

PEMBANGUNAN APLIKASI TIKET DINAMIS BERBASIS WEBSITE

Zaky Muhammad Yopi Rusyana¹, Andri Heryandi²

^{1,2}Universitas Komputer Indonesia

Jl. Dipatiukur No. 112-116, Cobleng Bandung

E-mail : zakymuh123@gmail.com¹, andri@heryandi.net²

ABSTRAK

Tiket merupakan alat untuk tanda bukti pembayaran, pembelian maupun garansi dari sebuah transaksi. Tiket telah berkembang seiring dengan jalan menjadi tiket online. Banyak factor yang mempengaruhi seorang konsumen untuk membeli tiket secara online yaitu pengalaman antri yang lebih mudah dan kemudahan dalam pembelian. PT Inzpire Tech merupakan salah satu perusahaan IT yang menyediakan solusi yang unik dalam model bisnis, perusahaan ini bertempat di Bandung dan Jakarta. Perusahaan ini melayani permintaan dari client untuk pembuatan aplikasi dalam bentuk web ataupun mobile. Menurut hasil wawancara bahwa mereka mengalami kesulitan dalam pembuatan tiket dari berbagai client. Pembuatan tiket secara dinamis digunakan untuk menyelesaikan permasalahan yang terjadi. Seorang client nantinya hanya perlu melakukan usaha minimal untuk melakukan pembuatan template tiket dan dengan mudah membuat tiket yang dibuat secara khusus untuk mereka. Penerapan dari hasil pengujian terhadap sistem yang dibangun menggunakan black box dan pengujian beta mendapatkan kesimpulan bahwa sistem ini dapat membantu membuat, mengubah dan menghapus template tiket sesuai keinginan client. Sistem ini juga dapat terhubung dengan data dinamis yang dikirimkan dari website client dan lebih mudah pembuatannya dari prosedur sebelumnya.

Kata kunci : Tiket, Website, Khusus, Penerbangan, *Self-service*

1. PENDAHULUAN

1.1. Latar Belakang

Tiket merupakan alat untuk tanda bukti pembayaran, pembelian maupun garansi dari sebuah transaksi. Karena tiket telah berkembang seiring jaman, maka tiket online pun menjadi solusi dari tiket. Banyak factor yang mempengaruhi seorang konsumen membeli tiket perjalanan secara online yaitu pengalaman antri, kemudahan dalam pembelian, ketepatan penyampaian informasi dan efisiensi waktu dan biaya [1].

Penggunaan tiket yang tidak harus mengeluarkan dokumen secara fisik mempunyai banyak keuntungan untuk para penggunanya diantara lain adalah pengguna sistem tiket atau kita sebut saja E-ticketing ini tidak dibatasi oleh waktu dan tenaga. Biaya yang dibutuhkan untuk komunikasi antara pelanggan dan petugas pun dapat ditekan [2]

PT. Inzpire Tech merupakan salah satu perusahaan IT yang menyediakan solusi yang unik dalam bisnis model. Perusahaan ini bertempat di Bandung dan Jakarta. Banyak aplikasi yang telah dibangun oleh perusahaan ini diantaranya AerOne, AerOticker dan pocketmus. Selain itu perusahaan ini melayani permintaan dari client untuk pembuatan aplikasi dalam bentuk web ataupun mobile. Perusahaan ini pun menjalani bisnis dalam model bisnis to bisnis, dimana mereka menjual *engine* aplikasi yang digunakan pada beberapa *client*. Model aplikasi ini selain memudahkan dalam maintenance juga menjadikannya lebih ekonomis dari aplikasi yang bermodel khusus/*private* untuk client.

Menurut hasil wawancara dengan bapak Arief selaku Project Manager bahwa mereka mengalami kesulitan dalam pembuatan tiket dari permintaan berbagai client. Client sering meminta untuk penambahan design tiket baru serta perubahan design tiket. Client ingin dalam perusahaannya tersebut ada sebuah tiket yang membedakan antara tiket biasa dan tiket khusus. Tiket yang dibuat terkandung informasi-informasi hasil pengolahan data dalam sistem seperti pembelian dan pembelanjaan.

Berdasarkan hasil observasi terdapat sebuah website canva.com yang dapat menyediakan sebuah halaman website yang dapat mendesign serta membuat tiket secara online. Dari aplikasi tersebut tidak ditemukan adanya integrasi informasi dengan sistem, seperti data informasi yang dinamis berdasarkan kebutuhan dan bila ada perubahan dari data harus dilakukan adanya perubahan design.

Dari permasalahan yang telah dipaparkan di atas, maka dapat disimpulkan perlu adanya pembangunan sebuah perangkat lunak berbasis web yang dapat melakukan perubahan, pembuatan serta penambahan tiket secara *custom* oleh user dan dapat terintegrasi dengan informasi sistem yang telah ada. Dengan memanfaatkan teknologi seperti javascript serta API untuk pengelolaan design tiket dan API untuk pengiriman dan penerimaan data untuk

pengintegrasian antara sistem yang dibangun dengan sistem yang sedang berjalan.

1.2. Tujuan Penelitian

Tujuan yang akan dicapai dalam penelitian ini adalah:

- a. Membuat aplikasi berbasis website yang dapat melakukan perubahan, penambahan serta pembuatan tiket secara costum.
- b. Membuat aplikasi yang dapat terintegrasi dengan informasi pada sistem website utama.

1.3. Metodologi Penelitian

Metodologi penelitian merupakan suatu proses yang digunakan untuk memecahkan suatu masalah yang logis, dimana memerlukan data-data untuk mendukung terlaksananya suatu penelitian. Metode pengumpulan data yang digunakan adalah metode seperti berikut:

Metode Pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

a. Studi Literatur

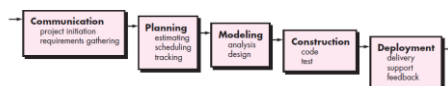
Pengumpulan data dengan cara mempelajari, meneliti dan menelaah berbagai literatur-literatur dari perpustakaan yang bersumber dari buku-buku, teks, jurnal ilmiah, situs-situs di internet dan bacaan-bacaan yang berkaitan dengan pembangunan aplikasi.

b. Wawancara(interview)

Tahap ini penulis melakukan observasi di tempat studi kasus yaitu PT.Inzpire untuk mengetahui kebutuhan sistem, karakteristik pengguna dan pendukung dalam pembangunan sistem.

1.4. Metode Pembangunan Perangkat Lunak

Metode dalam pembuatan perangkat lunak menggunakan paradigam perangkat lunak waterfall model dari R. Pressman. Yang meliputi beberapa tahapan sebagai berikut:



Gambar 1. 1 Waterfall Model

a. Communication

Menganalisa kebutuhan perangkat lunak dan pengumpulan data. Kebutuhan perangkat lunak diidentifikasi dengan cara melakukan kuesioner untuk mengumpulkan data-data yang dibutuhkan.

b. Planning

Menyusun rencana-rencana yang akan dilakukan selama pembangunan perangkat lunak meliputi pembuatan jadwal dari setiap tahap yang dilakukan dan mempersiapkan semua kebutuhan.

c. Modeling

Memodelkan setiap kebutuh perangkat lunak menggunakan *tools* diantaranya flowmap, ERD dan

UML, serta melakukan perancangan basis data, struktur menu, antarmuka, pesan dan procedural.

d. Construction

Penulisan kode dengan bahasa pemrograman dan MySql sebagai database management system (DBMS) berdasarkan hasil pemodelan dan kemudian dilakukan pengujian terhadap kode program yang dibuat menggunakan metode *black box testing*.

e. Deployment

Penyerahan perangkat lunak kepada *user* untuk mengetahui *feedback* dari program yang telah dibuat.

2. LANDASAN TEORI

2.1. Tiket

Tiket adalah dokumen berbentuk cetak, melalui proses elektronik atau bentuk lainnya yang merupakan bukti adanya perjanjian dari dua belah pihak. Pihak pemberi dan penerima. Digunakan sebagai bukti untuk mendapatkan pelayanan dan bukti pembayaran [3].

2.2. Website

Website atau bisa kita dibidang situs dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara dan atau gabungan dari semuanya, baik yang bersifat status maupun dinamis yang membentuk suatu rangkaian bangunan yang saling terkait. Rangkaian ini dihubungkan dengan jaringan-jaringan halaman.

Berikut merupakan jenis-jenis web berdasarkan sifat atau style-nya:

a. Website Dinamis

Website yang menyediakan content atau isi yang selalu berubah-ubah setiap saat. Bahasa pemrograman yang digunakan antara lain PHP, ASP, .NET dan memanfaatkan database MySQL atau MS SQL.

b. Website Statis

Content pada website statis jarang diubah, dengan menggunakan bahasa pemrograman seperti HTML dan belum memanfaatkan database seperti web profile dan web organisasi. [4]

2.3. HTML

HTML merupakan singkatan dari *Hyper Text Markup Language*. HTML merupakan bahasa pemrograman yang memiliki aturan atau sintak tertentu dalam menuliskannya. Sintak atau aturan itu berfungsi untuk menampilkan informasi pada browser dengan cara membaca kode-kode HTML. Tag-tag HTML selalu diawali dengan `<x>` dan diakhiri dengan `</x>` dimana x adalah tag HTML, tag tersebut seperti b, i, dan u [5].

Tabel 2. 1 Tag-tag HTML

Tag	Fungsi
	Membuat teks tebal
	Menampilkan image, imagemap atau animasi
<table>	Membuat tabel
Tag	Fungsi
<tr>	Membuat baris pada tabel
<p>	Membuat sebuah paragraf

2.4. PHP

PHP merupakan salah satu bahasa pemrograman berbasis web yang ditulis oleh dan untuk pengembangan web. PHP mempunyai keunggulan sebagai salah satu bahasa pemrograman berbasis web yaitu:

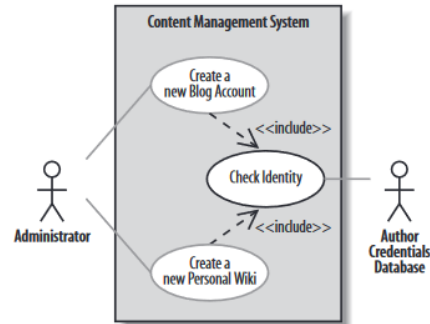
- Gratis. PHP dapat diunduh dan dipergunakan secara gratis. PHP dikembangkan oleh komunitas open source dan akan selalu didistribusikan secara gratis. Dari, oleh dan untuk pengembang.
- PHP berlisensi GNU *General Public License* (GPL). Hal tersebut menjadi jaminan bahwa semua versi PHP akan selalu didistribusikan secara gratis.
- Performa Handal. PHP sangatlah efisien. Dengan sebuah spesifikasi server yang tidak mahal, dapat melayani jutaan akses per hari.
- Dukungan Basisdata. PHP mendukung hampir semua perangkat basisdata, mulai dari MySQL, Oracle, PostgreSQL, Informix, Interbase, Sybase, MariaDB hingga SQLite.
- Pustaka Bawaan. PHP dibangun secara khusus untuk aplikasi berbasis web. Oleh karena itu, disediakan banyak sekali pustaka (*Library*) bawaan berkaitan dengan web yang dapat langsung digunakan.
- Cross Platform. PHP dapat dijalankan dengan baik pada hampir semua Sistem Operasi seperti Linux, Unix, Windows atau Mac Os.
- Mudah Dipelajari. Perintah-perintah PHP sangat mudah dipelajari, karena sebagian besar perintah PHP diadopsi dari bahasa pemrograman populer seperti Bahasa C/C++, Java dan Perl. [6]

2.5. UML

UML merupakan singkatan dari *Unified Modelling Language* yang secara alih bahasa artinya pemodelan standar. Bahasa pemodelan bisa apa saja yang mengandung notasi dan deskripsi. Namun ada beberapa keuntungan dan kerugian dalam penggunaan UML sebagai pemodelan salah satunya yaitu bahasa yang formal dan jelas, dengan begitu kita tidak perlu khawatir gambaran akan model kita akan disalah artikan. [7]

2.6. Use Case

Use Case menjelaskan fungsionalitas dari sistem yang akan dimodelkan dari sudut pandang luar. Use Case nantinya akan dibutuhkan guna menjelaskan apa yang nanti akan dilakukan. [7].



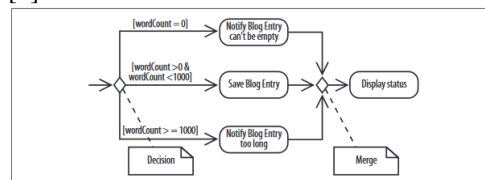
Gambar 2. 1 Use Case Diagram

Use case name	Create a new Personal Wiki												
Related Requirements	Requirement A.2.												
Goal In Context	A new or existing author requests a new personal Wiki from the Administrator.												
Preconditions	The author has appropriate proof of identity.												
Successful End Condition	A new personal Wiki is created for the author.												
Failed End Condition	The application for a new personal Wiki is rejected.												
Primary Actors	Administrator.												
Secondary Actors	Author Credentials Database.												
Trigger	The Administrator asks the CMS to create a new personal Wiki.												
Main Flow	<table border="1"> <thead> <tr> <th>Step</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>The Administrator asks the system to create a new personal Wiki.</td> </tr> <tr> <td>2</td> <td>The Administrator enters the author's details.</td> </tr> <tr> <td>3</td> <td>The author's details are verified using the Author Credentials Database.</td> </tr> <tr> <td>4</td> <td>The new personal Wiki is created.</td> </tr> <tr> <td>5</td> <td>A summary of the new personal Wiki's details are emailed to the author.</td> </tr> </tbody> </table>	Step	Action	1	The Administrator asks the system to create a new personal Wiki.	2	The Administrator enters the author's details.	3	The author's details are verified using the Author Credentials Database.	4	The new personal Wiki is created.	5	A summary of the new personal Wiki's details are emailed to the author.
Step	Action												
1	The Administrator asks the system to create a new personal Wiki.												
2	The Administrator enters the author's details.												
3	The author's details are verified using the Author Credentials Database.												
4	The new personal Wiki is created.												
5	A summary of the new personal Wiki's details are emailed to the author.												
Extensions	<table border="1"> <thead> <tr> <th>Step</th> <th>Branching Action</th> </tr> </thead> <tbody> <tr> <td>3.1</td> <td>The Author Credentials Database does not verify the author's details.</td> </tr> <tr> <td>3.2</td> <td>The author's new personal Wiki application is rejected.</td> </tr> </tbody> </table>	Step	Branching Action	3.1	The Author Credentials Database does not verify the author's details.	3.2	The author's new personal Wiki application is rejected.						
Step	Branching Action												
3.1	The Author Credentials Database does not verify the author's details.												
3.2	The author's new personal Wiki application is rejected.												

Gambar 2. 2 Use Case Scenario

2.7. Activity Diagram

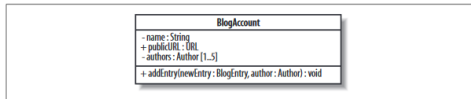
Activity Diagram digunakan untuk menjelaskan lebih detail lagi bagaimana sistem akan mengerjakan tujuan-tujuannya. *Activity Diagram* memperlihatkan aksi yang saling berhubungan satu sama lain untuk menunjukkan proses yang ada dalam sebuah sistem. [7]



Gambar 2. 3 Activity Diagram

2.8. Class Diagram

Class diagram merupakan blueprint dari object yang akan kita buat. Seperti contohnya mobil, mobil memiliki ribuan jenis namun class mobil hanyalah satu, beroda empat dan mempunyai mesin. Sebuah class memiliki details tentang hal penting dalam model dan sistem. [7]



Gambar 2. 4 Class Diagram

2.9. CSS

CSS atau Cascading Style Sheet secara sederhana adalah sebuah metode yang digunakan untuk mempersingkat penulisan tag HTML seperti font, color, text dan tabel menjadi lebih singkat. CSS digunakan untuk mengatur tampilan dokumen. CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda. CSS mempunyai beberapa kegunaan dan manfaat antara lain:

- Mempersingkat penulisan tag HTML. Tidak perlu lagi mendefinisikan setiap tag dengan property dan nilai yang sama.
- Mempercepat proses rendering atau pembacaan HTML karena tidak terdapat pengulangan penulisan.
- Mudah dan Cepat dalam maintenance dikarenakan file CSS yang dibuat secara terpisah. Apabila ingin melakukan perubahan cukup merubah satu file tanpa merombah seluruh file HTML.
- CSS bisa melakukan apa yang tidak bisa dilakukan oleh HTML, misalnya kita bisa memberikan warna pada input box atau scrollbar.

Penulisan script CSS dibagi menjadi tiga bagian yaitu:

a. Inline Style Sheet

Penulisan sintak CSS di dalam elemen HTML. Cara penulisannya sama seperti kita menuliskan property pada tag HTML. Misalkan kita ingin mengubah warna teks pada tag <p>, maka cara penulisannya adalah sebagai berikut:

```
<p style="color: blue">
```

b. Embedded Style Sheet

Penulisan sintaks CSS di dalam document HTML dengan metode embedded style sheet menggunakan tag <style> dan </style>.

```
<style type='text/css'>
h1{color: red}
</style>
```

c. Linked Style Sheet

Penulisan skrip CSS di halaman yang berbeda atau terpisah dari HTML. Penggunaanya dengan melakukan link ke file css yang telah dibuat. Untuk metode ini menggunakan tag <link rel> yang ditempelkan pada bagian tag <HEAD> pada HTML [8].

2.10. JAVASCRIPT

Javascript merupakan bahasa skrip populer yang dipakai untuk menciptakan halaman web yang berinteraksi dengan pengguna dan dapat merespon

event yang terjadi pada halaman. Javascript merupakan perekat yang menyatukan halaman-halaman web.

Javascript terkait dengan browser, ia sangat terintegrasi dengan HTML. Ketika browser memuat sebuah halaman, server akan mengirimkan konteks utuh dari dokumen, termasuk HTML dan statement-statement Javascript. Konten HTML kemudian dibaca dan diinterpretasi baris-demi-baris sampai tag pembuka javascript dibaca, pada saat itu interpreter Javascript mengambil alih. Ketika tag penutup javascript diraih, pemrosesan HTML berlanjut.

Program javascript dipakai untuk mendeteksi dan beraksi terhadap event-event yang disebabkan oleh pengguna. Dengan ini kita dapat memperbaiki situs web dengan bantuan navigasional, kotak dialog dan citra dinamis. Javascript digunakan untuk mengendalikan tampilan halaman. Javascript data menghasilkan halaman web yang terstruktur, bergaya dan interaktif. [9]

2.11. Framework

Framework adalah kumpulan perintah atau fungsi dasar yang membentuk aturan-aturan tertentu dan saling berinteraksi satu sama lain sehingga dalam pembuatan aplikasi website, kita harus mengikuti aturan dari framework. Dengan Framework, kita tidak perlu memikirkan kode perintah/fungsi dasar aplikasi website kita. Seperti bagaimana mengambil data dari database untuk ditampilkan. Kita hanya perlu memikirkan kode perintah/fungsi dasar dari aplikasi website kita. [10]

3. ANALISIS

3.1. Analisis Prosedur yang sedang berjalan

Analisis prosedur yang sedang berjalan merupakan uraian secara sistematis aktivitas-aktivitas yang terjadi dalam pembuatan template tiket di perusahaan penyedia. Adapun prosedur dari analisis prosedur pembuatan template tiket yang sedang berjalan pada perusahaan penyedia adalah sebagai berikut:

- Client membuat rancangan template tiket yang akan dibuat.
- Client memberikan rancangannya kepada perusahaan penyedia
- Perusahaan penyedia memasukan rancangan pada daftar pembuatan template
- Perusahaan penyedia melakukan pembuatan template tiket
- Perusahaan penyedia menyampaikan informasi bahwa template berhasil dibuat dan siap digunakan.

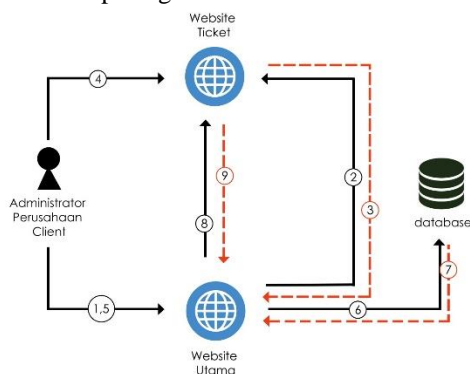
Analisis sistem yang dibangun merupakan gambaran lengkap sistem yang diusulkan. Analisis sistem yang diusulkan ini juga didasarkan pada analisis masalah, analisis sistem yang sedang berjalan dan analisis sistem sejenis. Sehingga dari ketiga

analisis tersebut akan didapatkan gambaran aplikasi yang akan diusulkan berikut dengan prosedur aplikasi yang akan diusulkan. Pada penelitian ini, aplikasi yang akan diusulkan harus memenuhi kebutuhan dan solusi dari masalah yaitu pembuatan template tiket secara *costum*.

Adapun prosedur dari analisis sistem yang diusulkan yaitu:

- a. Client membuka halaman website yang dibuat khusus untuk pembuatan template tiket.
- b. Client melakukan pembuatan template sesuai keinginannya.
- c. Client menyimpan template yang telah dibuat.
- d. Client memilih template yang akan digunakan.

Tahap analisis ini bertujuan untuk mendapatkan gambaran umum sistem yang akan dibangun. Baik itu gambaran pengguna hingga transaksi data yang akan dibangun. Gambaran arsitektur sistem dari aplikasi pembangun dokumentasi tiket dinamis berbasis web dapat dilihat pada gambar 3.10.



Gambar 3. 1 Arsitektur sistem yang akan dibangun.

Berikut merupakan penjelasan dari gambar 3.10 aplikasi dokumentasi tiket dinamis berbasis website:

- a. Administrator Perusahaan Client melakukan aktivasi pada website utama.
- b. Website utama melakukan aktivasi ke dalam sistem dengan mendaftarkan diri dan meminta username & password
- c. Website tiket memberikan respon bahwa akun telah diaktifkan dan dapat melakukan pembuatan tiket.
- d. Administrator Perusahaan Client masuk dan melakukan pembuatan template tiket
- e. Administrator Perusahaan Client melakukan cetak perdokumen yang akan dicetak.
- f. Website Utama meminta data transaksi mengenai data yang akan dicetak ke database.
- g. Database memberikan data yang diminta Website Utama.
- h. Website Utama melakukan pengiriman data ticket ke website tiket untuk bahan penggabungan tiket antara data tiket dan template tiket.
- i. Website tiket merespon dengan memberikan tiket yang telah digabungkan dengan template tiket.

Analisis pembuatan template digunakan untuk mengetahui bagaimana pengguna dari sistem yang

akan dibangun membuat template. Analisis ini dilakukan dengan melihat hasil analisis dari analisis dokumen dan analisis aplikasi sejenis. Untuk memenuhi kebutuhan dari data dinamis juga pembuatan template yang mudah maka dipilihlah metode *drag and drop*.

Drag and drop merupakan sebuah metode untuk memindahkan sebuah objek dari satu posisi ke posisi yang lain dengan men-*drag* objek dan men-*drop* objek ke posisi yang dituju. Pengguna nantinya diberikan banyak pilihan objek-objek yang berguna untuk melakukan pembuatan template.

Objek-objek tersebut antara lain adalah teks, gambar serta tabel. Teks yang disediakan yaitu teks dengan ukuran font yang beragam antara lain ukuran 24px, 18px dan 12px. Gambar didapatkan dari data pengguna yang terdapat dari sistem utama. Gambar ini didapatkan saat pertama pengguna melakukan pendaftaran pada sistem tiket. Gambar dapat dimodifikasi, diupload serta dihapus. Tabel berguna untuk memasukan data dinamis yang nantinya akan dikombinasikan saat pengguna menekan tombol cetak tiket pada sistem utama. Tabel mempunyai judul tabel dan kolom tabel, sementara baris pada tabel akan menyesuaikan.

Objek-objek tersebut diciptakan melalui tag-tag HTML. Tag-tag html yang dipergunakan untuk membuat objek-objek yaitu:

- a. Tag

Tag digunakan untuk pengenalan suatu objek teks. Pemilihan tag dibandingkan dengan objek teks lainnya karena tag ini dapat disesuaikan posisinya saat pembuatan template. Berikut adalah tag span yang digunakan.

```
<span style="font-size: 36px; white-space: nowrap;">Add header type 1</span>
```

- b. Tag <table>

Tag <table> digunakan untuk pengenalan objek tabel. Tabel ini akan mempunyai beberapa kolom dan satu buah nama tabel. Berikut adalah tag tabel yang digunakan.

```
<table class="draggable ui-widget-content ui-draggable ui-draggable-handle" style="position: relative; width: 675px; inset: 40px auto auto 47px; height: 58px; width="85%" border="1"><tbody><tr><th style="border:0;" colspan="2">bambang</th></tr><tr><td class="tdclick" style="padding: 3px;font-size:13px;">1</td><td class="tdclick" style="padding: 3px;font-size:13px;">2</td></tr></tbody></table>
```

- c. Tag

Tag digunakan untuk pengenalan objek gambar. Objek gambar tersebut dapat secara langsung diinputkan sesuai dengan keinginan client. Berikut merupakan tag image yang digunakan.


```
curl -s \
-X POST \
https://eztiket.web.id/API_TIKET/generate \
-H 'Content-Type: application/json' \
-d '{
  "agency": "Nama Agency",
  "agency_contact": "kontak Agency",
  "issuedby": "Nama pembuat",
  "secret": "Kode Secret",
  "issueddate": "waktu Pembuatan",
  "transaction_detail": [
    "book_id": "Kode Book",
  ],
  "flight_schedule": [
    "flight_no": "Nomer penerbangan",
  ],
  "flight_passenger": [
    "flight_passenger": "Data Penumpang"
  ],
}'
```

Gambar 3. 3 Panduan API yang dikirimkan dari sistem luar tiket untuk generate tiket

Setelah sistem tiket menerima data dinamis melalui API. Sistem tiket mengganti teks menjadi data dinamis. Dengan kode PHP teks tersebut diganti melalui data masukan.

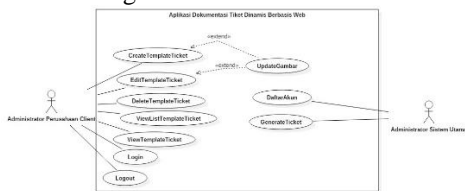
```
str_replace("&lt;&lt; &gt;&gt;",
$agency, $content);
```

Semua data akan digantikan berdasarkan kode penanda yang telah dimasukan. Setelah semua proses telah dilaksanakan maka sistem tiket akan melemparkan kembali data template tiket yang telah dikombinasikan. Data tersebut data dipergunakan untuk melihat tiket, mencetak tiket serta mengirimkannya via email.

```
"nama_tiket" : "Nama Tiket",
"ukuran" : "Ukuran Tiket",
"content" : "File siap cetak",
}
```

Gambar 3. 4 Hasil Generate Tiket

Use Case Diagram akan memperlihatkan hubungan antara aktor dan use case. Aktor merepresentasikan seorang user atau subsistem lain yang akan berinteraksi dengan sistem. Use case merupakan kejadian yang menggambarkan interaksi antara user dengan sistem.

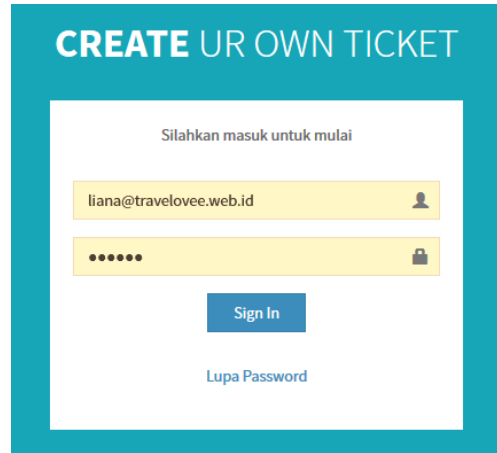


Gambar 3. 5 Use Case Diagram Dokumentasi Tiket Dinamis

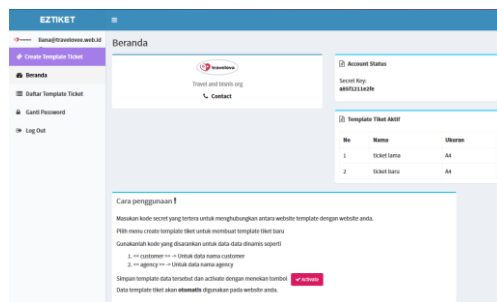
4. IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi

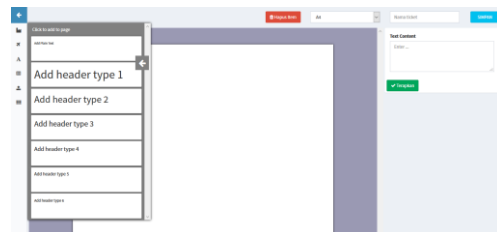
Tampilan program berguna untuk implementasi dari perancangan antarmuka, berikut merupakan tampilan dari program yang telah dibuat.



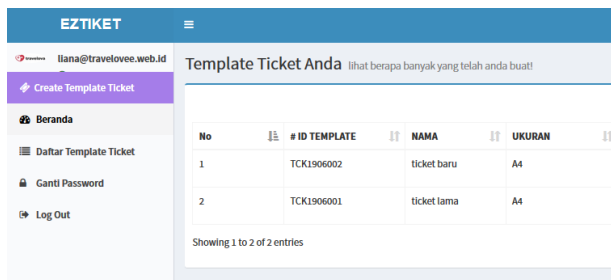
Gambar 4. 1 Tampilan Login



Gambar 4. 2 Tampilan Halaman Beranda



Gambar 4. 3 Tampilan halaman create template, edit template dan lihat template



Gambar 4. 4 Tampilan halaman daftar template

4.2. Pengujian

Blackbox adalah pengujian yang digunakan untuk menguji sistem yang baru. Pengujian ini berfokus pada persyaratan fungsional perangkat lunak. Rencana pengujian yang akan dilakukan dengan menguji sistem secara alpha dan beta. Pengujian Aplikasi tiket dinamis ini menggunakan data uji berdasarkan data asset yang ada pada PT. Inzpire.

4.3. Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus sample uji yang telah dilakukan. Berdasarkan hasil pengujian ini maka dapat ditarik kesimpulan bahwa perangkat lunak ini telah sesuai dengan harapan dan kebutuhan user serta bebas error sehingga layak dipergunakan.

5. Kesimpulan dan saran

5.1. Kesimpulan

Berdasarkan hasil analisis dan implementasi yang didapat setelah melakukan penelitian dalam pengembangan “Pembangunan aplikasi tiket dinamis berbasis website”. Maka diperoleh kesimpulan dari tujuan pengembangan aplikasi yang telah dianalisis dan diuji sehingga didapatkan hasil sebagai berikut:

- Website template tiket memudahkan dalam pembuatan tiket secara custom/khusus.
- Website template tiket yang dibuat telah sesuai dengan data dinamis/data asli pada website yang menggunakannya.
- Dengan adanya aplikasi ini lebih memudahkan pembuatan template dari pembuatan sebelumnya.

5.2. Saran

Aplikasi tiket dinamis berbasis website yang dibangun ini masih banyak kekurangannya. Oleh

karena itu ada beberapa saran yang dapat digunakan sebagai panduan pengembangan perangkat lunak kearah yang lebih baik guna mendukung pertumbuhan pengguna dan konten pada perangkat lunak ini. Adapun saran yang didapatkan adalah:

- Penambahan fitur-fitur untuk melakukan perubahan-perubahan objek serta penambahan objek seperti garis dan penambahan icon-icon.
- Penambahan hint-hint untuk “how to use” saat melakukan pembuatan template tiket

6. DAFTAR PUSTAKA

- [1] I. P. H. H. Duari, Tiket Penerbangan Domestik, Deepublish, 2018.
- [2] E. B. Setiawan, “Analisis Pengaruh Nilai Teknologi Informasi Terhadap Keunggulan Bersaing Perusahaan(Studi Kasus Pemanfaatan E-Ticketing Terhadap Loyalitas Pengguna Jasa Kereta Api),” *Jurnal Sains, Teknologi Dan Industri*, vol. 12, no. 2, p. 206, 2015.
- [3] A. Juansyah, “Pembangunan Aplikasi Child Tracker Berbasis Assite - Global Positioning System (A-GPS) dengan Platform Android,” *Jurnal Ilmiah Komputer dan Informatika*, vol. 1, p. 2, 2015.
- [4] R. Hidayat, Cara Praktis Membangun Website Gratis, Jakarta: Elex Media Komputindo, 2010.
- [5] Anhar, PHP & MySQL secara otodidak, Jakarta: MediaKita, 2010.
- [6] A. Solichin, Pemrograman Web dengan PHP dan MySQL, Budi Luhur, 2016.
- [7] R. M. & K. Hamilton, Learning UML 2.0, Sebastopol: O'Reilly Media, Inc., 2006.
- [8] D. Juju, Kupas Tuntas CSS, Jakarta: Elex Media Komputindo, 2007.
- [9] R. Sianipar and M. Ridwan, Javascript Untuk Professional, Sparta Publisher, 2018.
- [10] S. M. Wardana, Menjadi master PHP dengan Framework CodeIgniter, Jakarta: Elex Meida Komputindo, 2010.