

PEMBANGUNAN APLIKASI *E-TICKETING* MENGUNAKAN TEKNOLOGI QR CODE BERBASIS WEB DAN AKSES MOBILE PADA BRT TRANS PADANG

Ari Bespriadi ¹⁾, Alizar Hasan ²⁾, Ricky Akbar ³⁾

^{1,3)} Jurusan Sistem Informasi Fakultas Teknologi Informasi Universitas Andalas

Kampus Universitas Andalas Limau Manis Padang

e-mail : ari.bespriadi@gmail.com¹⁾, rickyakbar@fti.unand.ac.id³⁾

²⁾ Jurusan Teknik Industri Fakultas Teknik Universitas Andalas

Kampus Universitas Andalas Limau Manis Padang

e-mail: alizarhasan@ft.unand.ac.id²⁾

ABSTRACT

E-ticketing is a method to manage the sales process of the activities of the customer trips without removing the physical document. Currently ticketing transaction process at UPT Trans Padang was still using a conventional ticket and manual recording in business processes that are affecting speed of service. Therefore, in order to solving these problems, it takes an application for managing the ticketing process that occurs in UPT Trans Padang. This application is built by using the QR Code technology in its implementation and equipped with mobile features so that passengers can view their account information wherever and whenever they need. Development of application is using the waterfall method through phase analysis, design, and implementation and system testing. In the analysis system, it is used Business Process Model Notation, Use Case Diagrams, Use Case Scenarios, Sequence Diagrams and Class Analysis. The process of designing was creating with an Entity Relationship Diagram, Application Architecture, and Class Diagrams. In the implementation the phase is describing Limitation Implementation Specifications Software and Hardware, Encoding, and Implementation Interface. The process of application testing was conducted using black box testing. The test results showed applications output are in accordance with the functional requirement

Key words

E-Ticketing, QR Code, Mobile, Trans Padang

1. Pendahuluan

Pada awal tahun 2014, Dishubkominfo kota padang meresmikan 10 unit BRT Trans Padang sebagai sarana transportasi baru yang lebih nyaman dan handal. BRT (Bus Rapid Transit) atau *busway* merupakan bus dengan kualitas tinggi yang berbasis sistem transit yang cepat,

nyaman, dan biaya murah untuk mobilitas perkotaan dengan menyediakan jalan untuk pejalan kaki, infrastrukturnya, operasi pelayanan yang cepat dan sering, perbedaan dan keunggulan pemasaran dan layanan kepada pelanggan [1].

Mengingat BRT Trans Padang ini baru beroperasi dan belum memiliki sistem informasi dan aplikasi serta pencatatan rekap penjualan tiket yang masih manual, hal ini tentu sangat rawan mengingat banyaknya resiko yang dapat ditimbulkan seperti duplikasi data, kehilangan data, data yang tidak konsisten, sulit dipantau, *human error*, dan lain-lain. Dalam implementasinya juga ditemukan beberapa kendala, seperti penumpang yang membayar namun tidak mendapatkan tiket, uang kembalian yang tidak diberikan oleh pramugara, bahkan pramugara Trans Padang yang kecolongan meminta ongkos ke penumpang ketika penumpang bus *over capacity* [2].

Untuk mengatasi permasalahan tersebut, maka perlu dibangun sebuah aplikasi e-ticketing untuk melakukan pencatatan setiap transaksi yang terjadi. Untuk menunjang sistem agar mampu melakukan pencatatan lebih cepat dan akurat, penulis memilih untuk menggunakan metode *scanning* menggunakan teknologi QR Code karena memiliki beberapa kelebihan, seperti ketahanan terhadap kerusakan dan kotoran, serta mudah untuk diimplementasikan.

Melihat perkembangan teknologi yang terjadi dewasa ini, pertumbuhan pengguna *smartphone* di Indonesia semakin meningkat pesat, berdasarkan hasil riset yang dilakukan oleh Yahoo! Dan Mindshare saat ini ada sekitar 41,3 juta pengguna *smartphone* di Indonesia [3]. *Smartphone* menjadi pilihan bagi para pengguna karena mobilitas yang dimilikinya membuat para pengguna mampu mendapatkan info yang tepat dan akurat dimanapun dan kapanpun. Melihat pesatnya perkembangan *smartphone* ini dapat menjadi pertimbangan untuk mengembangkan aplikasi tidak hanya berbasis *web* namun juga memiliki fitur *mobile*.

2. Tinjauan Pustaka

Untuk membangun sebuah sistem informasi dan aplikasi yang efektif serta sesuai dengan tujuannya, maka perlu adanya kajian yang mendasari pembangunan dan penyelesaian penelitian. Beberapa hal yang menjadi tinjauan pustaka adalah mengenai sistem informasi, media yang digunakan dalam membangun sistem, serta hal-hal yang terkait dengan BRT dan e-ticketing.

2.1 Bus Rapid Transit

Bus Rapid Transit atau lebih sering disingkat menjadi BRT adalah sebuah sistem transportasi berbasis bus yang beroperasi dalam suatu koridor dengan memanfaatkan salah satu jalur pada jalan utama sebagai jalur khususnya, yang tidak mengizinkan kendaraan lain memasuki jalur tersebut [4].

2.2 E-Ticketing

E-Ticketing atau *electronic ticketing* adalah suatu cara untuk mendokumentasikan proses penjualan dari aktifitas perjalanan pelanggan tanpa harus mengeluarkan dokumen berharga secara fisik ataupun *paper ticket* [5]. Dalam implementasinya, *e-ticketing* untuk BRT Trans Padang akan menggunakan sistem *deposit money*, dimana pengguna mengisi saldo kartu *e-ticketing* nya yang digunakan dalam pembayaran tiket untuk sekali jalan.

2.3 QR Code

QR Code (*Quick Response Code*) merupakan kode batang (barcode) 2 dimensi yang dikembangkan oleh Denso Wave, yaitu sebuah divisi dari Denso Cooperation yang merupakan sebuah perusahaan Jepang dan dipublikasikan pada tahun 1994 dengan fungsionalitas utama yaitu dapat dibaca dengan mudah oleh pemindai (*scanner*) [6]. Di dalam *website* resminya, Denso Wave Incorporated (2014) mengklaim QR Code mampu menangani berbagai macam tipe data seperti *numeric*, alfabet, kanji, kana, hiragana, symbol, biner, dan *control code* dan mampu meng-encode hingga 7.098 *character* dalam satu symbol, selain itu QR Code juga diklaim tahan terhadap kotoran, dan kerusakan. QR Code dapat digunakan pada komputer yang memiliki *webcam* atau *smartphone* yang memiliki aplikasi *scanner* QR Code, arahkan gambar QR Code tersebut pada kamera *smartphone* atau kamera *webcam*, selanjutnya program akan memindai data yang tertanam pada QR Code dan menerjemahkannya. Contoh dari QR Code dapat dilihat pada gambar 1



Gambar 1. Contoh QR Code

3. Metode Penelitian

Dalam penelitian dan pembangunan aplikasi *e-ticketing* menggunakan teknologi QR Code berbasis *web* dengan fitur *mobile* pada BRT Trans Padang, perlu adanya metode yang digunakan dalam pengumpulan data dan metode yang digunakan dalam pembangunan aplikasi.

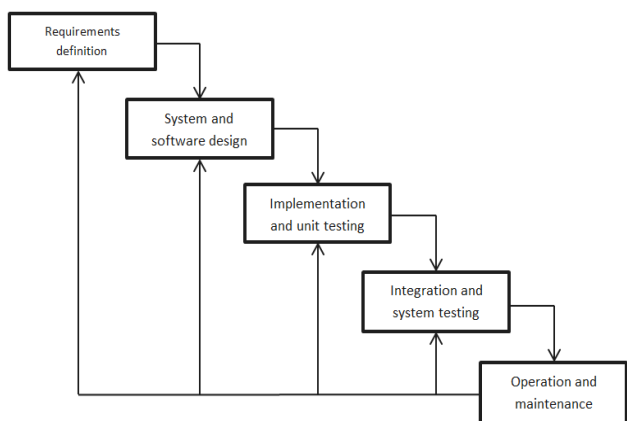
3.1 Metode Pengumpulan Data

Metode-metode yang digunakan dalam mengumpulkan data untuk membangun aplikasi ini adalah:

1. Wawancara
Pengumpulan data dilakukan dengan cara tanya jawab dengan pihak dari UPT Trans Padang mengenai hal-hal yang berkaitan dengan proses *ticketing* di Trans Padang
2. Studi Literatur
Penelusuran literatur dilakukan dengan cara mengumpulkan referensi pendukung seperti jurnal-jurnal *online*, penelitian yang telah dilakukan, sumber internet, dan buku yang berkaitan dengan pembangunan aplikasi *e-ticketing*.

3.2 Metode Pengembangan Sistem

Metode yang digunakan dalam untuk membangun aplikasi ini adalah metode *waterfall*. Pengembangan dengan metode ini harus dilakukan secara berurutan dimana output dari tahapan sebelumnya merupakan input untuk tahap selanjutnya. Dalam pembangunan aplikasi *e-ticketing* ini, tahapan yang dilalui hanya sampai pada tahap ketiga yaitu pada tahap *Implementation and Unit Testing*. Model *waterfall* dapat dilihat pada gambar 2. Menurut Sommerville [7], tahap-tahap utama dari *waterfall* model adalah memetakan kegiatan-kegiatan pengembangan dasar, yaitu:



Gambar 2. Waterfall Model (Sommerville 2003)

1. Requirement Analysis and Definition

Pada tahapan ini, dilakukan analisa dan pengumpulan data sebanyak-banyaknya terhadap kebutuhan sistem dan aplikasi dengan cara wawancara dan studi literatur sehingga akan tercipta sebuah aplikasi yang mampu memenuhi kebutuhan penggunaanya.

2. System and Software Design

Tahapan ini berfokus pada struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural dan akan menghasilkan sebuah software requirement yang dijadikan acuan oleh programmer untuk melakukan aktivitas pembuatan aplikasinya

3. Implementation and Unit Testing

Dari software requirement yang dihasilkan pada tahap perancangan, maka software requirement tersebut diterjemahkan ke dalam bahasa pemrograman tertentu. Setelah coding selesai, maka akan dilakukan testing untuk menemukan bug/kesalahan pada program dan kemudian diperbaiki.

4. Analisa Sistem

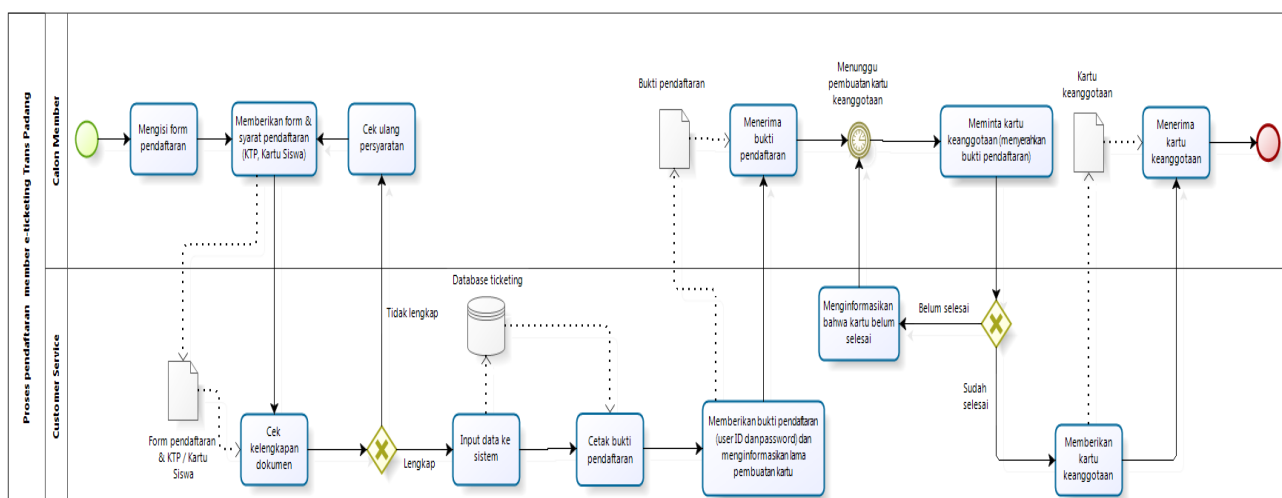
Di dalam bab ini akan dijelaskan analisis dari kebutuhan sistem dan aplikasi berdasarkan hasil pengumpulan data di lapangan serta rancangan sistem dari hasil analisa data yang didapatkan. Proses analisa sistem dengan Business Process Modelling Notation (BPMN), use case, dan sequence diagram.

4.1 BPMN Pendaftaran member yang diusulkan

Proses ini menjelaskan bagaimana tahapan dalam melakukan pendaftaran keanggotaan e-ticketing Trans Padang, dimana yang melakukan input data adalah customer service. Tahapan yang dilakukan dalam proses pendaftaran keanggotaan adalah sebagai berikut:

1. Calon anggota mengisi formulir yang telah disediakan oleh pihak Trans Padang
2. Calon anggota menyerahkan formulir beserta syarat pendaftaran lainnya (KTP/Kartu Siswa)
3. Customer service melakukan pengecekan syarat-syarat yang diberikan, jika sesuai maka akan dilakukan input ke dalam aplikasi e-ticketing Trans Padang
4. Customer service memberikan bukti pendaftaran dalam secarik kertas berisi data-data yang dibutuhkan seperti username dan password dan menginformasikan lama pembuatan kartu
5. Calon anggota mendatangi UPT Trans Padang untuk menerima kartu anggota sesuai waktu yang telah ditentukan dengan memberikan bukti pendaftaran yang telah diberikan sebelumnya.

Berdasarkan prosedur pendaftaran keanggotaan diatas, maka BPMN untuk proses pendaftaran keanggotaan dapat dilihat pada gambar 3

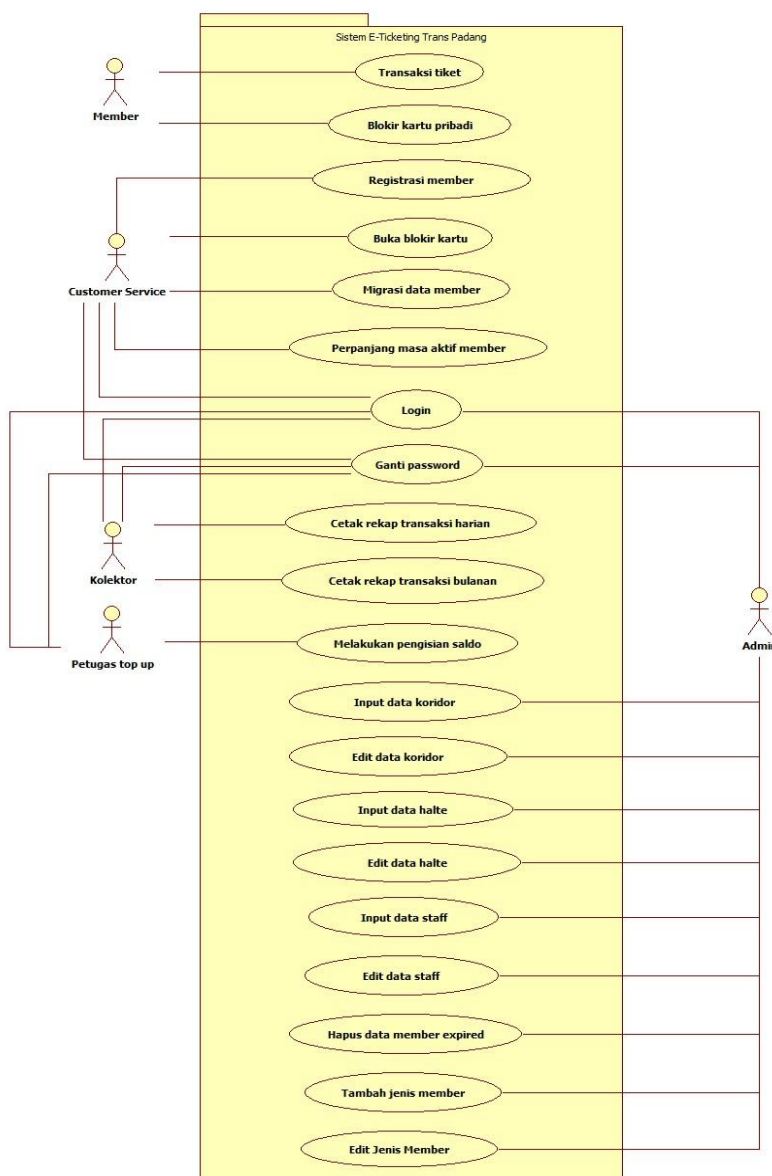


Gambar 3 BPMN pendaftaran member yang diusulkan

4.2 Use Case Diagram

Use Case Diagram untuk aplikasi *e-ticketing* Trans Padang ini memiliki 5 aktor, yaitu penumpang, *customer service*, kolektor, petugas *top up*, dan admin. Untuk mendukung mobilitas penumpang yang tinggi, maka disediakan sebuah aplikasi *mobile* untuk melihat

informasi pribadi akunnya seperti sisa pulsa, masa aktif kartu, dan informasi lainnya dimanapun dan kapanpun. Untuk lebih lengkapnya, *use case diagram* untuk aplikasi *e-ticketing* Trans Padang dapat dilihat pada gambar 4.

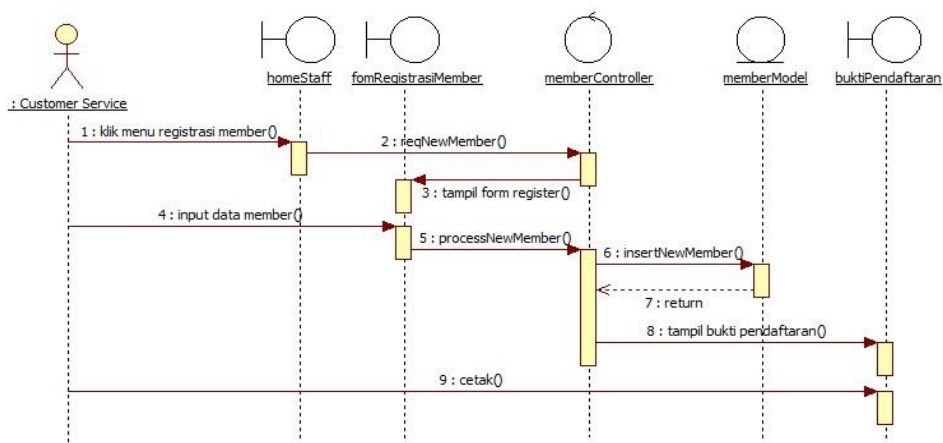


Gambar 4 Use Case Diagram

4.3 Sequence Diagram Pendaftaran Member

Dalam proses pendaftaran *member* baru, aktor yang berperan adalah *Customer Service* (CS) dimana proses dimulai dengan CS mengklik menu registrasi *member* dan menginputkan data *member* dalam halaman

formInputMember, setelah itu transaksi akan diproses dan disimpan ke dalam *database*. Setelah *input* berhasil maka sistem akan mengarahkan ke halaman cetak bukti pendaftaran untuk *member*. Untuk lebih jelas, bisa dilihat pada gambar 5



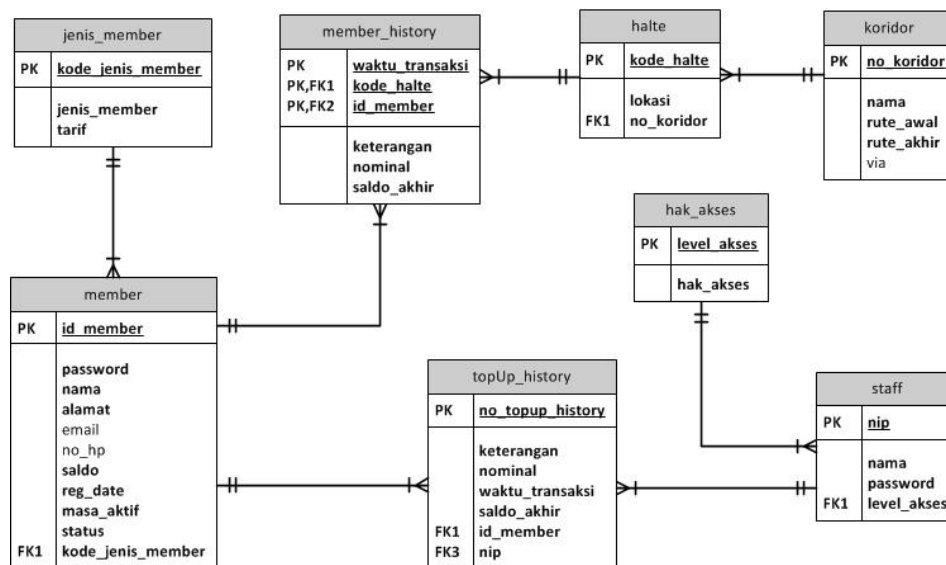
Gambar 5 Sequence Diagram Pendaftaran Member

5. Perancangan

Tahap perancangan sistem dimulai dengan melakukan perancangan *Entity Relational Diagram* (ERD), dan *class diagram*.

5.1 Entity Relational Diagram (ERD)

Dari hasil pengamatan proses bisnis *ticketing* yang terjadi saat ini serta hasil analisis beberapa dokumen yang diperoleh dari UPT Trans Padang yang dipecah menjadi beberapa tabel, yaitu tabel member, tabel jenis_member, tabel member_history, tabel halte, tabel koridor, tabel top_up_history, tabel staff, dan tabel hak_akses. Dari 8 tabel tersebut, maka dapat digambarkan *Entity Relational Diagram* (ERD) dari sistem *e-ticketing* Trans Padang seperti pada gambar 6



Gambar 6 Entity Relational Diagram (ERD) aplikasi E-Ticketing

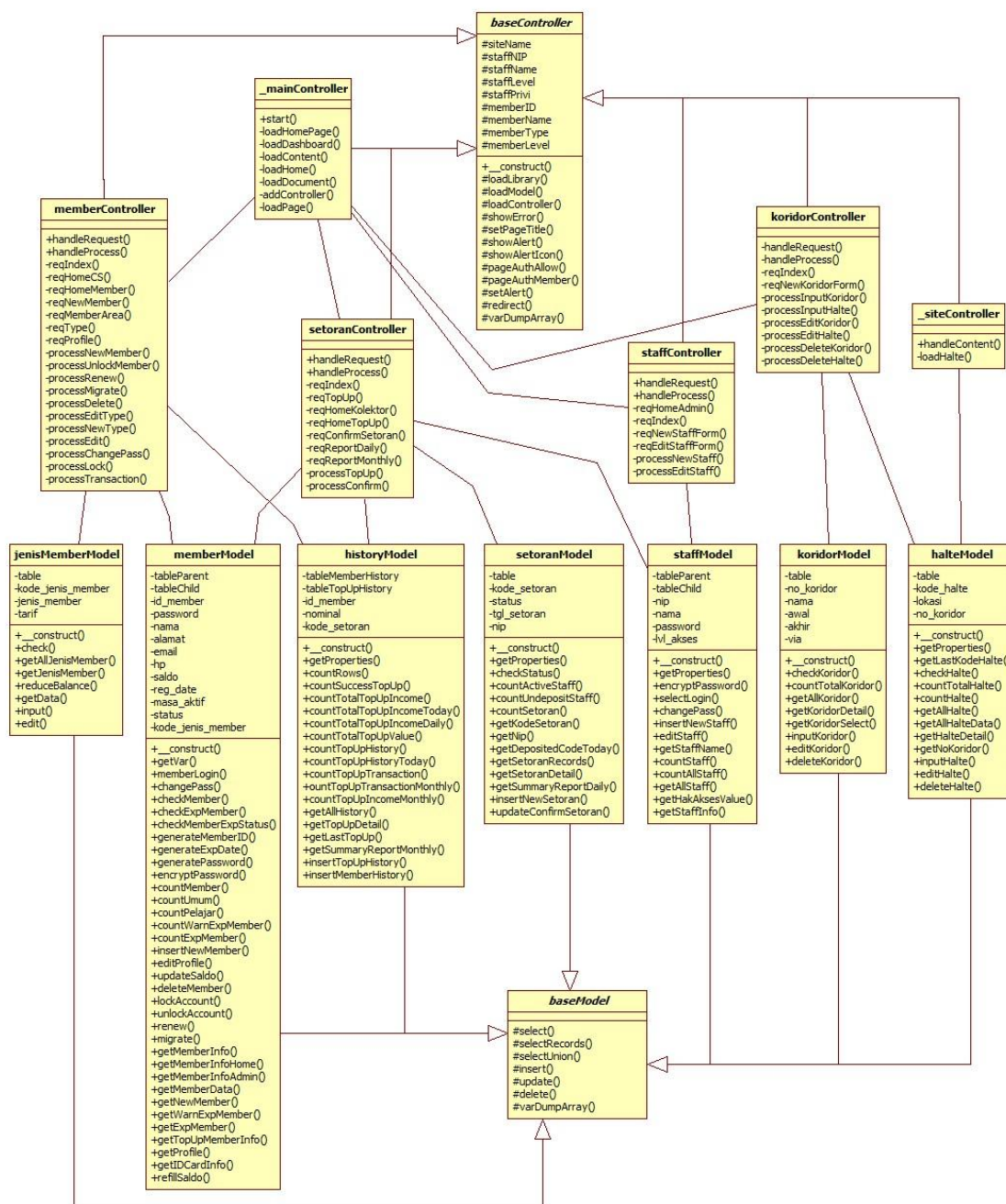
5.2 Class Diagram

Class Diagram yang dirancang untuk pembangunan aplikasi ini mengikuti arsitektur *Model-View-Controller* (MVC). Dari hasil rancangan class diagram terdapat 6 *controller* yaitu *_mainController*, *_siteController*, *memberController*, *koridorController*, *setoranController*, dan *staffController* dimana masing-masing *controller* tersebut menjembatani *view* dan *model* dari aplikasi yang dibangun.

Dalam class diagram ini, terdapat class *baseController* yang merupakan sebuah *abstract class* yaitu kelas dasar dari setiap *class controller* yang ada di dalam sistem, *baseController* memiliki atribut dan fungsi dasar yang dimiliki oleh setiap *controller*. Alur utama aplikasi diatur oleh class *_mainController* untuk mengatur bagian *back-end* (*dashboard*), dan class *_siteController* yang mengatur bagian *front-end*, dimana masing-masing class ini terhubung dengan class

memberController, staffController, setoranController, dan koridorController yang menghubungkan antara bagian view (user interface) dengan class model. Selain itu, di dalam class diagram ini juga terdapat baseModel yang merupakan abstract class yang menjadi class dasar dari setiap class model, dimana class baseModel ini

memiliki fungsi dasar dari setiap class model seperti select, update, delete yang diperlukan untuk melakukan interaksi dengan database sesuai ketentuan model dari struktur MVC. Untuk lebih jelasnya dapat melihat hasil rancangan class diagram pada gambar 7



Gambar 7 Class Diagram

6. Implementasi

Tahap implementasi merupakan tahap penterjemahan perancangan berdasarkan hasil analisis dan perancangan kedalam bahasa pemrograman sehingga dapat

menghasilkan aplikasi yang berjalan sebagaimana mestinya. Tahapan implementasi ini menghasilkan sebuah aplikasi berbasis web menggunakan bahasa pemrograman PHP, javascript, jQuery menggunakan database MySQL dan aplikasi mobile yang dikembangkan menggunakan bahasa pemrograman

Basic4ndroid. Tahap implementasi yang akan dibahas disini mencakup pengkodean dan implementasi antar muka.

6.1 Pengkodean

Dalam implementasinya, bahasa pemrograman yang digunakan dalam melakukan pengkodean adalah PHP dimana pemrograman dilakukan menggunakan konsep OOP dengan arsitektur MVC. Berikut beberapa contoh potongan dari *script* program.

1) Script Class memberController

Class controller merupakan sebuah class yang berfungsi sebagai penghubung antara *view* dan *model*. Class memberController terdiri dari fungsi-fungsi yang berhubungan dengan *control view* dengan model yang berhubungan dengan data *member*. Sebagai salah satu contohnya dibawah ini terdapat potongan *script* dari class memberController untuk menambahkan data *member* baru

```
private function processNewMember() {
    $memberModel = new memberModel(NULL, NULL,
    $_POST['nama'], $_POST['alamat'], $_POST['email'],
    $_POST['hp'], NULL, NULL,
    $_POST['jenisMember']);
    $result = $memberModel->insertNewMember();
    if($result == 1){
        $this->redirect
        ("./?ctrl=member&act=register&print=buktiPenda
        ftaran");
    }
    elseif($result == 2){
        $this->setAlert("danger", "Pendaftaran
        member gagal. Terjadi kesalahan SQL");
        $this-
        >redirect("./?ctrl=member&act=register");
    }
    else{
        $this->setAlert("danger", "<b>Pendaftaran
        member gagal.</b> Email atau No HP sudah ada
        di dalam database");
        $this-
        >redirect("./?ctrl=member&act=register");
    }
}
```

2) Script Class memberModel

Class model yang ada pada potongan *script* di bawah ini merupakan model untuk data member, dimana di dalam class ini terdapat beberapa *function* untuk memanipulasi data. Salah satu *function* tersebut adalah *insertNewMember*. *Function* *insertNewMember* berfungsi untuk mengelola beberapa *business logic* untuk menambahkan data *member* sebelum akhirnya di inputkan ke dalam *database*. Berikut merupakan potongan *script* untuk function *insertNewMember*

```
public function insertNewMember() {
    if(empty($this->email) && empty ($this-
    >hp)){
        $row['member'] = 0;
    }
    elseif(!empty($this->email)){
        $where = "email = '$this->email'";
```

```
$row = $this->select("COUNT(*) AS
member", $this->tableParent, $where, NULL,
NULL, NULL);
    }
    elseif(!empty($this->hp)){
        $where = "no_hp = '$this->hp'";
        $row = $this->select("COUNT(*) AS
member", $this->tableParent, $where, NULL,
NULL, NULL);
    }
    if($row['member'] >= 1){
        return 0;
    }
    else{
        $idMember = $this-
        >generateMemberID($this->kode_jenis_member);
        $pass = $this->generatePassword();
        $hashPass = $this-
        >encryptPassword($pass);
        $expDate = $this->generateExpDate();
        $regDate = date("Y-m-d");
        if(empty($this->email) && empty($this-
        >hp)){
            $field = "id_member, password, nama,
            alamat, reg_date, masa_aktif,
            kode_jenis_member";
            $values = "'$idMember', '$hashPass',
            '$this->nama', '$this->alamat', '$regDate',
            '$expDate', '$this->kode_jenis_member'";
        }
        elseif(!empty($this->hp)){
            $field = "id_member, password, nama,
            alamat, no_hp, reg_date, masa_aktif,
            kode_jenis_member";
            $values = "'$idMember', '$hashPass',
            '$this->nama', '$this->alamat', '$this->hp',
            '$regDate', '$expDate', '$this-
            >kode_jenis_member'";
        }
        elseif(!empty($this->email)){
            $field = "id_member, password, nama,
            alamat, email, reg_date, masa_aktif,
            kode_jenis_member";
            $values = "'$idMember', '$hashPass',
            '$this->nama', '$this->alamat', '$this-
            >email', '$regDate', '$expDate', '$this-
            >kode_jenis_member'";
        }
        else{
            $field = "id_member, password, nama,
            alamat, email, no_hp, reg_date, masa_aktif,
            kode_jenis_member";
            $values = "'$idMember', '$hashPass',
            '$this->nama', '$this->alamat', '$this-
            >email', '$this->hp', '$regDate',
            '$expDate', '$this->kode_jenis_member'";
        }
        $result = $this->insert($this-
        >tableParent, $field, $values);
        if($result){
            $_SESSION['print']['memberID'] =
            $idMember;
            $_SESSION['print']['pass'] = $pass;
            return 1;
        }
    }
}
```

```

    }
    else{
        return 2;
    }
}
}

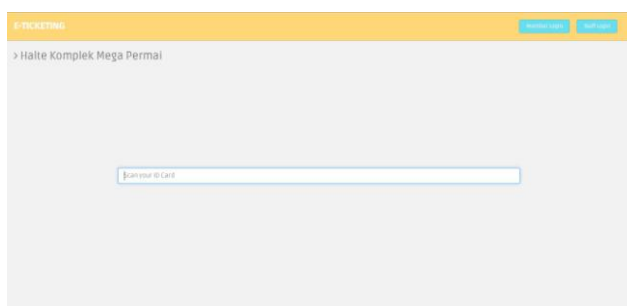
```

6.2 Implementasi Antar Muka

Implementasi antar muka mencakup antar muka aplikasi *web* dan aplikasi *mobile*. Berikut beberapa antar muka dari implementasi aplikasi yang dibangun.

1) Antar muka aplikasi *web*

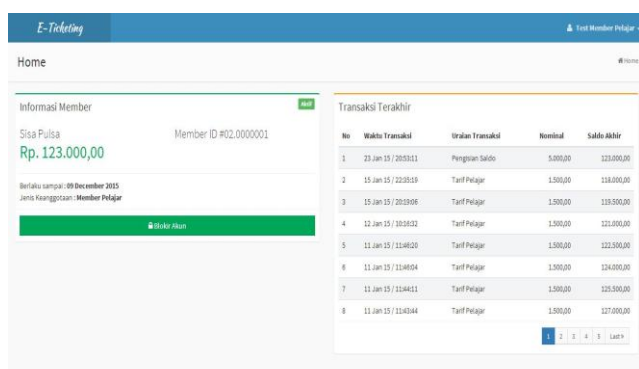
Dalam bagian ini akan ditampilkan beberapa contoh dari antar muka aplikasi web, yaitu antar muka halaman transaksi pembayaran *ticketing* pada gambar 8, halaman *login member* pada gambar 9, dan halaman *dashboard member* pada gambar 10



Gambar 8 Halaman transaksi pembayaran *ticketing*



Gambar 9 Halaman *login member*



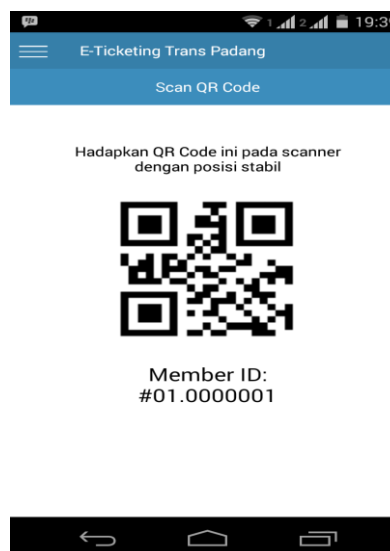
Gambar 10 Halaman *Dashboard Member*

2) Antar muka aplikasi *mobile*

Pada bagian ini akan ditampilkan beberapa contoh dari antar muka aplikasi *mobile*, yaitu antar muka halaman informasi *member* pada gambar 11, dan halaman *show QR Code* pada gambar 12



Gambar 11 Halaman informasi *member mobile*



Gambar 12 Halaman *Show QR Code mobile*

7. Pengujian

Tahap pengujian sistem merupakan proses untuk memeriksa apakah aplikasi yang dihasilkan telah memenuhi standar yang telah ditetapkan. Pengujian aplikasi *e-ticketing* ini menggunakan metode pengujian *black box testing* yang berfokus kepada syarat fungsional dari aplikasi yang dibangun. Metode pengujian *black box testing* digunakan untuk aplikasi *web* dan *mobile* yang diuji langsung oleh penulis.

7.1 Fokus Pengujian

Fokus pengujian dari aplikasi *e-ticketing* yang dibangun dapat dilihat pada tabel 1

Tabel 1 Fokus Pengujian Aplikasi *e-Ticketing*

No	Item yang diuji	Aplikasi yang diuji	Proses yang diuji
1	Login member	Web dan mobile	Verifikasi user
2	Login Staff	Web	Verifikasi user
3	Data staff	Web	Input
			Edit
			Search
4	Data member	Web	Register
			Search
			Lock Account
			Unlock Account
			Perpanjang Masa aktif
			Migrasi Akun
			Delete
5	Data koridor	Web	Input
			Edit
			Delete
			Search
6	Data halte	Web	Input
			Edit
			Delete
7	History	Web	Transaksi Pembayaran
			Top up
8	Data jenis member	Web	Insert
			Edit
9	Laporan	Web	Rekap Transaksi Harian
			Rekap Transaksi Bulanan

7.2 Kasus dan Hasil Pengujian

Bagian ini menjelaskan mengenai kasus serta hasil dari pengujian yang dilakukan. Pengujian dilakukan berdasarkan fokus pengujian yang dipaparkan pada table1. Pengujian menggunakan metode *black box testing*, yaitu pengujian berdasarkan fungsional yang memperhatikan *input* dan *output* dari sistem. Berikut merupakan beberapa contoh dari hasil pengujian.

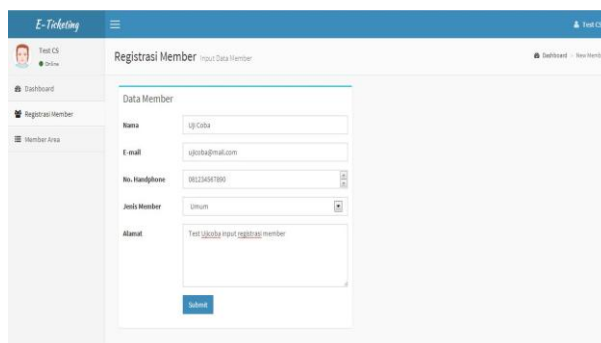
1. Registrasi Member

Salah satu pengujian dari data *member* adalah pengujian proses *Register*. Pengujian dilakukan dalam 2 kondisi yaitu kondisi normal dan kondisi salah, dimana dalam kondisi salah, data email dan no hp *member* terdeteksi sudah ada di dalam *database* dan input ditolak. Hasil pengujian dapat dilihat pada tabel 2

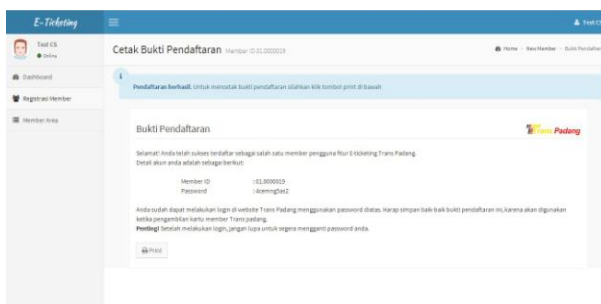
Tabel 2 Pengujian data *member*

Kasus dan hasil uji (normal)	
<i>Register</i>	
Data masukan	Data lengkap <i>member</i>
Yang diharapkan	Data tersimpan dan muncul bukti pendaftaran
Pengamatan	Data tersimpan dan sistem memunculkan bukti pendaftaran
Kesimpulan	Diterima
Kasus dan hasil uji (salah)	
<i>Register</i>	
Data masukan	Data email/no hp sudah ada di dalam <i>database</i>
Yang diharapkan	Data tidak tersimpan, dan muncul pemberitahuan
Pengamatan	Data <i>member</i> tidak disimpan dan muncul pemberitahuan
Kesimpulan	Diterima

Dalam melakukan uji coba registrasi *member* pada kondisi normal, *input* data dilakukan dengan menggunakan aplikasi pada kasus uji coba normal, *customer service* menginputkan data *member* di *form* aplikasi yang tersedia seperti pada gambar 13. Jika data berhasil diinputkan maka program akan menampilkan notifikasi berhasil input dan cetak bukti pendaftaran seperti pada gambar 14



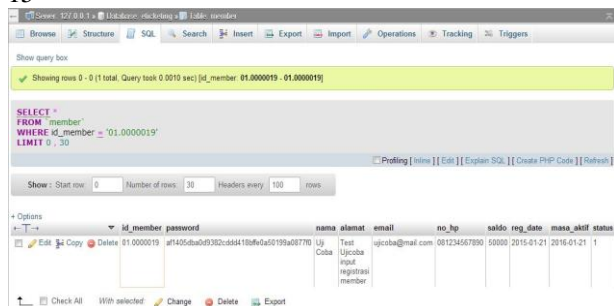
Gambar 13 Pengujian registrasi *member* dengan kondisi normal



Gambar 14 Notifikasi berhasil input data *member* dan bukti pendaftaran

Untuk melakukan pembuktian bahwa data sudah diinputkan sesuai dengan keinginan, maka pengujian dilanjutkan dengan mengakses langsung ke *database*

dan melakukan *query select* dengan *field id_member* yang diinputkan sebelumnya. Pembuktian dengan *query database* dapat dilihat pada gambar 15



Gambar 15 Pengecekan pengujian registrasi *member* dengan kondisi normal

8. Kesimpulan

Dalam pengembangan sistem ini mulai dari tahap analisis, perancangan sampai ke tahap implementasi dan pengujian dapat diambil kesimpulan bahwa pembangun aplikasi *e-ticketing* menggunakan teknologi QR Code berbasis *web* dengan fitur *mobile* pada BRT Trans Padang telah berhasil dan sesuai dengan fungsional yang diinginkan dengan uraian sebagai berikut:

1. Kegiatan proses bisnis *e-ticketing* yang terjadi pada BRT Trans Padang saat ini yaitu, permintaan tiket, distribusi tiket, dan penyetoran hasil penjualan. Kegiatan tersebut kebanyakan dilakukan secara manual tanpa adanya sistem komputerisasi
2. Tahapan analisis dan perancangan yang digunakan dalam pembangunan aplikasi *e-ticketing* pada BRT Trans Padang yaitu menggunakan *Business Process Modelling Notation* (BPMN), *use case*, *sequence diagram*, *class analysis*, *class diagram*, dan *Entity Relationship Diagram* (ERD).
3. Dalam membangun aplikasi, penulis menggunakan konsep *Object Oriented Programming* (OOP) dengan arsitektur *Model-View-Controller* (MVC). Proses pemrograman dilakukan dengan menggunakan bahasa pemrograman PHP, pada sisi *server* dan menggunakan javascript, dan jQuery pada sisi *client* untuk pembangunan aplikasi berbasis *web*, menggunakan bahasa pemrograman basic4android untuk pembangunan aplikasi *mobile* dan menggunakan DBMS MySQL sebagai *database*.
4. Pengujian sistem dilakukan dengan menggunakan metode *black box testing* untuk menguji fungsional sistem. Hasil pengujian menunjukkan bahwa aplikasi *e-ticketing* sudah sesuai dengan kebutuhan fungsional dan dapat berjalan sebagaimana mestinya.
5. Aplikasi *e-ticketing* menunjukkan mobilitas yang tinggi dengan fitur *mobile* pada sistem informasi *e-ticketing* yang memungkinkan pengguna untuk melihat informasi akunnya dimanapun dan kapanpun saat dibutuhkan.

REFERENSI

- [1] Nasrulloh, Mokhammad. 2010. *Sistem Bus Rapid Transit Di Jakarta: Integrasi Perkotaan Dan Dampak Lingkungan*. Tesis pada Universitas Indonesia: Tidak Diterbitkan.
- [2] Haluan. 2014. *Ada Kecurangan Di Trans Padang*. [Online] Tersedia di: <http://harianhaluan.com/index.php/berita/haluan-padang/30038-ada-kecurangan-di-trans-padang> [Diakses 2 September 2014].
- [3] Ningrum, Dewi Widya. 2013. *Akan Ada 103,7 Juta Pengguna Smartphone di Indonesia*. [Online] Tersedia di: <http://tekno.liputan6.com/read/731892/akan-ada-1037-juta-pengguna-smartphone-di-indonesia> [Diakses 2 September 2014].
- [4] Mazaya, Aghnia, dkk. 2012. *Analisis Permintaan Transjakarta*. Skripsi S1 pada Universitas Bina Nusantara: Tidak Diterbitkan.
- [5] Khasan, Nur. 2009. *Sistem Informasi Manajemen Mengenai E-ticketing pada Transportasi*. [Online] Tersedia di: <http://mediacomfensclub.blogspot.com/2009/07/makalah-sistem-informasi-manajemen-sim.html> [Diakses 2 September 2014]
- [6] Denso Wave Incorporated. *What is a QR Code?*. [Online] Tersedia di: <http://www.qrcode.com/en/about/> [Diakses 3 September 2014]
- [7] Saragih, Larhotdianto. 2013. *Pengembangan Sistem Informasi Manajemen Tambang Di PT Sebuku Iron Lateritic Ores*. Tugas Akhir pada Universitas Atma Jaya Yogyakarta: Tidak Diterbitkan.