**PROPOSAL SKRIPSI**

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

Disusun oleh:

Friendy Budi Sulistyo (102410101009)

**PROGRAM STUDI SISTEM INFORMASI**

**UNIVERSITAS JEMBER**

**2016**

1. **JUDUL**

Pengembangan Sistem Penjualan Tiket *Online* Event Musik Dengan Fitur *Preview* (studi kasus: *event organizer* side story family).

1. **LATAR BELAKANG**

Perkembangan teknologi telekomunikasi dan internet menyebabkan terjadinya perubahan kultur manusia sehari-hari. Dalam era yang disebut *information age* ini, media elektronik menjadi salah satu media andalan untuk melakukan komunikasi dan bisnis.

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*. Banyak situs-situs *website* yang sudah menerapkan sistem *e-commerce*. Begitu pula dengan *e-commerce* dalam bidang entertainment khususnya musik. Baik dalam event konser, festival, *music expo*, indie, album launching, *training event* dan lain-lain.

Penerapan sistem *e-commerce* dalam penjualan tiket event musik bertujuan untuk memudahkan akses informasi dan pemesanan tiket untuk konsumen ketika EO menyelenggarakan suatu event yang di dalamnya terdapat artis-artis favorit yang akan tampil, dimana konsumen harus mempunyai sebuah tiket event untuk bisa melihat event tersebut. 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 pertemuan secara langsung dengan agen.

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 tertarik untuk merancang sistem penjualan tiket event musik secara *online* melalui *web*site untuk memudahkan dalam mengakses 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 tanggapan konsumen dengan adanya sistem ini, untuk itu penulis melakukan penelitian dan merancang kuesioner untuk penelitian tersebut. Dari dasar permasalahan inilah penulis ingin mengambilnya sebagai judul tugas akhir.

1. **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.
3. **BATASAN MASALAH**
4. Program ini mengenai sistem penjualan *online* tiket event musik.
5. Sistem yang dibangun berbasis *website.*
6. User atau pengguna sistem ini adalah admin dari EO (*Event Organizer*) dan konsumen pemesan tiket.
7. Output yang dihasilkan dari program ini berupa tiket event, preview event, dan data transaksi.
8. **TUJUAN PENELITIAN**
9. Untuk merancang sistem penjualan tiket *online* event musik dengan fitur preview.
10. Untuk memberikan kemudahan dalam pemesanan tiket event musik.
11. Sebagai bentuk penerapan teknologi saat ini untuk pengembangan sistem penjualan tiket event musik.
12. Sebagai bentuk uji coba penulis untuk mengetahui tanggapan konsumen terhadap fitur *preview* dari sistem penjualan *online* tiket event musik.
13. **MANFAAT PENELITIAN**

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 tanggapan konsumen tentang event yang diselenggarakan dengan fitur-fitur yang ada didalam 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.

1. **TINJAUAN PUSTAKA**
2. 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 penjualan serta pembelian melalui media elektronik.

1. 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.

1. Konsep Dasar *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.

1. 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 di selenggarakan 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.

1. *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.

1. *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.*

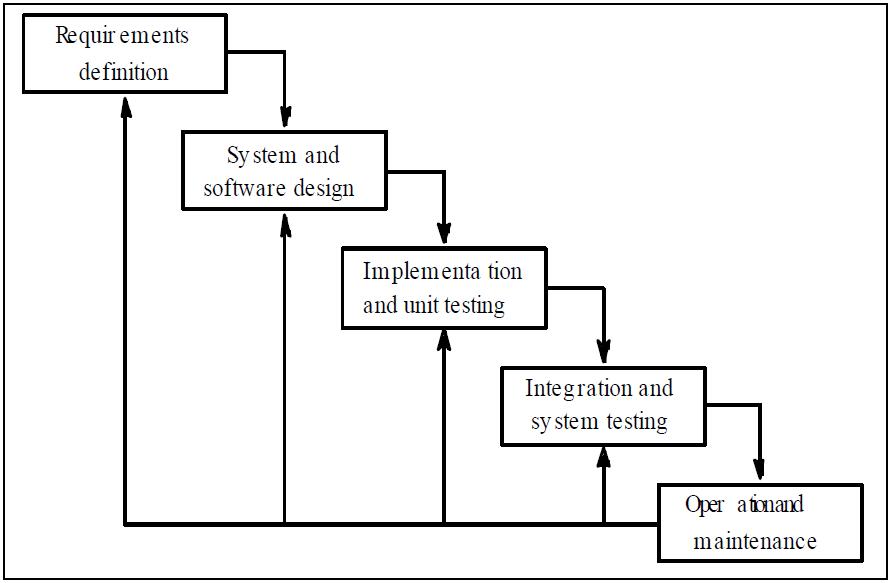
*MySQL* adalah perangkat lunak yang pertama kali oleh didukung oleh bahasa pemprograman *script* untuk internet (*PHP* dan *Perl*). *MySQL* dan *PHP* adalah pasangan *software* pengembangan aplikasi web yang ideal. *MySQL* sering digunakan untuk membuat aplikasi berbasis web dengan menggunakan bahasa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

1. *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.

1. **METODOLOGI PENELITIAN**
2. 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 1. Fase-fase dalam model *waterfall*

Sumber: Proboyekti, 2011

* 1. *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.

* 1. *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* 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 skenario 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.

* 1. *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).*

* 1. *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).

* 1. *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.

1. Fitur *Preview*

Seperti yang telah dijelaskan sebelumnya, bahwa *preview* adalah ulasan informasi tentang suatu produk yang ditawarkan dalam suatu penjualan. Dalam hal ini penulis meneliti bagaimana tanggapan dari konsumen akan adanya penerapan fitur ini dalam penjualan *online* tiket event musik.

Untuk mengetahui bagaimana tanggapan konsumen tentang penerapan fitur *preview* sebagai bentuk nilai lebih dari sistem ini, penulis merancang kuesioner dengan mengajukan beberapa pertanyaan yang dijadikan sebagai bahan untuk menghitung seberapa besar persentase konsumen yang akan memutuskan membeli tiket dengan penerapan fitur *preview* yang mampu menjadi daya tarik dalam penjualan tiket *online* event musik.

Untuk perhitungannya, 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. Nama skala ini diambil dari nama penciptanya *Rensis Likert,* pendidik dan ahli psikolog Amerika Serikat. Rensis Likert telah mengembangkan sebuah skala untuk mengukur sikap masyarakat ditahun 1932. Ketika menanggapi pertanyaan dalam skala Likert, responden menentukan tingkat persetujuan mereka terhadap suatu pernyataan dengan memilih salah satu dari pilihan yang tersedia. Pengungkapan sikap dengan menggunakan skala Likert sangat popular di kalangan para ahli dan peneliti. Hal ini dikarenakan selain praktis, skala Likert yang dirancang baik pada umumnya memiliki reabilitas yang memuaskan.

Skala Likert umumnya digunakan untuk mengukur sikap melalui serangkaian kalimat yang mengatakan sesuatu mengenai objek yang hendak diungkap. Skala Likert berwujud kumpulan pernyataan sikap yang ditulis, disusun, dan dianalisis sedemikian rupa sehingga respon seseorang terhadap pernyataan tersebut dapat diberikan angka (skor) dan kemudian dapat diinterpretasikan. Skala Likert tidak hanya terdiri dari hanya satu stimulus atau satu pernyataan saja melainkan selalu berisi banyak item (*multiple item measure*).

Skala Likert adalah sebuah tipe skala psikometri yang menggunakan angket dan menggunakan skala yang lebih luas dalam penelitian survei. 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. Dalam pendekatan ini tidak diperlukan adanya kelompok panel penilai (*judging group*) dikarenakan nilai skala setiap pernyataan tidak akan ditentukan oleh derajat ketertarikannya masing-masing, akan tetapi ditentukan oleh distribusi respon setuju atau tidak setuju dari segenap responden yang bertindak sebagai kelompok uji coba.

Kelompok uji coba ini hendaknya memiliki karakteristik yang semirip mungkin dengan karakteristik individu yang hendak diungkap sikapnya oleh skala yang disusun. Selain itu, agar hasil analisis dalam penskalaannya lebih cermat dan stabil. Responden yang dipilih sebagai kelompok uji coba harus berjumlah banyak sehingga distribusi skor mereka lebih bervariasi. Banyaknya responden dalam uji coba ini menurut saran Gable adalah sekitar 6 sampai 10 kali lipat banyaknya pernyataan yang akan dianalisis.

Prosedur penskalaan dengan metode Likert didasari oleh dua asumsi yaitu:

* + 1. Setiap pernyataan yang telah ditulis dapat disepakati sebagai pernyataan yang termasuk disukai dan pernyataan yang tidak disukai.
    2. Untuk pernyataan positif, jawaban yang diberikan oleh responden yang memiliki sikap positif harus diberi nilai yang lebih tinggi dari jawaban yang diberikan oleh responden yang bersikap negatif. Demikian sebaliknya, untuk pernyataan negatif, jawaban yang diberikan oleh responden bersikap negatif harus diberi nilai yang lebih tinggi dari jawaban responden yang bersikap positif.

Responden mengindikasi tingkat keyakinan mereka dengan evaluasi objektif ataupun subjektif. Biasanya dalam skala Likert terbagi dalam lima kategori yang digunakan, lima kategori tersebut adalah:

|  |  |
| --- | --- |
| **Pernyataan Positif (+)** | **Pernyataan Negatif (-)** |
| 1. Sangat Tidak Setuju (skor 1) 2. Tidak Setuju (skor 2) 3. Netral (skor 3) 4. Setuju (skor 4) 5. Sangat Setuju (5) | * + 1. Sangat Setuju (1)     2. Setuju (2)     3. Netral (3)     4. Tidak Setuju (4)     5. Sangat Tidak Setuju (5) |

Skala Likert adalah metode skala bipolar, menentukan positif atau negatif respon pada sebuah pernyataan. Terkadang skala Likert digunakan pada metode pilihan yang mana pilihan tengah itu “netral”. Skala Likert mungkin menjadi subjek dari beberapa alasan. Responden mungkin menggunakan respon yang tergolong ekstrim, setuju dengan pernyataan yang diutarakan, atau berusaha untuk menggambarkan pola pikir individu atau kelompok dalam bentuk yang lebih nyata.

Untuk skor perhitungannya, sebagai penjelasannya adalah sebagai berikut:

* Responden yang menjawab sangat setuju (5) berjumlah n orang
* Responden yang menjawab setuju (4) berjumlah n orang
* Responden yang menjawab netral (3) berjumlah n orang
* Responden yang menjawab tidak setuju (2) berjumlah n orang
* Responden yang menjawab sangat tidak setuju (1) berjumlah n orang

Rumus:

S = n Pn

S = Nilai Skor / Pilihan Jawaban

n = jumlah responden yang memilih

Pn = Pilihan angka Skor Likert (1 – 5)

Dari perhitungan tersebut didapatkan total skor untuk masing-masing jawaban mulai dari sangat setuju hingga jawaban sangat tidak setuju. Kemudian dari masing-masing total skor tersebut dijumlahkan untuk mendapatkan jumlah total skor semua jawaban.

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 untuk jawaban sangat setuju adalah 5Tn Y. Jumlah nilai terendah untuk jawaban sangat tidak setuju adalah 1Tn X. Jika total skor yang didapat sebelumnya adalah (TS), maka interpretasi responden 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) |

1. 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.

1. Objek Penelitian

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

1. Instrumen Penelitian
   1. Komputer

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

* 1. *Software*

*Software* adalah sekumpulan data elektronik yang disimpan dan diatur oleh komputer berupa program aplikasi atau instruksi yang akan menjalankan suatu perintah. Dengan menggunakan program aplikasi ini, komputer dapat 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.

1. **GAMBARAN UMUM**

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 pengaruh dan hasil yang didapat dari penjualan *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.

- Tiket

- *Review* User

- Gallery Dokumentasi

- Data dan Laporan Transaksi

Output

- Beranda

- Login User

- Event

- Gallery

- Tentang

Menu

- Event

- Stock Tiket

- Dokumentasi

- Data User

- Komentar User

Input

- *Preview* Event

- Pesan Tiket

- Cetak Tiket

- Data User

- *Review* User

- Admin EO Side Story Family

- User (member)

- User (guest)

Fitur

Hak Akses

Sistem Penjualan *Online* Tiket Event

Gambar 2. Gambaran Umum Sistem

## Metode Penelitian

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

**Implementasi sistem**

**Desain sistem**

**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,

**Penetapan Fitur Preview**

**Mulai**

Gambar 3.1 Diagram alir penelitian

Sumber : Hasil analisis (2014)

### 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 kami 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 kami ajak bekerja sama. Dalam hal ini kami bekerja sama dengan salah satu EO yaitu side story family. Data yang kami ambil berupa profil usaha, pengalaman eo, seta data dokumentasi event.

1. Data Konser Musik

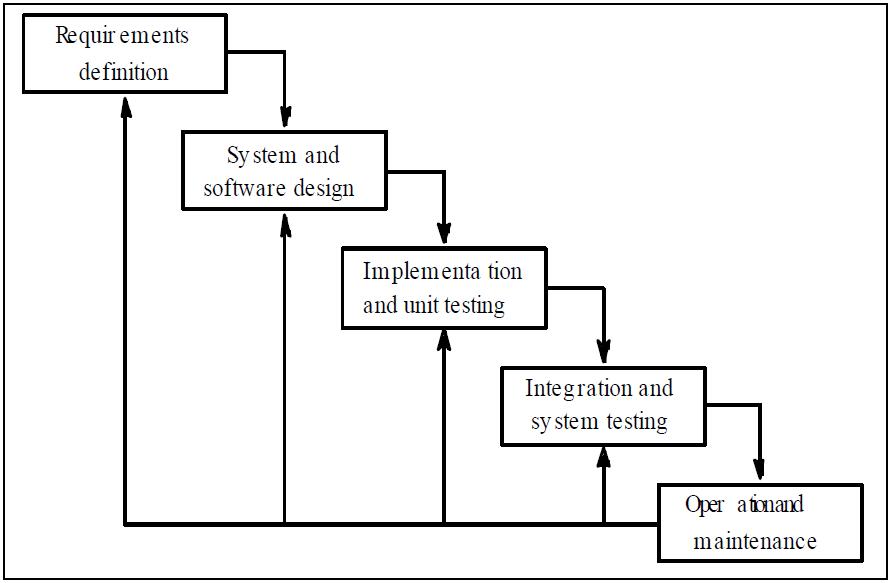
Merupakan data yang berisi tentang kegiatan konser musikyang tersediabaik di jember atau area sekitar yang sekiranya dapat dijadikan sebagai objek penelitian.

1. Kuisioner

Pengumpulan data menggunakan kuisioner untuk mendapatkan data nilai ketertarikan terhadap sebuah event. Kuisioner diberikan kepada user dari sistem yang kami rancang dan telah menggunakan fitur arau melaksanakan skenaria yang kami buat.

### 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.



#### Analisis Kebutuhan

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* 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 skenario 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.

#### Desain Sistem

Desain atau perancangan dalam membangun pemetaan kesesuaian lahan kakao menggunakan *Unified Modeling Language* (UML) yang dirancang menggunakan konsep *Object-Oriented Programming* (OOP). Dokumen desain sistem yang akan dibuat meliputi *Usecase Diagram, Usecase Skenario,sequence diagram, activity diagram, class Diagram* dan *Entity* *Relatioship* *Diagrams* (ERD).

#### Penulisan Kode Program

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 Yii.*

#### Pengujian

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).

#### Pemeliharaan

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

# DESAIN DAN PERANCANGAN

Bab ini akan menguraikan tentang proses desain dan perancangan sistem untuk mengimplementasi pemetaan kesesuaian lahan kakao di kabupaten Jember dengan metode TOPSIS. Proses desain dan perancangan sistem dimulai dari analisis kebutuhan fungsional dan non-fungsional sistem, dilanjutkan dengan pembuatan *usecase diagram,* skenario*, activity diagram, sequence diagram, class diagram dan entity relation diagram* (ERD).

## Studi Literatur

Studi literatur utama yang membantu dalam penelitian ini adalah Pedoman teknis budidaya tanaman kakao yang diterbitkan oleh Pusat Penelitian kopi dan kakao Indonesia.

## Pengumpulan Data

Pengumpulan data dilakukan agar pemetaan kesesuaian lahan kakao di kabupaten Jember sesuai dengan kebutuhan *user*. Data-data yang dikumpulkan adalah sebagai berikut :

1. Data EO

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

1. Data Konser Musik

Merupakan data yang berisi tentang kegiatan konser musikyang tersediabaik 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 kami rancang dan telah menggunakan fitur arau melaksanakan skenaria yang kami buat.

## Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak dalam penelitian ini yaitu dengan cara mengidentifikasi permasalahan yang ada untuk kemudian dicatat dan dijadikan bahan untuk mulai membangun aplikasi pemetaan kesesuaian lahan kakao dengan metode TOPSIS. Analisis kebutuhan yang dilakukan meliputi proses pengumpulan data kebutuhan fungsional, kebutuhan non-fungsional.

**Note: update tanggal 15 Desember 2016**

Kebutuhan fungsional sistem pada penelitian ini adalah sebagai berikut:

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

Kebutuhan non-fungsional sistem pada penelitian ini adalah :

* 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.

## Desain Sistem

Desain sistem adalah tahapan yang digunakan untuk memodelkan sistem yang akan dibuat. Desain sistem yang akan dibuat meliputi *Usecase Diagram, Usecase Skenario,sequence diagram, activity diagram, class Diagram* dan *Entity* *Relatioship* *Diagrams* (ERD). Desain dalam membangun Sistem pemetaan kesesuaian lahan ini lebih lanjut akan dijelaskan pada lampiran A.

### Usecase *Diagram*

*Usecase diagram* digunakan untuk menggambarkan fitur yang dibuat dalam sistem dan juga menggambarkan siapa saja yang dapat menggunakan fitur tersebut. *Usecase diagram* untuk sistem aplikasi tiket *online* ini dapat dilihat pada Gambar 4.1:



Gambar 4.1 *Usecase* sistem

Sumber : Hasil Analisis, 2016

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. | *Guest* | Aktor ini dapat menggunakan fitur yang ada pada sistem antara lain: melihat, memesan, mencetak, dan mereview event serta mengubah data *guest*. |

Tabel 4.2 Definisi *Usecase*

|  |  |  |
| --- | --- | --- |
| No | Usecase | Deskripsi |
|  | Mengelola event | * + - Merupakan *usecase* untuk mengelola event. |
|  | Melihat event | * + - Merupakan *usecase* untuk melihat event. |
|  | Memesan tiket | * + - Merupakan *usecase* untuk memesan tiket*.* |
|  | Mencetak tiket | * + - Merupakan *usecase* untuk mencetak tiket*.* |
|  | Mereview event | * + - Merupakan *usecase* untuk mereview event*.* |
|  | Mengelola data user | * + - Merupakan iusecase untuk mengelola data user |
|  | Mengubah data user | * + - Merupakan *usecase* untuk mengubah data user. |

### Fungsi 1: Mengelola Event

#### Skenario: Mengelola Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-001 | |
| Name *usecase* | Mengelola event | |
| Aktor | Admin | |
| Deskripsi | Merupakan *usecase* untuk mengelola 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 | - | |

#### Diagram Aktivitas: Mengelola Event



#### Diagram Sekuens: Mengelola Event



### Fungsi 2: Melihat Event

#### Skenario: Melihat Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-002 | |
| Name *usecase* | Melihat event | |
| Aktor | Admin, *guest* | |
| Deskripsi | Merupakan *usecase* untuk melihat event. | |
| Relasi | - | |
| Kondisi awal | Data event sudah ditambahkan ke dalam sistem | |
| Kondisi akhir | - | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin / *guest* memilih pilihan tampil daftar event | 1. Sistem menampilkan daftar event yang tersedia |
| Alur alternatif | - | |
| Kebutuhan khusus | - | |

#### Diagram Aktivitas: Melihat Event



#### Diagram Sekuens: Melihat Event



### Fungsi 3: Memesan Tiket

#### Skenario: Memesan Tiket

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-003 | |
| Name *usecase* | Memesan tiket | |
| Aktor | Admin, *guest* | |
| Deskripsi | Merupakan *usecase* untuk memesan tiket. | |
| Relasi | - | |
| Kondisi awal | - | |
| Kondisi akhir | Tiket event berhasil dipesan | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin, *guest* memilih pilihan pesan tiket 2. Admin, *guest* memilih event dengan menekan tombol pesan tiket 3. Admin, *guest* menekan tombol *reserve ticket* | 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 | |

#### Diagram Aktivitas: Memesan Tiket



#### Diagram Sekuens: Memesan Tiket



### Fungsi 4: Mencetak Tiket

#### Skenario: Mencetak Tiket

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-004 | |
| Name *usecase* | Mencetak tiket | |
| Aktor | *Guest* | |
| Deskripsi | Merupakan *usecase* untuk mencetak tiket. | |
| Relasi | - | |
| Kondisi awal |  | |
| Kondisi akhir |  | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. *Guest* memilih pilihan melihat tiket yang telah dipesan 2. Guest memilih pilihan cetak tiket | 1. Sistem menammpilkan list tiket yang sudah dipesan dan dilunasi user tersebut 2. Sistem mengubah tiket ke dalam format pdf |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus | Tiket sudah dipesan dan lunas. | |

#### Diagram Aktivitas: Mencetak Tiket



#### Diagram Sekuens: Mencetak Tiket



### Fungsi 5: Mereview Event

#### Skenario: Mereview Event

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-005 | |
| Name *usecase* | Mereview event | |
| Aktor | Admin, *guest* | |
| Deskripsi | Merupakan *usecase* untuk mereview event. | |
| Relasi | - | |
| Kondisi awal |  | |
| Kondisi akhir |  | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. Admin, *guest* memilih pilihan tampil daftar event 2. Admin, *guest*  menekan tombol pesan tiket 3. Admin, *guest* mengisi form review event 4. Admin, *guest* menekan tombol review | 1. Sistem menampilkan daftar event 2. Sistem menampilkan form review event 3. Sistem menampilkan review event yang baru ke dalam list event |
| Alur alternatif |  | |
|  |  |  |
| Kebutuhan khusus |  | |

#### Diagram Aktivitas: Mereview Event



#### Diagram Sekuens: Mereview Event



### Fungsi 6: Mengelola Data User

#### Skenario: Mengelola Data User

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-006 | |
| Name *usecase* | Mengelola data user | |
| Aktor | Admin, *guest* | |
| 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 |  | |

#### Diagram Aktivitas: Mengelola Data User



#### Diagram Sekuens: Mengelola Data User



### Fungsi 7: Mengubah Data User

#### Skenario: Mengubah Data User

|  |  |  |
| --- | --- | --- |
| Kode *usecase* | UC-007 | |
| Name *usecase* | Mengubah data user | |
| Aktor | *Guest* | |
| Deskripsi | Merupakan *usecase* untuk mengubah data user (*username* dan *password*). | |
| Relasi | - | |
| Kondisi awal |  | |
| Kondisi akhir |  | |
| Alur kejadian normal | Aktor | Sistem |
|  | 1. *Guest* memilih pilihan *user setting* 2. *Guest* memperbaharui data *username* dan *password* 3. *Guest* 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 |  | |

#### Diagram Aktivitas: Mengubah Data User



#### Diagram Sekuens: Mengubah Data User



### Koding

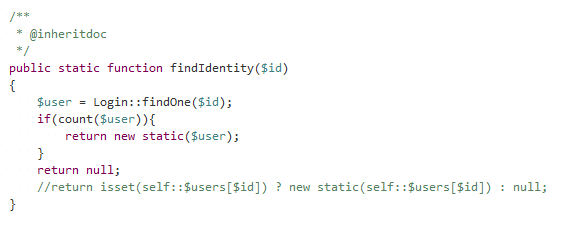
Pada tahap ini, desain selanjutnya akan diterjemahkan kedalam bentuk *coding* atau pengkodean menggunakan bahasa pemrograman yaitu *Page Hyper Text Pre-Process*(PHP), *Hyper Text Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan *Javascript.* Manajemen data menggunakan *database Mysql*. Dalam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

1. Login
   1. *Function Login* (Controllers/SiteController)



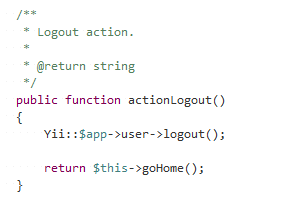
Gambar 4.10 *Function Login* (Controllers/SiteController)

1. *Function login* (models /user)



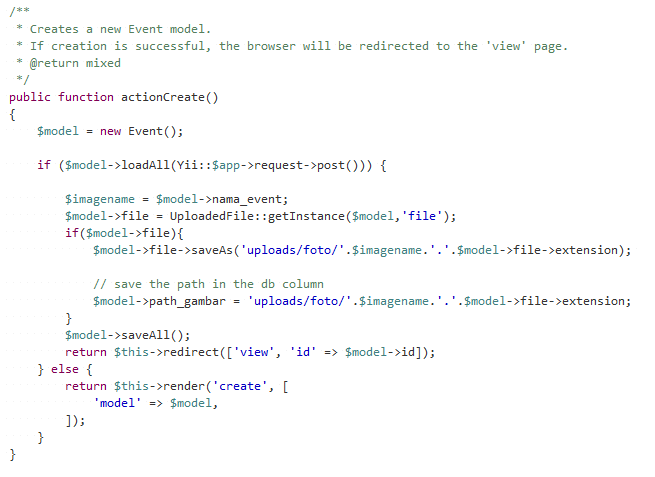
Gambar 4.12 *Function login* (models/User)

1. Logout
   * 1. Function logout (controllers/SiteController)

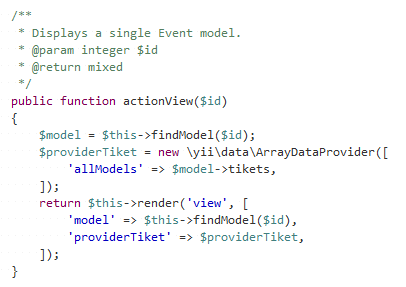


Gambar 4.13 Function logout (controllers/SiteController)

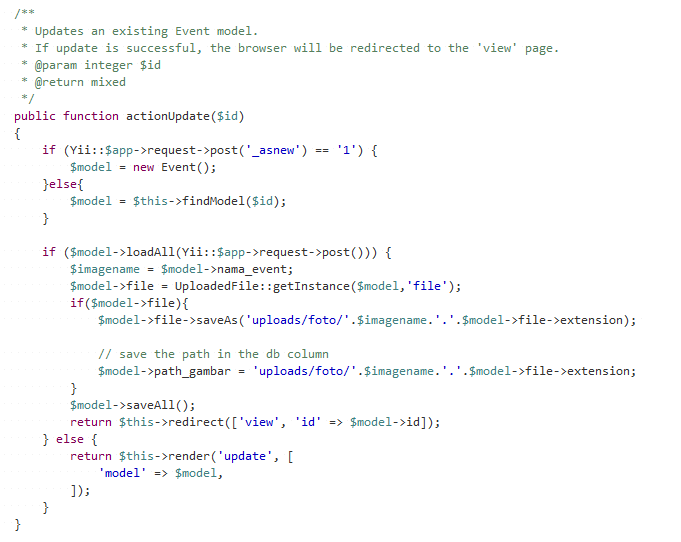
1. Event (controllers/EventController)
2. Admin - Membuat Event



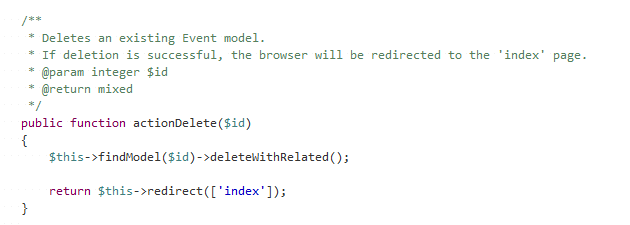
1. Admin – Melihat Event



1. Admin – Mengupdate Event



1. Admin Menghapus Event



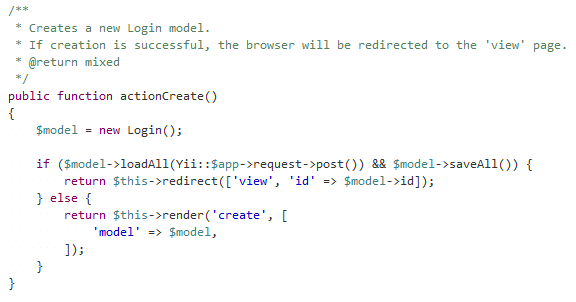
1. Tiket (controllers/TiketController)
   1. User Memesan Tiket



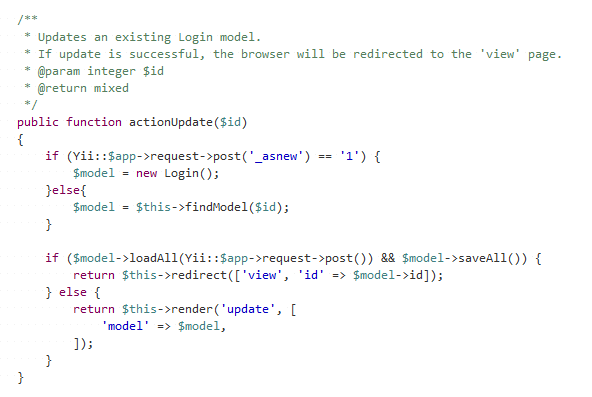
* 1. User Mencetak Tiket



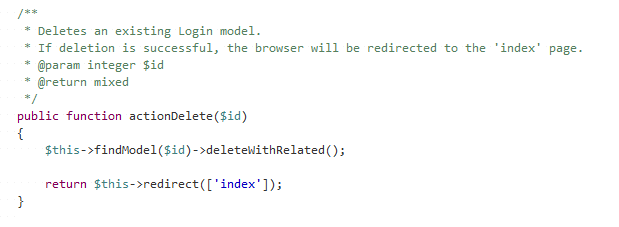
1. User (controller/UserController)
   1. Admin dan User Membuat



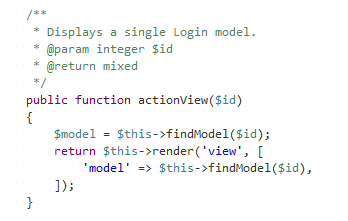
* 1. Admin dan User Mengupdate



* 1. Admin dan User Menghapus



* 1. Admin dan User Melihat



1. **IMPLEMENTASI**

Dari penelitian ini nantinya akan didapatkan pengembangan sistem penjualan *online* tiket event musik sehingga penjualan mampu mengatasi keterbatasan waktu dan tempat dengan fasilitas *e-marketing* dengan fitur *preview* dan beberapa fitur-fitur sistem yang dirancang oleh penulis. Solusi yang didapat selanjutnya akan diuji coba pada bagian administrasi EO Side Story dimana merupakan tempat dilakukannya studi kasus dari penelitian ini.

1. **LUARAN YANG DIHARAPKAN**

Luaran yang diharapkan dalam penelitian ini adalah:

1. Skripsi
2. Artikel Ilmiah
3. Aplikasi Web Sistem Penjualan Tiket *Online* Event Musik dengan fitur *preview*
4. **JADWAL KEGIATAN**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Kegiatan** | **Bulan** | | | | | |
| **Agt** | **Sep** | **Okt** | **Nov** | **Des** | **Jan** |
| 1. | Penyusunan dan Pengajuan Proposal |  |  |  |  |  |  |
| 2. | Seminar Proposal |  |  |  |  |  |  |
| 3. | Studi Literatur dan Observasi |  |  |  |  |  |  |
| 4. | Analisis dan Perancangan Sistem |  |  |  |  |  |  |
| 5. | Implementasi |  |  |  |  |  |  |
| 6. | Uji Coba dan Evaluasi |  |  |  |  |  |  |
| 7. | Penyusunan Laporan Skripsi |  |  |  |  |  |  |
| 8. | Perbaikan Skripsi |  |  |  |  |  |  |
| 9. | Sidang Skripsi |  |  |  |  |  |  |

1. **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.

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

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

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.

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

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

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