

**SISTEM INFORMASI PEMBELIAN TIKET WISATA TAMAN
UJUNG SUKASADA KARANGASEM BALI BERBASIS
WEBSITE**

SKRIPSI



Oleh:

AHMAD HUDAY
2021502003

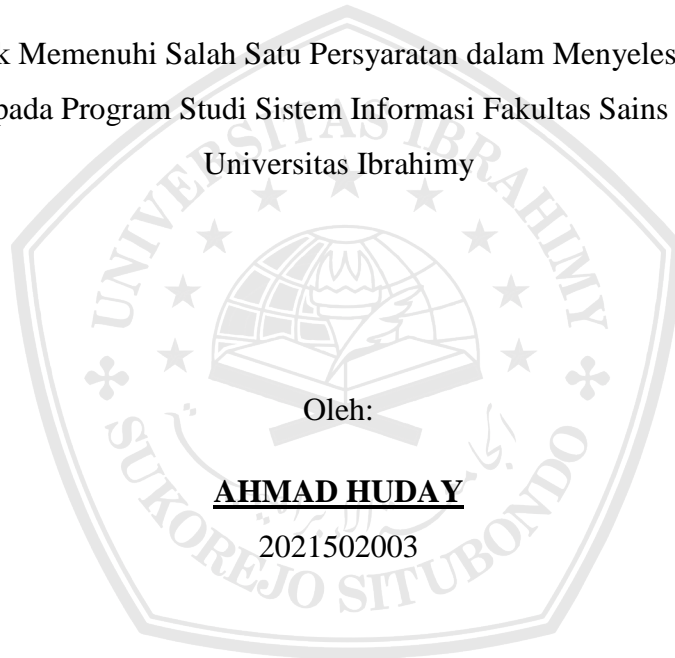
**PROGRAM STUDI SI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMY
SITUBONDO**

2025

**SISTEM INFORMASI PEMBELIAN TIKET WISATA TAMAN
UJUNG SUKASADA KARANGASEM BALI BERBASIS
WEBSITE**

SKRIPSI

Diajukan untuk Memenuhi Salah Satu Persyaratan dalam Menyelesaikan Program Sarjana (S-1) pada Program Studi Sistem Informasi Fakultas Sains dan Teknologi Universitas Ibrahimi



Oleh:

AHMAD HUDAY

2021502003

**PROGRAM STUDI SI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI UNIVERSITAS IBRAHIMI
SITUBONDO**

2025

PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini :

Nama : **Ahmad Hiday**
NPM : 2021502003
Program Studi : Sistem Informasi
Fakultas : Fakultas Sains dan Teknologi

Menyatakan dengan sebenarnya, bahwa tugas akhir/skripsi ini secara keseluruhan adalah hasil penelitian atau karya saya sendiri, kecuali pada bagian-bagian yang dirujuk sebagai sumber referensi dan disebutkan dalam daftar pustaka. Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa tugas akhir/skripsi ini hasil plagiasi, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Situbondo, 03 Juli 2025

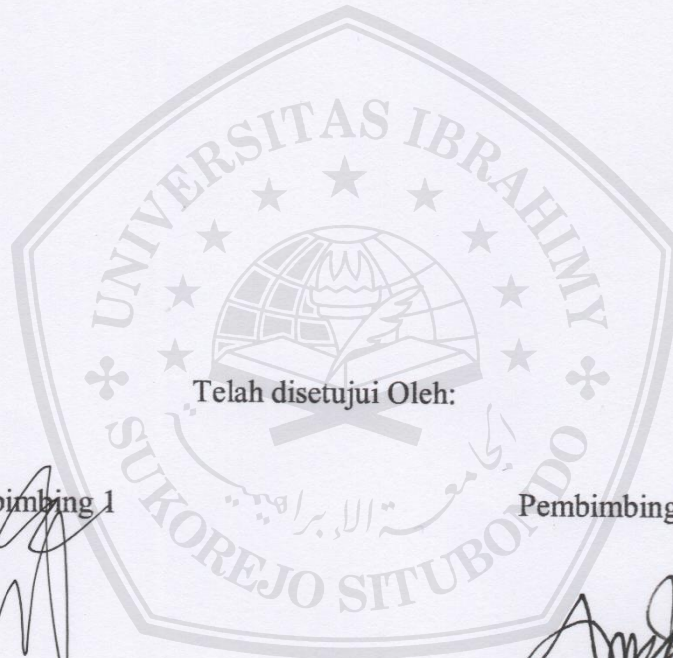
Saya yang menyatakan



Ahmad Hiday

PÉRSETUJUAN PEMBIMBING

Nama : Ahmad Hiday
NPM : 2021502003
Judul : **Sistem Informasi Pembelian Tiket Wisata Taman Ujung Sukasada
Karangasem Bali.**



Pembimbing 1

Achmad Bajjuri, M.Kom

NIDN : 0715078902

Pembimbing 2

A. Namdani, M.Kom

NIDN : 0730118806

HALAMAN PENGESAHAN

SKRIPSI

**SISTEM INFORMASI PEMBELIAN TIKET WISATA TAMAN
UJUNG SUKASADA KARANGASEM BALI BERBASIS
WEBSITE**

AHMAD HUDAY

2021502003

telah dipertahankan di depan dewan penguji Sidang/Munaqasyah Skripsi pada hari Kamis, Tanggal 24 Juli 2024 sebagai salah satu syarat memperoleh gelar Sarjana (S.Kom) pada Fakultas Sains dan Teknologi Universitas Ibrahimiy.

Tim Penguji,

Ketua Sidang,

Sekretaris Sidang,

Firman Santoso, M.Kom
NIDN : 0722129201

Faigo Sabut Panotogo, A.Md. Pi
NIDN : -

Penguji I,

Penguji II,

Zaehol Fatah, M.Kom
NIDN : 0715057801

Abd. Ghofur, M.Kom
NIDN : 0711088303

Mengetahui
Dekan,

Abd. Ghofur, M.Kom
NIDN : 0711088303

MOTTO

“JADIKANLAH SABAR DAN SHALAT SEBAGAI PENOLONGMU”

(QS. Al-Baqarah: 153)



HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahim, Allahumma Sholli A'la Sayyidina Muhammad.

Puji dan puji syukur kepada Allah yang maha segalanya, Tugas Akhir Skripsi ini saya persembahkan kepada:

1. Para Masyayikh Pondok Pesantren Salafiyah Syafi'iyah Sukorejo Situbondo wabilhusus KHR. Ach. Azaim Ibrahimi, S.Sy, M.H, beserta ahlul bait yang setiap harinya mendo'akan dan mencintai santrinya.
2. Kepada Bapak, Sang Ksatria Tak Berkuda dan kepada Ibu, Sang Bidadari Tak Bersayap yang selalu mendukung saya.
3. Saudara-saudaraku yang memberikan semangat dengan memberikan pengalaman-pengalamannya.
4. Dosen pembimbing 1 “Bapak. Achmad Baijuri, M.Kom”, dan pembimbing 2 “Bapak. A. Hamdani, M.Kom” beserta seluruh dosen Univertas Ibrahimi.
5. Teman-teman Aliansi Santri Bajang Karangasem tempat berbagi cerita baik suka maupun duka.
6. Kepada keluarga kecil Asrama Sunan Ampel bagian atas terutama Asrama Sunan Ampel No.25 yang sudah saya anggap saudara.
7. Teman seperjuangan Prodi Sistem Informasi Angkatan 2021, telah menemani susah senang dalam membuat tugas perkuliahan.
8. Seseorang yang Allah tulis di Lauhul Mahfuz.

KATA PENGANTAR

Segala puji syukur peneliti sampaikan kepada Allah SWT, karena atas Rahmat dan Hidayah-Nya, perencanaan, pelaksanaan dan penyelesaian tugas akhir/skripsi dengan judul “Sistem Informasi Pembelian Tiket Wisata Taman Ujung Sukasada Karangasem Bali.” sebagai salah satu syarat penyelesaian program diploma/sarjana dapat terselesaikan dengan baik dan lancar.

Kesuksesan ini dapat peneliti peroleh karena dukungan beberapa pihak. Peneliti menyampaikan terima kasih kepada:

1. KHR. Ach. Azaim Ibrahimi selaku Pengasuh Pondok Pesantren Salafiyah Syafi'iyah
2. KH. Ahmad Fadlail, MH. selaku Rektor Universitas Ibrahimi.
3. Bapak Abd. Ghofur, M.Kom. selaku Dekan Fakultas Sains dan Teknologi.
4. Bapak Achmad Baijuri, M.Kom. selaku Ketua Program Studi Sistem Informasi.
5. Bapak. Achmad Baijuri, M.Kom. dan Bapak A. Hamdani, M.Kom. selaku pembimbing I dan II, dan.
6. Jajaran staf dan karyawan Taman Sukasada Ujung Karangasem Bali yang sudah mengizinkan untuk melakukan penelitian.

Semoga semua amal baik yang telah diberikan oleh Bapak/Ibu kepada peneliti mendapat balasan yang sebaik mungkin dari Allah SWT, Amin.

Situbondo, 03 Juli 2025

Peneliti

ABSTRAK

Ahmad Hiday. 2025. **Sistem Informasi Pembelian Tiket Wisata Taman Ujung Sukasada Karangasem Bali**. Skripsi. Program Studi Sistem Informasi. Universitas Ibrahimi. Pembimbing: (I) Achmad Baijuri, M.Kom., (II) A.Hamdani, M.Kom.

Sistem informasi pembelian tiket *online* ini bertujuan untuk merancang sistem yang memuat pembelian tiket secara *online* dan laporan keuangan secara otomatis berbasis website di Taman Sukasada Ujung Karangasem Bali. Dalam pembelian tiket di taman ini masih dilakukan secara manual dan juga pelaporan keuangan pembelian tiket masih manual dengan menulis di kertas dan dicatat ulang lagi ke *excel*. Keterbatasan pada sistem ini mengakibatkan antrian ketika pembelian tiket dan juga mungkin akan terjadi kesalahan dalam pencatatan pembelian tiket. Oleh karena itu, pengembangan sistem ini diharapkan akan memberikan solusi yang efektif dan efisien bagi taman dalam mengelola pembelian tiket dan laporan keuangannya. Proses pengembangan sistem ini menggunakan metode *waterfall*. Dalam tahap pengkodean atau pengimplementasian desain ke dalam suatu program yang akan dibuat dalam bahasa *PHP*. Pengujian sistem yang digunakan adalah sistem pengujian *Self-Testing / Developer Testing* dimana dari perhitungan tingkat keberhasilan dari total *case* dengan jumlah *test case* yang dilakukan sebesar 95%. Dengan ada sistem ini akan terjadi percepatan sebesar 50% atau dua kali lipat dalam pembelian tiket yang sebelumnya harus melakukan pembelian dan pengecekan tiket di satu tempat, sekarang pembelian bisa dilakukan dimana saja dan di loket cukup mengecek kode *QR* dengan sistem *scan*. Dengan hal ini, Taman Sukasada Ujung bisa memberikan kenyamanan kepada para pengunjung dalam segi antrian pembelian tiket, dan juga mempermudah dalam laporan keuangan pembelian tiket.

Kata Kunci: Sistem Informasi, *Waterfall*, *PHP*, Sukasada, Karangasem.

TRANSLITERASI

Ahmad Hiday. 2025. *Information System for Purchasing Tickets for Ujung Sukasada Park Tourism Karangasem Bali*. Thesis. Information Systems Study Program. Ibrahimi University. Advisors: (I) Achmad Baijuri, M.Kom., (II) A.Hamdani, M.Kom.

This online ticket purchase information system aims to design a system that includes online ticket purchases and automatic financial reports based on the website at Taman Sukasada Ujung Karangasem Bali. Ticket purchases at this park are still done manually and also the financial reporting of ticket purchases is still manual by writing on paper and re-recording it in excel. The limitations of this system result in queues when purchasing tickets and there may also be errors in recording ticket purchases. Therefore, the development of this system is expected to provide an effective and efficient solution for the park in managing ticket purchases and financial reports. The process of developing this system uses the waterfall method. In the coding or implementation stage of the design into a program that will be created in PHP. The system testing used is the Self-Testing / Developer Testing testing system where the calculation of the success rate of the total case with the number of test cases carried out is 95%. With this system, there will be an acceleration of 50% or two times in ticket purchases which previously had to make purchases and check tickets in one place, now purchases can be made anywhere and at the counter simply by checking the QR code with the scan system. With this, Sukasada Ujung Park can provide convenience to visitors in terms of ticket purchase queues, and also facilitate accurate purchase financial reports.

Keywords: *Information System, Waterfall, PHP, Sukasada, Karangasem*

DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN.....	ii
PERSETUJUAN PEMBIMBING	iii
HALAMAN PENGESAHAN.....	iv
MOTTO	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR.....	vii
ABSTRAK	viii
TRANSLITERASI.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	2
1.3 Rumusan Masalah	3
1.4 Batasan Masalah.....	3
1.5 Tujuan Penelitian.....	3
1.6 Manfaat Penelitian.....	4

1.7 Metode Penelitian.....	4
1.7.1 Jenis Penelitian.....	4
1.7.2 Teknik Pengumpulan Data.....	4
1.7.3 Metode Pengembangan Sistem	5
1.8 Sistematika Pembahasan	8
BAB II TINJAUAN PUSTAKA.....	10
2.1 Penelitian Terdahulu	10
2.2 Landasan Teori.....	13
2.2.1 Pengertian Sistem Informasi	13
2.3 Pemodelan.....	14
2.3.1 <i>Entity Relationship Diagram (ERD)</i>	14
2.3.2 <i>Context Diagram</i>	14
2.3.3 <i>Data Flow Diagram (DFD)</i>	15
2.3.4 Database dan <i>MySQL (My Sequential)</i>	15
2.3.5 <i>PHP (Hypertext Preprocessor)</i>	15
2.3.6 <i>Flowchart</i>	16
2.4 Perangkat Lunak yang Digunakan	18
2.4.1 <i>Draw.io</i>	18
2.4.2 <i>Midtrans</i>	18
2.4.3 <i>Power Designer</i>	19

2.4.4 XAMPP.....	19
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	20
3.1 Gambaran Umum Objek Penelitian	20
3.1.1 Keadaan Sistem yang Berjalan.....	21
3.1.2 Kelebihan Sistem.....	22
3.1.3 Kelemahan Sistem.....	22
3.2 Alur Proses	23
3.2.1 Identifikasi dan Analisis Proses Bisnis	23
3.2.2 Identifikasi dan Analisis Kebutuhan	27
3.2.3 Idenetifikasi dan Analisis Alternatif Solusi	33
3.3 Desain Sistem.....	35
3.3.1 Desain Output.....	35
3.3.2 Desain Input	36
3.3.3 Desain Proses	39
3.3.4 Identifikasi dan Desain Database	44
3.3.5 Identifikasi dan Desain <i>User Interfacee</i>	48
BAB IV IMPLEMENTASI SISTEM	52
4.1 Konstruksi Sistem	52
4.1.1 Kebutuhan Sistem	52
4.1.2 Instalasi Sistem.....	53

4.1.3 Segmen Program	64
4.2 Skenario Pengujian.....	103
4.3 Pengujian	108
4.3.1 Cara Kerja Sistem.....	108
4.3.2 Hasil Pengujian	114
4.4 <i>Maintenance</i>	115
BAB V PENUTUP	116
5.1 Kesimpulan.....	116
5.2 Saran.....	117
DAFTAR PUSTAKA	118
LAMPIRAN.....	120
CURICULUM VITAE.....	126

DAFTAR TABEL

Tabel 2. 1 Simbol-Simbol <i>Flowchart</i>	16
Tabel 3. 1 Registrasi <i>User</i>	28
Tabel 3. 2 Tabel <i>Login User</i> dan Admin.....	28
Tabel 3. 3 Pembelian Tiket	29
Tabel 3. 4 Pembelian Tiket	30
Tabel 3. 5 Kebutuhan Spesifikasi Sistem.....	32
Tabel 3. 6 Analisis Kelayakan	34
Tabel 3. 7 Proses Pembelian Tiket.....	40
Tabel 3. 8 <i>Scan Barcode</i>	41
Tabel 3. 9 Database	44
Tabel 3. 10 Identifikasi <i>Interface</i>	48
Tabel 4. 1 Tabel Segmen Koneksi	65
Tabel 4. 2 Segmen Registrasi.....	66
Tabel 4. 3 Segmen <i>Login</i>	68
Tabel 4. 4 Segmen <i>Logout</i>	72
Tabel 4. 5 Segmen <i>Form</i> Pembelian.....	73
Tabel 4. 6 Segmen Keranjang	82
Tabel 4. 7 Segmen Riwayat	87
Tabel 4. 8 Segmen <i>Dashboard Admin</i>	95
Tabel 4. 9 <i>Developer Testing</i> oleh <i>Developer</i>	104
Tabel 4. 10 Perhitungan <i>Developer Testing</i> oleh <i>Developer</i>	114

DAFTAR GAMBAR

Gambar 1. 1 Metode <i>Waterfall</i>	6
Gambar 3. 1 <i>Flowchart User</i>	24
Gambar 3. 2 <i>Flowchart Login User dan Admin</i>	25
Gambar 3. 3 <i>Flowchart Pembelian Tiket</i>	26
Gambar 3. 4 <i>Flowchart Laporan Pembelian Tiket</i>	27
Gambar 3. 5 <i>Invoice Pembelian Tiket</i>	36
Gambar 3. 6 <i>List Pengunjung</i>	36
Gambar 3. 7 <i>Registrasi</i>	37
Gambar 3. 8 <i>Login</i>	37
Gambar 3. 9 <i>Halaman Menu Pembelian</i>	38
Gambar 3. 10 <i>Halaman Detail Pesanan</i>	39
Gambar 3. 11 <i>Arsitektur Aplikasi</i>	42
Gambar 3. 12 <i>Context Diagram</i>	42
Gambar 3. 13 <i>Data Flow Diagram</i>	43
Gambar 3. 14 <i>Entity Relationship Diagram</i>	47
Gambar 3. 15 <i>Dashboard Website</i>	49
Gambar 3. 16 <i>Dashboard Admin</i>	49
Gambar 3. 17 <i>Beranda User</i>	50
Gambar 3. 18 <i>Registrasi</i>	50
Gambar 3. 19 <i>Login</i>	51

Gambar 4. 1 Install <i>XAMPP</i>	54
Gambar 4. 2 Pilih Komponen <i>XAMPP</i>	55
Gambar 4. 3 Lokasi Instalasi.....	55
Gambar 4. 4 Layanan <i>XAMPP</i>	56
Gambar 4. 5 Website <i>Visual Studio Code</i>	57
Gambar 4. 6 Mengunduh <i>Visual Studio Code</i>	58
Gambar 4. 7 Proses Unduhan <i>Visual Studio Code</i>	58
Gambar 4. 8 Proses <i>Install Visual Studio Code</i>	59
Gambar 4. 9 Lokasi Folder Instaslasi.....	60
Gambar 4. 10 Folder <i>Start Menu</i>	60
Gambar 4. 11 Pilih Komponen <i>Visual Studio Code</i>	61
Gambar 4. 12 Proses <i>Install Visual Studio Code</i>	62
Gambar 4. 13 Tahap Akhir <i>Install Visual Studio Code</i>	62
Gambar 4. 14 Situs <i>Chrome</i>	63
Gambar 4. 15 <i>File Chrome</i>	63
Gambar 4. 16 <i>Install Chrome Selesai</i>	64
Gambar 4. 17 <i>Landing Page</i>	108
Gambar 4. 18 Registrasi.....	109
Gambar 4. 19 Login	109
Gambar 4. 20 <i>Dashboard User</i>	110
Gambar 4. 21 <i>Form Pembelian</i>	110
Gambar 4. 22 Menu Keranjang.....	111
Gambar 4. 23 <i>Snap Token Pembayaran</i>	111

Gambar 4. 24 Pembayaran Sukses	112
Gambar 4. 25 Menu Riwayat	112
Gambar 4. 26 Halaman Detail Pesanan.....	113
Gambar 4. 27 Halaman <i>Dashboard</i> Admin	113



BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia adalah sebuah negara yang sangat kaya akan destinasi alam yang banyak macamnya. Selain itu, Indonesia juga memiliki banyak situs bersejarah peninggalan kerajaan yang hingga kini masih terawat dengan baik. Karena keunikan tempat-tempat tersebut, baik dari segi keindahan maupun nilai historisnya, banyak situs bersejarah dijadikan objek wisata oleh pemerintah daerah. Pariwisata juga menjadi sektor andalan dalam pembangunan daerah, karena mampu memberikan kontribusi signifikan terhadap pendapatan daerah melalui beragam potensi wisata yang menarik.[1]

Taman Ujung Sukasada adalah salah satu destinasi wisata yang populer di Karangasem, Bali. Selain menawarkan keindahan alamnya, tempat ini juga menyimpan banyak kisah sejarah terkait Kerajaan Karangasem, karena Taman Ujung Sukasada merupakan salah satu warisan peninggalan kerajaan tersebut. Pembangunan taman ini dimulai pada tahun 1909 atas gagasan Anak Agung Anglurah Ketut Karangasem. Perancangnya adalah arsitek asal Belanda bernama van Den Hentz dan arsitek Cina bernama Loto Ang, serta melibatkan seorang undagi atau arsitek tradisional Bali. Taman Ujung sendiri merupakan pengembangan dari kolam Dirah yang telah dibangun pada tahun 1901. Proses pembangunan Taman Ujung selesai pada tahun 1921.[2]

Taman Ujung Sukasada senantiasa dipadati oleh pengunjung setiap hari, baik dari wisatawan lokal maupun turis mancanegara. Saat ini, sistem pembelian tiket masuk masih dilakukan secara manual, di mana pengunjung membeli tiket langsung di lokasi dengan menyerahkan uang tunai kepada petugas loket, lalu tiket diberikan kepada pengunjung. Setelah itu, petugas akan memeriksa tiket untuk memastikan setiap pengunjung sudah memiliki tiket resmi. Proses ini memerlukan sangat banyak waktu dan tenaga. Untuk mengatasi kendala tersebut, direncanakan pengembangan sistem baru yang memungkinkan pembelian tiket dilakukan secara *online*, dengan pembayaran melalui *gateway payment* agar mendukung transaksi non-tunai. Maka dari latar belakang yang sudah dipaparkan, penulis akan membahas masalah ini dalam tugas akhir dengan harapan sistem yang dirancang dapat membantu mempercepat dan mempermudah proses pembelian tiket sehingga lebih efisien dan efektif.

1.2 Identifikasi Masalah

Dari penjelasan di atas, terdapat dua masalah yang telah kami temukan, yaitu:

1. Proses pembelian tiket di Taman Sukasada masih dilakukan secara manual atau *offline*.
2. Pembayaran tiket di Taman Sukasada masih menggunakan sistem pembayaran secara tunai.

1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, kami merumuskan permasalahan dalam penelitian ini sebagai berikut:

1. Bagaimana merancang agar proses pembelian tiket masuk di Taman Sukasada dapat dilakukan secara *online*.
2. Bagaimana merancang agar proses pembayaran tiket masuk di Taman Sukasada dapat dilakukan secara non-tunai melalui *payment gateway*.

1.4 Batasan Masalah

Dalam hal ini, batasan masalah ditetapkan untuk memusatkan fokus penelitian sehingga dapat diperoleh hasil yang sesuai dengan tujuan. Adapun ruang lingkup permasalahan yang akan dibahas meliputi:

1. Perancangan sistem informasi untuk pembelian tiket Taman Sukasada secara *online*.
2. Pembuatan sistem pembayaran tiket masuk melalui *payment gateway*.

1.5 Tujuan Penelitian

Tujuan dari pembuatan website sistem informasi ini antara lain:

1. Merancang sistem informasi pembelian tiket masuk Taman Ujung Sukasada Karangasem Bali guna memudahkan proses pembelian tiket secara *online*.
2. Mempermudah proses pembayaran tiket melalui *payment gateway* agar dapat mengurangi antrean panjang.

1.6 Manfaat Penelitian

Manfaat yang dapat diperoleh dari penelitian ini adalah sistem informasi tersebut mampu memberikan kemudahan bagi calon pengunjung dalam melakukan pembelian tiket masuk taman melalui website yang akan dikembangkan.

1.7 Metode Penelitian

Dalam penyusunan skripsi ini, kami melakukan observasi langsung ke lokasi penelitian dengan mengamati serta mengumpulkan data yang dibutuhkan guna mendukung penulisan laporan penelitian ini. Adapun penjelasan lebih rinci mengenai metode penelitian yang kami gunakan disajikan berikut ini.

1.7.1 Jenis Penelitian

Jenis penelitian yang kami terapkan adalah penelitian studi kasus. Studi kasus merupakan metode penelitian yang mengeksplorasi suatu fenomena dalam konteks nyata dengan menganalisis satu atau beberapa kasus secara mendalam. Metode ini sering dimanfaatkan dalam penelitian sistem informasi untuk memahami penerapan, adopsi, atau dampak suatu sistem dalam lingkungan tertentu.

1.7.2 Teknik Pengumpulan Data

Berikut ini adalah teknik pengumpulan data yang kami gunakan untuk memperoleh informasi yang dibutuhkan dalam penelitian ini:

a. Observasi (Pengamatan)

Melalui observasi, kami melakukan pengamatan serta pencatatan secara teliti terhadap objek yang diteliti, dengan cara mengumpulkan bahan atau sampel yang nantinya berguna dalam memecahkan permasalahan.

b. Wawancara (*Interview*)

Wawancara dilakukan secara langsung dengan staf yang menangani proses pembelian tiket *online*. Dalam kegiatan ini, kami mengajukan pertanyaan secara langsung untuk memperoleh informasi yang jelas dan valid, sehingga data yang diperoleh sesuai dengan kondisi nyata.

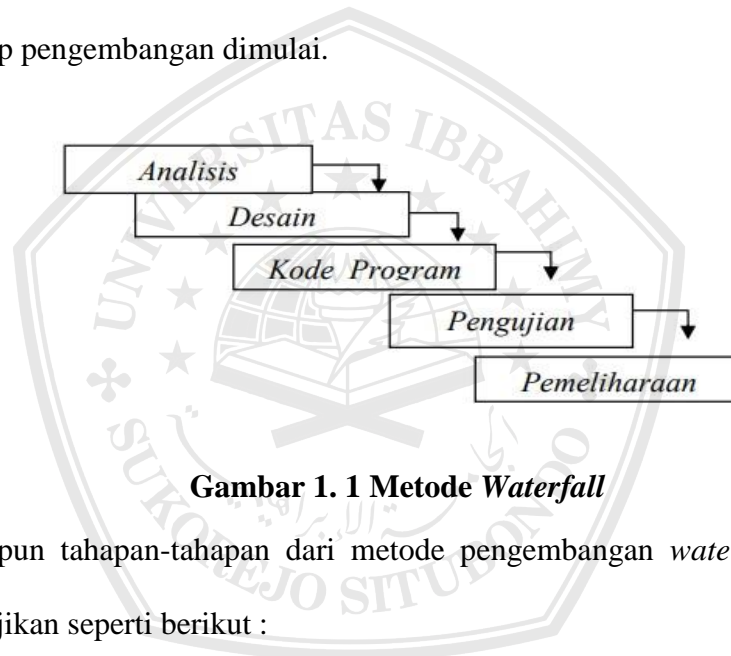
c. Studi Kepustakaan (*Library Research*)

Studi kepustakaan dilakukan untuk mengumpulkan data dan informasi melalui berbagai sumber pustaka, seperti dokumen, buku, majalah, serta literatur sejarah.[3] Dalam metode ini, penulis memperoleh data dengan membaca dan menelaah buku-buku referensi yang berkaitan dengan topik penelitian.

1.7.3 Metode Pengembangan Sistem

Metode pengembangan yang kami gunakan adalah metode *waterfall*, yaitu salah satu model pengembangan perangkat lunak berbasis alur air terjun yang menggunakan pendekatan sekuensial atau

berurutan. Prosesnya dimulai dari tahap analisis, desain, pengkodean, pengujian, hingga pemeliharaan. Disebut model *waterfall* karena setiap tahap harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya, dan semuanya dilakukan secara berurutan.[4] Metode *waterfall* ini membantu mengatasi kompleksitas yang sering muncul, karena model ini merinci apa saja yang harus dilakukan perangkat lunak, termasuk mengumpulkan serta menetapkan kebutuhan sistem sebelum tahap pengembangan dimulai.



Gambar 1. 1 Metode Waterfall

Adapun tahapan-tahapan dari metode pengembangan *waterfall* dapat disajikan seperti berikut :

a. Analisa Sistem

Pada tahap analisis sistem, kami melakukan analisis terhadap kebutuhan bisnis yang ingin dicapai melalui sistem yang akan dikembangkan. Proses ini dilakukan dengan mewawancarai pihak-pihak terkait untuk memahami kebutuhan serta harapan mereka terhadap sistem tersebut. Tujuan dari analisis kebutuhan ini adalah untuk mengidentifikasi kebutuhan yang diperlukan dalam perancangan, baik

berupa dokumen maupun sumber lainnya yang dapat mendukung dalam merumuskan solusi atas permasalahan yang ada, baik dari sisi pengguna maupun admin.[5]

b. Desain

Tahap ini adalah bentuk implementasi dari analisis kebutuhan yang kemudian dituangkan ke dalam perancangan dengan memanfaatkan perangkat keras dan perangkat lunak komputer.[6] Peneliti akan menyusun desain alur proses bisnis terkait pembelian tiket. Selain itu, juga akan dirancang database dan dibuat desain awal tampilan website.

c. Kode Program

Pada tahap ini, peneliti mulai mengembangkan website berdasarkan hasil analisis kebutuhan, dengan membuat *form* input dan output yang dirancang untuk memudahkan penggunaan website, baik bagi pengguna maupun admin, menggunakan bahasa pemrograman yang dikuasai oleh programmer.

d. Pengujian

Pengujian ini dilakukan dengan tujuan untuk memeriksa apakah program yang telah dibuat sudah sesuai dengan kebutuhan yang telah direncanakan. Selain itu, pengujian ini juga bertujuan untuk memastikan apakah program sudah siap digunakan dan bebas dari kendala.

e. Pemeliharaan

Pemeliharaan adalah tahap yang bertujuan untuk menjaga agar sistem tetap berfungsi secara normal. Pada tahap ini dilakukan upaya

pengecahan agar sistem terhindar dari masalah atau gangguan yang dapat mengakibatkan kerusakan.

1.8 Sistematika Pembahasan

Penyusunan sistematika penulisan di tujukan agar bisa memudahkan dalam menjelaskan tujuan dari bab yang akan dibahas, berikut sistematika penulisanya :

a. **BAB I : PENDAHULUAN**

Pada bab pertama menjelaskan latar belakang, identifikasi masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian, jenis penelitian, teknik pengumpulan data, metode pengembangan sistem dan sistematika pembahasan.

b. **BAB II : TINJAUAN PUSTAKA**

Pada bab kedua memaparkan tinjauan pustaka yang menjadi acuan dari penelitian ini dan membahas landasan teori yang berkaitan dengan Sistem Informasi Pembelian Tiket Wisata Taman Ujung Sukasada Karangasem Bali.

c. **BAB III : ANALISIS DAN PERANCANGAN SISTEM**

Pada bab ketiga menjelaskan gambaran umum objek penelitian mulai dari keadaan sistem yang sedang berjalan berupa kelebihan serta kekurangan atau kelemahan dari sistem. Selanjutnya membahas alur proses bisnis dari sistem tersebut. Identifikasi dan analisis kebutuhan baik fungsional maupun non fungsional, identifikasi dan analisis alternatif solusi,

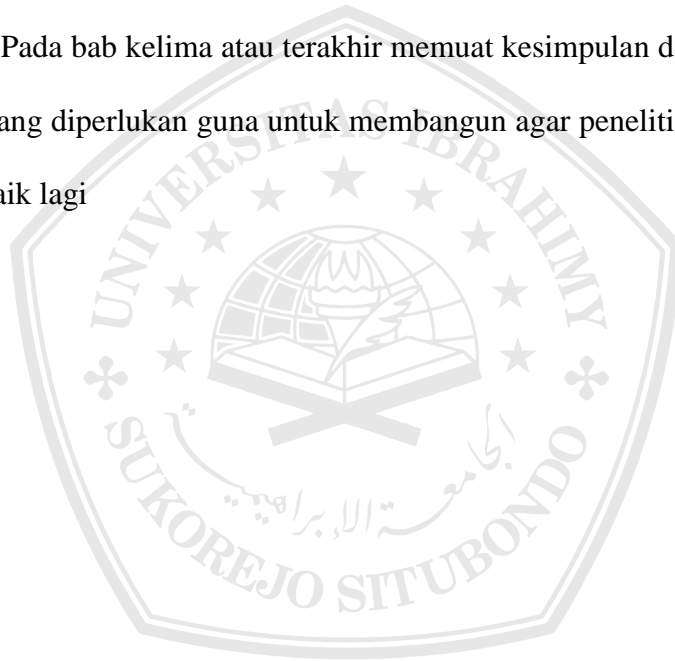
desain sistem terdiri dari desain input, desain output, desain proses, desain database, dan desain *user interface*.

d. BAB IV : IMPLEMENTASI SISTEM

Pada bab keempat menjelaskan konstruksi sistem dari kebutuhan sistem, instalasi sistem, segmen program, skenerio pengujian, pengujian sistem, cara kerja sistem/manual book, hasil pengujian dan maintenance.

e. BAB V : PENUTUP

Pada bab kelima atau terakhir memuat kesimpulan dari laporan dan saran yang diperlukan guna untuk membangun agar penelitian kedepannya lebih baik lagi



BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Berikut ini adalah beberapa jurnal yang kami gunakan sebagai tinjauan pustaka guna membandingkan antara penelitian yang sudah dilakukan oleh peneliti terdahulu dengan penelitian yang sedang kami buat. Berikut antara lainnya :

Perancangan Aplikasi Pemesanan Tiket Wisata di Kecamatan Donorojo Menggunakan Metode *Payment Gateway*. Penelitian ini dilakukan oleh Hadi Wiyono dan Muhammad Fachrie, mahasiswa Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta. Studi ini membahas tentang pengelolaan objek wisata di Kecamatan Donorojo yang hingga kini masih dilakukan secara manual dalam berbagai aspeknya. Informasi mengenai objek wisata di wilayah ini sebagian besar belum terdigitalisasi, sehingga menimbulkan ketidakjelasan terkait data penting yang seharusnya diketahui calon wisatawan, seperti detail objek wisata, harga tiket, dan jadwal kunjungan. Proses pemesanan tiket pun kerap dianggap rumit dan membingungkan, sehingga menyulitkan wisatawan dalam menyusun rencana kunjungan. Untuk menjawab permasalahan tersebut, dikembangkanlah aplikasi pemesanan tiket wisata yang terintegrasi dengan metode *Payment Gateway* guna memberikan solusi yang lebih praktis dan nyaman bagi wisatawan yang ingin berkunjung ke Donorojo. Dalam bidang

pariwisata, penggunaan *Payment Gateway* memberikan banyak manfaat. Wisatawan kini dapat dengan mudah memperoleh informasi lengkap mengenai destinasi, harga tiket, serta layanan lainnya, sekaligus melakukan pembayaran tiket atau layanan secara aman dan cepat melalui *platform* daring, sehingga menghilangkan kerumitan yang sering terjadi pada metode pembayaran tradisional. Selain memudahkan wisatawan, *Payment Gateway* juga membawa keuntungan besar bagi pelaku usaha pariwisata, seperti pengelolaan pembayaran yang lebih efisien, penurunan risiko penipuan, dan peningkatan efisiensi operasional. Dengan demikian, *Payment Gateway* bukan sekadar alat pembayaran, tetapi juga menjadi elemen penting dalam meningkatkan efektivitas bisnis di sektor pariwisata.[7]

Perancangan Sistem Informasi Pembelian Tiket Wisata Berbasis

Web. Penelitian ini dilakukan oleh Aulia Khasanah Putri, Putri Khoddijah, Syahida, dan Mentari Tri Indah Rahmayani, yang merupakan mahasiswa dari Program Studi Ekonomi Syariah, Syariah dan Ekonomi Islam, serta Ilmu Komputer dan Manajemen Dakwah di Sekolah Tinggi Agama Islam (STAIN) Bengkalis. Selat Baru merupakan salah satu desa yang berada di Kecamatan Bantan, Kabupaten Bengkalis, Provinsi Riau, Indonesia. Saat ini, Pantai Indah Selat Baru belum memiliki sistem informasi pembelian tiket secara daring dan masih menggunakan tiket manual. Kondisi ini dinilai kurang efektif dan efisien, khususnya bagi calon wisatawan yang ingin menghindari antrean saat memasuki kawasan wisata tersebut. Selain itu, sistem manual juga merepotkan petugas yang harus mencatat data pengunjung secara manual di buku. Untuk

mengatasi permasalahan ini, dikembangkanlah Sistem Informasi Pembelian Tiket Berbasis Web yang menyediakan layanan pembelian tiket, pembayaran, validasi tiket melalui pemindaian *barcode*, serta informasi seputar Pantai Indah Selat Baru. Diharapkan dengan hadirnya website ini, calon wisatawan dapat lebih mudah mengakses objek wisata tersebut, dan petugas pun tidak perlu lagi melakukan pencatatan jumlah pengunjung secara manual.[8]

Perancangan Sistem Informasi Penjualan Tiket Kolam Renang Yonif 136 Berbasis Website. Penelitian ini dilakukan oleh Renata Hutasoit dan Mesri Silalahi, mahasiswa Program Studi Sistem Informasi, Universitas Putera Batam. Di Kota Batam terdapat sebuah kolam renang yang dikenal dengan nama kolam renang Yonif 136. Kolam renang ini menjadi salah satu destinasi wisata yang banyak diminati oleh masyarakat dari berbagai kalangan, mulai dari anak-anak, remaja, dewasa, hingga orang tua. Saat ini, untuk dapat menikmati fasilitas kolam renang tersebut, pengunjung harus membeli tiket langsung kepada petugas, di mana sistem yang digunakan masih bersifat manual. Hal ini sering menimbulkan kerumunan akibat keterbatasan tempat serta hilangnya data ketika jumlah pengunjung membludak. Oleh karena itu, penulis merancang sebuah sistem penjualan tiket guna mengatasi permasalahan tersebut. Perancangan sistem ini diharapkan dapat mengurangi antrean panjang dan kepadatan di loket pendaftaran. Selain itu, penerapan penjualan tiket secara *online* memungkinkan biaya menjadi lebih efisien, proses pendaftaran lebih mudah, dan lebih praktis tanpa terhambat oleh waktu.[9]

Alasan kami menggunakan ketiga jurnal diatas karena jurnal tersebut memiliki kesamaan dalam hal permasalahan dan solusi yang ingin diterapkan dalam penelitian ini kedepannya diantaranya pembuatan sistem informasi yang mewadahi tentang pendaftaran pengunjung tempat wisata yang juga pembayarannya dilakukan secara *online*.

2.2 Landasan Teori

2.2.1 Pengertian Sistem Informasi

a. Sistem

Sistem dapat diartikan sebagai suatu kesatuan yang tersusun dari dua atau lebih komponen atau sub-sistem yang saling berinteraksi untuk mencapai tujuan tertentu.[10]

b. Informasi

Informasi adalah sekumpulan data yang telah dihimpun dan memiliki makna sehingga dapat dimanfaatkan.[11] Dapat dikatakan bahwa informasi terbentuk dari data-data yang diperoleh melalui berbagai metode, seperti hasil observasi, wawancara, atau cara lainnya yang dapat memastikan data tersebut sesuai dengan kenyataan.

c. Sistem Informasi

Sistem informasi secara teknis diartikan sebagai kumpulan komponen yang saling berkaitan dan berfungsi untuk mengumpulkan (atau mengambil kembali), mengolah, menyimpan, serta menyebarkan informasi guna mendukung proses

pengambilan keputusan dan pengendalian dalam suatu organisasi. Selain itu, sistem informasi juga berperan dalam membantu manajer dan karyawan dalam menganalisis permasalahan, memvisualisasikan hal-hal yang kompleks, serta mengembangkan produk baru.[12]

d. Website

Website atau situs adalah kumpulan halaman yang terdiri dari berbagai bagian yang menyajikan informasi dalam bentuk data digital, seperti teks, gambar, video, audio, dan animasi, yang dikirim melalui internet.[13]

2.3 Pemodelan

2.3.1 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) merupakan sebuah diagram atau ilustrasi yang menunjukkan bagaimana informasi dibuat, disimpan, dan digunakan dalam sistem bisnis.[14]

2.3.2 *Context Diagram*

Diagram konteks adalah diagram yang berisi satu proses utama dan digunakan untuk menggambarkan batasan atau ruang lingkup sebuah sistem. Diagram konteks merupakan tingkat paling atas dari *DFD (Data Flow Diagram)* yang menampilkan seluruh aliran masuk dan keluar dari sistem serta berfungsi untuk memetakan model lingkungannya. Diagram konteks adalah diagram yang

menunjukkan hubungan atau interaksi antara sebuah sistem dan entitas terkait.[15]

2.3.3 *Data Flow Diagram (DFD)*

Data Flow Diagram (DFD) merupakan sebuah model logis yang digunakan untuk menggambarkan asal data, tujuan data yang keluar dari sistem, lokasi penyimpanan data, proses yang menghasilkan data tersebut, serta hubungan antara data yang disimpan dan proses yang diterapkan pada data tersebut.[16]

2.3.4 Database dan *MySQL (My Sequential)*

MySQL adalah perangkat lunak yang termasuk dalam kategori *DBMS (Database Management System)* dan bersifat *open source*. *Open source* berarti bahwa perangkat lunak ini disertai dengan kode sumber (*source code*), yaitu kode yang digunakan untuk membangun *MySQL*, selain juga menyediakan *file executable*, yakni kode yang dapat langsung dijalankan di sistem operasi.[17]



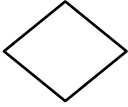
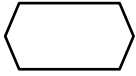
2.3.5 *PHP (Hypertext Preprocessor)*

Menurut Mad-coms, “*PHP (Hypertext Preprocessor)* merupakan bahasa skrip yang dapat disematkan atau digabungkan ke dalam *HTML*. *PHP* juga dikenal sebagai bahasa pemrograman skrip *server-side* yang dirancang khusus untuk pengembangan web. Disebut bahasa pemrograman *server-side* karena proses *PHP* dijalankan pada komputer server.”[18]


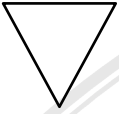


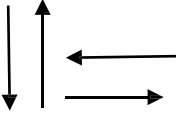
2.3.6 Flowchart

Bagan alir (*flowchart*) merupakan teknik analitis berbentuk gambar yang digunakan untuk menggambarkan berbagai aspek dalam sistem informasi secara jelas, singkat, dan logis. Bagan alir mendokumentasikan bagaimana proses bisnis dijalankan serta bagaimana dokumen bergerak di dalam organisasi. *Flowchart* menggambarkan alur sistem, prosedur, dan pengendalian internal yang diterapkan perusahaan.[19]

Tabel 2. 1 Simbol-Simbol *Flowchart*

Simbol	Fungsi
	Simbol Proses digunakan untuk menunjukkan operasi yang dijalankan oleh komputer.
	Simbol Manual untuk menunjukkan proses yang dilakukan secara manual tanpa bantuan komputer.
	Simbol <i>Decision</i> untuk menunjukkan kondisi tertentu yang dapat menghasilkan kemungkinan jawaban, yaitu ya atau tidak.
	Simbol <i>Predefined Process</i> digunakan untuk menunjukkan tempat penyimpanan proses pengolahan bertujuan memberikan nilai awal.

Tabel 2. 1 Simbol-Simbol *Flowchart* (Tabel Lanjutan)

Simbol	Fungsi
	<p>Simbol <i>Keying Operation</i> digunakan untuk menunjukkan jenis operasi yang dijalankan menggunakan mesin yang dilengkapi dengan <i>keyboard</i>.</p>
	<p>Simbol <i>Offline-Storage</i> digunakan untuk menunjukkan bahwa data akan disimpan ke dalam media penyimpanan tertentu.</p>
	<p>Simbol <i>Punch Card</i> digunakan untuk menunjukkan bahwa data input berasal dari kartu atau output dituliskan ke dalam kartu.</p>
	<p>Simbol Dokumen digunakan untuk menyatakan bahwa input berasal dari dokumen berbentuk kertas atau output dicetak ke dalam kertas.</p>
	<p><i>Flow Direction Symbol</i> digunakan untuk menghubungkan satu simbol dengan simbol lainnya dalam suatu alur proses.</p>

2.4 Perangkat Lunak yang Digunakan

2.4.1 *Draw.io*

Draw.io merupakan sebuah situs web yang dirancang khusus untuk membuat diagram *UML* secara *online*. *Platform* ini memiliki tampilan yang sangat responsif dan terintegrasi dengan layanan penyimpanan *file* milik *Google*, yaitu *Google Drive*, sehingga menjadi alternatif praktis untuk membuat diagram *UML* dalam waktu yang lebih efisien.[20]

2.4.2 *Midtrans*

Midtrans adalah layanan pembayaran yang menyediakan fitur untuk mempermudah proses uji coba pembayaran. Dengan hanya memasukkan kode transaksi dan menekan tombol bayar, transaksi dapat diselesaikan dengan cara yang praktis. Selain itu, *Midtrans* juga berperan sebagai salah satu *payment gateway* yang mendukung kebutuhan para pelaku bisnis *online* dengan menawarkan berbagai metode pembayaran. Layanan ini membantu industri untuk beroperasi lebih mudah sekaligus meningkatkan penjualan. *Midtrans* adalah *platform* yang berfungsi sebagai *payment gateway*. *Midtrans* menjadi sistem pembayaran yang digunakan untuk memfasilitasi transaksi antara pembeli dan penjual. *Platform* ini memiliki fitur yang telah terintegrasi dengan *e-commerce* sesuai dengan kebutuhan transaksi

pembayaran *online*, baik melalui kartu debit, kartu kredit, maupun layanan penarikan dan pengiriman uang tunai.[21]

2.4.3 *Power Designer*

Power Designer adalah perangkat lunak pemodelan yang berfungsi mendukung desain perangkat lunak berbasis model serta mempermudah perancangan database yang terstruktur dan efisien.[22]

2.4.4 *XAMPP*

XAMPP adalah perangkat lunak gratis yang mendukung berbagai sistem operasi dan merupakan kumpulan dari beberapa program. *XAMPP* berfungsi sebagai alat yang menyatukan sejumlah perangkat lunak dalam satu paket. Dengan menginstal *XAMPP*, Anda tidak perlu lagi melakukan instalasi dan konfigurasi web server *Apache*, *PHP*, dan *MySQL* secara manual. *XAMPP* akan secara otomatis menginstal dan mengonfigurasikannya untuk Anda, atau dengan kata lain, melakukan konfigurasi secara otomatis.[23]

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Gambaran Umum Objek Penelitian

Taman Ujung Sukasada merupakan taman yang menyimpan sejarah dari Puri Karangasem, dan juga termasuk bangunan peninggalan dari Kerajaan Karangasem. Taman Ujung memiliki struktur bangunan yang sangat unik dan pemandangan taman yang sangat indah dan rapi.

Taman tersebut adalah taman sejarah kerajaan. Destinasi wisata Taman Soekasada Ujung Karangasem pertama kali didirikan oleh Raja Karangasem, I Gusti Bagus Jelantik, yang menyandang gelar Anak Agung Anglurah Ketut Karangasem, pada tahun 1909. Beliau dikenal memiliki minat dan keahlian tinggi dalam bidang seni dan arsitektur. Raja sendiri yang merancang serta memimpin proses pembangunannya. Dalam berbagai kunjungannya ke luar negeri, beliau mendapat inspirasi untuk merancang taman ini dengan menggabungkan unsur-unsur arsitektur Eropa dan Belanda. Pembangunan taman dilakukan secara bertahap, di mana sebagian bangunan selesai pada tahun 1921, dan seluruh kompleks rampung pada tahun 1926. Pada tahun 1937, Taman Ujung Karangasem diresmikan melalui sebuah prasasti marmer yang ditulis dalam aksara Latin dan Bali, serta menggunakan dua bahasa, yakni Melayu dan Bali. Luas lahan yang dimiliki taman soekasada ujung keseluruhannya kurang lebih yaitu mencapai 400 hektar, tetapi sekarang hanya tersisa tinggal sekitar 10 hektar, kebanyakan tanah tersebut sudah dibagikan

kepada masyarakat pada masa *land reform*. Pada tahun 1963, Bali mengalami bencana alam berupa erupsi Gunung Agung. Letusan ini memicu gempa bumi yang menyebabkan sebagian bangunan di kawasan taman kerajaan mengalami kerusakan parah. Pemulihan taman tersebut baru dimulai kembali pada tahun 2001 dengan bantuan dari Bank Dunia. Seluruh bangunan berhasil dipugar dan berdiri kembali secara utuh pada tahun 2004, serta diresmikan oleh Gubernur Bali saat itu, Dewa Beratha. Taman Sukasada awalnya dibangun pada tahun 1909 atas prakarsa Anak Agung Anglurah Ketut Karangasem. Dalam proses pembangunannya, turut terlibat arsitek Belanda bernama Van Den Hentz dan arsitek Tionghoa bernama Loto Ang, serta seorang *undagi* atau arsitek tradisional Bali. Taman Ujung sendiri merupakan perluasan dari Kolam Dirah yang sebelumnya telah dibangun pada tahun 1901. Proyek pembangunan taman ini rampung pada tahun 1921.[2]

Pada saat ini Taman Ujung menjadi situs pariwisata yang sering dikunjungi turis asing maupun lokal yang dapat meningkatkan perekonomian masyarakat kota Karangasem Bali. Terkadang juga Taman Ujung digunakan dalam pameran budaya yang sering dilakukan satu tahun sekali dan kegiatan lainnya yang berbaur budaya, dalam rangka mengenalkan kepada para generasi muda, agar tak lupa dengan budaya yang mereka miliki

3.1.1 Keadaan Sistem yang Berjalan

Sistem yang masih diterapkan sekarang masih melakukan interaksi langsung antara pengunjung dan petugas loket. Hal ini dapat memberikan pengalaman yang lebih personal bagi pengunjung, terutama bagi mereka

yang lebih nyaman berinteraksi dengan manusia secara langsung. Sistem ini juga membuat pengunjung tidak perlu memiliki pengetahuan atau keterampilan khusus untuk membeli tiket secara manual. Sistem manual mungkin lebih fleksibel dalam menangani situasi-situasi khusus, seperti perubahan rencana atau permintaan khusus dari pengunjung

3.1.2 Kelebihan Sistem

Sistem pembelian tiket masuk taman secara *offline* memiliki beberapa kelebihan. Salah satunya adalah kemudahan akses bagi semua kalangan, terutama bagi pengunjung yang tidak terbiasa menggunakan teknologi atau tidak memiliki perangkat digital. Selain itu, sistem ini tidak memerlukan koneksi internet, sehingga tetap bisa digunakan di daerah yang memiliki sinyal lemah. Pengunjung spontan juga dapat langsung membeli tiket tanpa harus memesan sebelumnya. Interaksi langsung dengan petugas loket memungkinkan pengunjung mendapatkan bantuan dan informasi secara cepat. Sistem ini juga cocok untuk acara berskala kecil dan tidak memerlukan biaya tambahan seperti pada *platform online*.

3.1.3 Kelemahan Sistem

Sistem yang masih diterapkan sekarang masih melakukan interaksi langsung antara pengunjung dan petugas loket. Hal ini dapat memberikan pengalaman yang lebih personal bagi pengunjung, terutama bagi mereka yang lebih nyaman berinteraksi dengan manusia secara langsung. Sistem

ini juga membuat pengunjung tidak perlu memiliki pengetahuan atau keterampilan khusus untuk membeli tiket secara manual. Sistem manual mungkin lebih fleksibel dalam menangani situasi-situasi khusus, seperti perubahan rencana atau permintaan khusus dari pengunjung

3.2 Alur Proses

Alur proses adalah alur yang menggambarkan langkah-langkah yang diperlukan dalam menyelesaikan tugas atau mencapai tujuan tertentu. Pada bagian alur proses akan menjelaskan identifikasi proses dan analisis proses yang tengah berjalan pada sistem tersebut sehingga bisa memudahkan untuk memahami proses yang terjadi.

3.2.1 Identifikasi dan Analisis Proses Bisnis

a. Identifikasi Proses Bisnis

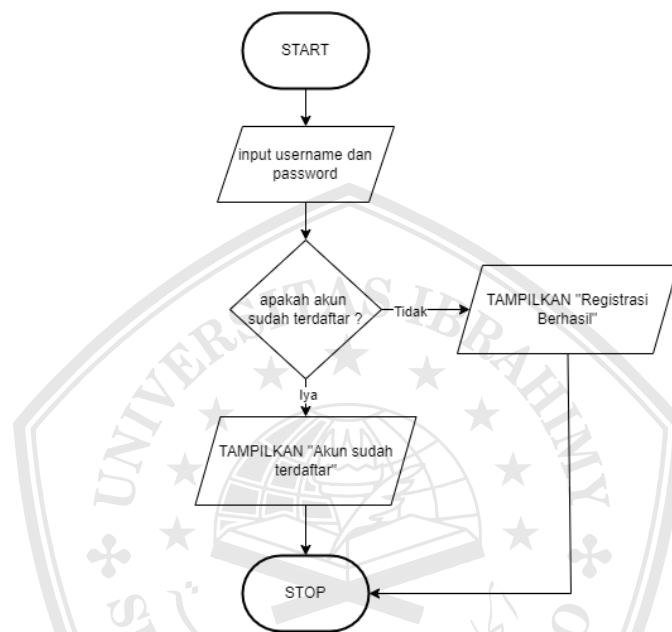
Berdasarkan hasil penelitian kami maka dapat diidentifikasi bahwa identifikasi proses yang berjalan di instansi sebagai berikut:

1. *Form* registrasi untuk *user*
2. *Form login* untuk admin dan *user*
3. *Form* pembelian dan pembayaran tiket masuk
4. Laporan pembelian tiket

b. Analisis proses bisnis

1. *Form Registrasi User*

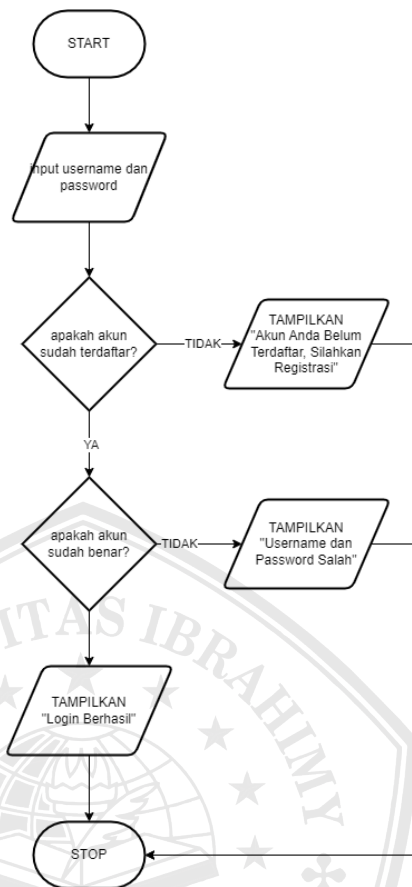
User saat akan menggunakan webiste diwajibkan untuk melakukan *login* terlebih dahulu. Namun jika belum mempunyai akun, maka diharapkan untuk melakukan registrasi akun. Berikut akan kami paparkan *flowchart login* bagi *user* berikut.



Gambar 3.1 *Flowchart User*

2. *Form Login User* dan Admin

Setelah melakukan registrasi *user* diharuskan melakukan *login* agar bisa mengakses semua fitur yang ada pada beranda *user*, seperti pembelian tiket, pembayaran tiket dan riwayat pembelian tiket. Sedangkan admin, setelah melakukan *login* akan dialihkan ke beranda admin yang menampilkan semua data pembeli atau pengunjung taman.

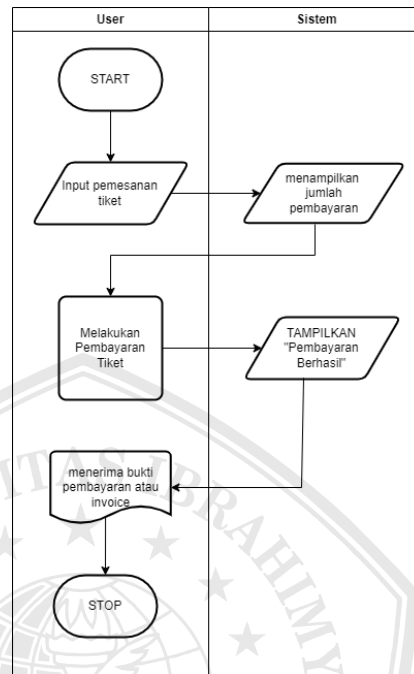


Gambar 3. 2 Flowchart Login User dan Admin

3. Form Pembelian Tiket

Dalam proses pembelian tiket, *user* akan melakukan input pemesanan, kemudian akan ditampilkan harga yang harus dibayar. Maka *user* harus *top-up* dengan melakukan pembayaran melalui *payment gateway*, yang nantinya akan kami lakukan melalui *midtrans*. Jadi nanti di *midtrans* sudah ada proses pembayaran dari berbagai pihak bank dan juga sudah mendukung sistem *virtual account*, setelah pembayaran akan diberikan *invoice* atau bukti pembayaran dan juga kode

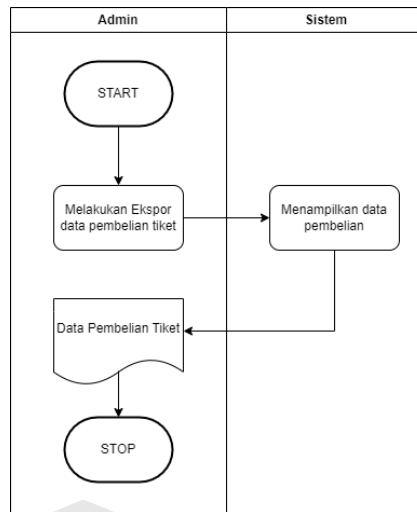
QR *user* yang akan digunakan nantinya ketika akan masuk ke taman.



Gambar 3. 3 Flowchart Pembelian Tiket

4. Laporan Pembelian Tiket

Ketika akan melakukan pelaporan admin cukup melakukan download data pembelian tiket di webiste, data tersebut yang nantinya akan dilaporkan kepada kepala taman.



Gambar 3. 4 Flowchart Laporan Pembelian Tiket

3.2.2 Identifikasi dan Analisis Kebutuhan

Dari hasil penelitian yang kami lakukan pada di Taman Ujung Sukasada Ujung, maka dapat kami identifikasi dan analisa kebutuhan-kebutuhan sistem yang akan dibuat atau dirancang. Berikut ini adalah identifikasi dan analisis kebutuhannya.

a. Identifikasi Kebutuhan Fungsional

Disamping dari proses pembelian tiket masuk. Dibutuhkan juga halaman yang berfungsi menyimpan berkas data para pengunjung yang berkunjung ke Taman Ujung, maka ada beberapa hal yang diperlukan atau dibutuhkan guna membangun sistem kerjasama.

1. Proses Registrasi *User*

Pada tabel ini menggambarkan kebutuhan fungsional proses registrasi bagi *user* yang mengharuskan untuk menginput *username* dan juga *password*.

Tabel 3. 1 Registrasi *User*

<i>User</i>	Kebutuhan Fungsional
Melakukan registrasi atau pendaftaran akun	<i>Form Registrasi</i>

2. Proses *Login Admin dan User*

Berikut ini kebutuhan fungsional dalam proses *login admin dan user*. Di dalam proses ini *admin dan user* harus memasukkan *username* dan *password* yang sudah di daftarkan saat melakukan registrasi akun.

Tabel 3. 2 Tabel *Login User dan Admin*

<i>Admin dan User</i>	Kebutuhan Fungsional
Melakukan <i>login</i> atau memasukkan akun ke website	<i>Form Login</i>

3. Pembelian Tiket

Kebutuhan fungsional selanjutnya adalah pembelian tiket masuk taman. *User* akan diberikan *form* yang wajib diisi untuk melakukan pembelian tiket. Setelah itu *user* bisa melakukan pembayaran kapan saja dengan metode *payment gateway* yang di kelola langsung oleh pihak *midtrans*.

Tabel 3. 3 Pembelian Tiket

<i>User</i>	Kebutuhan Fungsional
Melakukan input <i>form</i> pembelian tiket	<i>Form</i> input pembelian tiket
Melakukan pembayaran tiket	<i>Form</i> pembayaran tiket

4. Laporan

Kebutuhan fungsional selanjutnya adalah data pembelian tiket masuk taman yang akan dilaporkan ke kepala taman. Data yang akan dilaporkan adalah data keseluruhan dari pembelian tiket. Di beranda admin tersebut bisa memilih kategori pembelian tiket yang akan dilaporkan seperti kategori Berdasarkan asal pengunjung dan juga ada kategori berdasarkan tanggal pembelian tiket.

Tabel 3. 4 Pembelian Tiket

Admin	Kebutuhan Fungsional
Download data pembelian tiket	Fitur download data pembelia tiket

b. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan fungsional adalah langkah setelah mengidentifikasi apa yang dibutuhkan dalam penelitian ini. Analisis ini dimaksudkan agar semua kebutuhan-kebutuhan dalam proses bisnis terpenuhi. Berikut ini adalah kebutuhan fungsional dari semua proses bisnis pengajuan kerjasama.

1. Proses Registrasi *User*

Pada proses registrasi, *user* cukup memasukkan nama atau *username* dan password, karena dengan adanya *username* dan password, *user* ketika akan melakukan *login* maka harus memasukkan *username* dan password yang telah di daftarkan.

2. Proses *Login*

Pada proses *login* admin dan *user* memiliki halaman *login* yang berbeda, karena jika admin melakukan *login* maka akan masuk ke dalam menu beranda yang hanya bisa diakses oleh admin. Sedangkan *user* atau pengunjung akan masuk ke

dalam menu beranda *user* yang akan menampilkan *form* pembelian tiket.

3. Pembelian Tiket

User setelah melakukan *login* akan diarahkan ke menu beranda *form* pembelian tiket yang berisi identitas pengunjung dan harga tiket yang harus dibayar oleh pengunjung. Ketika sudah lengkap mengisi *form* pembelian tiket, pembelian tiket akan di masukkan ke dalam keranjang, sesudah itu *user* harus melakukan pembayaran tiket, yang bisa dilakukan dengan metode *payment gateway* dari *midtrans* dan disini juga disajikan banyak pilihan metode pembayaran seperti via bank, virtual account dan lainnya. Saat pembayaran sudah sukses maka pengunjung akan menerima *invoice* bukti pembayaran yang bisa di unduh pada menu riwayat. Pada *invoice* tersebut sudah ditampilkan kode QR yang akan *discan* ketika akan memasuki taman.

4. Laporan

Dalam pelaporan ini admin bisa langsung melihat semua data pembelian tiket pada menu beranda admin. Dalam menu ini juga sudah disediakan fitur ekspor data pembelian yang sudah berbentuk excel dan ini memudahkan admin dalam melakukan perekapan pembelian tiket.

c. Analisis Kebutuhan Non Fungsional

1. Keamanan Sistem

Aspek terpenting dalam menjaga sebuah transaksi adalah keamanan. Di *Midtrans*, keamanan data pengguna dan transaksi dapat dikatakan terjamin karena *platform* ini telah terbukti andal dan digunakan oleh berbagai perusahaan besar.

2. Kinerja Sistem

Kinerja sistem perlu optimal agar proses dapat berjalan dengan lancar. Jangan sampai sistem yang seharusnya dirancang untuk menyederhanakan justru memperlambat proses pengajuan. Oleh karena itu, diperlukan sistem yang andal agar proses kerja sama dapat berlangsung dengan baik. Berikut ini adalah spesifikasi minimum sistem yang dibutuhkan.

Tabel 3. 5 Kebutuhan Spesifikasi Sistem

Komponen	Spesifikasi
Hardware	
PC dan Komputer	1. Prosesor intel core i3 2. Ram 4 GB 3. SSD 500 GB
Software	
Sistem Operasi	Windows 10
Browser	<i>Chrome</i>

3. Pemeliharaan

Pemeliharaan sistem merupakan hal yang wajib dilakukan, karena tidak dapat dipungkiri bahwa seiring waktu, sistem bisa mengalami error akibat berbagai faktor. Dengan melakukan pemeliharaan, sistem dapat terus diperbaiki dan diperbarui sesuai kebutuhan.

3.2.3 Idenetifikasi dan Analisis Alternatif Solusi

a. Identifikasi Alternatif Solusi

Identifikasi alternatif solusi dari sistem ini akan dijabarkan pada bagian ini

1. Bagian sistem yang akan dikomputerisaasi

Pada sistem website pembelian tiket ini bagian yang akan dikomputerisasi adalah dari data diri pengunjung dan riwayat pembelian dari *user*.

2. Manfaat Sistem

Manfaat yang akan didapatkan ketika website pembelian tiket *online* sudah dibuat yaitu akan memudahkan proses pembelian tiket. Disamping itu, secara tidak langsung semua data pembelian tiket bisa di ekspor dari website langsung untuk dijadikan laporan.

3. Alat Output

Alat output yang dipakai dalam sistem ini yaitu monitor laptop/komputer dan printer

4. Alat Input

Alat input yang digunakan yaitu mouse, keyboard dan code scanner.

5. Alat Penyimpanan Data

Sedangkan pada penyimpan data kerjasama yang digunakan yaitu database dari *MySQL* dan *midtrans*.

b. Analisis Kelayakan Solusi

Analisis kelayakan alternatif solusi yaitu analisis yang dilakukan guna memberikan kriteria alternatif solusi yang lebih spesifik pada sistem pembelian tiket.

Tabel 3. 6 Analisis Kelayakan

Kriteria Kelayakan	Bobot	Alternatif
Kelayakan Operasional	30%	Mendukung dalam membantu semua kebutuhan fungsional
Kelayakan Teknologi dan Keahlian	30%	Teknologi yang akan digunakan menjalankan sistem sudah memadai dan keahlian personil memadai dalam menjalankan sistem ini.

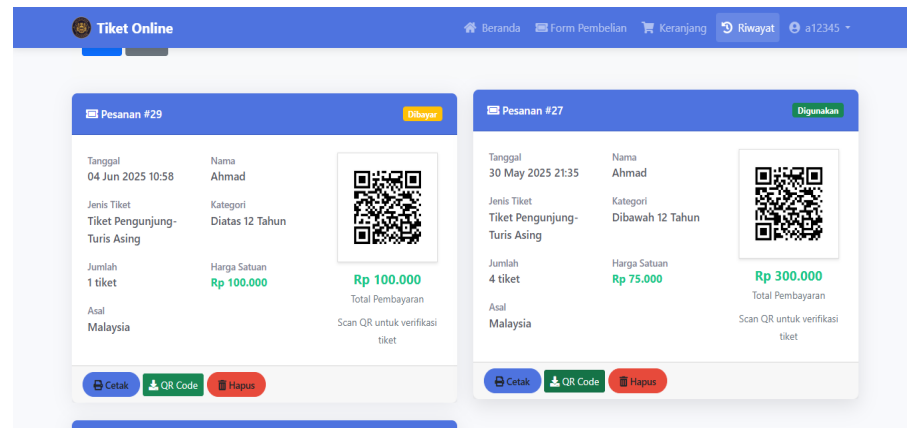
Tabel 3. 6 Analisis Kelayakan (Tabel Lanjutan)

Kriteria Kelayakan	Bobot	Alternatif
Kelayakan Biaya Operasional	20%	Membutuhkan biaya untuk pembutan website yang termasuk di dalamnya keamanan dan pemeliharaan website
Kelayakan Perawatan Sistem	20%	Perawatan sistem berguna ketika adanya pembaruan dan mencegah dari terjadinya kesalahan sistem.

3.3 Desain Sistem

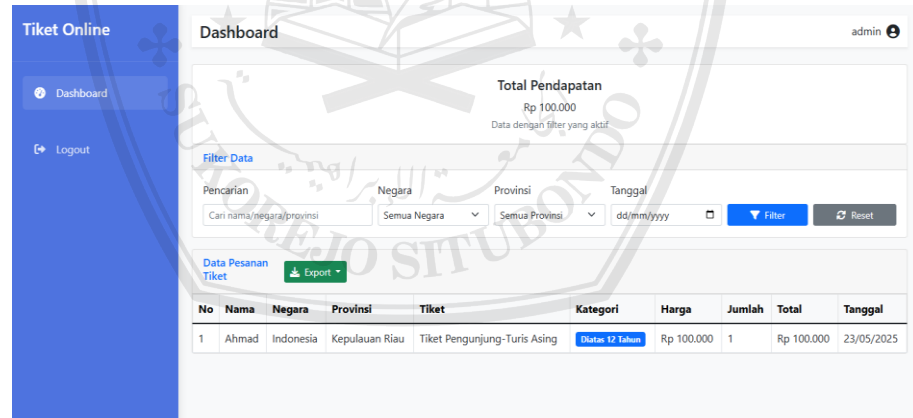
3.3.1 Desain Output

Desain output adalah rancangan tampilan dari hasil transaksi di dalam suatu sistem. Dalam sitem pembelian tiket ini akan menghasilkan output bagi *user* berupa invoice atau bukti pembayaran yang akan di berikan kepada penjaga loket nantinya, bahwa sudah benar-benar melakukan pembayaran.



Gambar 3. 5 Invoice Pembelian Tiket

Sedangkan berikut ini adalah gambar output yang berupa *list* pengunjung yang membeli tiket yang akan tampil pada website milik admin.



Gambar 3. 6 List Pengunjung

3.3.2 Desain Input

Pada desain input akan ditampilkan rancangan desain input yang mencakup semua hal yang berkaitan pengiriman data ke database sistem, berikut ini untuk lebih jelasnya mengenai desain input.

a. *Form Registrasi*

Pada gambar berikut merupakan tampilan registrasi untuk *user*. Jika pengguna atau *user* belum memiliki akun maka tidak akan bisa melakukan *login*. Oleh karena itu, bagi *user* harus membuat akun yang akan digunakan dalam proses *login* nantinya.

Gambar 3. 7 Registrasi

b. *Form Login*

Sebelum mengoperasikan website pembelian tiket maka semua pengguna baik *user* maupun admin diharuskan melakukan *login*. Gambar dibawah ini merupakan desain dari halaman *login* bagi *user* dan admin

Gambar 3. 8 Login

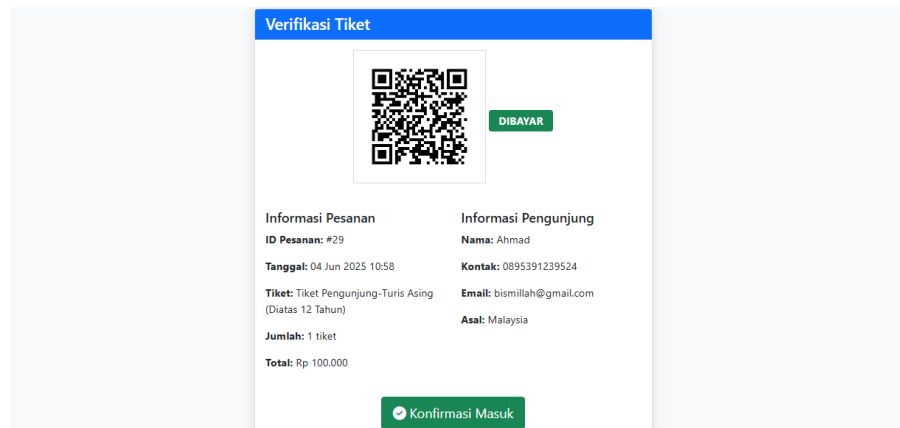
c. *Form Pemesanan Tiket*

Dalam menu pembelian tiket *user* diwajibkan mengisi semua *form* pembelian yang berisi identitas pembeli dan harga tiket akan otomatis mengikuti dari data pengunjung dari segi negara, provinsi dan usia dari pengunjung yang akan membeli tiket masuk taman sukasada ujung.

Gambar 3. 9 Halaman Menu Pembelian

d. *Form Cek Tiket*

Dalam *Form* cek tiket ini nantinya yang akan berfungsi mengecek kode QR yang sudah didapatkan *user* setelah melakukan pembayaran. Ketika akan masuk ke taman akan disiapkan *scan barcode* yang akan menampilkan apakah *user* sudah bayar atau belum.



Gambar 3. 10 Halaman Detail Pesanan

3.3.3 Desain Proses

Perancangan sistem yang akan dibuat disini adalah perancangan sistem *online*, jika nantinya ada pembelian *offline* maka akan dilayani langsung oleh penjaga loket dengan sistem manual. Disamping hal itu, kami menekankan mengenai pembayaran parkir itu diluar dari pembayaran tiket, karena pengelolaan tiket dan pengelolaan lahan parkir itu berbeda seksi. Dalam desain proses ini akan dibahas mengenai proses yang akan diterapkan dalam aplikasi yang akan dibuat. Berikut ini adalah hasil dari gambaran rancangan pemodelan sistem pembelian tiket *online*

a. Identifikasi Desain Proses

Untuk memperjelas desain proses pembelian tiket dan pembayaran tiket yang akan dilakukan oleh *user* atau pengunjung disajikan tabel dibawah ini

Tabel 3. 7 Proses Pembelian Tiket

Nama Proses	Deskripsi Proses	Input Proses	Output Proses
Proses Pembelian tiket oleh <i>user</i>	<i>User</i> yang melakukan pembelian tiket mengisi data diri.	Input data diri dan jumlah tiket yang akan dibeli.	Output data diri dan harga tiket yang harus dibayar oleh pengunjung atau <i>user</i>
Proses pembayaran tiket	<i>User</i> akan melakukan pembayaran tiket yang dipesan	Melakukan top up pembayaran melalui <i>payment gateway</i>	Print out hasil pembayaran tiket

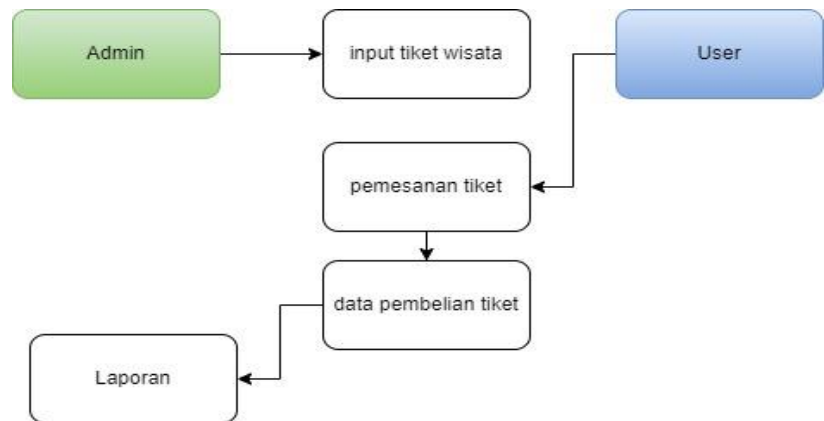
Tabel berikut akan menampilkan proses *scan barcode user* yang akan memunculkan status dari *user* sudah bayar atau belum.

Tabel 3. 8 *Scan Barcode*

Nama Proses	Deskripsi Proses	Input Proses	Output Proses
<i>Scan barcode</i>	<i>User</i> melakukan <i>scan barcode</i> yang didapat saat pembelian tiket	<i>Scanner</i> akan otomatis memeriksa kode QR	Status <i>user</i> sudah melakukan pembayaran atau belum

b. Arsitektur Aplikasi

Arsitektur aplikasi adalah gambaran tentang interaksi admin dan *user* dengan sistem yang dilakukan berdasarkan kebutuhan. Berikut adalah gambaran arsitektur pada sistem informasi pembelian tiket *online* di Taman Ujung Sukasada Karangasem Bali.

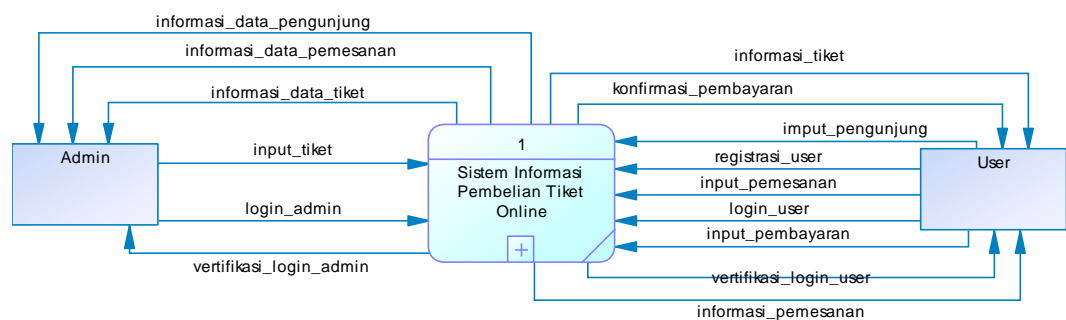


Gambar 3. 11 Arsitektur Aplikasi

c. Pemodelan Sistem

1. Context Diagram (CD)

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup dari sistem. Diagram Konteks merupakan level tertinggi dari *DFD* (*Data Flow Diagram*) yang menggambarkan seluruh input dan output dari sistem serta berfungsi memetakan model lingkungan.

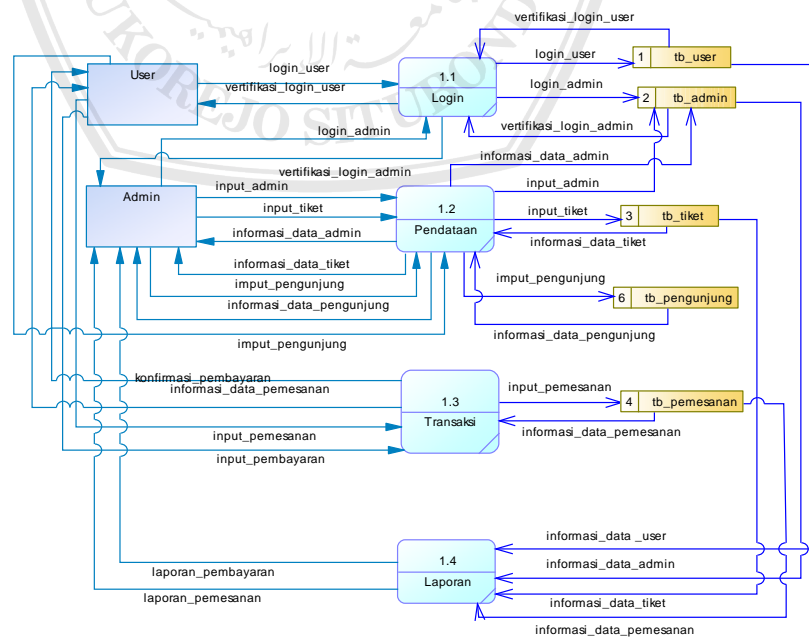


Gambar 3. 12 Context Diagram

Dari gambar diatas dijelaskan bahwa ada *user* yang berperan sebagai pelaku bisnis. Dimana *user* bisa melakukan registrasi, *login*, input pemesanan dan menerima konfirmasi pembayaran. Sedangkan admin akan menerima semua data pembelian tiket yang berguna sebagai bahan laporan pembelian tiket.

2. Data Flow Diagram (DFD)

Data Flow Diagram ini digambarkan bagaimana proses pengolahan data, seperti asal data, tujuan data, dimana disimpannya data dan hasil dari data tersebut. Disamping itu juga menampilkan interaksi antar data yang terjadi dalam sistem ini



Gambar 3. 13 Data Flow Diagram

3.3.4 Identifikasi dan Desain Database

a. Identifikasi Tabel Database

Berikut ini adalah tabel database yang harus dibuat dari sistem pembelian tiket yang akan dibuat yang meliputi tabel admin, tabel *user*, tabel tiket, tabel pengunjung tabel pesanan.

Tabel 3. 9 Database

Nama Tabel	Nama Field	Tipe Data	Lebar	Key
Admin	Id	int	11	Primary Key
	Username	varchar	25	
	password	varchar	25	
User	Id	int	11	Primary Key
	Username	varchar	25	
	Password	varchar	25	
	email	varchar	25	
Tiket	Id	int	11	Primay Key
	Nama_tiket	enum	"Tiket"	

Tabel 3. 9 Database (Tabel Lanjutan)

Nama Tabel	Nama Field	Tipe Data	Lebar	Key
			Pengunjung- Turis Asing', 'Tiket Pengunjung- Domestik', 'Tiket- Pengunjung- Lokal', 'Tiket Guide'	
	Kategori	enum	'Dibawah 12 Tahun', 'Diatas 12 Tahun'	
	Harga	int	11	

Tabel 3. 9 Database (Tabel Lanjutan)

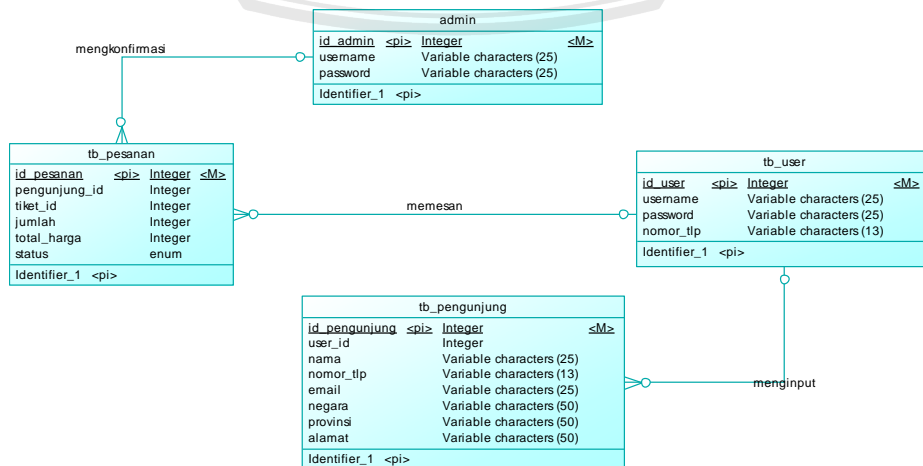
Nama Tabel	Nama Field	Tipe Data	Lebar	Key
pengunjung	id	int	11	Primary Key
	User_id	int	11	
	nama	varchar	25	
	Nomor_tlp	varchar	13	
	email	varchar	25	
	negara	varchar	50	
	provinsi	varchar	50	
pesanan	alamat	varchar	50	
	id	int	11	
	Pengunjung_id	int	11	
	Tiket_id	int	11	
	jumlah	int	11	
	Total_harga	int	11	

Tabel 3. 9 Database (Tabel Lanjutan)

Nama Tabel	Nama Field	Tipe Data	Lebar	Key
	Status	enum	'keranjang', 'dibayar', 'digunakan'	

b. Pemodelan Database

Untuk lebih mudah dipahami kami menyajikan pemodelan sistem dari sistem pembelian tiket *online* menggunakan ERD (*Entity Relationship Diagram*). *Entity Relationship Diagram* adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis



Gambar 3. 14 Entity Relationship Diagram

3.3.5 Identifikasi dan Desain *User Interface*

a. Identifikasi *Interface*

Tabel berikut ini menampilkan hasil dari identifikasi interface dari sistem informasi pembelian tiket yang akan dibuat.

Tabel 3. 10 Identifikasi *Interface*

Nama <i>Interface</i>	Jenis <i>Interface</i>	Bentuk <i>Interface</i>	Deskripsi
<i>Dashboard</i> Website	GUI	Gambar	Informasi singkat tentang taman meliputi gambar dan sejarah
Beranda Admin	GUI	Gambar	Halaman untuk melihat data pembelian
Beranda <i>User</i>	GUI	Gambar	Halaman untuk pembelian tiket
Halaman <i>Login</i>	GUI	Gambar	Halaman <i>login</i> untuk admin dan <i>user</i>
Halaman Registrasi	GUI	Gambar	Halaman registrasi <i>user</i> .

b. Desain *Interface*

1. *Dashboard* Website

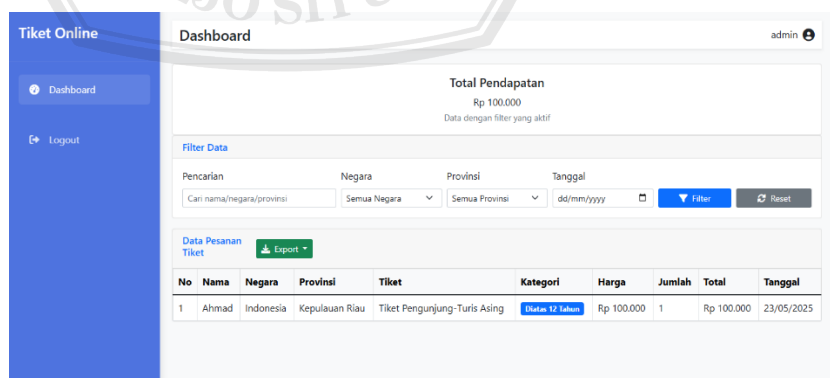
Halaman pertama yang akan muncul yaitu *dashboard* website yang menampilkan informasi singkat tentang taman dan menu *login* serta registrasi.



Gambar 3. 15 *Dashboard* Website

2. Beranda Admin

Dalam beranda admin akan menampilkan semua hasil pembelian tiket yang sudah berhasil.



Gambar 3. 16 *Dashboard* Admin

3. Beranda *User*

Beranda *user* akan menampilkan informasi singkat terkait Taman Sukasada menu *form* pembelian, keranjang, riwayat serta menu log out.



Gambar 3. 17 Beranda *User*

4. Halaman Registrasi

Bagi semua pengunjung yang belum mempunyai akun maka harus melakukan registrasi dengan memasukkan *username* dan *password*.

Gambar 3. 18 Registrasi

5. Halaman *Login*

Ketika sudah mempunyai akun pengunjung wajib melakukan *login* untuk bisa mengakses website dan melakukan pembelian tiket.



Gambar 3. 19 Login

BAB IV

IMPLEMENTASI SISTEM

4.1 Konstruksi Sistem

4.1.1 Kebutuhan Sistem

Webiste pembelian tiket taman ujung ini menggunakan bahasa *PHP* dan dengan template bootstrap serta menggunakan *MySQL* sebagai penyimpan data atau databasenya. Sistem informasi ini bisa dijalankan di semua versi OS, tapi disini yang digunakan yaitu Windows 10. Saat menjalankan website ini sangat perlu untuk memperhatikan perangkat lunak dan perangkat keras, karena kedua perngakat ini yang akan mendukung dalam melakukan penggunaan website ini. Berikut ini adalah perangkat-perangkat nag dibutuhkan.

a. Software

Software yang dibutuhkan untuk dalam mendukung website ini, membutuhkan beberapa aplikasi, berikut diantaranya.

1. Operasi Sistem Windows 10
2. *XAMPP*
3. Browser, contohnya chrome da sejenisnya
4. Visusal Studio Code

b. Hardware

Sedangkan hardware atau perangkat keras yang dibutuhkan adalah sebagai berikut.

1. PC atau laptop
2. Mouse
3. Keyboard
4. Hardisk 500GB
5. RAM 8 GB

4.1.2 Instalasi Sistem

a. Cara Instalasi *XAMPP*

Sebelum memulai proses instalasi *XAMPP*, ada beberapa hal penting yang perlu dipersiapkan agar pemasangan berjalan dengan lancar, pastikan komputer Anda memenuhi persyaratan minimum untuk menjalankan *XAMPP*:

1. Menggunakan sistem operasi Windows, macOS, atau Linux.
2. Memiliki RAM minimal 4 GB agar kinerja tetap optimal.
3. Menyediakan ruang penyimpanan kosong setidaknya 1 GB untuk file instalasi dan konfigurasi *XAMPP*.
4. Unduh *installer XAMPP* versi terbaru dari situs resmi untuk memastikan kompatibilitas yang maksimal.

Setelah semua persiapan di atas dilakukan, Anda dapat melanjutkan ke tahap instalasi. Panduan berikut akan menjelaskan cara menginstal *XAMPP* pada sistem operasi Windows.

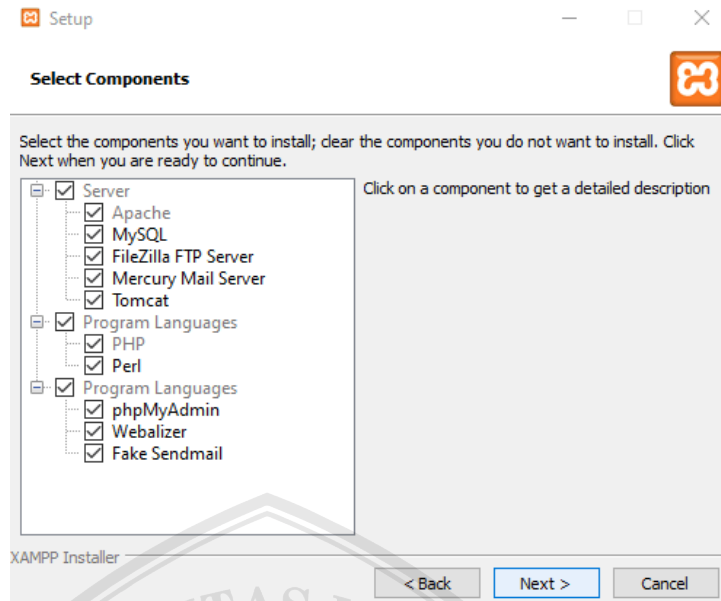
Menginstal *XAMPP* di Windows cukup mudah dan hanya memerlukan beberapa langkah sederhana. Berikut langkah-langkahnya:

1. Unduh *installer XAMPP* dari situs resmi Apache Friends dan pilih versi untuk Windows.



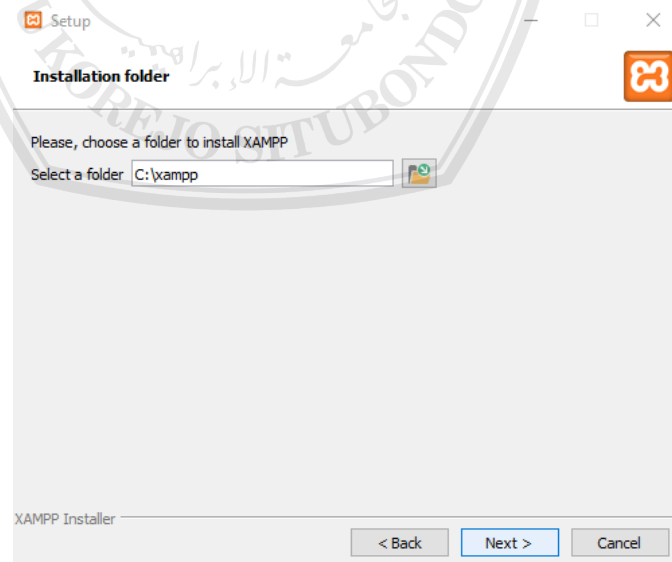
Gambar 4.1 *Install XAMPP*

2. Jalankan file *installer* yang telah diunduh dengan klik dua kali untuk memulai proses instalasi.
3. Pilih komponen yang ingin Anda instal, seperti Apache, *MySQL*, dan *PHP*.



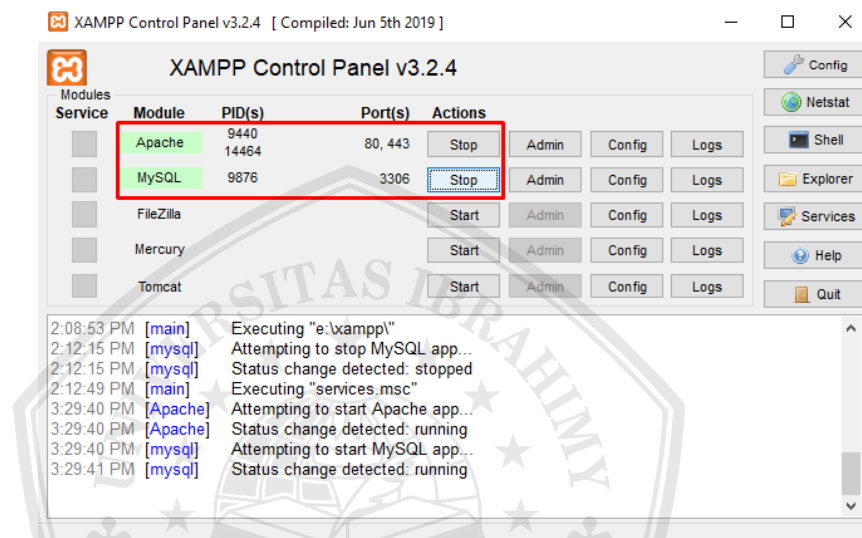
Gambar 4. 2 Pilih Komponen XAMPP

4. Tentukan lokasi instalasi (secara default di *C:\xampp*), kemudian klik **Next** untuk melanjutkan.



Gambar 4. 3 Lokasi Instalasi

5. Tunggu hingga proses instalasi selesai, lalu klik **Finish** untuk membuka *XAMPP* Control Panel.
6. Aktifkan layanan Apache dan *MySQL* melalui *XAMPP* Control Panel.



Gambar 4. 4 Layanan XAMPP

7. Buka browser dan akses alamat <http://localhost> untuk memastikan bahwa instalasi berhasil.

Setelah proses ini selesai, Anda sudah dapat mulai membuat dan menguji situs web langsung dari server lokal Anda.

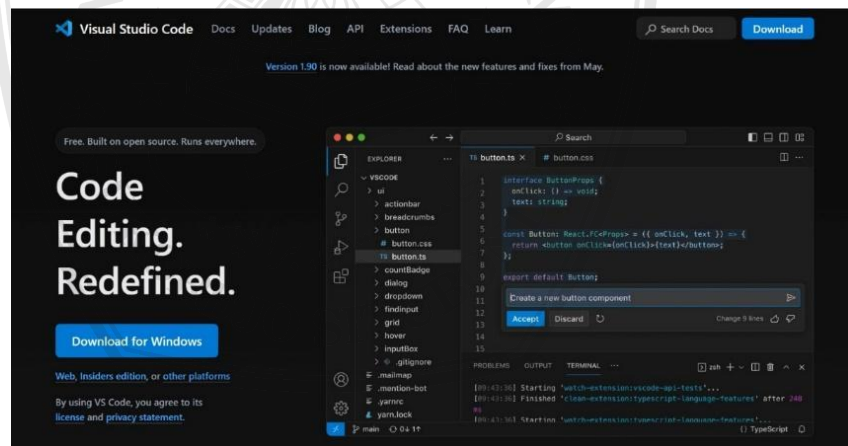
- b. Cara Instalasi *Visual Studio Code*

Visual Studio Code (VS Code) merupakan salah satu editor kode paling populer dan serbaguna yang banyak digunakan oleh para pengembang. Dengan dukungan untuk berbagai bahasa

pemrograman, antarmuka yang ramah pengguna, serta ribuan ekstensi, VS Code menjadi pilihan ideal baik untuk pemula maupun profesional. Artikel ini akan memandu Anda melalui langkah-langkah instalasi VS Code di sistem operasi Windows. Berikut langkah-langkah instalasinya.

1. Akses Situs Resmi *Visual Studio Code*

Buka browser Anda dan kunjungi laman <https://code.visualstudio.com/>. Di halaman utama, klik tombol “Download for Windows” yang terletak di sisi kiri untuk memulai proses unduh.



Gambar 4.5 Website *Visual Studio Code*

2. Proses Unduhan

Setelah tombol diklik, halaman baru akan muncul yang menandakan bahwa proses pengunduhan telah dimulai.



Gambar 4.6 Mengunduh *Visual Studio Code*

Jika unduhan tidak berjalan otomatis, Anda dapat menggunakan direct download link yang tersedia. *File installer* akan muncul di riwayat unduhan browser Anda.



Gambar 4.7 Proses Unduhan *Visual Studio Code*

3. Mulai Proses Instalasi

Setelah file berhasil diunduh, klik dua kali pada file tersebut di riwayat unduhan atau melalui folder Downloads di *File Manager* untuk memulai instalasi.

4. Setujui Persyaratan Lisensi

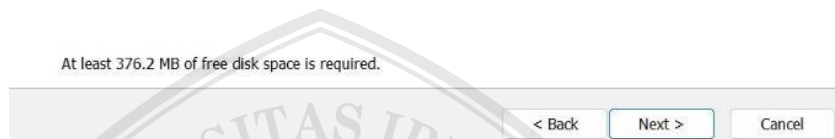
Akan muncul jendela persetujuan lisensi. Bacalah dengan cermat, kemudian centang kotak “I accept the agreement” dan klik Next untuk melanjutkan.



Gambar 4. 8 Proses *Install Visual Studio Code*

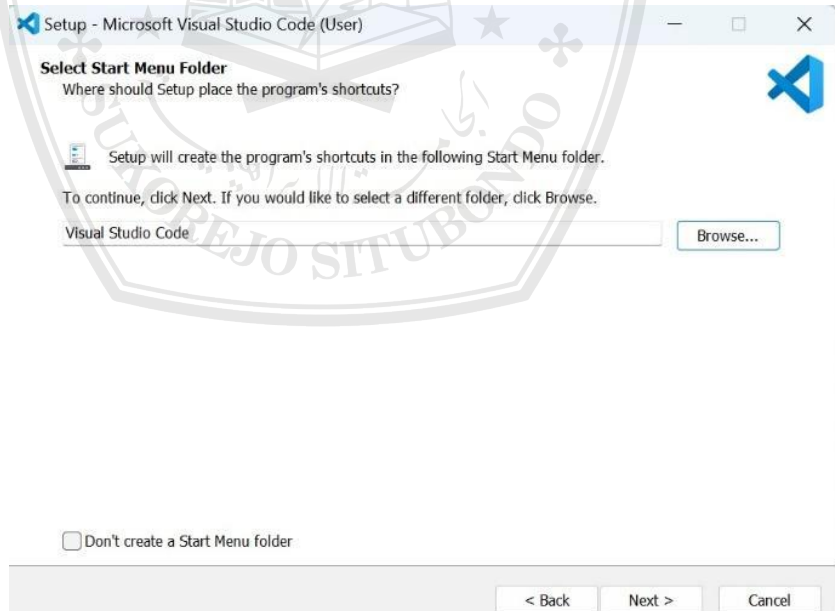
5. Tentukan Lokasi Instalasi

Secara default, VS Code akan dipasang di folder Program Files. Anda bisa mengganti lokasi instalasi dengan menekan tombol Browse, lalu pilih folder sesuai keinginan Anda. Setelah itu, klik Next.



Gambar 4.9 Lokasi Folder Instalasi

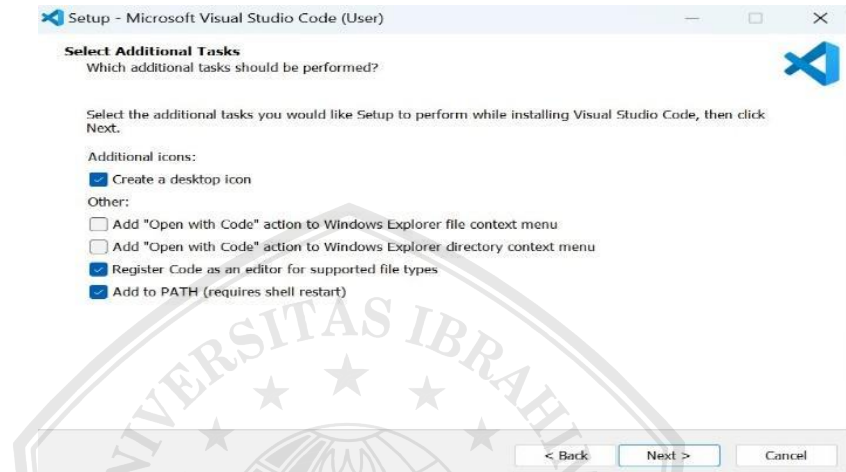
6. Pilih Folder *Start* Menu



Gambar 4.10 Folder *Start* Menu

Tentukan folder untuk shortcut pada menu *Start*. Anda bisa menggunakan nama folder bawaan atau memilih lokasi lain yang tersedia.

7. Pilih Komponen Tambahan

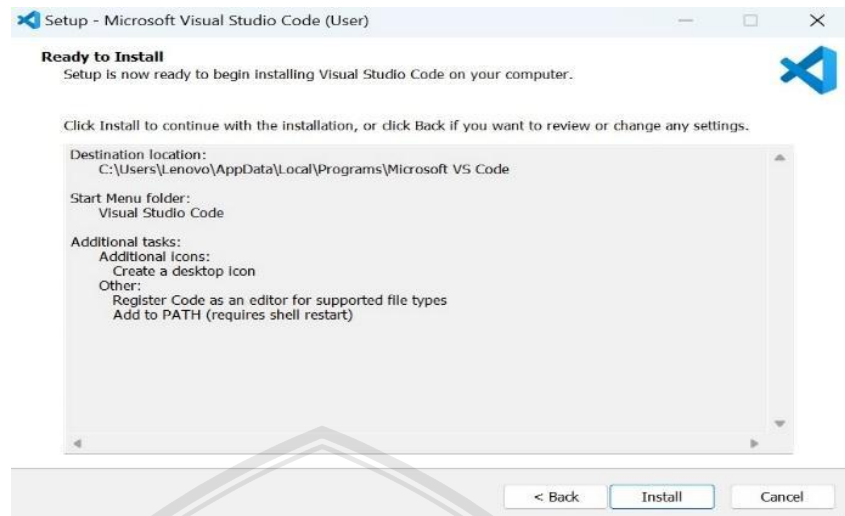


Gambar 4. 11 Pilih Komponen *Visual Studio Code*

Pada tahap ini, Anda dapat memilih beberapa opsi tambahan, seperti:

- a) Menambahkan VS Code ke PATH
- b) Membuat shortcut di desktop
- c) Menghubungkan VS Code dengan jenis file tertentu
- d) Centang opsi yang Anda perlukan, kemudian klik Next.

8. Mulai Proses Instalasi



Gambar 4. 12 Proses *Install Visual Studio Code*

Klik tombol *Install* untuk memulai pemasangan VS Code.

Tunggu hingga prosesnya selesai.

9. Selesai



Gambar 4. 13 Tahap Akhir *Install Visual Studio Code*

Setelah instalasi selesai, klik *Finish*. *Visual Studio Code* kini siap digunakan.

c. Cara Instalasi *Chrome*

1. Buka Situs Google *Chrome*

Unduh browser perusahaan yang paling terpercaya

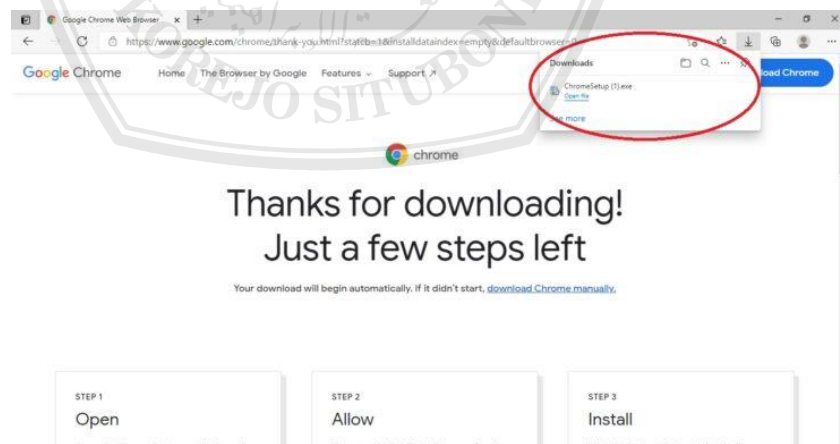
Chrome Enterprise menghadirkan kontrol perusahaan dan integrasi mudah ke browser yang sudah dikenal dan disukai pengguna. Dapatkan insight penting dan [mulai kelola Chrome](#) sekarang.

Download Chrome

Gambar 4. 14 Situs *Chrome*

Akses halaman resmi Google *Chrome* melalui tautan <https://www.google.com/chrome/>, kemudian klik tombol *Download Chrome*. *File* instalasi akan otomatis terunduh ke perangkat Anda.

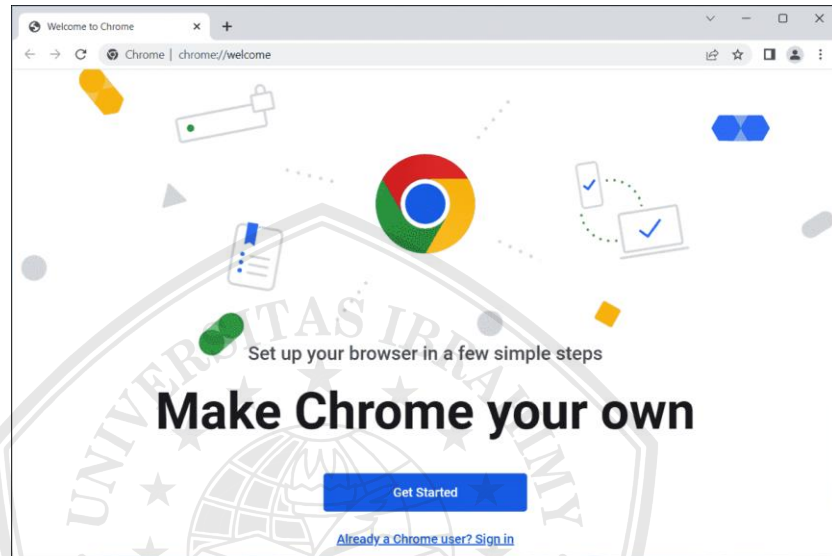
2. Jalankan *File* Instalasi



Gambar 4. 15 *File Chrome*

Setelah proses unduhan selesai, buka file *ChromeSetup* yang telah diunduh untuk memulai instalasi. Proses pengunduhan lanjutan akan berlangsung, diikuti oleh proses instalasi otomatis.

3. Instalasi Selesai



Gambar 4. 16 *Install Chrome Selesai*

Setelah instalasi selesai, Google *Chrome* akan terbuka secara otomatis. Klik *Get Started* atau masuk dengan akun Google Anda untuk mulai menggunakan browser ini.

4.1.3 Segmen Program

Segmen program adalah rangkaian sebuah pernyataan yang ditulis dalam bahasa pemrograman yang dapat dipahami oleh manusia.

a. Segmen Koneksi

Tabel 4. 1 Tabel Segmen Koneksi

```

<?php
// File: koneksi.php

$host = 'localhost';      // Server database
$user = 'root';          // Username database
$password = '';          // Password database
$dbase = 'tiket_online'; // Nama database

// Membuat koneksi
$conn = new mysqli($host, $user, $password,
$dbase);

// Cek koneksi
if ($conn->connect_error) {
    die("Koneksi gagal: " . $conn-
>connect_error);
}
...
?>

```

File koneksi.php merupakan file inti yang menghubungkan aplikasi dengan database *MySQL*. Pada bagian konfigurasi, parameter koneksi database didefinisikan meliputi *\$host* (server database, biasanya 'localhost' untuk pengembangan), *\$user* (username database seperti 'root' untuk *XAMPP/MAMP*), *\$password* (kosong untuk pengembangan lokal), dan *\$database* (nama database 'tiket_online'). Koneksi dibuat menggunakan objek *mysqli* dengan parameter yang telah ditentukan, dilengkapi pengecekan error yang akan

menghentikan eksekusi script dan menampilkan pesan error jika koneksi gagal.

b. Segmen Registrasi

Tabel 4. 2 Segmen Registrasi

```
<?php
session_start();
session_destroy();
include 'koneksi.php';

$error = '';

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = trim($_POST['username']);
    $email = trim($_POST['email']);
    $password = trim($_POST['password']);
    $confirm_password =
trim($_POST['confirm_password']);
    // Validasi input
    if (empty($username) || empty($email) ||
empty($password) || empty($confirm_password)) {
        $error = "Semua field harus diisi!";
    } elseif (strlen($username) < 5) {
        $error = "Username minimal 5 karakter!";
    } elseif (!filter_var($email,
FILTER_VALIDATE_EMAIL)) {
        $error = "Format email tidak valid!";
    } elseif (strlen($password) < 6) {
        $error = "Password minimal 6 karakter!";
    } elseif ($password !== $confirm_password) {
        $error = "Password dan konfirmasi
password tidak sama!";
    } else {
        // Cek username dan email sudah ada atau
belum
        $stmt = $conn->prepare("SELECT id FROM
user WHERE username = ? OR email = ?");
        $stmt->bind_param("ss", $username,
$email);
```

Tabel 4. 2 Segmen Registrasi (Tabel Lanjutan)

```

$stmt->execute();
    $result = $stmt->get_result();
    if ($result->num_rows > 0) {
        $error = "Username atau email sudah
terdaftar!";
    } else {
        // Hash password
        $hashed_password =
password_hash($password, PASSWORD_DEFAULT);

        // Simpan ke database
        $stmt = $conn->prepare("INSERT INTO
user (username, email, password) VALUES (?, ?,
?)");
        $stmt->bind_param("sss", $username,
$email, $hashed_password);

        if ($stmt->execute()) {
            $_SESSION['register_success'] =
"Pendaftaran berhasil! Silakan login dengan akun
Anda.";
            header("Location: login.php");
            exit();
        } else {
            $error = "Terjadi kesalahan saat
mendaftar. Silakan coba lagi.";
        }
    }
}
}
?>
<!DOCTYPE html>
<html lang="id">
...
</div>

<script src="https://code.jquery.com/jquery-
3.6.0.min.js"></script>
<script>
    $(document).ready(function() {

```

Tabel 4. 2 Segmen Registrasi (Tabel Lanjutan)

```
// Form submission animation

$('#registerForm').on('submit', function() {
    $('#button[type="submit"]').html(
    '<i class="fas fa-spinner fa-spin me-
    2"></i>Mendaftarkan...');
    $('#button[type="submit"]').prop(
    'disabled', true);
    });
});
</script>
</body>
</html>
```

File register.php memproses pendaftaran *user* dengan validasi ketat: cek kelengkapan data, panjang *username* (min 5 karakter), *format* email, panjang password (min 6 karakter), dan kecocokan password. Jika valid, sistem cek duplikasi *username*/email di database, hash password, lalu simpan data. Berhasil akan redirect ke *login* dengan pesan sukses, gagal tampilkan error. Dilengkapi fitur keamanan (prepared statement, password hashing) dan animasi loading saat submit.

c. Segmen *Login*

Tabel 4. 3 Segmen *Login*

```
<?php
session_start();
include 'koneksi.php';

if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $username = trim($_POST['username']);
```

Tabel 4.3 Segmen *Login* (Tabel Lanjutan)

```

$password = trim($_POST['password']);

$user_type = isset($_POST['user_type']) ?
$_POST['user_type'] : 'user';

// Hapus semua session sebelumnya
$_SESSION = array();
session_destroy();
session_start();

// Tentukan tabel berdasarkan tipe user
$table = ($user_type == 'admin') ? 'admin' :
'user';

$stmt = $conn->prepare("SELECT id, username,
password FROM $table WHERE username = ?");
$stmt->bind_param("s", $username);
$stmt->execute();
$result = $stmt->get_result();

if ($result->num_rows === 1) {
    $user = $result->fetch_assoc();
    if (password_verify($password,
$user['password'])) {
        // Regenerate session ID untuk
mencegah fixation
        session_regenerate_id(true);

        // Set session baru
        $_SESSION['user_id'] = $user['id'];
        $_SESSION['username'] =
$user['username'];
        $_SESSION['loggedin'] = true;
        $_SESSION['user_type'] = $user_type;
// Tambahkan tipe user ke session
        // Set session success message
        $_SESSION['success'] = "Login
berhasil! Selamat datang " .
htmlspecialchars($user['username']);

```

Tabel 4. 3 Segmen *Login* (Tabel Lanjutan)

```
// Redirect berdasarkan tipe user
    $redirect = ($user_type == 'admin')
? 'admin/dashboard.php' : 'index.php';
    header("Location: $redirect");

exit();
    }
}

// Jika username/password salah
$_SESSION['error'] = "Username atau password
salah";
    $redirect = ($user_type == 'admin') ?
'login.php?admin=1' : 'login.php';
    header("Location: $redirect");
    exit();
}

// Cek apakah login sebagai admin
$is_admin_login = isset($_GET['admin']) &&
$_GET['admin'] == '1';
?>

<!DOCTYPE html>
<html lang="id">
<head>
...
<script>
        document.addEventListener('DOMContent
tLoaded', function() {
            const notifications =
document.querySelectorAll('.notification');
            const navbar =
document.querySelector('.navbar');
            const navbarHeight = navbar ?
navbar.offsetHeight : 60;

            notifications.forEach(notificati
on => {
                // Set posisi notifikasi di bawah navbar
```

Tabel 4. 3 Segmen *Login* (Tabel Lanjutan)

```
notification.style.top = `${navbarHeight +
20}px`;

        // Tampilkan notifikasi
        setTimeout(() => {
            notification.classList.add('
active');}, 100);

        // Auto close setelah 5
detik

        setTimeout(() => {
            notification.classList.r
emove('active');

            setTimeout(() =>
notification.remove(), 500);
        }, 5000);
    });

    // Close button manual
    document.querySelectorAll('.noti
fication-close').forEach(button => {
        button.addEventListener('cli
ck', function() {
            const notification =
this.closest('.notification');
            notification.classList.r
emove('active');

            setTimeout(() =>
notification.remove(), 500);
        });
    });
});
</script>
</body>
</html>
```

File login.php merupakan halaman autentikasi yang mendukung *login* ganda untuk *user* dan *admin*. Sistem ini menerapkan beberapa lapis keamanan seperti hashing password, regenerasi session

ID, dan sanitasi input untuk mencegah serangan. Proses *login* dimulai dengan validasi kredensial di database, kemudian memberikan respons berbeda - mengarahkan *user* ke halaman beranda atau admin ke *dashboard* jika berhasil, atau menampilkan notifikasi error jika gagal.

d. Segmen *Logout*

File logout.php menangani proses terminasi session pengguna secara aman dengan tiga langkah utama yaitu menghapus semua data session, membersihkan cookie session dari browser, dan menghancurkan session server-side. Sistem ini juga mengirim pesan konfirmasi *logout* yang akan ditampilkan di halaman *login* setelah redirect, dengan perlindungan melalui `urlencode()` untuk memastikan keamanan parameter URL. Proses ini dirancang untuk mencegah session hijacking dan memastikan pengguna benar-benar *logout* dari sistem.

Tabel 4. 4 Segmen *Logout*

```
<?php
session_start();

// Simpan pesan logout sebelum menghancurkan
session
$logout_message = "Anda berhasil logout. Sampai
jumpa kembali!";

// Hapus semua data session
$_SESSION = array();
```

Tabel 4. 4 Segmen *Logout* (Tabel Lanjutan)

```
// Hapus cookie session

if (ini_get("session.use_cookies")) {

$params = session_get_cookie_params();
    setcookie(session_name(), '', time() -
42000,
        $params["path"],
        $params["domain"],
        $params["secure"],
        $params["httponly"]
    );
}

// Hancurkan session
session_destroy();

// Redirect dengan parameter GET
header("Location: login.php?logout=1&message="
. urlencode($logout_message));
exit();
?
```

e. Segmen *Form* Pembelian**Tabel 4. 5 Segmen *Form* Pembelian**

```
<?php
session_start();
ob_start();
include 'koneksi.php';

// Periksa apakah user sudah login
if (!isset($_SESSION['user_id'])) {
    $_SESSION['redirect_after_login'] =
'form_pembelian.php';
    header("Location: login.php");
    exit();
}
```

Tabel 4. 5 Segmen *Form* Pembelian (Tabel Lanjutan)

```
// Ambil user_id dari session
$user_id = $_SESSION['user_id'];

// Daftar negara di dunia
$negara_dunia = [
    'Indonesia',...
];
// Daftar provinsi di Indonesia
$provinsi_indonesia = [
    'Aceh', ...
];

// Proses form
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Ambil dan sanitasi input
    $nama = mysqli_real_escape_string($conn,
$_POST['nama']);
    $telpon = mysqli_real_escape_string($conn,
$_POST['telpon']);
    $email = filter_var($_POST['email'],
FILTER_SANITIZE_EMAIL);
    $negara = mysqli_real_escape_string($conn,
$_POST['negara']);
    $provinsi = ($negara == 'Indonesia') ?
mysqli_real_escape_string($conn,
$_POST['provinsi']) : 'Luar Negeri';
    $tiket = mysqli_real_escape_string($conn,
$_POST['nama_tiket']);
    $kategori = mysqli_real_escape_string($conn,
$_POST['kategori']);
    $jumlah = intval($_POST['jumlah']);
    // Validasi kelengkapan data
    if (empty($nama) || empty($telpon) ||
empty($email) || empty($negara) ||
($negara == 'Indonesia' &&
empty($provinsi)) || empty($tiket) ||
empty($kategori) || $jumlah <= 0) {
        echo "<div class='alert alert-
warning'>Harap lengkapi semua data dengan
benar!</div>";
        exit();}
}
```

Tabel 4. 5 Segmen Form Pembelian (Tabel Lanjutan)

```
// Ambil harga tiket
$query_tiket = "SELECT id, harga FROM
tb_tiket WHERE nama_tiket = ? AND kategori = ?";
$stmt_tiket = mysqli_prepare($conn,
$query_tiket);
mysqli_stmt_bind_param($stmt_tiket, "ss",
$tiket, $kategori);
mysqli_stmt_execute($stmt_tiket);
$result_tiket =
mysqli_stmt_get_result($stmt_tiket);

if (mysqli_num_rows($result_tiket) > 0) {
    $row_tiket =
mysqli_fetch_assoc($result_tiket);
    $tiket_id = $row_tiket['id'];
    $harga = $row_tiket['harga'];
    $total = $harga * $jumlah;
} else {
    echo "<div class='alert alert-
danger'>Tiket tidak ditemukan dalam
database</div>";
    exit();
}
mysqli_stmt_close($stmt_tiket);

// Cek apakah pengunjung sudah ada
berdasarkan email DAN user_id
$query_pengunjung = "SELECT id FROM
pengunjung WHERE email = ? AND user_id = ? LIMIT
1";
$stmt_pengunjung = mysqli_prepare($conn,
$query_pengunjung);
mysqli_stmt_bind_param($stmt_pengunjung,
"si", $email, $user_id);
mysqli_stmt_execute($stmt_pengunjung);
$result_pengunjung =
mysqli_stmt_get_result($stmt_pengunjung);

if (mysqli_num_rows($result_pengunjung) > 0)
{
```

Tabel 4. 5 Segmen Form Pembelian (Tabel Lanjutan)

```
// Pengunjung sudah ada

$pengunjung =
mysqli_fetch_assoc($result_pengunjung);
    $pengunjung_id = $pengunjung['id'];
    // Update data pengunjung jika ada
perubahan
    $update_pengunjung = "UPDATE pengunjung
SET nama = ?, no_telpon = ?, negara = ?,
provinsi = ? WHERE id = ?";
    $stmt_update_pengunjung =
mysqli_prepare($conn, $update_pengunjung);
    mysqli_stmt_bind_param($stmt_update_peng
unjung, "ssssi", $nama, $telpon, $negara,
$provinsi, $pengunjung_id);
    mysqli_stmt_execute($stmt_update_pengunj
ung);
    mysqli_stmt_close($stmt_update_pengunjun
g);
} else {
    // Pengunjung baru, simpan data dengan
user_id yang sesuai
    $insert_pengunjung = "INSERT INTO
pengunjung (nama, no_telpon, email, negara,
provinsi, user_id) VALUES (?, ?, ?, ?, ?, ?)";
    $stmt_insert_pengunjung =
mysqli_prepare($conn, $insert_pengunjung);
    mysqli_stmt_bind_param($stmt_insert_peng
unjung, "sssss", $nama, $telpon, $email,
$negara, $provinsi, $user_id);
    mysqli_stmt_execute($stmt_insert_pengunj
ung);

    $pengunjung_id =
mysqli_insert_id($conn);
    mysqli_stmt_close($stmt_insert_pengunjun
g);}
    mysqli_stmt_close($stmt_pengunjung);
    // Cek apakah tiket yang sama sudah ada di
keranjang untuk pengunjung ini
    $check_pesanan = "SELECT id, jumlah FROM
tb_pesanan WHERE pengunjung_id = ? AND tiket_id
= ? AND status = 'keranjang'";
```

Tabel 4. 5 Segmen Form Pembelian (Tabel Lanjutan)

```
$stmt_check_pesanan = mysqli_prepare($conn,
$check_pesanan);
mysqli_stmt_bind_param($stmt_check_pesanan,
"ii", $pengunjung_id, $tiket_id);
mysqli_stmt_execute($stmt_check_pesanan);
$result_pesanan =
mysqli_stmt_get_result($stmt_check_pesanan);

if (mysqli_num_rows($result_pesanan) > 0) {
// Tiket sudah ada, update jumlah
$pesanan =
mysqli_fetch_assoc($result_pesanan);
$new_jumlah = $pesanan['jumlah'] +
$jumlah;
$new_total = $harga * $new_jumlah;

$update_pesanan = "UPDATE tb_pesanan SET
jumlah = ?, total_harga = ? WHERE id = ?";
$stmt_update_pesanan =
mysqli_prepare($conn, $update_pesanan);
mysqli_stmt_bind_param($stmt_update_pesa
nan, "idi", $new_jumlah, $new_total,
$pesanan['id']);
mysqli_stmt_execute($stmt_update_pesanan
);
mysqli_stmt_close($stmt_update_pesanan);
} else {
// Tiket baru, buat pesanan
$insert_pesanan = "INSERT INTO
tb_pesanan (pengunjung_id, tiket_id, jumlah,
total_harga, status) VALUES (?, ?, ?, ?,
'keranjang)";
$stmt_insert_pesanan =
mysqli_prepare($conn, $insert_pesanan);
mysqli_stmt_bind_param($stmt_insert_pesa
nan, "iiid", $pengunjung_id, $tiket_id, $jumlah,
$total);
mysqli_stmt_execute($stmt_insert_pesanan
);
mysqli_stmt_close($stmt_insert_pesanan);
}
mysqli_stmt_close($stmt_check_pesanan);
```

Tabel 4. 5 Segmen *Form* Pembelian (Tabel Lanjutan)

```
// Notifikasi sukses
$_SESSION['notifikasi'] = [
    'judul' => 'Pembelian Berhasil!',
    'pesan' => 'Tiket berhasil ditambahkan
ke keranjang belanja',
    'tipe' => 'sukses',
    'waktu' => 2
];

ob_end_clean();
header("Location: keranjang.php");
exit();
}
ob_end_flush();
?>

<!DOCTYPE html>
<html lang="id">
...
// Hitung harga saat ada perubahan
$('#nama_tiket, #kategori, #negara,
#provinsi, #jumlah').on('change input',
hitungHarga);

// Fungsi hitung harga
function hitungHarga() {
    const tiket =
$('#nama_tiket').val();
    const kategori =
$('#kategori').val();
    const negara = $('#negara').val();
    const provinsi = (negara ===
'Indonesia') ? $('#provinsi').val() : '';
    const jumlah =
parseInt($('#jumlah').val()) || 0;
    let harga = 0;
    if (tiket && kategori && negara &&
(negara !== 'Indonesia' || provinsi)) {
        if (tiket === 'Tiket Guide') {
            harga = (kategori ===
'Diatas 12 Tahun') ? 20000 : 10000;
```

Tabel 4. 5 Segmen *Form* Pembelian (Tabel Lanjutan)

```
        } else {
            if (negara !== 'Indonesia')
            {
                harga = (kategori ===
'Diatas 12 Tahun') ? 100000 : 75000;
                } else if (provinsi !==
'Bali') {
                    harga = (kategori ===
'Diatas 12 Tahun') ? 35000 : 25000;
                    } else {
                        harga = (kategori ===
'Diatas 12 Tahun') ? 25000 : 15000;
                    }
                }
            $('#tampilanHarga').text(formatR
upiah(harga));
            $('#harga').val(harga);
            const total = harga * jumlah;
            $('#tampilanTotal').text(formatR
upiah(total));
            $('#total').val(total);
        } else {
            $('#tampilanHarga').text('Rp
0');
            $('#tampilanTotal').text('Rp
0');
            $('#harga').val('');
            $('#total').val('');
        }
    }
    // Format mata uang Rupiah
    function formatRupiah(angka) {
        return 'Rp ' +
angka.toString().replace(/\B(?=(\d{3})+(?!\d))/g
, '.');
    }

    // Hitung harga awal
    hitungHarga();
```

Tabel 4. 5 Segmen *Form* Pembelian (Tabel Lanjutan)

```

// Isi form dengan data pengunjung jika tersedia
(dari session)
    <?php
        if (isset($_SESSION['user_id'])) {
            // Ambil data pengunjung terakhir
            untuk user ini (opsional)
                $query_last_visitor = "SELECT * FROM
pengunjung WHERE user_id = ? ORDER BY id DESC
LIMIT 1";
                $stmt_last_visitor =
mysqli_prepare($conn, $query_last_visitor);
                mysqli_stmt_bind_param($stmt_last_vi
sitor, "i", $user_id);
                mysqli_stmt_execute($stmt_last_visit
or);
                $result_last_visitor =
mysqli_stmt_get_result($stmt_last_visitor);
                if
(mysqli_num_rows($result_last_visitor) > 0) {
                    $visitor =
mysqli_fetch_assoc($result_last_visitor);
                    ?>
                    $('#nama').val("<?=  

htmlspecialchars($visitor['nama']) ?>");
                    $('#telpon').val("<?=  

htmlspecialchars($visitor['no_telpon']) ?>");
                    $('#email').val("<?=  

htmlspecialchars($visitor['email']) ?>");
                    // Set negara & provinsi
                    $('#negara').val("<?=  

htmlspecialchars($visitor['negara'])
?>").trigger('change');
                    <?php if ($visitor['negara'] ==
'Indonesia'): ?>
                        setTimeout(function() {
                            $('#provinsi').val("<?=  

htmlspecialchars($visitor['provinsi'])
?>").trigger('change');
                        }, 300);
                    <?php endif; ?>

```

Tabel 4. 5 Segmen *Form* Pembelian (Tabel Lanjutan)

```
<?php
    }
    mysqli_stmt_close($stmt_last_visitor
);
    }
    ?>
    });
</script>
<?php include 'footer.php'; ?>
```

Form Pembelian Tiket Online adalah sistem pemesanan digital yang aman dan mudah digunakan, dengan verifikasi *login* melalui session management sebelum mengakses *form* pembelian lengkap berisi validasi real-time data pribadi, kontak, dan asal daerah. Fitur utamanya mencakup harga dinamis berdasarkan jenis tiket (Pengunjung/Guide), kategori usia (dibawah/diatas 12 tahun), dan wilayah asal (Bali/luar Bali/mancanegara). Dibangun dengan *PHP* (prepared statement), *MySQL*, *jQuery*, dan *Bootstrap*, sistem ini menyimpan data pengunjung, termasuk update tiket sudah ada di keranjang, dilengkapi auto-fill data riwayat, tampilan harga real-time *format* Rupiah, dan notifikasi sukses untuk pengalaman pengguna yang optimal.

f. Segmen Keranjang

Tabel 4. 6 Segmen Keranjang

```

<?php
session_start();
include 'koneksi.php';

if (isset($_GET['delete'])) {
    $id = $_GET['delete'];

    // Hanya hapus pesanan yang dipilih
    mysqli_query($conn, "DELETE FROM tb_pesanan
WHERE id=$id");

    $_SESSION['notifikasi'] = [
        'judul' => 'Berhasil Dihapus!',
        'pesan' => 'Item telah berhasil dihapus
dari keranjang',
        'tipe' => 'sukses'
    ];

    header("Location: keranjang.php");
    exit();
}

if (isset($_GET['bayar'])) {
    $id = $_GET['bayar'];
    mysqli_query($conn, "UPDATE tb_pesanan SET
status='dibayar' WHERE id=$id");

    // midtrass

    $_SESSION['notifikasi'] = [
        'judul' => 'Pembayaran Berhasil!',
        'pesan' => 'Pembayaran tiket telah
berhasil diproses',
        'tipe' => 'sukses',
        'redirect' => 'riwayat.php'
    ];

    header("Location: keranjang.php");

    exit();}

```

Tabel 4. 6 Segmen Keranjang (Tabel Lanjutan)

```
// Ganti query di keranjang.php dengan:
$user_id = $_SESSION['user_id'] ?? 0;
$where_clause = "WHERE tb_pesanan.status =
'keranjang' AND pengunjung.user_id = $user_id";
if (!empty($search)) {
    $where_clause .= " AND (pengunjung.nama LIKE
'%" . $search . "%'
                                OR pengunjung.email LIKE
'%" . $search . "%'
                                OR pengunjung.no_telpon
LIKE '%" . $search . "%'
                                OR tb_tiket.nama_tiket
LIKE '%" . $search . "%'
                                OR tb_tiket.kategori LIKE
'%" . $search . "%')";
}
$data = mysqli_query($conn, "SELECT
                                tb_pesanan.id AS
pesan_id,
                                pengunjung.nama,
                                pengunjung.no_telpon
,
                                pengunjung.email,
                                pengunjung.negara,
                                pengunjung.provinsi,
                                tb_tiket.nama_tiket,
                                tb_tiket.kategori,
                                tb_tiket.harga,
                                tb_pesanan.jumlah,
                                tb_pesanan.total_har
ga
                                FROM tb_pesanan
                                JOIN pengunjung ON
tb_pesanan.pengunjung_id = pengunjung.id
                                JOIN tb_tiket ON
tb_pesanan.tiket_id = tb_tiket.id
                                $where_clause
                                ORDER BY tb_pesanan.id
DESC");
?>
```

Tabel 4. 6 Segmen Keranjang (Tabel Lanjutan)

```
<!-- midtranspunya -->

<script type="text/javascript"
  src="https://app.sandbox.midtrans.com/snap
/snap.js"
  data-client-key="SB-Mid-client-
rb3Ub_oxVDYUVqUM"></script>

<!-- sipp -->
<!DOCTYPE html>
<html lang="id">
...
function tampilkanNotifikasi(judul, pesan, tipe)
{
  const icon = {
    'sukses': 'fas fa-check-circle',
    'error': 'fas fa-times-circle',
    'peringatan': 'fas fa-exclamation-
circle'
  };
  const warna = {
    'sukses': '#1cc88a',
    'error': '#e74a3b',
    'peringatan': '#f6c23e'
  };
  const notifikasiHTML = `
  <div class="notifikasi-mewah">
    <div class="notifikasi-header">
      <div class="notifikasi-icon"
style="background-color: ${warna[tipe] ||
'#1cc88a'}">
        <i class="${icon[tipe] ||
'fas fa-info-circle'}"></i>
      </div>
    </div>
    <div class="notifikasi-
judul">${judul}</div>
    <div class="notifikasi-
```

Tabel 4. 6 Segmen Keranjang (Tabel Lanjutan)

```
pesan">${pesan}</div>
    </div>
</div>
    <div class="notifikasi-progress">
        <div class="notifikasi-progress-
bar" style="background-color: ${warna[tipe] ||
'#1cc88a'}"></div>
    </div>
</div>`;
$('body').append(notifikasiHTML);

// Hilangkan notifikasi setelah 2 detik
setTimeout(() => {
    $('.notifikasi-
mewah').css('animation', 'fadeOut 0.6s
forwards');
    setTimeout(() => $('.notifikasi-
mewah').remove(), 600);

    // Redirect jika ada
    <?php
if(isset($_SESSION['notifikasi']['redirect'])):
?>
        window.location.href = '<?=
$_SESSION['notifikasi']['redirect'] ?>';
    <?php endif; ?>
}, 2000);
}
// Tampilkan notifikasi jika ada
<?php if(isset($_SESSION['notifikasi'])): ?>
    $(document).ready(function() {
        tampilkanNotifikasi(
            '<?=
$_SESSION['notifikasi']['judul'] ?>',
            '<?=
$_SESSION['notifikasi']['pesan'] ?>',
            '<?=
$_SESSION['notifikasi']['tipe'] ?>'
        );
    });
<?php unset($_SESSION['notifikasi']); ?>
```

Tabel 4. 6 Segmen Keranjang (Tabel Lanjutan)

```

    <?php endif; ?>
</script>
<script type="text/javascript"
    src="https://app.sandbox.midtrans.com/snap/s
nap.js"
    data-client-key="SB-Mid-client-
rb3Ub_oxVDYUVqUM"></script>
<script>
function bayar(pesananId) {
    fetch('snap_token.php?id=' + pesananId)
        .then(res => res.text())
        .then(snapToken => {
            snap.pay(snapToken, {
                onSuccess: function(result) {
                    window.location.href =
'sukses_midtrans.php?id=' + pesananId;
                },onPending: function(result) {
                    alert("Menunggu
pembayaran...");}, onError: function(result) {
                    alert("Terjadi kesalahan
saat pembayaran!");
                }
            });
        });
}
</script>
<script>
    // Aktifkan dropdown
    const dropdownElementList =
[...].slice.call(document.querySelectorAll('.dropdo
wn-toggle'))
    const dropdownList =
dropdownElementList.map(function
(dropdownToggleEl) {
        return new
bootstrap.Dropdown(dropdownToggleEl)
    });
</script>
<?php include 'footer.php'; ?>
</body>
</html>

```

Segmen ini menangani operasi database terkait manajemen keranjang belanja melalui query SQL yang mencakup: (1) operasi DELETE untuk menghapus item spesifik berdasarkan parameter ID, (2) operasi UPDATE untuk mengubah status transaksi menjadi 'dibayar', serta (3) operasi SELECT dengan klausa JOIN yang menggabungkan tabel *tb_pesanan*, *pengunjung*, dan *tb_tiket* untuk menampilkan data terstruktur. Sistem mengimplementasikan mekanisme keamanan dasar melalui isolasi data berbasis session *user_id* dan sanitasi parameter input. Pada layer presentasi, modul ini mengintegrasikan *Midtrans Payment Gateway* melalui *Snap* API dengan implementasi tiga handler callback (*onSuccess*, *onPending*, *onError*) untuk menangani berbagai skenario transaksi. Komponen notifikasi dibangun secara dinamis melalui DOM manipulation dengan jQuery, menampilkan feedback interaktif yang secara otomatis termanage melalui *setTimeout()* function.

g. Segmen Riwayat

Tabel 4. 7 Segmen Riwayat

```
<?php
session_start();
include 'koneksi.php';
require_once 'phpqrcode/qrlib.php';
// Periksa apakah user sudah login
if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit(); }
```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```

// Ambil user_id dari session
$user_id = $_SESSION['user_id'];

// Buat folder temp jika belum ada
if (!file_exists('temp')) {
    mkdir('temp', 0755, true);
}

// Konfigurasi
$limit = 10; // Jumlah data per halaman
$page = isset($_GET['page']) ?
(int)$_GET['page'] : 1;
$offset = ($page - 1) * $limit;
$search = $_GET['search'] ?? '';
$start_date = $_GET['start_date'] ?? '';
$end_date = $_GET['end_date'] ?? '';
$order_by = $_GET['order_by'] ?? 'created_at';
$order = $_GET['order'] ?? 'DESC';
// Hapus riwayat jika ada permintaan
if (isset($_GET['hapus'])) {
    $id = $_GET['hapus'];

    // Cek apakah pesanan ini milik user yang
    sedang login
    $check_ownership = mysqli_prepare($conn,
"SELECT p.id FROM tb_pesanan p
                                                                    JOI
N pengunjung pg ON p.pengunjung_id = pg.id
                                                                    WHE
RE p.id = ? AND pg.user_id = ?");
    mysqli_stmt_bind_param($check_ownership,
"ii", $id, $user_id);
    mysqli_stmt_execute($check_ownership);
    $result_ownership =
mysqli_stmt_get_result($check_ownership);
    if (mysqli_num_rows($result_ownership) == 0)
    {
        // Pesanan bukan milik user ini
        $_SESSION['notifikasi'] = [

```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```

        'judul' => 'Akses Ditolak!',
        'pesan' => 'Anda tidak memiliki
akses untuk menghapus riwayat ini',
        'tipe' => 'error'
    ];
    header("Location: riwayat.php");
    exit();}

    // Jika pesanan milik user ini, lanjutkan
    penghapusan
    $pesanan =
    mysqli_fetch_assoc(mysqli_query($conn, "SELECT *
FROM tb_pesanan WHERE id=$id"));
    $pengunjung_id = $pesanan['pengunjung_id'];
    // Mulai transaksi
    mysqli_begin_transaction($conn);

    try {
        // Hapus pesanan
        mysqli_query($conn, "DELETE FROM
tb_pesanan WHERE id=$id");

        // Hapus pengunjung
        mysqli_query($conn, "DELETE FROM
pengunjung WHERE id=$pengunjung_id");

        // Commit transaksi jika berhasil
        mysqli_commit($conn);

        $_SESSION['notifikasi'] = [
            'judul' => 'Berhasil Dihapus!',
            'pesan' => 'Riwayat transaksi telah
berhasil dihapus',
            'tipe' => 'sukses'
        ];
    } catch (Exception $e) {
        // Rollback jika terjadi error
        mysqli_rollback($conn);

        $_SESSION['notifikasi'] = [
            'judul' => 'Gagal Menghapus!',

```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```
        'pesan' => 'Terjadi kesalahan saat
menghapus riwayat: ' . $e->getMessage(),
        'tipe' => 'error'
    ];
}
header("Location: riwayat.php");
exit();
}
// Query dasar - Filter berdasarkan user_id yang
sedang login
$query = "SELECT tb_pesanan.id AS pesanan_id,
tb_pesanan.created_at,
    pengunjung.nama, pengunjung.no_telpon,
pengunjung.email,
    pengunjung.negara,
pengunjung.provinsi,
    tb_tiket.nama_tiket,
tb_tiket.kategori, tb_tiket.harga,
    tb_pesanan.jumlah,
tb_pesanan.total_harga,
    tb_pesanan.status as status_pesanan
FROM tb_pesanan
JOIN pengunjung ON
tb_pesanan.pengunjung_id = pengunjung.id
JOIN tb_tiket ON tb_pesanan.tiket_id =
tb_tiket.id
WHERE tb_pesanan.status IN ('dibayar',
'digunakan') AND pengunjung.user_id = $user_id";

// Tambahkan pencarian
if (!empty($search)) {
    $search = mysqli_real_escape_string($conn,
$search);
    $query .= " AND (pengunjung.nama LIKE
'%"$search%" OR tb_pesanan.id LIKE '%"$search%"");
}
// Tambahkan filter tanggal
if (!empty($start_date)) {
    $query .= " AND tb_pesanan.created_at >=
'$start_date 00:00:00'";
}
```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```
}
if (!empty($end_date)) {
    $query .= " AND tb_pesanan.created_at <=
'$end_date 23:59:59'";
}
// Hitung total data untuk paginasi
$total_query = "SELECT COUNT(*) as total FROM
($query) as total_query";
$total_result = mysqli_query($conn,
$total_query);
$total_data =
mysqli_fetch_assoc($total_result)['total'];
$total_pages = ceil($total_data / $limit);

// Tambahkan sorting dan limit
$query .= " ORDER BY $order_by $order LIMIT
$limit OFFSET $offset";
$data = mysqli_query($conn, $query);
?>
<!DOCTYPE html>
<html lang="id">
...
function cetakRiwayat(pesananId) {
    const printWindow = window.open('', '',
'width=800,height=600');
    printWindow.document.write('<html><head>
<title>Cetak Tiket #' + pesananId + '</title>');
    printWindow.document.write('<link
rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstr
ap/4.5.2/css/bootstrap.min.css">');
    printWindow.document.write('<style>');
    printWindow.document.write('body {
padding: 20px; }');
    printWindow.document.write('.kartu-cetak
{ border: 1px solid #ddd; border-radius: 10px;
padding: 20px; }');
    printWindow.document.write('.header-
cetak { text-align: center; margin-bottom: 20px;
}');
    printWindow.document.write('.qr-code {
```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```
width: 150px; height: 150px; margin: 0 auto;
}');

    printWindow.document.write('</style>');
    printWindow.document.write('</head><body
>');

    // Ambil konten kartu yang akan dicetak
    const kartu =
document.getElementById('kartu-' +
pesananId).innerHTML;
    printWindow.document.write('<div
class="header-cetak">');
    printWindow.document.write('<h3>Tiket
Wisata</h3>');
    printWindow.document.write('<p>ID
Pesanan: #' + pesananId + '</p>');
    printWindow.document.write('</div>');
    printWindow.document.write(kartu);
    printWindow.document.write('</body></htm
l>');
    printWindow.document.close();

    setTimeout(() => {
        printWindow.print();
        printWindow.close();
    }, 500);
}

// Fungsi tampilkan notifikasi
function tampilkanNotifikasi(judul, pesan,
tipe) {
    const icon = {
        'sukses': 'fas fa-check-circle',
        'error': 'fas fa-times-circle',
        'peringatan': 'fas fa-exclamation-
circle'
    };
    const warna = {
        'sukses': '#1cc88a',
        'error': '#e74a3b',
        'peringatan': '#f6c23e'
    };
};
```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```
const notifikasiHTML = `
  <div class="notifikasi-mewah">
    <div class="notifikasi-header">
      <div class="notifikasi-icon"
style="background-color: ${warna[tipe] ||
'#1cc88a'}">
        <i class="${icon[tipe] ||
'fas fa-info-circle'}"></i>
      </div>
      <div>
        <div class="notifikasi-
judul">${judul}</div>
        <div class="notifikasi-
pesan">${pesan}</div>
      </div>
      <div class="notifikasi-progress">
        <div class="notifikasi-progress-
bar" style="background-color: ${warna[tipe] ||
'#1cc88a'}"></div>
      </div>
    </div>`;

$('body').append(notifikasiHTML);

// Hilangkan notifikasi setelah 2 detik
setTimeout(() => {
  $('.notifikasi-
mewah').css('animation', 'fadeOut 0.6s
forwards');
  setTimeout(() => $('.notifikasi-
mewah').remove(), 600);
}, 2000);
}

// Tampilkan notifikasi jika ada
<?php if(isset($_SESSION['notifikasi'])): ?>
  $(document).ready(function() {
    tampilkanNotifikasi(
      '<?=
$_SESSION['notifikasi']['judul'] ?>',
      '<?='
```

Tabel 4. 7 Segmen Riwayat (Tabel Lanjutan)

```

$_SESSION['notifikasi']['pesan'] ?>',
        '<?='
$_SESSION['notifikasi']['tipe'] ?>'
        );
    });
    <?php unset($_SESSION['notifikasi']); ?>
    <?php endif; ?>
</script>
<script>
    // Aktifkan dropdown
    const dropdownElementList =
    [].slice.call(document.querySelectorAll('.dropdown-
toggle'))
    const dropdownList =
dropdownElementList.map(function
(dropdownToggleEl) {
    return new
bootstrap.Dropdown(dropdownToggleEl)
    });
</script>
<?php include 'footer.php'; ?>
</body>
</html>

```

Segmen riwayat.php merupakan modul sistem e-tiket yang menangani manajemen riwayat transaksi pengguna secara komprehensif. Modul ini memulai dengan verifikasi sesi pengguna untuk memastikan keamanan akses data. Fungsi utamanya mencakup: (1) menampilkan riwayat pembelian tiket dengan kemampuan filter berdasarkan tanggal dan pencarian kata kunci, (2) sistem paginasi untuk mengelola data dalam jumlah besar, serta (3) fitur penghapusan riwayat yang dilengkapi mekanisme validasi kepemilikan dan transaksi database yang aman. Modul ini juga menyediakan fitur pencetakan tiket dalam

format printer-friendly dengan embed QR Code untuk keperluan verifikasi.

h. Segmen *Dashboard Admin*

Tabel 4. 8 Segmen *Dashboard Admin*

```
<?php
session_start();

// Cek apakah user sudah login dan sebagai
admin
if (!isset($_SESSION['loggedin'])) {
    header("Location: ../login.php");
    exit();
}
if (!isset($_SESSION['user_type']) ||
$_SESSION['user_type'] != 'admin') {
    header("Location: ../index.php");
    exit();
}
// Koneksi database
include '../koneksi.php';

// Include library untuk export
require '../vendor/autoload.php';
use PhpOffice\PhpSpreadsheet\Spreadsheet;
use PhpOffice\PhpSpreadsheet\Writer\Xlsx;
use Dompdf\Dompdf;

// Filter pencarian
$search = isset($_GET['search']) ? $conn-
>real_escape_string($_GET['search']) : '';
$filter_negara = isset($_GET['negara']) ? $conn-
>real_escape_string($_GET['negara']) : '';
$filter_provinsi = isset($_GET['provinsi']) ?
$conn->real_escape_string($_GET['provinsi']) :
'';
$filter_tanggal = isset($_GET['tanggal']) ?
$conn->real_escape_string($_GET['tanggal']) :
'';
$filter_status = isset($_GET['status']) ? $conn-
```

Tabel 4. 8 Segmen *Dashboard* Admin (Tabel Lanjutan)

```

>real_escape_string($_GET['status']) : '';

// Proses export data
if (isset($_GET['export'])) {
    $export_type = $_GET['export'];

    // Query data dengan filter
    $query = "SELECT p.nama, p.negara,
p.provinsi,
                tk.nama_tiket, tk.kategori,
                tk.harga,
                ps.jumlah, ps.total_harga,
ps.created_at as tanggal_pembelian,
                ps.status as status_pesanan
FROM tb_pesanan ps
JOIN pengunjung p ON
ps.pengunjung_id = p.id
JOIN tb_tiket tk ON ps.tiket_id =
tk.id
WHERE ps.status IN ('dibayar',
'digunakan')";

    // Tambahkan filter pencarian jika ada
    if (!empty($search)) {
        $query .= " AND (p.nama LIKE '%$search%'
OR p.negara LIKE '%$search%' OR p.provinsi LIKE
'%$search%')";
    }
    // Filter negara
    if (!empty($filter_negara)) {
        $query .= " AND p.negara =
'$filter_negara'";
    }
    // Filter provinsi
    if (!empty($filter_provinsi)) {
        $query .= " AND p.provinsi =
'$filter_provinsi'";
    }
    // Filter tanggal
    if (!empty($filter_tanggal)) {

```

Tabel 4. 8 Segmen *Dashboard Admin* (Tabel Lanjutan)

```

        $query .= " AND DATE(ps.created_at) =
'$filter_tanggal'";
    }

    // Filter status
    if (!empty($filter_status)) {
        $query .= " AND ps.status =
'$filter_status'";
    }

    $query .= " ORDER BY ps.created_at DESC";
    $result = $conn->query($query);

    if ($export_type == 'excel') {
        // Export ke Excel
        $spreadsheet = new Spreadsheet();
        $sheet = $spreadsheet->getActiveSheet();

        // Header
        $sheet->setCellValue('A1', 'Nama
Pengunjung');
        $sheet->setCellValue('B1', 'Negara');
        $sheet->setCellValue('C1', 'Provinsi');
        $sheet->setCellValue('D1', 'Jenis
Tiket');
        $sheet->setCellValue('E1', 'Kategori');
        $sheet->setCellValue('F1', 'Harga
Satuan');
        $sheet->setCellValue('G1', 'Jumlah
Tiket');
        $sheet->setCellValue('H1', 'Total
Harga');
        $sheet->setCellValue('I1', 'Status');
        $sheet->setCellValue('J1', 'Tanggal
Pembelian');
        // Data
        $row = 2;
        while ($data = $result->fetch_assoc()) {
            $sheet->setCellValue('A' . $row,
$data['nama']);
            $sheet->setCellValue('B' . $row,
$data['negara']);

```

Tabel 4. 8 Segmen *Dashboard Admin* (Tabel Lanjutan)

```
        $sheet->setCellValue('C'.$row,  
$data['provinsi']);  
        $sheet->setCellValue('D'.$row,  
$data['nama_tiket']);  
        $sheet->setCellValue('E'.$row,  
$data['kategori']);  
        $sheet->setCellValue('F'.$row,  
$data['harga']);  
        $sheet->setCellValue('G'.$row,  
$data['jumlah']);  
        $sheet->setCellValue('H'.$row,  
$data['total_harga']);  
        $sheet->setCellValue('I'.$row,  
$data['status_pesanan']);  
        $sheet->setCellValue('J'.$row,  
$data['tanggal_pembelian']);  
        $row++;  
    }  
  
    // Format header  
    foreach (range('A','J') as $col) {  
        $sheet->getColumnDimension($col)-  
>setAutoSize(true);  
        $sheet->getStyle($col.'1')-  
>getFont()->setBold(true);  
    }  
  
    // Download file  
    header('Content-Type:  
application/vnd.openxmlformats-  
officedocument.spreadsheetml.sheet');  
    header('Content-Disposition:  
attachment;filename="data_pesanan.xlsx");  
    header('Cache-Control: max-age=0');  
  
    $writer = new Xlsx($spreadsheet);  
    $writer->save('php://output');  
    exit;  
}  
}  
$negara_query = "SELECT DISTINCT negara FROM  
pengunjung ORDER BY negara";
```

Tabel 4. 8 Segmen *Dashboard Admin* (Tabel Lanjutan)

```

$provinsi_query = "SELECT DISTINCT provinsi FROM
pengunjung WHERE provinsi != '' ORDER BY
provinsi";
$negara_result = $conn->query($negara_query);
$provinsi_result = $conn-
>query($provinsi_query);

// Query data untuk ditampilkan di dashboard
dengan filter
$query = "SELECT p.nama, p.negara, p.provinsi,
            tk.nama_tiket, tk.kategori,
            tk.harga,
            ps.jumlah, ps.total_harga,
ps.created_at as tanggal_pembelian,
            ps.status as status_pesanan
FROM tb_pesanan ps
JOIN pengunjung p ON ps.pengunjung_id
= p.id
JOIN tb_tiket tk ON ps.tiket_id =
tk.id
WHERE ps.status IN ('dibayar',
'digunakan')";

// Tambahkan filter pencarian jika ada
if (!empty($search)) {
    $query .= " AND (p.nama LIKE '%$search%' OR
p.negara LIKE '%$search%' OR p.provinsi LIKE
'%$search%')";
}

// Filter negara
if (!empty($filter_negara)) {
    $query .= " AND p.negara =
'$filter_negara'";
}

// Filter provinsi
if (!empty($filter_provinsi)) {
    $query .= " AND p.provinsi =
'$filter_provinsi'";
}

```

Tabel 4. 8 Segmen *Dashboard Admin* (Tabel Lanjutan)

```

// Filter tanggal
if (!empty($filter_tanggal)) {
    $query .= " AND DATE(ps.created_at) =
'$filter_tanggal'";
}

// Filter status
if (!empty($filter_status)) {
    $query .= " AND ps.status =
'$filter_status'";
}

$query .= " ORDER BY ps.created_at DESC";
$result = $conn->query($query);

// Hitung total pendapatan dengan filter yang
sama
$total_query = "SELECT SUM(ps.total_harga) as
total
                FROM tb_pesanan ps
                JOIN pengunjung p ON
ps.pengunjung_id = p.id
                WHERE ps.status IN ('dibayar',
'digunakan')";

if (!empty($search)) {
    $total_query .= " AND (p.nama LIKE
'%$search%' OR p.negara LIKE '%$search%' OR
p.provinsi LIKE '%$search%')";
}
if (!empty($filter_negara)) {
    $total_query .= " AND p.negara =
'$filter_negara'";
}
if (!empty($filter_provinsi)) {
    $total_query .= " AND p.provinsi =
'$filter_provinsi'";
}
if (!empty($filter_tanggal)) {
    $total_query .= " AND DATE(ps.created_at) =
'$filter_tanggal'";
}

```

Tabel 4. 8 Segmen *Dashboard Admin* (Tabel Lanjutan)

```

}

if (!empty($filter_status)) {
    $total_query .= " AND ps.status =
'$filter_status'";
}

$total_result = $conn->query($total_query);
$total_pendapatan = $total_result-
>fetch_assoc()['total'];
?>
<!DOCTYPE html>
<html lang="id">
...
    <!-- Toggle Button (Mobile) -->
    <button type="button" id="sidebarCollapse"
class="btn btn-primary me-3 d-lg-none">
        <i id="sidebarToggle" class="fas fa-
bars"></i>
    </button>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.
3/dist/js/bootstrap.bundle.min.js"></script>
    <script>
        // Sidebar Toggle
        const sidebar =
document.getElementById('sidebar');
        const sidebarOverlay =
document.querySelector('.sidebar-overlay');
        const sidebarToggle =
document.getElementById('sidebarToggle');
        const sidebarCollapse =
document.getElementById('sidebarCollapse');
        // Toggle for mobile
        function toggleSidebar() {
            sidebar.classList.toggle('active');
            sidebarOverlay.classList.toggle('act
ive');
            document.body.classList.toggle('side
bar-open');
        }

```

Tabel 4. 8 Segmen *Dashboard Admin* (Tabel Lanjutan)

```

// Toggle for desktop
function toggleCollapse() {
    document.body.classList.toggle('side
bar-collapsed');
}

// Event listeners
sidebarToggle.addEventListener('click',
toggleSidebar);
sidebarCollapse.addEventListener('click'
, toggleSidebar);
sidebarOverlay.addEventListener('click',
toggleSidebar);

// Auto close sidebar when clicking nav
link on mobile
document.querySelectorAll('#sidebar
.nav-link').forEach(link => {
    link.addEventListener('click',
function() {
        if (window.innerWidth < 992) {
            toggleSidebar();
        }
    });
});

// Auto close sidebar when resizing to
desktop
window.addEventListener('resize',
function() {
    if (window.innerWidth >= 992) {
        sidebar.classList.remove('active
');
        sidebarOverlay.classList.remove(
'active');
        document.body.classList.remove('
sidebar-open');
    }
});
</script>
</body>
</html>

```

Dashboard admin ini dirancang khusus untuk administrator dengan menerapkan mekanisme autentikasi ketat yang memastikan hanya pengguna dengan tipe 'admin' yang dapat mengaksesnya, sementara pengguna tidak sah akan secara otomatis dialihkan ke halaman *login* atau beranda.

Fitur utama *dashboard* ini mencakup kemampuan filtering data yang canggih, memungkinkan administrator untuk menyaring transaksi berdasarkan berbagai kriteria seperti nama pengunjung, asal negara, provinsi, status pembayaran (dibayar atau digunakan), serta rentang tanggal tertentu. Untuk kebutuhan pelaporan, sistem dilengkapi dengan fungsi export data ke *format* Excel menggunakan library PhpSpreadsheet yang secara otomatis menerapkan filter aktif pada data yang diekspor.

4.2 Skenario Pengujian

Pengujian *Self Testing* atau *Developer Testing* adalah pengujian perangkat lunak yang berkembang pada era 1960-an salah satu tokoh yang berperan penting dalam metode pengujian perangkat lunak ini yaitu Glenford J. Myers dalam karyanya yaitu *The Art of Software Testing*" (1979).

Tabel 4. 9 Developer Testing oleh Developer

No	Modul/Fitur	Skenario Pengujian	Input/Test Action	Hasil Diharapkan	Status
1	<i>Login</i>	<i>Login</i> dengan data valid	<i>Username & Password</i> benar	Berhasil masuk ke <i>dashboard user</i>	Pass
2	<i>Login</i>	<i>Login</i> dengan password salah	<i>Username</i> benar, password salah	Pesan error muncul	Pass
3	Register	Register dengan semua field valid	<i>Username,</i> email, password, konfirmasi	Akun berhasil dibuat	Pass
4	Register	Register dengan email yang sudah digunakan	Email yang sama	Muncul pesan "email sudah terdaftar"	Pass

Tabel 4. 9 *Developer Testing oleh Developer (Tabel Lanjutan)*

No	Modul/Fitur	Skenario Pengujian	Input/Test Action	Hasil Diharapkan	Status
5	Pemesanan Tiket	Pilih tiket dan klik tombol beli	1 tiket dipilih	Lanjut ke checkout	Pass
6	Pemesanan Tiket	Coba beli tanpa memilih tiket	Tidak ada tiket dipilih	Muncul pesan error	Pass
7	Checkout	Proses checkout berhasil dengan metode pembayaran valid	Pilih metode <i>Midtrans</i> , klik Bayar	Dialihkan ke <i>payment gateway</i>	Pass
8	<i>Payment Gateway</i>	Transaksi sukses (simulasi/callb ack)	Kode transaksi benar	Status pembayaran berubah ke "paid"	Pass

Tabel 4. 9 *Developer Testing oleh Developer (Tabel Lanjutan)*

No	Modul/Fitur	Skenario Pengujian	Input/Test Action	Hasil Diharapkan	Status
9	<i>Payment Gateway</i>	Transaksi gagal	Kode salah / saldo tidak cukup	Status berubah ke "failed"	Pass
10	Keranjang	Tambah beberapa tiket ke keranjang	Pilih lebih dari 1 tiket	Total harga sesuai jumlah	Pass
11	Keranjang	Hapus item dari keranjang	Klik hapus	Item terhapus	Pass
12	Validasi Input	Input <i>form</i> kosong pada pemesanan	Kosong semua input	Tidak bisa lanjut checkout	Pass
13	Role Access	Admin <i>login</i>	<i>Login</i> sebagai admin	Redirect ke <i>dashboard</i> admin	Pass
14	Role Access	<i>User</i> coba akses halaman admin	Akses URL admin manual	Muncul pesan <i>Username</i> password salah	Pass

Tabel 4. 9 *Developer Testing oleh Developer (Tabel Lanjutan)*

No	Modul/Fitur	Skenario Pengujian	Input/Test Action	Hasil Diharapkan	Status
15	Riwayat Pembelian	Tampilkan Riwayat	Login ke Riwayat	Muncul Riwayat	Pass
16	<i>Invoice/QR</i>	transaksi sukses, QR code muncul di <i>invoice</i>	Pembayaran selesai	QR tampil di <i>invoice</i>	Pass
17	Laporan Otomatis	Admin lihat laporan pembelian	<i>Login</i> admin → buka laporan	Data transaksi terbaru muncul	Pass
18	Laporan Otomatis	Download laporan sebagai PDF/Excel	Klik tombol unduh laporan	<i>File</i> terunduh sesuai data	Pass
19	Laporan Otomatis	Filter dengan tanggal	Pilih tanggal awal dan akhir	Data laporan sesuai filter	Pass
20	Forgot Password	Uji fitur lupa password	Klik "Lupa Password",	Link dikirim ke alamat email	Fail

4.3 Pengujian

4.3.1 Cara Kerja Sistem

a. *Landing Page*

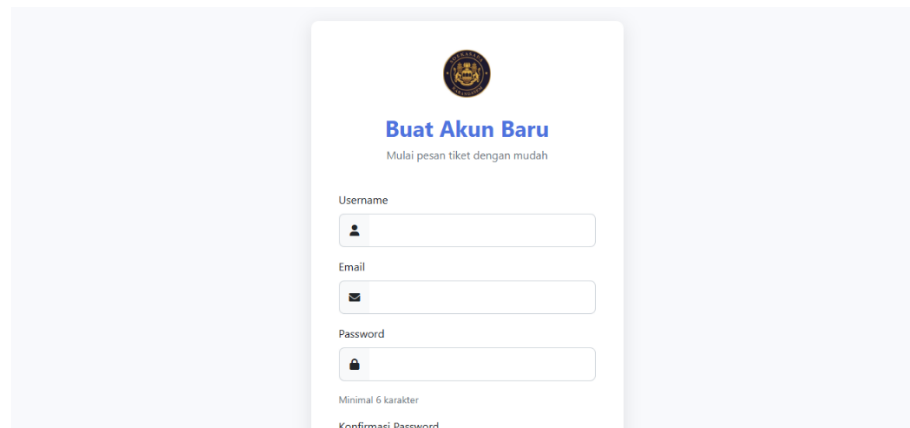
Landing page adalah halaman yang pertama kali yang akan tampil ketika akan melakukan pembelian tiket. Di halaman ini ada menu *login* dan *daftar* yang berfungsi untuk mengelola pendaftaran akun pengunjung.



Gambar 4.17 *Landing Page*

b. *Registerasi User*

Sebelum melakukan *login* semua pengunjung wajib membuat akun agar nantinya bisa melakukan *login* ke website pembelian tiket taman.



Buat Akun Baru
Mulai pesan tiket dengan mudah

Username

Email

Password

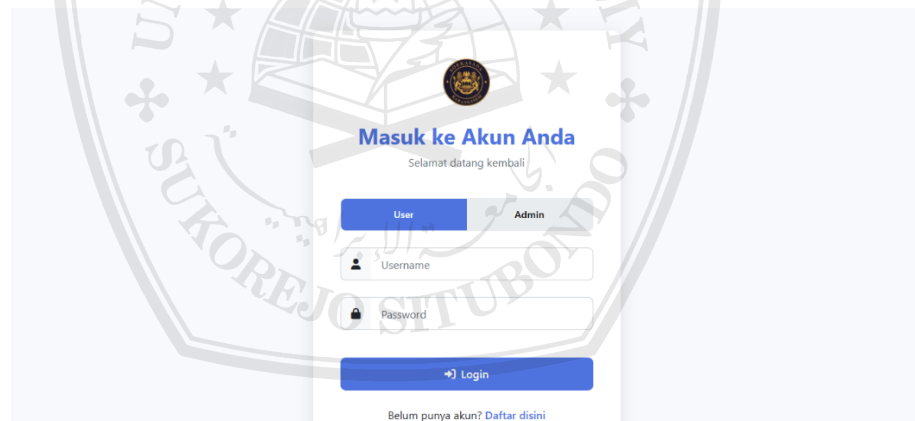
Minimal 6 karakter

Konfirmasi Password

Gambar 4. 18 Registrasi

c. *Login*

Setelah melakukan registrasi pengunjung harus *login* sebelum bisa mengakses menu pembelian tiket.



Masuk ke Akun Anda
Selamat datang kembali

User Admin

Username

Password

→ Login

Belum punya akun? [Daftar disini](#)

Gambar 4. 19 Login

d. *Dashboard User*

Dashboard user akan muncul ketika *user* sudah melakukan *login*. Pada Halaman ini disediakan menu *form* pembelian tiket yang berfungsi untuk *user* yang akan melakukan pembelian tiket



Gambar 4. 20 Dashboard User

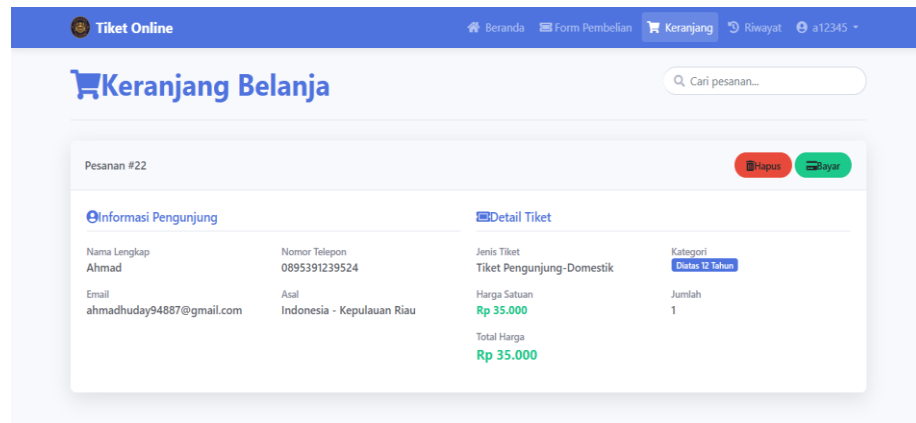
e. Menu *Form Pembelian Tiket*

Di dalam menu *form* pembelian tiket disediakan *form* yang wajib di isi oleh *user* ketika akan melakukan pembelian tiket

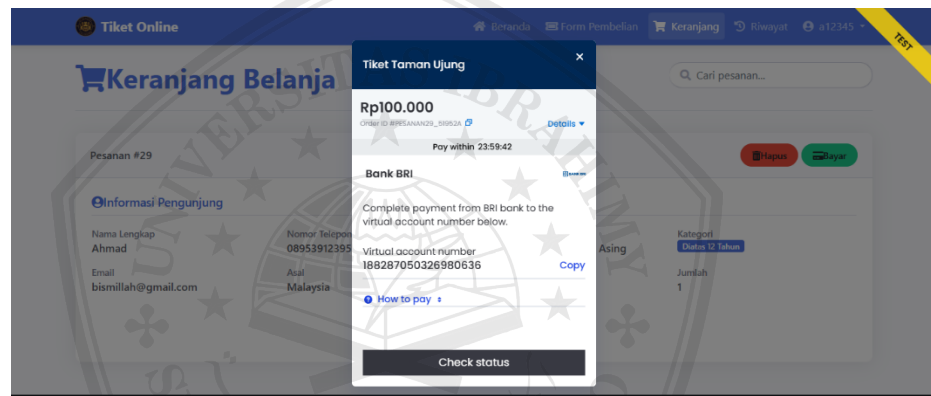
Gambar 4. 21 Form Pembelian

f. Menu Keranjang

Pada menu keranjang ditampilkan data yang sudah di isi tadi pada *form* pembelian dan juga disediakan tombol hapus untuk menghapus pembelian tiket. Dan juga ada tombol bayar yang berfungsi untuk melakukan pembayaran tiket.

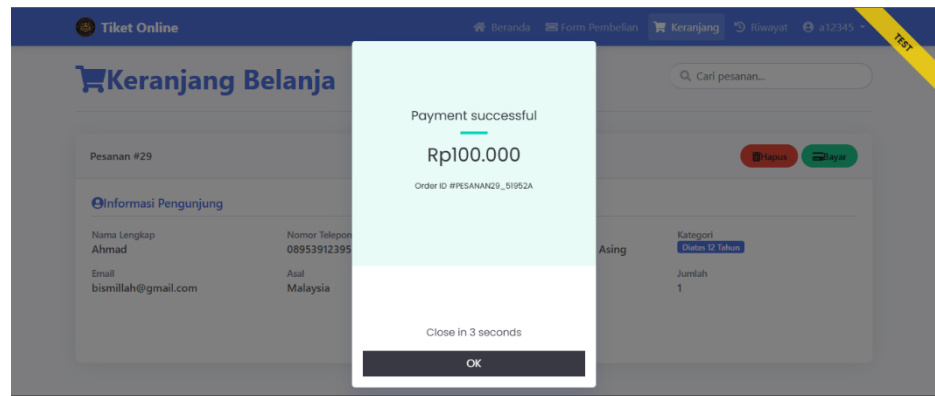


Gambar 4. 22 Menu Keranjang



Gambar 4. 23 Snap Token Pembayaran

Gambar diatas merupakan contoh dari snap token yang akan muncul ketika menekan tombol bayar. *User* bebas memilih metode pembayaran yang akan digunakan, baik berupa qris, virtual account dan lainnya.

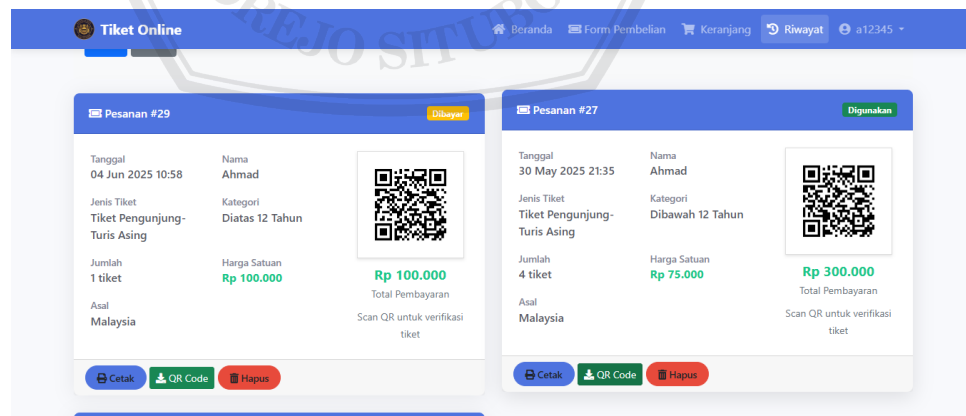


Gambar 4. 24 Pembayaran Sukses

Setelah melakukan pembayaran melalui virtual account maka akan muncul notif bahwa pembayaran sukses yang akan otomatis menampilkan menu keranjang.

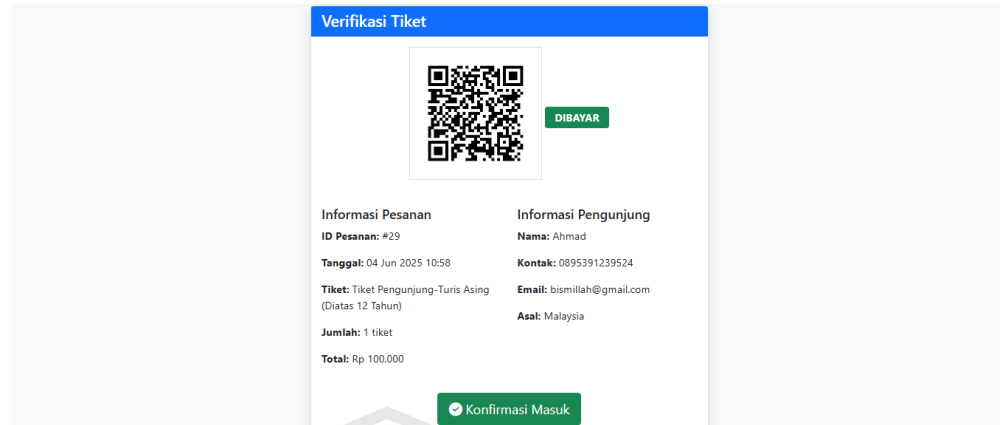
g. Menu Riwayat

Menu riwayat akan menampilkan pembayaran tiket yang sudah lunas dibayar, dan juga ada *invoice* pembayaran serta kode QR yang nantinya akan di *scan* ketika akan masuk ke taman.



Gambar 4. 25 Menu Riwayat

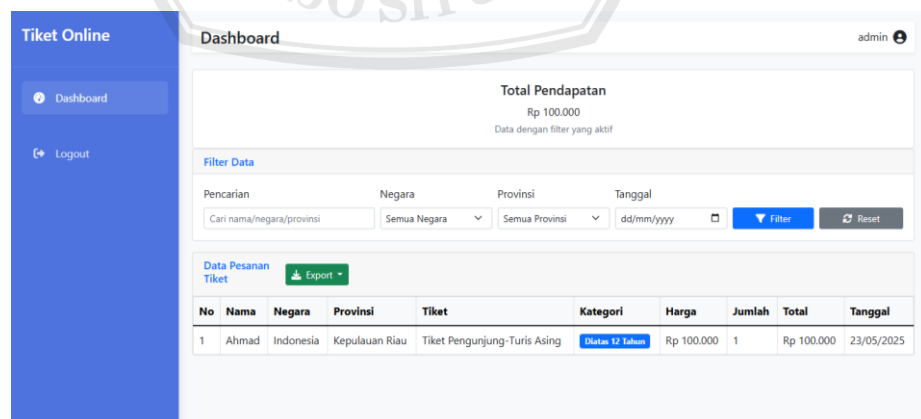
h. *Scan* Kode QR



Gambar 4. 26 Halaman Detail Pesanan

Ketika melakukan *scan* kode qr akan memunculkan detail pesanan seperti diatas dan admin akan melakukan konfirmasi bahwa tiket itu telah digunakan dengan menekan tombol konfirmasi masuk, dan akan merubah status tiket di keranjang yang sebelumnya dibayar menjadi digunakan.

i. *Dashboard* Admin



Gambar 4. 27 Halaman *Dashboard* Admin

Halaman *dashboard* admin ada dua menu yaitu menu *dashboard* yang menampilkan total pendapatan, data pesanan tiket dan juga bisa melakukan export data pemesanan tiket yang nantinya akan berbentuk excel.

4.3.2 Hasil Pengujian

Adapun pengujian sistem ini sudah dilakukan sebanyak sepuluh kali percobaan dengan semua fitur yang ada namun masih ada beberapa fitur yang belum bisa berjalan sesuai dengan yang diharapkan. Dan tabel berikut ini akan kami tampilkan hasil dari perhitungan *developer testing* yang telah dilakukan.

Tabel 4. 10 Perhitungan Developer Testing oleh Developer

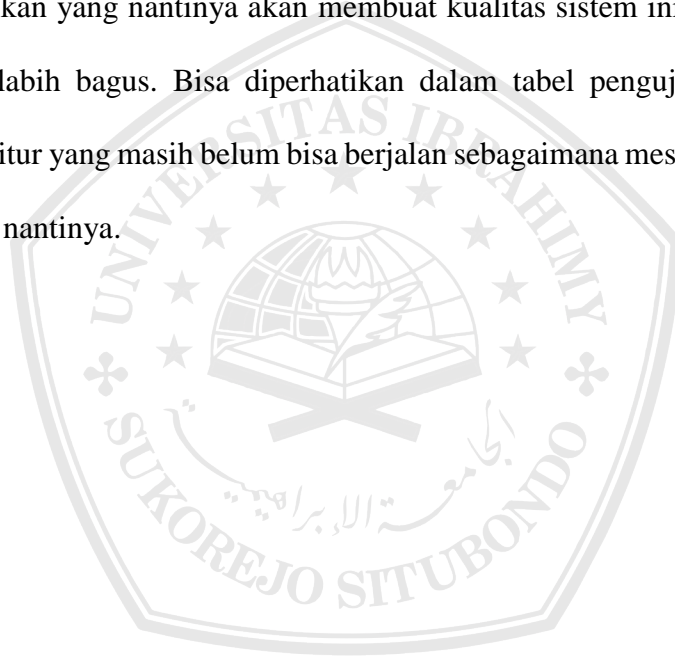
Keterangan	Nilai
Total Test Case	20
Test Case Pass	19
Test Case Fail	1
Tingkat Keberhasilan (%)	$(19 \div 20) \times 100 = 95\%$

19 dari 20 *test case* berhasil (Pass) dengan 1 *test case* gagal pada Fitur Forgot Password dan Tingkat keberhasilan pengujian: 95.65%. Jadi Aplikasi secara fungsional sudah layak, tetapi fitur “forgot password”

perlu diperbaiki Dengan adanya pembelian *online* akan terjadi percepatan dalam hal pembelian yang dulunya harus membeli dan mengecek tiket di loket, sekarang cukup hanya mengecek kode QR. Jadi terjadi peningkatan sebesar 50% .

4.4 Maintenance

Sistem informasi pembelian tiket ini butuh perawatan, perkembangan dan perbaikan yang nantinya akan membuat kualitas sistem ini menjadi lebih baik dan lebih bagus. Bisa diperhatikan dalam tabel pengujian masih ada beberapa fitur yang masih belum bisa berjalan sebagaimana mestinya yang bisa diperbaiki nantinya.



BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian dan pengujian yang telah dilakukan, dapat disimpulkan hal-hal sebagai berikut:

1. Pembelian tiket kini dapat dilakukan secara *online* melalui website yang telah dirancang. Setiap transaksi yang dilakukan oleh pengguna akan langsung tercatat secara otomatis ke dalam sistem admin dan tersusun dalam bentuk laporan yang dapat diunduh dalam *format* Excel.
2. Dari segi metode pembayaran, sistem telah terintegrasi dengan layanan *payment gateway*, sehingga pengguna dapat melakukan pembayaran secara non-tunai dengan aman dan mudah.
3. Sistem ini mempermudah dan mempercepat proses pembelian tiket, dengan kuantifikasi peningkatan efisiensi hingga 50 persen. Jika sebelumnya pembeli harus melalui dua tahap, yaitu antre di loket pembelian dan kemudian antre lagi di pintu pengecekan tiket, kini proses cukup dilakukan satu kali melalui pemindaian *barcode* tiket.
4. Antrean panjang saat pembelian tiket dapat diatasi dengan adanya sistem ini, karena pengguna tidak lagi perlu datang langsung dan mengantre untuk mendapatkan tiket secara manual.
5. Proses pelaporan yang sebelumnya dilakukan secara manual atau tulisan kini berubah menjadi otomatis, karena sistem sudah merekap

semua data pembelian dan menyediakan fitur ekspor laporan dalam *format* Excel.

6. Berdasarkan hasil pengujian *developer* (*self-testing*) yang dilakukan terhadap 20 skenario uji, sebanyak 19 skenario berhasil dijalankan dengan baik, sedangkan 1 skenario masih mengalami kegagalan. Dengan demikian, tingkat keberhasilan sistem dalam pengujian adalah sebesar 95 persen, yang menunjukkan bahwa sistem telah berjalan dengan baik secara fungsional dan layak untuk digunakan.

5.2 Saran

Dalam sistem informasi yang sudah dirancang ini masih banyak hal yang perlu dikembangkan dari segi fitur-fitur yang masih belum terpenuhi dan dan berjalan sesuai dengan yang direncanakan . Disamping itu, sistem ini juga masih dalam tahap uji coba. Dari hasil uji coba yang sudah dilakukan ada beberapa hal yang masih perlu diperbaiki dan dikembangkan.

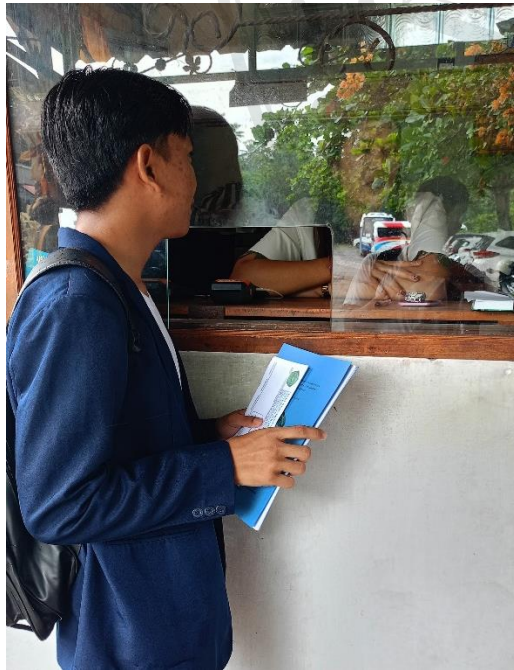
DAFTAR PUSTAKA

- [1] A. M. Darmawan, R. A. Setiawan, L. N. Fitriana, and T. Agustin, "Perancangan Sistem Informasi Pengelolaan Administrasi Berbasis Web Pada Wisata Umbul Pelem," pp. 342–347.
- [2] R. Dewi and A. Ardi, "Wisata Sejarah Kerajaan Berbasis Masyarakat Dalam Meningkatkan Perekonomian Masyarakat di Kota Karangasem Taman Soekasad Bali," vol. 8, no. 1, pp. 69–80, 2021.
- [3] T. H. E. Role, O. F. Development, O. F. Performance, M. Of, S. In, and C. Health, "(LIBRARY RESEARCH)," vol. 3, no. 2, pp. 28–42, 2021.
- [4] M. Mailasari and E. D. Sikumbang, "Sistem Informasi Perpustakaan Menggunakan Metode *Waterfall*," vol. 08, no. September, pp. 207–214, 2019.
- [5] D. S. Purnia, A. Rifai, and S. Rahmatullah, "Penerapan Metode *Waterfall* dalam Perancangan Sistem Informasi Aplikasi Bantuan Sosial Berbasis Android," pp. 1–7, 2019.
- [6] A. Nurseptaji *et al.*, "IMPLEMENTASI METODE WATERFALL PADA PERANCANGAN WATERFALL METHOD IMPLEMENTATION IN DESIGN," vol. 1, no. 2, pp. 49–57, 2021, doi: 10.24176/detika.v1i2.6101.
- [7] H. Wiyono and M. Fachrie, "Jurnal Indonesia : Manajemen Informatika dan Komunikasi Perancangan Aplikasi Pemesanan Tiket Wisata di Kecamatan Jurnal Indonesia : Manajemen Informatika dan Komunikasi," vol. 5, no. 1, pp. 574–586, 2024.
- [8] A. K. Putri, P. Khoddijah, M. Tri, I. Rahmayani, M. Dakwah, and A. Info, "Perancangan sistem informasi pembelian tiket wisata berbasis web," vol. 4, no. 2, pp. 445–455, 2023, doi: 10.46576/djtechno.
- [9] P. Sistem, I. Penjualan, and T. Kolam, "RENANG YONIF 136 BERBASIS WEBSITE," vol. 04, 2022.
- [10] N. Tamsir, "Sistem Informasi Pemesanan Tiket Berbasis Web Pada PT Caraka Travelindo Makassar," vol. X, no. 2, pp. 1–10.
- [11] I. Hadi, H. Fahmi, and M. Saleh, "Perancangan aplikasi pengelolaan tiket wisata pada desa wisata lantan berbasis web," vol. 1, no. 1, pp. 1–7, 2022.
- [12] Y. Nopianti, E. Fadilah, P. S. Informasi, P. S. Informasi, P. S. Informasi, and P. Tiket, "Sistem Informasi Pemesanan Tiket Travel Berbasis Web (Studi Kasus: Lantra Wisata Travel PO. Sejahtera)," vol. 3, pp. 143–150, 2017.
- [13] M. Christi, W. Hayuhardhika, N. Putra, and B. T. Hanggara, "Rancang

Bangun Sistem Informasi dan Pelayanan E-Ticket (Booking Online) pada Wisata Pendakian Gunung Budheg Tulungagung menggunakan Website dengan Framework Laravel,” vol. 7, no. 1, pp. 83–91, 2023.

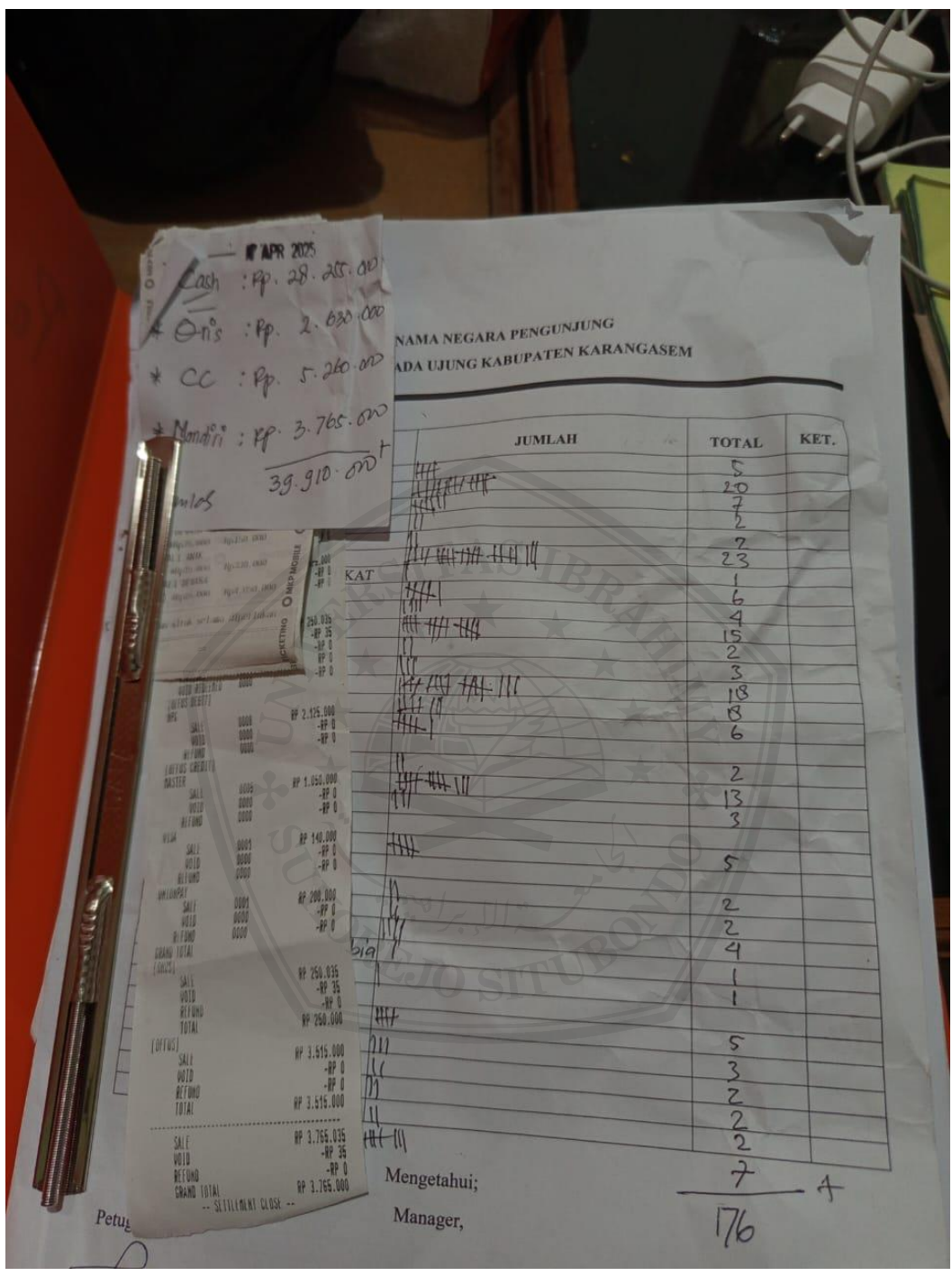
- [14] W. Jannah, I. F. Astuti, and S. Maharani, “BIMBINGAN BELAJAR BERBASIS WEB (STUDI KASUS: LEMBAGA BIMBINGAN BELAJAR TADICA),” vol. 10, no. 1, pp. 47–53, 2015.
- [15] D. Mirwansyah, K. A. Zahro, and M. Irfan, “JURNAL LOCUS : Penelitian & Pengabdian PERANCANGAN SISTEM INFORMASI MONITORING AKADEMIK DENGAN,” vol. 2, no. 12, pp. 1201–1207, 2023.
- [16] W. Jannah, I. F. Astuti, and S. Maharani, “Rancang Bangun Sistem Informasi Bimbingan Belajar Berbasis Web (Studi Kasus: Lembaga Bimbingan Belajar Tadica),” *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 10, no. 1, p. 47, 2016, doi: 10.30872/jim.v10i1.20.
- [17] F. H. Utami, “Aplikasi Pelayanan Antrian Pasien Menggunakan Metode FCFS Menggunakan *PHP* dan *MySQL*,” vol. 18, no. 1, pp. 153–160, 2022.
- [18] M. M. Fajar and U. Chotijah, “Sistem Informasi Manajemen Layanan Kearsipan (Si Malak) Berbasis Web,” *J. Inform. dan Tek. Elektro Terap.*, vol. 10, no. 3, 2022, doi: 10.23960/jitet.v10i3.2701.
- [19] J. Akuntansi, F. Ekonomi, D. Bisnis, and U. Pattimura, “Analisis Sistem Informasi Akuntansi Siklus Pendapatan Menggunakan *DFD* Dan *Flowchart* Pada Bisnis Porobico,” vol. 1, no. 2, 2023.
- [20] T. Masyarakat *et al.*, “Pelatihan Pembuatan UML (Unified Modelling Language) Menggunakan Aplikasi Draw . io Pada Prodi Sistem Informasi Universitas Muhammadiyah Jambi Training On Making UML (Unified Modeling Language) Using The Draw . io Application In The Information Systems Study Program , Muhammadiyah Jambi University,” vol. 1, no. 2, 2024.
- [21] G. K. Wardana, B. Rahayudi, W. Hayuhardika, and N. Putra, “Pengembangan E-Commerce dengan Integrasi API *Payment Gateway Midtrans*,” vol. 5, no. 11, pp. 4770–4774, 2021.
- [22] J. Sistem, I. Aplikasi, and T. Informasi, “Abstrak,” vol. 1, no. 2, pp. 209–218, 2024.
- [23] S. Audita, S. T. Siska, and A. Budiman, “Perancangan Sistem Jadwal Dan Absensi Mengajar Guru Menggunakan Visual Studio 2012 Dan *Mysql*,” vol. 02, no. 01, pp. 21–30, 2022.

LAMPIRAN



11 MAY 2025

Cash	=	30.615.000
Site	=	3.900.000
BRN	=	1.440.000
M	=	1.550.000
		<hr/>
		37.505.000



**DATA NAMA NEGARA PENGUNJUNG
TAMAN SOEKASADA UJUNG KABUPATEN KARANGASEM**

TANGGAL: 1 APR 2025

NO	WISMAN	JUMLAH	TOTAL	KET.
1	BELANDA		5	
2	PRANCIS		20	
3	JERMAN		7	
4	ITALIA		2	
5	BELGIA		7	
6	RUSSIA		23	
7	AMERIKA SERIKAT		1	
8	INGGRIS		6	
9	SPANYOL		4	
10	AUSTRALIA		15	
11	CINA		2	
12	JEPANG		3	
13	KOREA		18	
14	INDIA		8	
15	CEKO		6	
16	UKRAINA		2	
17	AUSTRIA		13	
18	POLANDIA		3	
19	FILIPINA		5	
20	SINGAPURA		2	
21	HUNGARIA		2	
22	THAILAND		4	
23	CANADA		1	
24	SWISS Lebanon		1	
25	TAIWAN		4	
26	MALAYSIA columbia		1	
27	DENMARK		1	
28	NEWZEALAND		5	
29	Turkey		3	
30	Lituania		2	
31	Latvia		2	
32	Iran		2	
33	Pakistan		2	
	Bulgaria		7	4
			<u>176</u>	

Mengetahui;
Manager,

Petugas I
Werdhi Astawa

Petugas II
Srianih

DAFTAR TRANSAKSI PRODUK

Metode	Kategori Produk	Kode Produk	Nama Produk	Harga Jual	Kuantitas	Total Jual
UJUN CC		TU-001	DOMESTIK	35000	1	35000
UJUN QRIS		TU-001	DOMESTIK	35000	42	1470000
UJUN TUNAI		TU-001	DOMESTIK	35000	168	5880000
UJUN QRIS		TU-002	DOMESTIK	25000	6	150000
UJUN TUNAI		TU-002	DOMESTIK	25000	9	225000
UJUN QRIS		TU-003	INTERNASIONAL	100000	50	5000000
UJUN TUNAI		TU-003	INTERNASIONAL	100000	4	400000
UJUN CC		TU-003	INTERNASIONAL	100000	126	12600000
UJUN QRIS		TU-003	INTERNASIONAL	75000	2	150000
UJUN TUNAI		TU-003	INTERNASIONAL	75000	2	150000
UJUN CC		TU-004	INTERNASIONAL	75000	1	75000
UJUN TUNAI		TU-004	INTERNASIONAL	75000	2	150000
UJUN QRIS		TU-007	KITAS DEWASA	75000	4	300000
UJUN TUNAI		TU-007	KITAS DEWASA	75000	4	300000
UJUN QRIS		TU-007	KITAS DEWASA	75000	4	300000
UJUN TUNAI		TU-007	KITAS DEWASA	15000	4	60000
UJUN QRIS		TU-018	KTP BALI A	15000	41	615000
UJUN TUNAI		TU-018	KTP BALI A	15000	16	400000
UJUN QRIS		TU-019	KTP BALI D	25000	16	400000
UJUN TUNAI		TU-019	KTP BALI D	25000	345	8625000

DAFTAR TRANSAKSI GROUPING BY PRODUK

Metode	Kode Produk	Nama Produk	Harga Jual	Kuantitas	Total Jual
UJUN	TU-001	DOMESTIK DEWASA	35000	211	7385000
UJUN	TU-002	DOMESTIK ANAK (6-12 tahun)	25000	15	375000
UJUN	TU-003	INTERNASIONAL DEWASA	100000	180	18000000
UJUN	TU-004	INTERNASIONAL ANAK	75000	4	300000
UJUN	TU-007	KITAS DEWASA ANAK	75000	7	525000
UJUN	TU-018	KTP BALI ANAK	15000	45	675000
UJUN	TU-019	KTP BALI DEWASA	25000	361	9025000
					36285000

DAFTAR TRANSAKSI GROUPING BY METODE PEMBAYARAN

Metode	Kategori Produk	Harga Jual	Kuantitas	Total Jual
UJUN CC		35000	1	35000
UJUN CC		75000	3	225000
UJUN CC		100000	50	5000000
UJUN QRIS		15000	4	60000
UJUN QRIS		25000	22	550000
UJUN QRIS		35000	42	1470000
UJUN QRIS		75000	2	150000
UJUN TUNAI		100000	4	400000
UJUN TUNAI		15000	41	615000
UJUN TUNAI		25000	354	8850000
UJUN TUNAI		35000	168	5880000
UJUN TUNAI		75000	6	450000
UJUN TUNAI		100000	126	12600000



**TAMAN SOEKASADA UJUNG WATER PALACE
MANAGER TAMAN SOEKASADA UJUNG**

Alamat : Desa Tumbu, Karangasem, Bali, Indonesia (80811)

Telpon : +62 811-3822-1919

Website : tamansoekasadaujung.com

SURAT KETERANGAN TELAH MELAKUKAN PENELITIAN

Nomor : 10.10/21/TSU/V/2025

Yang Bertanda Tangan dibawah ini:

Nama : Pratik Agung Made Dewandra Djelantik
Jabatan : Manager Taman Soekasada Ujung
Alamat : Desa Tumbu, Karangasem, Bali

Dengan ini menerangkan bahwa:


Nama : AHMAD HUDAY
NPM : 2021502003
Status : Mahasiswa Universitas Ibrahimy, Sukorejo, Situbondo, Jawa Timur
Alamat : Br. Dinas Kecicang Islam, Bungaya Kangin, Bebandem, Karangasem, Bali

Telah melakukan penelitian di Taman Soekasada Ujung, sebagai bahan penulisan skripsi dengan judul "SISTEM INFORMASI PEMBELIAN TIKET WISATA TAMAN UJUNG SUKASADA KARANGASEM BALI BERBASIS WEBSITE."



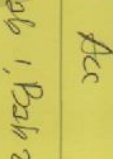

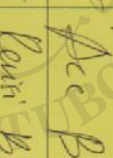
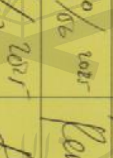
Demikian surat keterangan ini dibuat dengan sebenarnya untuk dapat digunakan sebagaimana mestinya.

Karangsem 19 Mei 2025,
Manager Taman Soekasada Ujung,



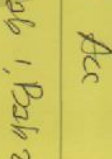

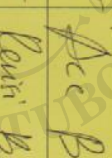
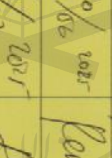



Ni Ntm Sri Dewi Hartati

Pembimbing I : Agelmaso Ramguk

TANGGAL	CATATAN	PARAF
02/01/2015	Proposal, Revisi	
17/01/2015	Acc proposal	
21/01/2015	Bab 1-2	
02/02/2015	Acc Bab 1-2, Bab 3	
20/06/2015	Bab 3-4. Program	
5/07/2015	Acc	

Pembimbing II : A. Hamdani.

NO	TANGGAL	CATATAN	PARAF
1	05/01/2015	Revisi Proposal	
2	20/01/2015	Proposal Acc	
3	21/01/2015	Revisi Bab 1, Bab 2, Bab 3	
4	05/02/2015	Acc Bab 1, Bab 2, Bab 3	
5	20/06/2015	Revisi Bab 4, Bab 5	
6	06/07/2015	Acc	

CURICULUM VITAE



AHMAD HUDAY lahir di Karangasem pada tanggal 21 Mei 2003. Saat ini merupakan mahasiswa aktif pada Program Studi Sistem Informasi. Saya merupakan putra dari pasangan Abdurahman (ayah) dan Raodah (ibu). Riwayat pendidikan saya dimulai dari Madrasah Ibtidaiyah Negeri Bungaya (2015) sebagai jenjang sekolah dasar, kemudian saya melanjutkan pendidikan ke tingkat menengah pertama di Madrasah Tsanawiyah Negeri Amlapura (2018). Setelah itu, saya menyelesaikan pendidikan menengah atas di Madrasah Aliyah Negeri Karangasem (2021). Dalam perjalanan pendidikan saya, saya juga aktif dalam berbagai kegiatan organisasi. Saya pernah menjadi bagian dari OSIS MTs Negeri Amlapura, serta melanjutkan kiprah organisasi di OSIS MA Negeri Karangasem. Di luar lingkungan sekolah, saya juga pernah aktif di organisasi IKSASS Bali, khususnya pada Seksi Sumber Daya Manusia. Saat ini, saya berdomisili di Jalan Teuku Umar, tepatnya di Dusun Kecicang Islam, Desa Bungaya Kangin, Kecamatan Bebandem, Kabupaten Karangasem, Provinsi Bali.