

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian ini merupakan penelitian yang membahas tentang rancang bangun website *e-ticketing*. Tinjauan pustaka yang masih berhubungan dengan penelitian ini antara lain sebagai berikut.

Penelitian pertama yaitu penelitian yang dilakukan oleh Yudha Herlambang dengan judul “Sistem Marketplace Event dan Pemesanan E-Ticket (Studi Kasus : Universitas Islam Indonesia)” tahun 2021. Penelitian ini memiliki tujuan menciptakan sebuah sistem sebagai wadah informasi yang mencakup semua *event* yang diselenggarakan oleh organisasi, jurusan, fakultas, dan universitas kampus UII. Tak hanya itu, sistem ini juga memudahkan mahasiswa untuk mengakses informasi tentang *event* yang diselenggarakan dalam satu platform, serta dapat dijadikan pilihan pengganti dari tiket konvensional dengan tiket elektronik (*e-ticket*) dan memudahkan mahasiswa dalam pemesanan tiket. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *waterfall* [8].

Penelitian kedua yaitu penelitian yang dilakukan oleh Rhesa Elian Nugroho dengan judul “Pembuatan Sistem Informasi ”ETICK” (*Event Registration and Ticketing*) Menggunakan Framework Laravel” tahun 2019. Penelitian ini memiliki tujuan untuk membuat sebuah sistem informasi berupa website *e-ticketing* yang digunakan untuk sebuah acara pertunjukan seni, musik, pameran, dan lainnya yang dibuat untuk kebutuhan suatu daerah sebagai alat untuk mempromosikan acara dan *platform* untuk membeli tiket acara, sehingga dapat mempermudah penyelenggara acara dan calon peserta. Pada penelitian ini, metode perancangan perangkat lunak yang digunakan dalam penelitian ini adalah *waterfall* [9].

Penelitian ketiga yaitu penelitian yang dilakukan oleh Alvian Rizki Bachtiar dan Iwan Krisnadi dengan judul “Manajemen Bisnis Event

Kampus Dengan Memanfaatkan Aplikasi Mobile Untuk Meningkatkan Pengunjung Di Era 4.0” tahun 2020. Penelitian ini memiliki tujuan membangun sebuah sistem informasi pendaftaran keanggotaan dan *event* pada Himpunan Mahasiswa dan UKM universitas berbasis android. Nantinya aplikasi tersebut dapat memudahkan mahasiswa berpartisipasi dalam setiap *event*, dan membuat proses memilih menjadi lebih efisien [10].

Penelitian keempat yaitu penelitian yang dilakukan oleh Gharin Irzan Musyaffa dengan judul “Sistem Informasi Manajemen Konveksi CV. Matahari Berbasis Website” tahun 2022. Penelitian ini memiliki tujuan menghasilkan sistem informasi manajemen konveksi yang mempermudah pihak konveksi dalam mengelola data dan membantu mempromosikan usaha mereka melalui sistem ini [11].

Penelitian kelima yaitu penelitian yang dilakukan oleh Muhammad Mahrus Zain, Eileen Noviana, dan Edi Tri Prayitno dengan judul “*Redesign User Interface and User Experience on a New Student Admission Website Using a Design Thinking Method*” tahun 2024. Penelitian ini memiliki tujuan untuk mendesain ulang antarmuka pengguna (UI) dan pengalaman pengguna (UX) pada situs penerimaan mahasiswa baru di Politeknik Caltex Riau (PCR) dengan menggunakan metode *design thinking* [12].

Tabel 2. 1 Penelitian Terkait

No.	Judul penelitian (peneliti, tahun)	Jenis Penyebaran Informasi	Metode Penelitian	Jenis Pemesanan Tiket Event	Studi Kasus
1.	Sistem Marketplace Event dan Pemesanan E-Ticket (Studi Kasus : Universitas Islam Indonesia) (Y. Herlambang, 2021)	Menggunakan website	<i>Waterfall</i>	Berupa tiket digital tanpa perlu dicetak	Seluruh <i>event</i> yang diselenggarakan oleh pihak organisasi, jurusan, fakultas, dan universitas dikampus UII
2.	Pembuatan Sistem Informasi "ETICK" (<i>Event Registration and Ticketing</i>) Menggunakan Framework Laravel (R. E. Nugroho, 2019)	Menggunakan website	<i>Waterfall</i>	Berupa tiket elektronik (digital) yang perlu di cetak terlebih dahulu	Mengenai <i>event</i> kesenian (pertunjukan seni, musik, pameran, dll untuk suatu daerah)
3.	Manajemen Bisnis Event Kampus Dengan Memanfaatkan Aplikasi Mobile Untuk Meningkatkan Pengunjung Di Era 4.0 (Bachtiar & Krisnadi, 2020)	Menggunakan aplikasi berbasis Android	<i>Waterfall</i>	Tidak menyertakan pembelian tiket <i>event</i>	Pendaftaran mahasiswa yang akan bergabung dalam keanggotaan dan <i>event</i> yang diadakan oleh himpunan mahasiswa dan UKM

No.	Judul penelitian (peneliti, tahun)	Jenis Penyebaran Informasi	Metode Penelitian	Jenis Pemesanan Tiket Event	Studi Kasus
4.	Sistem Informasi Manajemen Konveksi CV. Matahari Berbasis Website (G.I. Musyaffa, 2022)	Menggunakan website	<i>Waterfall</i>	Tidak menyertakan pembelian tiket <i>event</i>	Mengelola data manajemen konveksi serta membantu mempromosikan konveksi melalui sistem informasi manajemen konveksi.
5.	<i>Redesign User Interface and User Experience on a New Student Admission Website Using a Design Thinking Method</i> (M.Zain, E.Noviana, E.Prayitno, 2024)	Menggunakan website	<i>Design Thinking</i>	Tidak menyertakan pembelian tiket <i>event</i>	Perancangan ulang antarmuka pengguna (UI) dan pengalaman pengguna (UX) pada situs web penerimaan mahasiswa baru di Politeknik Caltex Riau (PCR)

2.2 Landasan Teori

2.2.1 *Website*

Website adalah kumpulan halaman yang menampilkan bermacam-macam informasi berupa teks, data, gambar, dan video ataupun kombinasi dari seluruh komponen tersebut. *Website* memiliki sifat yang tetap (statis) dan sifat yang berubah-ubah (dinamis). Tak hanya itu, *website* juga merupakan sistem yang memudahkan pengguna dalam mencari informasi seperti teks, gambar, video, dan sejenisnya dengan cara terhubung melalui jaringan internet [11].

2.2.2 *E-Ticket*

Electronic Ticket (e-ticket) adalah tiket elektronik yang merupakan sebuah tiket berbentuk digital yang biasanya berfungsi sebagai tiket untuk transportasi, pertunjukan, bioskop, bahkan penginapan. *E-ticket* merupakan metode untuk mencatat dan menyimpan data yang diperlukan tanpa perlu menggunakan dokumen fisik, karena semua informasi yang dibutuhkan telah disimpan dalam sistem komputer [1].

2.2.3 *Design Thinking*

Design Thinking adalah pendekatan yang berpusat pada manusia (*human centered*) untuk memecahkan masalah dan memperkenalkan inovasi baru [6]. Metode ini berpusat pada manusia yang dimana menciptakan kembali suatu permasalahan, lalu melakukan pengumpulan ide (*brainstorming*), kemudian mengambil langkah pendekatan secara praktis untuk menciptakan prototipe serta melakukan pengujian guna memecahkan permasalahan yang tidak jelas atau tidak diketahui sebelumnya [7]. Metode ini terdiri dari lima fase yaitu *empathize*, *define*, *ideate*, *prototype*, dan *testing*.



Gambar 2. 1 Tahapan *Design Thinking* [7]

Berikut penjelasan terkait 5 fase yang ada pada metode *design thinking*.

1. *Empathize*

Fase ini berfokus pada pemahaman mendalam terkait perspektif, kebutuhan, dan tantangan yang dirasakan pengguna. Langkah awal yang dilakukan adalah dengan menggali informasi tentang pengguna.

2. *Define*

Fase ini menganalisis dan menguraikan semua informasi yang telah didapatkan dari fase sebelumnya. Tujuannya untuk merumuskan permasalahan dan tantangan yang sesuai yang ingin diatasi.

3. *Ideate*

Fase ini menemukan ide solusi untuk mengatasi permasalahan yang telah ditentukan. Langkah yang dilakukan adalah dengan *brainstorming* untuk menghasilkan ide-ide kreatif dan inovatif, dan mencari solusi alternatif dalam menyelesaikan masalah yang dihadapi.

4. *Prototype*

Setelah memperoleh ide solusi, fase ini adalah membuat prototipe atau model sederhana. Tujuannya adalah memberikan gambaran berupa model fisik terkait ide solusi yang telah diperoleh sebelumnya.

5. *Testing*

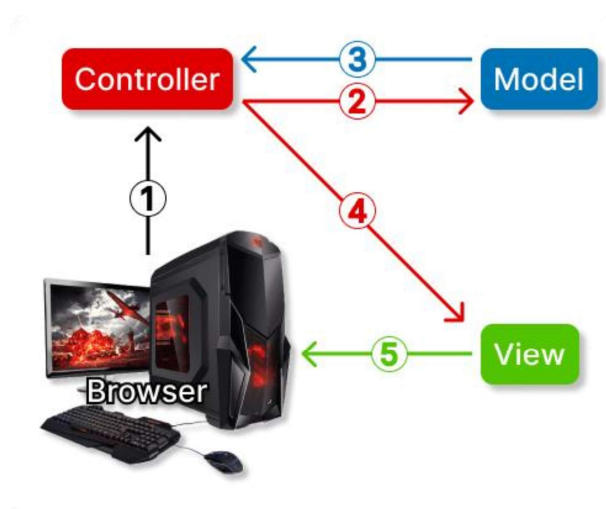
Fase terakhir ini adalah menguji prototipe yang telah dibuat kepada pengguna yang sesuai. Pengujian ini menghasilkan umpan balik yang berharga untuk membantu memahami keefektifan dan kecocokan ide solusi yang diusulkan. Dari hasil uji coba dan umpan balik, dapat melakukan iterasi, penyesuaian, atau perubahan desain untuk mencapai solusi akhir yang baik sesuai kebutuhan pengguna.

2.2.4 Rancang Bangun

Menurut Widyawati, perancangan sistem adalah cara untuk mewakili suatu sistem dengan tujuan menciptakan sistem baru atau meningkatkan sistem yang sudah ada [13]. Menurut para ahli ini, perancangan sistem merupakan langkah dalam mengembangkan dan memperbaiki sistem atau aplikasi yang ada maupun yang baru, dengan menggunakan komponen-komponen dari analisis sistem [14]. Proses perancangan sistem bertujuan untuk menghasilkan aplikasi berkualitas yang memenuhi kebutuhan yang diinginkan. Proses ini melibatkan tahapan seperti analisis kebutuhan, desain sistem, pengkodean, pengujian, dan implementasi. Perancangan sistem juga melibatkan penggunaan teknologi dan metodologi yang tepat untuk memastikan aplikasi beroperasi secara efektif dan sesuai dengan standar kualitas yang diharapkan.

2.2.5 Framework Laravel

Laravel adalah *framework* yang sangat baik dalam konteks sistem informasi yang bermanfaat untuk pengembangan proyek [15]. *Framework* ini dikenal sebagai solusi pemrograman terbaik untuk mengembangkan website yang intuitif dan interaktif berbasis PHP [16].



Gambar 2. 2 Komponen *Framework Laravel* [16]

Framework Laravel memiliki beberapa komponen, antara lain sebagai berikut.

1. *Model*

Mewakili struktur data dan berisi fungsi-fungsi yang membantu individu dalam mengelola basis data, termasuk tugas-tugas seperti memasukkan data ke dalam basis data, melakukan pembaruan data, dan berbagai fungsi lainnya.

2. *View*

Merupakan bagian yang bertanggung jawab untuk menampilkan data ke pengguna.

3. *Controller*

Merupakan kelas yang mengelola logika aplikasi dan berfungsi sebagai perantara antara *model* dan *view*.

2.2.6 PHP

PHP (Hypertext Preprocessor) adalah salah satu bahasa pemrograman yang beroperasi di dalam server web dan bertanggung jawab mengolah data di server. Ketika data dikirim oleh klien, data tersebut akan diolah dan disimpan dalam database server web, dan dapat ditampilkan kembali saat diakses.

Untuk menciptakan sebuah situs web yang dinamis dan mudah diperbarui setiap waktu melalui browser, diperlukan sebuah program yang mampu mengolah data dari komputer klien atau dari server itu sendiri agar dapat disajikan dengan nyaman di browser. PHP adalah salah satu program yang dapat dijalankan di server dan dianggap cukup handal dalam hal tersebut [17].

2.2.7 Efektivitas Penyelesaian Tugas

Efektivitas dapat diukur dengan mengukur tingkat keberhasilan penyelesaian tugas. Sebagai ukuran *usability* dasar, jika partisipan berhasil menyelesaikan tugasnya, maka tingkat keberhasilan tugas dihitung dengan memberikan nilai biner '1' dan jika partisipan tidak berhasil menyelesaikan tugas akan diberikan nilai biner '0'. Dengan demikian, efektivitas dapat dinyatakan sebagai persentase menggunakan persamaan (1) [18].

$$\bar{E} = \frac{\sum_{j=1}^R \sum_{i=1}^N n_{ij}}{RN} \times 100\%$$

Keterangan :

\bar{E} adalah tingkat penyelesaian tugas (*completion rate*)

N adalah total tugas

R adalah jumlah partisipan

n_{ij} adalah hasil dari tugas i oleh partisipan j, n_{ij} akan bernilai 1 atau 0.





2.2.8 UML (*Unified Modeling Language*)


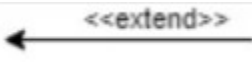
Unified Modeling Language (UML) adalah standar bahasa untuk mendokumentasikan, merancang, dan memodelkan sistem perangkat lunak. UML menggunakan notasi grafis untuk menggambarkan struktur, fungsi, dan interaksi komponen dalam sistem. Sebagai bahasa pemodelan yang efektif, UML memfasilitasi visualisasi dan dokumentasi, serta dapat menghasilkan kode pemrograman yang siap diimplementasikan. UML membantu mendeskripsikan dan merancang sistem perangkat lunak, terutama yang berorientasi objek[19].

2.2.9 *Use Case Diagram*

Use case menggambarkan cara pengguna berinteraksi dengan sistem dan menjelaskan fungsinya. Setiap skenario menunjukkan urutan kejadian antara pengguna dan sistem, dimulai oleh individu, sistem lain, perangkat lunak, atau urutan waktu [20].

Tabel 2. 2 Simbol *Use Case Diagram*






Simbol	Keterangan
	Use Case : abstraksi dan interaksi antara sistem dan aktor
	Aktor : mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case
	Generalisasi : menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Association : abstraksi penghubung antara aktor dengan use case

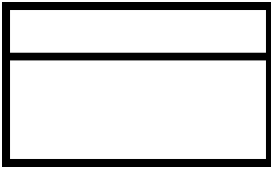
	<p>Include : menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.</p>
	<p>Extend : menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainya jika suatu kondisi terpenuhi</p>

2.2.10 Activity Diagram

Activity Diagram adalah jenis diagram alur kerja sistem yang dihasilkan dari *use case* untuk menggambarkan aliran aktivitas atau tindakan dalam sistem yang akan diimplementasikan. Diagram ini berfungsi untuk memberikan visualisasi urutan aktivitas atau proses dalam sistem, memahami alur proses secara keseluruhan, dan menggambarkan proses bisnis dalam sistem [21].

Tabel 2. 3 Simbol *Activity Diagram*

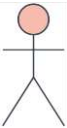


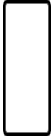
Simbol	Keterangan
	<p>Node Awal : komponen sebagai status awal atau komponen untuk memulai aktivitas</p>
	<p>Node Akhir : komponen sebagai status akhir atau komponen untuk mengakhiri sebuah aktivitas.</p>
	<p>Aktivitas : sebuah aktivitas atau tindakan yang dilakukan pada sistem</p>
	<p>Percabangan/Descision : situasi jika aktivitas lebih dari satu, sehingga terdapat kondisi untuk menentukan aktivitas yang akan dituju</p>
	<p>Penggabungan/Join : dimana lebih dari satu aktivitas lalu digabungkan jadi satu</p>






	<p>Swimlane : batasan atau boundary sistem yang memisahkan proses bisnis interaksi sistem dengan interaksi pengguna yang bertanggung jawab terhadap aktivitas.</p>
---	---

2.2.11 Sequence Diagram

Sequence diagram adalah diagram yang menunjukkan kolaborasi antara objek-objek yang berinteraksi di dalam elemen-elemen sebuah kelas. Diagram ini digunakan untuk menggambarkan interaksi antara objek-objek dalam dan sekitar suatu sistem. Diagram ini menunjukkan pesan-pesan yang dikirim antara objek-objek tersebut dari waktu ke waktu. *Sequence diagram* memiliki dimensi vertikal yang mewakili waktu dan dimensi horizontal yang mencakup objek-objek yang terlibat dalam interaksi [22].

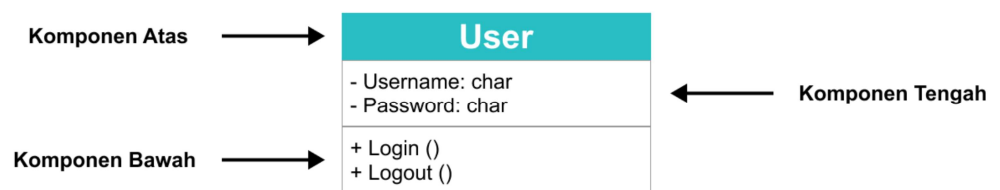
Tabel 2. 4 Simbol *Sequence Diagram*

Simbol	Keterangan
	<p>Aktor : Menunjukkan entitas yang berinteraksi dengan atau berada di luar sistem.</p>
	<p>Garis Hidup/Lifeline : Melambangkan perjalanan waktu yang meluas ke bawah.</p>
	<p>Objek : Menyatakan objek yang berinteraksi pesan</p>
	<p>Waktu Aktif : Menunjukkan objek dalam keadaan aktif dan berinteraksi, semua yang terkait dengan waktu</p>

	aktif ini merupakan tahap yang dilakukan di dalamnya.
	Pesan Create : dimana lebih dari satu aktivitas lalu digabungkan jadi satu
	Pesan Asinkron : Pesan asinkron tidak memerlukan respons sebelum pengirim dapat melanjutkan. Hanya panggilan yang perlu dicantumkan dalam diagram.
	Pesan Sinkron : Digunakan ketika pengirim harus menunggu respons terhadap pesan sebelum melanjutkan. Diagram harus menampilkan panggilan dan responsnya.
	Pesan Balasan : Diwakili oleh garis putus-putus dengan panah bergaris, pesan-pesan ini adalah balasan panggilan.
	Pesan Tipe Hapus : Diwakili oleh garis padat dengan ujung panah penuh, diakhiri dengan tanda X, pesan ini menghancurkan sebuah objek.

2.2.12 Class Diagram

Class Diagram adalah jenis diagram yang menggambarkan secara rinci struktur kelas, termasuk deskripsi, atribut, metode, dan hubungan antara objek yang bersifat statis. Diagram ini berfungsi untuk menggambarkan struktur sistem secara mendetail, menunjukkan skema atau proses sistem, serta memberikan gambaran terperinci tentang hubungan antar objek [23].



Gambar 2. 3 Komponen *Class Diagram* [23]

Class diagram terdiri dari beberapa komponen yang digunakan untuk merancang diagram, seperti yang ditunjukkan pada gambar diatas, yaitu:

1. Komponen atas berfungsi untuk memberikan nama kelas.
2. Komponen tengah berisi kumpulan atribut yang menjelaskan sebuah kelas.
3. Komponen bawah berisi metode-metode yang mendeskripsikan operasi pada kelas sebagai interaksi dengan data.