# SKRIPSI

## PENGEMBANGAN SISTEM PENJUALAN TIKET ONLINE EVENT MUSIK DENGAN FITUR PREVIEW

Disusun oleh:

Friendy Budi Sulistyo (102410101009)

**PROGRAM STUDI SISTEM INFORMASI**

**UNIVERSITAS JEMBER**

**2016**

# BAB 1. PENDAHULUAN

Bab ini merupakan bab awal dari laporan tugas akhir. Pada bab ini akan dibahas tentang latar belakang, rumusan masalah, tujuan dan manfaat, batasan masalah, dan sistematika penulisan.

## Latar Belakang

Perkembangan teknologi telekomunikasi dan internet menyebabkan terjadinya perubahan kultur budaya manusia. Dalam era yang disebut *information age* ini, media elektronik dan komputer menjadi media yang diandalkan untuk melakukan berbagai aktifitas dan kegiatan. Perkembangan teknologi berkembang amat pesat. Setiap saat dikembangkan perangkat - perangkat baru untuk mendukung kemudahan hidup manusia. Infrastruktur teknologi saat ini bukan hanya sekedar pelengkap saja namun sudah menjadi kebutuhan pokok, salah satunya di dunia penjualan. Teknologi yang saat ini mampu memberikan sarana penjualan yang lebih menarik, salah satunya adalah *e-commerce*.

*E-commerce* adalah metode transaksi penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti [*internet*](https://id.wikipedia.org/wiki/Internet) atau [*televisi*](https://id.wikipedia.org/wiki/Televisi), [*www*](https://id.wikipedia.org/wiki/Www), atau [jaringan komputer](https://id.wikipedia.org/wiki/Jaringan_komputer) lainnya. *E-commerce* dapat melibatkan transfer dana elektronik, pertukaran data elektronik, sistem manajemen inventori otomatis, dan sistem pengumpulan data otomatis.

Dalam ilmu perdagangan, metode *e-commerce* juga termasuk dalam ilmu ekonomi, yang mana mempelajari perilaku manusia dalam memilih dan menciptakan kemakmuran. Inti permasalahan ekonomi adalah adanya ketidakseimbangan antara kebutuhan manusia yang tidak terbatas dengan alat pemuas kebutuhan yang jumlahnya terbatas. Permasalahan tersebut kemudian menimbulkan kelangkaan.

Namun ada sebuah peningkatan *trend* untuk mengaplikasikan ide dan metode ekonomi dalam konteks yang lebih luas. Fokus analisis ekonomi adalah “Pembuatan Keputusan” dalam berbagai bidang di mana orang dihadapi pada pilihan – pilihan. Tentu saja hal ini juga berlaku untuk metode *e-commerce.*

Gary Becker dari University of Chicago adalah seorang perintis *trend* ini. Dalam artikel – artikelnya ia menerangkan bahwa ekonomi seharusnya tidak ditegaskan melalui pokok persoalannya, tetapi sebaiknya ditegaskan sebagai pendekatan untuk menerangkan perilaku manusia.

Menurut Scroder (1967), dalam bidang ilmu pengambilan keputusan dan sistem informasi, mesin komputer digambarkan sebagai sistem dengan kapasitas terbatas untuk memproses informasi. Fenomena *“Overload Information”* didefinisikan sebagai penurunan tingkat respon karena data informasi melebihi batas kapasitas. Dari konsep di atas, ilmuwan yang ahli dalam bidang ini menggambarkan manusia sebagai sistem pengolah informasi dengan kapasitas pemrosesan informasi yang terbatas. Bila beban informasi yang diberikan meningkat, *“Decision Maker”* atau pembuat keputusan juga akan meningkatkan usahanya untuk mengolahnya. Begitu data informasi yang masuk melampaui kapasitas pemrosesan, seseorang akan kesulitan dengan memproses informasi produk dan tingkat respon akan menurun. (Schroder et al., 1967; Grisé and Gallupe, 2000).

Namun beberapa peneliti kemudian menyatakan bahwa ternyata definisi dari konten informasi yang berlebih dikaji ulang dari data asli mereka, berlawanan dengan kesimpulan bahwa ''lebih banyak informasi lebih baik". Untuk memperbaiki argumen tentang definisi *“Overload Information”* dan hasil penjelasan data yang berbeda dengan lapangan, beberapa peneliti menggabungkan faktor-faktor lain, termasuk kualitas informasi, variabilitas atribut, kesamaan alternatif, dan distribusi tingkat atribut di antara alternatif, atau menyesuaikan metodologi yang berbeda untuk menguji efek *“Overload Information”* yang tetap menghasilkan hasil yang tidak konsisten. (Malhotra et al., 1982; Malhotra, 1982, 1984; Muller, 1984; Keller dan Staelin, 1987; Lurie, 2004; Lee dan Lee, 2004).

Penerapan sistem *e-commerce* dalam penjualan tiket event musik bertujuan untuk memudahkan pemahaman dan akses informasi serta pemesanan tiket untuk menghindari terjadinya fenomena *“Overload Information”* dalam sistem *e-commerce* penjualan tiket online ini terhadap konsumen ketika EO menyelenggarakan suatu event yang di dalamnya terdapat artis-artis favorit yang akan tampil. Dalam penjualan tiket manual, mayoritas para konsumen merasa kesulitan untuk mendapatkan tiket event karena harus bersusah payah untuk mencari agen tiket, pengisian formulir, pembayaran, dan lain-lain, yang mengakibatkan para konsumen merasa “kurang praktis” karena tata cara perolehan tiket secara manual membutuhkan prosedur – prosedur dan pertemuan langsung dengan agen penjual tiket.

Side Story Family adalah salah satu EO (*Event* Organizer) atau penyedia layanan jasa dalam bidang entertainment ini. Namun penulis mengetahui bahwa saat ini kegiatan transaksional masih menggunakan cara manual. Sebelum menggunakan internet, konsumen harus datang ke agen-agen dari EO untuk memesan tiket, tidak jarang pula konsumen kecewa karena tiket yang tersedia telah habis terjual. Hal ini sering menjadi permasalahan karena konsumen tidak dapat melihat jadwal dan jumlah tiket yang tersisa sedangkan pihak EO tidak dapat menginformasikan secara langsung kepada konsumen.

Berdasarkan masalah tersebut penulis merancang sistem penjualan tiket event musik *online* melalui *web*site dengan fitur *preview* untuk memudahkan pemahaman informasi yang berhubungan dengan event dan EO Side Story Family, baik profil maupun layanan melalui *website* sehingga menciptakan efektifitas dan efisiensi dalam hal pemasaran dan pelayanan jasa. Selain itu, penulis juga ingin mengetahui bagaimanakah hasil penerapan metode ini dalam sistem pemasaran dan penjualan tiket ini, untuk itu penulis melakukan penelitian untuk mengetahui hasil tersebut. Dari dasar permasalahan inilah penulis ingin mengambilnya sebagai judul tugas akhir.

## Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, dapat diidentifikasikan masalah-masalah yang terjadi, yaitu:

1. Bagaimana melakukan analisa dan perancangan sistem penjualan *online* tiket event musik menggunakan *website*.
2. Bagaimana menampilkan informasi dan pemesanan tiket event secara efektif dengan fitur *preview*, untuk mempermudah konsumen dalam memperoleh informasi dan tiket event.

## Tujuan dan Manfaat

Berikut ini merupakan tujuan yang ingin dicapai dan manfaat yang ingin diperoleh dalam penelitian ini.

### Tujuan

Tujuan dari penelitian ini adalah:

1. Untuk merancang sistem penjualan tiket *online* event musik dengan fitur preview.
2. Untuk memberikan kemudahan dalam pemesanan tiket event musik.
3. Sebagai bentuk penerapan teknologi saat ini untuk pengembangan sistem penjualan tiket event musik.
4. Untuk mengetahui hasil penerapan tiket *online* dengan fitur *preview* menggunakan metode skala Likert.

### Manfaat

Hasil penelitian dan pembuatan sistem ini diharapkan mampu untuk membantu EO Side Story Family dalam mempromosikan event yang akan diselenggarakan dan membantu penjualan tiket event melalui *website*. Selain dari pihak EO, pihak konsumen juga diharapkan merasakan kemudahan dalam memperoleh informasi dan pemesanan tiket event*.* Sistem ini dijadikan sebagai bentuk penerapan teknologi saat ini yang mampu melakukan berbagai hal dalam mempermudah kehidupan manusia.

Manfaat penelitian bagi Event Organizer:

1. Membantu dalam mempromosikan event.
2. Memudahkan penjualan tiket event.
3. Membantu dalam pengamanan data transaksi dengan penyimpanan komputer atau database online.
4. Mengetahui *feedback* dari konsumen dan hasil penjualan tiket online tentang event yang diselenggarakan dengan fitur-fitur yang ada di dalam sistem.

Manfaat penelitian bagi konsumen:

1. Memberikan fasilitas untuk pembelian tiket event.
2. Menyediakan fasilitas pembayaran tiket event.
3. Memberikan informasi event yang bisa diakses kapanpun dan dimanapun.
4. Memberikan fasilitas pada konsumen untuk memberikan ulasan seputar event.
5. Memberikan informasi untuk konsumen yang membutuhkan jasa dan pelayanan dari EO Side Story Family.

Manfaat penelitian bagi penulis:

1. Sebagai sarana belajar untuk mengembangkan dan menerapkan ilmu pengetahuan yang telah didapat untuk dipraktekkan dalam dunia nyata.
2. Menambah bekal pengetahuan yang dapat dipergunakan guna menghadapi kehidupan bermasyarakat di masa yang akan datang.
3. Menambah wawasan pengetahuan di dalam maupun di luar lingkungan akademik sesuai dengan studi kasus yang diteliti.
4. Sebagai sarana untuk menambah pengalaman dalam menganalisa suatu kasus dan menemukan bagaimana solusi yang harus diterapkan.

Manfaat penelitian bagi pembaca:

Penelitian ini dapat digunakan sebagai referensi dalam suatu penelitian atau menyusun skripsi, pengalaman, pengetahuan tambahan dan mengaplikasikan semua teori yang didapat selama menempuh kuliah sehingga dapat digunakan sebagaimana mestinya.

## Batasan Masalah

Agar tidak terjadi penyimpangan dalam proses penelitian dan perancangan sistem, maka ditetapkan beberapa batasan permasalahan. Adapun batasan masalah dalam penelitian ini adalah:

1. Program ini mengenai sistem penjualan *online* tiket event musik.
2. Sistem yang dibangun berbasis *website.*
3. User atau pengguna sistem ini adalah admin dari EO (*Event Organizer*) dan konsumen pemesan tiket.
4. Output yang dihasilkan dari program ini berupa tiket event, dan hasil perhitungan skala Likert.

## Sistematika Penulisan

1. Pendahuluan

Bab ini menjelaskan tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan skripsi.

1. Tinjauan Pustaka

Bab ini menjelaskan landasan teori dan konsep penelitian, dan kajian metode *fitur preview* yang berkaitan dengan rumusan masalah dalam penelitian.

1. Metode Penelitian

Bab ini menjelaskan metode penelitian, tempat dan waktu penelitian, metode pengumpulan data, metode analisis data, dan pengembangan aplikasi.

1. Pengembangan Sistem

Bab ini berisi tentang langkah-langkah yang ditempuh dalam proses menganalisis dan merancang sistem yang akan dibangun meliputi desain, *coding*, dan pengujian sistem.

1. Hasil dan Pembahasan

Bab ini menjelaskan secara rinci pemecahan masalah melalui analisis yang disajikan dalam bentuk deskripsi dibantu dengan ilustrasi berupa tabel dan gambar untuk memperjelas hasil penelitian.

1. Penutup

Bab ini terdiri atas kesimpulan atas penelitian yang telah dilakukan dan saran untuk penelitian selanjutnya.

# BAB 2. TINJAUAN PUSTAKA

Dalam bab ini dijelaskan teori-teori dan pustaka yang digunakan dalam penelitian. Teori-teori dan pustaka yang digunakan dan dibahas dalam penelitian ini, yaitu penelitian terdahulu, konsep dasar *e-commerce*, konsep dasar penjualan, konsep dasar fitur *preview*, *Event Organizer,* bahasa pemrograman *PHP* & *MySQL*, dan aplikasi web server *XAMPP*.

## Penelitian Terdahulu

Penelitian ini dilakukan berdasarkan teori dan studi yang dilakukan oleh peneliti terdahulu. Adapun penelitian tersebut antara lain:

1. Peneliti terdahulu dengan judul Implementasi *e-commerce* sebagai media penjualan *online* Studi Kasus Pada Toko Pastbrik Kota Malang (Shabur Miftah: 2015). Penelitian ini mengambil judul tentang Implementasi *E-Commerce* Sebagai Media Penjualan Online. Tujuan dari penelitian ini adalah untuk mengimplementasikan website *e-commerce* yang berfungsi sebagai media promosi dan penjualan elektronik. Serta mengetahui kendala yang dihadapi dalam mengimplementasikan website *e-commerce*. Jenis penelitian yang dilakukan adalah kualitatif. Implementasi *e-commerce* dengan menggunakan software *opencart* pada toko Pastbrik akan dapat membantu mengurangi biaya yang dikeluarkan serta dapat menyampaikan informasi secara detail mengenai produk maupun harga spesial yang diberikan kepada konsumen secara *online* dan memudahkan proses transaksi tanpa harus datang ke toko secara langsung sehingga dapat bersaing dengan toko yang sejenis dan mendapatkan hasil yang lebih maksimal.
2. Peneliti terdahulu dengan judul Sistem Informasi Penjualan Berbasis Web (E-Commerce) Menggunakan Joomla Pada Mutiara Fashion (Elizaandayani Ginting: 2013). E-Commerce merupakan suatu kontak transaksi perdagangan antara penjual dan pembeli dengan menggunakan media internet. Keuntungan yang diperoleh dengan menggunakan transaksi melalui e-commerce adalah untuk meningkatkan pendapatan dengan menggunakan penjualan online yang biayanya lebih murah dan juga sekaligus biaya-biaya operasional seperti kertas, pencetakan katalog. Adapun sistem e-commerce yang digunakan adalah dengan joomla, joomla adalah salah satu nama web developer yang berbasisi CMS (Content Management System) dan bersifat open source yang dapat dengan lelusan untuk melakukan content. Dari berbagai penjelasan yang telah diuraikan dalam laporan ini, maka dapat disimpulkan beberapa hal sebagai berikut:
3. Aplikasi ini menyajikan informasi terkait dengan Mutiara Fashion yang ditujukan pada konsumen.
4. Dengan adanya aplikasi ini, informasi tentang produk terbaru di Mutiara Fashion dapat terpublikasi dengan baik.
5. Dengan adanya aplikasi ini maka konsumen dapat melihat info tentang produk.
6. Penelitian terdahulu dengan judul Analisa dan Perancangan Sistem Informasi Penjualan *Online (E-Commerce)* pada CV.Selaras Batik Menggunakan Analisis Deskriptif (Himawan1, Asep Saefullah2 dan Sugeng Santoso3: 2014). Batik merupakan salah satu pakaian tradisional yang menjadi ciri khas dan kebanggaan bagi bangsa Indonesia, dimana batik juga telah dikenal secara luas baik dalam skala nasional dan internasional. Peluang dalam kegiatan bisnis penjualan batik yang meningkat beberapa waktu ini dilihat oleh banyak pihak sebagai prospek yang menjanjikan. Penjualan batik secara umum di beberapa wilayah di Indonesia masih dilakukan dengan cara yang sederhana dan klasik. Untuk dapat menjangkau lebih banyak pelanggan serta memperluas area pemasaran sekaligus promosi batik, maka diperlukan media yang dapat menunjang kegiatan promosi, transaksi penjualan dan informasi batik dengan optimal. Salah satu media yang menjadi perhatian masyarakat sekarang ini adalah dengan menggunakan media *website online*.

*E-Commerce* merupakan salah satu kegiatan transaksi bisnis baik barang dan jasa yang dilakukan secara elektronik dengan menggunakan jaringan internet CV.Selaras Batik menyadari bahwa peranan media *website* dapat digunakan untuk meningkatkan omset penjualan dan juga sebagai media sarana dalam memasarkan produk batik yang mereka miliki untuk menjangkau pasar yang lebih luas lagi serta dapat meminimalkan biaya-biaya operasional seperti biaya iklan dan juga biaya sewa toko secara fisik. Penulisan paper ini bertujuan untuk menganalisa pengaruh dari penggunaan media *website online* dalam hal ini adalah *e-commerce* dengan menggunakan pendekatan B2C (*Business-to-Customer*) dalam upaya meningkatkan transaksi penjualan batik pada CV.Selaras Batik dan juga media penyampaian informasi baik pada para pelanggan dan masyarakat umum dalam memperoleh informasi mengenai produk-produk batik yang dipasarkan oleh CV.Selaras Batik.

1. Penelitian terdahulu dengan judul Sistem Informasi Pemesanan Tiket Pesawat Berbasis Web pada Bana Tour (PT. Wali Angkasamitra Utama) (Nur Aeni Hidayaha, Nia Kumaladewib dan Serlyta Efryllac: 2010). Bana Tour (PT Wali Angkasamitra Utama) adalah salah satu biro perjalanan berdasarkan lisensi yang diperoleh dari Dinas Pariwisata DKI Jakarta tentang "Biro Perjalanan Lisensi Permanen". Bana Tour (PT Wali Angkasamitra Utama) juga terdaftar sebagai anggota penuh di ASITA (Asosiasi Perusahaan Perjalanan Wisata) dengan sertifikat keanggotaan no. NIA: 01294 / VIII / DPP / 08. Di Bana Tour, divisi yang menangani pemesanan tiket dari pelanggan Bana Tour adalah divisi tiket. Pemesanan tiket sebelumnya telah dilakukan via telepon atau pelanggan datang langsung ke kantor Bana Tour. Peneliti mengamati proses pemesanan tiket melalui telepon tidak efektif dan efisien dari sisi waktu, yang telah digunakan untuk mencatat nama dan identitas pelanggan. Dan juga tidak ada informasi tentang penerbangan keberangkatan di situs Bana Tour. Sehubungan dengan hal tersebut, peneliti membangun sistem informasi pemesanan tiket berbasis *website*. Tujuan penelitian adalah data pemesanan tiket dari pelanggan akan diatur dengan baik dan memberikan informasi jadwal keberangkatan penerbangan. Metode pengembangan sistem yang digunakan adalah waterfall berurutan dengan pengembangan model berorientasi objek. Langkah-langkah untuk metode waterfall adalah *System Initiation*, *System Analysis*, *System Design*, dan *System Implementation*. Dan untuk database, sistemnya menggunakan PHP dan MySQL.

Dari hasil pembahasan pembuatan sistem informasi pemesanan tiket pesawat berbasis web ini, hasil yang didapat adalah sebagai berikut:

1. Bagian Ticketing Bana Tour mudah dalam memperoleh informasi pemesanan tiket pesawat dari para pelanggan terutama data nama dan nomor identitas tanpa harus mengulang kembali menyebutkan data-data pemesanan tersebut ke pelanggan. Karena bagian ticketing Bana Tour mendapatkan langsung data-data pemesanan dari pelanggan secara detail setelah pelanggan melakukan pemesanan tiket pesawat melalui Sistem Informasi Pemesanan Tiket Pesawat Berbasis Web Bana Tour ini.
2. Pelanggan Bana Tour dapat melakukan pemesanan tiket pesawat kapan pun dan dimana pun mereka berada karena sistem ini berbasis website. Sehingga dapat diakses dimana pun dan kapan pun.
3. Sistem Informasi Pemesanan Tiket Pesawat Berbasis Web ini memberikan informasi jadwal keberangkatan pesawat terbang, sehingga baik pihak Bana Tour maupun pelanggan dapat melihat informasi jadwal keberangkatan tersebut.
4. Hasil dari penilaian pimpinan Bana Tour terhadap sistem ini yaitu sistem dapat memberikan kemudahan bagi staf bagian ticketing dalam mendapatkan informasi dan data pemesanan tiket pesawat dari para pelanggan. Pimpinan Bana Tour juga dapat melihat data-data transaksi pemesanan tiket pesawat terbang yang terjadi setiap harinya.

Berdasarkan penelitian-penelitian terdahulu, bisa disimpulkan bahwa penerapan sistem *e-commerce* dalam penjualan *onlune* bertujuan untuk memudahkan pemahaman serta akses informasi dan untuk menghindari terjadinya fenomena *“Overload Information”*. Maka dari itu penerapan sistem *e-commerce* dengan fitur *preview* dalam sistem penjualan *online* tiket event musik ini dibuat untuk menampilkan informasi dan pemesanan tiket event secara efektif dan efisien, sebagai fasilitas untuk mempermudah konsumen dalam memperoleh informasi dan tiket event. Untuk meningkatkan stabilitas dan eksistensi Side Story Family Management, sistem ini diharapkan mampu membantu mempromosikan event yang akan diselenggarakan sehingga menghasilkan sebuah informasi akurat yang digunakan untuk penjualan tiket event maupun pemesanan jasa EO melalui *website*.

## Konsep Dasar *E-Commerce*

Menurut Laudon & Laudon (1998), *e-commerce* adalah suatu proses membeli dan menjual produk – produk secara elektronik oleh konsumen dan dari perusahaan ke perusahaan dengan komputer sebagai perantara transaksi bisnis.

Menurut David Baum (1999, pp. 36-34) yang diterjemahkan oleh Onno W. Purbo, *e-commerce* merupakan satu set dinamis teknologi, aplikasi, dan proses bisnis yang menghubungkan perusahaan, konsumen, dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan, dan informasi yang dilakukan secara elektronik.

Menurut Robert A. Zinko (2009), *e-commerce* merupakan suatu tindakan melakukan transaksi bisnis secara elektronik dengan menggunakan internet sebagai media komunikasi yang paling utama.

Secara singkat, *e-commerce* dapat diartikan yaitu semua bentuk transaksi penyebaran, penjualan, pembelian, serta pemasaran barang maupun jasa melalui media elektronik.

## Konsep Dasar Penjualan

Didalam manajemen perusahaan, penjualan dapat didefinisikan sebagai fungsi organisasi dan seperangkat proses untuk menciptakan, mengkomunikasikan, dan memberikan nilai lebih kepada konsumen dan untuk membina hubungan kepada mereka dengan cara yang menguntungkan organisasi dan para *stakeholders.*

Menurut *American Marketing Association,* penjualan adalah suatu aktivitas, kumpulan kebiasaan, dan proses untuk membuat, berkomunikasi, memberikan, dan bertukar penawaran yang memiliki nilai bagi konsumen, klien, mitra, dan masyarakat pada umumnya (AMA, 2007).

Menurut Philip Kotler (2009:22), Konsep penjualan adalah meyakini bahwa para konsumen dan perusahaan bisnis, tidak akan secara teratur membeli cukup banyak produk-produk yang ditawarkan oleh organisasi tertentu. Oleh karena itu, organisasi yang bersangkutan harus melakukan usaha penjualan dan promosi yang agresif.

## Konsep Dasar Fitur *Preview*

Menurut Christopher P. Furner dan Robert A. Zinko dalam jurnal “Electronic Markets”, konsep fitur *preview* atau bisa diartikan sebagai ulasan informasi yang ada didalam suatu produk, adalah informasi yang bertujuan guna memberikan pemahaman yang bisa memberikan suatu nilai lebih dan menjadi suatu daya tarik tersendiri untuk konsumen akan kepercayaan dan daya beli produk. Dalam penelitian khusus yang dilakukan oleh media, sampai sejauh manakah luasnya informasi akan karakteristik suatu produk yang dijual dapat mempengaruhi kepercayaan dan daya beli untuk konsumen.

Dalam hal ini, penulis menguraikan sebuah studi yang menerapkan konsep preview produk secara online dalam pengembangan sistem tiket event musik.

## EO (Event Organizer)

EO adalah penyedia jasa professional penyelenggara acara. Tugas EO adalah membantu kliennya (*client*) untuk dapat menyelenggarakan acara yang diinginkan. Menurut buku "EO, 7 Langkah Jitu Membangun Bisnis Event Organizer" yang ditulis oleh professional Event Organizer, Yudhi Megananda, EO dalam konteks sebuah bisnis memiliki definisi sebagai usaha dalam bidang jasa yang secara sah ditunjuk oleh clientnya, guna mengorganisasikan seluruh rangkaian acara, mulai dari perencanaan, persiapan, eksekusi hingga evaluasi, dalam rangka membantu mewujudkan tujuan yang diharapkan client dengan membuat acara. Setiap acara yang diselenggarakan bertujuan untuk memperoleh keuntungan di kedua belah pihak, baik penyelenggara maupun yang hadir pada saat kegiatan berlangsung. Keuntungan ini tidak harus bersifat material namun juga bisa bersifat non material.

## *PHP*

*PHP* adalah bahasa pemrograman *server-side* yang digunakan secara luas untuk menangani pembuatan dan pengembangan sebuah situs webdan bisa digunakan bersamaan dengan *HTML*. *PHP* dikembangan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh The PHP Group.

*PHP* disebut bahasa pemrograman *server-side* karena *PHP* diproses pada komputer server. Hal ini berbeda dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses pada web browser *client.*

Pada awalnya *PHP* merupakan singkatan dari *Personal Home Page.* Sesuai dengan namanya *PHP* digunakan untuk *website* pribadi. Dalam beberapa tahun perkembangannya, *PHP* menjadi bahasa pemrograman yang sangat populer karena selain powerful, *PHP* juga dapat digunakan secara gratis dan bersifat *Open Source.*

Saat ini *PHP* adalah singkatan dari *PHP: Hypertext Preprocessor,* suatu kepanjangan yang *rekursif,* yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri.

## MySQL

*MySQL* adalah sebuah perangkat lunak *Data Base Management System* (DBMS) yang didistribusikan secara gratis dibawah lisensi *General Public License* (GPL) dimana setiap orang bebas untuk menggunakannya, namun tidak diperbolehkan menjadi produk turunan yang bersifat komersial.

*MySQL* adalah pengembangan DBMS yang pertama yaitu *SQL (Structured Query Language)* untuk pengolahan dan pengoperasian *database*, terutama untuk seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah dan otomatis. Keandalan suatu sistem DBMS dapat diketahui dari cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah *SQL* yang dibuat oleh user maupun program-program aplikasinya. Sebagai *database* server, *MySQL* dapat dikatakan lebih unggul dibandingkan *database* server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query *MySQL* bisa sepuluh kali lebih cepat dari *PostgreSQL* dan lima kali lebih cepat dibanding *Interbase.*

*MySQL* adalah perangkat lunak yang pertama kali oleh didukung oleh bahasa pemprograman *script* untuk internet (*PHP* dan *Perl*). *MySQL* dan *PHP* adalah pasangan *software* pengembangan aplikasi web yang ideal. *MySQL* sering digunakan untuk membuat aplikasi berbasis web dengan menggunakan bahasa pemprograman *script PHP*.

## *XAMPP*

*XAMPP* adalah aplikasi gratis yang mendukung banyak sistem operasi yang mampu mengelola basis data dan merupakan kompilasi dari beberapa program untuk menjalankan fungsinya sebagai server yang berdiri sendiri (*localhost*), terdiri dari beberapa program seperti *Apache HTTP Server, MySQL database,* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* sendiri merupakan singkatan dari X (empat sistem operasi), Apache, MySQL, PHP, dan Perl.

# BAB 3. METODOLOGI PENELITIAN

## Metode Penelitian

Metode penelitian yang akan digunakan dalam membangun sistem meliputi studi literatur, pengumpulan data, metode perancangan sistem, dan fitur *preview*. Alur penelitian dapat dilihat pada gambar 3.1.

**Desain sistem**

**Implementasi sistem**

**Selesai**

**Analisis kebutuhan**

**Pemrograman sistem**

**Analisis Hasil pengolahan data**

**Pengujian sistem**

**Studi literatur**

- Buku  
- Jurnal online

-Situs internet

**Pengumpulan data**

- data EO  
- data Konser Musik

- kuisioner,

**Penerapan Fitur Preview**

**Mulai**

Gambar 3.1 Diagram alur penelitian

Sumber: Hasil Analisis (2017)

### Studi Literatur

Referensi yang dibutuhkan didapat dari berbagai literatur yaitu buku, jurnal *online*, contoh skripsi dan situs-situs internet. Tujuan dari studi literatur adalah untuk memperkuat landasan teori.

### Pengumpulan data

Pengumpulan data bertujuan untuk mendapatkan data yang diperlukan untuk mengembangkan aplikasi sistem online tiket konser musik data tersebut akan diolah menjadi sebuah informasi yang akan ditampilkan kepada *user.* Data yang dikumpulkan meliputi data EO yang telah penulis ajak bekerja sama, data konser musik yang tersedia, dan kuisioner. Berikut adalah cara memperoleh data yang diperlukan:

1. Data EO

Data EO adalah data yang berupa profil dari EO yang penulis ajak bekerja sama. Dalam hal ini penulis bekerja sama dengan salah satu EO yaitu side story family. Data yang diambil berupa profil usaha, pengalaman eo, serta data dokumentasi event.

1. Data Konser Musik

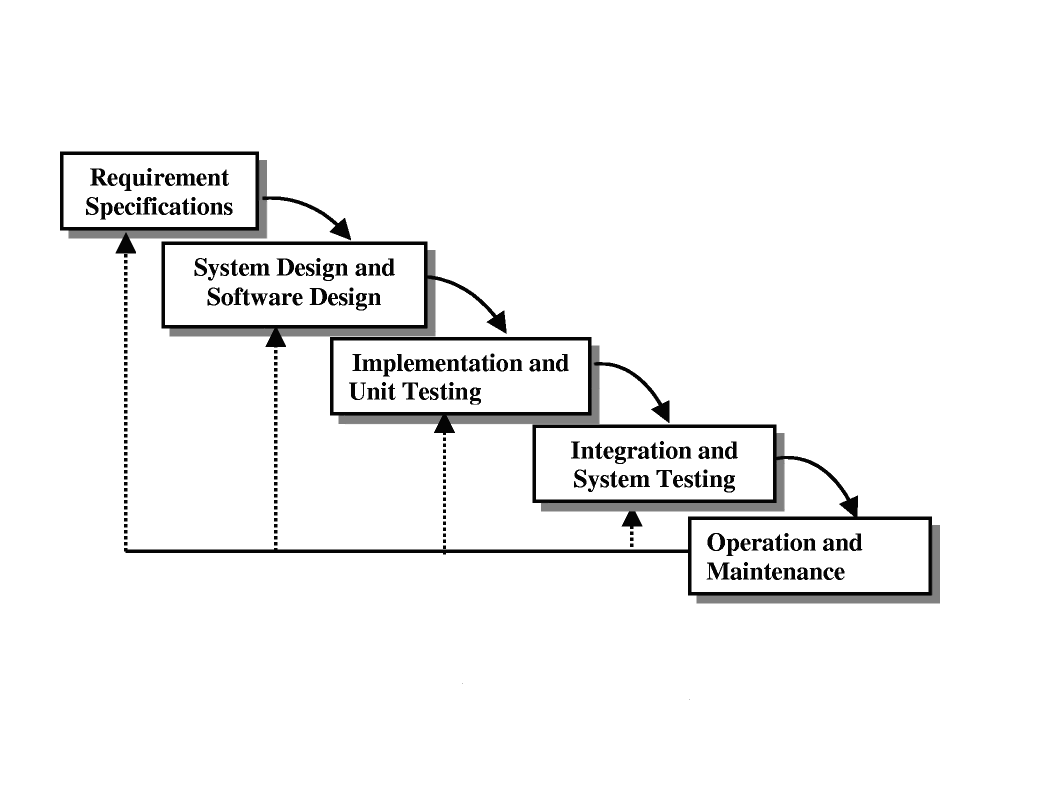
Merupakan data yang berisi tentang kegiatan konser musik yang tersedia baik di jember atau area sekitar yang sekiranya dapat dijadikan sebagai objek penelitian.

1. Kuisioner

Pengumpulan data menggunakan kuisioner untuk mendapatkan data nilai ketertarikan terhadap sebuah event. Kuisioner diberikan kepada user dari sistem yang penulis rancang dan telah menggunakan fitur atau melaksanakan skenario yang telah dibuat.

### Metode Perancangan Sistem

Model perancangan sistem yang digunakan adalah *System Development Life Cycle* (SDLC) dengan metode *Waterfall.* Menurut Pressman (1997) dalam Proboyekti (2011) model *waterfall* adalah model klasik yang bersifat matematis, berurutan dalam membangun *software*. Fase-fase dalam model *waterfall* dapat dilihat pada gambar berikut.



Gambar 3.2 Fase-fase dalam model *waterfall*

Sumber: Proboyekti, 2011

#### ***Requirements Analysis and Definition***

Pada tahap ini dilakukan pengumpulan kebutuhan secara lengkap melalui studi literatur dan survei lapangan kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh sistemyang akan dibangun. Pada penelitian ini, pengumpulan data didapat dari pihak EO Side Story, beberapa responden, dan jurnal-jurnal yang berkaitan dengan penelitian ini. Dari beberapa jawaban yang diterima tersebut, penulis bisa merancang apa saja fasilitas dan fitur yang harus dibuat. Hal ini sangat diperlukan mengingat sistemharus dapat berinteraksi dengan elemen-elemen lain seperti *hardware, database,* dan lain-lain.

#### ***System and Software Design***

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka *software engineer* harus mengerti tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan, *user interface,* dan sebagainya. Dari dua aktivitas tersebut harus didokumentasikan kepada user.

Pada penelitian ini, pembuatan desain sistem ini menggunakan *Unified Modeling Language* (UML) yang dirancang dengan konsep *Object Oriented Design.* Pemodelan UML yang digunakan adalah sebagai berikut:

##### *Use Case Diagram*

*Use Case Diagram* adalah model yang menggambarkan aktifitas yang dilakukan oleh sistem dari sudut pandang luar. Diagram *use case* menggambarkan “apa” yang dilakukan sistem, bukan “bagaimana” sistem bekerja. Sebuah *use case* merepresentasikan sebuah interaksi antara *actor* dengan *system*. Diagram *use case* sangat membantu penyusunan *requirements* sebuah sistem, menerjemahkan rancangan kepada *client*, dan merancang *test case* untuk semua fitur yang ada pada sistem.

##### *Scenario Diagram*

*Scenario Diagram* berfungsi untuk menjelaskan alur sistem dari fitur yang ada di *job specification* dan *job description* yang ada pada diagram *use case. Scenario* menjelaskan alur sistem dan keadaan yang terjadi pada kejadiantertentu.

##### *Activity Diagram*

*Activity Diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing aktivitas berawal, keputusan yang mungkin terjadi, dan bagaimana aktivitas berakhir. *Activity Diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada eksekusi.

##### *Sequence Diagram*

*Sequence Diagram* menggambarkan aliran logika interaksi antar objek yang mengindikasikan komunikasi antar objek di dalam sistem yang disusun pada suatu urutan (*timeline).*

##### *Class Diagram*

*Class Diagram* menggambarkan struktur dan deskripsi *class* serta hubungan antar *class,* sehingga memudahkan dalam proses pengkodean.

##### *Entity Relationship Diagram (ERD)*

*ERD* merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan data yang mempunyai hubungan antar relasi.

#### ***Implementation and Unit Testing***

Pada tahap ini, desain program diterjemahkan ke dalam kode-kode program menjadi sebuah aplikasi web yang diimplementasikan dengan menggunakan bahasa pemrograman *Hypertext Prepocessor* (PHP) dan *framework Code Igniter (CI).*

#### ***Integration and System Testing***

Pada tahap ini, program diuji coba dengan menggunakan *white box* dan *black box*. *White box testing* merupakan cara pengujian dengan melihat ke dalam modul untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan luaran yang tidak sesuai dengan proses yang dilakukan, maka baris-baris program, variabel, dan parameter yang terlibat pada unit tersebut akan diperiksa satu persatu dan diperbaiki, kemudian di-*compile* ulang.

*Black box testing* adalah metode pengujian perangkat lunak yang memeriksa fungsionalitas dari aplikasi yang bertentangan dengan struktur internal atau kerja. Pengetahuan khusus dari kode aplikasi atau struktur internal dan pengetahuan pemrograman pada umumnya tidak diperlukan. Metode ini memfokuskan pada keperluan fungsional dari perangkat lunak. Ujicoba *black box* bukan merupakan alternatif dari ujicoba *white box*, tetapi merupakan pendekatan yang melengkapi untuk menemukan kesalahan lainnya, selain menggunakan metode *white box* (Rusadi, 2013).

#### ***Operation and Maintenance***

Pada tahap ini dilakukan pemeliharaan sistem untuk mengatasi *bug* atau sistem *error* yang tidak ditemukan sebelumnya, atau jika ada penambahan fitur baru.

## Fitur *Preview*

Seperti yang telah dijelaskan sebelumnya, bahwa *preview* adalah ulasan informasi tentang suatu produk yang ditawarkan dalam suatu penjualan. Untuk menghindari fenomena *“Overload Information”* dalam sistem penjualan tiket, fitur *preview* berperan untuk menyampaikan informasi yang jelas dan detail namun tidak berlebihan. Dalam hal ini penulis meneliti bagaimana hasil penjualan tiket dengan adanya penerapan fitur ini dalam penjualan *online* tiket event musik.

Untuk mengetahui hasil penjualan tiket dengan penerapan fitur *preview*, penulis merancang fitur sistem yang mampu mengetahui hasil penjualan tiket dengan beberapa kategori yang telah ditentukan yang dijadikan sebagai bahan untuk menghitung seberapa besar persentase konsumen yang membeli tiket dengan penerapan fitur *preview* yang mampu menjadi daya tarik dalam penjualan tiket *online* event musik.

Untuk perhitungan hasil penjualan, penulis menggunakan metode skala Likert. Skala Likert menurut Djaali (2008:28) adalah skala yang digunakan untuk mengukur persepsi, sikap, atau pendapat seseorang maupun kelompok terhadap suatu objek berdasarkan definisi operasional yang telah ditetapkan oleh peneliti.

Skala Likert adalah sebuah tipe skala psikometri yang menggunakan skala yang lebih luas dalam penelitian. Metode rating yang dijumlahkan (*summated rating*) popular juga dengan nama penskalaan model Likert. Metode Likert merupakan metode penskalaan pernyataan sikap yang menggunakan distribusi respon sebagai dasar penentuan nilai skalanya.

Untuk perhitungan hasil penjualan tiket dikategorikan sebagai berikut:

* Konsumen yang berkunjung, melihat *preview* event, membeli tiket dan memberikan komentar ataupun feedback (skor 5)
* Konsumen yang berkunjung, melihat *preview* event, dan membeli tiket (skor 4)
* Konsumen yang berkunjung dan membeli tiket (skor 3)
* Konsumen yang berkunjung dan melihat *preview* (skor 2)
* Konsumen yang hanya berkunjung saja (1)

Rumus:

S = n Pn

S = Nilai Skor

n = jumlah konsumen dari masing-masing kategori

Pn = Pilihan angka Skor Likert (1 – 5)

Dari perhitungan tersebut didapatkan total skor untuk masing-masing kategori. Kemudian dari masing-masing total skor tersebut dijumlahkan untuk mendapatkan jumlah total skor.

TS = S1 + S2 + S3 + S4 + S5

TS = Total Skor

Untuk mendapatkan hasil interpretasi, harus diketahui dahulu nilai tertinggi (Y) dan nilai terendah (X) dengan rumus sebagai berikut:

Nilai tertinggi Y = Pn (5) Tn

Nilai terendah X = Pn (1) Tn

Pn = Pilihan angka skor Likert

Tn = Total Responden

Jumlah nilai tertinggi adalah 5Tn Y. Jumlah nilai terendah adalah 1Tn X. Jika total skor yang didapat sebelumnya adalah (TS), maka interpretasi hasil penjualan adalah nilai yang dihasilkan dengan menggunakan rumus index % sebagai berikut:

INDEX % = 100

Sebelum menyelesaikannya, dibutuhkan kategori interval (jarak) untuk nilai interpretasi. Untuk rumus intervalnya sebagai berikut:

I =

I = = 20

Maka nilai 20 adalah interval dari terendah 0 % sampai dengan 100%. Untuk kriteria interpretasi index skor berdasarkarkan interval adalah sebagai berikut:

|  |  |
| --- | --- |
| **Persentase** | **Kriteria** |
| 0 % - 19,99 %  20 % - 39,99 %  40 % - 59,99 %  60 % - 79,99 %  80 % - 100 % | Sangat (tidak setuju / buruk / kurang sekali)  Tidak setuju / kurang  Cukup / Netral  Setuju (Baik / Suka)  Sangat (Setuju / Baik / Suka) |

## Subjek Penelitian

Perancangan sistem penjualan *online* tiket event musik dengan fitur *preview*, studi kasus ini ditujukan pada EO Side Story Family dan sejumlah responden untuk diajukan kuesioner dengan beberapa pertanyaan.

## Objek Penelitian

Sistem penjualan *online* tiket event musik dengan fitur *preview* dengan menggunakan bahasa pemrograman *PHP* dan pengolahan basis data *MySQL*.

## Instrumen Penelitian

* + - 1. Komputer

Komputer adalah perangkat keras atau hardware yang digunakan untuk menyusun penulisan laporan yang menggunakan *PC Desktop*.

* + - 1. *Software*

*Software* adalah program aplikasi komputer dengan menggunakan instruksi tertentu yang akan menjalankan suatu perintah. *Software* yang digunakan untuk bahasa pemrograman *PHP* adalah *notepad++*, untuk pengolahan basis data *MySQL* menggunaan aplikasi *XAMPP*.

* + - 1. Catatan Lapangan

Catatan lapangan diperlukan sebagai acuan dalam pengembangan rancangan. Catatan lapangan berisi data-data kelemahan dan kekurangan dari rancangan yang dibuat sehingga dengan adanya acuan ini dapat mengembangan rancangan yang ada.

* + - 1. *Camera Digital*

*Camera Digital* digunakan untuk pengambilan dokumentasi penelitian dan dokumentasi untuk sistem ini.

## Gambaran Umum Sistem

Sistem yang akan dirancang adalah sistem penjualan *online* tiket event musik dengan fitur *preview* yang ditujukan untuk:

* Memudahkan promosi event
* Memudahkan penjualan dan pembelian tiket event
* Memberikan informasi event untuk konsumen dengan fitur *preview*
* Menyediakan fasilitas kepada konsumen untuk memberikan *feedback* (umpan balik) ataupun ulasan terhadap event.
* Sebagai bentuk penelitian dan uji coba penulis untuk mengetahui bagaimana hasil penerapan dari penjualan tiket *online* dengan fitur *preview.*

Sistem ini dibangun menggunakan bahasa pemrograman *PHP* dan pengolahan basis data *MySQL.* Pada aplikasi ini terdapat tata cara penggunaan aplikasi dan *input* data baru. *User interface* dirancang untuk mudah dipahami dengan *size* aplikasi yang *fullscreen* ataupun *minimize* untuk kenyamanan penggunaan. Gambaran umum sistem dapat dilihat pada gambar berikut.

- Event

- Stock Tiket

- Dokumentasi

- Data User

- Komentar User

- Beranda

- Login

- Event

- Tiket

- About

- Admin EO Side Story Family

- User (member)

- User (guest)

- *Preview* Event

- Pesan Tiket

- Cetak Tiket

- Create, Update, Delete User

- *Review* User

Sistem Penjualan *Online* Tiket Event

Hak Akses

Fitur

Input

Menu

Output

- Tiket

- *Review* User

- Kode Pembayaran

- Data Penjualan Tiket

- Hasil Likert

Gambar 3.3 Gambaran Umum Sistem

# BAB 4. PENGEMBANGAN SISTEM

Bab ini menguraikan penjelasan mengenai analisis kebutuhan dan perancangan hingga tahap pengkodean dan pengujian aplikasi yang digunakan dalam proses pengembangan sistem penjualan tiket online event musik dengan menggunakan fitur *preview* pada EO Side Story Family.

## Deskripsi Umum Sistem

Deskripsi umum dari sistem yang dikembangkan dalam penelitian ini akan dijelaskan lebih detail pada SOP (*Statement Of Purpose*) sistem dan fungsi sistem.

### SOP (*Statement Of Purpose*)

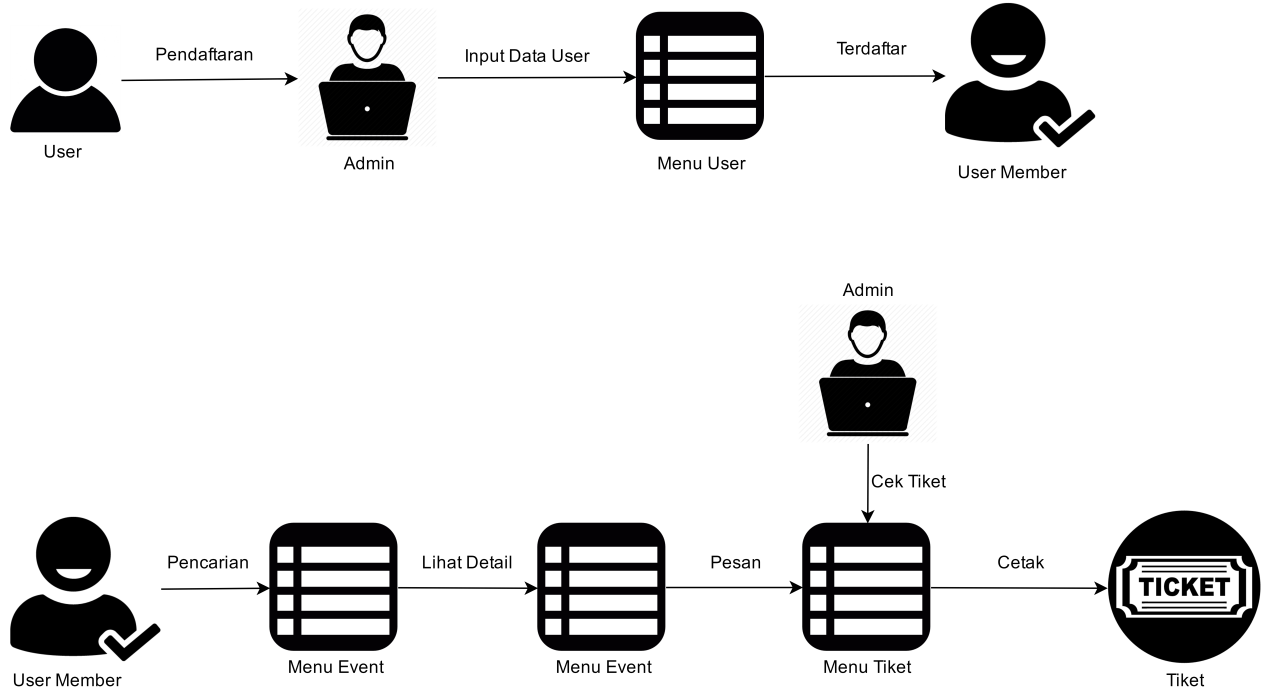
Pengembangan sistem sistem penjualan tiket diperlukan untuk menambah daya tarik dan promosi event. Upaya yang dilakukan ditujukan untuk menambah minat beli konsumen akan event yang diselenggarakan dan memudahkan akses informasi event dan pemesanan tiket event. Dari penelitian ini, penulis menerapkan pengembangan sistem penjualan tiket dengan fitur *preview.* Sistem penjualan tiket ini akan menampilkan *list* atau daftar event mulai dari yang sedang diselenggarakan, event yang akan datang maupun event yang telah selesai. Event yang telah selesai masih ditampilkan dalam daftar untuk di *review* oleh user sehingga mampu mengetahui bagaimana *feedback* dari konsumen, daftar event ini hanya ditampilkan dalam waktu tertentu.

### Fungsi Sistem

Sistem yang akan dibuat berupa sistem informasi penjualan tiket online event musik. Sistem ini merupakan sistem yang memberikan kemudahan dalam pemesanan tiket event musik. Pada sistem ini juga disediakan fitur review event yang bertujuan untuk mengetahui ulasan konsumen terhadap event tertentu.

### *Workflow*

*Workflow* adalah aliran kerja atau suatu informasi dari proses bisnis, baik secara keseluruhan maupun sebagian dimana dokumen atau informasi tugas tersebut diteruskan dari satu partisipan ke partisipan lain sesuai dengan prosedur atau ketentuan yang berlaku. *Workflow* sistem penjualan tiket *online* event musik dapat dilihat pada gambar 4.1.



Gambar 4.1 *Workflow* Sistem Penjualan Tiket *Online*

Gambar 4.1 merupakan *workflow* sistem penjualan tiket *online* event musik dengan fitur *preview* dimana proses input data pendaftaran member dilakukan oleh aktor admin. Setelah user terdaftar sebagai member, tahap selanjutnya adalah pencarian event musik sesuai dengan keinginan user member. Setelah menemukan event musik yang diinginkan, untuk memesan tiket event user member diharuskan untuk melihat informasi detail dari event tersebut. Tahap berikutnya setelah pemesanan tiket adalah pembayaran. User member yang sudah membayar dapat langsung mencetak tiket melalui sistem ini.

## Analisis Kebutuhan Perangkat Lunak

Berdasarkan metode fitur *preview,* analisis diperlukan untuk mengidentifikasi permasalahan yang ada kemudian dijadikan sebagai bahan untuk pengembangan sistem. Analisis kebutuhan yang dilakukan meliputi proses pengumpulan data kebutuhan fungsional dan kebutuhan non-fungsional sistem.

Kebutuhan fungsional sistem pada penelitian ini adalah sebagai berikut:

1. Sistem dapat melakukan CRUD (*Create, Review, Update, Delete*) Event
2. Sistem dapat memesan Tiket
3. Sistem dapat mencetak Tiket
4. Sistem dapat mereview Event
5. Sistem dapat melakukan CRUD (*Create, Read, Update, Delete*) User

Kebutuhan non-fungsional sistem pada penelitian ini adalah sebagai berikut:

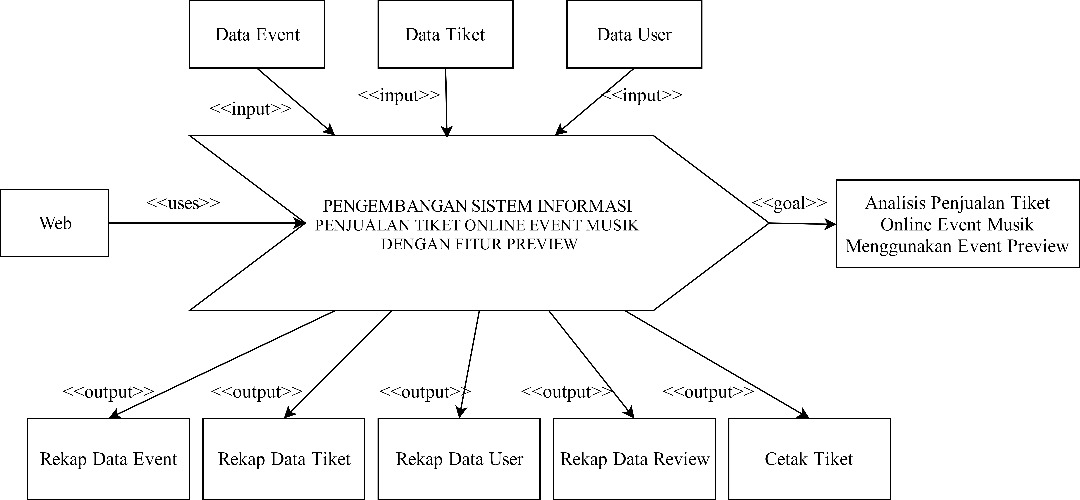
* 1. Sistem dapat mendukung proses login pengguna berdasarkan hak akses yang sudah ditentukan. Hak akses terbagi menjadi dua yaitu *user* dan admin.
  2. Proses enkripsi untuk *password* *member* dan admin sebagai syarat autentifikasi keamanan.
  3. Sistem dapat menampilkan hasil perhitungan dari penerapan fitur *preview* pada penjualan tiket *online* dengan metode skala Likert.

## Desain Sistem

Desain sistem adalah tahapan yang digunakan untuk memodelkan sistem yang akan dibuat. Desain sistem yang akan dibuat meliputi *usecase diagram, scenario, sequence diagram, activity diagram, class diagram* dan *Entity* *Relatioship* *Diagram* (ERD).

### *Business Process*

*Business Process* digunakan untuk menggambarkan sebuah proses suatu aplikasi secara keseluruhan, lengkap dengan *resources* dan informasi yang dibutuhkan seperti *input* dan *output* aplikasi, sehingga mendorong terjadinya proses dan tujuan yang telah ditentukan, seperti pada gambar 4.2.



Gambar 4.2 *Business Process* Sistem Penjualan Tiket *Online*



### *Usecase Diagram*

*Usecase diagram* digunakan untuk menggambarkan fitur yang dibuat dalam sistem dan juga menggambarkan siapa saja yang dapat menggunakan fitur tersebut. *Usecase diagram* sistem penjualan tiket *online* dapat dilihat pada Gambar 4.3.



Gambar 4.3 *Usecase Diagram* Sistem Penjualan Tiket *Online*

Pada *usecase* tersebut terdapat dua aktor. Definisi Aktor mendeskripsikan *user* yang terlibat pada sistem. Definisi aktor dapat dilihat pada tabel 4.1 dan definisi *usecase* dapat dilihat pada tabel 4.2 di bawah ini.

Tabel 4.1 Definisi Aktor *Usecase*

|  |  |  |
| --- | --- | --- |
| No | Aktor | Definisi Tugas |
| 1. | Admin | Aktor ini dapat menggunakan semua fitur yang ada pada sistem. |
| 2.  3. | *User*  *Guest* | Aktor ini dapat menggunakan fitur yang ada pada sistem antara lain: melihat, preview, memesan, mencetak, dan mereview event serta mengubah data *user*.  Aktor ini hanya dapat melihat event yang ada dalam sistem. |

Tabel 4.2 Definisi *Usecase*

|  |  |  |
| --- | --- | --- |
| No | Usecase | Deskripsi |
|  | Mengelola event | * + - Merupakan *usecase* untuk mengelola event. |
|  | Melihat event | * + - Merupakan *usecase* untuk melihat event. |
|  | Preview Event | * + - Merupakan *usecase* untuk melihat detail dari event. |
|  | Memesan tiket | * + - Merupakan *usecase* untuk memesan tiket*.* |
|  | Mencetak tiket | * + - Merupakan *usecase* untuk mencetak tiket*.* |
|  | Mereview event | * + - Merupakan *usecase* untuk mereview (memberikan komentar, penilaian, saran) event*.* |
|  | Registrasi user | * + - Merupakan *usecase* untuk membuat data user baru |
|  | Mengelola data user | * + - Merupakan *usecase* untuk mengelola data user |
|  | Mengubah data user | * + - Merupakan *usecase* untuk mengubah username dan password user |
|  | Melihat hasil likert | * + - Merupakan *usecase* untuk melihat hasil *likert* |

### *Scenario Diagram*

*Scenario Diagram* digunakan untuk menjelaskan atau menceritakan fitur atau isi yang ada di *usecase diagram*. *Scenario* menjelaskan alur sistem dan keadaan yang akan terjadi ketika terjadi suatu event tertentu. *Scenario* pada pengembangan sistem penjualan tiket *online* dapat dilihat pada tabel 4.3.3 dibawah ini.

Tabel 4.3.3 Tabel *Scenario* Preview Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-03 | |
| Name *usecase* | Preview event | |
| Aktor | Admin, *user* | |
| Deskripsi | Merupakan *usecase* untuk melihat detail event. | |
| Relasi | - | |
| Kondisi awal | Sistem menampilkan daftar event | |
| Kondisi akhir | Sistem menampilkan preview/detail event | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin / *user* memilih pilihan event 2. Admin/*user* memilih pilihan melihat detail event | 1. Sistem menampilkan daftar event yang tersedia 2. Sistem menampilkan detail event yang dipilih |
| Alur alternatif | - | |
| Kebutuhan khusus | - | |

Tabel 4.3.9 Tabel *Scenario* Melihat Hasil Likert

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-009 | |
| Name *usecase* | Melihat hasil *likert* | |
| Aktor | Admin | |
| Deskripsi | Merupakan *usecase* untuk melihat hasil *likert* | |
| Relasi | - | |
| Kondisi awal | Sistem menghitung likert sesuai dengan kategori untuk user dan guest | |
| Kondisi akhir | Sistem menampilkan hasil perhitungan likert | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin memilih pilihan melihat hasil *likert* | 1. Sistem menampilkan hasil *likert* |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus |  | |

### *Activity Diagram*

*Activity diagram* menggambarkan aktivitas aktor dan sistem yang saling berhubungan dalam suatu aktivitas atau event. *Activity diagram* menggambarkan berbagai alur aktivitas dalam sistem yang dirancang berawal decision. *Activity diagram* harus sesuai dengan *scenario* sistem yang telah dirancang. Sistem memberikan respon pada aktivitas yang dilakukan aktor. Berikut adalah penjelasan *activity diagram* dari sistem yang dirancang.

#### ***Activity Diagram* Melihat Event**

*Activity diagram* melihat event menggambarkan alur aktivitas padafitur melihat event. Fitur melihat event memiliki aktivitas utama, yaitu menampilkan daftar event yang tersedia. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.5.



Gambar 4.5 *Activity Diagram* Melihat Event

#### ***Activity Diagram* Melihat Hasil Likert**

*Activity diagram* melihat hasil *likert* menggambarkan alur aktivitas padafitur melihat hasil *likert*. Fitur melihat hasil *likert* memiliki aktivitas utama, yaitu menampilkan hasil *likert*. Detail aktifitas dalam melihat event ditunjukan pada gambar 5.1.

Gambar 5.1 *Activity Diagram* Melihat Hasil Likert

#### ***Activity Diagram* Mengelola Event**

*Activity diagram* mengelola event menggambarkan alur aktivitas padafitur pengelolaan event. Fitur melihat pengelolaan event memiliki aktivitas utama, yaitu menampilkan daftar event, menambah event, menghapus event, dan memperbaharui event yang tersedia. Aktifitas dalam melihat pengelolaan event ditunjukkan pada pada lampiran gambar 4.4.

#### ***Activity Diagram* Memesan Tiket**

*Activity diagram* memesan tiket menggambarkan alur aktivitas padafitur memesan tiket. Fitur memesan tiket memiliki aktivitas utama, yaitu menampilkan daftar event yang tersedia dan mengolah pemesan tiket event. Detail aktifitas dalam melihat event ditunjukan pada lampiran gambar 4.6.

#### ***Activity Diagram* Mencetak Tiket**

*Activity diagram* mencetak tiket menggambarkan alur aktivitas padafitur mencetak tiket. Fitur mencetak tiket memiliki aktivitas utama, yaitu menampilkan tiket yang telah dipesan dan mengubah form pesanan tiket ke dalam format pdf. Detail aktifitas dalam melihat event ditunjukan pada lampiran gambar 4.7.

#### ***Activity Diagram* Mereview Event**

*Activity diagram* mereview event menggambarkan alur aktivitas padafitur review event. Fitur review event memiliki aktivitas utama, yaitu menampilkan form review event dan menampilkannya. Detail aktifitas dalam melihat event ditunjukan pada lampiran gambar 4.8.

#### ***Activity Diagram* Mengelola Data User**

*Activity diagram* mengelola data user menggambarkan alur aktivitas padafitur data user. Fitur mengelola data user memiliki aktivitas utama, yaitu menambah data user dan menghapus data user. Detail aktifitas dalam melihat event ditunjukan pada lampiran gambar 4.9.

#### ***Activity Diagram* Mengubah Data User**

*Activity diagram* mengubah data user menggambarkan alur aktivitas padafitur update data user. Fitur mengubah data user memiliki aktivitas utama, yaitu menampilkan form update data user dan memperbaharui data user. Detail aktifitas dalam melihat event ditunjukan pada lampiran gambar 5.0.

### *Sequence Diagram*

*Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Berikut adalah penjelasan *sequence diagram* dari sistem yang dirancang.

#### ***Sequence Diagram* Melihat Event**

*Sequence diagram* mengelola event digunakan untuk menunjukkan interaksi antar objek pada fitur melihat event. Objek yang terlibat pada fitur melihat event dengan *admin* sebagai aktor, antara lain *lihat* (UIEvent), *controller* (EventController), *model* (Event). Detail *sequence diagram* melihat event ditunjukan pada gambar 5.3.

 Gambar 5.3 *Sequence Diagram* Melihat Event

#### ***Sequence Diagram* Mengelola Event**

*Sequence diagram* mengelola event digunakan untuk menunjukkan interaksi antar objek pada fitur pengelolaan event. Objek yang terlibat pada fitur mengelola event dengan *admin* sebagai aktor, antara lain *tampil (*UIKelolaEvent; FormTambahEvent; FormPerbaharuiEvent), *controller (*EventController), *model* (Event). Detail *sequence diagram* mengelola event ditunjukan pada lampiran gambar 5.2.

#### ***Sequence Diagram* Memesan Tiket**

*Sequence diagram* memesan tiket digunakan untuk menunjukkan interaksi antar objek pada fitur memesan tiket. Objek yang terlibat pada fitur memesan tiket dengan *admin* sebagai aktor, antara lain *tampil* (UIHome; UIEvent; UIPemesananTiket), *controller* (EventController; TiketController), *model* (Event; Tiket). Detail *sequence diagram* memesan tiket ditunjukan pada lampiran gambar 5.4.

#### ***Sequence Diagram* Mencetak Tiket**

*Sequence diagram* mencetak tiket digunakan untuk menunjukkan interaksi antar objek pada fitur cetak tiket. Objek yang terlibat pada fitur cetak tiket dengan *admin* sebagai aktor, antara lain *tampil* (UIListEvent), *controller* (TiketController), *model* (Tiket). Detail *sequence diagram* mencetak tiket ditunjukan pada lampiran gambar 5.5.

#### ***Sequence Diagram* Mereview Event**

*Sequence diagram* mereview event digunakan untuk menunjukkan interaksi antar objek pada fitur review event. Objek yang terlibat pada fitur mereview event dengan *admin* sebagai aktor, antara lain *tampil* (UIHome; UIEvent; UIPemesananTiket), *controller* (EventController; TiketController), *model* (Event; Review). Detail *sequence diagram* mereview event ditunjukan pada lampiran gambar 5.6.

#### ***Sequence Diagram* Mengelola Data User**

*Sequence diagram* mengelola data user digunakan untuk menunjukkan interaksi antar objek pada fitur pengelolaan data user. Objek yang terlibat pada fitur mengelola data user dengan *admin* sebagai aktor, antara lain *tampil* (UIKelolaUser; FormTambahUser; FormPerbaharuiUser), *controller* (UserController), *model* (User). Detail *sequence diagram* mengelola data user ditunjukan pada lampiran gambar 5.7.

#### ***Sequence Diagram* Mengubah Data User**

*Sequence diagram* mengubah data user digunakan untuk menunjukkan interaksi antar objek pada fitur mengubah data user. Objek yang terlibat pada fitur mengubah data user dengan *admin* sebagai aktor, antara lain *tampil* (FormPerbaharuiUser), *controller* (UserController), *model* (User). Detail *sequence diagram* mengubah data user ditunjukan pada lampiran gambar 5.8.

#### **4.3.4.8 Sequence Diagram Melihat Hasil Likert**

*Sequence diagram* melihat hasil *likert* digunakan untuk menampilkan hasil likert. Objek yang terlibat pada fitur melihat hasil *likert* dengan *admin* sebagai aktor, antara lain *tampil* (*UILikert*), *controller* (*LikertController*), *model* (User). Detail *sequence diagram* mengubah data user ditunjukan pada lampiran gambar 5.9.

### *Class Diagram*

Setelah melalui tahap pembuatan desain dengan *sequence diagram*, tahap selanjutnya membuat desain perancangan *class diagram*. *Class diagram* berisi method dan data yang berbeda namun memiliki hubungan dengan yang lainnya. *Class Diagram* lebih lengkapnya dapat dilihat pada gambar 5.9.

 Gambar 5.9 *Class Diagram* Sistem Tiket Online Side Story Family

### *Entity Relationship Diagram*

Setelah pembuatan *class diagram,* tahap perancangan selanjutnya adalah membuat desain *database*. Desain ini berisi basis data yang akan digunakan oleh sistem. *Entity Relationship Diagram* lebih lengkapnya dapat dilihat pada gambar 6.0.



Gambar 6.0 *Entity Relationship Diagram* Sistem Tiket Online Side Story Family

## Implementasi

Pada tahap ini, desain selanjutnya akan diterjemahkan kedalam bentuk *coding* atau pengkodean menggunakan bahasa pemrograman yaitu *Page Hyper Text Pre-Process*(PHP), *Hyper Text Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan *Javascript.* Manajemen data menggunakan *database Mysql*. Dalam perancangan menggunakan *framework Yii2* untuk memudahkan di dalam pengembangan dan penulisan *coding* di dalam sebuah pemrograman. Proses pengkodean dalam membangun sistem ini lebih lanjut akan dijelaskan pada lampiran B. Proses pengkodean dalam membangun Sistem meliputi fitur login, logout, event, tiket dan user dapat dilihat pada lampiran A.

## Pengujian Sistem

Tahapan pengujian aplikasi merupakan suatu tahapan yang dilakukan secara sistematis untuk menguji dan mengevaluasi sistem dengan menggunakan sebuah metode pengujian sistem. Hal tersebut dilakukan dengan tujuan untuk mengevaluasi apakah kebutuhan sistem telah terpenuhi dan sistem layak untuk digunakan oleh pengguna. Agar pengujian yang dilakukan lebih valid, maka tahap pengujian sistem ini dilakukan dengan menggunakan dua metode, yaitu *white box* dan *black box*.

### Metode White-box

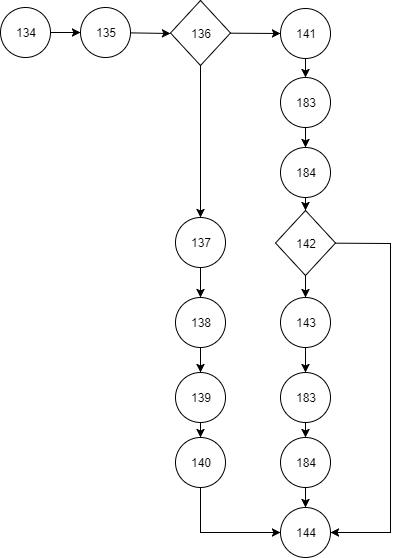
Pengujian sistem dengan metode *white box* dilakukan untuk menguji sistem dari segi desain dan kode program. Hal tersebut bertujuan untuk mengevaluasi apakah sistem mampu menghasilkan fungsi-fungsi, input, dan output yang sesuai dengan spesifikasi dari kebutuhan sistem itu sendiri. Pengujian dengan metode *white box* dilakukan oleh penulis dengan cara menghitung *independent path* yaitu dengan menggunakan suatu pengukuran kuantitatif *cyclomatic complexity*, *listing program*, penentuan jalur independen, dan *test case*. Tahapan-tahapan pengujian dengan metode *white box* ini akan diterapkan pada fitur yang dinilai dapat mewakili sistem ini. Tahapan pengujian jalur dasar meliputi:

* 1. *Listing Programe*

*Listing programe* merupakan baris-baris kode yang nantinya akan diuji. Setiap langkah dari kode-kode yang ada diberi nomor baik menjalankan *statement* biasa atau penggunaan kondisi dalam program.

* 1. Diagram Alir

Diagram alir merupakan notasi sederhana yang digunakan untuk merepresentasikan aliran kontrol. Aliran kontrol yang digambarkan merupakan hasil penomoran dari *listing* program*.* Diagram alir digambarkan dengan *node-node* (simpul) yang dihubungkan dengan *edge-edge* (garis) yang menggambarkan alur jalannya program. Diagram alir fitur *preview* dapat dilihat pada gambar 6.1.



Gambar 6.1 Diagram Alir Fitur *Preview*

* 1. Kompleksitas Siklomatik

Kompleksitas siklomatik merupakan metrik perangkat lunak yang menyediakan ukuran kuantitatif dari kompleksitas logis suatu program. Bila digunakan dalam konteks teknik pengujian jalur dasar, nilai yang dihitung untuk kompleksitas siklomatik mendefinisikan jumlah-jumlah jalur independen dalam basis set suatu program. Kompleksitas siklomatik fitur *preview* berdasarkan diagram alir adalah sebagai berikut:

CC = EDGE-NODE+2

CC = 16-15+2

CC = 3

* 1. Jalur Program Independen

Jalur independen adalah jalur yang melalui program yang memperkenalkan setidaknya satu kumpulan pernyataan-pernyataan pemrosesan atau kondisi baru. Bila dinyatakan dalam grafik alir, jalur independen harus bergerak setidaknya sepanjang satu *edge* yang belum dilintasi sebelum jalur tersebut didefinisi. Jalur program independen fitur *preview* sebagai berikut:

Jalur 1 = 134-135-136-137-138-139-140-145

Jalur 2 = 134-135-136-141-183-184-142-143-183-184-144

Jalur 3 = 134-135-136-141-183-184-142-144

* 1. Pengujian Basis Set

Pada pengujian basis set, bagian ini diberikan contoh data yang akan memaksa pelaksanaan jalur di basis set. Data yang dieksekusi kemudian dimasukkan ke dalam grafik alir dan diuji apakah sudah melewati basis set yang tersedia. Pengujian sistem telah memenuhi syarat kelayakan perangkat lunak jika salah satu jalur yang diuji telah dieksekusi setidaknya satu kali. Pengujian basis set fitur *preview* dapat dilihat pada tabel 4.3.

Tabel 4.3 *Test Case* Fitur *Preview*

|  |  |
| --- | --- |
| Jalur 1 | |
| *Test Case* | Jika user role adalah admin, maka menampilkan daftar event beserta preview event |
| Target yang diharapkan | Menampilkan daftar event beserta preview event |
| HasilPengujian | Benar |
| Path/Jalur | 134-135-136-137-138-139-140-145 |
| Jalur 2 | |
| *Test Case* | Jika user role adalah user, maka menampilkan daftar event, preview event, melakukan perhitungan likert. |
| Target yang diharapkan | Menampilkan daftar event, preview event, melakukan perhitungan likert. |
| HasilPengujian | Benar |
| Path/Jalur | 134-135-136-141-183-184-142-143-183-184-144 |
| Jalur 3 | |
| *Test Case* | Jika user role adalah guest, maka menampilkan daftar event, melakukan perhitungan likert. |
| Target yang diharapkan | Menampilkan daftar event dan melakukan perhitungan likert. |
| HasilPengujian | Benar |
| Path/Jalur | 134-135-136-141-183-184-142-144 |

### Metode Black-box

Pengujian *black box* berfungsi untuk menguji aplikasi dari segi spesifikasi fungsional sistem dengan tujuan mengetahui apakah fungsi-fungsi, inputan, dan keluaran sistem sesuai dengan spesifikasi yang dibutuhkan oleh pengguna. Pengujian dengan metode *black box* dilakukan oleh EO Side Story Family. Hasil pengujian dengan metode *black box* dapat dilihat pada lampiran B.

# BAB 5. HASIL DAN PEMBAHASAN

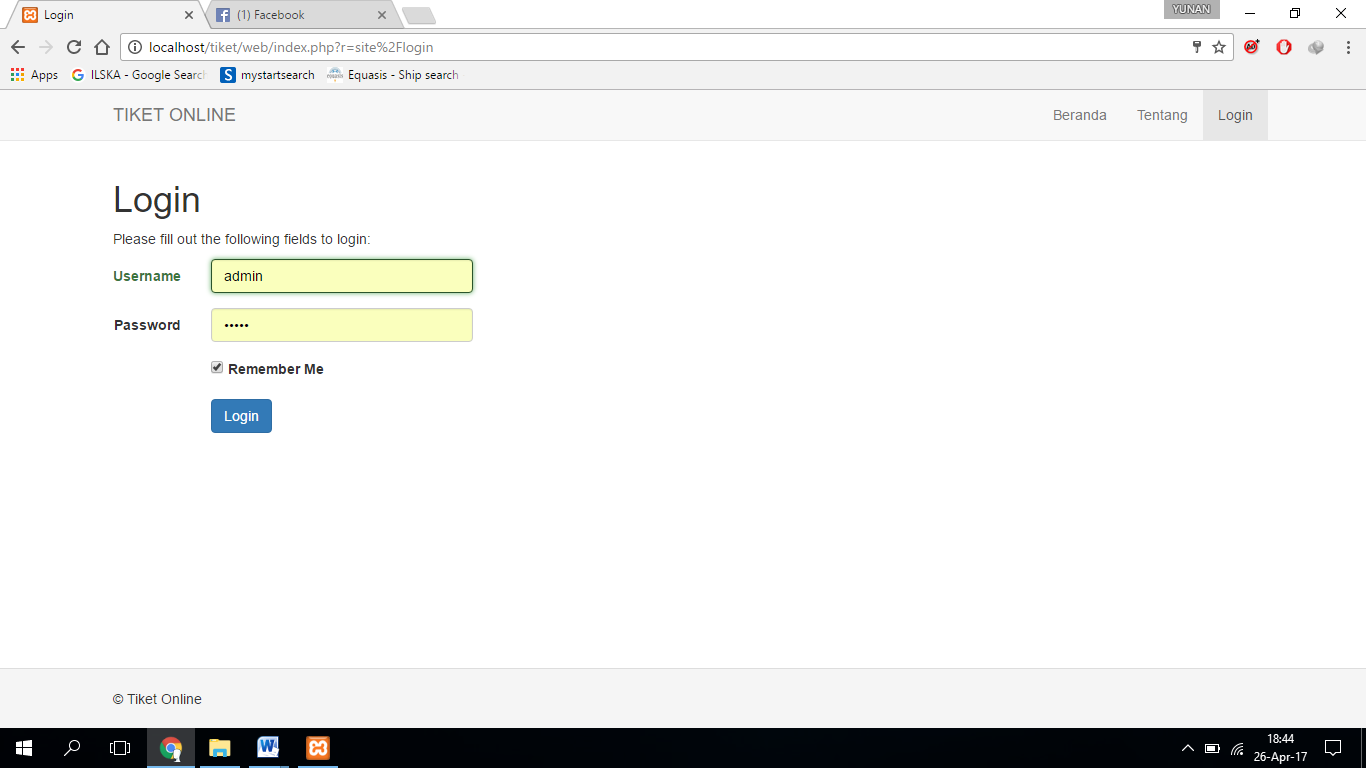
Bab ini memaparkan hasil dan pembahasan sistem informasi penjualan tiket online event musik dengan fitur *preview* yang meliputi implementasi sistem informasi penjualan tiket *online* event musik dan hasil implementasi fitur *preview* pada sistem informasi penjualan tiket *online* event musik.

## Hasil Implementasi Kode Program pada Sistem Informasi Penjualan Tiket *Online* Event Musik

Penelitian ini menghasilkan sebuah sistem informasi penjualan tiket online event musik beserta fitur-fiturnya. Pada sistem ini terdapat tiga hak akses, yaitu *admin* yang dapat mengakses fitur mengelola event, melihat event, memesan tiket, mereview event, dan mengelola data user, kemudian *user* yang dapat mengakses fitur melihat event, memesan tiket, mencetak tiket, mereview event, mengubah data user yang bersangkutan dan yang terakhir adalah *guest* yang hanya dapat melihat sistem. Penjelasan mengenai fitur-fitur yang terdapat dalam sistem informasi penjualan tiket online event musik ini adalah sebagai berikut.

### Fitur Login

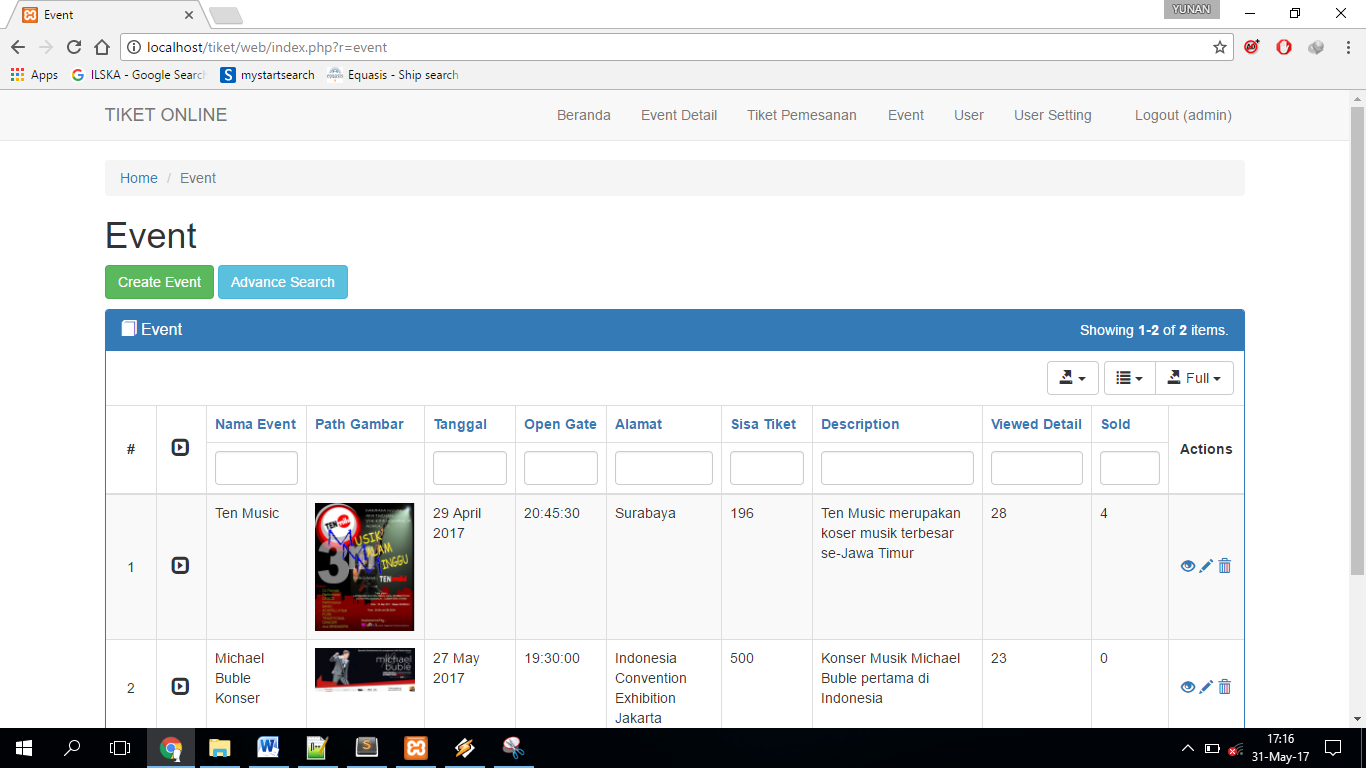
Untuk dapat mengakses fitur-fitur yang terdapat dalam sistem informasi penjualan tiket online event musik ini, setiap pengguna diharuskan *login* dengan memasukkan *username* dan *password*. Setelah *login*, sistem akan menampilkan halaman *Home* dan fitur-fitur yang ada dapat diakses berdasarkan level usernya.Dapat dilihat pada gambar 5.1.



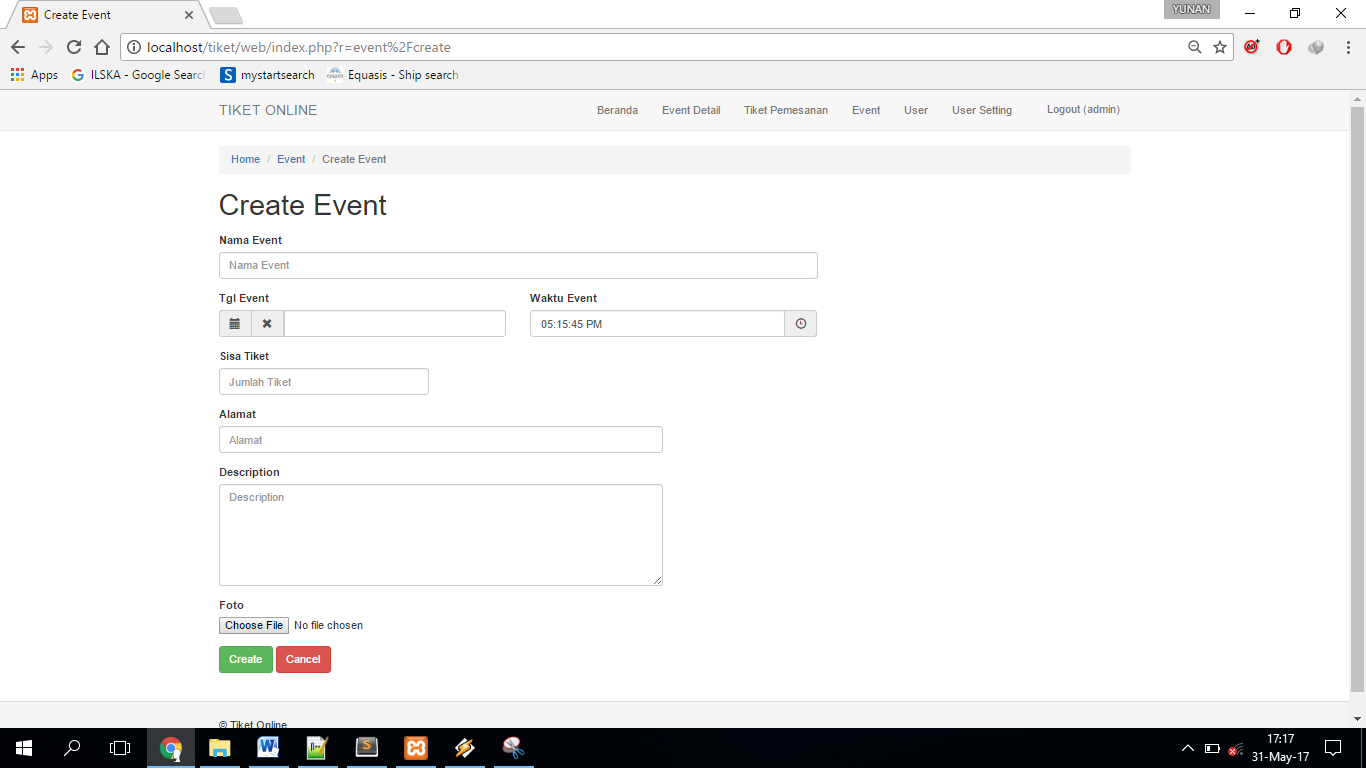
Gambar 5.1 Fitur Login

### Fitur Mengelola Event

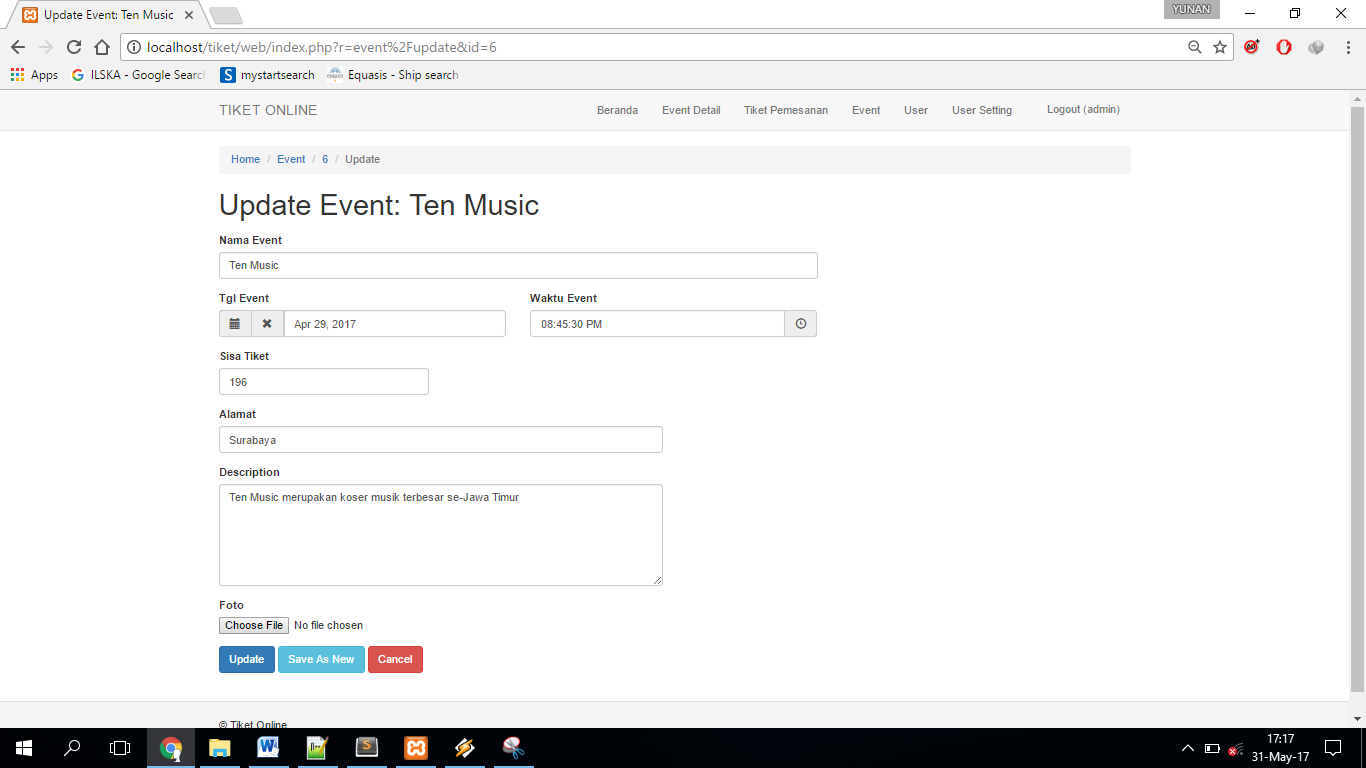
Pada fitur ini menunjukkan tampilan sistem dalam mengelola (menambah, memperbaharui, dan menghapus) data event, dapat dilihat pada gambar 5.2. Fitur ini dapat diakses hanya oleh *admin*. Fitur tambah data event merupakan fitur untuk menambah data event pada tampilan sistem form tambah data event dapat dilihat pada gambar 5.3. Fitur ubah data event merupakan fitur untuk memperbaharui data event pada tampilan sistem form ubah data event dapat dilihat pada gambar 5.4. Fitur hapus data event merupakan fitur untuk menghapus data event pada tampilan mengelola data event.



Gambar 5.2 Fitur Mengelola Data Event



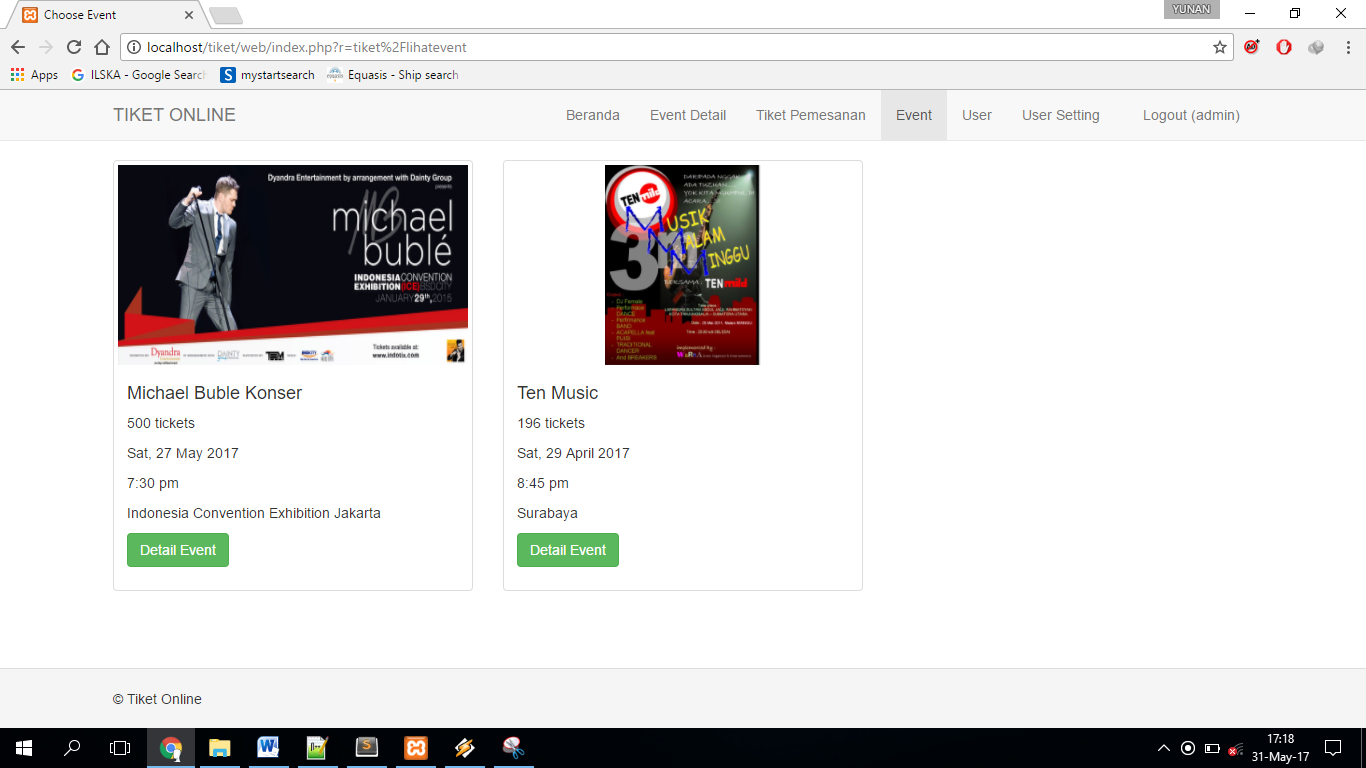
Gambar 5.3 Fitur Tambah Data Event



Gambar 5.4 Fitur Ubah Data Event

### Fitur Melihat Event

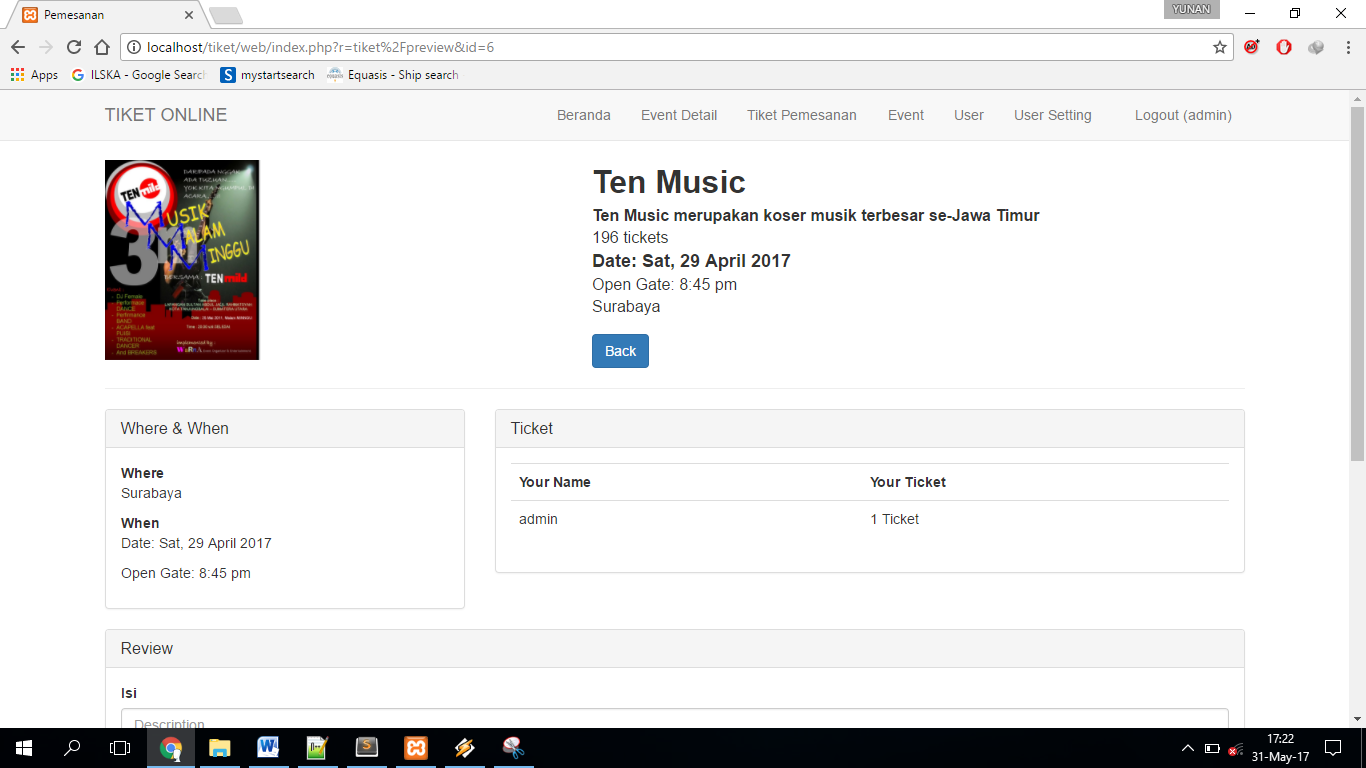
Fitur ini merupakan fitur yang digunakan untuk melihat detail dari suatu event. *User* atau *guest* yang ingin membeli tiket event harus masuk ke dalam tampilan detail dari event tersebut. Dapat dilihat pada gambar 5.5.



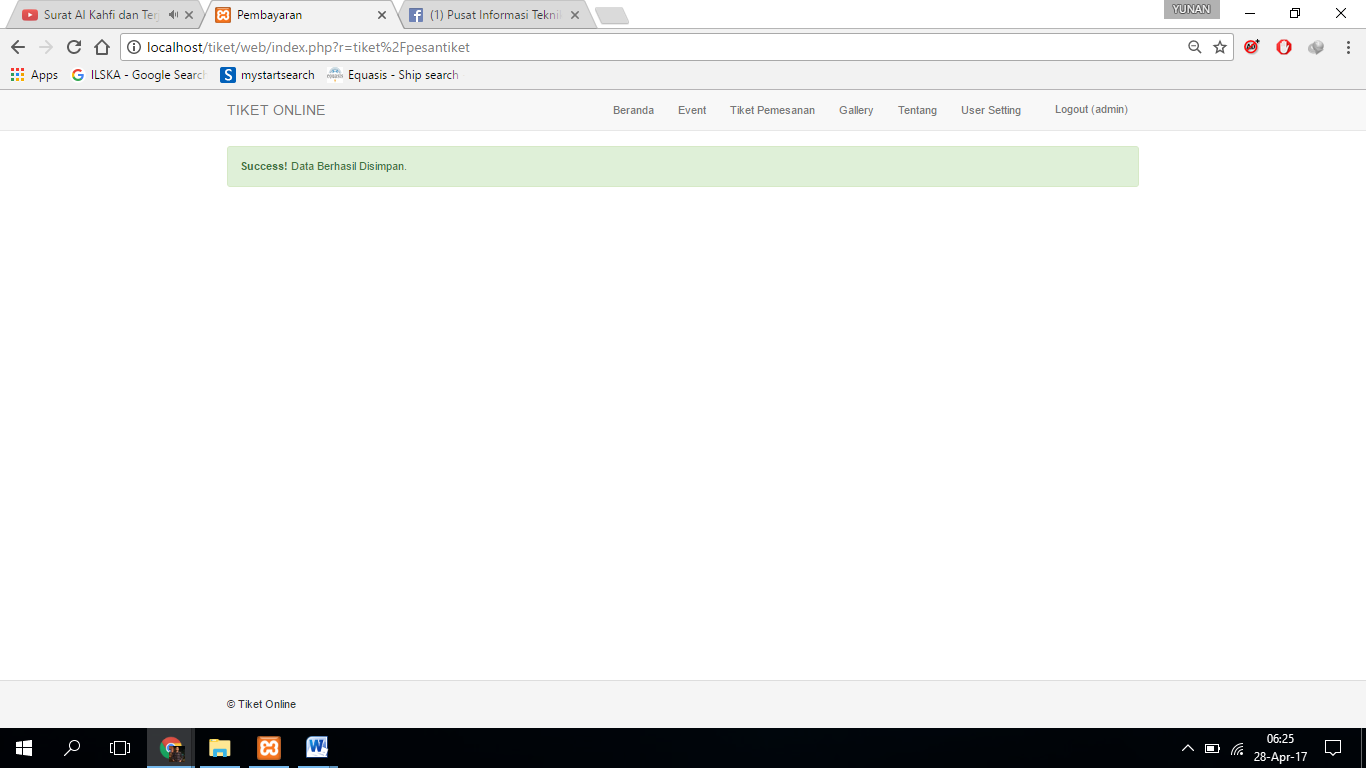
Gambar 5.5 Fitur Melihat List Event

### Fitur Memesan Tiket

Pada fitur ini menunjukkan tampilan sistem berhasil untuk memesan tiket. Fitur ini merupakan fitur inti dari sistem informasi penjualan tiket online event musik. Tampilan fitur menu pesan tiket dapat dilihat pada gambar 5.6 dan 5.7.

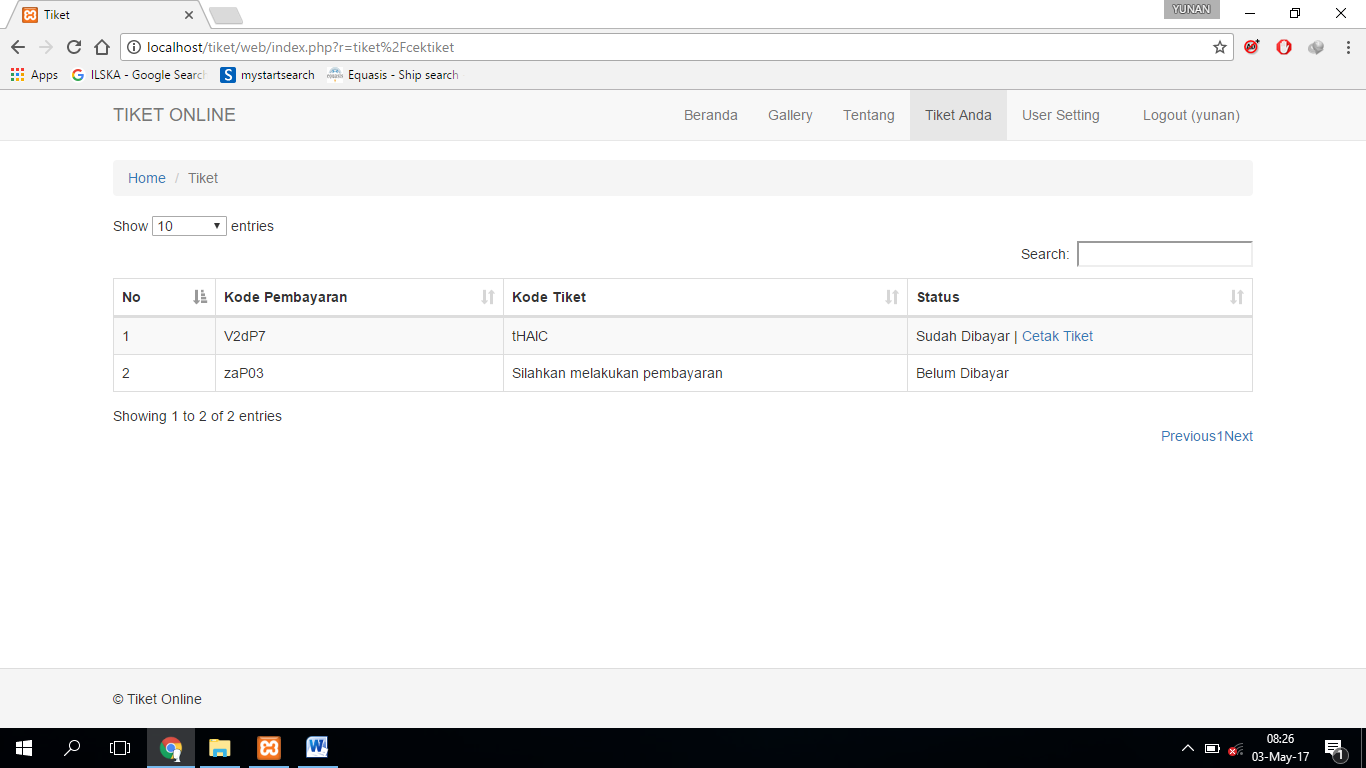


Gambar 5.7 Gambar menu pesan tiket



Gambar 5.8 Gambar berhasil memesan tiket

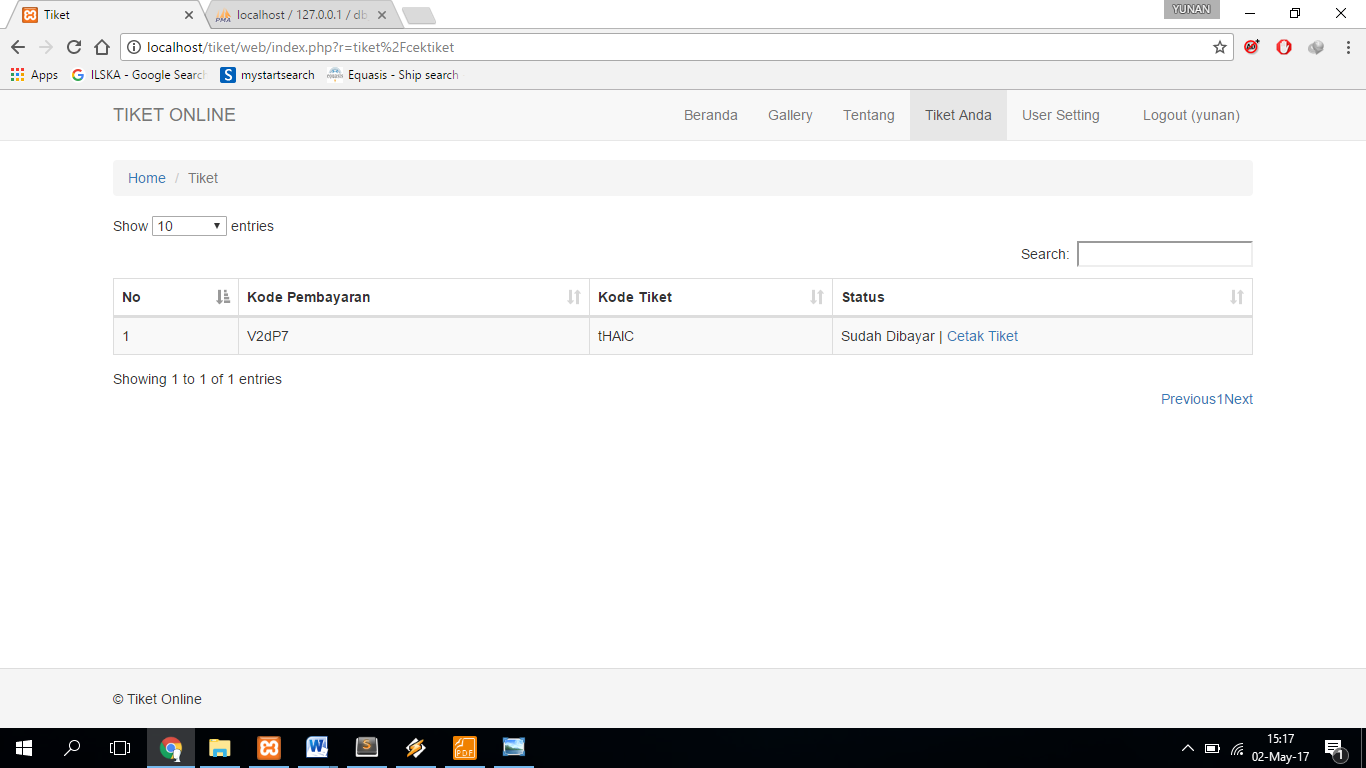
Setelah berhasil memesan tiket, user akan mendapatkan kode pembayaran yang digunakan untuk melakukan pembayaran tiket melalui transfer. Kode pembayaran secara otomatis akan di-*generate* oleh sistem. Kode pembayaran tiket terdapat pada menu Tiket Anda, seperti pada gambar 5.9.



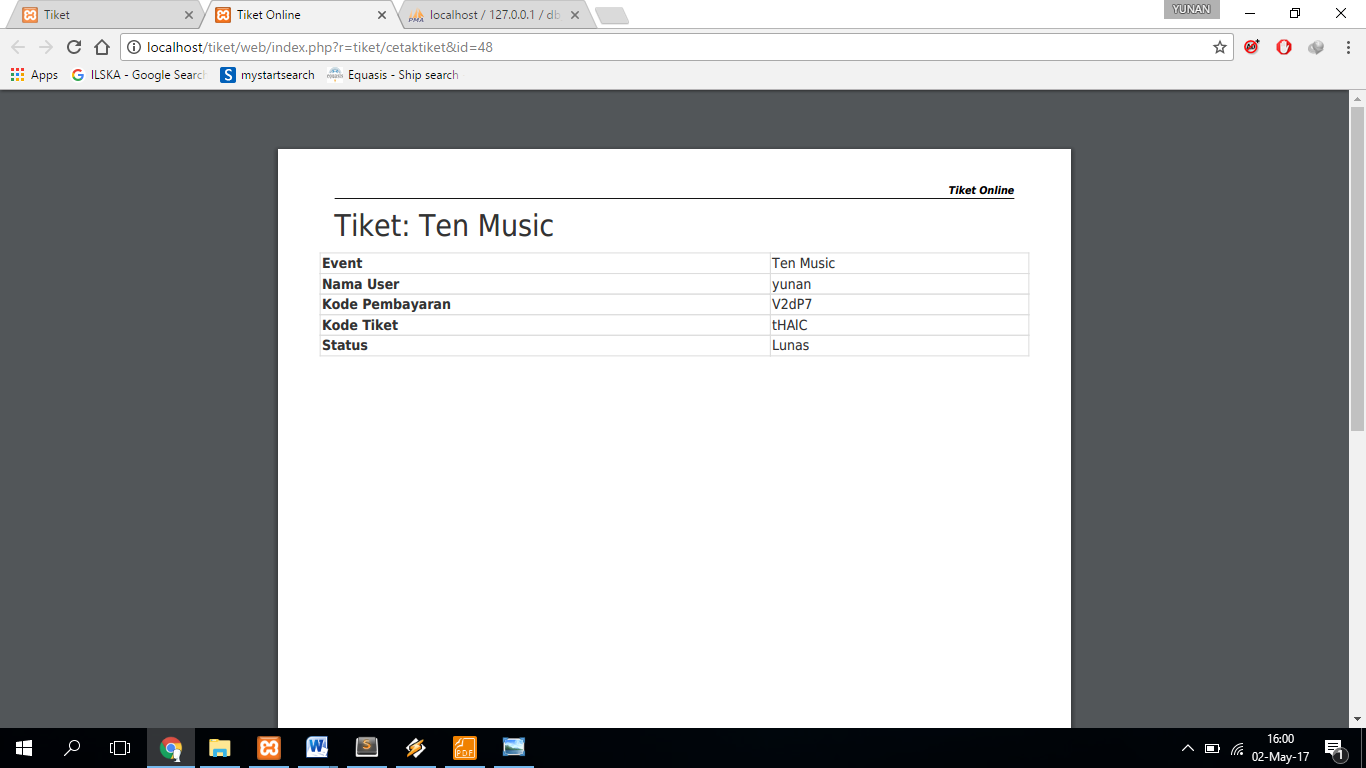
Gambar 5.9 Hasil generate kode pembayaran

### Fitur Mencetak Tiket

Fitur ini merupakan fitur yang digunakan untuk mengubah tiket ke bentuk file dengan format pdf. Fitur ini berfungsi ketika user sudah melakukan pelunasan pada tiket yang dipesan. Tampilan fitur ini dapat dilihat pada gambar 6.0 dan gambar 6.1. Gambar 6.0 merupakan tampilan untuk cetak tiket ketika tiket tersebut sudah terbayar. Gambar 6.1 merupakan tampilan file berformat pdf dari tiket yang akan dicetak.



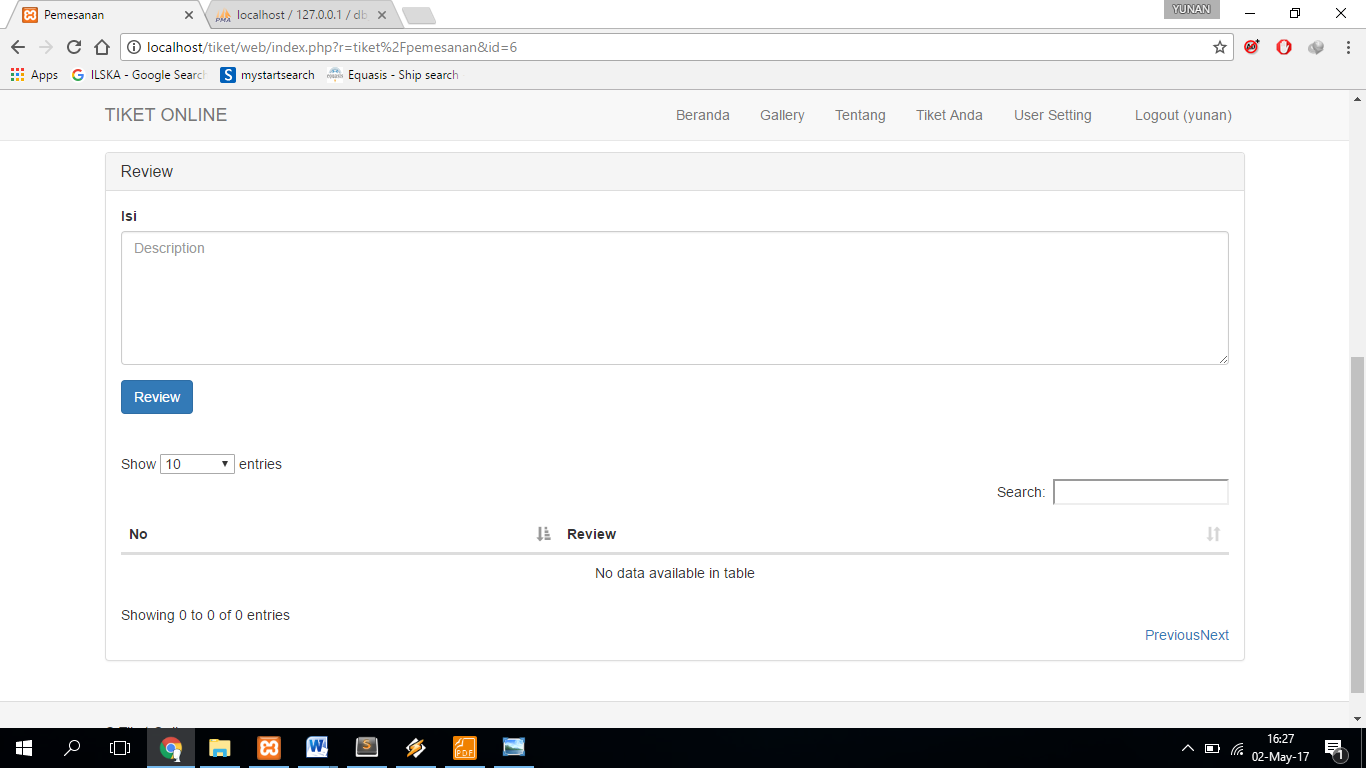
Gambar 6.0 Tampilan fitur cetak tiket



Gambar 6.1 Tampilan tiket dalam bentuk pdf

### Fitur Mereview Event

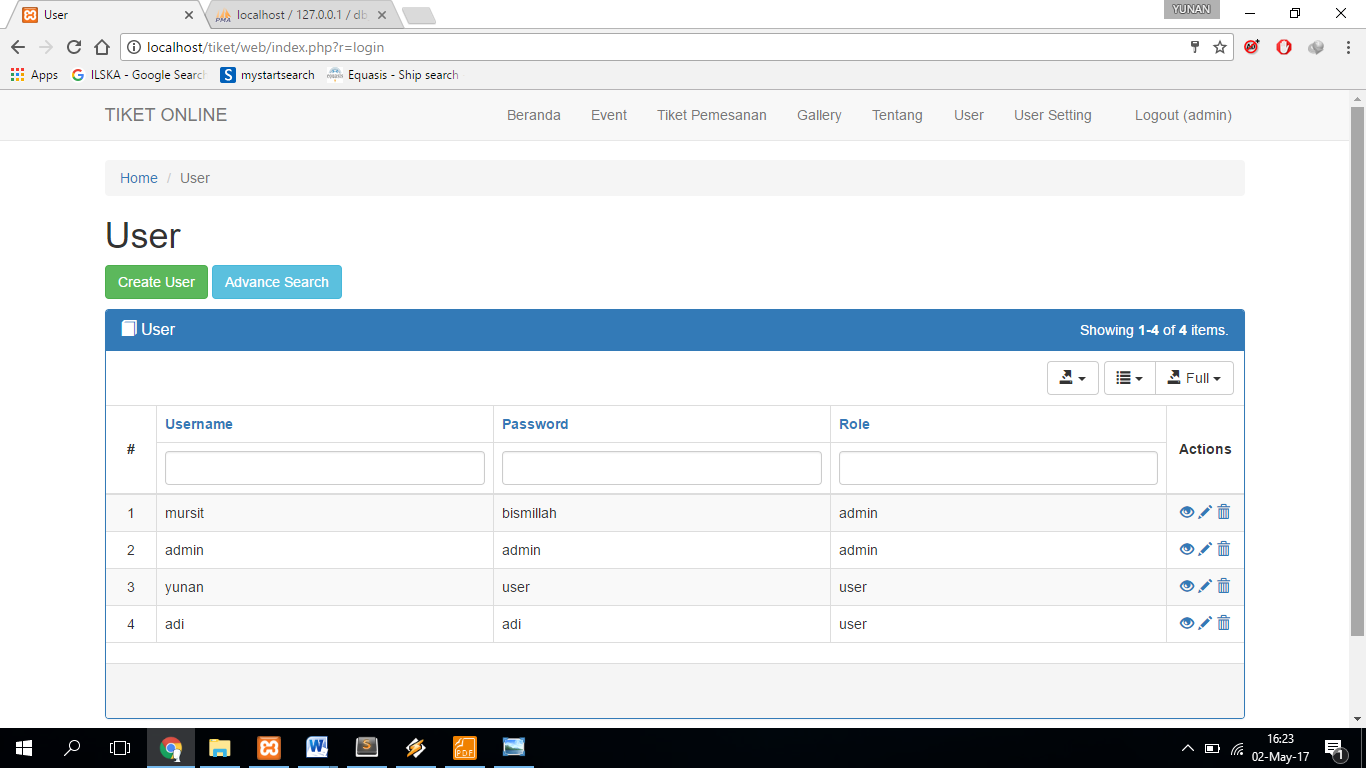
Fitur ini merupakan fitur yang digunakan pengguna untuk memberikan review pada event. Fitur ini berguna untuk pengguna yang lain dalam mengetahui tambahan informasi dari suatu event. Tampilan fitur ini dapat dilihat pada gambar 6.2.



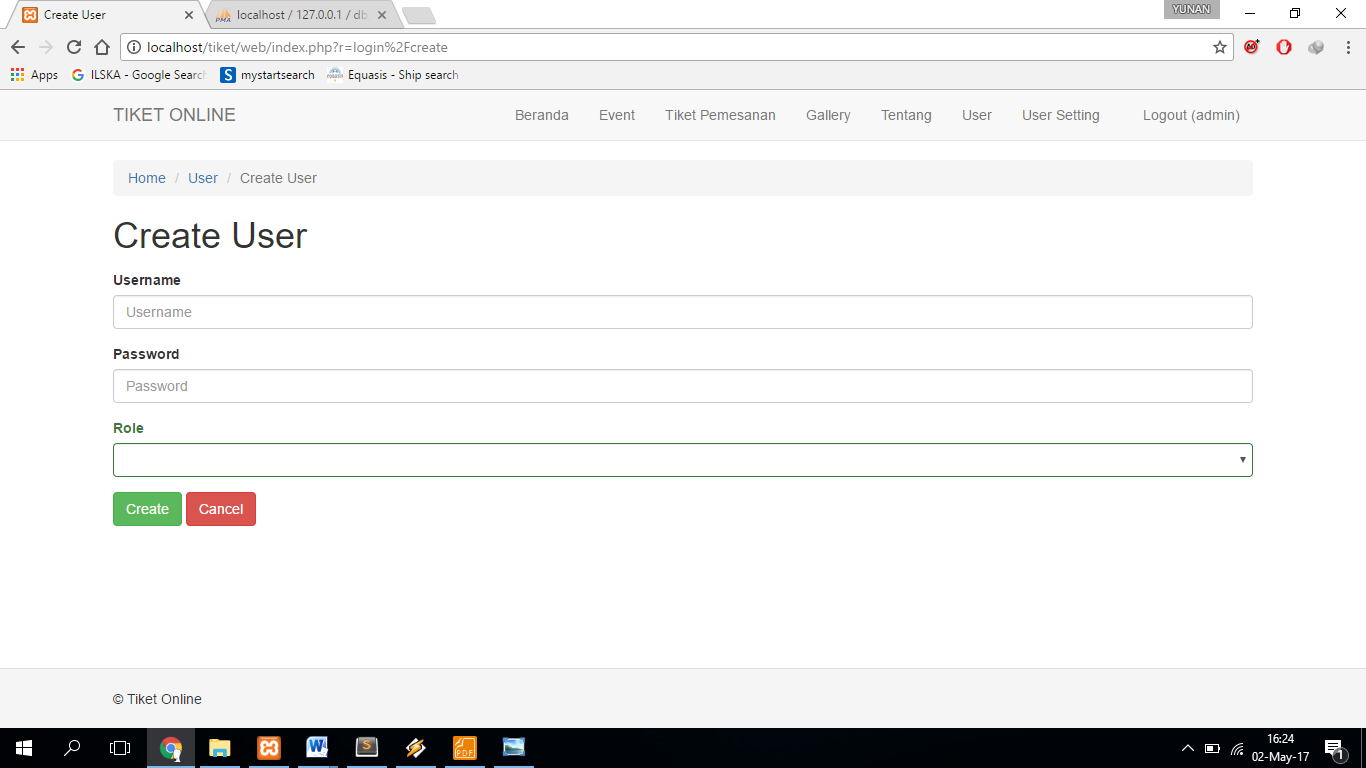
Gambar 6.2 Tampilan fitur mereview event

### Fitur Mengelola Data User

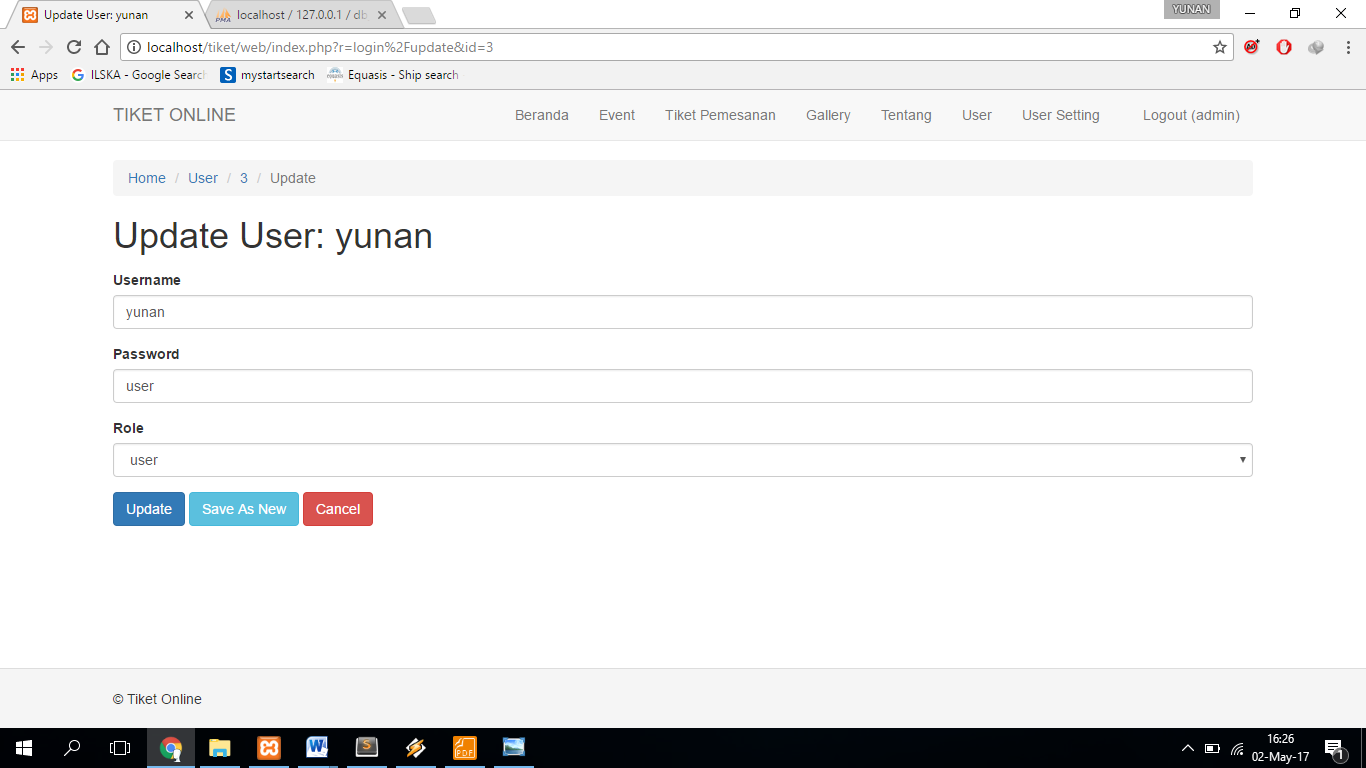
Fitur ini merupakan fitur untuk mengelola (menambah, memperbahrui, menghapus) data user , dapat dilihat pada gambar 6.3 yang hanya dapat diakses oleh admin. Fitur tambah data user merupakan fitur untuk menambah data user pada tampilan sistem, form tambah data user dapat dilihat pada gambar 6.4. Fitur ubah data user merupakan fitur untuk memperbaharui data user pada tampilan sistem form ubah data user dapat dilihat pada gambar 6.5. Fitur hapus data user merupakan fitur untuk menghapus data user pada tampilan mengelola data user, dapat dilihat pada gambar 6.6.



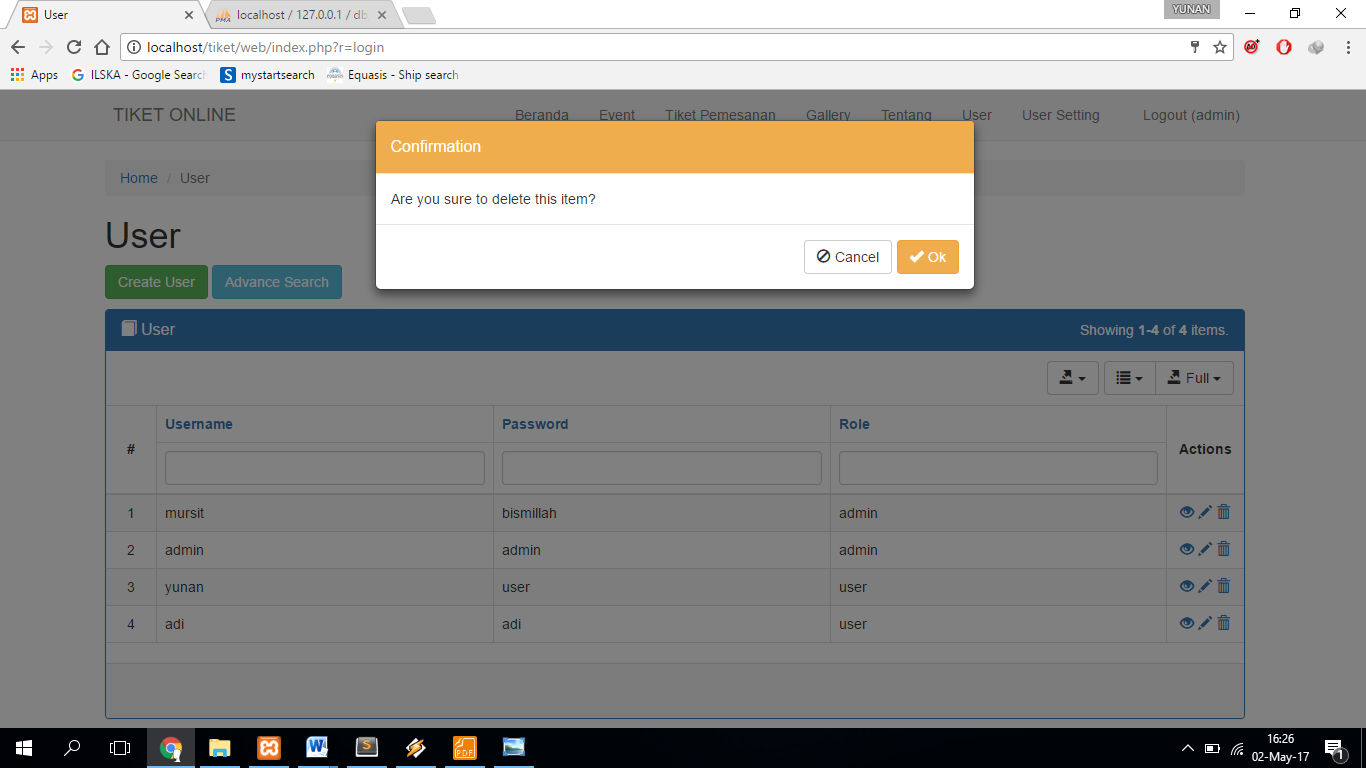
Gambar 6.3 Tampilan fitur mengelola data user



Gambar 6.4 Tampilan menambah data user



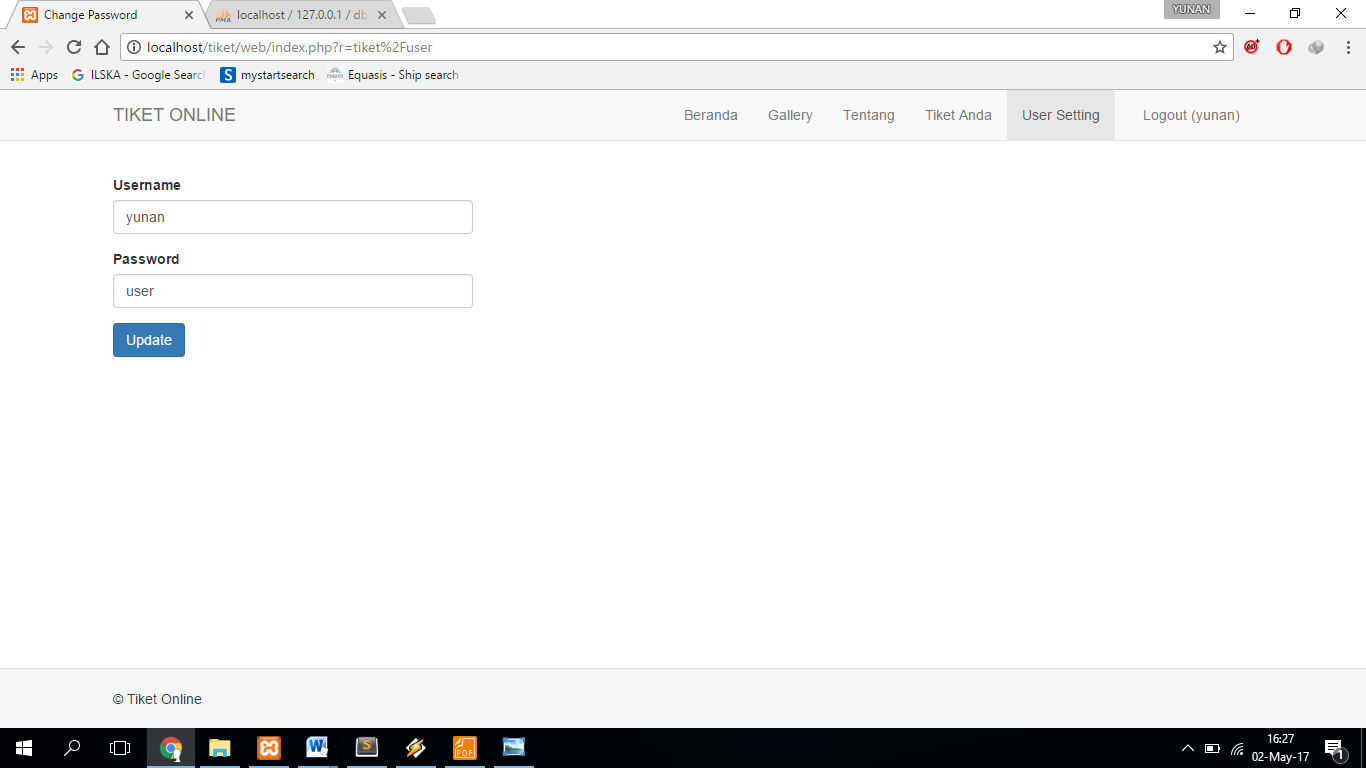
Gambar 6.5 Tampilan memperbaharui data user



Gambar 6.6 Tampilan menghapus data user

### Fitur Mengubah Data User

Fitur ini merupakan fitur yang hanya dapat diakses oleh *user/guest*, yang digunakan untuk mengubah username dan password user/guest yang bersangkutan. Tampilan dari fitur ini dapat dilihat pada gambar 6.7.



Gambar 6.7 Tampilan fitur mengubah data user

## Pembahasan Sistem Informasi Penjualan Tiket Online Event Musik

Implementasi fitur *preview* dalam sistem penjualan tiket online digunakan untuk memberikan detail event secara informatif serta kemudahan dalam pemesanan tiket. Berdasarkan metode skala likert, penilaian hasil penjualan tiket dan penerapan fitur *preview* dapat dikonversikan dalam persentase dengan kategori - kategori yang bisa dijadikan ukuran seberapa baik fitur *preview* ini memberi informasi yang informatif dan persuasif terhadap konsumen.

Berdasarkan hasil uji coba dari sekian responden, skala Likert merepresentasikan hasil dari penerapan metode *preview* untuk sistem penjualan tiket online musik dengan perhitungan sebagai berikut.

* Konsumen yang berkunjung, melihat *preview* event, membeli tiket dan memberikan komentar ataupun feedback (skor 5)
* Konsumen yang berkunjung, melihat *preview* event, dan membeli tiket (skor 4)
* Konsumen yang berkunjung dan membeli tiket (skor 3)
* Konsumen yang berkunjung dan melihat *preview* (skor 2)
* Konsumen yang hanya berkunjung saja (1)

Rumus:

S = n Pn

S = Nilai Skor

n = jumlah konsumen dari masing-masing kategori

Pn = Pilihan angka Skor Likert (1 – 5)

Kemudian dari masing-masing total skor tersebut dijumlahkan untuk mendapatkan jumlah total skor.

TS = S1 + S2 + S3 + S4 + S5

TS = Total Skor

Untuk mendapatkan hasil interpretasi, harus diketahui dahulu nilai tertinggi (Y) dan nilai terendah (X) dengan rumus sebagai berikut:

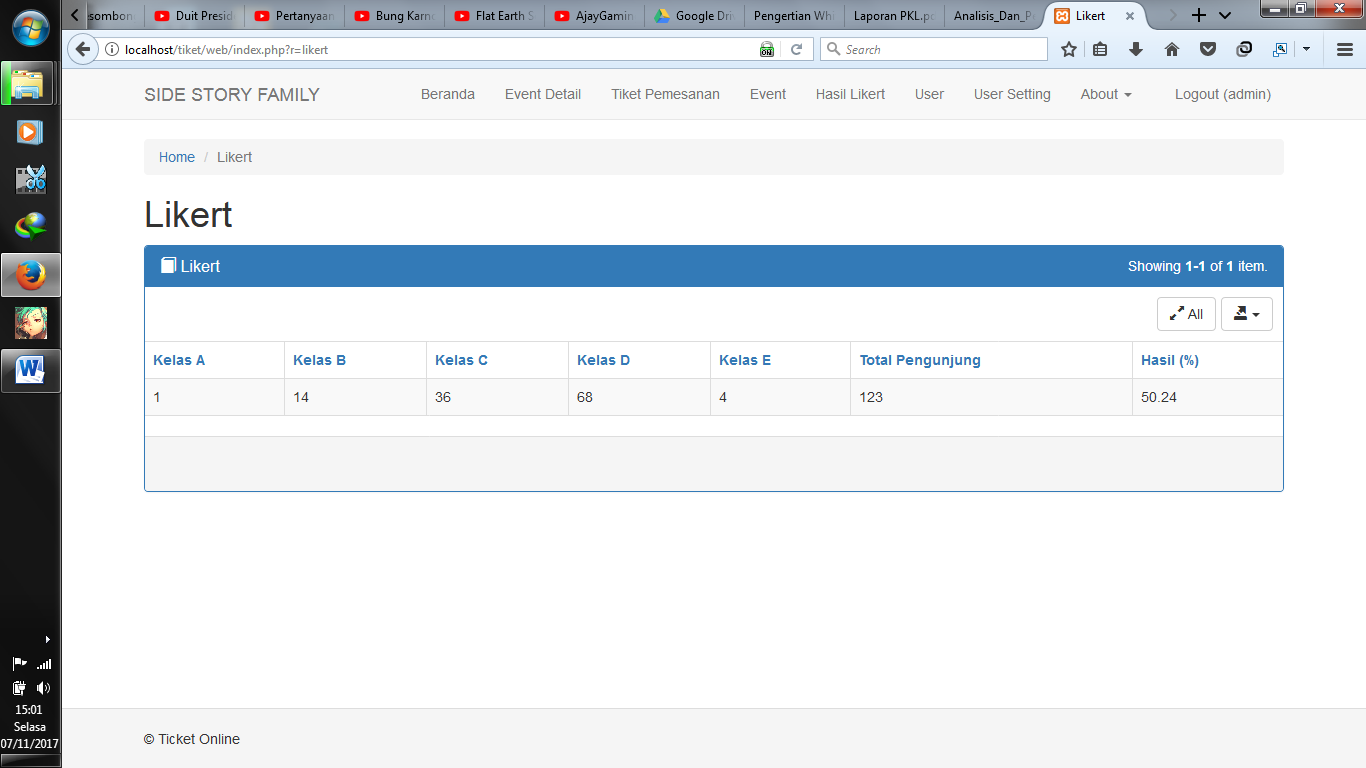
Nilai tertinggi Y = Pn (5) Tn

Nilai terendah X = Pn (1) Tn

Pn = Pilihan angka skor Likert

Tn = Total Responden

Dari kategori perhitungan yang tersebut, hasil perhitungan skala Likert dapat dilihat pada gambar 7.0.



Gambar 6.8 Hasil Perhitungan Likert

Dari gambar 6.8, didapatkan hasil dari perhitungan skala likert untuk fitur *preview.* Nilai tersebut didapatkan dengan metode perhitungan sebagai berikut:

Kelas A = skor 5 x 1 = 5

Kelas B = skor 4 x 14 = 56

Kelas C = skor 3 x 36 = 108

Kelas D = skor 2 x 68 = 136

Kelas E = skor 1 x 4 = 4

Jumlah nilai tertinggi adalah 5Tn Y. Jumlah nilai terendah adalah 1Tn X. Jika total skor yang didapat sebelumnya adalah (TS), maka interpretasi hasil penjualan adalah nilai yang dihasilkan dengan menggunakan rumus index % sebagai berikut:

INDEX % = 100

INDEX % = 100

INDEX % = 100

INDEX % = **50,2439 %**

Untuk kriteria interpretasi index skor berdasarkarkan interval adalah sebagai berikut:

|  |  |
| --- | --- |
| **Persentase** | **Kriteria** |
| 0 % - 19,99 %  20 % - 39,99 %  40 % - 59,99 %  60 % - 79,99 %  80 % - 100 % | Sangat (tidak setuju / buruk / kurang sekali)  Tidak setuju / kurang  Cukup / Netral  Setuju (Baik / Suka)  Sangat (Setuju / Baik / Suka) |

Setelah index diketahui, maka hasil penilaian penerapan fitur *preview* dalam pengembangan sistem informasi tergolong **cukup** informatif, efisien, dan efektif untuk menghindari terjadinya fenomena *“Overload Information”* dalam transaksi *e-commerce*. Oleh sebab itu, semoga dengan penerapan fitur ini dalam penjualan tiket *online* untuk EO Side Story Family diharapkan mampu menjaga eksistensi dan meningkatkan stabilitasnya dalam hal pemasaran barang dan jasa.

# BAB 6. PENUTUP

Bab ini berisi mengenai kesimpulan dan saran dari peneliti tentang penelitian yang telah dilakukan. Kesimpulan dan saran tersebut diharapkan dapat digunakan sebagai acuan pada penelitian selanjutnya.

## Kesimpulan

Kesimpulan dari hasil penelitian yang dilakukan oleh peneliti adalah sebagai berikut:

1. Aaaa

## Saran

# DAFTAR PUSTAKA

Azwar, Saifuddin. (1995). *Sikap Manusia: Teori dan Pengukurannya.* Yogyakarta: Pustaka Pelajar.

Eko Indrajit, Richardus. (2001). *E-commerce: Strategi Bisnis Di Dunia Maya*. Jakarta: PT. Elex Media Komputindo.

Furner, Christopher P dan Robert A. Zinko. (2009). *Electronic Markets: International Journal on Networked Business*. Berlin: Springer Berlin Heidelberg.

Gable, Robert K. (1986). *Instrument Development in the Afektice Domain.* Boston: Kluwer-Nijhoff Publishing.

Ginting, Elizaandayni. (2013). *Aplikasi Penjualan Berbasis Web (E-Commerce) Menggunakan Joomla Pada Mutiara Fashion.* Bandung: Universitas Widyatama.

Grisé ML, Gallupe RB. (2000). *Information overload: addressing the productivity paradox in face-to-face electronic meetings*. Journal of Management Information Systems 2000:157–85. Winter.

Hall, Shane. (2010). *How to Use the Likert Scale in Statistical Analysis.* (online)

Hidayah NA, dkk. (2010). *Sistem Informasi Pemesanan Tiket Pesawat Berbasis Web pada Bana Tour (PT. Wali Angkasamitra Utama).* Studia Informatika: Jurnal Sistem Informasi, 3(1), 2010, 1-6.

Himawan, dkk. (2014). *Analisa dan Perancangan Sistem Informasi Penjualan Online (E-Commerce) pada CV Selaras Batik Menggunakan Analisis Deskriptif.* Scientific Journal of Informatics, Vol. 1, No. 1, Mei 2014. ISSN 2407-7658.

Jogiyanto. (2007). *Model Kesuksesan Sistem Teknologi Informasi*. Yogyakarta: Andi.

Keller KL, Staelin R. (1987). *Effects of quality and quantity of information on decision effectiveness*. Journal of Consumer Research 1987(14):200–13. September.

Kotler, Philip dan Kevin Keller. (2009). *Marketing Management 13th edition*. New Jersey: Pearson Education, Inc.

Laudon, Kenneth C dan Jane P. Laudon (1998). *Computer and Information System, Edisi Ke-5*. United Stataes Of America: Times Mirror Higher Education Group.

Likert, R. (1932). *A Technique for the Measurement of Attittudes.* Archives of Physchology, 140, 1-55.

Lee BK, Lee WN. (2004). The effect of information overload on consumer choice quality in an on-line environment. Psychology and Marketing 2004(21:3):159–83. March.

Lurie NH. (2004). *Decision making in information-rich environments: the role of information structure*. Journal of Consumer Research 2004(30:4):473–86. March.

Malhotra NK, Jain AK, Lagakos SW. (1982). *The information overload controversy: an alternative viewpoint*. Journal of Marketing 1982(46):27–37. Spring.

Malhotra NK. (1982). *Information load and consumer decision making*. Journal of Consumer Research 1982(8):419–30. March.

Malhotra NK. (1984). *Reflection on the information overload paradigm in consumer decision making.* Journal of Consumer Research 1984(10:4):436–40.

Maya, I. (2010). *Proses Pengembangan Aplikasi Web*. (online)

Miftah, Shabur. (2015). *Implementasi E-Commerce Sebagai Media Penjualan Online.* Jurnal Administrasi Bisnis (JAB) Vol.29 No.1 Desember 2015. Malang: Universitas Brawijaya.

Muller TE. (1984). *Buying response to variations in product information load.* Journal of Applied Psychology 1984(69):300-6.

Risnita. (2012). *Pengembangan Skala Model Likert.* (online)

Schroder HM, Driver MJ, Streufert S. (1967). *Human Information Processing*. Holt, Rinehart, and Winston.

Y.-C. Chen et al. (2009). *Electronic Commerce Research and Applications 8.* 48-58. Taipei: Soochow University.

Wikipedia. (2010). *Likert Scale.* (online)

# LAMPIRAN

## Lampiran A. Tabel Skenario

### Tabel 4.3.1 Tabel *Scenario* Mengelola Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-01 | |
| Name *usecase* | Mengelola event | |
| Aktor | Admin | |
| Deskripsi | Merupakan *usecase* untuk membuat, memperbaharui dan menghapus event. | |
| Relasi | - | |
| Kondisi awal | Data event belum ada (menambah), data sudah ada dan belum terhapus (menghapus), data sudah ada dan belum diperbaharui (memperbaharui). | |
| Kondisi akhir | Data telah ada (menambah), data telah dihapus (menghapus), data telah berubah (memperbaharui). | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin memilih pilihan untuk mengelola event 2. Admin memilih 1 dari 3 pilihan    1. Admin memilih pilihan menambah / membuat event    2. Admin mengisi form tambah event   6.1. Admin menekan tombol simpan / buat   * 1. Admin memilih event yang akan dihapus   4.2. Admin menekan tombol hapus   * 1. Admin memilih event yang akan diperbaharui   2. Admin menekan tombol perbaharui   6.3. Admin mengisi form perbaharui data event  7.3. Admin menekan tombol simpan / perbaharui | 1. Sistem menampilkan data-data event yang sudah ada beserta 3 pilihan pengelolaan event (menambah, menghapus, memperbaharui)    1. Sistem menampilkan form tambah event   7.1. Sistem menyimpan data event tersebut  A.1. Data event tidak lengkap  8.1. Sistem menampilkan notifikasi berhasil membuat event baru.   * 1. Sistem menampilkan notifikasi berhasil menghapus event   2. Sistem menampilkan form perbaharui event   8.3. Sistem menyimpan data event yang sudah diperbaharui  A.1. Data event tidak lengkap  9.3. Sistem menampilkan notifikasi berhasil memperbaharui data event |
| Alur alternatif | A.1. Data event tidak lengkap | |
|  | A.1.2. Kembali ke alur kejadian normal no. 5.1 (untuk menambah) dan 6.3 (untuk memperbaharui) | A.1.1. Sistem menampilkan notifikasi bahwa data event yang diisikan tidak lengkap |
| Kebutuhan Khusus | - | |

### Tabel 4.3.2 Tabel *Scenario* Melihat Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-02 | |
| Name *usecase* | Melihat event | |
| Aktor | Admin, *user, guest* | |
| Deskripsi | Merupakan *usecase* untuk melihat event. | |
| Relasi | - | |
| Kondisi awal | Sistem menampilkan halaman beranda | |
| Kondisi akhir | Sistem menampilkan daftar event yang sudah didaftarkan sesuai jadwal event | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin / *user* / *guest* memilih pilihan Event | 1. Sistem menampilkan daftar event yang tersedia |
| Alur alternatif | - | |
| Kebutuhan khusus | - | |

### Tabel 4.3.4 Tabel *Scenario* Memesan Tiket

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-004 | |
| Name *usecase* | Memesan tiket | |
| Aktor | Admin, *user* | |
| Deskripsi | Merupakan *usecase* untuk memesan tiket. | |
| Relasi | - | |
| Kondisi awal | Sistem menampilkan daftar event / preview event | |
| Kondisi akhir | Sistem menampilkan notifikasi berhasil memesan tiket | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin, *user* memilih pilihan preview event 2. Admin, *user* memilih event dengan menekan tombol lihat detail 3. Admin, *user* menekan tombol pesan tiket | 1. Sistem menampilkan daftar event 2. Sistem menampilkan detail dari event yang dipilih 3. Sistem menampilkan notifikasi berhasil memesan tiket |
| Alur alternatif | - | |
| Kebutuhan khusus | Event telah tersedia dalam sistem | |

### Tabel 4.3.5 Tabel *Scenario* Mencetak Tiket

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-005 | |
| Name *usecase* | Mencetak tiket | |
| Aktor | *User, admin* | |
| Deskripsi | Merupakan *usecase* untuk mencetak tiket. | |
| Relasi | - | |
| Kondisi awal | Tiket event berhasil dipesan dan sudah dikonfirmasi oleh admin | |
| Kondisi akhir | Sistem menampilkan tiket dalam bentuk .pdf | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. *User* memilih menu melihat tiket yang telah dipesan 2. Setelah admin menerima konfirmasi pembayaran dari user, admin mengubah status tiket dari “Belum Bayar” menjadi “Sudah Bayar” 3. *User, admin* memilih pilihan cetak tiket | 1. Sistem menampilkan list tiket yang sudah dipesan 2. Sistem menampilkan tiket yang sudah dibayar 3. Sistem menampilkan tiket dalam bentuk pdf dan siap dicetak |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus | Tiket sudah dipesan dan lunas. | |

### Tabel 4.3.6 Tabel *Scenario* Mereview Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-006 | |
| Name *usecase* | Mereview event | |
| Aktor | Admin, *user* | |
| Deskripsi | Merupakan *usecase* untuk mereview event. | |
| Relasi | - | |
| Kondisi awal | Sistem menampilkan preview event dan kolom komentar | |
| Kondisi akhir | Sistem menampilkan hasil review user/admin yang telah diinputkan | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin, *user* memilih menu tampil daftar event 2. Admin, *user* menekan tombol detail event 3. Admin, *user* mengisi form review event 4. Admin, *user* menekan tombol review | 1. Sistem menampilkan daftar event 2. Sistem menampilkan halaman preview event beserta kolom review 3. Sistem menampilkan hasil komentar user/admin yang telah diinputkan |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus |  | |

### Tabel 4.3.7 Tabel *Scenario* Registrasi User

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-007 | |
| Name *usecase* | Registrasi User | |
| Aktor | *Guest* | |
| Deskripsi | Merupakan *usecase* untuk membuat data user baru | |
| Relasi | - | |
| Kondisi awal | Sistem menampilkan halaman register | |
| Kondisi akhir | Sistem menampilkan halaman login | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. *Guset* menekan tombol mendaftar 2. *Guest* mengisi kolom username dan password      1. *Guest* menekan tombol *register* | 1. Sistem menampilkan form *register* 2. Sistem menampilkan halaman login |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus |  | |

### Tabel 4.3.7 Tabel *Scenario* Mengelola Data User

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-007 | |
| Name *usecase* | Mengelola data user | |
| Aktor | Admin | |
| Deskripsi | Merupakan *usecase* untuk mengelola data user. | |
| Relasi | - | |
| Kondisi awal | Data user belum ada (menambah), data sudah ada dan belum terhapus (menghapus), data sudah ada dan belum diperbaharui (memperbaharui). | |
| Kondisi akhir | Data telah ada (menambah), data telah dihapus (menghapus), data telah berubah (memperbaharui). | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin memilih pilihan untuk mengelola data user 2. Admin memilih 1 dari 3 pilihan    1. Admin memilih pilihan menambah / membuat data user    2. Admin mengisi form tambah data user   6.1. Admin menekan tombol buat / *create*   * 1. Admin memilih data user yang akan dihapus   4.2. Admin menekan tombol hapus   * 1. Admin memilih data user yang akan diperbaharui   2. Admin menekan tombol perbaharui   6.3. Admin mengisi form perbaharui data user  7.3. Admin menekan tombol perbaharui / update | 1. Sistem menampilkan data-data user yang sudah ada beserta 3 pilihan pengelolaan event (menambah, menghapus, memperbaharui)    1. Sistem menampilkan form tambah data user   7.1. Sistem menyimpan data user tersebut  A.1. Data user tidak lengkap  8.1. Sistem menampilkan notifikasi berhasil menambahkan data user baru.   * 1. Sistem menampilkan notifikasi berhasil menghapus data user   2. Sistem menampilkan form perbaharui data user   8.3. Sistem menyimpan data user yang sudah diperbaharui  A.1. Data user tidak lengkap  9.3. Sistem menampilkan notifikasi berhasil memperbaharui data user |
| Alur alternatif | A.1. Data event tidak lengkap | |
|  | A.1.2. Kembali ke alur kejadian normal no. 5.1 (untuk menambah) dan 6.3 (untuk memperbaharui) | A.1.1. Sistem menampilkan notifikasi bahwa data event yang diisikan tidak lengkap |
| Kebutuhan khusus |  | |

### Tabel 4.3.8 Tabel *Scenario* Mengubah Data User

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-008 | |
| Name *usecase* | Mengubah data user | |
| Aktor | *User* | |
| Deskripsi | Merupakan *usecase* untuk mengubah data user (*username* dan *password*). | |
| Relasi | - | |
| Kondisi awal |  | |
| Kondisi akhir |  | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. *User* memilih pilihan *user setting* 2. *User* memperbaharui data *username* dan *password* 3. *User* menekan tombol *update* | 1. Sistem menampilkan form *update username* dan *password* user yang bersangkutan 2. Sistem menampilkan notifikasi berhasil memperbaharui data user (*username* dan *password*) |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus |  | |

## Lampiran B. Activity Diagram

### *Activity Diagram* Mengelola Event

*Activity diagram* mengelola event menggambarkan alur aktivitas padafitur pengelolaan event. Fitur melihat pengelolaan event memiliki aktivitas utama, yaitu menampilkan daftar event, menambah event, menghapus event, dan memperbaharui event yang tersedia. Aktifitas dalam melihat pengelolaan event ditunjukkan pada pada lampiran gambar 4.4.



Gambar 4.4 *Activity Diagram* Mengelola Event

### Activity Diagram Melihat Event

*Activity diagram* melihat event menggambarkan alur aktivitas padafitur melihat event. Fitur melihat event memiliki aktivitas utama, yaitu menampilkan daftar event yang tersedia. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.5.



Gambar 4.5 *Activity Diagram* Melihat Event

### Activity Diagram Preview Event

*Activity diagram* preview event menggambarkan alur aktivitas padafitur preview event. Fitur preview event memiliki aktivitas utama, yaitu menampilkan informasi detail dari suatu event yang tersedia. Detail aktifitas dalam preview event ditunjukan pada gambar 4.6.



Gambar 4.6 *Activity Diagram* Preview Event

### Activity Diagram Memesan Tiket

*Activity diagram* memesan tiket menggambarkan alur aktivitas padafitur memesan tiket. Fitur memesan tiket memiliki aktivitas utama, yaitu menampilkan daftar event yang tersedia dan mengolah pemesan tiket event. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.7.



Gambar 4.7 *Activity Diagram* Memesan Tiket

### Activity Diagram Mencetak Tiket

*Activity diagram* mencetak tiket menggambarkan alur aktivitas padafitur mencetak tiket. Fitur mencetak tiket memiliki aktivitas utama, yaitu menampilkan tiket yang telah dipesan dan mengubah form pesanan tiket ke dalam format pdf. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.7.



Gambar 4.7 *Activity Diagram* Mencetak Tiket

### Activity Diagram Mereview Event

*Activity diagram* mereview event menggambarkan alur aktivitas padafitur review event. Fitur review event memiliki aktivitas utama, yaitu menampilkan form review event dan menampilkannya. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.8.



Gambar 4.8 *Activity Diagram* Review Event

### *Activity Diagram* Registrasi User

*Activity diagram* registrasi user menggambarkan alur aktivitas padafitur registrasi user untuk *guest*. Fitur registrasi user memiliki aktivitas utama, yaitu menambah data user. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.9.



Gambar 4.8 *Activity Diagram* Registrasi User

### *Activity Diagram* Mengelola Data User

*Activity diagram* mengelola data user menggambarkan alur aktivitas padafitur data user. Fitur mengelola data user memiliki aktivitas utama, yaitu menambah data user dan menghapus data user. Detail aktifitas dalam melihat event ditunjukan pada gambar 4.9.



Gambar 4.9 *Activity Diagram* Mengelola Data User

### *Activity Diagram* Mengubah Data User

*Activity diagram* mengubah data user menggambarkan alur aktivitas padafitur update data user. Fitur mengubah data user memiliki aktivitas utama, yaitu menampilkan form update data user dan memperbaharui data user. Detail aktifitas dalam melihat event ditunjukan pada gambar 5.0.



Gambar 5.0 *Activity Diagram* Mengubah Data User

### *Activity Diagram* Melihat Hasil Likert

*Activity diagram* melihat hasil *likert* menggambarkan alur aktivitas padafitur melihat hasil *likert*. Fitur melihat hasil *likert* memiliki aktivitas utama, yaitu menampilkan hasil *likert*. Detail aktifitas dalam melihat event ditunjukan pada gambar 5.1.



Gambar 5.1 *Activity Diagram* Melihat Hasil Likert

## Lampiran C. Sequence Diagram

### *Sequence Diagram* Mengelola Event

*Sequence diagram* mengelola event digunakan untuk menunjukkan interaksi antar objek pada fitur pengelolaan event. Objek yang terlibat pada fitur mengelola event dengan *admin* sebagai aktor, antara lain *tampil (*UIKelolaEvent; FormTambahEvent; FormPerbaharuiEvent), *controller (*EventController), *model* (Event). Detail *sequence diagram* mengelola event ditunjukan pada gambar 5.2.



Gambar 5.2 *Sequence Diagram* Mengelola Event

### Sequence Diagram Melihat Event

*Sequence diagram* mengelola event digunakan untuk menunjukkan interaksi antar objek pada fitur melihat event. Objek yang terlibat pada fitur melihat event dengan *admin* sebagai aktor, antara lain *lihat* (UIEvent), *controller* (EventController), *model* (Event). Detail *sequence diagram* melihat event ditunjukan pada gambar 5.3.



Gambar 5.3 *Sequence Diagram* Melihat Event

### Sequence Diagram Memesan Tiket

*Sequence diagram* memesan tiket digunakan untuk menunjukkan interaksi antar objek pada fitur memesan tiket. Objek yang terlibat pada fitur memesan tiket dengan *admin* sebagai aktor, antara lain *tampil* (UIHome; UIEvent; UIPemesananTiket), *controller* (EventController; TiketController), *model* (Event; Tiket). Detail *sequence diagram* memesan tiket ditunjukan pada gambar 5.4.



Gambar 5.4 *Sequence Diagram* Memesan Tiket

### Sequence Diagram Mencetak Tiket

*Sequence diagram* mencetak tiket digunakan untuk menunjukkan interaksi antar objek pada fitur cetak tiket. Objek yang terlibat pada fitur cetak tiket dengan *admin* sebagai aktor, antara lain *tampil* (UIListEvent), *controller* (TiketController), *model* (Tiket). Detail *sequence diagram* mencetak tiket ditunjukan pada gambar 5.5.



Gambar 5.5 *Sequence Diagram* Mencetak Tiket

### Sequence Diagram Mereview Event

*Sequence diagram* mereview event digunakan untuk menunjukkan interaksi antar objek pada fitur review event. Objek yang terlibat pada fitur mereview event dengan *admin* sebagai aktor, antara lain *tampil* (UIHome; UIEvent; UIPemesananTiket), *controller* (EventController; TiketController), *model* (Event; Review). Detail *sequence diagram* mereview event ditunjukan pada gambar 5.6.



Gambar 5.6 *Sequence Diagram* Mereview Event

### *Sequence Diagram* Registrasi User

*Sequence diagram* registrasi user digunakan untuk menunjukkan interaksi antar objek pada fitur registrasi data user. Objek yang terlibat pada fitur registrasi user dengan *guest* sebagai aktor, antara lain *tampil* (FormRegister; FormLogin), *controller* (UserController), *model* (User). Detail *sequence diagram* mengelola data user ditunjukan pada gambar 5.7.



Gambar 5.6 *Sequence Diagram* Registrasi User

### *Sequence Diagram* Mengelola Data User

*Sequence diagram* mengelola data user digunakan untuk menunjukkan interaksi antar objek pada fitur pengelolaan data user. Objek yang terlibat pada fitur mengelola data user dengan *admin* sebagai aktor, antara lain *tampil* (UIKelolaUser; FormTambahUser; FormPerbaharuiUser), *controller* (UserController), *model* (User). Detail *sequence diagram* mengelola data user ditunjukan pada gambar 5.7.



Gambar 5.7 *Sequence Diagram* Mengelola Data User

### *Sequence Diagram* Mengubah Data User

*Sequence diagram* mengubah data user digunakan untuk menunjukkan interaksi antar objek pada fitur mengubah data user. Objek yang terlibat pada fitur mengubah data user dengan *admin* sebagai aktor, antara lain *tampil* (FormPerbaharuiUser), *controller* (UserController), *model* (User). Detail *sequence diagram* mengubah data user ditunjukan pada gambar 5.8.



Gambar 5.8 *Sequence Diagram* Mengubah Data User

### *Sequence Diagram* Melihat Hasil Likert

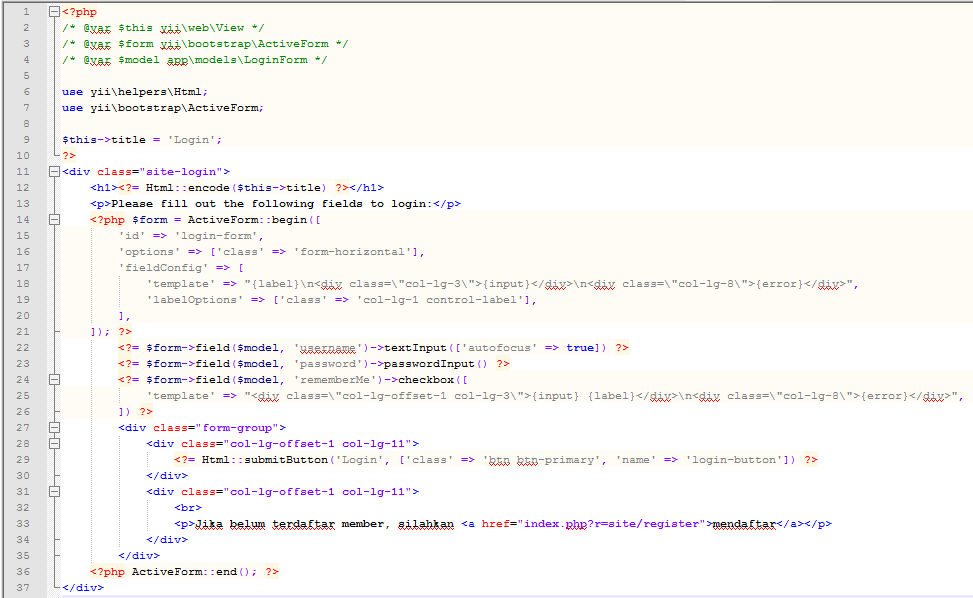
*Sequence diagram* melihat hasil *likert* digunakan untuk menampilkan hasil likert. Objek yang terlibat pada fitur melihat hasil *likert* dengan *admin* sebagai aktor, antara lain *tampil* (*UILikert*), *controller* (*LikertController*), *model* (User). Detail *sequence diagram* mengubah data user ditunjukan pada lampiran gambar 5.9.



Gambar 5.9 *Sequence Diagram* Melihat Hasil Likert

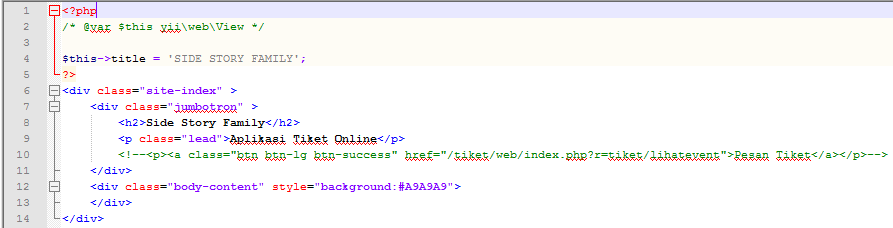
## Lampiran D. Implementasi Kode Program (Tampilan)

### Login



Gambar A. Kode program tampilan login (views/site/login.php)

### Home



Gambar A. Kode program tampilan Home (views/site/index.php)

### Event

#### Mengelola Event



Gambar A.3.1. Kode program tampilan mengelola event (views/event/index.php)

bagian 1



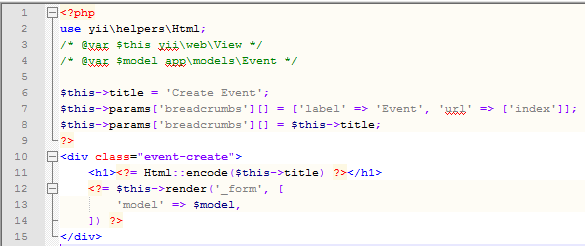
Gambar A.3.1. 2 Kode program tampilan mengelola event (views/event/index.php) bagian 2



Gambar A.3.1. 3 Kode program tampilan mengelola event (views/event/index.php)

bagian 3

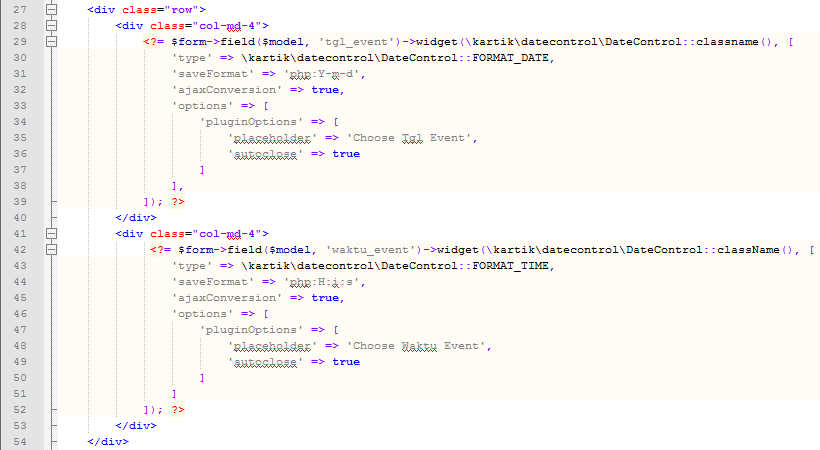
#### Menambah Event



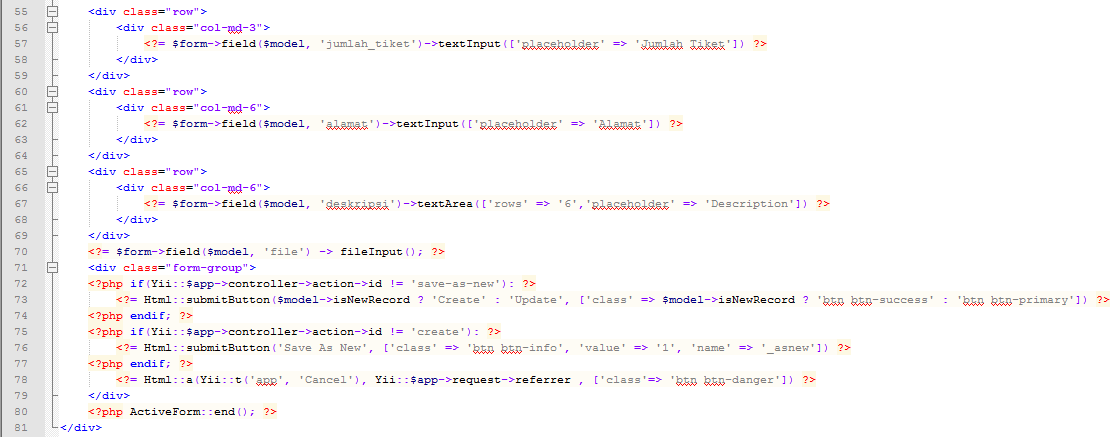
Gambar A.3.2. Kode program tampilan menambah event (views/event/create.php)



Gambar A.3.2.2 Kode program tampilan menambah event (views/event/\_form.php) bagian 1

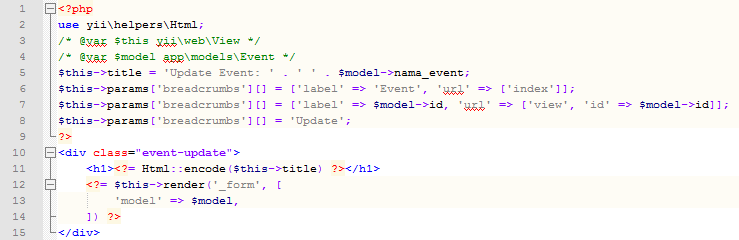


Gambar A.3.2.3 Kode program tampilan menambah event (views/event/\_form.php) bagian 2



Gambar A.3.2.4 Kode program tampilan menambah event (views/event/\_form.php) bagian 3

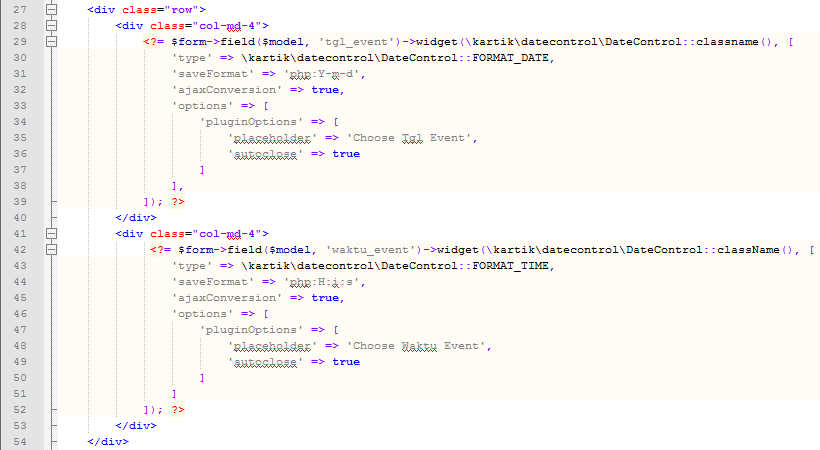
#### Update Event



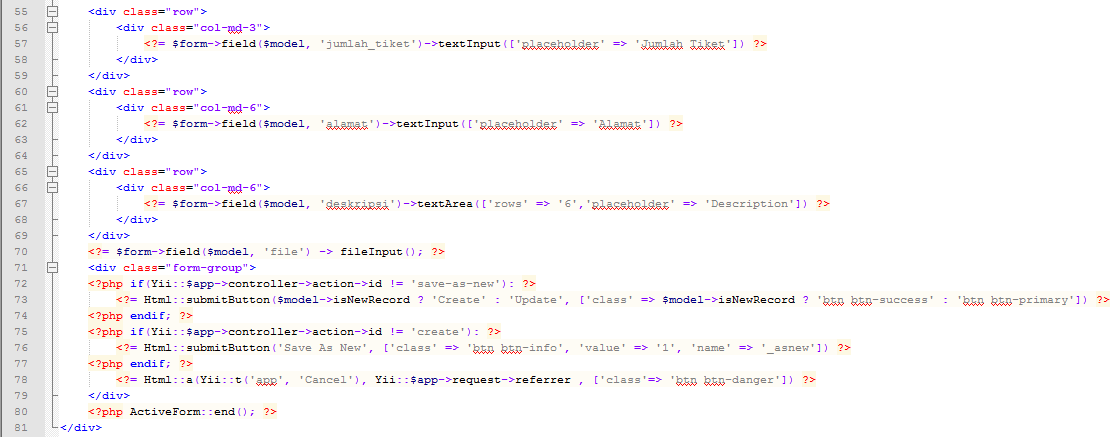
Gambar A.3.3. Kode program tampilan update event (views/event/update.php)



Gambar A.3.3. Kode program tampilan update event (views/event/\_form.php) bagian 1



Gambar A.3.3. Kode program tampilan update event (views/event/\_form.php) bagian 2



Gambar A.3.3. Kode program tampilan update event (views/event/\_form.php) bagian 3

#### Melihat List Event



Gambar A.3.4 Kode program tampilan melihat event (views/tiket/event.php)

#### Melihat Detail Event



Gambar A.3.5. Kode program tampilan melihat detail event (views/event/view.php)

bagian 1

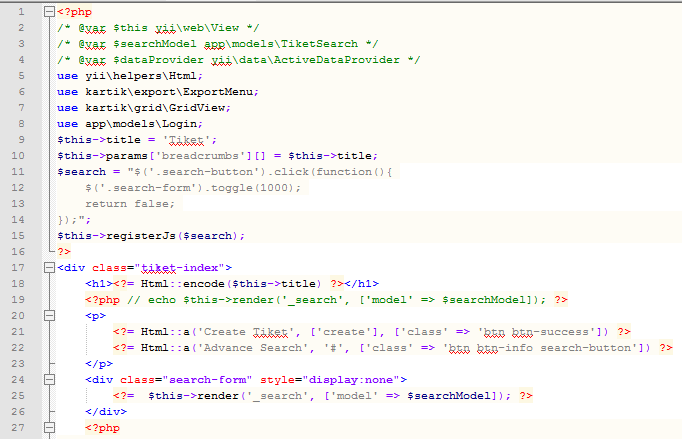


Gambar A.3.5. Kode program tampilan melihat detail event (views/event/view.php)

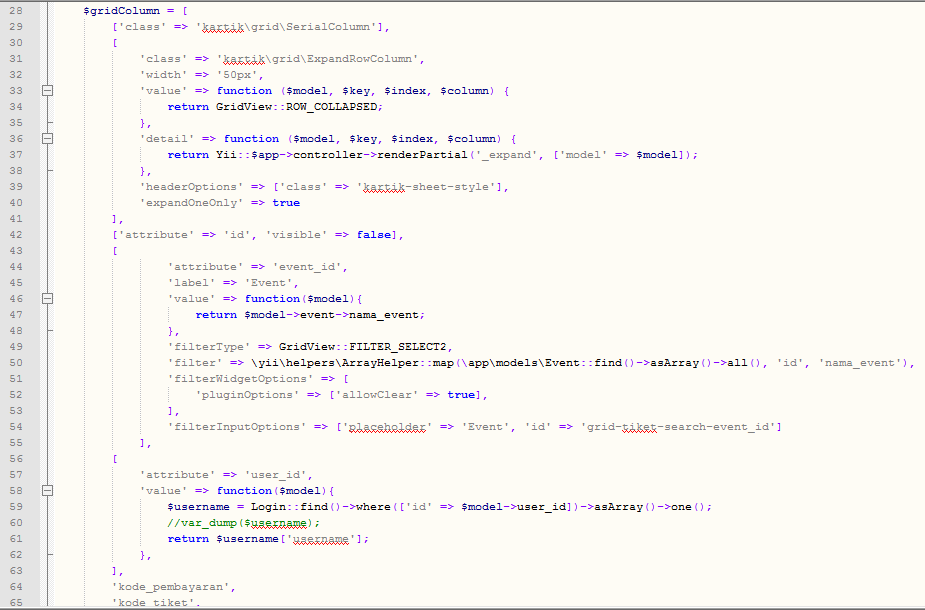
bagian 2

### Tiket

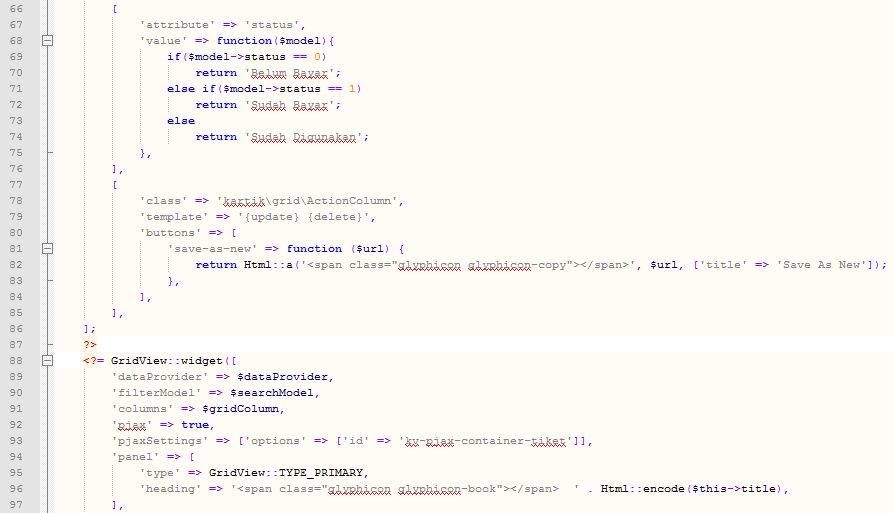
#### Mengelola Tiket



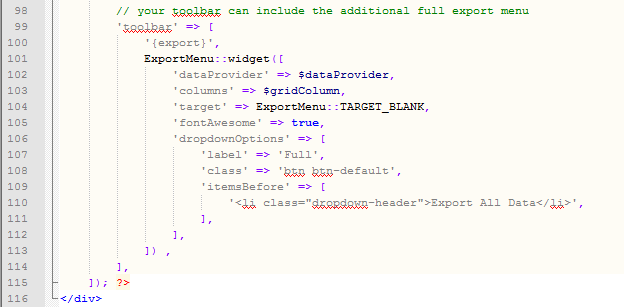
Gambar A.4.1. Kode program tampilan mengelola tiket (views/tiket/index.php) bagian 1



Gambar A.4.1. Kode program tampilan mengelola tiket (views/tiket/index.php) bagian 2

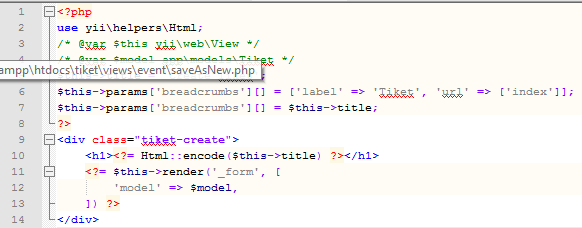


Gambar A.4.1. Kode program tampilan mengelola tiket (views/tiket/index.php) bagian 3

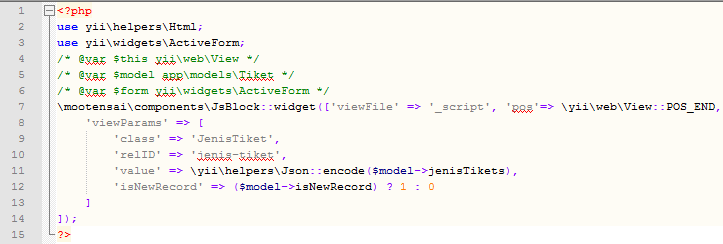


Gambar A.4.1. Kode program tampilan mengelola tiket (views/tiket/index.php) bagian 4

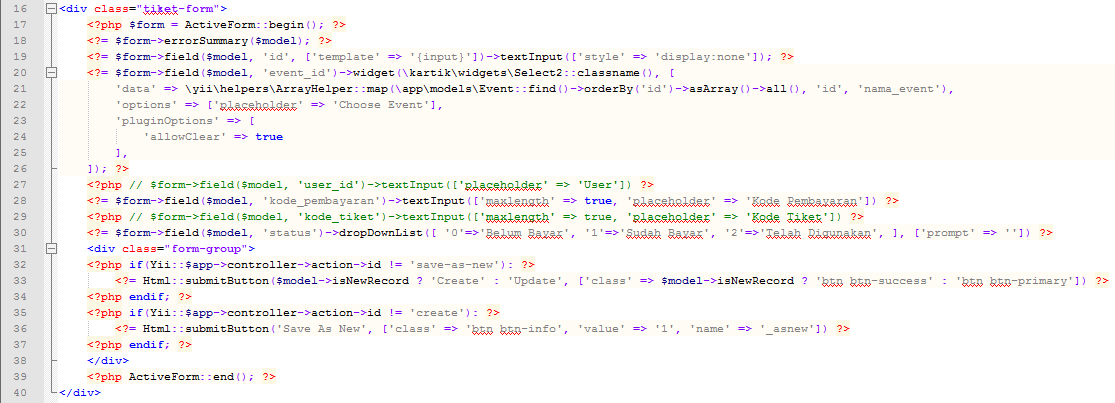
#### Menambah Tiket



Gambar A.4.2. Kode program tampilan menambah tiket (views/tiket/create.php)

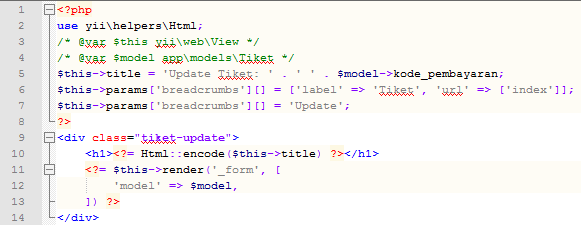


Gambar A.4.2. Kode program tampilan menambah tiket (views/tiket/\_form.php) bagian 1

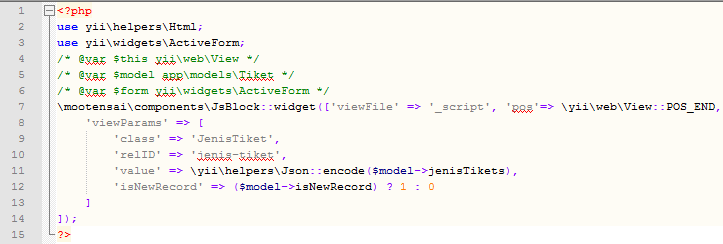


Gambar A.4.2. Kode program tampilan menambah tiket (views/tiket/\_form.php) bagian 2

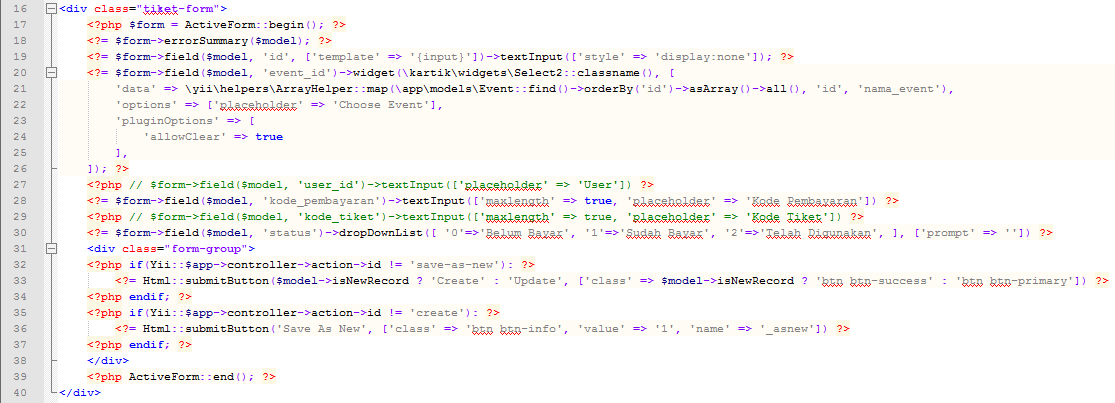
#### Update Tiket



Gambar A.4.3. Kode program tampilan update event (views/tiket/update.php)

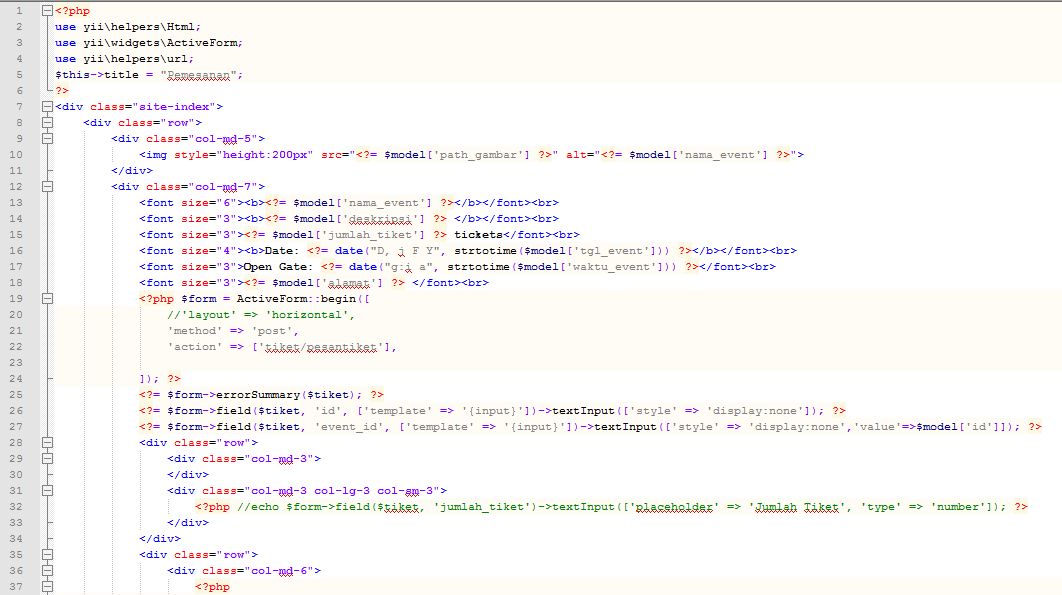


Gambar A.4.3. Kode program tampilan update event (views/tiket/\_form.php) bagian 1



Gambar A.4.3. Kode program tampilan update event (views/tiket/\_form.php) bagian 2

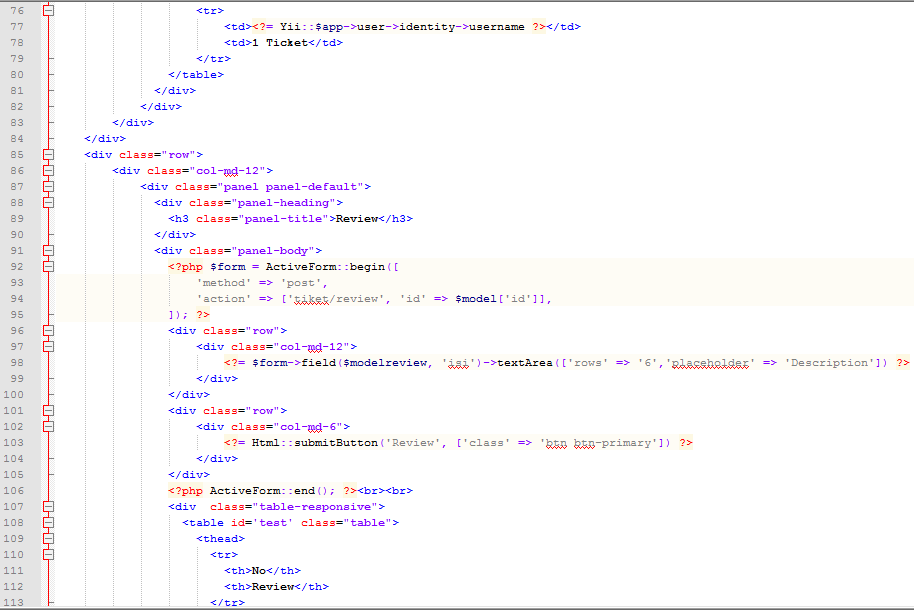
#### Preview Tiket



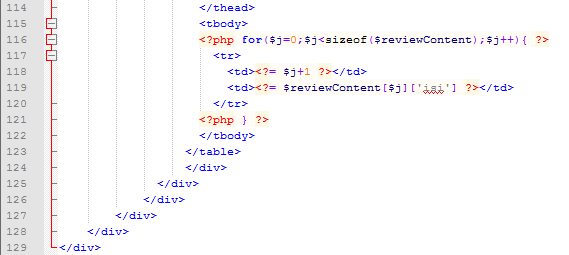
Gambar A.4.4. Kode program tampilan pesan tiket dan review event (views/tiket/pemesanan.php) bagian 1



Gambar A.4.4. Kode program tampilan pesan tiket dan review event (views/tiket/pemesanan.php) bagian 2

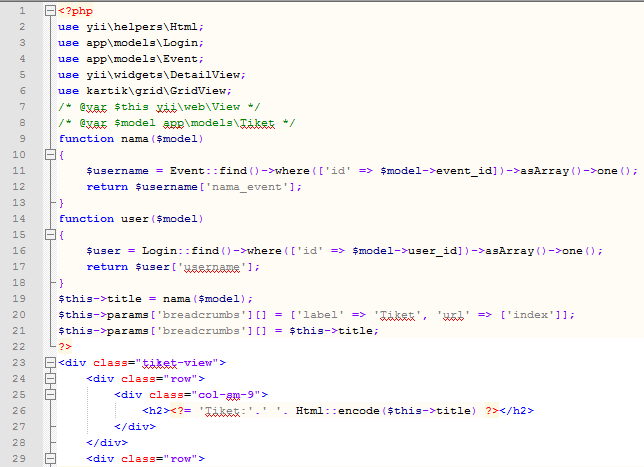


Gambar A.4.4. Kode program tampilan pesan tiket dan review event (views/tiket/pemesanan.php) bagian 3



Gambar A.4.4. Kode program tampilan pesan tiket dan review event (views/tiket/pemesanan.php) bagian 4

#### Cetak Tiket



Gambar A.4.5. Kode program tampilan cetak tiket (views/tiket/\_cetaktiket.php) bagian 1



Gambar A.4.5. Kode program tampilan cetak tiket (views/tiket/\_cetaktiket.php) bagian 2

### User

#### Mengelola User

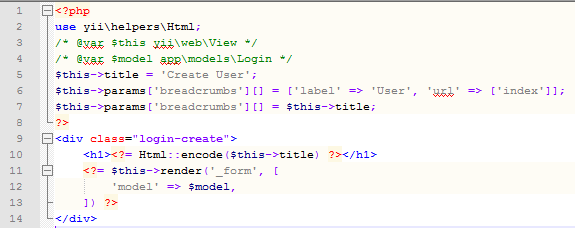


Gambar A.5.1. Kode program tampilan mengelola user (views/login/index.php) bagian 1



Gambar A.5.1. 2 Kode program tampilan mengelola user (views/login/index.php) bagian 2

#### Menambah User

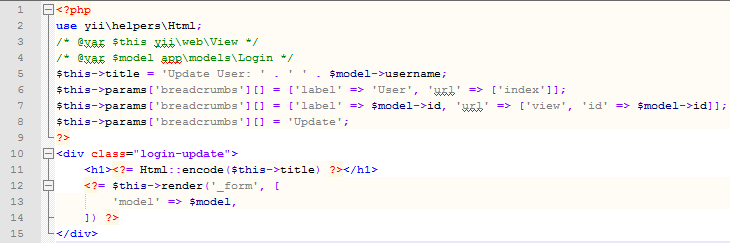


Gambar A.5.2. Kode program tampilan menambah user (views/login/create.php)



Gambar A.5.2. 2 Kode program tampilan menambah user (views/login/\_form.php)

#### Update User



Gambar A.5.3. Kode program tampilan update user (views/login/update.php)

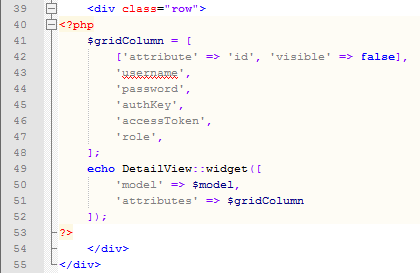


Gambar A.5.3. Kode program tampilan update user (views/login/\_form.php)

#### Melihat Detail User



Gambar A.5.4. Kode program tampilan melihat detail user (views/login/view.php) bagian 1



Gambar A.5.4. Kode program tampilan melihat detail user (views/login/view.php) bagian 2

## Lampiran C. Hasil Pengujian

### Lampiran Black Box Testing

| No. | Menu | Fungsi | Kasus | Hasil | Keterangan |
| --- | --- | --- | --- | --- | --- |
| 1. | Login | Menu ini berfungsi sebagai keamanan sistem. memilah *user* yang dapat menggunakan sistem ini. | Ketika *guest* memasukan nama *username* dan password dengan benar, sistem menampilan halaman beranda beserta menu sesuai level user | Sistem menampilan halaman beranda beserta menu sesuai level user | Berhasil |
|  |  |  |  |  |
|  |  |  |  |  |
| 2. | Register | Menu ini berfungsi untuk mendaftarkan data user baru | Ketika *guest* memasukkan username dan password dengan benar, sistem menampilkan halaman login | Sistem menampilkan halaman login | Berhasil |
| 3 | Beranda | Menampilkan halaman awal ketika admin, user, guest mengakses sistem | Ketika *guest* mengakses sistem, admin, user telah melakukan login, sistem akan menampilkan halaman beranda | Sistem menampilkan halaman beranda | Berhasil |
| 4. | Event Detail | Untuk melihat data event, detail event, ubah data event, dan hapus data event | Ketika *admin* klik menu event detail, sistem menampilkan halaman mengelola data event | Sistem menampilkan halaman mengelola event | Berhasil |
|  |  |  | Ketika *admin* klik tombol create event, sistem menampilkan form tambah event | Sistem menampilkan form tambah event | Berhasil |
|  |  |  | Ketika *admin* klik tombol *create* pada form tambah event, sistem menyimpan data event menampilkan di halaman data event | Sistem menyimpan data event menampilkan di halaman data event | Berhasil |
|  |  |  | Ketika *admin* klik tombol view event, sistem menampilkan detail event | Sistem menampilkan detail event | Berhasil |
|  |  |  | Ketika *admin* klik tombol update event, sistem menampilkan form update data event | Sistem menampilkan form update data event | Berhasil |
|  |  |  | Ketika *admin* klik tombol *update* pada form update data event, sistem menyimpan data event menampilkan di halaman data event | Sistem menyimpan data event menampilkan di halaman data event | Berhasil |
|  |  |  | Ketika *admin* klik tombol hapus, sistem menampilkan pesan konfirmasi hapus data event, jika dipilih OK maka sistem menghapus data event yang dipilih dan menampilkan halaman mengelola data event | Sistem menampilkan pesan konfirmasi hapus data event, jika dipilih OK maka sistem menghapus data event yang dipilih dan menampilkan halaman mengelola data event | Berhasil |
| 5. | Event | Melihat list event, memesan tiket, mereview event | Ketika *user* klik menu event, sistem menampilkan list event | Sistem menampilkan list event | Berhasil |
|  |  |  | Ketika *user* klik tombol pesan tiket, sistem menampilkan detail dari event yang dipilih beserta kolom untuk mereview event yang dipilih | Sistem menampilkan detail dari event yang dipilih beserta kolom untuk mereview event yang dipilih | Berhasil |
|  |  |  | Ketika *user* klik tombol pesan tiket sekarang, sistem menampilkan notifikasi berhasil memesan tiket dan berhasil menyimpn data pemesanan tiket | Sistem menampilkan notifikasi berhasil memesan tiket dan berhasil menyimpn data pemesanan tiket | Berhasil |
|  |  |  | Ketika *user* mereview event kemudian klik tombol review, sistem menyimpan review dan menampilkannya pada halaman detail event | Sistem menyimpan review dan menampilkannya pada halaman detail event | Berhasil |
| 6. | Tiket Pemesanan | Untuk melihat data tiket, detail tiket, ubah data tiket, dan hapus data tiket | Ketika *admin* klik menu tiket pemesanan, sistem menampilkan halaman data tiket | Sistem menampilkan halaman data tiket | Berhasil |
|  |  |  | Ketika *admin* klik tombol create tiket, sistem menampilkan form tambah tiket | Sistem menampilkan form tambah tiket | Berhasil |
|  |  |  | Ketika *admin* klik tombol *create* pada form tambah tiket, sistem menyimpan data tiket menampilkan di halaman data tiket | Sistem menyimpan data tiket menampilkan di halaman data tiket | Berhasil |
|  |  |  | Ketika *admin* klik tombol update tiket, sistem menampilkan form update data event | Sistem menampilkan form update data event | Berhasil |
|  |  |  | Ketika *admin* klik tombol *update* pada form update data tiket, sistem menampilkan form update data tiket | Sistem menampilkan form update data tiket | Berhasil |
|  |  |  | Ketika *admin* klik tombol hapus, sistem menampilkan pesan konfirmasi hapus data tiket, jika dipilih OK maka sistem menghapus data tiket yang dipilih dan menampilkan halaman mengelola data tiket | Sistem menampilkan pesan konfirmasi hapus data tiket, jika dipilih OK maka sistem menghapus data tiket yang dipilih dan menampilkan halaman mengelola data tiket | Berhasil |
|  |  |  | Ketika *admin* klik tombol ok pada *waning* hapus data user, sistem Menghapus data user menampilkan di halaman data tiket | Sistem menghapus data user menampilkan di halaman data tiket | Berhasil |
| 7. | Tiket Anda | Untuk melihat tiket yang sudah dipesan, status dari tiket tersebut, serta untuk mencetak tiket | Ketika user klik menu Tiket Anda, sistem menampilkan list tiket yang sudah dipesan beserta status pembayaran tiket tersebut | Sistem menampilkan list tiket yang sudah dipesan beserta status pembayaran tiket tersebut | Berhasil |
|  |  |  | Ketik user klik tombol mencetak, sistem menampilkan tiket dalam bentuk pdf dan siap untuk dicetak | Sistem menampilkan tiket dalam bentuk pdf dan siap untuk dicetak | Berhasil |
| 8. | User | Untuk menambahkan, melihat, mengubah, dan hapus data user | Ketika *admin* klik menu User, sistem menampilkan halaman mengelola data user | sistem menampilkan halaman mengelola data user | Berhasil |
|  |  |  | Ketika *admin* klik tombol create user, sistem menampilkan form create user | Sistem menampilkan form create user | Berhasil |
|  |  |  | Ketika *admin* klik tombol *create* pada form create user, sistem menyimpan data user yang diinputkan dan menampilkan halaman mengelola data user | Sistem menyimpan data user yang diinputkan dan menampilkan halaman mengelola data user | Berhasil |
|  |  |  | Ketika *admin* klik tombol view user, sistem menampilkan detail dari user yang dipilih | Sistem menampilkan detail dari user yang dipilih | Berhasil |
|  |  |  | Ketika *admin* klik tombol update user, sistem menampilkan form update data user yang dipilih | Sistem menampilkan form update data user yang dipilih | Berhasil |
|  |  |  | Ketika *admin* klik tombol *update* pada form update data user, sistem menyimpan data user dan menampilkan halaman mengelola data user | Sistem menyimpan data user dan menampilkan halaman mengelola data user | Berhasil |
|  |  |  | Ketika *admin* klik tombol hapus, sistem menampilkan pesan konfirmasi hapus data user, jika dipilih OK maka sistem menghapus data user yang dipilih dan menampilkan halaman mengelola data user | Sistem menampilkan pesan konfirmasi hapus data user, jika dipilih OK maka sistem menghapus data user yang dipilih dan menampilkan halaman mengelola data user | Berhasil |
| 9. | *User Setting* | Mengganti username dan password dari user yang bersangkutan | Ketika *user* klik menu user setting | Menampilkan form ubah username dan password | Berhasil |
|  |  |  | Ketika user klik tombol update | Menyimpan username dan password yang baru | Berhasil |
| 10. | Profil | Untuk menampilkan profil EO (Event Orginizer) Event Musik | Ketika admin, user, guest klik menu profil, sistem menampilkan halaman profil EO Event Musik | Sistem menampilkan halaman profil EO Event Musik | Berhasil |
| 11. | Pemesanan Layanan | Untuk menamppilkan detail harga pemesanan layanan, dokumentasi panggung dan *equipment*, beserta kontak person EO Event Musik | Ketika admin, user, guest klik menu Pemesanan Layanan, sistem menampilkan detail harga pemesanan layanan, dokumentasi panggung dan *equipment*, beserta kontak person EO Event Musik | Sistem menampilkan detail harga pemesanan layanan, dokumentasi panggung dan *equipment*, beserta kontak person EO Event Musik | Berhasil |
| 12. | *Logout* | Untuk mengeluarkan admin, user dari sistem | Ketika admin, user klik menu *Logout*, sistem mengeluarkan akun yang bersangkutan dan menampilkan halaman beranda sebagai guest | Sistem mengeluarkan akun yang bersangkutan dan menampilkan halaman beranda sebagai guest | Berhasil |