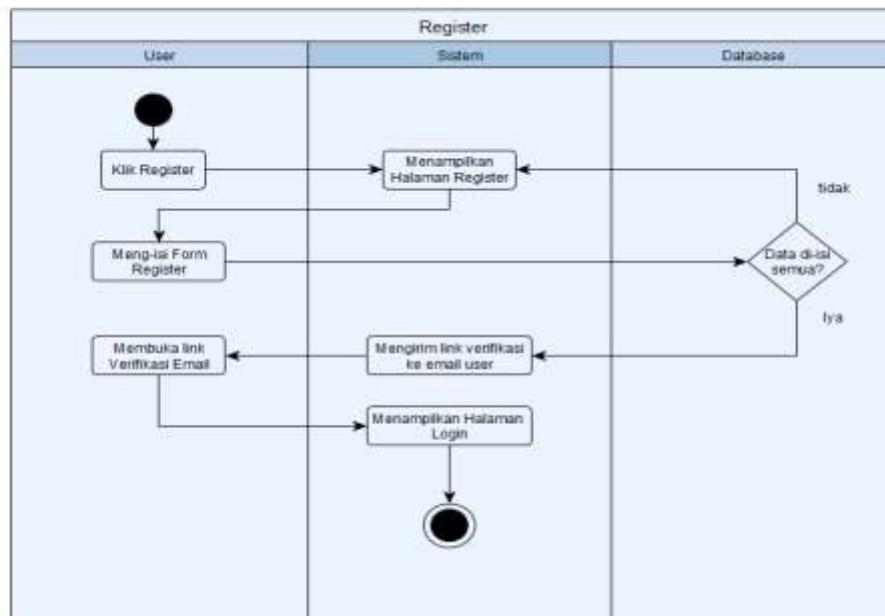
Menurut *Software seni* (Firmansyah, 2025), *Activity Diagram* membantu developer dan analis sistem untuk menggambarkan bagaimana proses berjalan secara logis dan visual, sehingga mempercepat proses desain dan dokumentasi sistem informasi.

1. *Login*

Activity Diagram login menggambarkan proses autentikasi pengguna ke dalam sistem. Proses dimulai ketika pengguna mengklik tombol *Login*, lalu sistem menampilkan form login. Setelah pengguna memasukkan *Email* dan *Password*, sistem akan memverifikasi data ke dalam database. Sistem menampilkan halaman *website* yang sesuai dengan peran pengguna (penyewa, pelanggan, atau admin). Proses ini menekankan pentingnya keamanan dan validasi kredensial dalam sistem informasi. Gambar 3 adalah *activity diagram* dari login

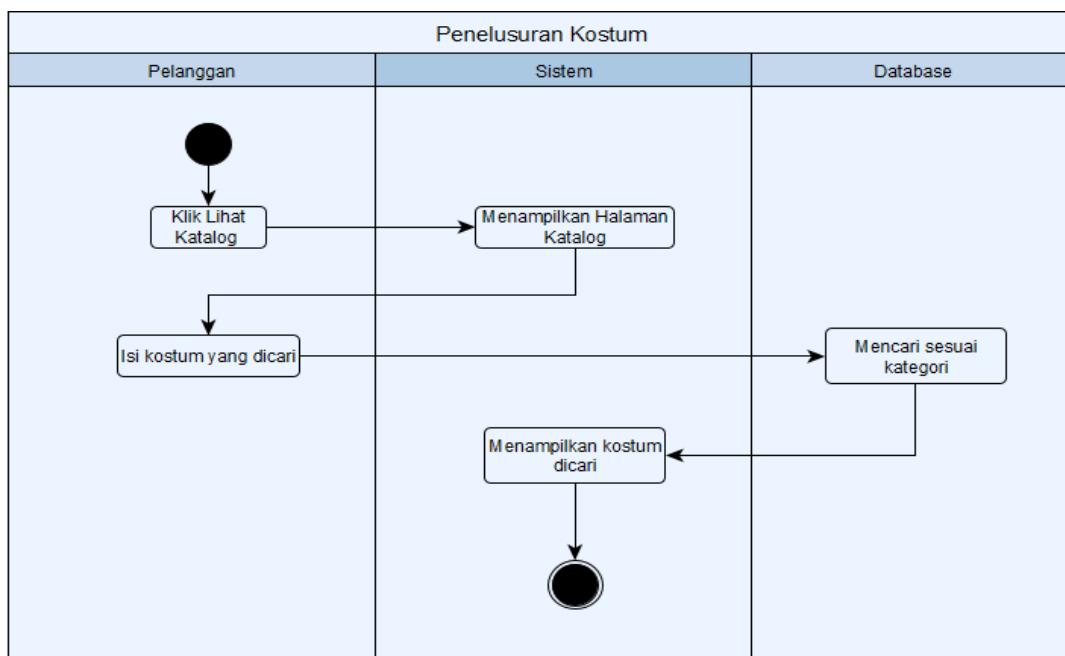
Gambar 3. *Activity Diagram Login*

2. *Register Diagram* ini menunjukkan alur pengguna baru mendaftar ke sistem. Proses diawali dengan mengklik *Register*, kemudian pengguna mengisi form pendaftaran. Setelah form lengkap, sistem mengirimkan link verifikasi ke *Email* pengguna. Apabila link verifikasi diakses, sistem menampilkan halaman *login*. Gambar 4 adalah *activity diagram* dari *register*

Gambar 4. *Activity Diagram Register*

3. Penelusuran Kostum

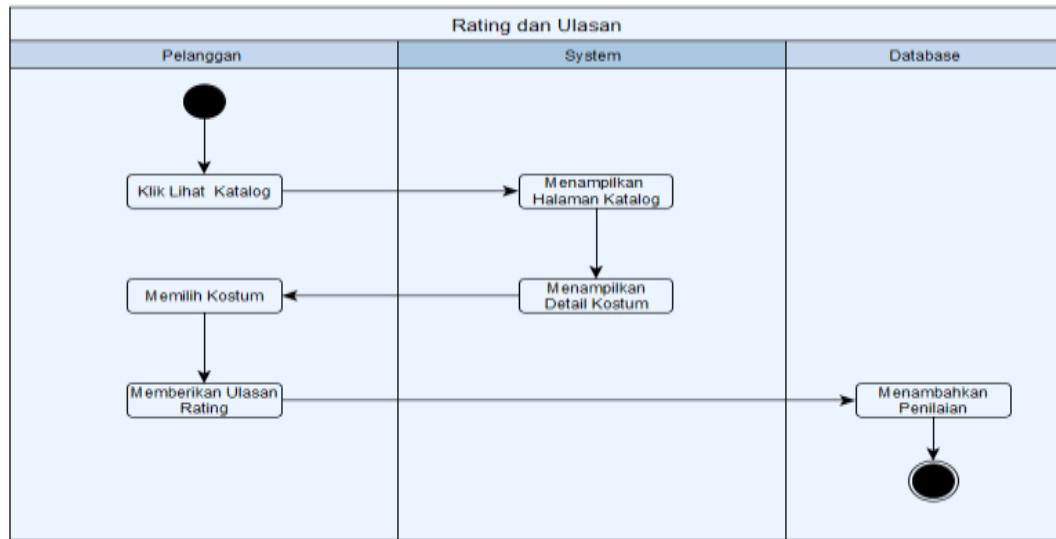
Diagram ini menggambarkan proses pencarian kostum oleh pelanggan. Dimulai dari klik *Lihat Katalog*, sistem menampilkan daftar kostum. Pengguna kemudian mengisi kata kunci atau kategori, sistem mencocokkannya di *database*, dan menampilkan hasil pencarian. Proses ini penting dalam meningkatkan pengalaman pengguna melalui pencarian yang cepat dan responsif. Gambar 5 adalah *activity diagram* dari penelusuran kostum.



Gambar 5. *Activity Diagram* Penelusuran Kostum

4. Rating dan Ulasan

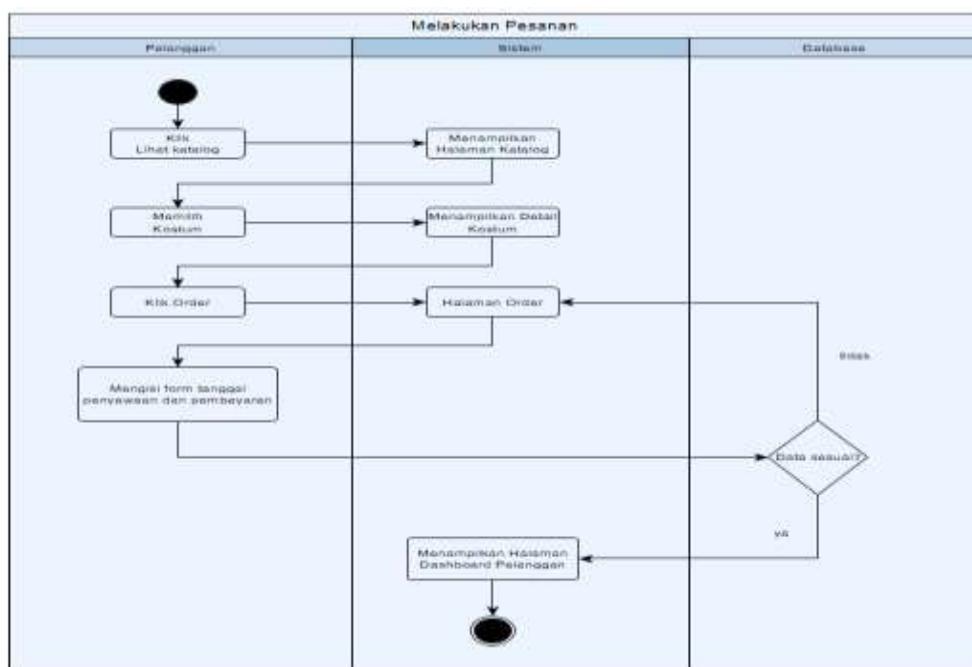
Proses dimulai ketika pelanggan membuka katalog dan memilih kostum. Setelah menyelesaikan penyewaan, pelanggan dapat memberikan ulasan berupa penilaian dan komentar. Sistem lalu menyimpan ulasan tersebut ke dalam database. Diagram ini merepresentasikan fitur *feedback* dua arah, yang penting untuk membangun reputasi penyewa dan meningkatkan kepercayaan pengguna. Gambar 6 adalah *activity diagram* dari rating dan ulasan.



Gambar 6. *Activity Diagram* dari rating dan ulasan

5. Melakukan Pemesanan

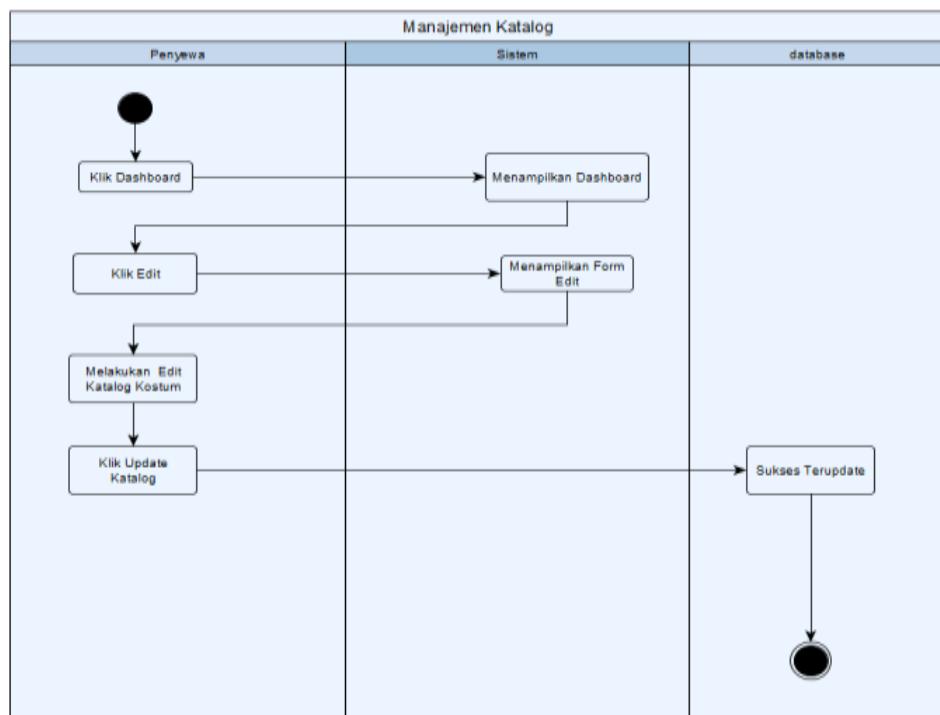
Diagram ini menunjukkan proses pelanggan memesan kostum. Dimulai dari memilih katalog, lalu klik *Order*, kemudian mengisi tanggal penyewaan dan pembayaran. Sistem memverifikasi validitas data yang dimasukkan. Jika valid, sistem menampilkan dashboard pelanggan sebagai konfirmasi. Diagram ini menegaskan pentingnya validasi dan form interaktif pada sistem penyewaan berbasis *website*. Gambar 7 adalah *activity diagram* dari melakukan pemesanan.



Gambar 7. *Activity Diagram* Melakukan Pemesanan

6. Manajemen Katalog

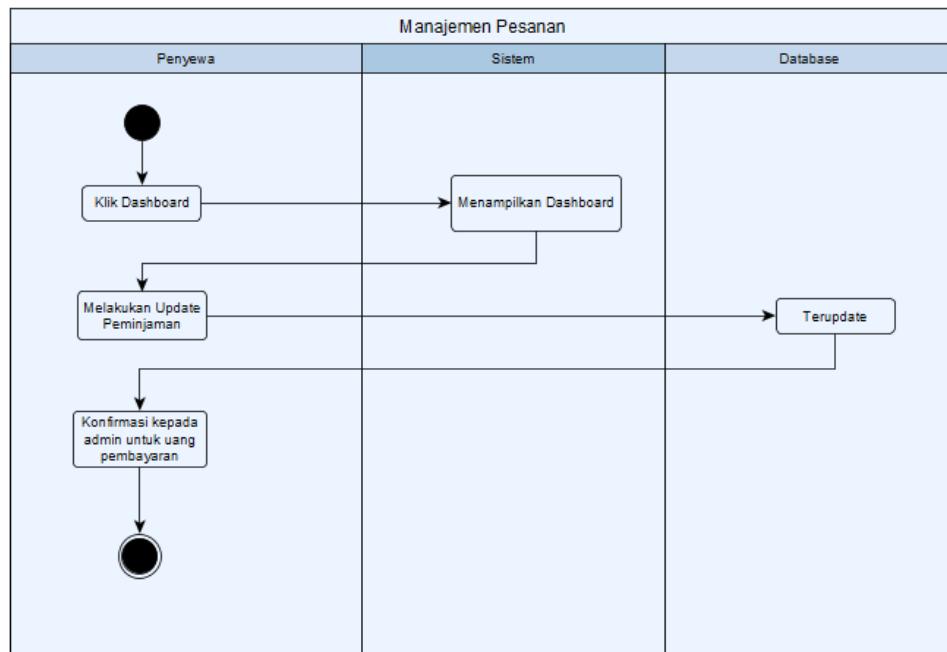
Aktivitas ini dilakukan oleh penyewa. Proses dimulai dari akses *dashboard*, klik *Edit*, kemudian sistem menampilkan form edit katalog. Penyewa memperbarui data dan menyimpannya. Sistem lalu memperbarui data di *database* dan memberi konfirmasi sukses. Diagram ini menggambarkan bagaimana sistem mendukung pengelolaan konten dinamis oleh penyedia jasa. Gambar 8 adalah *activity diagram* manajemen katalog



Gambar 8. *Activity Diagram* Manajemen Katalog

7. Manajemen Pesanan

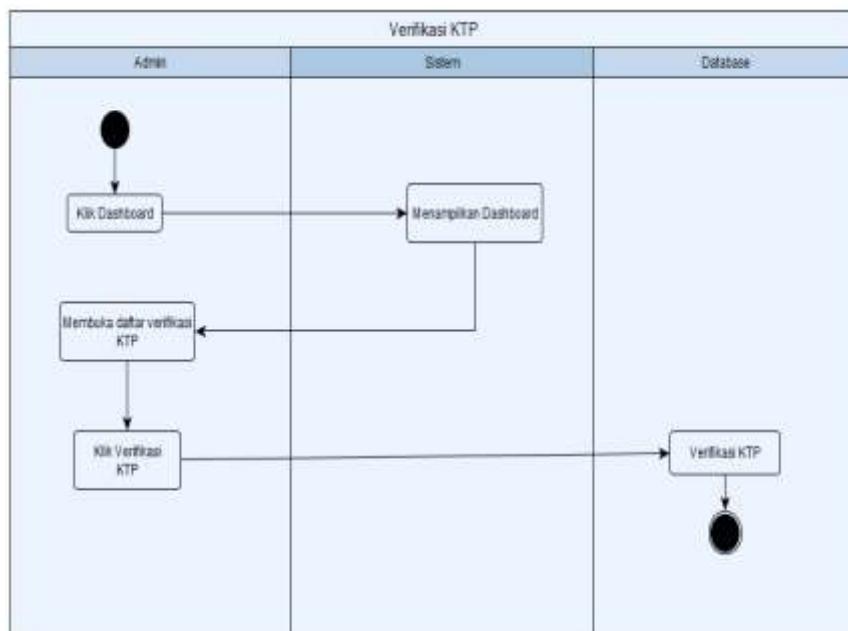
Diagram ini menunjukkan penyewa yang memperbarui status peminjaman dari dashboard. Setelah melakukan *update*, sistem memperbarui data di *database*. Penyewa juga dapat mengonfirmasi status pembayaran kepada admin. Diagram ini mencerminkan hubungan antara penyewa, sistem, dan admin dalam proses transaksi. Gambar 9 adalah *activity diagram* dari manajemen pesanan.



Gambar 9. *Activity Diagram* Manajemen Pesanan

8. Verifikasi KTP

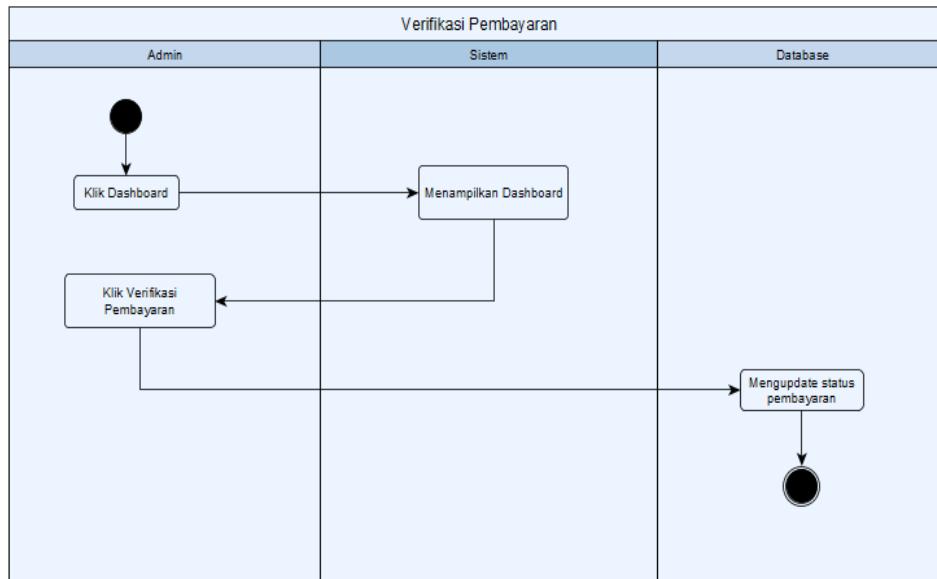
Aktivitas ini melibatkan admin yang memverifikasi identitas pengguna. Admin membuka dashboard, melihat daftar verifikasi, dan memilih pengguna untuk diverifikasi. Sistem kemudian memproses permintaan dan menandai status verifikasi pada database. Ini mencerminkan mekanisme keamanan dan kepercayaan berbasis KYC (*Know Your Customer*). Gambar 10 adalah *activity diagram* dari verifikasi KTP.



Gambar 10. *Activity Diagram* Verifikasi KTP

9. Verifikasi Pembayaran

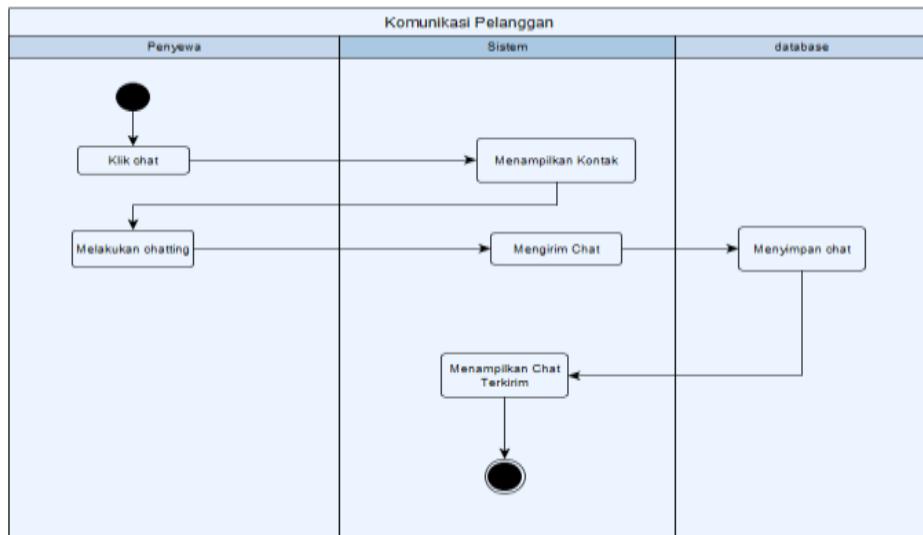
Diagram ini juga dilakukan oleh admin. Proses dimulai dari dashboard, lalu admin memilih verifikasi pembayaran. Sistem kemudian memperbarui status pembayaran pada database. Proses ini penting untuk memastikan transaksi penyewaan berjalan dengan sah dan terverifikasi. Gambar 11 adalah *activity diagram* dari verifikasi pembayaran.



Gambar 11. *Activity Diagram* Verifikasi Pembayaran

10. Komunikasi dengan Pelanggan

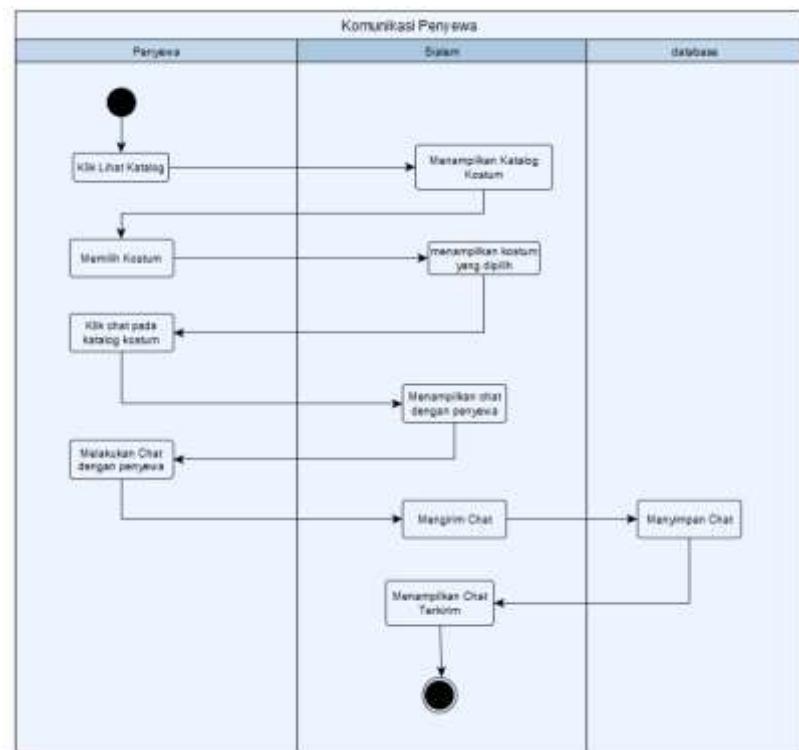
Diagram ini menggambarkan interaksi pesan antara penyewa dan pelanggan. Setelah klik *Chat*, sistem menampilkan daftar kontak. Pengguna dapat mengirim pesan, yang kemudian disimpan di database dan ditampilkan kembali sebagai pesan terkirim. Diagram ini menunjukkan fitur *real-time communication* sebagai penunjang layanan pelanggan. Gambar 12 adalah *activity diagram* dari komunikasi dengan pelanggan.



Gambar 12. *Activity Diagram* Komunikasi Pelanggan

11. Komunikasi dengan Penyewa

Activity Diagram ini menjelaskan alur proses komunikasi antara pelanggan dan penyewa dalam sistem informasi penyewaan kostum berbasis *web*. Fitur komunikasi ini sangat penting untuk memperlancar interaksi terkait detail kostum, negosiasi waktu peminjaman, atau pertanyaan teknis lainnya sebelum pemesanan dilakukan. Gambar 13 adalah *activity diagram* dari komunikasi dengan penyewa.



Gambar 13. *Activity Diagram* Komunikasi dengan Penyewa

3.2.2 Use Case Description

Deskripsi setiap *use case diagram* adalah sebagai berikut:

Tabel 4. *Use Case Description* untuk *Register*

Use Case Name : <i>Register</i>	ID: UC-01 Priority : Tinggi
Actor : Pengguna Baru (Calon Penyewa/Pelanggan)	
Description: Use case ini menggambarkan proses pendaftaran akun baru untuk pengguna yang ingin menjadi penyewa atau pelanggan dalam sistem.	
Trigger: Pengguna baru ingin membuat akun di sistem penyewaan	
Preconditions: Pengguna belum memiliki akun di sistem.	
Normal Course:	
1. Pengguna membuka halaman registrasi.	
2. Sistem menampilkan form registrasi dengan field: fullname, alamat, no_telepon, <i>Email</i> , <i>Password</i> , role, provinsi, kota, bio.	
3. Pengguna mengisi semua field yang diperlukan.	
4. Pengguna men-submit form registrasi.	
5. Sistem memvalidasi data yang dimasukkan.	
6. Jika valid, sistem menyimpan data ke tabel users dengan status <i>Email_verified</i> = false.	
7. Sistem mengirim <i>Email</i> verifikasi dengan token ke alamat <i>Email</i> pengguna.	
8. Sistem menampilkan pesan konfirmasi untuk memeriksa <i>Email</i> verifikasi.	
Sub Flows:	
Sistem melakukan validasi format <i>Email</i> dan kekuatan <i>Password</i> .	
Sistem membuat token verifikasi unik dengan masa berlaku 24 jam.	
Alternate / Exceptional Flows:	
1. Data tidak valid (<i>Email</i> sudah terdaftar, format <i>Email</i> salah, <i>Password</i> lemah): Sistem menampilkan pesan <i>Error</i> spesifik	
2. Gagal mengirim <i>Email</i> verifikasi: Log <i>Error</i> untuk admin	
3. Pengguna tidak mengisi field wajib: Sistem highlight field yang kosong Tampilkan pesan "Field ini wajib diisi"	
Jumlah Transaksi: Disesuaikan dengan kebutuhan proses <i>registrasi</i> pengguna baru dalam sistem penyewaan.	

Tabel 5. *Use Case Description* untuk *Login*

Use Case Name : Login	ID: UC-02 Priority : Tinggi
Actor : Admin, Penyewa, Pelanggan	
Description: Use case ini menggambarkan proses autentikasi pengguna untuk mengakses sistem	
Trigger: Pengguna ingin mengakses fitur dalam sistem	
Preconditions: Pengguna sudah memiliki akun yang terdaftar dan <i>Email</i> sudah terverifikasi.	
Normal Course:	
1. Pengguna membuka halaman login.	
2. Sistem menampilkan form login dengan field <i>Email</i> dan <i>Password</i> .	
3. Pengguna mengisi kredensial login.	
4. Pengguna men-submit form login.	
5. Sistem memverifikasi kredensial dengan data di tabel users.	
6. Jika valid, sistem membuat session untuk pengguna.	
7. Sistem mengarahkan pengguna ke dashboard sesuai role (admin/penyewa/pelanggan).	
Sub Flows:	
Sistem melakukan <i>rate limiting</i> untuk mencegah <i>brute force attack</i>	
Sistem mencatat waktu login terakhir dan <i>IP address</i>	
Sistem memperbarui status " <i>last_login</i> " di database	
Alternate / Exceptional Flows:	
Kredensial tidak valid: Sistem menampilkan pesan " <i>Email</i> atau <i>Password</i> salah"	
Tambah counter gagal login	
<i>Email</i> belum terverifikasi:	
Sistem tampilkan pesan verifikasi diperlukan	
Jumlah Transaksi: Disesuaikan dengan kebutuhan pengguna terhadap sistem.	

Tabel 6. *Use Case Description* untuk Verifikasi KTP

Use Case Name : Verifikasi KTP	ID: UC-03 Priority : Tinggi
<hr/>	
Actor : Admin	
<p>Description: Use case ini menggambarkan proses admin memverifikasi dokumen KTP yang diunggah oleh pengguna untuk keperluan KYC.</p> <p>Trigger: Admin perlu memverifikasi dokumen KTP pengguna.</p> <p>Preconditions: Admin sudah login dan terdapat pengguna yang telah mengunggah KTP (ktp_path tidak null).</p>	
<p>Normal Course:</p> <ol style="list-style-type: none"> 1. Admin memilih menu "Verifikasi KTP" di dashboard. 2. Sistem menampilkan daftar pengguna yang belum diverifikasi (verifikasi_ktp = false). 3. Admin memilih pengguna yang akan diverifikasi. 4. Sistem menampilkan detail pengguna dan foto KTP yang diunggah. 5. Admin memeriksa keaslian dan validitas dokumen KTP. 6. Admin mengubah status verifikasi_ktp menjadi true atau false. 7. Sistem memperbarui data di tabel users. 8. Sistem mengirim notifikasi ke pengguna tentang status verifikasi. <p>Sub Flows:</p> <p>Sistem mencatat waktu dan ID admin yang melakukan verifikasi.</p> <p>Sistem menyimpan <i>log</i> aktivitas verifikasi (audit trail).</p> <p>Sistem memungkinkan admin menambahkan catatan/verifikasi manual</p> <p>Sistem menghindari pengguna yang sudah diverifikasi muncul kembali dalam daftar.</p> <p>Alternate / Exceptional Flows:</p> <p>Sistem menampilkan pesan <i>Error</i>; admin meminta pengguna unggah ulang.</p> <p>Jumlah Transaksi: Disesuaikan dengan jumlah pengguna yang mengunggah dokumen KTP untuk diverifikasi setiap harinya.</p>	

Tabel 7. *Use Case Description* untuk Verifikasi Pembayaran

Use Case Name : Verifikasi Pembayaran ID: UC-04 Priority : Tinggi

Actor : Admin

Description: Use case ini menggambarkan proses admin memverifikasi bukti pembayaran dari pesanan pelanggan.

Trigger: Admin perlu memverifikasi pembayaran pesanan.

Preconditions: Admin sudah login dan terdapat pesanan dengan status_pembayaran = 'pending'.

Normal Course:

1. Admin memilih menu "Verifikasi Pembayaran" di dashboard.
2. Sistem menampilkan daftar pesanan dengan status pembayaran pending.
3. Admin memilih pesanan yang akan diverifikasi.
4. Sistem menampilkan detail pesanan dan bukti pembayaran yang diunggah.
5. Admin memeriksa keaslian bukti pembayaran.
6. Admin mengubah status_pembayaran menjadi 'verified' atau 'rejected'.
7. Sistem memperbarui data di tabel pesanan.
8. Jika verified, sistem mengubah status_peminjaman menjadi 'active'.
9. Sistem mengirim notifikasi ke pelanggan dan penyewa.

Sub Flows:

Sistem mencatat waktu verifikasi dan ID admin yang memverifikasi.

Sistem menyimpan log aktivitas sebagai audit trail.

Admin dapat menambahkan catatan verifikasi (misalnya alasan penolakan).

Alternate / Exceptional Flows:

Bukti pembayaran tidak jelas atau tidak sah: Admin menolak dan menambahkan catatan.

File tidak dapat dibuka / tidak valid: Sistem menampilkan pesan *Error*, admin minta pengguna unggah ulang.

Tidak ada pesanan pending: Sistem menampilkan pesan "Tidak ada pembayaran yang perlu diverifikasi".

Jumlah Transaksi: Disesuaikan dengan jumlah pesanan yang menunggu

verifikasi pembayaran setiap harinya.

Tabel 8. *Use Case Description* untuk Penelusuran Kostum

Use Case Name : Penelusuran Kostum ID: UC-05 Priority : Tinggi

Actor : Pelanggan

Description: Use case ini menggambarkan proses pencarian dan penelusuran kostum oleh pelanggan.

Trigger: Pelanggan ingin mencari kostum untuk disewa

Preconditions: Pelanggan sudah login ke sistem

Normal	Course	:
---------------	---------------	---

1. Pelanggan membuka halaman penelusuran kostum.
2. Sistem menampilkan form pencarian dengan filter: nama_kostum, kategori, ukuran, gender, kota, provinsi, harga.
3. Pelanggan memasukkan kata kunci atau mengatur filter pencarian.
4. Pelanggan men-submit pencarian.
5. Sistem mencari data di tabel katalog dengan status = 'available' dan visible = true.
6. Sistem menampilkan hasil pencarian dengan foto_kostum, nama_kostum, harga_sewa, rating.
7. Pelanggan dapat melihat detail kostum atau menambahkan ke *Wishlist*.

Sub Flows: -

Alternate / Exceptional Flows: -

Jumlah Transaksi: Disesuaikan dengan aktivitas pencarian pengguna terhadap kostum setiap harinya

Tabel 9. *Use Case Description* untuk Melakukan Pemesanan

Use Case Name : Melakukan Pemesanan ID: UC-06 Priority : Tinggi

Actor : Pelanggan

Description: Use case ini menggambarkan proses pemesanan kostum oleh pelanggan.

Trigger: Pelanggan ingin menyewa kostum yang dipilih.

Preconditions: Pelanggan sudah login, memilih kostum yang tersedia, dan KTP sudah terverifikasi.

Normal Course:

1. Pelanggan memilih tombol "Sewa" pada detail kostum.
2. Sistem menampilkan form pemesanan dengan field: tanggal_sewa, tanggal_kembali, catatan.
3. Pelanggan mengisi informasi pemesanan.
4. Sistem menghitung total_harga berdasarkan durasi sewa dan harga_sewa.
5. Pelanggan mengonfirmasi pesanan.
6. Sistem menyimpan data ke tabel pesanan dengan status_pembayaran = 'pending'.
7. Sistem menampilkan detail pesanan dan instruksi pembayaran.
8. Pelanggan mengunggah bukti_pembayaran.
9. Sistem memperbarui bukti_pembayaran di tabel pesanan.

Sub Flows:

Sistem menampilkan estimasi total harga secara otomatis saat tanggal dipilih

Sistem memvalidasi tanggal sewa agar tidak bertabrakan dengan pesanan lain

Alternate / Exceptional Flows:

Tanggal sewa tidak valid (tanggal kembali < tanggal sewa): Sistem tampilkan pesan *Error*

Kostum tidak tersedia: Sistem tampilkan pesan “Kostum tidak tersedia untuk tanggal tersebut”

Jumlah Transaksi: Disesuaikan dengan jumlah pesanan yang dilakukan pelanggan setiap hari.

Tabel 10. *Use Case Description* untuk Komunikasi dengan Penyewa

Use Case Name : Komunikasi Penyewa ID: UC-07 Priority : Tinggi

Actor : Pelanggan

Description: Use case ini menggambarkan proses komunikasi antara pelanggan dan penyewa melalui fitur chat.

Trigger: Pelanggan ingin berkomunikasi dengan penyewa kostum

Preconditions: Pelanggan sudah login dan memilih kostum atau memiliki pesanan aktif

Normal Course:

1. Pelanggan memilih tombol "Chat Penyewa" pada detail kostum atau pesanan.
2. Sistem membuka interface chat dengan penyewa yang bersangkutan.
3. Sistem menampilkan riwayat chat sebelumnya (jika ada) dari tabel chat.
4. Pelanggan mengetik pesan dan mengirim.
5. Sistem menyimpan pesan ke tabel chat dengan sender_id, receiver_id, pesan, waktu_kirim.
6. Sistem mengirim notifikasi real-time ke penyewa.
7. Penyewa dapat membalas pesan melalui interface yang sama.

Sub Flows: -

Alternate / Exceptional Flows: -

Jumlah Transaksi: Disesuaikan dengan intensitas komunikasi antara pelanggan dan penyewa setiap hari.

Tabel 11. *Use Case Description* untuk Rating dan Ulasan

Use Case Name : Rating dan Ulasan	ID: UC-08	Priority : Tinggi
--	-----------	-------------------

Actor : Pelanggan

Description: Use case ini menggambarkan proses pemberian rating dan ulasan oleh pelanggan setelah menyewa kostum.

Trigger: Pelanggan ingin memberikan rating setelah mengembalikan kostum.

Preconditions: Pelanggan sudah menyelesaikan peminjaman (status_peminjaman = 'completed').

Normal Course:

1. Pelanggan membuka halaman riwayat pesanan.

2. Sistem menampilkan daftar pesanan yang sudah selesai dan belum di-rating.
3. Pelanggan memilih pesanan untuk diberi rating.
4. Sistem menampilkan form rating dengan field: type (katalog/penyewa), nilai (1-5), komentar.
5. Pelanggan mengisi rating dan ulasan.
6. Pelanggan men-submit form rating.
7. Sistem menyimpan data ke tabel rating.
8. Sistem memperbarui rating dan jumlah_rating di tabel katalog atau penyewa.

Sub Flows: -

Alternate / Exceptional Flows: -

Jumlah Transaksi: Disesuaikan dengan jumlah penyewaan yang telah selesai dan diberi ulasan oleh pelanggan.

Tabel 12. *Use Case Description* untuk Manajemen Katalog

Use Case Name : Manajemen Katalog ID: UC-09 Priority : Tinggi

Actor : Penyewa

Description: Use case ini menggambarkan proses pengelolaan katalog kostum oleh penyewa.

Trigger: Penyewa ingin menambah, mengubah, atau menghapus kostum dari katalognya.

Preconditions: Penyewa sudah login dan memiliki akun terverifikasi.

Normal Course:

1. Penyewa memilih menu "Kelola Katalog" di dashboard.
2. Sistem menampilkan daftar kostum milik penyewa dari tabel katalog.
3. Untuk menambah: Penyewa memilih "Tambah Kostum".
4. Sistem menampilkan form dengan field: nama_kostum, series, karakter, ukuran, gender, deskripsi, kategori, harga_sewa, foto_kostum.
5. Penyewa mengisi form dan mengunggah foto kostum.
6. Penyewa men-submit form.

7. Sistem memvalidasi dan menyimpan data ke tabel katalog dengan status = 'pending'.
8. Untuk mengubah/menghapus: Penyewa memilih kostum dan melakukan aksi yang diinginkan.

Sub Flows: -

Alternate / Exceptional Flows: -

Jumlah Transaksi: Disesuaikan dengan aktivitas pengelolaan katalog oleh penyewa setiap harinya.

Tabel 13. *Use Case Description* untuk Manajemen Pesanan

Use Case Name : Manajemen Pesanan ID: UC-10 Priority : Tinggi

Actor : Penyewa

Description: Use case ini menggambarkan proses pengelolaan pesanan kostum oleh penyewa.

Trigger: Penyewa perlu mengelola pesanan kostum yang masuk.

Preconditions: Penyewa sudah login dan memiliki pesanan yang masuk.

Normal Course:

1. Penyewa memilih menu "Kelola Pesanan" di dashboard.
2. Sistem menampilkan daftar pesanan kostum milik penyewa dari tabel pesanan.
3. Penyewa memilih pesanan yang akan dikelola.
4. Sistem menampilkan detail pesanan dengan opsi aksi sesuai status.
5. Penyewa dapat mengonfirmasi pesanan, mengunggah bukti_peminjaman, atau mengubah status_peminjaman.
6. Sistem memperbarui data di tabel pesanan.
7. Sistem mengirim notifikasi ke pelanggan tentang perubahan status.

Sub Flows: -

Alternate / Exceptional Flows: -

Jumlah Transaksi: Disesuaikan dengan jumlah pesanan aktif yang dikelola oleh penyewa setiap hari.

Tabel 14. *Use Case Description* untuk Komunikasi dengan Pelanggan

Use Case Name : Komunikasi Pelanggan ID: UC-11 Priority : Tinggi

Actor : Penyewa

Description: Use case ini menggambarkan proses komunikasi antara penyewa dan pelanggan melalui fitur chat.

Trigger: Penyewa ingin berkomunikasi dengan pelanggan atau merespons pesan masuk.

Preconditions: Penyewa sudah login dan memiliki pesanan aktif atau pesan masuk.

Normal Course:

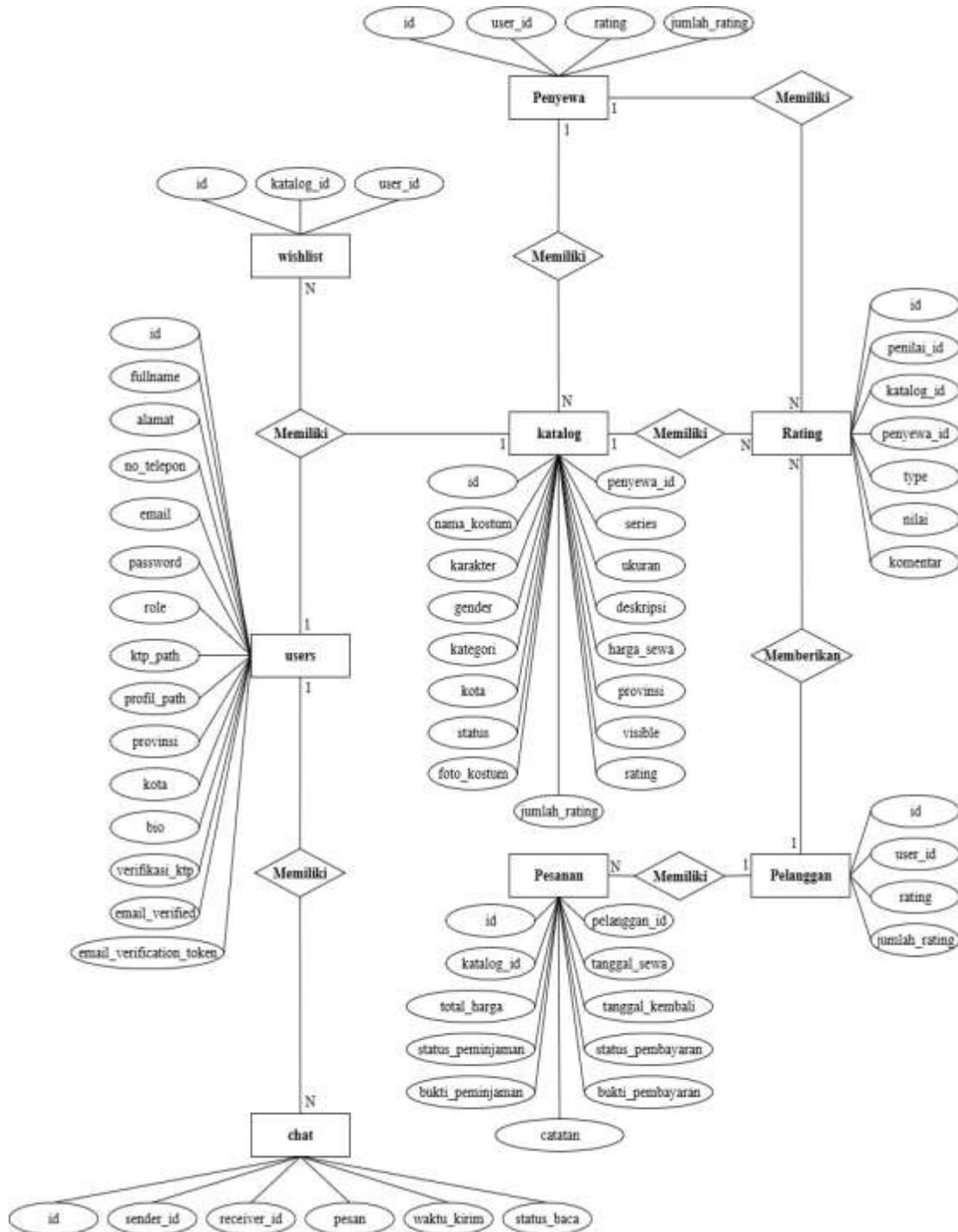
1. Penyewa memilih menu "Chat" atau menerima notifikasi pesan masuk.
2. Sistem menampilkan daftar percakapan dengan pelanggan dari tabel chat.
3. Penyewa memilih percakapan yang akan dibuka.
4. Sistem menampilkan interface chat dengan riwayat percakapan.
5. Penyewa mengetik balasan dan mengirim.
6. Sistem menyimpan pesan ke tabel chat dan mengirim notifikasi ke pelanggan.
7. Sistem mengubah status_baca pesan yang diterima menjadi 'read'

Sub Flows: -

Alternate / Exceptional Flows: -

Jumlah Transaksi: Disesuaikan dengan intensitas komunikasi antara penyewa dan pelanggan setiap harinya.

3.2.3 Entity Relationship Diagram (ERD)



Gambar 14. *Entity Relationship Diagram (ERD) Sistem Informasi Penyewaan Kostum Cosplay*

Entity Relationship Diagram (ERD) merupakan pemodelan yang menggambarkan

hubungan antar entitas dalam sistem informasi penyewaan kostum *cosplay*. ERD ini dirancang untuk mempermudah pemahaman aliran data, struktur database, dan keterkaitan antar komponen utama dalam sistem. Sistem ini melibatkan tiga jenis pengguna utama yaitu admin, penyewa, dan pelanggan.

Berikut penjelasan dari masing-masing entitas dan relasinya:

1. *Users*

Entitas ini menyimpan data semua pengguna yang terdaftar dalam sistem, baik sebagai admin, penyewa, maupun pelanggan. Atribut utama meliputi id, *Email*, dan role. Setiap pengguna dapat memiliki riwayat chat dan bisa berperan sebagai pengirim maupun penerima pesan.

2. Penyewa

Merupakan pengguna yang menyediakan kostum untuk disewakan. Entitas ini memiliki atribut id, user_id, dan rating. Satu penyewa dapat memiliki banyak katalog kostum. Penyewa juga bisa menerima rating dari pelanggan.

3. Pelanggan

Merupakan pengguna yang menyewa kostum. Entitas ini memiliki atribut id, user_id, dan rating. Pelanggan dapat melakukan pemesanan, memberikan rating, dan menambahkan kostum ke dalam *Wishlist*.

4. Katalog

Menyimpan daftar kostum yang ditawarkan oleh penyewa. Atribut utamanya adalah id, penyewa_id, dan nama_kostum. Setiap penyewa dapat memiliki lebih dari satu katalog. Katalog juga dapat menerima rating dan ditambahkan ke dalam *Wishlist* pelanggan.

5. Pesanan

Menyimpan data transaksi penyewaan kostum oleh pelanggan. Atribut utamanya adalah id dan pelanggan_id. Satu pelanggan dapat melakukan banyak pemesanan.

6. *Wishlist*

Fitur yang memungkinkan pelanggan menyimpan daftar kostum favorit. Setiap entri *Wishlist* terhubung ke user_id dan katalog_id. Seorang pengguna dapat

memiliki banyak *Wishlist*, dan satu kostum dapat masuk dalam *Wishlist* banyak pengguna.

7. Rating

Menyimpan penilaian dari pelanggan terhadap penyewa atau kostum. Atributnya meliputi id, penilai_id, dan type. Rating dapat diberikan oleh pelanggan kepada penyewa atau kepada kostum tertentu yang pernah disewa.

8. Chat

Menyimpan riwayat komunikasi antar pengguna, khususnya antara pelanggan dan penyewa. Atribut utamanya yaitu id, sender_id, dan receiver_id. Fitur ini mendukung interaksi langsung untuk membahas kostum atau transaksi.

Relasi Antar Entitas

1. Satu Users bisa menjadi Penyewa atau Pelanggan.
2. Satu Penyewa dapat memiliki banyak Katalog.
3. Satu Pelanggan dapat membuat banyak Pesanan dan Rating.
4. Satu Katalog dapat memiliki banyak Rating dan masuk ke banyak *Wishlist*.
5. Satu Users dapat mengirim dan menerima banyak Chat

3.2.4 Desain Interface

Desain *interface* adalah merancang tampilan dari *website* sistem informasi penyewaan yang dirancang agar tampilan *website* nyaman digunakan oleh *user* dan *admin* (Agustin et al, 2021). Berikut adalah rancangan desain yang dibuat sesuai dengan sistem informasi yang dibuat.

1. *Login dan Register*

Login : Memuat input *Email* dan *Password*, serta akses ke fitur reset *Password* atau registrasi akun baru.

Register : Nama lengkap, *Email*, kata sandi, alamat, kota, dan peran pengguna (penyewa/pelanggan).

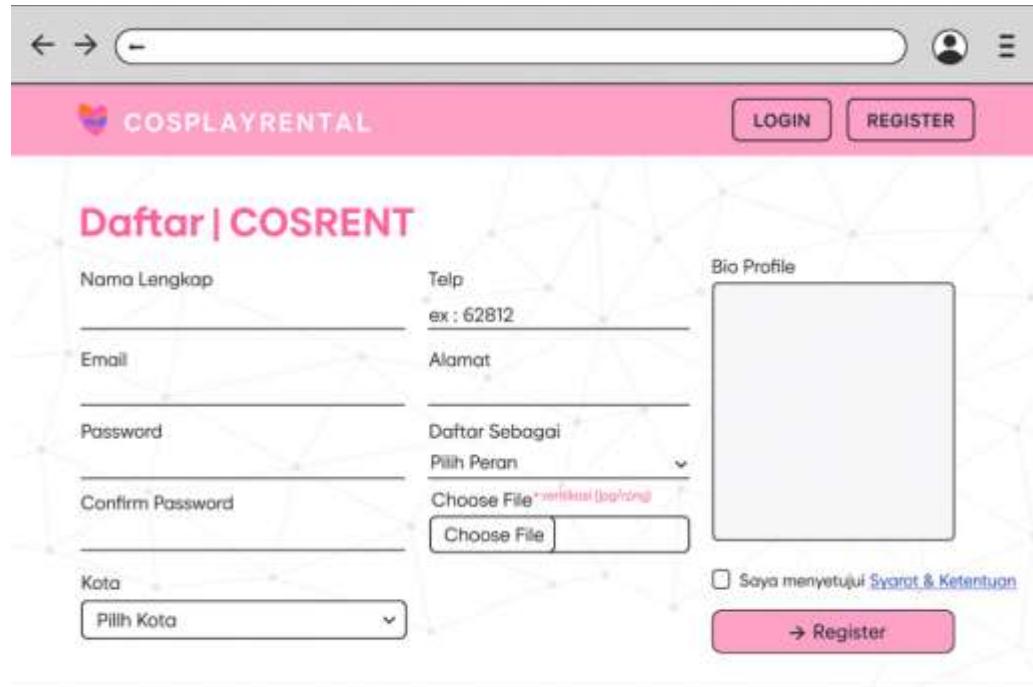
Fitur upload verifikasi (KTP) menambahkan unsur keamanan dan kepercayaan.

Checkbox syarat dan ketentuan menyesuaikan praktik privasi standar.

Desain mengikuti alur linear dan mudah diisi, penting untuk mempercepat proses onboarding pengguna. Gambar 15 dan 16 adalah rancangan desain dari *login* dan *register*.



Gambar 15. Desain Halaman Login



Gambar 16. Desain Halaman *Register*

2. Halaman Utama

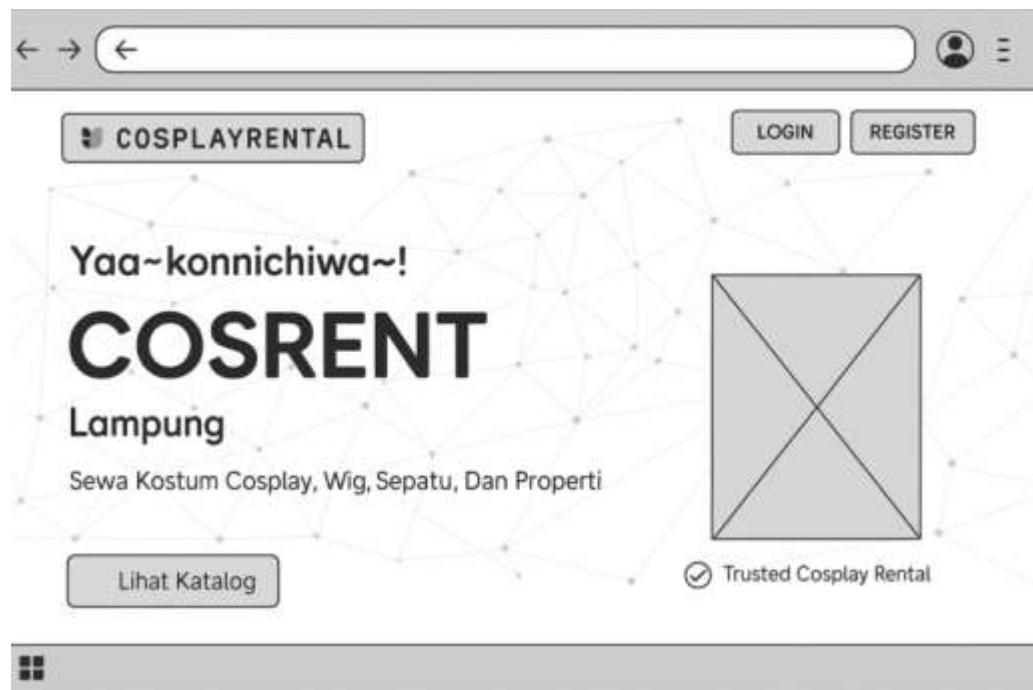
Halaman utama merupakan titik awal ketika pengguna pertama kali mengakses sistem *CosplayRental*. Halaman ini dirancang dengan pendekatan minimalis namun fungsional, dengan tujuan memberikan kesan pertama yang jelas dan navigasi yang langsung mengarahkan pengguna ke fitur inti dari sistem.

Salah satu elemen utama pada halaman ini adalah tombol “Lihat Katalog”, yang memungkinkan pengguna untuk langsung menjelajahi koleksi kostum yang tersedia tanpa perlu melakukan *login* terlebih dahulu. Hal ini memberikan pengalaman yang inklusif, di mana pengguna bisa terlebih dahulu mengeksplorasi konten sebelum memutuskan untuk membuat akun atau melakukan pemesanan. Strategi ini juga mampu meningkatkan rasa penasaran dan potensi konversi pengguna menjadi pelanggan aktif.

Selain itu, tampilan halaman ini menggunakan elemen visual sederhana, bersih, dan intuitif yang menekankan identitas merek “*CosplayRental*”. Dengan penempatan tombol aksi utama yang menonjol serta tampilan elemen desain yang rapi dan responsif, pengguna baru dapat dengan mudah memahami tujuan dari aplikasi ini serta cara menggunakannya tanpa mengalami kebingungan. Hal ini

juga sangat penting untuk meningkatkan *user experience* (UX) dan mengurangi tingkat bounce rate pada tahap awal kunjungan pengguna ke sistem.

Secara keseluruhan, halaman ini tidak hanya berfungsi sebagai gerbang masuk aplikasi, tetapi juga sebagai media promosi awal untuk menarik minat pengguna melalui penyajian desain antarmuka yang ramah, fungsional, dan fokus pada konversi ke fitur utama. Gambar 17 adalah rancangan dari desain halaman awal.



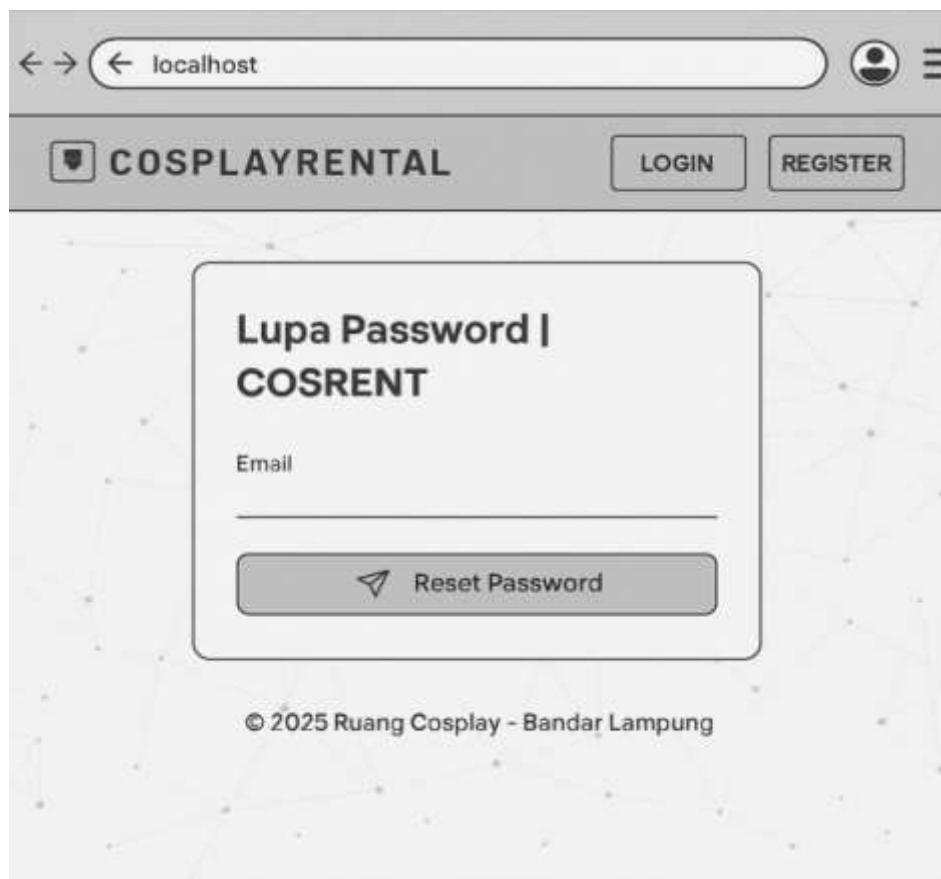
Gambar 17. Rancangan Desain Halaman Awal

3. Lupa Password dan Reset Password

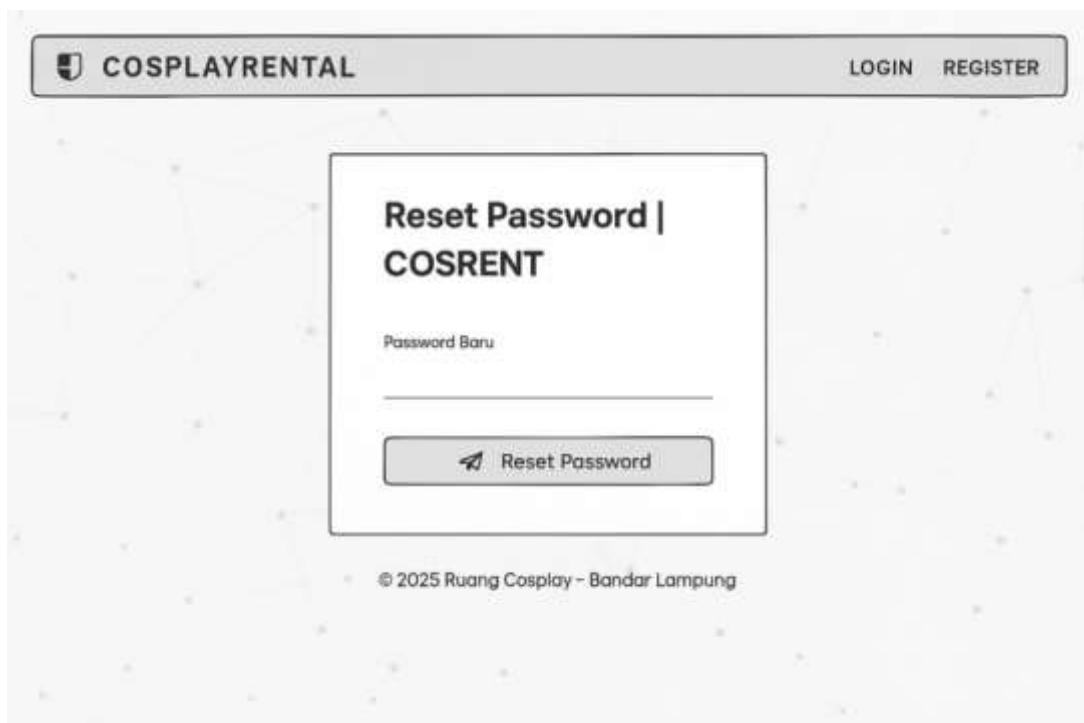
Lupa Password: masukkan *Email* untuk menerima tautan pengaturan ulang.

Reset Password: masukkan kata sandi baru setelah menerima tautan.

Proses ini mendukung aspek keamanan dan kenyamanan pengguna dalam mengakses kembali akunnya. Gambar 18 dan 19 adalah tampilan rancangan desain.



Gambar 18. Rancangan Desain Lupa Password



Gambar 19. Rancangan Desain Reset Password

4. Katalog dan Detail

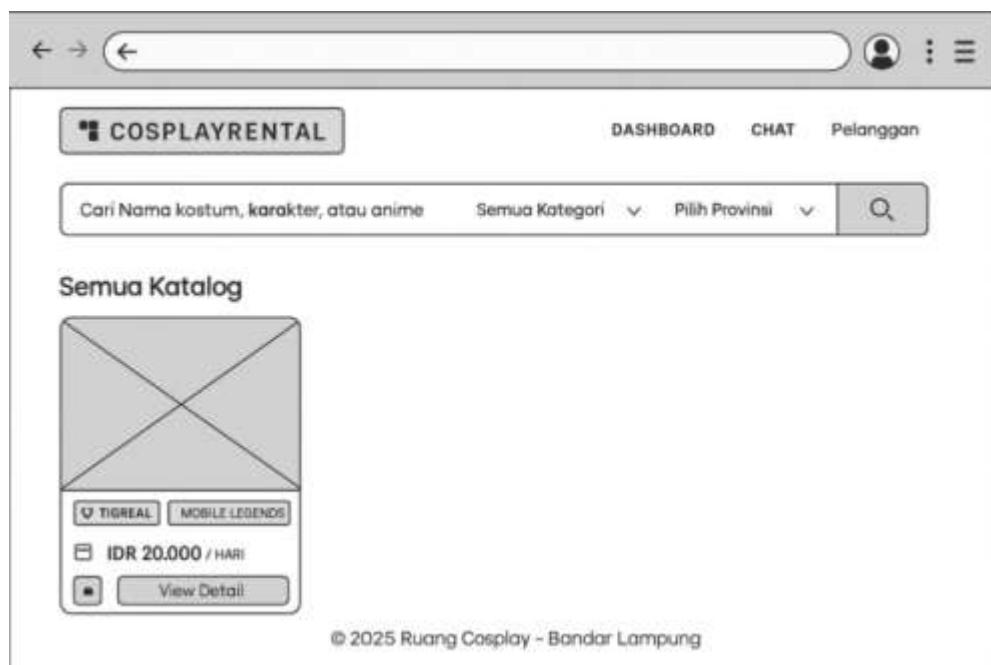
Katalog: Pelanggan dapat mencari kostum berdasarkan kategori, karakter, atau lokasi dan disediakan *filter dropdown* dan kolom pencarian.

Detail Katalog:

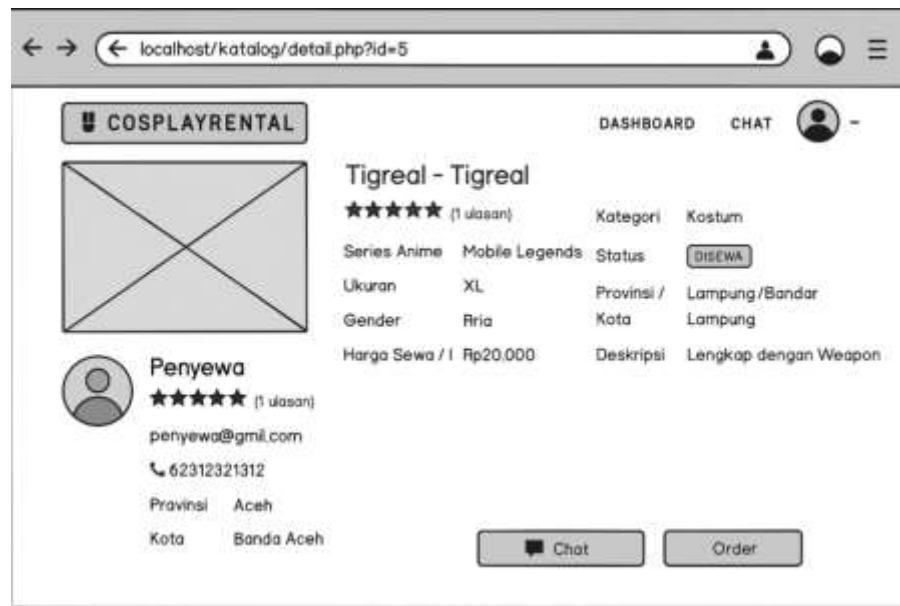
Menampilkan spesifikasi lengkap kostum, informasi harga, ukuran, lokasi, serta status ketersediaan (disewa/tersedia) dan dilengkapi dengan profil pemilik kostum (penyewa), *Email*, lokasi, dan rating.

Tersedia dua aksi: Chat (untuk bertanya langsung ke penyewa) dan Order.

Gambar 20 dan 21 adalah rancangan dari katalog dan detail



Gambar 20. Rancangan Desain Katalog



Gambar 21. Rancangan Desain Detail

5. Pemesanan

Diakses melalui halaman detail katalog.

Pelanggan mengisi: Tanggal sewa & tanggal kembali, unggah bukti pembayaran, catatan, dan deskripsi tambahan, dan harga akan dihitung otomatis durasi sewa.

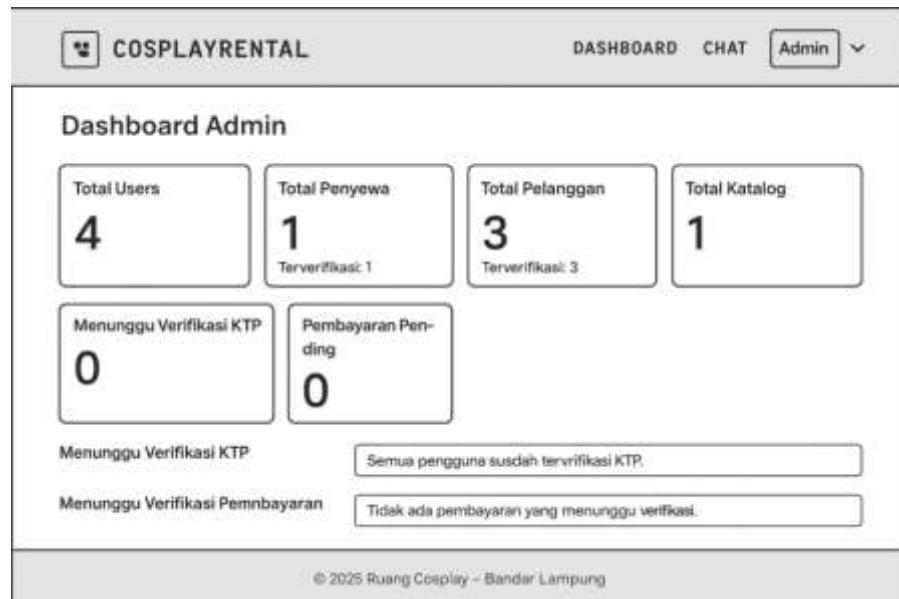
Tujuan: menyederhanakan proses penyewaan dengan sistem konfirmasi otomatis.

Gambar 22 adalah rancangan desain dari pemesanan

Gambar 22. Rancangan Desain *Order*

6. Dashboard Admin

Halaman *Dashboard* Admin berfungsi sebagai pusat kontrol utama bagi administrator sistem *Cosplay Rental*. Dari halaman ini, admin dapat memantau dan mengelola seluruh data pengguna, status verifikasi KTP, status pembayaran, serta jumlah katalog kostum yang tersedia di sistem. Gambar 23 adalah rancangan desain dari *dashboard* admin

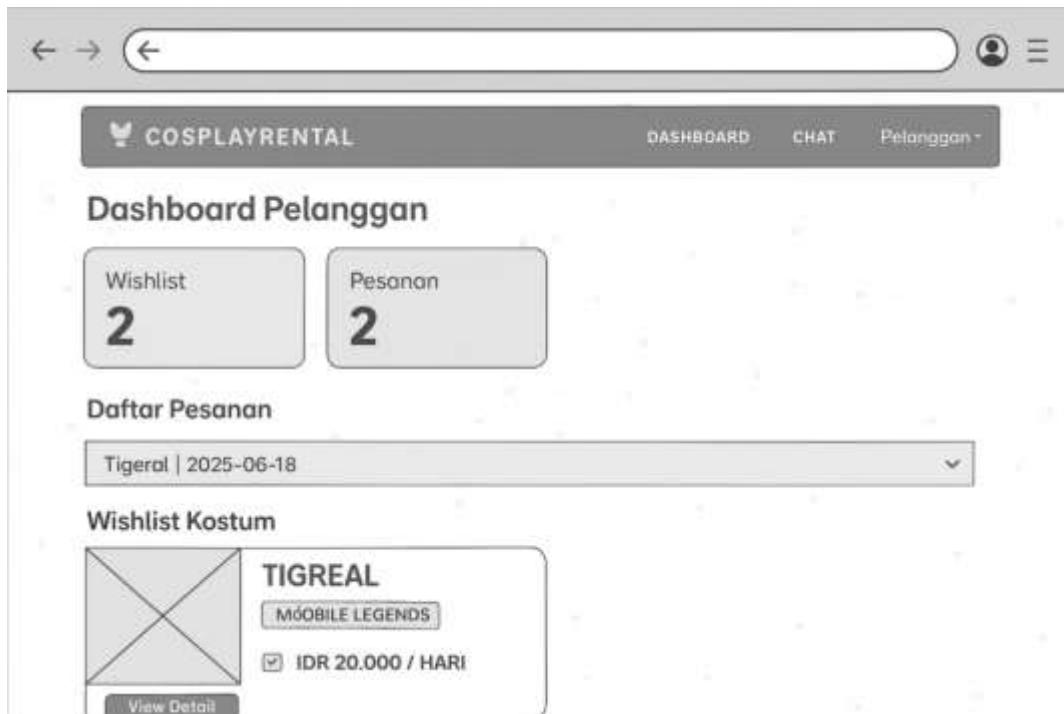


Gambar 23. Rancangan Desain *Dashboard* Admin

7. Dashboard Pelanggan

Menampilkan informasi utama seperti: Jumlah *Wishlist* dan pesanan aktif, daftar pesanan ditampilkan sesuai tanggal, dan *Wishlist* kostum ditampilkan dalam bentuk kartu, memudahkan pelanggan melacak keinginan mereka.

Menu navigasi tetap (*dashboard*, *chat*, *pelanggan*) menjaga konsistensi pengalaman pengguna. Gambar 24 adalah rancangan desain dari *dashboard* pelanggan.

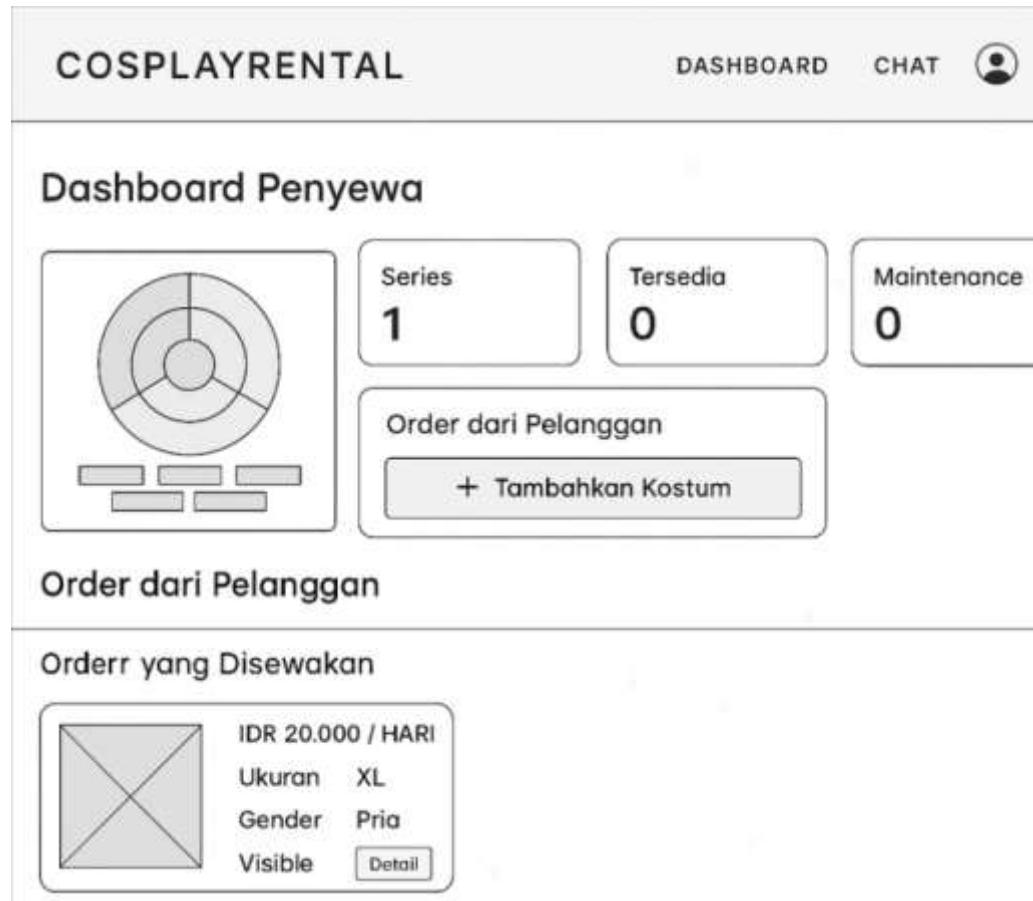


Gambar 24. Rancangan Desain *Dashboard* Pelanggan

8. Dashboard Penyewa

Fitur: Statistik jumlah katalog, ketersediaan, dan item *maintenance*, tombol untuk menambahkan kostum baru ke katalog, daftar kostum yang disewakan lengkap dengan info harga, ukuran, dan status visibilitas.

Penyewa dapat mengelola katalog dan memantau permintaan dari pelanggan secara real-time. Gambar 25 adalah rancangan desain dari *dashboard* penyewa.



Gambar 25. Rancangan Desain *Dashboard* Penyewa

9. Edit Profil

Halaman "Edit Profil" pada sistem *CosplayRental* dirancang untuk memberikan kemudahan kepada pengguna (penyewa) dalam memperbarui data pribadi secara mandiri. Halaman ini memiliki peran penting dalam memastikan keakuratan dan kevalidan data pengguna untuk mendukung proses sewa menyewa kostum secara aman dan transparan. Gambar 26 adalah tampilan dari rancangan desain edit profil

Gambar 26. Rancangan Desain Edit Profil

3.2.5 Cutover

- Pengujian Sistem

1. Alpha Testing

Salah satu bentuk penerapan dari black box testing adalah alpha testing. Tujuan dari metode ini adalah untuk memastikan bahwa setiap fitur yang dikembangkan berfungsi sesuai dengan spesifikasinya dan bebas dari kesalahan. Pengujian ini dilakukan oleh pengembang dengan menguji setiap fungsionalitas fitur yang telah dibuat untuk memastikan semuanya berjalan dengan baik.

Tabel 15. Skenario pengujian Autentikasi

No	Kelas Uji	Daftar Pengujian	Hasil yang Didapat
1	Login	Halaman Login Input <i>email</i> dan <i>password</i> kosong	Menampilkan halaman login Tidak berhasil login
		Input <i>email</i> atau <i>password</i> kosong	Tidak berhasil login
		Input <i>email</i> atau <i>password</i> Tidak berhasil login	

No	Kelas Uji	Daftar Pengujian	Hasil yang Didapat
2	Register	<p>salah</p> <p>Input <i>email</i> dan <i>password</i> Masuk ke dalam sistem / dashboard valid pengguna</p> <p>Akses halaman <i>register</i> Menampilkan form pendaftaran Mengisi form <i>register</i> kosong atau data tidak Muncul pesan <i>Error</i> validasi valid</p> <p>Mengisi form <i>register</i> Sistem mengirim <i>Email</i> verifikasi ke dengan data valid <i>Mailtrap</i></p> <p>Klik link verifikasi dari Akun berhasil diverifikasi, dapat <i>Email Mailtrap</i> login</p>	
3	Lupa Password	<p>Klik tautan “Lupa Menampilkan halaman form lupa Password”</p> <p>Input <i>Email</i> yang tidak Muncul pesan <i>Error</i> “<i>Email</i> tidak terdaftar ditemukan”</p> <p>Input <i>Email</i> yang terdaftar Sistem mengirim <i>Email</i> reset <i>Password</i> ke <i>Mailtrap</i></p> <p>Klik link reset <i>Password</i> Menampilkan halaman input dari <i>Mailtrap</i> <i>Password</i> baru</p> <p>Input <i>Password</i> baru dan <i>Password</i> berhasil diperbarui, pengguna diarahkan ke halaman konfirmasi <i>Password</i> login</p>	

Tabel 16. Skenario Pengujian Fitur Tambah Kostum

No	Kelas Uji	Daftar Pengujian	Hasil yang Didapat
1	Tambah Kostum	Form input lengkap dan Data kostum berhasil disimpan, valid notifikasi sukses	
2	Tambah	Form input kosong atau Muncul pesan peringatan	

No Kelas Uji	Daftar Pengujian	Hasil yang Didapat
Kostum	tidak lengkap	kesalahan (<i>Error</i>)

Tabel 17. Skenario Pengujian Fitur Pemesanan Kostum

No Kelas Uji	Daftar Pengujian	Hasil yang Didapat
1 Pemesanan Kostum	Memilih kostum dan mengisi form sewa	Pemesanan berhasil dan data tercatat di sistem
2 Pemesanan Kostum	Tidak memilih kostum	Muncul pesan <i>Error</i> karena input tidak valid

Tabel 18. Skenario Pengujian *Wishlist*

No Kelas Uji	Daftar Pengujian	Hasil yang Didapat
1 <i>Wishlist</i>	Klik "Tambah ke Kostum ditambahkan ke daftar favorit <i>Wishlist</i> " pengguna	
2 <i>Wishlist</i>	Klik "Hapus dari Kostum dihapus dari daftar favorit <i>Wishlist</i> "	

Tabel 19. Skenario Pengujian Rating dan Ulasan

No Kelas Uji	Daftar Pengujian	Hasil yang Didapat
1 Rating Ulasan	dan Pelanggan memberikan rating	Rating berhasil disimpan dan agregat diperbarui

Tabel 20. Skenario Pengujian *Chat*

No Kelas Uji	Daftar Pengujian	Hasil yang Didapat
1 Chat	Pengguna membuka	Menampilkan riwayat pesan

No Uji	Kelas	Daftar Pengujian	Hasil yang Didapat
halaman chat			
2 Chat		Mengirim pesan pengguna lain	ke Pesan terkirim, tampil di riwayat, status terbaca

2. *User Acceptance Test* (UAT)

Setelah pengujian sistem dilakukan oleh tim pengembang menggunakan metode *Black Box Testing* untuk memastikan bahwa semua fungsi sistem berjalan sesuai dengan tujuannya, maka tahap selanjutnya adalah melakukan pengujian *User Acceptance Test* (UAT). Pengujian ini bertujuan untuk mengetahui tingkat kepuasan pengguna terhadap sistem informasi penyewaan kostum *cosplay* karakter anime berbasis *website* di Bandar Lampung.

Metode pengujian UAT dilakukan dengan membagikan kuesioner kepada pengguna akhir sistem, seperti calon pelanggan dan penyewa, yang telah mencoba fitur-fitur pada sistem. Kuesioner ini menggunakan skala Likert dengan lima tingkatan jawaban:

Tabel 21. Bobot Skor Skala Likert

Jawaban	Skor/Bobot
Sangat Setuju (SS)	5
Setuju (S)	4
Netral (N)	3
Tidak Setuju (TS)	2
Sangat Tidak Setuju (STS)	1

Setelah pengguna mengisi kuesioner, dilakukan perhitungan untuk memperoleh nilai indeks kepuasan pengguna menggunakan rumus berikut (Riyadi, 2019):

$$\text{IKP}(\%) = \frac{\text{Skor maksimal}}{\text{Total skor yang diperoleh}} \times 100$$

Selanjutnya, hasil indeks kepuasan pengguna tersebut dibandingkan dengan kriteria evaluasi yang terdapat dalam tabel berikut

Tabel 22. Kriteria Indeks Kepuasan Pengguna

Rentang Indeks (%)	Keterangan
0% – 19,99%	Sangat Tidak Memuaskan
20% – 39,99%	Tidak Memuaskan
40% – 59,99%	Kurang Memuaskan
60% – 79,99%	Memuaskan
80% – 100%	Sangat Memuaskan

Jika hasil perhitungan indeks menunjukkan nilai di bawah 60%, maka sistem perlu dilakukan perbaikan lebih lanjut karena belum memenuhi ekspektasi pengguna. Namun, apabila nilai indeks berada pada rentang 60% hingga 100%, maka sistem dianggap layak digunakan dan dapat dilanjutkan ke tahap *cutover* atau serah terima kepada *stakeholder*.

V. KESIMPULAN DAN SARAN

5.1 Pengujian Sistem

Berdasarkan hasil penelitian, analisis kebutuhan sistem, proses pengembangan, serta pengujian menggunakan metode black box, maka dapat diambil beberapa kesimpulan sebagai berikut: Sistem informasi penyewaan kostum *cosplay* berbasis *website* berhasil dibangun dan diimplementasikan dengan fitur yang sesuai dengan kebutuhan pengguna, baik pelanggan, penyewa, maupun admin. Sistem ini memberikan solusi digital yang menggantikan proses penyewaan kostum yang sebelumnya dilakukan secara manual. Sistem ini menyediakan fitur utama yang lengkap meliputi, pendaftaran dan login pengguna berdasarkan peran (pelanggan/penyewa), manajemen katalog kostum oleh penyewa, pencarian dan pemesanan kostum oleh pelanggan, unggah bukti pembayaran dan proses verifikasi oleh admin, komunikasi melalui fitur *chat* internal, pengelolaan profil pengguna dan verifikasi KTP.

Desain sistem dibuat secara responsif, sehingga dapat diakses dengan baik pada perangkat desktop maupun mobile, serta memiliki antarmuka yang sederhana dan mudah dipahami oleh pengguna. Pengujian sistem menggunakan metode *black box testing* menunjukkan bahwa seluruh fungsi utama dapat berjalan dengan baik, menghasilkan output yang sesuai dengan input yang diberikan. Sistem mampu menangani validasi, kesalahan *input*, serta memberikan feedback yang sesuai. Sistem ini berkontribusi dalam meningkatkan efisiensi proses operasional penyewaan kostum, meminimalisir kesalahan pencatatan, mempercepat proses transaksi, serta memberikan pengalaman yang lebih baik kepada pelanggan dan penyedia jasa di wilayah Bandar Lampung khususnya.

5.2 Saran

Meskipun sistem telah berjalan baik, berikut beberapa saran untuk pengembangan lebih lanjut agar sistem dapat lebih optimal dan adaptif terhadap kebutuhan masa depan:

1. Integrasi sistem pembayaran otomatis, seperti QRIS atau payment gateway, agar proses pembayaran lebih efisien dan real-time.
2. Penambahan fitur rating dan ulasan terhadap kostum maupun penyewa, sehingga pelanggan bisa memberikan penilaian sebagai referensi bagi pengguna lainnya.
3. Peningkatan keamanan sistem, melalui penggunaan teknologi token CSRF, OTP untuk reset *Password*, serta enkripsi file upload (KTP, bukti pembayaran).
4. Implementasi notifikasi real-time, baik melalui *Email*, notifikasi dalam sistem, atau integrasi WhatsApp API untuk status pesanan, verifikasi, dan komunikasi cepat.
5. Pengembangan sistem pencarian berbasis gambar menggunakan teknologi AI (Image-Based Search) dengan fitur ini, pengguna cukup mengunggah gambar karakter atau referensi *cosplay*, lalu sistem secara otomatis menampilkan kostum paling relevan dari katalog. Teknologi ini akan meningkatkan kenyamanan dan akurasi pencarian terutama bagi pengguna yang tidak mengetahui nama karakter secara spesifik.
6. Pembuatan versi mobile app atau Progressive Web App (PWA) untuk menjangkau lebih banyak pengguna dan mendukung pengalaman yang lebih fleksibel di perangkat seluler.
7. Penyusunan dokumentasi API untuk membuka peluang integrasi dengan layanan eksternal seperti ekspedisi, e-wallet, atau komunitas *cosplay* lokal.

DAFTAR PUSTAKA

- Agustin, R., Hidayat, T., & Sulaiman, D. (2021). Desain antarmuka pengguna pada sistem informasi penyewaan kostum *cosplay* berbasis web. *Jurnal Teknologi Informasi*, 12(3), 98–105.
- Alfarizi, M. (2024). Penerapan Mailtrap dalam sistem pengujian *Email* berbasis cloud. *Jurnal Teknologi dan Informasi*, 6(1), 45–53.
- Mailtrap.io. (2024). *Mailtrap documentation*. <https://mailtrap.io>
- Mytaros, D. A. (2021). *Manajemen penyewaan barang dan jasa dalam ekonomi digital*. Jakarta: Media Informatika.
- Nauli, R. P. (2020). Pengembangan sistem informasi berbasis *website* untuk layanan penyewaan kostum. *Jurnal Sistem Informasi dan Teknologi*, 8(2), 115–122.
- Nurfauziah, L., Rahmat, D., & Arsyad, M. A. (2022). Penerapan black box testing pada pengujian perangkat lunak berbasis web. *Jurnal Rekayasa Perangkat Lunak*, 10(1), 32–39.
- O'Brien, J. A., & Marakas, G. M. (2011). *Introduction to information systems* (15th ed.). New York: McGraw-Hill.
- Pahrizal, H., Wulandari, N., & Lestari, S. (2022). Implementasi PHPMailer dalam sistem informasi pengiriman *Email* otomatis. *Jurnal Sistem Informasi dan Aplikasi*, 9(3), 87–93.
- Paramita, R. S., Nugroho, T. A., & Hasanah, I. (2025). Analisis perkembangan budaya *cosplay* di Indonesia. *Jurnal Budaya Populer*, 12(1), 15–26.
- Putra, A. R. (2019). Strategi pemasaran layanan penyewaan kostum karakter di era digital. *Jurnal Ekonomi Kreatif*, 7(1), 25–33.

- Putra, A. R. (2020). Analisis kebutuhan dalam pengembangan sistem penyewaan kostum *cosplay*. *Jurnal Sistem dan Teknologi Informasi*, 9(2), 77–84.
- Putra, A. R., Lestari, S., & Nugroho, T. A. (2020). Pengembangan sistem informasi penyewaan kostum berbasis web untuk pengelolaan inventaris dan transaksi. *Jurnal Manajemen Sistem Informasi*, 15(1), 112–118.
- Putri, M. L. (2024). Optimasi manajemen penyewaan berbasis sistem digital. *Jurnal Manajemen Informatika*, 5(2), 88–96.
- Satria, Y. A., Nuraini, D., & Hidayat, R. (2023). Penggunaan bahasa pemrograman dalam pengembangan sistem informasi web. *Jurnal Teknologi Informasi dan Komputer*, 11(4), 73–82.
- Sudana, I. G. P., Prasetyo, R., & Mahendra, A. (2019). Pemanfaatan PHPMailer dalam sistem notifikasi *Email* berbasis PHP. *Jurnal Ilmu Komputer*, 5(2), 101–108.
- Yudhistira, F., & Endah, M. P. (2024). Penerapan use case diagram dalam desain sistem informasi penyewaan kostum *cosplay*. *Jurnal Desain Sistem*, 14(2), 44–50.