

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Review jurnal penelitian terkait yang memiliki topik atau metode serupa dapat digunakan untuk mengidentifikasi kelebihan dan kekurangan hasil penelitian yang telah diteliti penulis terkait, sehingga ditemukan kesimpulan pengembangan penelitian yang baru berdasarkan kekurangan jurnal tersebut. dari penulis telah mencari jurnal serupa dan berikut daftar jurnalnya :

Tabel 2.1 Penelitian Terdahulu

No	Nama Jurnal	Masalah	Solusi Masalah	Kelebihan	Kekurangan
1	PENGEMBANGAN APLIKASI SISTEM E-TICKETING KONSER MUSIK DENGAN SEATING NUMBER PADA WEBSITE MOTIKDONG.COM (2020)	Pendapatan utama yang diinginkan para musisi berasal dari <i>live gig</i> tempat mereka tampil, dengan sebagian besar pendapatannya berasal dari penjualan tiket. Hal ini membawa mereka keuntungan.	Diharapkan dapat dirancang aplikasi penjualan tiket konser musik dengan target <i>platform website</i> , yang memiliki fitur pemilihan kursi dengan nomor kursi, dan nantinya akan digunakan untuk melakukan penjualan tiket konser secara daring.	<ul style="list-style-type: none"> • Aplikasi mempunyai fitur seating number • Aplikasi bisa mengirim tiket ke email 	Aplikasi tidak mempunyai aktor pemegang event Aplikasi tidak mempunyai sistem autentikasi untuk pengguna
2	RANCANG BANGUN APLIKASI E-TICKET BERBASIS WEB PADA CIPUTRA WATERPARK SURABAYA (2020)	Ciputra Waterpark saat ini masih melakukan penjualan tiket secara offline, dimana calon pembeli harus muncul langsung ke lokasi dan melakukan antrian untuk membeli tiket.	Sistem penjualan tiket online merupakan resam mengabadikan ikhtiar penjualan tanpa berkeinginan evidensi fisik. Semua keterangan yang relevan secara otomatis dicatat	Aplikasi mempunyai aktor Staff Keuangan	Aplikasi hanya berlaku untuk 1 event Aplikasi tidak mempunyai aktor pemegang event Aplikasi tidak mempunyai sistem

		Selain itu, diperlukan proses pengontrolan dan pengaturan statistik tiket yang terjual dengan jumlah pengunjung yang telah masuk sebagai laporan statistik <i>ticketing</i> terhadap bagian administrasi dan ekonomi.	secara digital di komputer, sehingga menyurutkan ongkos penyusunan era menyelaraskan tiket, menyurutkan santapan kertas, dan meniadakan pendataan dan penjualan yang bisa menyebabkan kesusahan kira konsumen atau perusahaan.		otentikasi untuk customer
3	PERANCANGAN SISTEM INFORMASI PENJUALAN TIKET ACARA BERBASIS WEB (2018)	Masih banyak acara-acara yang menjual tiket yang tidak menggunakan sistem berbasis web atau masih melakukan penjualan secara langsung di lokasi acara itu sendiri, hal ini menyebabkan penumpukan orang-orang yang ingin membeli tiket di kios booth acara. Selain itu, ada pula yang melakukan pembelian tiket secara online, namun pengunjung masih mengalami kesulitan dalam melakukan registrasi ketika sudah berada di lokasi acara yang telah diselenggarakan .	Membuat sebuah website yang berbeda dari yang lain, dimana pembeli dapat langsung mencetak tiket setelah melakukan pembayaran tiket tanpa harus melakukan registrasi terlebih dahulu. Hal ini membutuhkan menjelang meremehkan tamu bagian dalam mengerjakan pembicaraan pembelian tiket.	Aplikasi mempunyai sistem autentikasi untuk customer Aplikasi mempunyai sistem komentar Aplikasi mempunyai sistem tingkatan paket pada setiap event	Aplikasi tidak mempunyai aktor pemegang event Semua event hanya dikelola oleh admin

4	Sistem Marketplace Event dan Pemesanan E-Ticket (Studi Kasus : Universitas Islam Indonesia) (2020)	Penyelenggaraan acara seringkali meniti bantahan bagian dalam meluaskan advertensi dan memindahkan kandidat yang diinginkan. Pada orientasi lain, cekel juga terbiasa memperoleh petunjuk tuntas peri rencana yang akan diselenggarakan . Namun, petunjuk yang segar secara acak bisa memicu sejumlah cekel tidak memperoleh petunjuk tersebut.	Sehubungan dengan permasalahan tersebut, maka dibangunlah sistem UII Card Event Marketplace pada website yang sebelumnya digunakan pada setiap acara UII dan terhubung dengan sistem e-ticketing. Tujuan dari sistem ini adalah sebagai alat promosi dan sebagai platform khusus bagi para pengikut yang ingin mengetahui acara apa saja yang akan diadakan. Selain itu, kehadiran sistem tiket elektronik dapat mengurangi proporsi pembelian tiket acara.	Aplikasi mempunyai aktor pemegang event Aplikasi mempunyai sistem autentikasi untuk customer	Aplikasi tidak mempunyai sistem kelas pada tiket
5	PERANCANGAN SISTEM INFORMASI E-MARKETPLACE EVENT ONLINE UNTUK MENINGKATKAN PEMASARAN DIGITAL BERBASIS MOBILE (2023)	Acara-acara yang tersebar tidak dapat mencapai pengetahuan masyarakat mengenai event tersebut, hal ini disebabkan oleh informasi yang tersedia hanya pada salah satu platform tertentu.	Seorang peneliti akan membuat sebuah aplikasi mobile bernama "Eventloc", yang akan menjadi sarana penyedia informasi dan penjualan tiket event online. Aplikasi ini akan menjadi tempat untuk semua acara yang tersebar di berbagai	Aplikasi mendukung selain event seperti lomba, expo, festival Aplikasi mempunyai sistem gamifikasi Aplikasi dapat mencetak sertifikat ketika pengguna sudah	Tidak memiliki platform web

		platform, dengan cara menyatukan mereka dalam aplikasi Eventloc. Selain itu, aplikasi ini akan menambahkan metadata pada setiap acara, sehingga memudahkan pencarian acara berdasarkan jenis, kategori, atau tag-tag yang terkait.	mengikuti acara	
--	--	--	-----------------	--

2.2 Teori Terkait

2.2.1 Event

Event seringkali bisa dimaknai sebagai hal-hal yang dibuat oleh sekumpulan/organisasi yang diadakan untuk menggambarkan sesuatu yang bermakna dalam kehidupan manusia, baik sendiri maupun secara organisasi, dan terkait dengan budaya, adat, agama, atau tradisi (Noor, 2013).

Any Noor (2013) menyatakan event dapat dibedakan berdasarkan jenis-jenis kategori spesial event, yaitu:

1. *Cultural Event*

dapat diartikan sebagai event yang kegiatannya erat hubungannya dengan kebudayaan atau memiliki strata nilai kesosialan yang memiliki strata yang meningkat dalam masyarakat. Dengan searah teknologi yang berkembang secepat pesat, pembuatan acara budaya semakin menarik dan

dapat disesuaikan dengan nilai sekitar dan kondisi era masa depan, juga menghasilkan rangkaian yang efektif dan harmonis.

2. *Leisure Event*

Merupakan kategori event yang dibuat oleh sekumpulan/organisasi yang berkembang pesat dalam kegiatan olahraga, juga seringkali mengandung makna pertandingan dan menarik khalayak pengunjung.

3. *Organizational Event.*

Bentuk-bentuk acara yang diadakan dalam acara organisasi adalah hal-hal yang diarahkan dengan tujuan organisasi. Seperti bentuk acara dalam acara organisasi termasuk konferensi yang diadakan oleh pameran/expo, partai politik yang dijalankan oleh kolektif/perusahaan yang sejalan dengan kepentingan kolektif/perusahaan itu.

4. *Personal Event*

Seringkali diartikan sebagai kegiatan yang melibatkan kolektif yang berasal dari orang sekitar, dan dapat dikatakan lebih sederhana. Misalnya, acara-acara seperti pesta pernikahan atau acara serupa termasuk dalam kategori acara pribadi.

2.2.2 E-Ticketing

Jerry Septian Massie (2020) menyatakan bahwa *e-ticketing* adalah metode bagi individu untuk membeli tiket *event* lokal. Fasilitas tersebut didesain untuk memudahkan kolektif orang-orang yang tidak memiliki jalan masuk ke sistem seperti ini, sehingga mereka dapat memperluas kegiatan dan menyebarkan tiket

secara online melalui situs web. Sistem e-tiket ini menyamankan pengguna untuk membeli tiket untuk segala acara dari situs web. Tiket bisa dibeli dengan kartu kredit/debit, cek, atau *cash*.

E-ticketing memiliki kelebihan yaitu dapat menghemat biaya produksi karena tidak perlu mencetak tiket fisik. Selain itu, pelanggan jangan merasa heran dengan tiket yang tak terlacak karena dapat dilihat kembali dan dicetak ulang. E-ticketing juga memiliki kode unik dan barcode validasi, sehingga persentase probabilitas tiket dapat dipalsukan oleh orang yang tidak bertanggung jawab akan minim (Yugianus, 2014).

2.2.3 Web

The World Wide Web juga dikenal sebagai laman *web*, hanyalah salah satu komponen dari *internet*. *Web* adalah komponen berbagi informasi yang berjalan di internet. Ia menggunakan Protokol *HTTP*, umumnya dikenal sebagai *HTTP*. Ini hanyalah salah satu dari banyak bahasa yang digunakan oleh perangkat untuk berkomunikasi dan berbagi data di *internet*. *Web* menggunakan *browser* yang mengakses dokumen yang dipublikasikan di internet yang disebut halaman laman web. Dokumen-dokumen ini biasanya saling terkait sehingga memungkinkan seseorang untuk melompat dari satu halaman laman *web* ke halaman laman *web* lainnya melalui *hyperlink*. Dokumen dapat berisi teks, grafik, audio, dan *video*. *Web* mendukung dokumen berformat khusus. *Dokumen* harus ditulis dalam bahasa markup seperti *HTML* yang dapat mendukung *hyperlink* dokumen. Ada sangat banyak *browser* yang telah dirilis hingga saat ini untuk memungkinkan pengguna *internet* mengakses *Web*. (Ozkaya & Islam, 2019)

2.2.4 QR Code

QR Code adalah suatu jenis kode matriks atau kode batang dua dimensi yang dikembangkan oleh Denso Wave, sebuah divisi Denso Corporation yang merupakan sebuah perusahaan Jepang dan dipublikasikan pada tahun 1994. Agar dapat membaca QR Code diperlukan sebuah pembaca atau pemindai berupa software yaitu QR Code Reader atau QR code Scanner yang harus diinstal pada perangkat telepon mobile. QR merupakan singkatan dari quick response atau respons cepat, yang sesuai dengan tujuannya adalah untuk menyampaikan informasi dengan cepat dan mendapatkan respons yang cepat pula. Berbeda dengan kode batang, yang hanya menyimpan informasi secara horizontal, kode QR mampu menyimpan informasi secara horizontal dan vertikal

2.2.5 Java

Rauf (2019) menyatakan, *Java* adalah salah satu bahasa pemrograman paling populer di dunia. Diperkirakan lebih dari tujuh miliar perangkat dan sembilan juta *developer* menggunakannya. *Java* banyak digunakan di berbagai sistem, seperti sistem penerbangan, *ATM*, *handphone*, dan komputer. Popularitasnya melonjak di tahun 1990-an karena dianggap ideal untuk membuat aplikasi *online*.

Dibandingkan dengan bahasa pemrograman lain, *Java* dikenal lebih mudah dipelajari. Menurut James Milligan, direktur teknologi digital di perusahaan rekrutmen Hays, "Konsep dasar *Java* bisa dikuasai melalui kursus *online*."

Selain itu, *Java* dianggap sebagai pondasi yang baik untuk mempelajari bahasa pemrograman lainnya. Jadi, bagi yang ingin belajar *coding*, mencoba *Java* sangat dianjurkan. *Java* juga merupakan perangkat lunak *open source*. Artinya,

kode sumbernya tersedia secara gratis dan dapat disebarluaskan serta dimodifikasi. Dengan konsep *open source*, *developer* memiliki lebih banyak kebebasan untuk berkreasi dan berbagi karyanya.

Saat ini, siapa saja dapat mengakses *Java* secara gratis melalui situs *web Oracle*. Mereka menyediakan *Java Development Kit (JDK)* yang bisa diunduh. *JDK* berisi semua yang dibutuhkan untuk membangun aplikasi dan komponen menggunakan bahasa pemrograman *Java*.

2.2.6 Spring Framework

Tanpa adanya alat dan teknologi yang baik, pengembangan perangkat lunak akan menurunkan efisiensi dalam pembuatan aplikasi. *Spring Framework* adalah sebuah *framework open source* yang berbasis *Java*, dibuat untuk mempermudah pengembangan perangkat lunak *Java*. Sebagai sebuah *framework* yang berbasis *Java*, *Spring* juga menerapkan kerangka berbasis objek. *Spring Framework* mempermudah pengembangan aplikasi, baik itu aplikasi *web*, aplikasi yang berdiri sendiri, aplikasi *enterprise*, dan lain-lain. *Spring Framework* mempermudah pengembangan *enterprise Java*. (Walls, 2014).

2.2.7 Spring Boot

Spring Boot adalah ekstensi *open source* dari *Spring Framework* yang dirancang untuk menyederhanakan pengembangan aplikasi *Spring*. Popularitas *Spring Boot* sebagian besar disebabkan oleh kemampuannya untuk membuat aplikasi berbasis *Spring* yang mandiri, siap produksi, dalam waktu singkat.

Spring Boot dirilis pada April 2014 untuk mengurangi beban pengembangan aplikasi *web* berbasis *Java*. Dengan adanya proyek ini memungkinkan pengembang untuk lebih fokus pada logika bisnis daripada kode teknis *boilerplate* dan

konfigurasi terkait. Spring Boot bertujuan untuk membuat aplikasi mandiri berbasis Spring yang siap produksi dengan sedikit perubahan konfigurasi dari pihak developer aplikasi. Spring Boot mengambil pandangan "berpendirian" terhadap Spring Framework, sehingga developer aplikasi dapat dengan cepat memulai dengan apa yang mereka butuhkan. Ini menyediakan lapisan tambahan antara Spring Framework untuk pengguna untuk menyederhanakan aspek konfigurasi tertentu. (Musib & Long, 2022)

2.2.8 PostgreSQL

Menurut Waruwu, T. S. (2019), *PostgreSQL*, atau dikenal juga sebagai *Postgres*, adalah sistem manajemen basis data relasional *open-source* yang mengutamakan kreativitas, fleksibilitas, dan kompatibilitas. *PostgreSQL* mampu melawan kompetitor yang serupa dengan basis data relasional besar lainnya dari vendor seperti *Oracle*, *MySQL*, *SQL Server*, dan lain-lain.

PostgreSQL memiliki banyak kemampuan yang mirip dengan basis data komersial lainnya, termasuk dukungan untuk perintah *SQL*. Dengan menggunakan perintah *SQL*, administrator basis data dapat berinteraksi dengan basis data *PostgreSQL* dengan lebih mudah, terutama untuk tugas manipulasi data seperti memasukkan, memperbarui, atau menghapus data.

Selain berlisensi *open source*, kelebihan *PostgreSQL* adalah dapat digunakan sebagai *database* pada berbagai bahasa pemrograman, dan juga pemrograman *desktop* maupun *web*.

2.2.9 Javascript

Javascript adalah bahasa pemrograman yang dieksekusi oleh browser dan berbasis pada client. Halaman jejaring ini dapat melakukan proses tambahan yang sesuatu yang tidak bisa dilakukan oleh skrip HTML pada umumnya. (Arifin, 2008).

Berdasarkan hasil penelitian Mitropoulos, Louridas, Salis, & Spinellis, (2019), *Javascript* mempresentasikan pengalaman interaktif bagi pengguna jejaring dengan banyak implementasi untuk mengubah elemen halaman web menggunakan Document Object Model. Hal ini disebabkan oleh kemudahan penggunaan skrip yang dapat digunakan kembali dan popularitas yang tinggi dari banyaknya *Application Programming Interface (API)* dan *flow data* dari *framework*.

2.2.10 ReactJS

Berdasarkan penelitian yang dilakukan oleh Tanudjaja, D., & Tanone, R. (2021), *React* adalah sebuah library yang dapat cepat dalam meng-render konten pada sisi client, hal ini disebabkan oleh penggunaan *Virtual Document Object Model (VDOM)* yang melihat yang serupa pada perubahan data pada *Document Object Model (DOM) tree* dengan *Document Object Model (DOM) tree* pada browser. *React* cuma akan melakukan rendering ulang terhadap komponen yang mengalami perbedaan *state*.

Keuntungan dari penggunaan *React* adalah penerapan komponen yang dapat digunakan kembali, sehingga skrip yang disusun akan lebih tertata, mudah dikendalikan dan dapat memperbanyak kecepatan proses *development*. Komponen *React* bisa dibuat dengan dua cara, yaitu dengan menggunakan komponen fungsi (*functional component*) atau komponen kelas (*class component*). Secara dasar, *developer* disarankan untuk menggunakan *functional component* karena penulisan kode menjadi lebih nyaman dibaca oleh *developer* lain.

2.2.11 Remix

Menurut Landgraf & Solomon (2023), *Remix* adalah *framework web* yang fokus pada pemanfaatan mendalam *platform web*. *Remix* bertujuan menghadirkan "pengalaman pengguna yang cepat, lancar, dan tangguh" menggunakan standar *web*. Formulir dan *tag HTML*, *URL*, *cookie*, *tag meta*, *header HTTP*, dan *Web Fetch API* semuanya menjadi elemen penting dalam *Remix*. Konvensi, pengungkit, dan primitif *Remix* dirancang dengan cermat sebagai lapisan abstraksi dari API dan standar web yang ada. Ini membuat *Remix* berbeda dari framework populer lainnya yang terasa lebih terpisah dari platform web.

React menjadi peran penting dalam *Remix*. *Remix* memanfaatkan fitur terbaru *React* secara wajar. Dengan *React 18*, *React* menjadi lebih canggih. Fitur terbaru *React* lebih ditujukan untuk pembuat *framework* daripada pembuat aplikasi. *Remix* menyediakan abstraksi yang diperlukan untuk memanfaatkan kemajuan terbaru ini.

Ketika dipasangkan dengan *React*, *Remix* menggunakan *routing* dan pengambilan data sisi klien, menciptakan pengalaman serupa dengan membangun aplikasi *single page application* (SPA) dengan *React*. Namun, *Remix* memiliki cakupan yang lebih luas daripada *React* dan menangani masalah tambahan dalam pengembangan *web*, seperti *caching*, sesi pengguna, dan mutasi data. Ini menjadikan *Remix* sebagai *framework web*.

Awalnya, *Remix* dimaksudkan sebagai *framework* berbasis lisensi. Namun pada Oktober 2021, tim pengembangan di balik *Remix* mengumumkan bahwa mereka telah mendapatkan pendanaan awal dan akan membuat *Remix open-source*. Pada November 2021, tim merilis *Remix v1* setelah 18 bulan pengembangan.

Setahun kemudian, pada Oktober 2022, Shopify mengakuisisi *Remix*. Tim *Remix* sekarang sepenuhnya fokus pada pengembangan *Remix* sambil tetap mengejar misi dan visi yang sama di Shopify.

Tim *Remix* telah merangkum filosofinya dengan baik dalam dokumentasi *Remix*. Beberapa poin yang ditekankan tim *Remix* adalah sebagai berikut:

- Menghindari Abstraksi Berlebihan: Dalam *Remix*, API dimaksudkan sebagai lapisan abstraksi tipis di atas platform web. Kesederhanaan adalah yang utama. *Remix* tidak menciptakan kembali sesuatu yang sudah ada.
- Melihat ke Depan dan ke Belakang: *Remix* memadukan teknologi mutakhir dan memadukannya dengan standar web yang teruji untuk menciptakan pendekatan baru. *Remix* memanfaatkan *streaming HTTP2* dan penerapan *edge* tetapi juga menggunakan formulir *HTML*, *cookie*, dan *URL*.
- Meningkatkan Pengalaman Pengguna secara Bertahap: Tujuannya adalah untuk tetap setia pada perilaku *default browser* dan dapat kembali ke sana kapan pun.
- Membuka Potensi Penuh Tumpukan *Web*: *Remix*, seperti yang dikatakan tim *Remix*, adalah tentang model *client/server*.
- Peduli terhadap Ukuran *Bundle*: Tim di balik *Remix* sangat peduli dengan tab jaringan dan ukuran *bundle* aplikasi Anda. Tujuannya adalah untuk memuat *resource* lebih sedikit dan memuat halaman secepat mungkin.