

PENGEMBANGAN APLIKASI TERMINAL BERBASIS *WEBSITE* DI DINAS PERHUBUNGAN PROVINSI JAWA BARAT DENGAN FRAMEWORK LARAVEL DAN METODE *WATERFALL*

Briliant Marista Zakka Billy¹, Rd. Rohmat Saedudin², Hanif Fakhurroja³

^{1,2,3} Universitas Telkom, Bandung

¹billyzakka@student.telkomuniversity.ac.id, ²rdrohmat@telkomuniversity.ac.id,

³haniff@telkomuniversity.ac.id

Abstrak

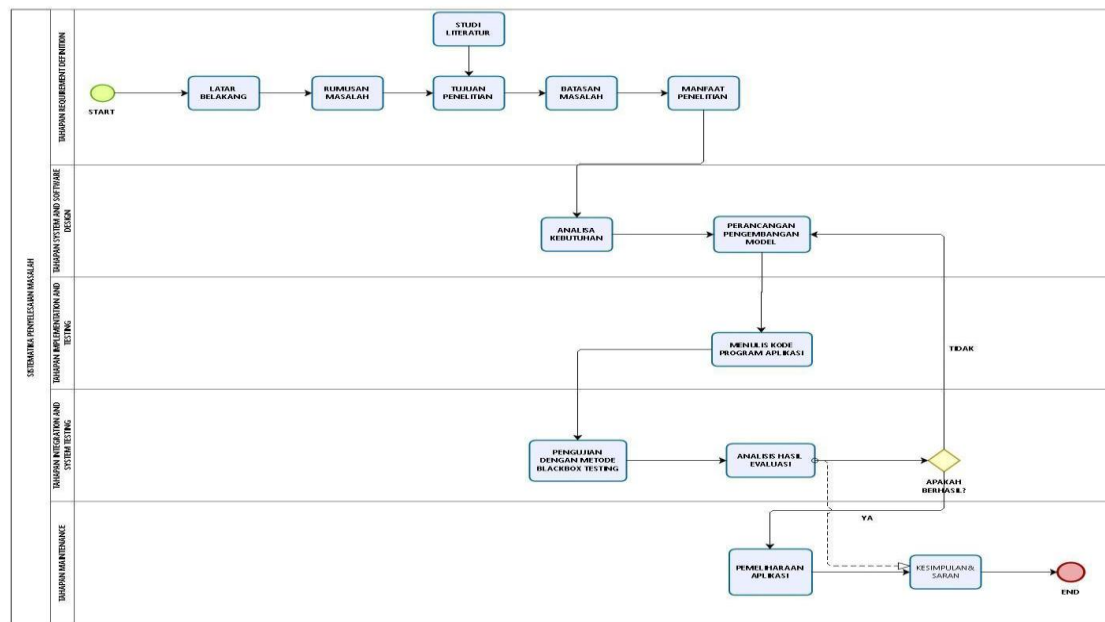
Website terminal merupakan aplikasi informatif mengenai jadwal perjalanan bus, jumlah penumpang bus, data perusahaan otobus yang beroperasi, serta daftar terminal tipe B di wilayah Jawa Barat. Identifikasi permasalahan yang ada menunjukkan ketidakadaan data yang *real time* pada aplikasi dan redundansi data dalam *database*, sehingga diperlukan *Enhancement* atau peningkatan spesifikasinya. Metode pengembangan aplikasi ini menggunakan metode *Waterfall* yang dimulai dari pengumpulan literatur yang relevan serta observasi kesalahan pada aplikasi dengan metode wawancara dan *Expert Judgement*. Selanjutnya, informasi proses bisnis tersebut dianalisis dengan metode *Black Box Testing*, kemudian diimplementasikan dalam DFD serta notasi ERD. Pengujian dan implementasi aplikasi dengan metode *User Acceptance Test* menunjukkan bahwa semua sistem pada beberapa fitur dapat berjalan 100% sukses. Hasil dari penelitian ini adalah sebuah media informatif berupa *website* yang memberikan informasi penjadwalan bus kepada masyarakat, serta mempermudah pencatatan jadwal bus dan daftar terminal tipe B milik Dinas Perhubungan Provinsi Jawa Barat.

Kata Kunci : *website terminal, framework Laravel, metode waterfall, dinas perhubungan, informasi, User Acceptance Test*

Abstract

The terminal website is an informative application regarding bus travel schedules, the number of bus passengers, data on bus companies operating, as well as a list of type B terminals in the West Java region. Identification of problems in the existing application shows the unavailability of real-time data in the application and data redundancy in the database. This problem requires enhancement or increasing specifications of the application. The development of the application uses the Waterfall method, starting from collecting relevant literatures and observing errors in applications by conducting interviews and *Expert Judgment*. Next, the information of the business process is analyzed with *Black Box Testing* in the form of DFD and ERD notation. Program testing and implementation conducted by *User Acceptance Test* defines all systems on several features that can run 100% successfully. The result of the research is an informative media in the form of a website to provide bus scheduling information to the public and to facilitate the recording of bus schedules and type B terminals list belonging to the West Java Provincial Transportation Service.

Keywords: *terminal website, Laravel framework, waterfall methodology, department of transportation, information, User Acceptance Test*



Gambar 1. Sistematika Penyelesaian Masalah Pengembangan Aplikasi Terminal Berbasis Website

Sumber: Bizagi Modeler.

1. Pendahuluan

Berdasarkan identifikasi permasalahan aktual, aplikasi terminal berbasis *website* milik Dinas Perhubungan Provinsi Jawa Barat memiliki beberapa permasalahan yang masif, diantaranya belum tersedianya data yang *real time* pada aplikasi terminal dan ketidakadaan relasi antardata dalam *database* yang menyebabkan terjadinya redudansi data pada aplikasi. Oleh karena itu, diperlukan *Enhancement* atau peningkatan spesifikasi pada sistem aplikasi ini agar terlihat *interactive* dan juga diperlukan adanya kerja sama antara petugas di terminal bus dengan admin dalam menginput data-data informasi. Maka, tujuan penelitian ini adalah mengimplementasikan dan mengevaluasi aplikasi terminal berbasis *website* menggunakan *framework* Laravel dengan metode *Waterfall* untuk meningkatkan dalam menyebarkan informasi sesuai dengan

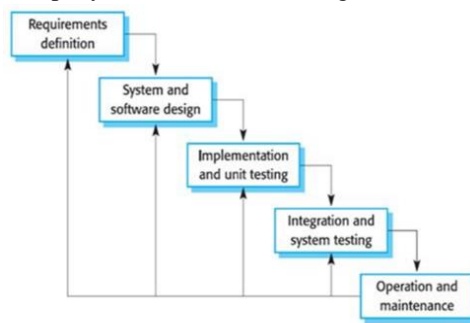
kebutuhan pengguna.

Framework Laravel sendiri adalah sebuah kerangka kerja *opensource* yang diciptakan oleh Taylor Otwell. *Framework* ini menawarkan alat-alat pengembangan arsitektur aplikasi yang dapat meningkatkan kecepatan pengembangan *web*. Struktur aplikasinya menggunakan metode MVC yang memisahkan data dari tampilan berdasarkan *Model* struktur data, *View* dari *User Interface*, dan *Controller* sebagai jembatan antara *Model* dan *View*. Pengembangan *web* menggunakan Laravel ini memiliki kelebihan yaitu waktu pengembangan *web* menjadi lebih cepat, jumlah pengunjung *website* meningkat dengan penggunaan di segala *browser*, dan moderasi *website* terbaik dengan kemasan *Modular Packaging Systems* (MPS).

2. Metode Penelitian

Langkah-langkah penelitian pengembangan aplikasi terminal dilakukan dengan sistematika penyelesaian masalah (Gambar 1) berdasarkan metode *Waterfall* (Gambar 2). Berikut penjelasan tiap tahapnya:

A. Metode *Waterfall* dengan penjelasan setiap tahap pada sistematika penyelesaian masalah sebagai berikut:



Gambar 1. (Metode Waterfall)

Sumber: Sommerville, 2011

1. Requirement Analysis / Definition

Sebelum melakukan pengembangan perangkat lunak, seorang pengembang harus mengetahui dan memahami bagaimana informasi kebutuhan pengguna terhadap sebuah perangkat lunak. Metode pengumpulan informasi ini diperoleh dengan cara observasi, memungkinkan adanya wawancara, atau sebagainya. Informasi yang diperoleh kemudian diolah dan dianalisis sehingga didapatkan data atau informasi yang lengkap mengenai spesifikasi kebutuhan pengguna akan perangkat lunak yang akan dikembangkan.

2. System and Software Design

Informasi mengenai spesifikasi kebutuhan dari tahap *Requirement Analysis* selanjutnya di analisa pada tahap ini untuk kemudian diimplementasikan pada desain pengembangan. Perancangan desain dilakukan dengan tujuan membantu memberikan gambaran lengkap mengenai apa yang harus dikerjakan. Tahap ini juga akan membantu pengembang untuk menyiapkan kebutuhan *hardware*

dalam pembuatan arsitektur sistem perangkat lunak yang akan dibuat secara keseluruhan.

3. Implementation and Unit Testing

Tahap *implementation and unit testing* merupakan tahap pemrograman. Pembuatan perangkat lunak dibagi menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya. Disamping itu, pada fase ini juga dilakukan pengujian dan pemeriksaan terhadap fungsionalitas modul yang sudah dibuat, apakah sudah memenuhi kriteria yang diinginkan atau belum.

4. Integration and System Testing

Setelah seluruh unit atau modul yang dikembangkan dan diuji di tahap implementasi selanjutnya diintegrasikan dalam sistem secara keseluruhan. Setelah proses integrasi selesai, selanjutnya dilakukan pemeriksaan dan pengujian sistem secara keseluruhan untuk mengidentifikasi kemungkinan adanya kegagalan dan kesalahan sistem.

5. Operation and Maintenance

Pada tahap terakhir dalam Metode *Waterfall*, perangkat lunak yang sudah jadi dioperasikan pengguna dan dilakukan pemeliharaan. Pemeliharaan memungkinkan pengembang untuk melakukan perbaikan atas kesalahan yang tidak terdeteksi pada tahap-tahap sebelumnya. Pemeliharaan meliputi perbaikan kesalahan, perbaikan implementasi unit aplikasi, peningkatan dan penyesuaian sistem sesuai dengan kebutuhan.

B. Blackbox Testing

Metode *Blackbox Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan. Estimasi banyaknya data uji dapat dihitung melalui banyaknya *field data entry* yang akan diuji, aturan entri yang

harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid. (Mustaqbal, M.S.M., Firdaus, R.F.F., dan Rahmadi, H.R; 2015).

Black Box Testing merupakan pengujian fungsionalitas yang merancang kasus berdasarkan informasi dari spesifikasi, *Black Box Testing* menguji tidak dari segi *source code* melainkan berfokus kepada sudut pandang *user* yang mengetahui *input* dan hasil *output*nya seperti yang dirancang (Nidhra & Dondeti, 2012).

C. User Acceptance Test (UAT)

User Acceptance Testing adalah pengujian yang dilakukan oleh pengguna dari sistem tersebut untuk memastikan fungsi-fungsi yang ada pada sistem tersebut telah berjalan dengan baik dan sesuai dengan kebutuhan pengguna (L. Binus, 2017).

Menurut Black (2002:7), *acceptance testing* biasanya berusaha menunjukkan bahwa sistem telah memenuhi persyaratan-persyaratan tertentu. Pada pengembangan *software* dan *hardware* komersial, *acceptance test* biasanya disebut juga "*alpha tests*" (yang dilakukan oleh pengguna *in-house*) dan "*beta tests*" (yang dilakukan oleh pengguna yang sedang menggunakan atau akan menggunakan sistem tersebut). *Alpha* dan *beta test* biasanya juga menunjukkan bahwa produk sudah siap untuk dijual atau dipasarkan. *Acceptance testing* mencakup data, *environment* dan skenario yang sama atau hampir sama pada saat *live* yang biasanya berfokus pada skenario penggunaan produk tertentu.

D. Pengumpulan & Sumber Data.

Bahwa metode pengumpulan data untuk mengembangkan aplikasi terminal

berbasis *website* dengan *framework* laravel ini, meliputi:

1. Data primer diperoleh melalui wawancara langsung dengan staff IT di Bagian Kepegawaian secara mendalam dengan observasi kesalahan dan kekurangan sistem pada aplikasi terminal berbasis *website* yang berhubungan dengan penelitian ini untuk dikembangkan lebih detail.
2. Data sekunder yang dibutuhkan diperoleh dari instansi pemerintahan yang berhubungan dengan penelitian ini yaitu Dinas Perhubungan Provinsi Jawa Barat melalui salah satu staff di Bagian Kepegawaian, serta literatur-literatur yang relevan seperti buku-buku, jurnal penelitian internet dan laporan-laporan yang berhubungan dengan penelitian ini.
3. Khusus untuk pengembangan aplikasi terminal berbasis *website* ini, penulis mendapatkan dari seorang staff Bagian Kepegawaian di Dinas Perhubungan Provinsi Jawa Barat dan akan dikembangkan dengan *software tool* yang bernama Visual Studio Code serta *framework* laravel versi terbaru.

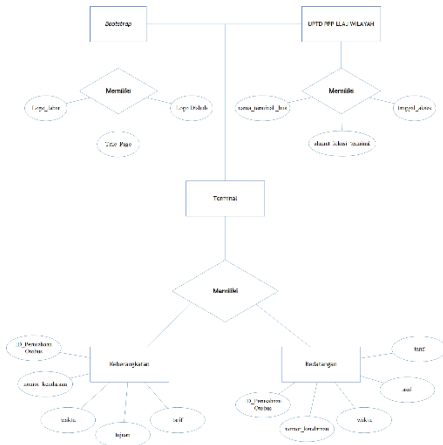
3. Hasil dan Pembahasan

Melihat metode yang digunakan dalam pengembangan aplikasi terminal berbasis *website* pada penelitian ini, akan memaparkan bagaimana hasil implementasi dan *testing* menggunakan metode *waterfall*, sedangkan untuk implementasi aplikasi *website* menggunakan *framework* Laravel dan *Blackbox Testing* untuk pengujian dan *User Acceptance Test* (UAT) sebagai evaluasi akhir dari hasil pengembangan pada aplikasi, sehingga dapat memenuhi tujuan dari penelitian ini, sebagai berikut:

A. Analisis Proses Bisnis

1. *Entity Relationship Diagram* (ERD). *Entity Relationship Diagram* (ERD) *Notation* dirancang untuk menggambarkan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk

memodelkan struktur data dan hubungan antar data, untuk meng gambarkannya digunakan beberapa notasi dan simbol pada aplikasi *website* terminal secara umum. Berikut merupakan ERD yang terdapat pada aplikasi *website* terminal pada Gambar 3 dibawah ini:

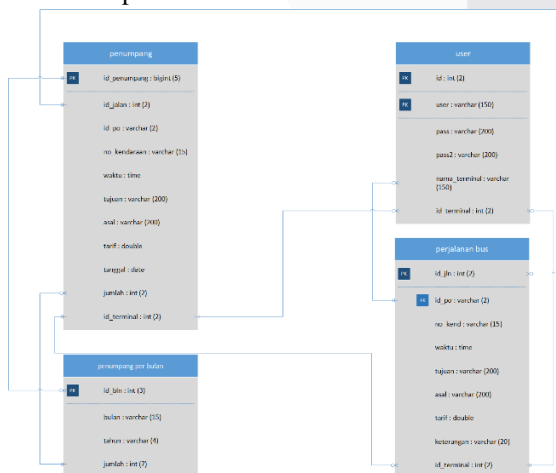


Gambar 3. (ERD Notation)

Sumber: Visio 2013

2. Class Diagram

Class Diagram dirancang berdasarkan kebutuhan proses bisnis yang terlihat pada *use case* deskripsi yang sudah dirancang dan memiliki alur berjalannya aplikasi pada *website* Terminal. Berikut merupakan rancangan *Class Diagram* dan juga gambaran relasi tabel dari aplikasi yang dimiliki oleh *website* Terminal pada Gambar 4 dibawah ini:



Gambar 4. (Class Diagram)

Sumber: Visio 2013

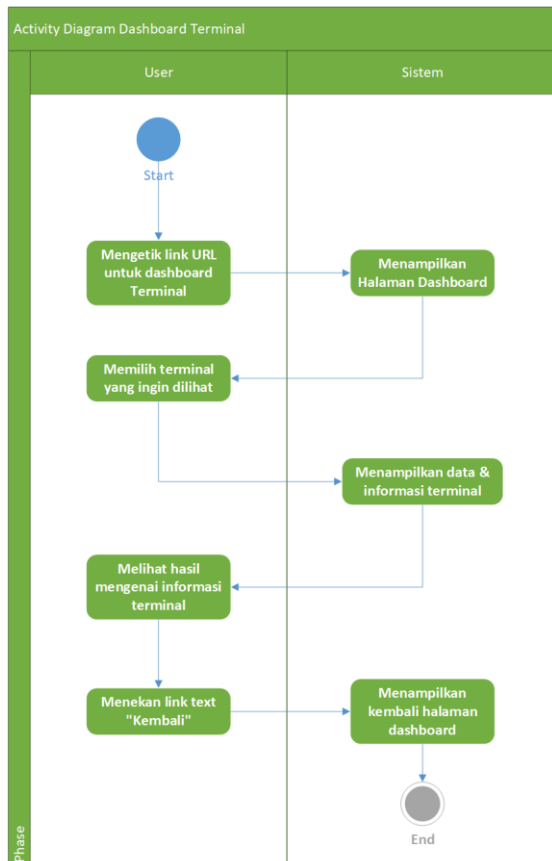
3. Activity Diagram

Activity Diagram dirancang untuk menggambarkan aktivitas *user* saat menggunakan aplikasi Terminal, sehingga ketika membangun aplikasi Terminal memiliki batasan sehingga hasil pengembangan sesuai dengan yang diharapkan dengan perancangan sebelumnya. *Activity diagram* ini dapat dijadikan parameter untuk pengujian *Blackbox Testing*, berikut merupakan *activity diagram* yang ada pada *website* Terminal. Berikut merupakan rancangan *Activity Diagram* dan juga gambaran relasi tabel dari aplikasi yang dimiliki oleh *website* Terminal pada Gambar 5 dan 6 dibawah ini:



Gambar 5. (Activity Diagram Halaman Admin Terminal)

Sumber: Visio 2013

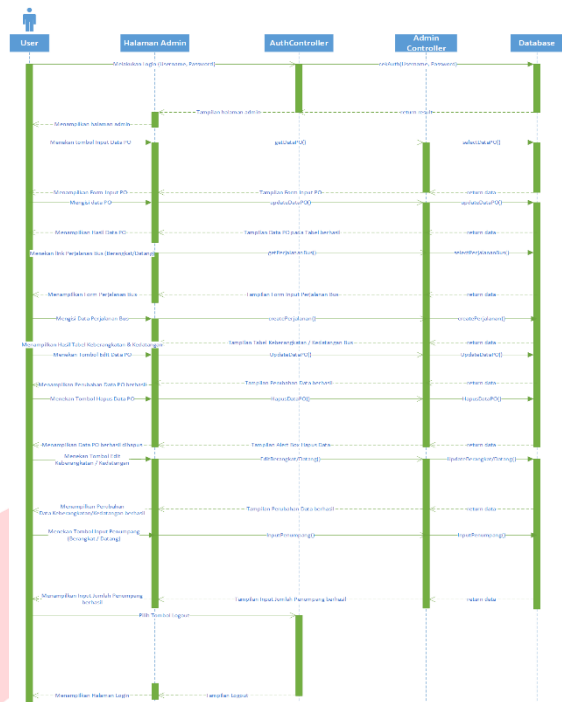


Gambar 6. (Activity Diagram Dashboard Terminal)

Sumber: Visio 2013

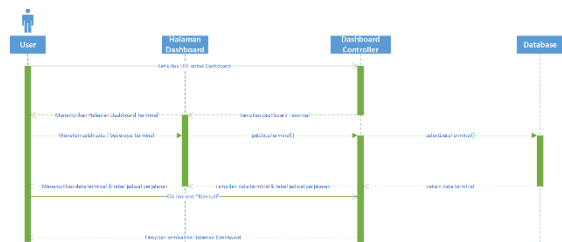
4. Sequence Diagram

Pada *Activity Diagram* telah dibahas bagaimana alur yang berjalan pada aplikasi Terminal untuk menggambarkan alur sistem didalamnya Terminal merancang *Sequence Diagram* untuk mempermudah pengembang merancang penulisan *method* yang digunakan dan interaksi antara aktor maupun teknologi didalamnya secara keseluruhan. Berikut merupakan *Sequence Diagram* Halaman Admin Terminal dan *Dashboard* Terminal yang terdapat pada Gambar 7 serta 8 dibawah ini.



Gambar 7. (*Sequence Diagram* Halaman Admin Terminal)

Sumber: Visio 2013

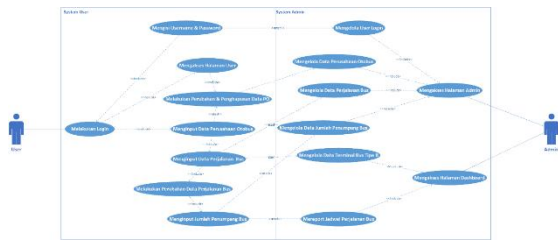


Gambar 8. (Sequence Diagram Dashboard Terminal)

Sumber: Visio 2013

5. Use Case Diagram

Use Case Diagram dirancang untuk menggambarkan hubungan interaksi yang terjadi antara sistem dengan lingkungan nyata. Pada diagram aplikasi *website* terminal ini melibatkan dua aktor yang terlibat yaitu *user* oleh petugas terminal dan admin internal. Berikut merupakan *Use Case Diagram* yang ada pada aplikasi *website* terminal pada Gambar 9 dibawah ini:



Gambar 9. (Use Case Diagram Aplikasi Terminal)

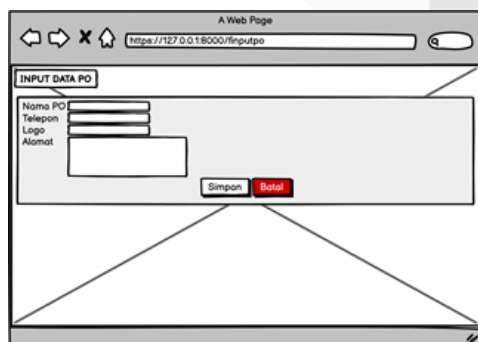
Sumber: Visio 2013

4. Desain Aplikasi

Pada bagian ini, disajikan hasil desain atau perancangan sistem informasi berupa *Mockup* dengan Balsamiq Wireframes versi terbaru, sebagai berikut:

I. Pada Input Data Page

Mockup Input Data Page juga merupakan acuan untuk mengembangkan *website* terminal ini. Pada bagian *mockup* ini terdapat komponen berupa *header text* atau yang dapat disebut dengan *title page* yang bertuliskan "Input Data PO". Kemudian, ada *form area* yang berisi data-data isian seperti nama Perusahaan Otobus, nomor telepon dari perusahaan tersebut, logo dari perusahaan otobus tersebut dan alamat perusahaan Otobus. Selanjutnya, ada komponen seperti *button* untuk menyimpan data serta membatalkan penyimpanan data perusahaan otobus tersebut seperti terdapat pada Gambar 10 dibawah ini.

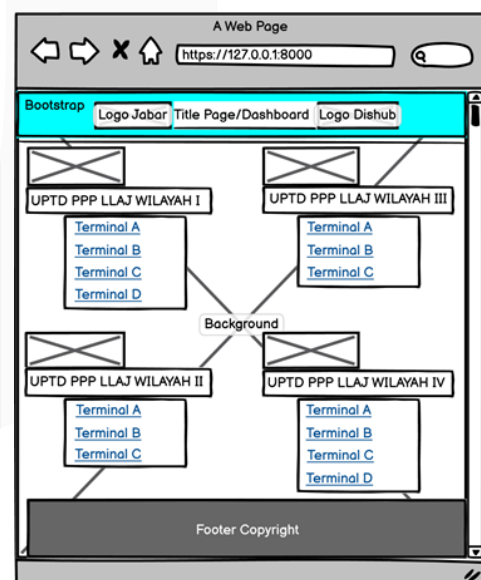


Gambar 10. (Mockup Input Data Page)

Sumber: Balsamiq Wireframes

II. Pada Dashboard Page

Mockup Landing Page User/Dashboard menjadi acuan untuk mengimplementasikan tampilan pada halaman utama yaitu halaman *landing page user*, halaman *landing page* terdiri dari *Bootstrap* berisi tulisan yang merupakan judul dari halaman tersebut dilengkapi dengan lambang dari Provinsi Jawa Barat dan logo Dinas Perhubungan. Lalu yang kedua adalah *background* yang menampilkan gambar mengenai suasana terminal bus dengan contoh seperti di salah satu terminal angkutan umum di wilayah Selatan Kota Bandung. Komponen berikutnya yaitu *header row*, kemudian gambar bus dengan warna yang berbeda, dibawahnya terdapat tulisan Unit Pelaksana Teknis Daerah, daftar terminal yang ditandai oleh *link text* dan *footer copyright*. Untuk *mockup* tersebut dapat dilihat pada Gambar 11 dibawah ini.



Gambar 11. (Mockup Dashboard Page)

Sumber: Balsamiq Wireframes

III. Pada Admin Dashboard

Mockup Master Data Page merupakan acuan untuk pengembangan *website* terminal ini dengan pola *Create, Read,*

Update, dan Delete. Mockup tersebut dapat dilihat pada Gambar 12 dibawah ini.

The mockup shows a web application interface for an administrator. It features a top navigation bar with a logo and a sidebar with user information and a 'Logout' button. The main content area is divided into several sections:

- DATA PERUSAHAAN OTOBIS**: A table with columns for ID Pers, Nama PO, Tip, Alamat, and Action (Edit, Hapus). It includes an 'Input PO' button.
- BERANGKAT**: A table with columns for ID Pers, No Kend, Waktu, Tujuan, Tarif, Action (Edit, Input), and Penumpang.
- KEDATANGAN**: A table with columns for ID Pers, No Kend, Waktu, Asal, Tarif, Action (Edit, Input), and Penumpang.
- JUMLAH PENUMPANG KEBERANGKATAN**: A table with columns for ID Pers, No Kend, Waktu, Asal, Tarif, Action, and Penumpang.
- JUMLAH PENUMPANG KEDATANGAN**: A table with columns for ID Pers, No Kend, Waktu, Asal, Tarif, Action, and Penumpang.

Each table has a 'Halaman : 1 dari 2' indicator at the bottom.

Gambar 12. (Mockup Admin Dashboard)

Sumber: Balsamiq Wireframes

5. Spesifikasi Kebutuhan Teknologi

Spesifikasi kebutuhan teknologi dijelaskan berdasarkan aspek *People*, *Process*, dan *Technology* yang saling berkaitan, serta terdiri dari dua analisis kebutuhan yang berbeda sebagai berikut:

- Kebutuhan fungsional adalah kebutuhan-kebutuhan yang memiliki keterkaitan langsung dengan sistem. Kebutuhan fungsional dari sistem ini meliputi, antara lain:
 - Kebutuhan bagi user meliputi masyarakat umum dan para calon penumpang bus, antara lain:

- Mendapatkan informasi mengenai jadwal keberangkatan dan kedatangan bus.
- Mendapatkan informasi data perusahaan otobus yang akan dihubungi sebagai calon bus yang akan digunakan.

- Kebutuhan bagi *administrator* meliputi operator, petugas lapangan, dan beberapa pejabat Dinas Perhubungan yang terkait, antara lain:

- Melakukan *login* serta *input* ke sistem *website*.
- Database Management System (DBMS)* dengan pola *CRUD*.
- Mengelola data keberangkatan maupun kedatangan bus, jumlah penumpang yang berangkat dan tiba di terminal, serta perusahaan otobus yang terdaftar di terminal bus tipe B milik Dinas Perhubungan Provinsi Jawa Barat.

- Kebutuhan non-fungsional adalah kebutuhan yang tidak secara langsung terkait dengan fitur tertentu di dalam sistem, antara lain:

- Kebutuhan perangkat keras (Hardware)**, antara lain:
 - Hardisk Internal 1 TB.
 - RAM 8.00 GB
- Kebutuhan perangkat lunak (Software)**, antara lain:
 - Sistem Operasi Windows 10 Ultimate.
 - Mysql dan phpMyAdmin sebagai media *database management system*.
 - Microsoft Office 2013 sebagai pembuatan dokumen tugas akhir.

- d. Pemrograman menggunakan bahasa HTML, PHP, dan MySQL.
- e. Visio 2013 untuk menganalisis serta merancang proses bisnis terhadap sistem aplikasi yaitu ERD Notation, Class Diagram, Activity Diagram, Sequence Diagram, dan Use Case Diagram.
- f. XAMPP v3.2.4 untuk media atau web server.
- g. Visual Studio Code, sebagai Editor Script Pemrograman.
- h. Laravel, framework yang digunakan untuk mengembangkan website terminal ini.
- i. Bizagi Modeler, digunakan untuk pembuatan alur sistematis penyelesaian masalah.
- j. Balsamiq Wireframes, untuk pembuatan desain perangkat lunak.

6. Hasil Implementasi & Pengujian

1. Pengujian pada halaman utama (Dashboard).

Tabel 1. (Pengujian Halaman Utama)

No	Input	Skenario	Hasil Pengujian	Keterangan
1	Klik teks beberapa terminal bus pada halaman menu utama atau Dashboard	Sistem akan menampilkan halaman profil terminal bus beserta jadwal bus	User masuk ke profil terminal bus dan jadwal bus	Sukses
2	Klik link text untuk kembali ke halaman utama	Sistem akan mengembalikan ke halaman utama	Sistem otomatis kembali ke halaman utama untuk membuka profil terminal lainnya	Sukses
3	Klik link text instansi pemerintahan pemilik website terminal	Sistem akan terhubung dengan web lain yang terkait	Sistem langsung terhubung dengan website lain yang terkait	Sukses

2. Pengujian pada halaman login dan logout administrator.

Tabel 2. (Pengujian Login & Logout Administrator)

No	Input	Skenario	Hasil Pengujian	Keterangan
1	Memasukkan username dan password sesuai dengan terminal yang dipilih	Sistem akan menampilkan halaman utama khusus admin untuk input database system	Sistem otomatis menampilkan halaman utama khusus admin	Sukses
2	Klik tombol Login pada halaman login form	Sistem akan menampilkan halaman utama khusus admin untuk input database system	Sistem langsung menampilkan halaman utama khusus admin	Sukses
3	Klik tombol Logout pada halaman utama khusus admin	Sistem akan mengembalikan tampilan ke halaman login	Sistem langsung kembali ke halaman login	Sukses

3. Pengujian pada halaman input data perusahaan otobus.

Tabel 3. (Pengujian Halaman Input Data Perusahaan Otobus)

No	Input	Skenario	Hasil Pengujian	Keterangan
1	Memasukkan data berupa nama perusahaan otobus, alamat, dan nomor telepon	Sistem akan menyimpan data perusahaan otobus secara langsung pada halaman utama khusus admin	Sistem otomatis menampilkan tabel data perusahaan otobus yang telah disimpan oleh user	Sukses
2	Klik keberangkatan perjalanan bus	Sistem akan menampilkan halaman berupa form untuk input data berangkat bus	Sistem langsung menampilkan halaman berupa form untuk input data berangkat bus	Sukses
3	Memasukkan data keberangkatan berupa nomor kendaraan, waktu berangkat, tujuan, dan tarif.	Sistem akan menyimpan data keberangkatan bus secara langsung pada halaman utama khusus admin	Sistem otomatis menampilkan tabel data keberangkatan bus yang telah disimpan oleh user	Sukses
4	Klik tiba perjalanan bus	Sistem akan menampilkan halaman berupa form untuk input data bus yang tiba	Sistem langsung menampilkan halaman berupa form untuk input data bus yang tiba	Sukses
5	Memasukkan data bus yang tiba berupa nomor kendaraan, waktu tiba, asal terminal dan tarif.	Sistem akan menyimpan data bus yang tiba secara langsung pada halaman utama khusus admin	Sistem otomatis menampilkan tabel data tibanya bus yang telah disimpan oleh user	Sukses
6	Klik tombol simpan atau batal pada halaman data berangkat maupun tiba perjalanan bus	Sistem akan menyimpan atau membatalkan data berangkat maupun tiba perjalanan bus	Sistem langsung menyimpan atau membatalkan perubahan data dengan menampilkan alert box	Sukses
7	Klik tombol edit data perusahaan otobus	Sistem akan menampilkan form untuk edit data perusahaan otobus dan langsung tersimpan	Sistem langsung menampilkan form edit data perusahaan otobus dan langsung menyimpannya	Sukses
8	Klik tombol untuk menghapus data perusahaan otobus	Sistem akan menampilkan alert box untuk menghapus data tersebut	Sistem menampilkan alert box untuk menghapus atau membatalkannya	Sukses

4. Pengujian pada halaman jadwal keberangkatan bus.

Tabel 4. (Pengujian Halaman Jadwal Keberangkatan Bus)

No	Input	Skenario	Hasil Pengujian	Keterangan
1	Klik tombol untuk edit data perjalanan bus	Sistem akan menampilkan form untuk edit data perjalanan bus yang berangkat dan langsung tersimpan	Sistem langsung menampilkan form edit data perjalanan bus yang berangkat dan langsung menyimpannya	Sukses
2	Klik tombol untuk input jumlah penumpang bus yang berangkat dari terminal asal	Sistem akan menampilkan form untuk input jumlah penumpang bus yang berangkat	Sistem langsung menampilkan halaman berupa form untuk input jumlah penumpang yang berangkat dan langsung disimpan oleh user	Sukses
3	Memasukkan dan menyimpan data keberangkatan berupa tanggal keberangkatan dan jumlah penumpang yang berangkat	Sistem akan menyimpan data keberangkatan bus secara langsung pada halaman utama khusus admin	Sistem otomatis menampilkan tabel data keberangkatan bus yang telah disimpan oleh user	Sukses

5. Pengujian pada halaman jadwal kedatangan bus.

Tabel 5. (Pengujian Halaman Jadwal Kedatangan Bus)

No	Input	Skenario	Hasil Pengujian	Keterangan
1	Klik tombol untuk edit data perjalanan bus	Sistem akan menampilkan form untuk edit data perusahaan otobus dan langsung tersimpan	Sistem langsung menampilkan form edit data perusahaan otobus dan langsung menyimpannya	Sukses
2	Klik tombol untuk input jumlah penumpang bus yang datang di terminal tujuan	Sistem akan menampilkan form untuk input jumlah penumpang bus yang datang di terminal tujuan	Sistem langsung menampilkan halaman berupa form untuk input jumlah penumpang yang datang dan langsung disimpan oleh user	Sukses
3	Memasukkan dan menyimpan data kedatangan penumpang berupa tanggal kedatangan dan jumlah penumpang yang tiba	Sistem akan menyimpan data kedatangan penumpang bus secara langsung pada halaman utama khusus admin	Sistem otomatis menampilkan tabel data jumlah kedatangan penumpang bus yang telah disimpan oleh user	Sukses

7. Evaluasi Akhir

Pada penelitian ini, tahap evaluasi akhir menggunakan kuesioner dengan *google form* dan percobaan aplikasi terminal berbasis *website* kepada pemilik *website* ini. Sehingga dapat disimpulkan, bahwa fitur-fitur pada aplikasi *website* ini dapat dijalankan 100% dengan baik dan sukses. Namun, aplikasi ini perlu adanya migrasi kepada *domain* untuk dapat diakses oleh umum khususnya pemilik *website* di seluruh *device* dan dapat meningkatkan performa dari aplikasi terminal guna sebagai data pelaporan mengenai jadwal perjalanan serta jumlah penumpang bus di beberapa terminal yang dikelola oleh Unit Pelaksana Teknis Daerah pada Dinas Perhubungan Provinsi Jawa Barat.

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan adalah hasil pengembangan aplikasi terminal berbasis *website* milik Dinas Perhubungan

Provinsi Jawa Barat menggunakan *framework* Laravel dan metode *Waterfall* secara lengkap meliputi pada tahap awal pengembangan dilakukan dengan cara pengumpulan data informasi dari seorang pegawai. Data tersebut berupa observasi wawancara dengan metode kualitatif dengan melampirkan pernyataan *Expert Judgement* mengenai kesalahan dan kekurangan *source code* pada *website* milik Dinas Perhubungan Provinsi Jawa Barat.

Setelah itu, hasil observasi dianalisis dalam bentuk UML *Diagram* dan *Entity Relationship Diagram* (ERD) secara notasi. Hasilnya didapatkan informasi yang lengkap mengenai spesifikasi kebutuhan pengguna akan perangkat lunak yang akan dikembangkan sesuai konsep *People*, *Process*, dan *Technology*.

Selanjutnya, pada tahap kedua pengembangan dilakukan dengan melakukan desain beberapa fitur atau tampilan pada *website* dalam bentuk *Mockup*. Tahap ketiga, pengembangan dilakukan dengan pemrograman terhadap fungsionalitas hasil *source code* yang belum memenuhi kriteria yang diinginkan serta pengujian secara ulang. Tahap keempat pengembangan adalah proses pengujian *website* terminal secara keseluruhan. Tahap kelima pengembangan sebagai tahap terakhir adalah evaluasi akhir dengan tingkat keberhasilan pengujian sebesar 100% menggunakan *Blackbox Testing* dan *User Acceptance Test*, yang sebelumnya telah dilakukan *Expert Judgement* kepada pemilik *website* ini.

Adapun dalam pengembangan *website* terminal ini, dilakukan juga manajemen *database*. Manajemen tersebut dilakukan dengan *import database* dengan data lain menggunakan *phpMyAdmin*. Sehingga, *database* yang sudah ada dapat terhubung dengan data lain. Selain itu, demi terjadinya peningkatan performa pada sistem *database*, diperlukan adanya kerjasama dengan petugas-petugas di terminal.

Referensi

- [1] Alatas, H. (2013), *Responsive Web Design dengan PHP dan Bootstrap*. Yogyakarta: Lokomedia.

- [2] Aldarmi, Saud A. 1998. *Real-Time Database Systems: Concepts and Design*. <http://www.homepage.cs.uri.edu/courses/attic/sc436/aldarmi98realtime.pdf>, 2007 [Jun. 2021]
- [3] Cholifah, Wahyu Nur., Yulianingsih, Sagita. Sri Melati. “*Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android Dengan Teknologi Phonegap*”. Jurnal String, 2018: Vol. 3, No.2.
- [4] Jendela Ilmu Informatika. “CSS. Internet: <http://aeccit.blogspot.com/2016/06/konsep-dasar-css.html>, 2016 [Jun. 3, 2021]
- [5] G. Adytiasari, “*Terminal Bus Tipe B Kabupaten Wonogiri*,” E-Journal Undip, vol. 3, no. 1, pp. 1–100, 2011.
- [6] Rizky Hybrid Blog. “HTML. Internet: <https://rizkyramadhansttg.wordpress.com/2008/07/15/dasar-pemrograman.html/>, 2008 [Jun. 23, 2021]
- [7] Siti's Blog. “HTML. Internet: <https://mbakshittek.files.wordpress.com/2010/01/makalah-html-sitek.pdf>, 2010 [Jun. 23, 2021]
- [8] Media Neliti. “HTML. Internet: <https://media.neliti.com/media/publications/219183-analisa-dan-desain-sistem-bimbingan-tuga.pdf>, 2005 [Jun. 3, 2021]
- [9] Husni. 2007. *Pemrograman Database Berbasis Web*. Yogyakarta: Graha Ilmu.
- [10] Hostinger. “JavaScript. Internet: <https://www.hostinger.co.id/tutorial/apa-itu-javascript>, 2021 [Jun. 3, 2021]
- [11] Kadir, A. 2008. *Belajar Database Menggunakan MySQL*. Yogyakarta: Andi.
- [12] Komarudin, M.K.M.Z. *Pengujian Perangkat Lunak Metode Black Box Berbasis Equivalence Partitions Pada Aplikasi Sistem Informasi Sekolah*. Jurnal Mikrotik, 2016; 6(3).
- [13] Kurniawan, Rulianto. (2010). *PHP dan MySQL untuk orang awam*. Palembang: Maxikom.
- [14] School of Computer Science - Binus University. “Laravel. Internet: <https://socs.binus.ac.id/2018/12/13/kelebihan-menggunakan-laravel-web-development/>, 2018 [Jun. 3, 2021]
- [15] Qwords. “Laravel. Internet: <https://qwords.com/blog/pengertian-framework-laravel/>, 2020 [Jun. 3, 2021]
- [16] Mustaqbal, M.S.M., Firdaus, R.F.F., dan Rahmadi, H.R. *Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus Aplikasi Prediksi Kelulusan SNMPTN)*. Jurnal Ilmiah Teknologi Terapan (JITTER), 2015; 1(3):31-36.
- [17] Candrajulianta. “MySQL. Internet: <http://candrajulianta.blogspot.com/2014/06/makalah-pengenalan-sql-dan-mysql.html>, 2014 [Jun. 23, 2021]
- [18] Bloging Dunia Pembelajaran. “MySQL. Internet: <https://edihsuaedih.blogspot.com/2016/09/makalah-basis-data-mysql.html>, 2016 [Jun. 3, 2021]
- [19] EduChannel Indonesia. “MySQL. Internet: <http://educhannel.id/blog/artikel/pengertian-mysql.html>, 2016 [Jun. 15, 2021]
- [20] Naista, D. (2017). *Codeigniter Vs Laravel*. Yogyakarta: CV. Lokomedia.
- [21] Dewaweb. “phpMyAdmin. Internet: <https://www.dewaweb.com/blog/panduan-phpmyadmin-untuk-pemula/>, 2021 [Jun. 4, 2021]
- [22] Niagahoster. “Pengertian XAMPP Lengkap dengan Cara Menggunakannya. Internet: <https://www.niagahoster.co.id/blog/cara-menggunakan-xampp/?amp>, 2020 [Jun. 4, 2021]
- [23] Medium.com. “PHP Artisan. Internet: <https://medium.com/easyread/apa-itu-artisan-cli-pada-laravel-62a94232a29a>, 2019 [Jun. 15, 2021]
- [24] Pressman, R.S. 2002. *Rekayasa Perangkat Lunak: Pendekatan Praktisi (Buku Dua)*. Yogyakarta: Penerbit Andi.

[25] R. Cahyadi, A. Aman, and F. Hanum, "Model Penjadwalan Keberangkatan Bus Dengan Strategi Alternatif Deadheading: Studi Kasus Po Raya," *Mat. dan Apl.*, vol. 12, no. 2, pp. 1–14, 2018.

[26] Safitri, Venny Kurnia (2016). *Analisis Perbandingan Kinerja Query Database Management System (DBMS) Antara PostgreSQL dengan MySQL*. Institut Teknologi Padang.

[27] Sasmito, Ginanjar Wiro. 2017. *Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal*, page 8.

[28] Setiawan, D. (2017). *Buku Sakti Pemrograman Web: HTML, CSS, PHP, MySQL, dan Javascript*. Yogyakarta: Start Up.

[29] Shah, Mahamad Qaium. (2015). *Responsive Web Development Using The Twitter Bootstrap Framework*. TURKU UNIVERSITY.

[30] Sommerville, I. (2011). *Software Engineering 9th Edition*. Addison-Wesley.

[31] Reznetworks. "Syntax CSS. Internet: <https://reznetworks.blogspot.com/2016/04/pengenal-an-css.html>, 2016 [Jun. 24, 2021]

[32] PT. APPKEY. "XAMPP. Internet: <https://appkey.id/pembuatan-website/teknologi-web/xampp-adalah/>, 2020 [Jun. 26, 2021]

[33] Dosen Pendidikan. "XAMPP. Internet: www.dosenpendidikan.co.id/xampp-adalah, 2021 [Jun. 26, 2021]